

# Arista 7500 Switch Architecture (*'A day in the life of a packet'*)

Arista Networks' award-winning Arista 7500 series was introduced in April 2010 as a revolutionary switching platform, which maximized data center performance, efficiency and overall network reliability. It raised the bar for switching performance, being five times faster, one-tenth the power draw and one-half the footprint compared to other modular data center switches.

Just three years later, the introduction of the Arista 7500E series modules and fabric delivers a three-fold increase in density and performance, with no sacrifices on functionality, table sizes or buffering, with industry-leading 1152 x 10GbE, 288 x 40GbE or 96 x 100GbE in the same quarter-rack 11RU chassis.

This white paper provides an overview of the switch architecture of the Arista 7500E series linecard and fabric modules and the characteristics of packet forwarding in the Arista 7500 series.

## SWITCH OVERVIEW

The Arista 7500 Series is a family of modular switches available in both 4-slot and 8-slot form factors that support a range of linecard options.

At a system level, the Arista 7508 with 7500E linecard and fabric modules scales up to 1,152 x 10GbE, 288 x 40GbE or 96 x 100GbE in the same quarter-rack 11RU chassis providing industry-leading performance and density without compromising on features/functionality or investment protection.



**Figure 1:** 7504E and 7508E with up to 1,152 10G ports, 288 40G ports or 96 100G ports.

**Table 1:** Arista 7500 Series Key Port and Forwarding Metrics

Characteristic	Arista 7504	Arista 7508
Chassis Height (RU)	7 RU	11 RU
Linecard Module slots	4	8
Supervisor Module Slots	2	2
Maximum System Density 10GbE ports w/ 7500E modules	576	1,152
Maximum System Density 40GbE ports w/ 7500E modules	144	288
Maximum System Density 100GbE ports w/ 7500E modules	48	96
System Fabric Raw Capacity (Tbps) / Usable Capacity (Tbps)	17 Tbps 15 Tbps	34 Tbps 30 Tbps
Maximum forwarding throughput per Linecard (Tbps)	2.88 Tbps per Linecard (144 x 10G / 36 x 40GbE)	
Maximum forwarding throughput per System (Tbps)	11.52 Tbps	23.04 Tbps
Maximum packet forwarding rate per Linecard (pps)	1.8B pps per Linecard	
Maximum packet forwarding rate per System (pps)	7.2B pps	14.4B pps

## ARISTA 7504 AND ARISTA 7508 CHASSIS AND MID-PLANE



**Figure 2: Left: Front of Arista 7504 and Arista 7508 chassis. Right: Fan/Fabric modules for Arista 7504/7508.**

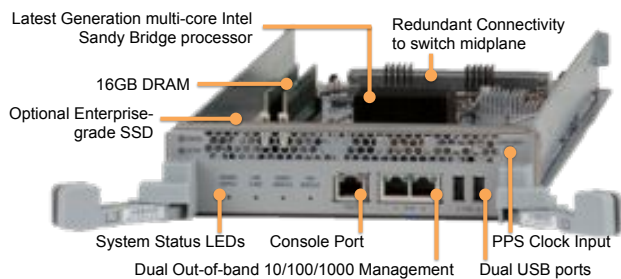
Both the 4-slot Arista 7504 and 8-slot Arista 7508 share common system architecture with identical fabric and forwarding capacity per slot. Linecards, Supervisors and power supplies are common across both chassis, the only differences are in the fabric/fan modules and number of linecard slots on each chassis. Airflow is always front-to-rear and all cabling (data and power) is at the front of the chassis.

Chassis design and layout is a key aspect that enables such high performance per slot: the fabric modules are directly behind linecard modules and oriented orthogonal to the linecard modules. This design alleviates the requirement to route high speed signal traces on the mid plane of the chassis, reducing the signal trace lengths and allowing more high speed signals to operate at faster speeds by being shorter lengths. This characteristic has also enabled Arista to scale the system from 10 Tbps with first generation modules in 2010 up to 30 Tbps in 2013 with second-generation 7500E series modules and is a big factor in how Arista has provided investment protection between first and second-generation modules.

Another benefit of not having high-speed traces on the mid plane is that it enables sections of the mid plane to be cut out providing a path for airflow to pass, supporting front-to-rear airflow without requiring large fans or large air intake/exhaust vents. This allows for a far more compact chassis to be built which in turn has less airflow restrictions which in turn allows smaller fans consuming less power to be utilized.

## SUPERVISOR-E MODULES

Supervisor modules on Arista 7500 Series switches are used for control-plane and management-plane functions only; all data-plane forwarding logic occurs on linecard modules and forwarding between linecard modules is always via the crossbar switch fabric modules.



**Figure 3:** Arista 7500 Supervisor-E Module.

Arista EOS®, the control-plane software for all Arista switches executes on multi-core x86 CPUs and in the case of the Supervisor-E, on a 4 core Intel Sandy Bridge Xeon. There is 16GB of DRAM for EOS, and as EOS runs on Linux and is extensible, the large RAM and fast multi-core CPUs provide headroom for running 3rd party software within the same Linux instance as EOS or within a guest virtual machine. An optional enterprise-grade SSD provides additional flash storage for logs, VM images or third party packages.

