

Direct Access Mode - DAM

Concepts and Applications Guide

Draft - v0.4 - 1/28/2001

Ewerton Vieira
Systems Engineer
Alteon WebSystems
evieira@alteon.com

Direct Access Mode, also called **DAM**, is a feature introduced at WebOS 5.2 that enables applications like Direct Access to real servers and Content Switching.

On the other hand, DAM can generate some undesired side effects.

Lack of understanding of how DAM works makes it difficult to do a good/safe design.

The following text tries to explain the concepts behind DAM and how to use it properly.

With the introduction of WebOS 8.0 and VMA things happen a little different now and many DAM limitations just vanished in the air. However, in order to try not to get the reader confused, we will first start with DAM and WebOS 6.0 and then we will show how things changed with WebOS 8.0 and VMA.

This text assumes the reader knows TCP/IP and also has a basic understanding of Alteon switches and WebOS concepts and operations.

DAM - Why?

The distributed processing architecture of our switches, one ASIC and two processors per physical port on the stackable switches, has lots of advantages and some limitations.

At **Client Processing**, when the switch replaces the VIP address by one of the RIPs, upon arrival of the first packet of a TCP session the switch creates an entry at the session table stored at that port. This is mainly used to enable the switch to send the next packets, belonging to that same TCP session, to the same real server. This intrasession persistency is vital for a L4 switch.

On the way back to the clients, the packets must have the source IP address replaced back from the RIP to the VIP, otherwise the clients will reject them. Alteon call this **Server Processing**. In most cases, this translation can be done very efficiently based on a static table called Service Mapping Table (SMT) that maps the pair (RIP,RPORT) to (VIP, VPORT). There is no need to look at the Session Table.

Under that distributed processing model (without VMA) our switches handle Client Processing at one port, where packets enter the switch to go to the VIP, and Server Processing at another port (where packets enter the switch to go back to the client).

This is ok for most of the situations. But it is not ok for a few. There are times when information gathered or stored at the ingress port must be retrieved to correctly do Server Processing. That is the case, for example, of direct access to load balanced servers.

In our architecture, in order to Server Processing to be able to use the information set by Client Processing we must have BOTH Client Processing and Server Processing happening in the same port so that they can share the same Session Table stored at the corresponding port's SRAM.

In WebOS 6.0, there are two ways of ensure Client and Server Processing at the same port: **DAM** or **PIP**. In WebOS 8.0 VMA takes care of that as we will see later.

With access to data assured by using the correct session table, there comes the other side (not less important) of DAM and PIP: they will also mean somewhat different, more complete, Client and Server Processing.

PIP, which we will not discuss here, however, has a very serious disadvantage: because PIP's Client Processing changes not only VIP->RIP but also CIP->PIP it impossible to gather good usage statistics from the web servers since all traffic will look as coming from the PIP.

On the other hand, PIP is the only way to assure that traffic will get back to the alteon switch to go through Server Processing in certain atypical topologies like one-armed load balancing.

DAM - Applications

Several applications need DAM. Some of them are listed bellow:

- Direct Access to Real Servers
- One Real Server serving Multiple VIPs
- Content Intelligent Switching (Layer 7 processing), like
 - ✓ URL based SLB
 - ✓ URL based WCR
 - ✓ URL based BWM
 - ✓ Cookie based persistence
 - ✓ Cookie based SLB
 - ✓ Cookie based BWM
 - ✓ SSL Session ID persistence

Later, under Applications Operation, we will study each of those applications in detail. And you will also find alternatives to DAM based on each scenario.

DAM - How to configure

To configure DAM is very simple. All you have to is:

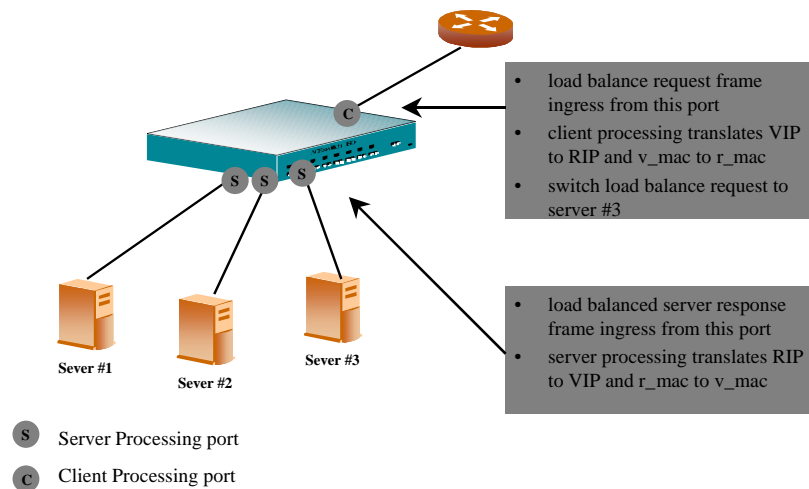
- At WebOS 5.2 and 6.0: Apply the command `/cfg/slb/direc en`
- At WebOS 8.x: Apply the command `/cfg/slb/adv/direct en`

