

了解 ATM 上的加权公平排队

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[规则](#)

[网络图](#)

[如何设置传输环限制](#)

[传输环路限制的影响](#)

[示例 A](#)

[示例 B](#)

[如何计算重量](#)

[如何计算计划时间](#)

[WFQ 如何工作](#)

[什么是微粒？](#)

[测试 A](#)

[测试 B](#)

[第 1 阶段](#)

[第 2 阶段](#)

[第 3 阶段](#)

[第 4 阶段](#)

[摘要](#)

[相关信息](#)

简介

本文档介绍使用加权公平队列(WFQ)技术的流量队列。

引入WFQ是为了启用慢速链路（如串行链路），以为所有类型的流量提供公平的处理。为此，WFQ根据第3层和第4层信息（如IP地址和TCP端口）将流量分类为不同的流（也称为会话）。它无需您定义访问列表即可执行此操作。这意味着低带宽流量实际上比高带宽流量具有优先级，因为高带宽流量与其分配的权重成正比地共享传输介质。

但是，WFQ有一些局限性：

- 如果流量显着增加，则无法扩展。
- 本地WFQ在高速接口（如ATM接口）上不可用。

基于类的加权公平队列(CBWFQ)提供了解决这些限制的解决方案。

与标准WFQ不同，CBWFQ允许您定义流量类。您还可以将带宽和队列限制等参数应用到它们。为

类分配的带宽用于计算该类的权重。此外，还会根据此计算与类标准匹配的每个数据包的权重。然后，WFQ将应用于类，类可以包括多个流，而不是流本身。

有关如何配置CBWFQ的详细信息，请参阅以下文档：

- [Cisco 7200、3600和2600路由器上基于每VC类的加权公平队列 \(每VC CBWFQ\)](#)
- [基于 RSP 的平台上的每 VC 基于类的加权公平排队](#)

ATM接口不支持直接在接口上使用fair-queue命令配置的基于本地流的WFQ。但是，使用支持CBWFQ的软件，您可以在默认类内配置基于流的WFQ，如本示例所示：

```
policy-map test
  class class-default
    fair-queue
!
interface ATMx/y.z point-to-point
 ip address a.b.c.d M.M.M.M
 pvc A/B
  service-policy output test
```

[先决条件](#)

[要求](#)

本文档没有任何特定的要求。

[使用的组件](#)