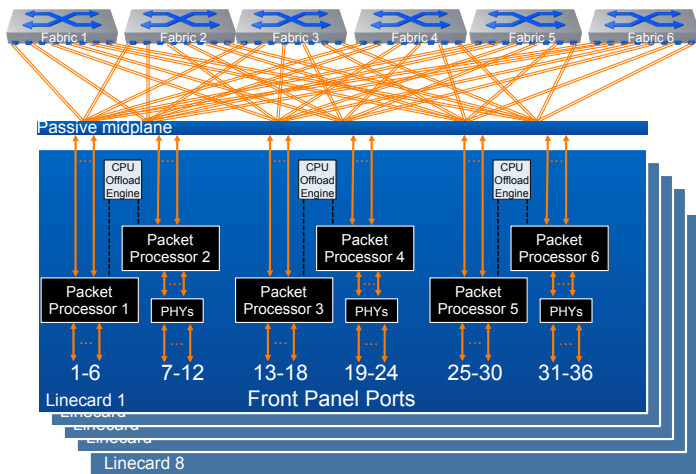
Out-of-band management is available via a serial console port and/or dual 10/100/1000 Ethernet interfaces. There are two USB2.0 interfaces that can be used for transferring images/logs or many other uses. A pulse-per-second clock input is provided for accurate clock synchronization. As an alternative, an accurate clock synchronization signal can be derived from the network interface on the first out-of-band 10/100/1000 Ethernet interface.

Supervisor-to-Supervisor and in-band connectivity between the Supervisor modules and data-plane forwarding on the linecard modules is provided by hot-swap PCI-Express (PCIe 3.0) point-to-point links. This provides a very efficient mechanism for the control-plane to update data-plane forwarding structures as well as an efficient mechanism for state transfer state between Supervisors.

## DISTRIBUTED PACKET FORWARDING IN THE ARISTA 7500 SERIES

### DISTRIBUTED DATA-PLANE FORWARDING

Both first and second-generation Arista 7500 Series linecard modules utilize packet processors on the linecard modules to provide distributed data-plane forwarding. Forwarding between ports on the same packet processor utilizes local switching and no fabric bandwidth is used. Forwarding across different packet processors uses all crossbar switch fabrics in a fully active/active mode.



**Figure 4:** Distributed Forwarding within an Arista 7500 Series

### CROSSBAR SWITCH FABRIC MODULES

Within the Arista 7500 Series up to six crossbar switch fabric modules are utilized in an active/active mode. Each crossbar switch fabric provides up to 320 Gbps fabric bandwidth full duplex (320 Gbps receive + 320 Gbps transmit). Packets are transmitted across the crossbar switch fabric as variable sized cells of up to 256 bytes

each (between 64 and 256 bytes) and all available crossbar switch paths are used active/active in order to reduce latency associated with serialization of larger frame sizes within the system and prevent hot spots associated with packet-based fabrics.

Besides data-plane packets, the crossbar switch fabric is also used for a number of other functions:

- **Virtual Output Queuing:** a distributed scheduling mechanism is used within the switch to ensure fairness for traffic flows contending for access to a congested output port. A credit request/grant loop is utilized and packets are queued in physical buffers on ingress packet processors within Virtual Output Queues (VoQs) until the egress packet scheduler issues a credit grant for a given input packet.
- **Distributed MAC learning:** when a new MAC address is learnt, moves or is aged out, the ingress packet processor with ownership of the MAC address will use capacity on the crossbar switch fabric to update other packet processors of the change.
- **Data-plane connectivity reachability packets:** all packet processors within the system send frequent periodic reachability messages to all other packet processors, validating the data-plane connectivity within the system.

Taking into account system headers on packets within the system, cell headers, VoQ scheduler credit request/grant messages, MAC learning packets and data-plane reachability messages, the six crossbar switch fabrics provide an aggregate of 3.84 Tbps usable bandwidth/slot. This is more than sufficient usable capacity not only for the up to 2.88 Tbps of forwarding capacity present on Arista 7500E Series linecards in a system with all fabric modules operational, but its also sufficient to still support full line-rate forwarding capacity per slot in the unlikely event of a failed fabric module.

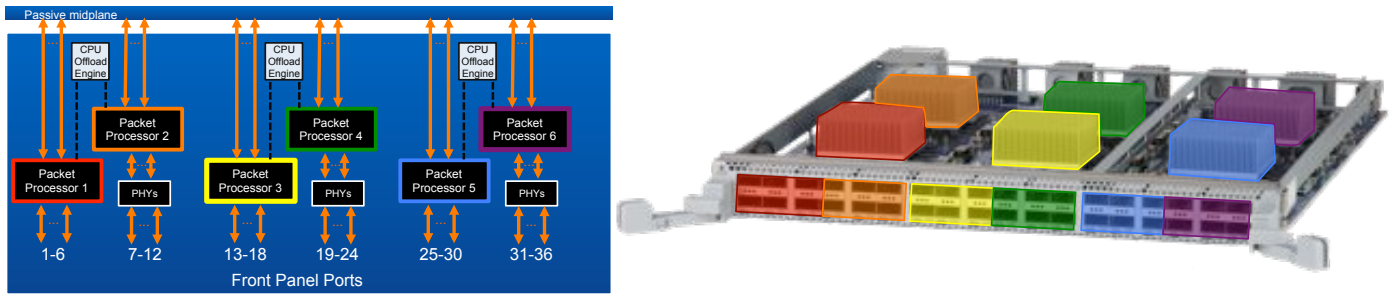
**Table 2:** Crossbar switch fabric performance characteristics

Fabric Parameters for second generation fabric modules	FE1600 (Fabric-E)
Fabric Link Speed	11.5 GHz
Fabric Links Active Per Linecard	192
Fabric Encoding Format	64b/66
Usable Capacity per Linecard slot (6 fabric modules active)	3.84 Tbps
Usable Capacity per Linecard slot (5 fabric modules active)	3.2 Tbps
Maximum Forwarding Throughput per 7500E Linecard	2.88 Tbps

## ARISTA 7500 SERIES LINECARD ARCHITECTURE

All stages associated with packet forwarding are performed in integrated system on chip (SoC) packet processors. A packet processor provides both the ingress and egress packet forwarding pipeline stages for packets that arrive or are destined to ports serviced by that packet processor. Each packet processor can perform local switching for traffic between ports on the same packet processor.