Note: When DAM is enabled, under WebOS 6.0, the configuration of a port as "server ena" means nothing. SPrr will happen at the ports configured "client ena" when the packet exits the switch.

DAM Concepts

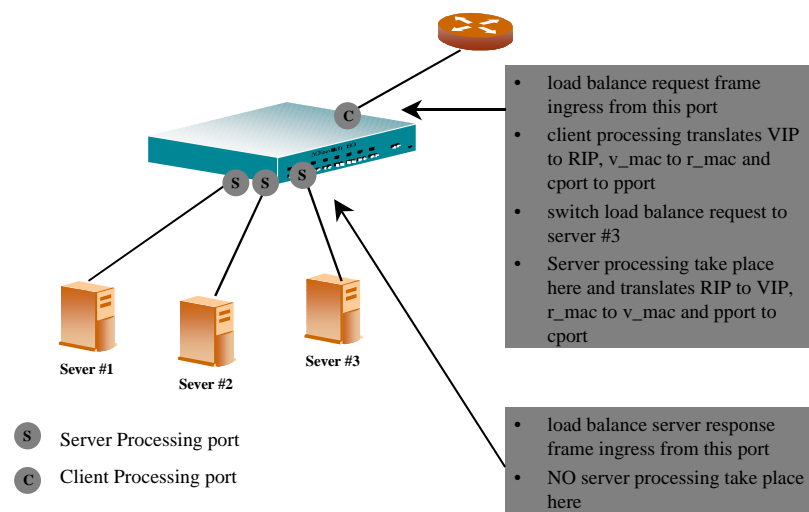
So the key point is that DAM enables Client and Server Processing in the same port so that Server Processing can handle atypical types of traffic like direct access to real servers.

See below the normal (DAM disabled) operation of CPr and SPPr.



Direct Access Mode Disabled

See below how load balancing happens if you enable DAM. Note that the source (cport) port is remapped in both directions.



Direct Access Mode Enabled

DAM - How it works

WebOS Internal Operations Concepts - The Session Table

Let's review some WebOS concepts that will help us understand DAM.

The **Session Table** is a data table stored in the SRAM of each ASIC. It holds information used by the switch to do its load balancing tasks.

Each line of the Session Table is called a **Session Table Entry**. The number of entries will determine the number of TCP sessions each ASIC can support. Because there is a part of the switch's memory reserved for the Session Table (varies according to the switch model), the size of the Session Table Entry will determine the number of entries.

A Session Table Entry has several fields. Some of them are always there. Some will be only be used according to the configuration of the switch (for instance if the switch is doing filtering, PIP or L7 processing).

Among those that are always there are:

- **CIP**, the source IP of the incoming packet from the client
- **CPORT**, the TCP source port of the incoming packet from the client
- **RSI**, the Real Server Index, a 0 byte field that tells which Real Server was assigned to that session

Among those that are sometimes there are:

- **RPORT**, the TCP destination port of the packet that is sent to the real server
- **VPORT**, the TCP destination port of the incoming packet from the client
- **VSrI**, the Virtual Server Index, a byte field that tells which Virtual Server is related to that session

The applications you run on the switch will determine the necessary fields on the Session Table. Currently WebOS supports 3 different formats for the Session Table. They are listed bellow with their space requirements per entry:

- **ABT** - Application Binding Table - 12 bytes
- **FBT** - Filter Binding Table - 16 bytes
- **PBT** - Proxy Binding Table - 24 bytes

With the ABT you can do SLB. If the switch will only support Filtering, in a FWLB config, for example, FBT is used. And if the switch needs to have PIPs or L7 processing, PBT will be used. SLB plus Filtering requires PBT.

The pair (CIP, CPORT) is called the **Session ID** because it identifies a unique TCP session. The session ID is the access key for the session table.

