

机制。存储转发机制是指在交换机能够开始向输出链路传输该分组的第一个比特之前，必须接收到整个分组。为了更为详细地探讨存储转发传输，考虑由两个经一台路由器连接的端系统构成的简单网络，如图 1-11 所示。一台路由器通常有多条繁忙的链路，因为它的任务就是把一个入分组交换到一条出链路。在这个简单例子中，该路由器的任务相当简单：将分组从一条（输入）链路转移到另一条唯一的连接链路。在图 1-11 所示的特定时刻，源已经传输了分组 1 的一部分，分组 1 的前沿已经到达了路由器。因为该路由器应用了存储转发机制，所以此时它还不能传输已经接收的比特，而是必须先缓存（即“存储”）该分组的比特。仅当路由器已经接收完了该分组的所有比特后，它才能开始向出链路传输（即“转发”）该分组。为了深刻领悟存储转发传输，我们现在计算一下从源开始发送分组到目的地收到整个分组所经过的时间。（这里我们将忽略传播时延——指这些比特以接近光速的速度跨越线路所需要的时间，这将在 1.4 节讨论。）源在时刻 0 开始传输，在时刻  $L/R$  秒，因为该路由器刚好接收到整个分组，所以它能够朝着目的地向出链路开始传输分组；在时刻  $2L/R$ ，路由器已经传输了整个分组，并且整个分组已经被目的地接收。所以，总时延是  $2L/R$ 。如果交换机一旦比特到达就转发比特（不必首先收到整个分组），则因为比特没有在路由器保持，总时延将是  $L/R$ 。而如我们将在 1.4 节中讨论的那样，路由器在转发前需要接收、存储和处理整个分组。



图 1-11 存储转发分组交换

现在我们来计算从源开始发送第一个分组直到目的地接收到所有三个分组所需的时间。与前面一样，在时刻  $L/R$ ，路由器开始转发第一个分组。而在时刻  $L/R$  源也开始发送第二个分组，因为它已经完成了发送整个第一个分组。因此，在时刻  $2L/R$ ，目的地已经收到第一个分组并且路由器已经收到第二个分组。类似地，在时刻  $3L/R$ ，目的地已经收到前两个分组并且路由器已经收到第三个分组。最后，在时刻  $4L/R$ ，目的地已经收到所有 3 个分组！

我们现在来考虑通过由  $N$  条速率均为  $R$  的链路组成的路径（所以，在源和目的地之间有  $N-1$  台路由器），从源到目的地发送一个分组的总体情况。应用如上相同的逻辑，我们看到端到端时延是：

$$d_{\text{端到端}} = N \frac{L}{R} \quad (1-1)$$

你也许现在要试着确定  $P$  个分组经过  $N$  条链路序列的时延有多大。

## 2. 排队时延和分组丢失

每个分组交换机有多条链路与之相连。对于每条相连的链路，该分组交换机具有一个输出缓存（output buffer）（也称为输出队列 output queue），它用于存储路由器准备发往那条链路的分组。该输出缓存在分组交换中起着重要的作用。如果到达的分组需要传输到某条链路，但发现该链路正忙于传输其他分组，该到达分组必须在该输出缓存中等待。因