The architecture of a linecard (in this case, the 36 port QSFP+ module DCS-7500E-36Q-LC) is shown below in Figure 5. Each of the six packet processors on the linecard services a group of 6 x 40G QSFP+ front panel ports, and is highlighted in a different color.



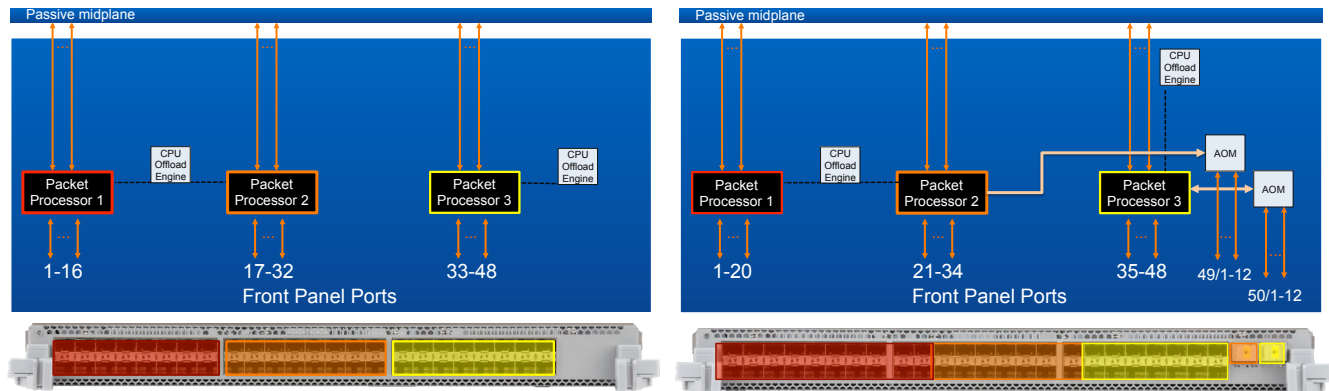
**Figure 5:** Arista DCS-7500E-36Q-LC Linecard Module architecture (**left:** logical linecard diagram, **right:** actual physical layout)

In addition to the packet processors on each linecard there are a number of other key elements:

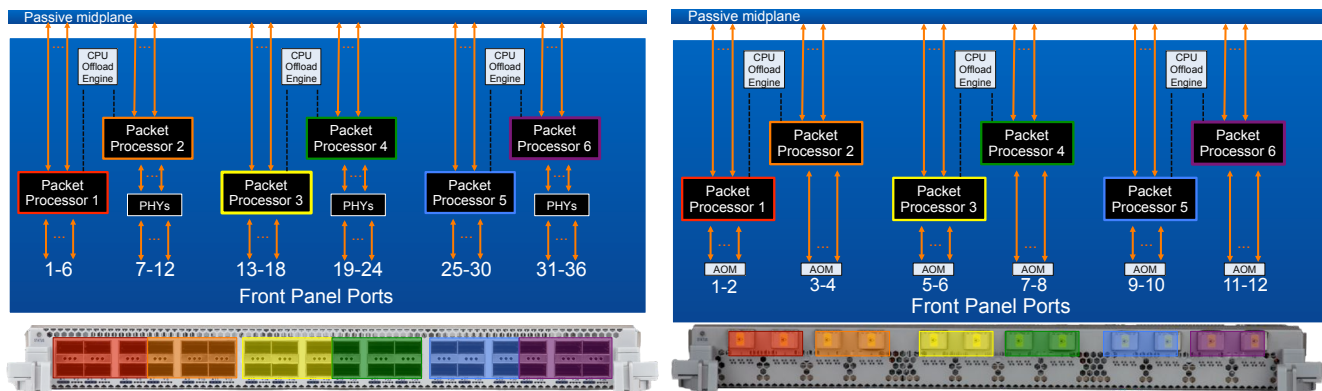
- **CPU Offload Engines** present on each linecard service up to two packet processors. These are used to accelerate the control-plane functions of programming forwarding tables, verifying the health of the system (validating memory tables, sending heartbeat / health-check packets looping within the system) as well as scaling the collection of statistics counters within the system.
- **PHYless front panel ports** further reduces what is already an incredibly low active component count within the system. Where trace lengths on the linecards allow, some front panel ports are driven directly from the packet processors. This improves reliability (with fewer active components), reduces power/heat and latency within the system. Arista pioneered this approach many years ago with other vendors now starting to copy this approach.

