

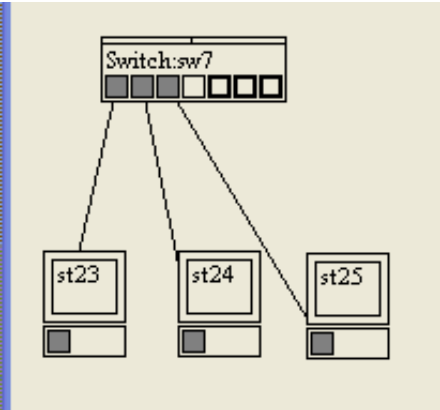
通过上一篇的学习，我们了解到集线器和交换机最大的区别是交换机可以知道帧往哪个端口发，而不是像集线器一样大嘴巴只能广播去传递消息。那么交换机是如何知道发给谁的呢？

## 一、帧转发和CAM表

为了将以太网帧（或简称“帧”）发送到正确的机器，交换机需要知道此帧中的目标 MAC 地址（Destination MAC Address），我们知道，这个地址是在帧的首部中。

交换机比之前我们学过的集线器强大得多，主要是因为它有一个表（维护在交换机的内存中），可以将交换机的一个 RJ45 接口（母头）和一个 MAC 地址相关联。这个表就是著名的 CAM 表。

比如有三台机器连接到了交换机上：



当23号机器想往25号机器发送一个帧的时候，交换机就可以读取帧头里面的目标MAC地址，根据 CAM 表中 MAC 地址与 RJ45 端口号的对应关系，就知道要发送给哪个端口（port）了。

25 号机器的 MAC 地址 (目标地址)	23 号机器的 MAC 地址 (源地址)	第 3 层 使用的协议	要发送的信息	CRC
-----------------------------	----------------------------	----------------	--------	-----

若此时CAM表为：

端口	MAC 地址
1	23 号机器的网卡的 MAC 地址
2	24 号机器的网卡的 MAC 地址
3	25 号机器的网卡的 MAC 地址

那么就知道25号机器的MAC绑定在3号端口上，交换机就会把帧发送到端口 3 上，继而这个帧又会到达 25 号机器。

交换机就是依据帧里面的目标 MAC 地址，和交换机自己的 CAM 表来把帧转发到正确的机器上的。

那么CAM表的记录是如何生成的呢？

## 二、CAM表的更新、帧转发和CAM表

还是以交换机连接三台设备为例，并且 23 号机器发送了一个帧给 25 号机器。

那么交换机拿到这一帧数据后，会发生下面的事情：

- 交换机读取此帧，看到源 MAC 地址 (Source MAC Address) ，是 23 号机器的网卡的 MAC 地址；
- 交换机发现这个帧是从端口 1 过来的，因此它就把端口 1 和 23 号机器的网卡的 MAC 地址关联起来，放入交换机的 CAM 表中；

这里我们可以看出，交换机是依靠帧来更新CAM表的，而不是主动获取机器的MAC地址。

更新后的 CAM 表如下所示：

端口	MAC 地址
1	23 号机器的网卡的 MAC 地址

此时，帧中的目标 MAC 地址（也就是 25 号机器的网卡的 MAC 地址）还没有在交换机的 CAM 表中，因此交换机还不知道应该把此帧发送到哪个端口。

为了保证目标机器能够接收到此帧，交换机就会把此帧发送给除了源 MAC 所关联的机器（此处是 23 号机器）以外的所有机器。在我们的例子中，就是发送给除了端口 1 以外的所有连接着机器的端口。这种操作通常被称为 flooding（泛洪），将从某个接口收到的数据向除该接口之外的所有接口发送出去。

当真正的25号机器收到这一帧之后就会处理，并且回复给23号机器。那么交换机又会收到一帧数据：

- 交换机读取此帧，看到源 MAC 地址，是 25 号机器的网卡的 MAC 地址；
- 交换机发现这个帧是从端口 3 过来的，因此它就把端口 3 和 25 号机器的网卡的 MAC 地址关联起来，放入交换机的 CAM 表中；