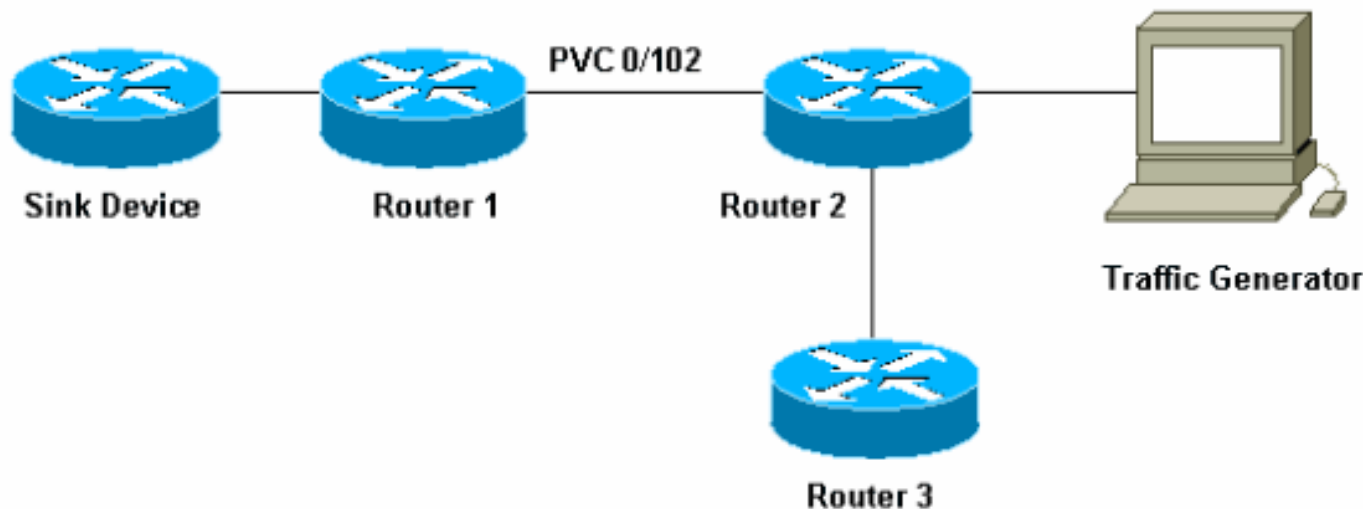
本文档不限于特定的软件和硬件版本。

[规则](#)

有关文档规则的详细信息，请参阅 [Cisco 技术提示规则](#)。

[网络图](#)

使用此设置可说明WFQ的工作原理：



在此设置中，数据包可以存储在以下两个队列之一：

- 端口适配器和网络模块上的硬件先进先出(FIFO)队列
- Cisco IOS®软件^中的路由器输入/输出[I/O]内存上的队列，其中可以应用服务质量(QoS)功能（如CBWFQ）

端口适配器上的FIFO队列在数据包被分段为信元以供传输之前存储这些数据包。当此队列已满时，端口适配器或网络模块会向IOS软件发出队列拥塞的信号。这种机制称为背压。收到此信号后，路由器停止向接口FIFO队列发送数据包，并将数据包存储在IOS软件中，直到队列再次不拥塞。当数据包存储在IOS中时，系统可以应用QoS。

如何设置传输环限制

此排队机制的一个问题是，接口上的FIFO队列越大，此队列末尾的数据包传输前的延迟就越长。这可能导致延迟敏感型流量（如语音流量）出现严重的性能问题。

永久虚电路(PVC)tx-ring-limit命令使您能够减小FIFO队列的大小。

```
interface ATMx/y.z point-to-point
ip address a.b.c.d M.M.M.M
PVC A/B
  tx-ring-limit
  service-policy output test
```

您可此处指定的数据包数量（对于Cisco 2600和3600路由器）或粒子数量（对于Cisco 7200和7500路由器）。

减小传输环的大小有两个优点：

- 它减少了数据包在分段之前在FIFO队列中等待的时间。
- 它加快了IOS软件中QoS的使用。

传输环路限制的影响

查看使用前面网络图中显示的设置的传输环限制的影响。假设：

- 流量生成器将流量（1500字节数据包）发送到接收设备，此流量会在router1和router2之间过载

!!!!!!!!!!!!!!.

Success rate is 92 percent (12/13), round-trip min/avg/max = 6028/6350/6488 ms

如图所示，传输环限制越大，ping往返时间(RTT)就越大。从此可以推断，较大的传输环限制可能导致传输中出现显著延迟。

如何计算重量

在示例A中的show queue atm输出中，您会看到为每个会话分配了权重。请更详细地查看此内容：

```
router2#show queue ATM 4/0.102
  Interface ATM4/0.102 VC 0/102
  Queuing strategy: weighted fair
  Total output drops per VC: 1505772
  Output queue: 65/512/64/1505772 (size/max total/threshold/drops)
  Conversations 2/3/16 (active/max active/max total)
  Reserved Conversations 0/0 (allocated/max allocated)

(depth/weight/discards/tail drops/interleaves) 1/32384/0/0/0
  Conversation 2, linktype: ip, length: 58
  source: 8.0.0.1, destination: 6.6.6.6, id: 0x2DA1, ttl: 254, prot: 1

(depth/weight/discards/tail drops/interleaves) 64/32384/1505776/0/0
  Conversation 15, linktype: ip, length: 1494
  source: 7.0.0.1, destination: 6.6.6.6, id: 0x0000, ttl: 63, prot: 255
```