## ARISTA 7500E LINECARD LAYOUT

Arista 7500E linecard modules have either three or six packet processors depending on the number and type of ports on the module. Each of the Arista 7500E linecard modules is shown below in figures 6 and 7:



**Figure 6:** **Left:** Arista DCS-7500E-48S-LC Linecard Module, **Right:** Arista DCS-7500E-72S-LC Linecard Module



**Figure 7: Left:** Arista DCS-7500E-36Q-LC Linecard Module, **Right:** Arista DCS-7500E-12CM-LC Linecard Module

At a high level, the packet forwarding architecture of each of these linecard modules is essentially the same: a group of front-panel ports (different transceiver/port/speed options) connected to a packet processor with connections to the crossbar switch fabric modules. Where each of these linecards differs is in the combination of transceiver and port types (shown in table 3):

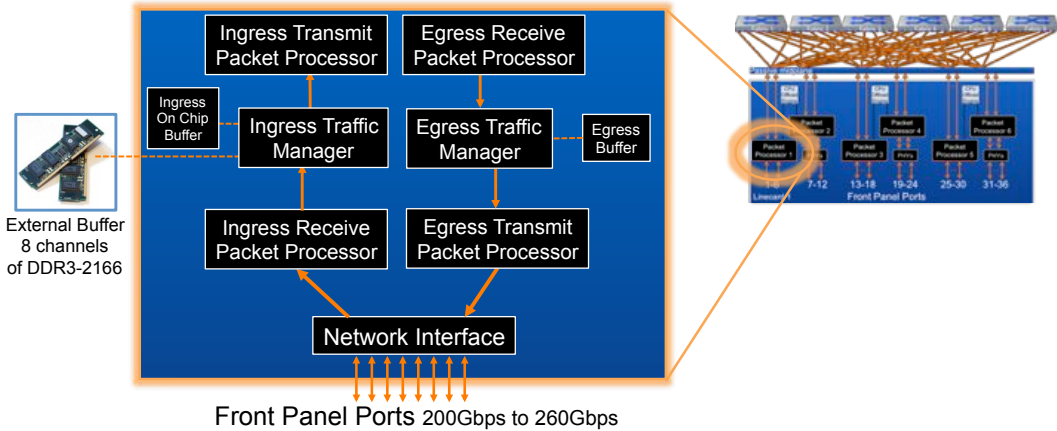
**Table 3:** Arista 7500E Series Linecard Module Port Characteristics

Linecard Port Characteristics	DCS-7500E-48S-LC	DCS-7500E-72S-LC	DCS-7500E-36Q-LC	DCS-7500E-12CM-LC
SFP+/SFP Transceiver Ports	48	48	-	-
QSFP+ Transceiver Ports	-	-	36	-
MXP (Multi-Speed) Ports	-	2	-	12
Maximum Number of 10G ports per Linecard	48 (48x SFP+)	72 (48 via SFP+, 24 via MXP)	144 (QSFP+ 4x10G breakout)	144 (12 x 12x10G MXP breakout)
Maximum Number of 40G ports per Linecard	-	6 (2 x 3x40G MXP breakout)	36 (36 x QSFP+)	36 (12 x 3x40G MXP breakout)
Maximum Number of 100G ports per Linecard	-	2 (2 x 100G MXP)	-	12 (12 x 100G MXP)
Maximum Number of 1G ports per Linecard	48 (48 x SFP)	48 (48 x SFP)	-	-

The Arista DCS-7500E-12CM-LC offers the most interface speed flexibility, highest system port density and an optimal price point. It has 12 x MPO/MTP ports with Arista Multi-speed Ports (MXP) based on embedded optics. This not only provides the industry’s highest 100GbE port density (12 x 100GBASE-SR10 ports) on a single linecard but also the industry’s first port that enables triple-speed operation of either 1x100GbE, 3 x 40GBASE-SR4, 12 x 10GBASE-SR and a unique linear price/port scale from 10G to 40G to 100G.

Combined, the Arista 7500E Series linecard module options provide the highest flexibility of transceiver types and optics, maximize system port density and optimize price points within the switch.

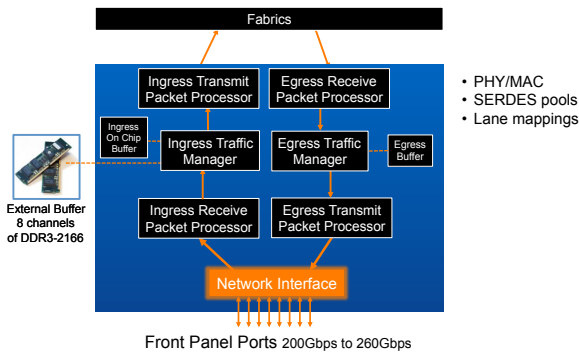
# ARISTA 7500E SERIES PACKET FORWARDING PIPELINE



**Figure 8:** Packet forwarding pipeline stages inside a packet processor on an Arista 7500E linecard module

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

## STAGE 1: NETWORK INTERFACE (INGRESS)



**Figure 9:** Packet Processor stage 1 (ingress): Network Interface

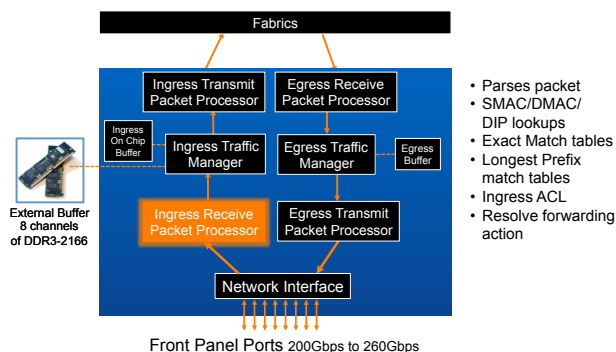
When packets/frames enter the switch, the first block they arrive at is the Network Interface stage. This is responsible for implementing the Physical Layer (PHY) interface and Ethernet Media Access Control (MAC) layer on the switch.

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 Ethernet interface is configured. Operation of the PHY for Ethernet 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.

As the Arista 7500E Series linecard modules provide flexibility in terms of multi-speed ports (e.g. an Arista MXP port can operate as 1x100G, 3x40G, 12x10G or combinations) the programmable lane mappings are also setup appropriately to map to the speed and type of interface configured.

