Large Scale Q-in-Q — (1) Scalable address learning

Mick Seaman

One of the concerns expressed about the scalability of a Q-in-Q solution for Provider Bridged Networks is the amount of MAC address learning required in large networks. This note describes a **scalable address learning** algorithm – rules that reduce the total learning requirement and provide linear scaling with the number of customer attachments as the network grows – even if multipoint, rather than point to point services predominate.

Use of scalable address learning together with shared VLAN learning, as standardized in 802.1Q, allows very large fault tolerant provider bridged networks to be built without requiring any single bridge to learn more than a small fraction of the total number of addresses seen by the network. By way of example, the note discusses a Q-in-Q network design capable of backhauling 38 million subscribers from 1200 locations to routers located in either or both of two major metropolitan areas, without requiring any single bridge to learn more than 100,000 addresses and remaining comfortably within the limit of 4094 service VLANs. While such a network may not be a good idea, the fact that it can be constructed with Q-in-Q technology is interesting.

This note summarizes my July 2003 802.1 meeting presentation.

The Perceived Problem

The argument that a Q-in-Q network 'does not scale' because of the escalating MAC address learning requirement can be summarized as follows:

1) The purpose of increasing the size of a Provider Bridged Network is to increase the number of customer attachments.
2) The customer attachments for any given customer are likely to be spread right across the network[1].
3) So as the network grows the number of addresses to be learnt in the core of the network grows linearly with the number of customer attachments[2].
4) And as the size of the network grows the number of bridges in the core grows too, more or less linearly[3] with the number of customers.
5) So the network resource devoted to address learning (sum of MAC addresses multiplied by the number of places each address is learnt) grows as the square of the number of customers.
6) But value provided only grows linearly as the number of customer attachments[4].
7) So the network does not scale, i.e. the cost divided by the value provided increase as the network size increases[5].

The Remedy

Put simply the solution is not to learn addresses in bridges where learning them does not affect where frames are sent.

It is already known that there is no need to learn addresses from frames transmitted on point-to-point links on a point-to-point service LAN[6].

[1] From experience this is not quite true, the attachment points for a given customer tend to be clumped together. However this effect may only contribute a constant multiplier, so the main point of the argument remains.

[2] It is often erroneously assumed that all customers are likely the early adopters, whereas in fact the number of addresses per attachment is likely to decline dramatically with network build out. However the solutions proposed in this note make this a non-issue, so it can be safely left for another day and a realistic discussion of network sizing.

[3] For the purposes of this argument all that has to be established is that the core is not of constant size, so even if an architecture is found with core growth as logN the conclusion remains.

[4] This is a slightly difficult point since the attractiveness of the network to a particular customer can increase dramatically once it reaches all the places that customer wishes to attach. The adoption costs for a new network technology can be so high for some customers that they won't use it without near universality. However the point stands, since (a) once a customer is attached at all desired points, additions to the core can increase network resources costs while having yielding no benefit to that customer (b) the point has to be seen in relation to competing technologies without such apparent limitations.

[5] For the argument to have any

Given a source address SA of a frame that is classified as belonging to a VLAN R that uses a FID F and is received on a port P1, the general rules for learning SA in/with F are as follows.

If P1 is attached to a point-to-point link, SA is learnt if and only if there is at least one VLAN T (T may be the same as R) that uses F, for which:

a) P1 is in the Member Set for T[7] (i.e. frames classified as belonging to T may be transmitted through P1), and

b) A port P2 (different from P1) is also in the Member Set for T, and

c) Frames classified as belonging to T can ingress through P3 (different from P1 and P2).

If P1 is attached to a shared medium, SA is learnt if and only if the conditions for the point-to-point link apply or:

a) P1 is in the Member set for T, and

b) A port P2 (different from P1) is also in the Member Set for T, and

c) Frames classified as belonging to T can ingress through P1[8] or through a P3 (different from P1 and P2).

In theory, learning an address on P1 can affect the forwarding or frames belonging to a given T even if P1 is not in the Member set for a T, since it can cause discard of frames destined for the address while they might otherwise flood. However it is unlikely that there will be sustained traffic on a VLAN for an address which is simply unreachable on that VLAN – as it must be if P1 is in the spanning tree for T but does not allow frames in T to egress toward the address.