使用WFQ时，可以使用以下公式计算每个会话的权重：

- $\text{weight} = 32384 / (\text{precedence} + 1)$ — 适用于Cisco IOS软件版本12.0(5)T及更高版本。
- $\text{weight} = 4096 / (\text{precedence} + 1)$ — 适用于12.0(5)T之前的Cisco IOS软件版本。

如何计算计划时间

现在，当数据包从IOS队列转发到端口适配器或网络模块FIFO队列时，可以使用这些权重来计算每个数据包的调度时间。

您可以使用此公式计算输出调度时间，其中queue_tail_time是当前调度时间：

输出调度时间 = queue_tail_time + pktsize * weight

WFQ 如何工作

本节介绍WFQ的工作原理。WFQ的原理是，权重较小的数据包（即小数据包）在发送时应获得优先级。

创建一个流，该流包含十个大数据包和四个小数据包（82字节），这些数据包使用流量生成器来验证这一点。

在本例中，router2是带PA-A3（ATM端口适配器）的Cisco 7200路由器。这一点很重要，因为端口适配器上的输出FIFO队列的大小以粒子表示，而不是以数据包表示。看到什么[什么是粒子吗？](#)的下界。

什么是微粒？

粒子缓冲不是为缓冲区分配一个连续的内存，而是分配不连续（分散的）内存，称为粒子，然后将它们链接在一起，以形成一个逻辑分组缓冲区。这称为粒子缓冲。在这种方案中，分组随后可以跨多个粒子分布。

在7200路由器中，粒度为512字节。

使用show buffers命令验证Cisco 7200路由器是否使用粒子：

```
router#show buffers
[snip]
Private particle pools:
FastEthernet0/0 buffers, 512 bytes (total 400, permanent 400):
  0 in free list (0 min, 400 max allowed)
  400 hits, 0 fallbacks
  400 max cache size, 271 in cache
ATM2/0 buffers, 512 bytes (total 400, permanent 400):
  0 in free list (0 min, 400 max allowed)
  400 hits, 0 fallbacks
  400 max cache size, 0 in cache
```

测试 A

以下是一些测试，以说明WFQ功能。在第一次测试中，查看带宽是否可以在不同会话之间共享。

在本测试中，您使流量生成器发送的流量足够快，以使路由器1和路由器2之间的PVC 0/102过载。请通过同一PVC执行从路由器3到路由器1的ping操作：

```
pound#ping ip
Target IP address: 6.6.6.6
Repeat count [5]: 100000
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 100000, 100-byte ICMP Echos to 6.6.6.6, timeout is 2 seconds:
..... (WFQ is enabled here)!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 98 percent (604/613), round-trip min/avg/max = 164/190/232 ms
```

如图所示，在接口上启用WFQ之前，流量会阻止其他流量通过并启动线路。一旦启用WFQ，ping操作就会成功。

从中您可以看到，使用WFQ，带宽可以在不同会话之间共享，而不会阻塞其他会话。

测试 B

这是带宽的共享方式。

流量生成器发送的流是由十个大数据包组成的突发流，后跟四个82字节的小数据包。您以100 Mbps的速率将此数据发送到router2。当您发送突发数据包时，router2 ATM接口上的输出队列为空。Router2通过10 KB PVC（这是一条非常慢的PVC）发送这些数据包，以确保输出队列发生拥塞。