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 RECEIVE PACKET PROCESSOR



**Figure 10:** Packet Processor stage 2 (ingress): Ingress Receive Packet Processor

The Ingress Receive Packet Processor stage is responsible for making forwarding decisions within the switch.

The first step is to parse the headers of the packet and extract all the key fields for forwarding (e.g. identify and parse the L2 Source MAC address, Destination MAC address, VLAN headers, Source IP Address, Destination IP Address, port numbers etc.) The packet parser is flexible and is not fixed logic. It is extensible to support future protocols and new methods of forwarding.

Following this, the switch needs to **determine whether forwarding should be at layer 2 (bridging) or layer 3 (routing)**. This is achieved by comparing the layer 2 frame Destination MAC (DMAC) address to the switch's MAC address for that interface. If it does match the layer 3 (routing) forwarding pipeline actions are used, otherwise layer 2 (bridging) forwarding pipeline actions will be used.

In the **layer 2 (bridging)** case, the switch performs a DMAC lookup in the MAC table for the VLAN and if present then knows what port to send the frame to and which packet processor to send the frame through to. If the DMAC lookup fails (device is not present in this VLAN) then the frame will be flooded to all ports within the VLAN, subject to storm-control thresholds for the port.

In the **layer 3 (routing)** case, the switch performs a lookup on the Destination IP address (DIP) within the VRF and if there is a match it knows what port to send the frame to and what packet processor it needs to send the frame through to. If the DIP matches a subnet local on this switch but there is no host route entry, the switch will initiate an ARP request to glean the MAC address for where to send the packet. If there is no matching entry at all the packet is dropped. IP TTL decrement also occurs as part of this stage.

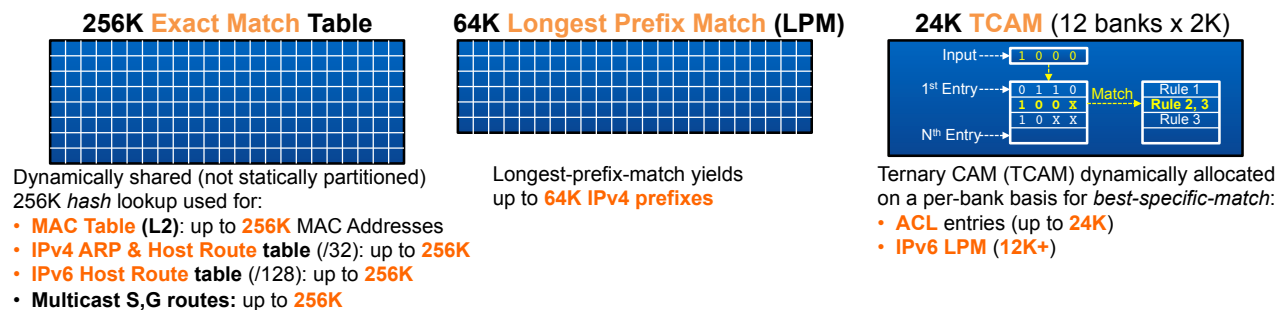
The primary reason for determining early on whether the forwarding action is bridging or routing is to enable optimization of memory tables within the packet processor. Both the MAC table (layer 2) and IP Host route entries (layer 3) are stored in the same **256K Exact Match Table**, which is a large hash entry table lookup. Utilizing the same memory table for both these resources without requiring it to be statically partitioned between resources provides significant flexibility in the size and scale of network designs where the switch can be deployed.

Layer 3 forwarding is performed using three hardware tables:

1. A hash lookup is performed in the **256K Exact Match Table** for the IP Host Route entries (IPv4 /32, IPv6 /128) and Multicast S,G entries



- In parallel, for IPv4 packets there is a lookup in the **Longest Prefix Match (LPM)** Table for the routing prefix
- In parallel, for IPv6 packets there is a lookup in the **TCAM** for the IPv6 routing prefix



**Figure 11:** Hardware tables within a packet processor associated with forwarding on Arista 7500E linecard modules

The best match from these three resources combined provides the layer 3 forwarding lookup result which either points to an adjacency entry or an adjacency group (if the route entry exists). If no match exists then the packet is dropped. An adjacency group entry match means there are multiple next-hop entries to choose from (Link Aggregation at layer 2 or Equal Cost Multi Pathing at layer 3). Whatever fields are configured for L2 or L3 load balancing are hashed to provide an index into the group and derive a single matching entry. The final matching adjacency entry provides details on where to send the packet (egress packet processor, output interface and a pointer to the output encapsulation/MAC rewrite on the egress packet processor),

For **multicast traffic**, the logic is almost the same except that the adjacency entry provides a Multicast ID. This indexes into a 64K table indicating output interfaces for the multicast stream. This provides both local (ingress) multicast expansion (for multicast traffic with groups on local ports) as well as information on whether to send the packet towards the crossbar switch fabric (which in turn will replicate the packet to all egress packet processors that have subscribers on the multicast group.)

In addition to forwarding lookups based on DMAC/DIP, this block can also perform a lookup based on the Source MAC (SMAC) or Source IP (SIP) address:

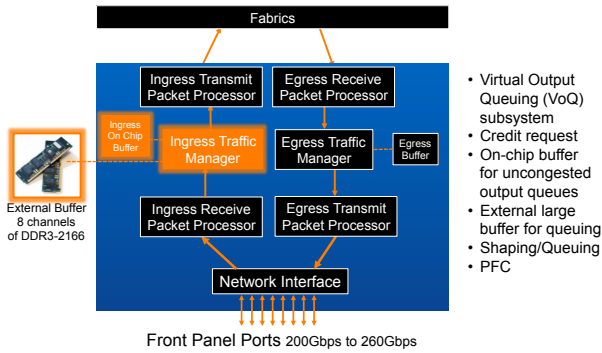
- For Layer 2, a lookup also happens on the Source MAC address. This action is performed to make sure the switch knows about the sending device on the port. If it is unknown (lookup miss) then **hardware MAC learning** will install the entry in the table and trigger other forwarding engines to learn about this device.
- For Layer 3, if **Unicast Reverse Path Filtering (uRPF)** is enabled, a L3 forwarding lookup on the Source IP is also performed, with appropriate actions taken based on the uRPF being in loose or strict mode.

