2017 CommNet 思考题文字部分

By mxl

1.

应用位于网络最高层，直接对用户提供服务；
业务位于网络中层，分为面向连接和无连接两类，为应用提供服务；
基本网络元素位于网络底层，是承载业务、执行物理传输的设备；
基本网络机制贯穿网络各层，负责网络资源分配、控制、安全保证等各项技术。

2.

二层交换机：连接多个子网，按 MAC 地址转发帧的数据链路层设备。
二层核心技术：根据源和目的节点的 MAC 地址，可用硬件快速实现存储、过滤和转发帧功能。
三层交换机：具备路由功能，并能以二层交换机技术转发数据的网络层设备。
三层核心技术：一次路由，多次转发。即可以调用路由功能建立完整的 IP 和 MAC 的映射(一次路由)，再使用二层交换技术完成转发(多次转发)。
二层交换机与路由器的异同：两者都是连接子网、实现数据转发的设备。二层交换机工作于数据链路层，仅凭 MAC 地址无法隔离广播域；路由器工作于网络层，能靠 IP 地址隔离广播域。
三层交换机与路由器的异同：两者都能连接采用不同路由协议的子网，选择路由路径并转发数据。三层交换机兼具路由器的功能和二层交换技术，可以以低成本更灵活地连接拓扑、分配子网带宽和配置信息资源。

3.

1） a）无效。因为集线器（HUB）无过滤作用，无缓冲区，只能向其它端口转发分组这样会使得网络负载越来越重，造成回路死锁。
2） a）有效。以太交换机具有交换机制，缓冲区以及转发机制。可以根据 MAC 地址对数据报进行过滤、隔离冲突域。交换机可以保存保存同一个分组而避免重复转发，从而避免回路死锁。
b）能。
c）设置交换机制：在规定时限内，统一分组在一个端口只能转发一次，否则将被丢弃。
3） a）有效。路由器可以根据不同目的 IP 地址选择不同的路由路径，选择不同端口进行转发，可以隔离冲突域以及避免回路死锁。
b）不能。IP 包有生命周期，每发生一次转发 IP 包的生存时间减 1，当生存时间为 0 时路由器将丢弃该包以避免永久的回路循环。

4.

网络性能优化角度：网络各层的参数存在关联、相互影响，跨层设计可以使网络结构更灵活、简单，网络适应性更强，还使网络控制模块能更全面、准确的了解网络状态，做出使整体的性能达到全局最优的控制决策，避免出现只有某几层局部最优或各层分别控制产生冲突的情况。但跨层设计会使控制机制更复杂、控制开销更大、优化目标难确定（效用函数）。

业务性能优化角度：网络各层服务最终需要使用户满意。而用户只关心体验质量（QoE），

而不在意网络如何提供服务。影响 QoE 的参数不局限于网络的某一层，因此采用跨层设计，不仅能根据业务特征简化或灵活调整网络结构，还能根据业务约束（约束条件）和网络状态调整网络性能以使用户获得满意的 QoE。但跨层设计不仅使控制复杂开销大，还需要全面了解业务的特征和用户需求的约束。

5.

严格分层：严格分层按照功能实现将整个网络分成多个层次，每个层次实现我们对这一层既定的目标（如物理层实现功率控制，链路层实现链路调度，网络层实现路由算法，传输层实现拥塞控制等等），层与层彼此独立，层之间以接口的形式互联。这样设计的优点是网络结构简洁明了，每一层的功能定位清晰，易于标准化，并且方便进行版本更新。这样设计的缺点是简化网络结构的同时也限制了网络结构，当我们针对某一项具体的性能指标进行设计时，往往由于网络结构的限制只能达到局部最优值，无法保证目前分层方式得出的局部最优值是全局最优，这一点在无线网络中尤为明显。

跨层设计：跨层设计抛弃了目前传统的网络分层，而是将网络当做一个整体进行考虑，对于不同的性能指标，将网络考虑成不同的模块结构。这样设计的优点是在整体上将问题考虑的更加全面，当我们确定需要主要满足的性能指标时，跨层设计从整个网络的角度提出最优化方案，这样设计得出的性能往往是（全局）最优的。这样设计的缺点在于不同性能指标对应的网络模块结构往往不同，这就极大的增加了网络结构的复杂度，使得网络在维护时面临的问题很大。

因此总的来说，分层与跨层的取舍其实网络效率与网络结构复杂度之间的折中。

7.