WebOS Internal Operations Concepts - Client and Server Processing

At **Client Processing (CPr)**, when the switch replaces the VIP by one of the RIPs, upon arrival of the first packet of a TCP session the switch creates an entry at the Session Table stored at that port. This is mainly used to enable the switch to send the next packets, belonging to that same TCP session, to the same real server.

On the way back to the clients, the packets must have the source IP address replaced back from the RIP to the VIP, otherwise the clients will reject them (will send a TCP reset). We call this **Server Processing (SPr)**.

In most cases, this translation can be done very efficiently based on a static table, derived from the switch's configuration, called **Service Mapping Table (SMT)**. SMT key (RIP,RPORT) maps to the corresponding pair (VIP, VPORT). When using SMT, there is no need to look at the Session Table to do SP_r. Let's call this mode **Express Server Processing (E-SP_r)**.

Client and Server Processing with DAM

When DAM is enabled the switch will use a more complex type of SP_r, which we will call **Advanced Server Processing (A-SP_r)**. It will also change slightly how CP_r will behave.

Mainly, **A-SP_r** will look at the Session Table for the corresponding entry and replace RIP by VIP based on the value (VSrI) found at the Session Table not based on the SMT (as done by E-SP_r).

In WebOS 6.x, DAM also forces SP_r to happen in the egress port instead of at the ingress port, so CP_r and SP_r can happen in the same port if traffic rules send the return packet back to the same port the original packet entered.

When traffic exits the switch by the same port it has entered the switch, both CP_r and SP_r can use the same Session Table. If DAM is enabled, A-SP_r will take over and we are able to deliver the intended benefits.

So, DAM works in two ways:

1. For the incoming packet: during CP_r, it replaces original packet's SPORT by a different port (actually the Session Table Entry number/index). It will also store the Virtual Server Index (VSrI) in that entry. This will allow A-SP_r to easily find the Session Table entry for the corresponding return packets.
2. For the returning packet: it will not go through E-SP_r at the incoming port but, instead, A-SP_r will happen at the outgoing port using the information stored at the Session Table.

DAM and Port Mapping

DAM and Port Mapping cannot be done when the switch is using the ABT format for the Session Table. The reason is that to be able to support Multiple VIPs per RIP, when using the ABT, CP_r stores the VSrI in the field that normally stores RPORT when DAM is disabled.

So if you need Port Mapping with DAM, you need to enforce the use of PBT format. All you need to do is to have either filtering (even with no filters applied), PIP or Layer 7 processing enabled because this will determine the use of PBT, that also stores VIP and VPORT of the incoming packet).

DAM and FWLB

For FWLB to work properly the addresses (SIP and DIP) of the packets must be the same (or exchange places) for hash to redirect to the same firewall. Load balanced incoming packets have SIP=CIP and DIP=VIP while the corresponding packets return with SIP=RIP and DIP=CIP. So we need SP_r to happen BEFORE the filter so that SIP is replaced from RIP back to VIP before the hash function is calculated.

In WebOS 6.0, DAM determines that SP_r be moved from the ingress port (where filter happens) to the egress port so it will happen AFTER the filtering stage. Under this condition, DAM, SLB and FWLB cannot be done at the same time.

In WebOS 8.0, although SPr remains being done before filtering, the only way we can guarantee that the packet is processed in the same port, using the same session table for CPr and A-SPr, is enabling VMA (Virtual Matrix Architecture).

DAM and WebOS 8.x

Why and how VMA saves the day? Very ingenious.

First we need to understand how VMA chooses the designated port: Every packet that enters a port that has L4 processing enabled (Client, Server or Filter processing) is inspected. If the source IP does NOT belong to a Real Server, the switch will use SIP as the key for the hash that designates the port at which that packet will be L4-processed. Otherwise, if it did come from a Real Server, the switch will use the DIP as the hash key.

This way VMA will assure that a Load Balanced stream of packets belonging to the same TCP session will always be have SPr at the same port as it had CPr. Plus it also assures a more uniform distribution of the processing because it will never use the IP of the Real Servers, a small set of IPs, that would consequently determine much more load in a few of the ports.

DAM - Caveats

WebOS 8.0 and VMA enabled

- Active-Active setups will not work with DAM enabled - One advantage of E-SPr is that no matter which Alteon switch gets the packet on its way back to the client in a High Availability setup (2 Alteon switches), SPr will work. If we enable DAM, the only way A-SPr will work is if the packet exits through the same switch it has entered.
- Port Mapping can only be done using the PBT session table format.