In parallel with forwarding table lookups, the TCAM is also used for performing Ingress ACL lookups (Port ACLs, Routed ACLs) and based on matches applying actions that are configured.

The packet forwarding pipeline always remains in the hardware data-plane. There are no features that can be enabled that cause the packet forwarding to drop out of the silicon (hardware) data-plane forwarding path. In cases where software assistance is required (e.g. traffic destined within a L3 subnet but for which the switch has not yet seen the end device provide an ARP and doesn't have the L3-to-L2 glue entry), **hardware rate limiters** and **Control-plane Policing** are employed to protect the control-plane from potential denial of service attacks.

[Note: Table sizes for various resources (IPv4 prefixes, IPv6 prefixes) are indicated as a range because maximum size is dependent on a variety of factors. Please contact your Arista technical representative if you require more detailed information.]

### STAGE 3: INGRESS TRAFFIC MANAGER



**Figure 12:** Packet Processor stage 3 (ingress): Ingress Traffic Manager

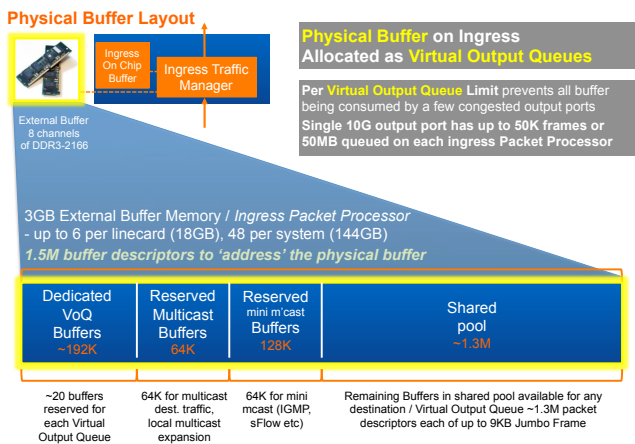
The Ingress Traffic Manager stage is responsible for packet queuing and scheduling within the switch.

Arista 7500 Series switches utilize Virtual Output Queuing (VoQ) where the majority of the buffering within the switch is on the input linecard. While the physical buffer is on the input, it represents packets queued on the output side (called virtual output queuing). VoQ is a technique that allows buffers to be balanced across sources contending for a congested output port and ensures fairness and QoS policies can be implemented in a distributed forwarding system.

When a packet arrives into the Ingress Traffic Manager, a VoQ credit request is forwarded to the egress port processor requesting a transmission slot on the output port. Packets are queued on ingress until such time as a VoQ grant message is returned (from the Egress Traffic Manager on the output port) indicating that the Ingress Traffic Manager can forward the frame to the egress packet processor.

While the VoQ request/grant credit loop is under way, the packet is queued in input buffers. On-Chip Buffer memory (2MB) is used for traffic destined to uncongested outputs (first packet queued to an egress VoQ) and Off-Chip Buffer memory (3GB) is used for packets not at the head of the queue. Large buffers are required to build a switch with sufficient buffering to handle both in-cast and microbursts traffic patterns that are typically seen in the spine layer of high performance scale-out networks.

External buffer memory is used because it's not feasible to build sufficiently large buffers "on-chip" due to transistor budget and semi-conductor manufacturing constraints on silicon die. To achieve the required performance and de-queue rates, there are 8 channels of DDR3-2166 memory for buffer storage on each packet processor. The large number of channels is to handle greater than 200Gbps and 300MPPS per packet processor.



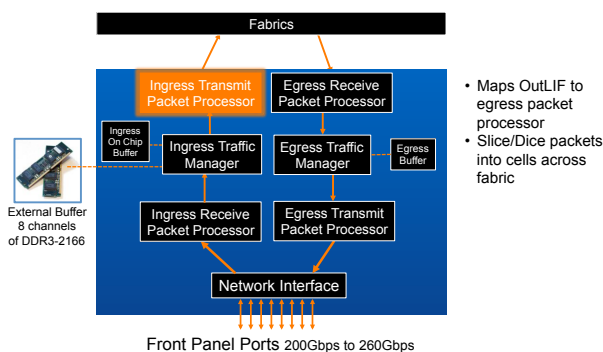
**Figure 13:** Physical Buffer on Ingress allocated as Virtual Output Queues

While there is a large amount of buffer memory available (2MB on-chip + 3GB off-chip per packet processor), this large a buffer allocated to a single output port would result in excessive queuing and associated high latency. Per-VoQ limits are applied to limit the maximum buffer queue depth available for a given output port queue. These are shown in table 4 below:

**Table 4:** Default per-VoQ Output Port Limits

Output Port Characteristics	Maximum Packet Queue Depth	Maximum Packet Buffer Depth (MB)	Maximum Packet Buffer Depth (msec)
VoQ for a 10G output port	50,000 packets	50 MB	5 msec
VoQ for a 40G output port	200,000 packets	200 MB	5 msec
VoQ for a 100G output port	500,000 packets	500 MB	5 msec