分几个阶段执行此测试，以简化此过程：

第 1 阶段

大流量包含10个482字节的数据包。由于PA-A3上的粒子为512字节，因此每个数据包在存储到端口适配器输出队列中时，无论大小，都应采用一个粒子。路由器的传输环限制为3(tx-ring-limit=3)。以下是您在接收器设备上看到的示例：

```
.Nov 7 15:39:13.776: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, len 482, rcvd 4
.Nov 7 15:39:13.776: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:39:14.252: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:39:14.252: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:39:14.732: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:39:14.732: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:39:15.208: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:39:15.208: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol

!--- Congestion occurs at this point. .Nov 7 15:39:15.512: IP: s=7.0.0.200 (FastEthernet0/1),
d=6.6.6.6, Len 82, rcvd 4 .Nov 7 15:39:15.516: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len
82, unknown protocol .Nov 7 15:39:15.644: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82,
rcvd 4 .Nov 7 15:39:15.644: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown
protocol .Nov 7 15:39:15.776: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4 .Nov
7 15:39:15.776: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol .Nov 7
15:39:15.904: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4 .Nov 7 15:39:15.904:
IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol .Nov 7 15:39:16.384: IP:
s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4 .Nov 7 15:39:16.384: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol .Nov 7 15:39:16.860: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4 .Nov 7 15:39:16.860: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol .Nov 7 15:39:17.340: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4 .Nov 7 15:39:17.340: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol .Nov 7 15:39:17.816: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4 .Nov 7 15:39:17.820: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol .Nov 7 15:39:18.296: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4 .Nov 7 15:39:18.296: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol .Nov 7 15:39:18.776: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4 .Nov 7 15:39:18.776: IP: s=7.0.0.1
(FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
```

您可以看到在82字节数据包之前发送的4个482字节数据包，这些数据包通常应该获得优先级。这就是为什么会发生。

由于突发主要由10个482字节数据包组成，因此这些数据包首先到达路由器，然后是82字节数据包。由于482字节的数据包到达时没有拥塞，因为没有流量，所以一个数据包会立即排队到端口适配器分段和重组(SAR)，被分组到信元中，并通过线路发送。换句话说，传输环仍为空。

您可以计算发送一个482字节数据包所需的时间比流量生成器发送总突发所需的时间长。因此，您可以假设，当第一个482字节数据包排入端口适配器的队列时，路由器中已存在更多482字节的突发数据包。因此，可以将更多482字节的数据包排入传输环的队列。另外3个482字节的数据包使用其中的三个自由粒子排队。

注意：当有自由粒子时，即使需要存储多个粒子，数据包也会排入传输环。

此时，由于这三个粒子已满，因此存在拥塞。因此，排队在IOS中启动。当四个82字节的数据包最终到达路由器时，会出现拥塞。这四个数据包已排队，并且WFQ用于两个流。查看使用show queue ATM命令的ATM队列，以查看以下内容：


```

router2#show queue ATM 4/0.102 vc 0/102
  Interface ATM4/0.102 VC 0/102
  Queuing strategy: weighted fair
  Total output drops per VC: 0
  Output queue: 10/512/64/0 (size/max total/threshold/drops)
    Conversations 2/4/16 (active/max active/max total)
    Reserved Conversations 0/0 (allocated/max allocated)

(depth/weight/total drops/no-buffer drops/interleaves) 4/32384/0/0/0
  Conversation 6, linktype: ip, length: 82
  source: 7.0.0.200, destination: 6.6.6.6, id: 0x0000, ttl: 63, prot: 255

```

```

(depth/weight/total drops/no-buffer drops/interleaves) 6/32384/0/0/0
  Conversation 15, linktype: ip, length: 482
  source: 7.0.0.1, destination: 6.6.6.6, id: 0x0000, ttl: 63, prot: 255

