Modified Deficit Round Robin

Unlike other Cisco platforms, the 12000 series of routers doesn't support priority queuing, custom queuing, or weighted fair queuing. Instead, they implement Modified Deficit Round Robin (MDRR) for congestion management. MDRR can be configured on inbound interfaces for packets outbound to the switching fabric, and outbound interfaces for packets awaiting transmission.

MDRR classifies packets into queues based on the precedence bits in the IP header. These queues are serviced round-robin, with a queue being allowed to transmit a number of bytes each time it's serviced (the *quantum*). A deficit counter tracks how many bytes a queue has transmitted in each round.

When the queues first are configured, their deficit counters are set to be equal with their quantum. As packets are transmitted from a queue, their lengths are subtracted from the deficit counter; when the deficit counter reaches 0 (or less than 0), the scheduler moves on to the next queue. In each new round, each non-empty queue's deficit counter is incremented by its quantum value.

Beyond these ToS-based queues, MDRR has a special queue that can be serviced in one of two modes:

- Strict priority mode
- Alternate mode

In strict priority mode, the special queue can be serviced whenever it is not empty. Strict priority mode provides low latency and delay for those packets in the special queue, but if enough packets are queued into the special queue, the remainder of the queues can starve.

In alternate mode, the scheduler alternates between the special queue and the rest of the ToS-based queues.

The quantum associated with each queue is configurable by configuring its weight. The number of bytes a queue can transmit during each round is:

Bytes dequeued = MTU + (configured weight – 1) * 512

Let's work through an example; assume we have three queues:

- **Queue 0—**

  Has a quantum of 1500 bytes; it's the low latency queue, configured to operate in alternate mode.

- **Queue 1—**

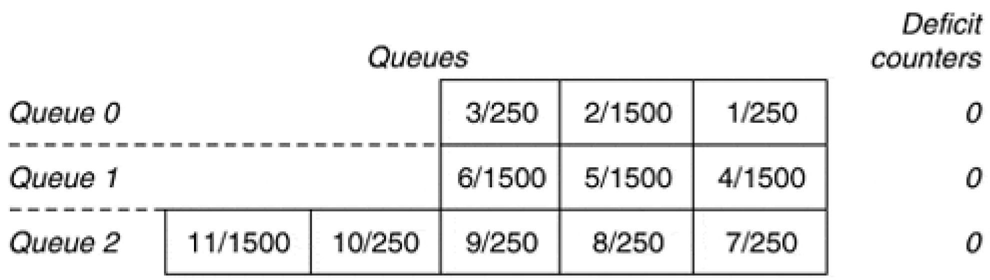
  Has a quantum of 3000 bytes.

- **Queue 2—**

  Has a quantum of 1500 bytes.

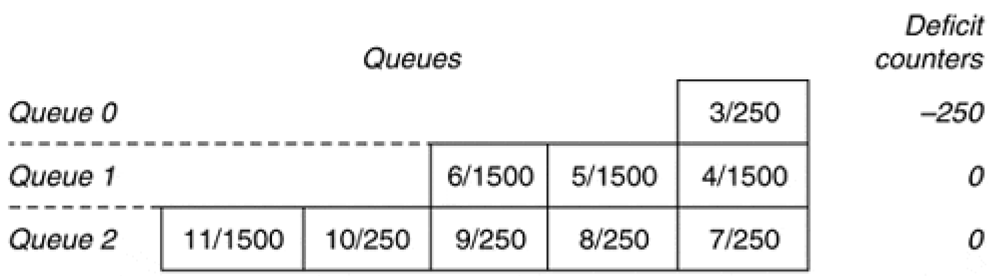
Figure 8-9 illustrates the initial state of the queues along with some packets that have been received and queued.

Figure 8-9. MDRR Initial State



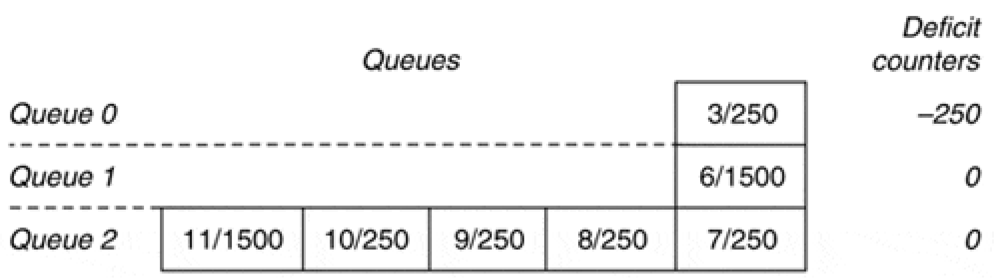
Queue 0 is serviced first; its quantum is added to its deficit counter. Packet 1, which is 250 bytes, is transmitted, and its size is subtracted from the deficit counter. Because queue 0's deficit counter is still greater than 0 (1500 – 250 = 1250), packet 2 is transmitted as well, and its length subtracted from the deficit counter. Queue 0's deficit counter is now –250, so queue 1 is serviced next; Figure 8-10 shows this state.