单通道协议中仅有暴露终端和隐藏终端两种问题终端，但在引入双信道协议后，问题终端得以进一步划分，以此图为例：

暴露发送终端：处于发送状态的暴露终端(初始通信对的发送端范围内，接收端范围外的终端)。如图，若 D、G 有终端处于发送状态，则为暴露发送终端。

暴露接收终端：处于接收状态的暴露终端。如图，若 D、G 有终端处于接收状态，则为暴露接收终端。

隐藏发送终端：处于发送状态的隐藏终端(初始通信对的发送端范围外，接收端范围内的终端)。如图，若 F、M 有终端处于发送状态，则为隐藏发送终端。

隐藏接收终端：处于接收状态的隐藏终端。如图，若 F、M 有终端处于接收状态，则为隐藏接收终端。

BAPU 协议是对 MACAW 协议向双信道的改进，其中 RTS，CTS，DS 在控制信道传输，DATA，ACK 在数据信道传输。假设该网络采用 BAPU 协议，那么：

暴露发送终端：以 D 为例，D 可以发送数据，但 D 发送的 DATA 与 B 发送的 ACK 会在 A 产生冲突，所以 BAPU 协议无法解决暴露发送终端问题。

暴露接收终端：以 G 为例，G 无法接收数据，同时 G 可以从控制信道将让 H 停止无谓的 RTS 请求，所以 BAPU 协议可以解决暴露接收终端问题。

隐藏发送终端：以 F 为例，F 无法发送数据，根据 RTS-CTS 握手机制，接收到 B 发送的 CTS 的 F 会终止发送，所以 BAPU 协议可以解决隐藏发送终端问题。

隐藏接收终端：以 M 为例，M 可以接收数据，但 N 发送的 DATA 与 B 发送的 ACK 会在 M 产生冲突，所以 BAPU 协议无法解决隐藏接收终端问题。

综上，BAPU 初步解决了暴露终端和隐藏终端你的问题，但并未完全解决。

11.
第一种方法，对于一个节点，当其完成一个数据包的发送后立即准许前继节点向其发送，这样可以加快响应速度，同时将负载均衡的分布在整个链路中，对于单一节点，不需要保有很大的缓冲区。
第二种方法，对于一个节点，当其确认一个数据包的成功发送后再准许前继节点向其发送，这样可以更好的保证数据传输的可靠性。对于单一节点，需要比较大的缓冲区来防止数据溢出。

两种方法相比，第二种需要比较大的内存。

当后继链路是卫星链路时，第一种方法比较适用，将后继链路的窗口设置相对大一些，可以达到比较好的传输效果。若使用第二种方法，由于卫星链路高时延的特点，等待包确认信息的时间太长，这需要发送节点保有很大的缓冲区以避免溢出，这样的设计是不经济的。

12.
a）$m \geq W$
因为特殊包的作用是告诉源节点当前目的节点正在等待哪个序号的包，相当于要通知源节点当前窗口中的哪个包发生了丢包，为了唯一确定丢包的准确位置，这要求窗口内所有包的取模结果不同，满足这样设计的窗口大小要求满足上述限制。

b）当发回的许可包含了按接收顺序的每个数据包的模m序号时，我们其实仅需要统计收回的序号数量就可以确认当前窗口中的第几个包出现接收错误，因此，这种情况下窗口大小与m的取值并无直接关系，所以会影响（a）中的回答。

c）不能，因为合适的窗口大小是目的节点根据网络情况反馈给源节点的，本质上是受目的节点控制的，如果源节点单独进行调整，那么接下来一段传输过程中目的节点需要重新调整使得窗口大小适应当前网络，因此独立调整窗口这种行为是没有任何益处的。

d）目的节点可以根据收到包标号，来确定源节点使用的窗口大小，然后通过减少许可发送的数目，来将窗口调节到被源节点使用的窗口大小以下。