WebOS 8.0 and VMA disabled, WebOS 6.0 or 5.2

- All those with VMA enabled plus:
- Cannot have default gateway load balancing because traffic could exit through a different port than it entered the switch (if you have more than one entry/exit port).
- FWLB and SLB cannot be done because A-SPr will happen at a different port than CPr and hence will not be able to use the same Session Table

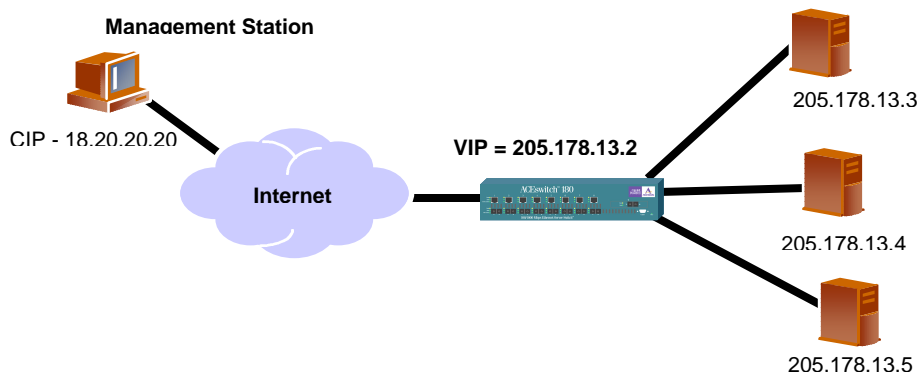
DAM - Applications Operation

Direct Access to Real Servers

It is very common that customers want to be able to monitor not only the operation of the Virtual Service (VIP+VPORT) but also each Real Service (RIP+RPORT) so that if one of the Real Services fail, they will know.

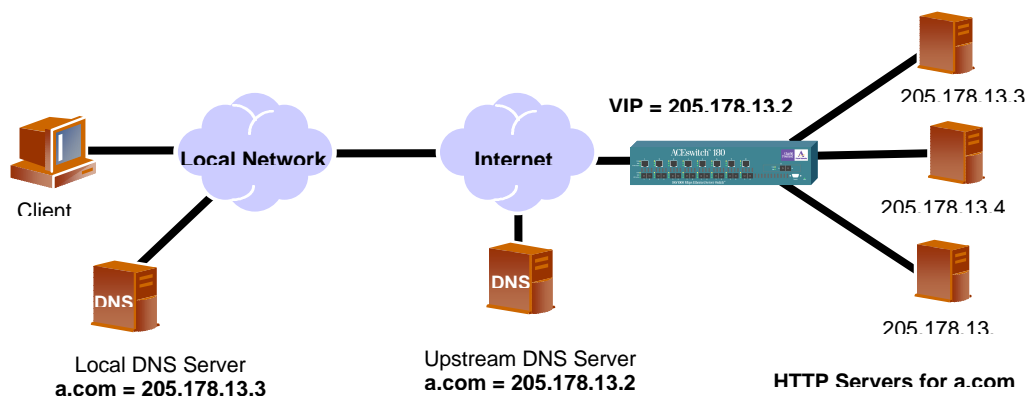
Using E-SPr, this cannot be done. You will need A-SPr.

The reason is that E-SPr cannot differentiate between the return packets corresponding to those that came to the VIP and suffered CPr from the packets that came directly to one of the RIPs. They have the same information (sip=RIPx, sport=RPORT), and hence will have RIP replaced by VIP on all packets. When the return packet corresponding to the one sent to RIPx returns having source address equals VIP, the TCP protocol on the client station will drop the packet.



When you enable DAM, during A-SPr the switch will look at the dport of the return packet and use that port number to look on the Session Table. If the packet belongs to the same session (dip=CIP) the switch will replace the VIP and dport (using CPORT and VSrI from the Session Table). If the switch does not find a match it will not do SPr (letting it to follow its normal flow at L2 or L3).