### STAGE 4: INGRESS TRANSMIT PACKET PROCESSOR



**Figure 14:** Packet Processor stage 4 (ingress): Ingress Transmit Packet Processor

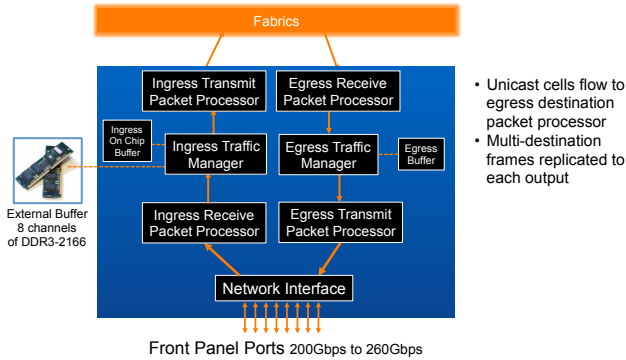
The Ingress Transmit Packet Processor stage is responsible for slicing packets into variable-sized cells and transmitting the cells across all available crossbar switch fabric paths. The health-tracer subsystem is continually checking the reachability between all paths and uses all available paths in an active/active manner.

Packets are sliced into variable-sized cells of up to 256 bytes and are transferred on up to 32 fabric links simultaneously. While the crossbar switch fabric is store-and-forward, this parallel-spray mechanism reduces serialization delay to at most 256 bytes. There are also no hot spots as every flow is always evenly balanced across all fabric paths.

Each cell has a header added to the front for the receiving packet processor to be able to reassemble and maintain in-order delivery. Forward Error Correction (FEC) is also enabled for traffic across the crossbar switch fabrics, both to correct errors (if they occur) but also to help monitor data-plane components of the system for any problems.

Packets destined to ports on the same packet processor can be switched locally at this stage (Ingress Transmit Packet processor to Egress Receive Packet Processor) and don't consume any fabric bandwidth resources.

## STAGE 5: CROSSBAR SWITCH FABRICS



**Figure 15:** Crossbar switch fabrics

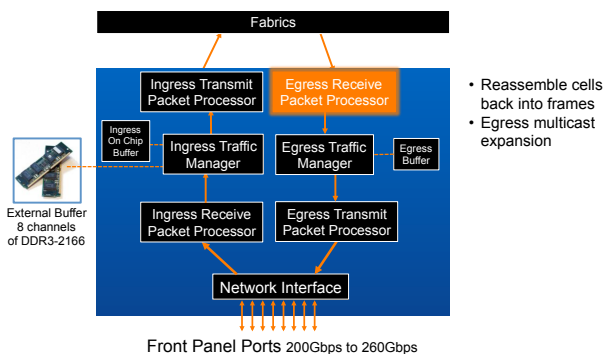
There are 6 crossbar switch fabrics in the rear of an Arista 7500 Series switch all operating in an active/active manner. These provide connectivity between all data-plane forwarding packet processors inside the system.

The crossbar switch fabrics forward based on cell headers which in the case of unicast traffic is a 7 bit address indicating the output packet processor to send the cell to.

For multicast traffic, there is multicast expansion inside the fabric. A 16-bit Multicast ID within the cell header is used as an index into a 64K multicast group table that uses a bitmap to indicate which packet processors should receive replication copies of the cell. Note that if there are multiple multicast receivers on an output packet processor, there is only one copy delivered per output packet processor as there is optimal egress multicast expansion inside the system.

Control-plane software maintains the multicast group table based on the fan-out of multicast groups across the system. IP multicast groups that share a common set of output packet processors reuse the same fabric Multicast ID.

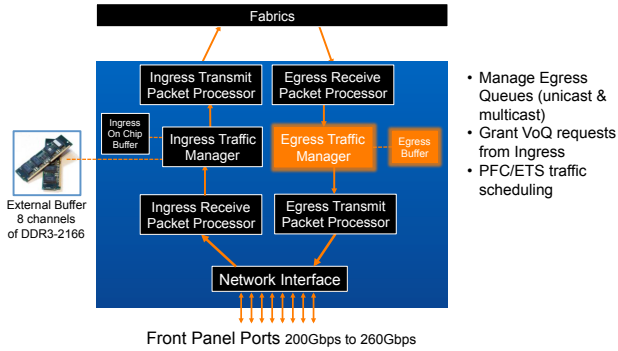
## STAGE 6: EGRESS RECEIVE PACKET PROCESSOR



**Figure 16:** Packet processor stage 6 (egress): Egress Receive Packet Processor

The Egress Receive Packet Processor stage is responsible for reassembling cells back into packets/frames. This is also the stage that takes a multicast packets/frame and replicates it into multiple output packets/frames if there are multiple receivers on this output packet processor.

## STAGE 7: EGRESS TRAFFIC MANAGER



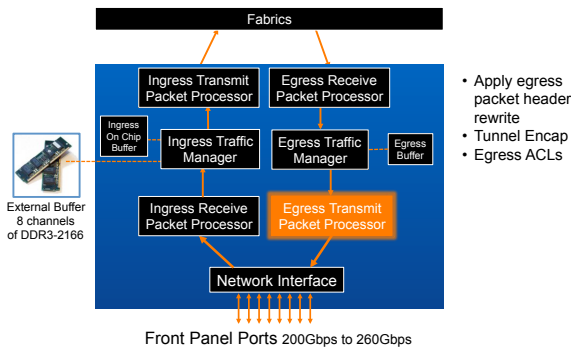
**Figure 17:** Packet processor stage 7 (egress): Egress Receive Packet Processor

The Egress Traffic Manager stage is responsible for the granting of VoQ credit requests from input packet processors and managing egress queues.