更新后的 CAM 表如下所示：

端口	MAC 地址
1	23 号机器的网卡的 MAC 地址
3	25 号机器的网卡的 MAC 地址

这样就实现了CAM表的更新，但是如果连接的MAC地址更新了，岂不是不好办？还好有TTL机制。

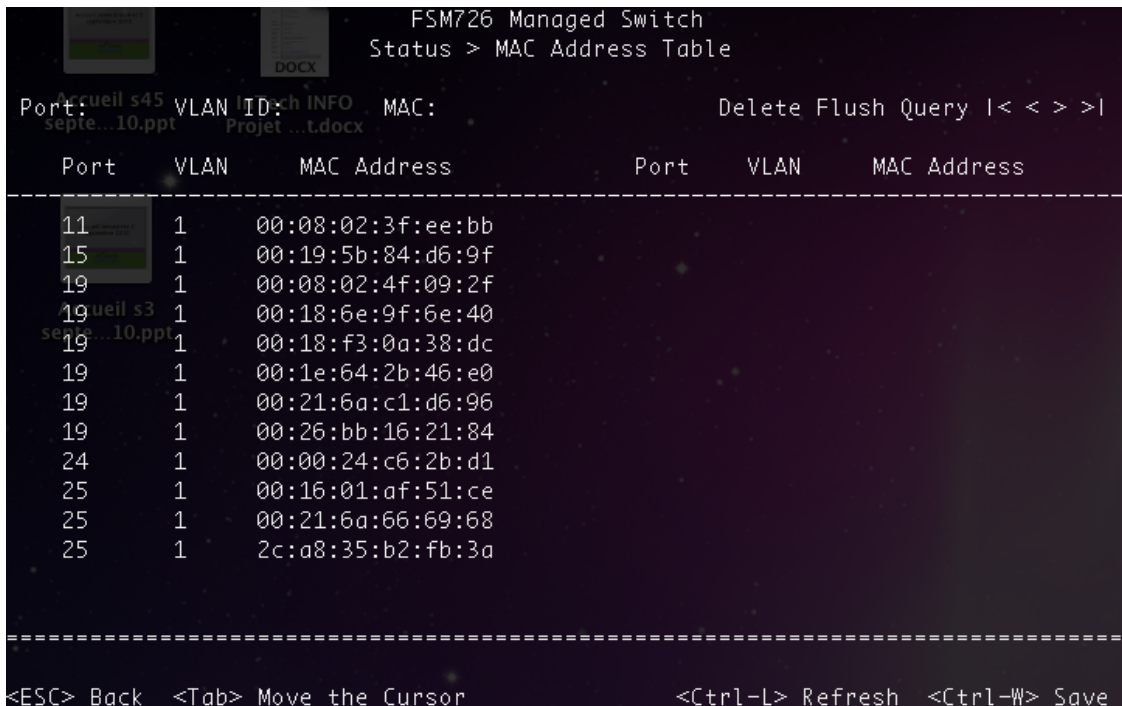
### 三、CAM表的生存时间TTL

TTL 是 Time To Live 的缩写，表示“生存时间”。因此，TTL 代表了一段时间。

即CAM表中的数据是有一定的生存时间的，过了这个时间，某个端口一个帧都没有流过，就把这段数据从 CAM 表中移除。

当然了，如果某个端口一直有数据在发送，那么交换机会不断地重置TTL。比如25号机器再次发送一个帧，它的 TTL 时间就会被重置。这样，交换机在每次接收到以太网帧的时候都会扩充 CAM 表的内容或者更新已有的内容。**如果过了很久，交换机都没有收到任何帧，那么它的 CAM 表就会被清空。**

下面展示一张真实的交换机的 CAM 表。Port 表示“端口”，MAC Address 表示“MAC 地址”：



```

FSM726 Managed Switch
Status > MAC Address Table

Port: ueil s45  VLAN ID: ch INFO  MAC:          Delete Flush Query |< > >|
septe...10.ppt  Projet ...t.docx

  Port    VLAN    MAC Address          Port    VLAN    MAC Address
-----
  11      1      00:08:02:3f:ee:bb
  15      1      00:19:5b:84:d6:9f
  19      1      00:08:02:4f:09:2f
  19      1      00:18:6e:9f:6e:40
  19      1      00:18:f3:0a:38:dc
  19      1      00:1e:64:2b:46:e0
  19      1      00:21:6a:c1:d6:96
  19      1      00:26:bb:16:21:84
  24      1      00:00:24:c6:2b:d1
  25      1      00:16:01:af:51:ce
  25      1      00:21:6a:66:69:68
  25      1      2c:a8:35:b2:fb:3a
-----
<ESC> Back  <Tab> Move the Cursor          <Ctrl-L> Refresh  <Ctrl-W> Save

```

可以看到，竟然有 6 台机器连接着此交换机的端口 19，也有 3 台机器连接着端口 25。这是为什么呢？因为有可能我们这台交换机上还连接着另一个交换机，这样所有连接在第二台交换机上的机器的 MAC 地址就很可能被显示连接在第一台交换机的这个端口上。