When there is no longer any VLAN T in FID F for which the above conditions hold for a given P1 then all the address entries in F that specify P1 can be deleted.

In addition to these learning rules it is necessary to prune the active topology of each service VLAN so that it comprises only the subtree of the spanning tree supporting the VLAN necessary to connect the customer points of attachment to that S-VLAN. GVRP was designed for this purpose, and is ideally suited to the task as it reprunes the trees after changes in the underlying physical topology have forced them to change.

Why this works

The proposed remedy is effective because the number of bridges that have to learn a given customer's addresses (that is the addresses for frames assigned to a given service VLAN) is now never more than the number of points of attachment[9] for that customer.

This is because the active topology of any given service VLAN is a tree all of whose leaves are customer points of attachment. Each branch that is added to the tree adds a customer point of attachment, and the worst case is that every branch is in a separate bridge[10].

So the address scaling problem of a exceedingly large bridged network is solved in an Provider Bridged Network[11] by recognizing that the latter is really just a large number of superimposed small networks, not one large mesh.

Examples

Figure 1 shows part of a fault tolerant provider bridged network with customer attachments for customers 1 thru 6. Bridges and bridge ports that do not attach to one of these customers have been omitted, and the six customers have been selected because all their service VLANs (one per customer) are supported by the same spanning tree instance [12].

Figure 2 shows just the active topology of the network, with the service VLANs supported by each bridge. Without scalable address learning, each bridge learns addresses on all the VLANs that pass through that bridge[13,14]. In particular note that the network root, in the upper left corner, learns addresses on all the service VLANs shown.

---

[6] Except possibly at the customer point of attachment to address the unlikely case when the customer is not learning them properly, but more likely to support diagnostics.

[7] If P1 were not in the member set for a given T it could still affect the forwarding behavior by causing frames to be dropped

[8] Frames attached to shared media links may not have to pass through the bridge at all, but through some other bridge attached to the same shared media.

[9] Minus two if learning is not required at the bridge port directly attached to the customer.

[10] A service VLAN that connects 3, 4, 5, or 6 customer points of attachment has 1, 2, 3, or 4 branches respectively.

[11] The same technique will, of course, work in an enterprise using VLANs. However in typical enterprise applications the stations using any particular VLAN tend to be more widely distributed so that most regions contain a few stations for the VLAN, and the learning savings are reduced. Further in a service provider network, points of attachment are provisioned (hopefully not manually) which provides a check of any concentration of learning in any particular switch. While the same check can be applied to a wireless station roving around the enterprise, it is likely that the depth of provisioning support will not be present even if an advanced security infrastructure is in place.

[12] This just simplifies the figures.

[13] This figure assumes that the VLANs have been pruned to the necessary subtree of the network. If the pruning has been done manually this can't be done without compromising the fault tolerance of the network, and learning from multicasts on each VLAN will occur in more bridges. Dynamic pruning (as provided by GVRP) is necessary to cut the learning down to this extent, and to lop-off unwanted branches that would otherwise defeat the scalable address learning algorithm.

[14] To be fair it should be noted that applying the well known rule for not learning on point-to-point S-VLANs removes the need for the example network root to learn on VLAN 1.

Figure 3 shows the learning to be carried out by each bridge[15] using scalable address learning. Note that the network root now only learns on two of the VLANs, while many of the bridges do not have to learn at all[16].

Figure 4 shows part of a ring network, with customer attachment points for service VLANs 1 thru 4. Each bridge is annotated with the VLANs that pass through it[17], and ordinarily each bridge would learn addresses on all the VLANs shown. Figure 5 shows the VLANs selected for address learning if scalable address learning is implemented.

It is tempting to characterize the difference in total learning requirements of figures 4 and 5 by calculating the latter as x% of the former. Unless an entire network is considered, this very much misses the point. In figure 5 additional bridges could be inserted, and ports added to existing bridges, without adding at all to the learning for the VLANs shown – unless the added bridges support customer attachments for those VLANs. The point is that scalable address learning scales because the learning requirement for a given service VLAN with a fixed number of attachments is now independent of the overall size of the provider network that supports many customers.

It is worth observing in passing that scalable address learning identifies all the cases for omitting learning that the two popular heuristics of 'don't learn on point-to-point' and 'only learn if the bridge is adding/removing traffic to/from the ring". In fact it performs slightly better because it identifies the attachment points for a VLAN that are nearest to the loop preventing cut in the active topology, and notes that learning does not have to take place there[18]. This is one of the cases where an algorithm that works on a general mesh works just as well as, or outperforms, algorithms that attempt to take advantage of the restricted topology of a ring[19].