```

在调试中，您可以看到，482字节的前四个数据包后跟82字节的数据包。这些小数据包在大数据包之前从路由器传出。这表示一旦发生拥塞，小数据包的优先级会高于大数据包。

使用“计算权重”部分中给出的权重和计划时间公式来验证这一点。

第 2 阶段

如果将传输环限制增加到五个，且大数据包为482字节，则根据上一输出，在发生拥塞之前，您应该看到六个482字节的数据包，然后是四个82字节的数据包，然后是另外四个482字节的数据包：

```

.Nov 7 15:49:57.365: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:49:57.365: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:49:57.841: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:49:57.845: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:49:58.321: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:49:58.321: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:49:58.797: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:49:58.801: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:49:59.277: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:49:59.277: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:49:59.757: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:49:59.757: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:49:59.973: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 7 15:49:59.973: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 7 15:50:00.105: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 7 15:50:00.105: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 7 15:50:00.232: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 7 15:50:00.232: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 7 15:50:00.364: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 7 15:50:00.364: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 7 15:50:00.840: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:50:00.844: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:50:01.320: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:50:01.320: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:50:01.796: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:50:01.800: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol
.Nov 7 15:50:02.276: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, rcvd 4
.Nov 7 15:50:02.276: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 482, unknown protocol

```

正如你们所见，这确实是所发生的。

第 3 阶段

粒度为512字节。因此，如果传输环以粒子表示，并且您使用的数据包比粒子大小稍大，则每个传

输环都包含两个粒子。这通过使用582字节的数据包和三个传输环来说明。使用这些参数时，您应该看到三个582字节的数据包。其中一个在ATM接口上发送，而没有流量，这会使三个粒子保持空闲。因此，可以再排两个数据包，然后是四个82字节的数据包，然后是七个582字节的数据包：

```
.Nov 7 15:51:34.604: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:34.604: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:35.168: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:35.168: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:35.732: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:35.736: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:35.864: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 7 15:51:35.864: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 7 15:51:35.996: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 7 15:51:35.996: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 7 15:51:36.124: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 7 15:51:36.124: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 7 15:51:36.256: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 7 15:51:36.256: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 7 15:51:36.820: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:36.820: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:37.384: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:37.388: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:37.952: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:37.952: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:38.604: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:38.604: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:39.168: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:39.168: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:39.732: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:39.736: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
.Nov 7 15:51:40.300: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, rcvd 4
.Nov 7 15:51:40.300: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 582, unknown protocol
```

第 4 阶段

以数据包大小1482 (三个粒子) 为例，定义一个传输环 (五个)。如果传输环在粒子中定义，您会看到类似的内容：

- 立即传输一个数据包
- 一个数据包，接收五个粒子中的三个
- 一个数据包排队，因为两个粒子是空闲的

```
.Nov 8 07:22:41.200: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:41.200: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:42.592: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:42.592: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:43.984: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:43.984: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:44.112: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 8 07:22:44.112: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 8 07:22:44.332: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 8 07:22:44.332: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 8 07:22:44.460: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 8 07:22:44.460: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 8 07:22:44.591: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, rcvd 4
.Nov 8 07:22:44.591: IP: s=7.0.0.200 (FastEthernet0/1), d=6.6.6.6, Len 82, unknown protocol
.Nov 8 07:22:45.983: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:45.983: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:47.371: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
```

```
.Nov 8 07:22:47.375: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:48.763: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:48.767: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:50.155: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:50.155: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:51.547: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:51.547: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:53.027: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:53.027: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
.Nov 8 07:22:54.415: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, rcvd 4
.Nov 8 07:22:54.419: IP: s=7.0.0.1 (FastEthernet0/1), d=6.6.6.6, Len 1482, unknown protocol
```

摘要

从所执行的测试中，您可以得出以下结论：

- 在没有WFQ的慢速PVC上，批量流量会影响小流量，例如ping在启用WFQ之前会停止。
- 传输环的大小(tx-ring-limit)决定了排队机制开始执行其工作的速度。当传输环限制增加时，ping RTT的增加会导致这种影响。因此，如果需要实施WFQ或LLQ，则降低传输环限制是有意义的。
- 使用CBWFQ的WFQ确实优先处理小流量，而非批量流量。

相关信息

- [ATM技术支持页](#)
- [拥塞管理概述](#)
- [技术支持和文档 - Cisco Systems](#)