## 四、一些捣蛋行为、CAM表的生存时间TTL

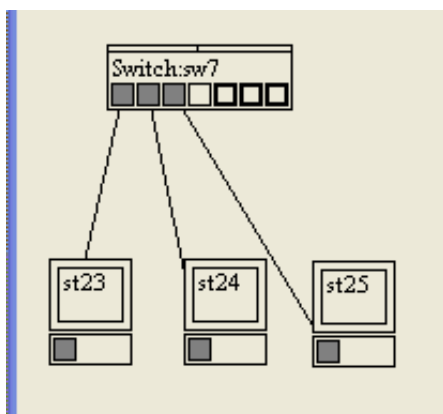
我们知道了交换机是靠CAM来实现往指定端口的转发数据帧的，那么如果我们持续向不存在的目标 MAC 地址发送大量的帧时，交换机不知道应该把这些帧发送到哪个端口，就会把它们发送到所有连着机器的端口... 这样交换机马上就会超负荷运作了。

或者如果每次都改变下帧中源MAC地址，CAM表会逐渐变大。CAM 表中的数据越多，交换机每次读取数据所花的时间就越长，不仅交换机的内存会不够用，交换机的资源也会被耗尽。当交换机饱和，没有时间去读取 CAM 表中的庞大数据时，它就会把收到的帧转发到所有连着机器的端口。

## 五、交换机带来的变革：跟CSMA/CD协议say bye bye!

我们之前学过，在总线拓扑中，依靠CSMA / CD 协议克服冲突，但是这种通过等待的解决方案还是很麻烦的，通信效率也不高。

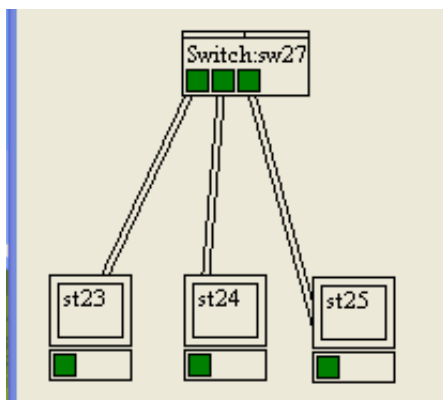
那么，交换机上也会有冲突吗？还是以这张图为例：



假设 23 号机器和 25 号机器同时发送信息给对方，会产生冲突吗？

由于我们使用的是双绞线，我们已经学习过双绞线的组成，我们使用不同的电线进行发送和接收。因此，发送的信息和接收的信息会交错，但是在不同的电线上。因此不会出现信息的冲突。

下面这个图更符合事实：



那如果23 号机器和 25 号机器同时向 24 号机器发送信息呢？在这种情况下，两条信息将同时到达 24 号机器的接收线上，是不是会发生冲突呢？

按理说这种情况下是有冲突发生的。我们说没有冲突，是因为这种情况已经被预计到了，交换机已经被设计好了来应对这种情况。

事实上，交换机拥有内存。当它接收到一个或多个帧时，它可以将其存储在内存里，仅当目标机器连接的双绞线的传输对空闲时才转发该帧。

因此，当交换机有两个帧要在同一对接收线上发送时，它会先发送第一个帧，然后再发送第二个帧，因此不会有冲突。

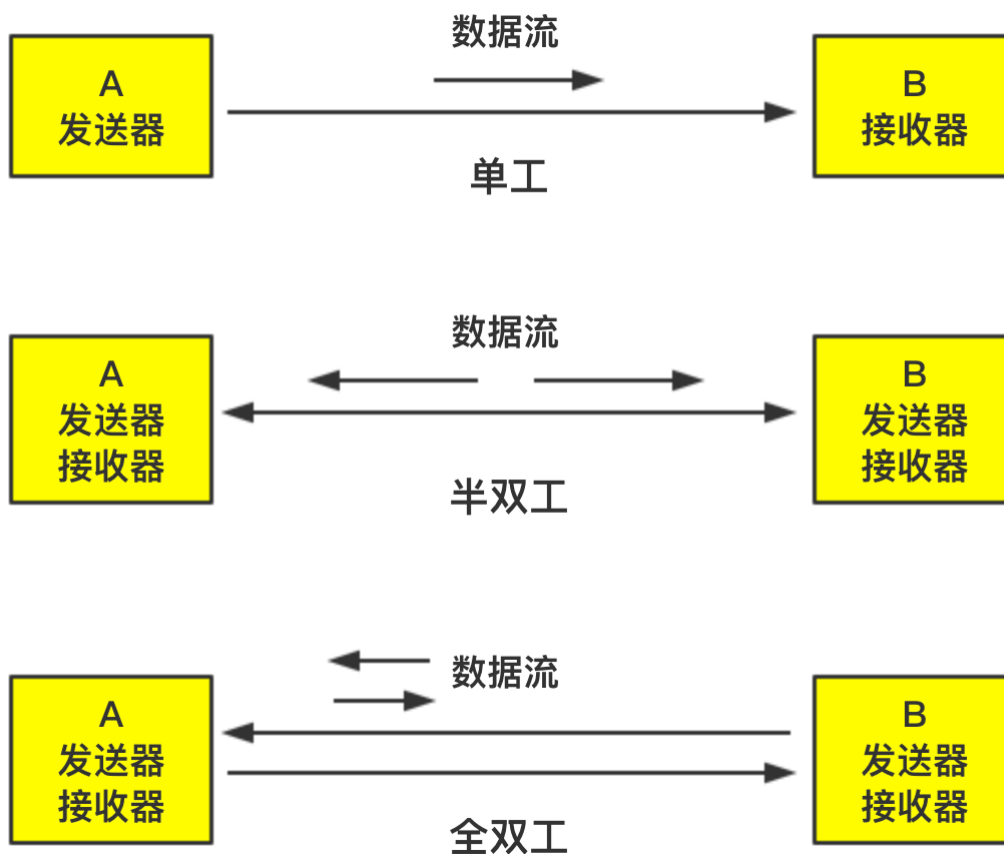
没有交换机之前，机器在发送帧之前必须侦听，确定网络是空闲时才能发送帧，这是 CSMA / CD 的基本原理。现在，只要一台机器想要发送帧，它就发送，而不必侦听是否有其他机器正在发送帧，因为交换机可以确保不发生冲突。