When an ingress packet processor requests to schedule a packet to the egress packet processor it is the Egress Traffic Manager stage that receives the request. If the output port is not congested then it will grant the request immediately. If there is congestion it will service requests in a fair manner between contending input ports, within the constraints of QoS configuration policy (e.g. output port shaping) while also conforming to PFC/ETS traffic scheduling policies on the output port.

The Egress Traffic Manager stage is also responsible for managing egress buffering within the system. There is an additional 3MB on-chip buffer used for egress queuing which is allocated as 32K buffers each of 256 bytes size. This buffer is mostly reserved for multicast traffic as unicast traffic has a minimal requirement for egress buffering due to the large ingress VoQ buffer.

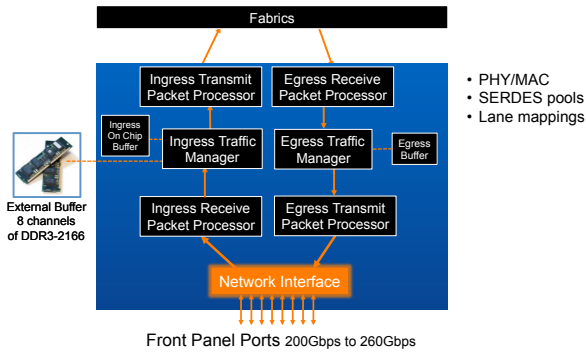
## STAGE 8: EGRESS TRANSMIT PACKET PROCESSOR



**Figure 18:** Packet processor stage 8 (egress): Egress Transmit Packet Processor

The Egress Transmit Packet Processor stage provides packet header rewrites such as putting the next-hop DMAC address in to a routed packet, enforcing any egress ACLs and performing any tunnel encapsulation operations. Any packet header rewrite instructions have been passed from the Ingress Receive Packet Processor stage instructing this stage on what fields to rewrite based on tables associated with the output port.

## STAGE 9: NETWORK INTERFACE (EGRESS)



**Figure 19:** Packet Processor stage 9 (egress): Network Interface

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 20:** 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.

## **INCREASING SYSTEM PERFORMANCE WITH CPU OFFLOAD ENGINES AND DATA-PLANE ACCELERATION**

To scale system performance in a distributed forwarding platform like the Arista 7500 Series, many control-plane components have been implemented with scale in mind. Some examples of this include:

- All software agents that program data-plane are resilient to high update rates and table churn and make extensive use of coalescing/deferring techniques.
- Data-plane table updates are performed in CPU Offload Engines present on each linecard. These monitor changes in the form of update logs in a shared memory space within the Supervisor across PCI-Express links and update the forwarding tables locally on the linecard modules. The result is that control-plane performance doesn't decrease as linecards are added to a system.
- Common steady-state CPU overhead tasks like polling hardware counters, PHYs/interface status etc., are offloaded locally to CPU Offload Engines on each linecard. Counters are updated frequently in a batch DMA transfer to shared memory spare on the Supervisor control-plane.
- Data-plane table updates like MAC learning, moves and aging is provided by the data-plane forwarding without software involvement. Installing a new linecard module in a system doesn't require software to program its MAC table; control-plane software instructs other packet processors in the system to update newly-instantiated linecard packet processors via a dump of the contents of each of their tables.

## **SYSTEM HEALTH TRACER AND INTEGRITY CHECKS**

Just as significant engineering effort has been invested in the software architecture of Arista EOS, the same level of detail has gone into system health and integrity checks within the system. There are numerous subsystems on Arista 7500 Series switches that validate and track the system health and integrity on a continual basis, which includes:

- All memories where code executes (control-plane and data-plane) are ECC protected; single bit errors are detected and corrected automatically, double bit errors are detected.
- All data-plane forwarding tables are parity protected with shadow copies kept in ECC protected memory on the control-plane. Continual hardware table validation verifies that the hardware tables are valid and truthful.
- All data-plane packet buffers are protected using CRC32 checking from the time a packet/frame arrives to the time it leaves the switch. The CRC32 is validated at multiple points through the packet forwarding pipeline to ensure no corruption has happened, or if there has been a problem it can be isolated.
- Forward Error Correction (FEC) is also utilized for traffic across the crossbar switch fabrics, both to correct errors (if they occur) but also to help monitor data-plane components of the system for problems.
- Data-plane forwarding elements are continually testing and checking reachability with all other forwarding elements

## CONCLUSION

Designed for large virtualized data centers and cloud networks the Arista 7500 Series modular switches are the industry's highest performance data center switches. The Arista 7500E is the second generation of the 7500 Series and delivers seamless upgrades ensuring investment protection of first generation fabrics, linecards and common equipment while setting a new standard for performance, density, reliability, and power efficiency. The Arista 7500E Series offers over 30Tbps of total capacity for 1,152 ports of 10GbE or 288 ports of 40GbE and support for 96 ports of wire-speed 100GbE using integrated optics that support flexible combinations of 10G, 40G and 100G modes on a single interface.

With front-to-rear airflow, redundant and hot swappable supervisor, power, fabric and cooling modules the system is purpose built for data centers. The 7500E Series is energy efficient with typical power consumption of under 4 watts per port for a fully loaded chassis.

All Arista products including the 7500E Series runs 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 together with stateful switchover without the loss of data plane forwarding.

All of these attributes make the Arista 7500E an ideal platform for building reliable, low latency, resilient and highly scalable data center networks.





**Santa Clara—Corporate Headquarters**

5470 Great America Parkway

Santa Clara, CA 95054

Tel: 408-547-5500

[www.aristanetworks.com](http://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](http://www.aristanetworks.com).