Shared VLAN Learning

If a bridge only supports individual VLAN learning, a simplified version of the scalable address learning rules introduced above can be used. If VLAN ingress and egress is the same for each VLAN and port, these are equivalent to learning from a frame received on a bridge port attached point-to-point link only if a total of three or more of the bridge's ports participate in the VLAN.

However shared VLAN learning offers an even greater potential for cost effective scaling. A common technique is to use a pair of VLANs to connect routers to a large number of subscribers[20]. Each router transmits on VLAN 1 (say) and receives on VLAN 2, while each subscriber receives on 1 and transmits on 2. This arrangement ensures that the subscribers cannot communicate directly with each other, since the provider edge equipment will not receive any subscriber attempt to transmit on 1. Frames specifically addressed from a router to one customer will not be received by others, since the provider bridged network filters these[21].

Figure 6 illustrates part of such a network, a number of routers each marked R1 are connected to subscribers S1, while router R2 serve subscribers S2 etc. All the S1 subscribers are attached to a single POP, S2 subscribers to another POP etc. Frames addressed to the S* subscribers do not fan out to multiple bridges until they reach the bridges that distribute traffic in each POP, so the learning of each batch of subscribers is confined to the POP level. If we hypothesize a provider bridged network constructed for the task of backhauling subscriber traffic from 1200 locations each serving no more than 100,000 subscribers for a total subscriber population of 38,000,000 (average ~ 31,000 per POP) then a network of this form should be capable of meeting the need[22], without requiring more than 2400 VLAN Ids plus a few.

Deployment

I hope it is clear that scalable address learning bridges can be arbitrarily added to networks comprising bridges without this capability in a

[15] The learning requirement is not affected if other bridges and ports that do not provide connectivity for the S-VLANs shown are added to the network – provided that dynamic VLAN pruning is in operation.

[16] This is not to suggest that bridges that are incapable of learning be deployed. My experience is that attachment points for the typical high-bandwidth business user are geographically clumped and that providing connectivity across the street or for a block or two is a valued part of the total connectivity offering. Deploying a service that required each customer to have geographically sparse attachment points would not be a good idea.

[17] If the network is not satisfactorily pruned, multicast traffic on the VLANs shown could spill into other bridges as well.

[18] See the lowest instance of attachment for VLAN 4 in Figure 5 for an example.

[19] It is about time the networking community gave up on rings, except as a way of trenching route diverse fiber. The cute tricks of ring specific technologies should be confined to an introductory undergraduate class. Quite apart from the fact that ring reconfiguration is no quicker, and often slower, than mesh reconfiguration once considerations above the physical plant (address flushing and relearning for bridging, for example) are taken into account, the latest general topology protocols handle rings rapidly and well (RSTP, for example), and most of the much advertised benefits of rings are lost once cost effective (subtending) spurs are added and rings are interconnected.

[20] The 'customer' for the VLANs is the ISP, who uses the VLANs to provide connectivity to his/her customers, referred to here as 'subscribers'.

[21] Various methods are used to ensure that traffic is never seen by the wrong subscriber, destination address filtering prior to egress from the provider network is typical.

[22] Bandwidth issues not considered here. Restricting ourselves to 64 spanning tree instances and no more than 80 Gb/s per switch will yield about 270 kb/s per user full time, though more can be obtained linearly with additional equipment.

network, or indeed progressively substituted for those bridges to enhance network capacity. Existing bridges do not have to be made aware of the improvements made in their neighbors. It is however a requirement that all (or at least a preponderance) of the bridges in the network implement GVRP correctly[23] if the full benefits are to be realized.

Other Effects and Considerations

Optimizing out learning for the special case of point-to-point VLANs is a result of the normal operation of the scalable address learning algorithm. This means that the provider network bridges don't have to be specifically configured with the knowledge that one VLAN is point-to-point while another is multipoint or potentially multipoint. The issue of whether point-to-point is a distinct service from multipoint becomes simply an issue of distinguishing service offerings made to the customer, not an issue of configuring the network infrastructure.