在交换机上这样的放弃了 CSMA / CD 的通信方式有一个术语，我们说此时网卡是以全双工的方式来运作的。

相反，在集线器或同轴电缆上使用 CSMA / CD 协议进行通信时，网卡是以半双工的方式来运作的。（同轴电缆就一根线，只能最多是半双工的模式；虽然集线器可以使用双绞线，提供了两对线，但在任一时刻，只能有一对线工作，所以集线器是半双工的）。

## 六、交换机带来的变革：全双工模式火力全开！

交换机工作模式是全双工模式，我们来复习下各种模式定义：



- **单工 (Simplex)**：数据传输是单向的。通信双方中，一方固定为发送端，另一方则固定为接收端，数据只能沿一个方向传输，类似汽车的单行道。
- **半双工 (Half Duplex)**：数据传输是双向的。数据在通信双方之间能够在两个方向上进行发送，但不能同时发送，因此又被称为双向交替通信。无线对讲机就是一种半双工设备，在同一时间内只允许一方讲话：“长江，长江，我是黄河，收到请回答，完毕”；“黄河，黄河，我是长江，已经收到，完毕”。
- **全双工 (Full Duplex)**：数据传输是双向的。通信双方在发送数据的同时也能够接收数据，两者可以同步进行，类似汽车的双向车道。目前我们打电话，以及手机的通话，都是全双工的例子。

**全双工是现在主流也是很棒的方式，但是有的时候也会出现问题。**

假设我们将 10 台机器连接到集线器。我们在集线器上，是在总线拓扑上，因此这些机器要以半双工的通信方式来交流，并且要用 CSMA / CD 协议。

**问题1：如果其中一个机器的网卡设置为全双工怎么办？**

好吧，这会使局面非常尴尬。很简单，因为其他 9 台机器都需要等待集线器空闲时才可以发送信息，如果有机器和它们同时发送信息，它们会认为发生了冲突，就会“闭嘴”。

但我们的全双工的机器“非常任性”，想什么时候发送信息就什么时候发送，毫不关心自己给别的 9 台机器造成的冲突。简而言之，这是一团糟。

更糟糕的是，如果那台全双工的机器正在下载很大的文件，它将不断传输信息，并阻止其他人发送信息，其他 9 台机器就不能工作了。

好在我们的网卡很智能，可以自己决定应该使用全双工还是半双工。因此，当网卡连接到集线器时，它将以半双工的方式运行；连接到交换机时，它将以全双工的方式运行。

## 问题2：如果我将一个集线器连接到一个交换机，会怎么样呢？

我们无法配置集线器，集线器就是要半双工的方式，它以总线拓扑的方式来运作，需要配合 CSMA / CD 协议才能很好地工作。

因此，如果把交换机与集线器相连，那么交换机将不得不去适应集线器的半双工的工作方式（交换机是会自己检测到某个端口连着集线器的，然后做出调整）。

事实上，也不是整台交换机都需要去适应集线器的工作方式，仅仅是交换机上和集线器连接的那个端口，这个端口需要以半双工的方式来运作，交换机上的其他端口则维持全双工的方式。

连接到交换机上的半双工机器，往往会受到很大的影响，比如因为每次有别的机器给它发送信息时，它就无法对外发送信息。而且，如果发生之前我们学过的 flooding（泛洪）的情况，帧通过广播方式发给除发送方以外的其他机器，那么这台半双工的机器也只能“闭嘴”。

幸运的是，这台机器是唯一受影响的，网络的其余部分将正常运行。当然了，对于使用这台机器的用户来说，就有点“哑巴吃黄连”了。

我们可以得出结论：**在大多数情况下，你不必为全双工还是半双工而操心。**