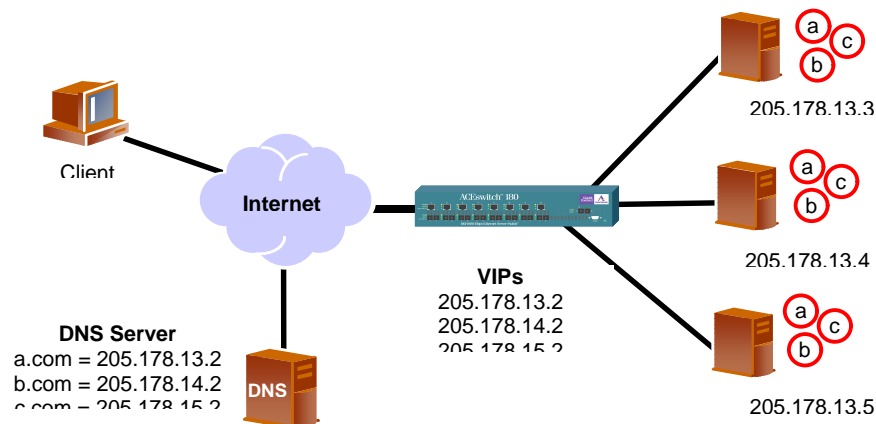
Besides monitoring Real Servers, one other example where direct access is needed is when you move your setup from round robin DNS to load balancing with a VIP (see picture bellow). Local DNSs will still have the real server addresses for a while and, if you don't enable DAM, the clients that get the site address from those DNSs will not be able to access the site until their cache is refreshed.



One Real Server serving multiple VIPs

A very common situation is to configure servers with more than one site if they don't have much traffic.

In the figure below, 3 sites, hence 3 virtual servers need to be supported by the 3 real servers available.



Normal server load balancing will not allow this because, during SPr, the SMT is not enough to determine which VIP it should translate back to. The key to SMT is (RIP, RPORT), and this key should be unique to map back to (VIP, VPORT). After configuring 3 virtual servers, the SMT would have 3 entries with the same key (RIP, RPORT), one for each VIP. SPr would probably use the first it sees.

Support for this scenario is achieved by enabling DAM because DAM's CPr will store the VSrI in the session table and A-SPr will replace RIP by VIP based on that field.

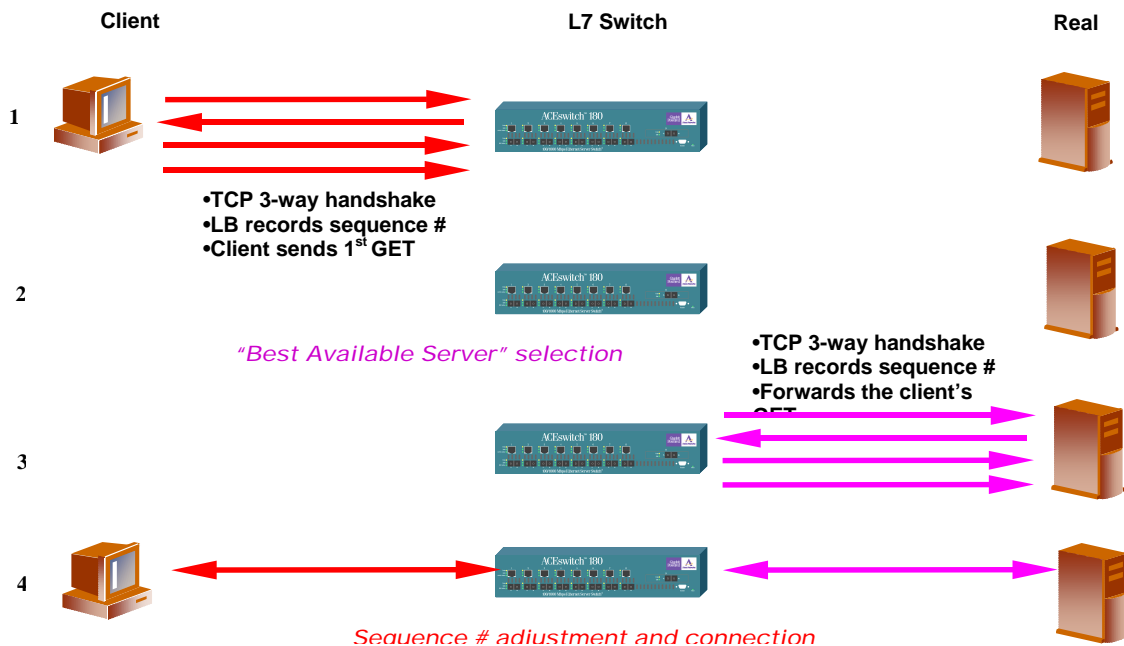
Content Intelligent Switching - Layer 7 Processing

Because when doing Content Intelligent Switching the decision to what real server to send the traffic needs to be done based on a packet that will come after TCP's 3-way handshake (usually the first HTTP GET), the switch must do what is called Delayed Binding.