Figure 8-10. MDRR State



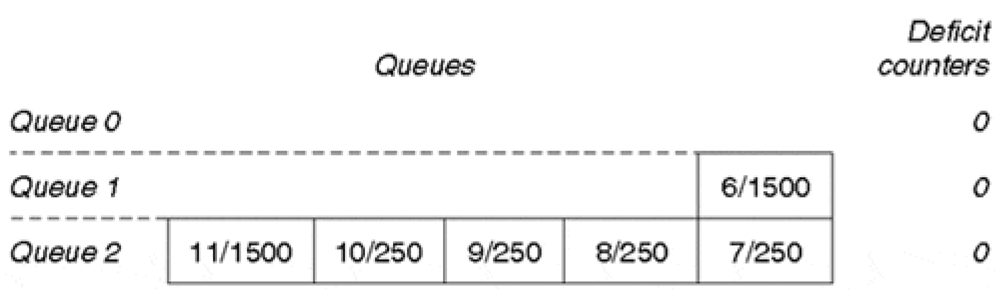
Queue 1's deficit counter is set to 3000 (0 + 3000 = 3000), and packets 4 and 5 are transmitted. With each packet transmitted, subtract the size of the packet from the deficit counter, so queue 1's deficit counter is reduced to 0. Figure 8-11 illustrates this state.

Figure 8-11. MDRR State



Because we are in alternate priority mode, we must return and service queue 0. Again, we add the quantum to the current deficit counter, and set queue 0's deficit counter to the result (–250 + 1500 = 1250). Packet 3 is now transmitted, because the deficit counter is greater than 0, and queue 0 is now empty. When a queue is emptied, its deficit counter is set to 0, as shown in Figure 8-12.

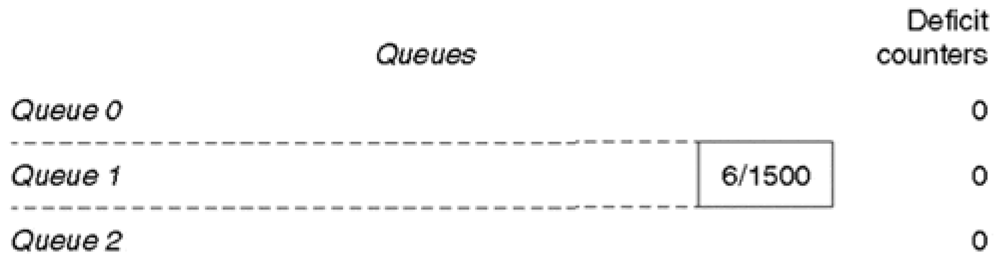
Figure 8-12. MDRR State



Queue 2 is serviced next; its deficit counter is set to 1500 (0 + 1500 = 1500). Packets 7 through 10 are transmitted, leaving the deficit counter at 500 (1500 – (4 * 250) = 500). Because the deficit counter is still greater than 0, packet 11 is transmitted also.

When packet 11 is transmitted, queue 2 is empty, and its deficit counter is set to 0, as shown in Figure 8-13.

Figure 8-13. MDRR State



Queue 0 is serviced again next (because we are in alternate priority mode). Because it's empty, we service queue 1 next, transmitting packet 6.

Configuration of MDRR

The command **cos-queue-group** places you in a subconfiguration mode, allowing you to create a CoS template, such as one of the following:

- **precedence—**

  Used to map the IP precedence to DRR queues and to configure the special low latency queue.

- **queue—**

  Sets the relative weight for each queue.

- **tx-cos—**

  Used to map cos-queue-group parameters on the transmit interface.

When using MDRR for inbound congestion management, cos-queue-groups are mapped to destination line card slots. The **slot-table-cos**and **rx-cos-slot** commands map the parameters defined in cos-queue-group to each of the destination slots.

Assume, for instance, that you have an OC-12 card in slot 2 of a 12000, and an OC-3 card in slot 3. You want to map precedence 7 traffic into the low-latency (special) queue, and service it in strict priority mode. You could use the configuration in Example 8-14.

Example 8-14. MDRR Configuration on GSR

Interface pos2/1 Description OC12 Interface ! Interface 3/1 Description OC-3 Interface tx-cos frfab ! rx-cos_slot 2 table-a ! slot-table-cos table-a destination-slot 3 tofab ! cos-queue-group frfab precedence 7 queue low-latency queue low-latency strict-priority 10 ! cos-queue-group tofab precedence 7 queue low-latency queue low-latency strict-priority 10 ! end

Last updated on 12/5/2001

Index terms contained in this section