Delayed Binding requires that the switch establishes one TCP connection with the client and, later, another with the chosen real server. When setting up the TCP connection with the client the switch must generate a sequence number of its own. Then, when the switch establishes the TCP connection with the server, the server will issue its own, different sequence number.

So, Delayed Binding, for its turn, requires that the switch adjusts the sequence numbers of the returning packets to the sequence numbers negotiated between the Alteon switch and the client machine during the 3-way handshake. The only way to do this, is if CPr and SPPr are done the same port so they share the same table (another table is used specifically for this). The consequence is that again we need DAM or PIP for their two characteristics: doing CPr and SPPr on the same port and doing special CPr and A-SPPr (now also supporting sequence number adjustments).

The figure bellow shows Delayed Binding.



DAM - Alternatives to,

If you decide that DAM is not an option there are a few alternatives. The alternatives will depend on the application.

Alternatives to DAM for Direct Access to Real Servers

- Define a second IP address for each real server and use this address for external access. Actually, we recommend you set a private IP address as a secondary IP to be used as the RPr and a public IP address as the primary (which will be used for outgoing TCP sessions, like SMTP or DNS needs).
- Direct Server Return - DSR does not need SPr because it does a different type of CPr (see details on WebOS Applications Guide).
- Port Mapping - For example, if you start a new instance of the web server at port 81, all you have to do is to access the server using that port instead of port 80. Because E-SPr will look at (sip=RIP, sport=80), traffic from sport=81 will be ignored.
- Enable L4 **mnet/mmask** - Traffic coming from that subnet will not go through for SPr.
- PIP - it also handles CPr and A-SPr in the same port

Alternatives to DAM for "Multiple Real Servers serving one VIP"

- At the Real Server, set one IP address per each VIP. As these IPs could be private IPs, this should not be a problem. Remember, though that there is a limit of 256 Real Servers per Alteon Switch (Tigon).
- PIP - it also handles CPr and A-SPr in the same port

Alternatives to DAM for Content Switching - L7 processing

- PIP - it also handles CPr and A-SPr in the same port

Alternatives to DAM for Active-Active setups

- PIP - Because PIP will force traffic to get back to the same switch (you will need 16 PIP addresses, but they could be private addresses)

DAM - Q & A

Q. Is that true that enabling DAM cuts the number of session entries in half?

A. No. DAM can work with ABT, no problem. It is enabling L7 processing that requires PBT and DAM. And PBT entries takes twice as much space than ABT entries, thus cutting in half the number of sessions supported by the switch.

Q. Why PIP is an alternative to DAM?

A. As DAM, PIP perform special CPr and A-SPr and assure the use of the same port because the packet will return from the server to the port because the DIP of the packet is the PIP.

Glossary

A-SPr	Advanced Server Processing, uses only the Session Table
CIP	Client IP, the IP address of the client machine that starts the request
CPr	Client Processing, replaces VIP->RIP
CPORT	Client Port - The source port address on the TCP packet coming from the client
DPORT	Destination Port - The destination port address on the TCP packet
DAM	Direct Access Mode - you should know this by heart, right? :)
DIP	Destination IP, the IP address in the Destination field of the IP packet
E-SPr	Express Server Processing, uses only the SMT
PIP	Proxy IP, actually the IP of a physical port of the switch. Also a method of CPr that replaces not only VIP by RIP but CIP to the IP of the port or PIP. This forces traffic returning from the server to get back to the port that did CPr.
RIP	Real IP, the IP address of a Real Server
RPORT	Real Port - The destination port address on the TCP packet exiting the switch destined to the Real Server/RIP. Corresponds to the rport option of virtual service for that VIP.
RSrI	Real Server Index, the number of the Real Server (8 bits)
SIP	Source IP, the IP address in the Source field of the IP packet
SMT	Service Mapping Table, which maps the pair (RIP,RPORT) to (VIP, VPORT)
SPr	Server Processing, replaces RIP->VIP. Two types: E-SPr and A-SPr
SPORT	Source Port - The source port address on the TCP packet
VIP	Virtual IP, the IP address of a Virtual Server
VPORT	Virtual Port - The destination port address on the TCP packet entering the switch destined to the VIP. Corresponds to a virtual service for that VIP.
VSrI	Virtual Server Index, the number of the Virtual Server (8 bits)
VMA	Virtual Matrix Architecture, a WebOS 8.0 feature that enables packets to be processed not only at the ingress port but on any of the first 8 ports of the switch (there is a hash algorithm to decide to which port the packet will be sent).