Some care has to be taken in implementations to consider the effects of VLAN registration propagation and transient membership during network reconfiguration. Discussion of how to build a first class implementation is beyond the scope of this note, however even an inferior implementation should quickly settle down and benefit from scalable address learning.

[23] The additional learning requirement imposed by having bridges in the network that do not implement GVRP goes beyond an inability to support scalable address learning.

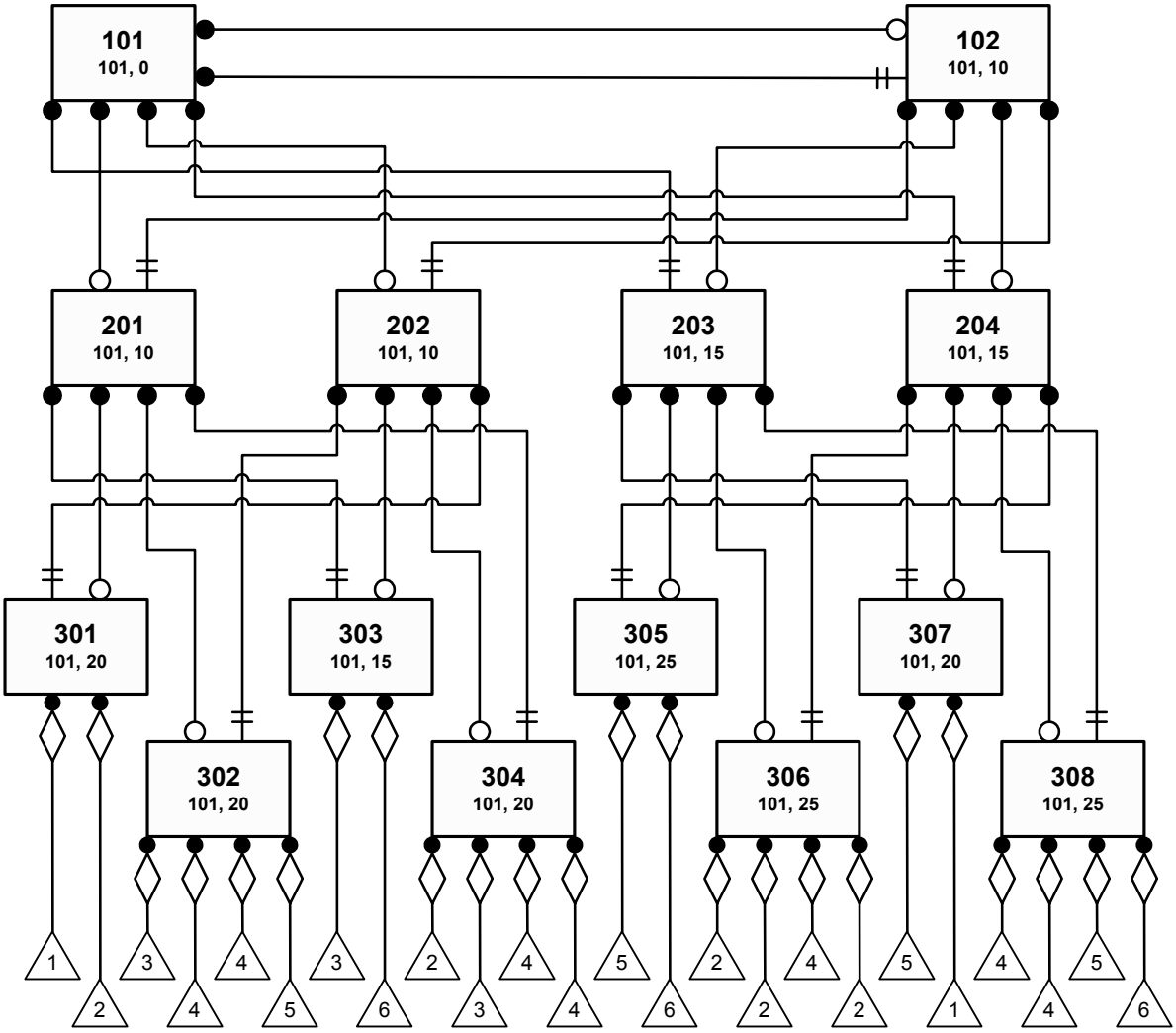

Figure 1 – Example structure provider network (part, with service VLANs 1 thru 6)

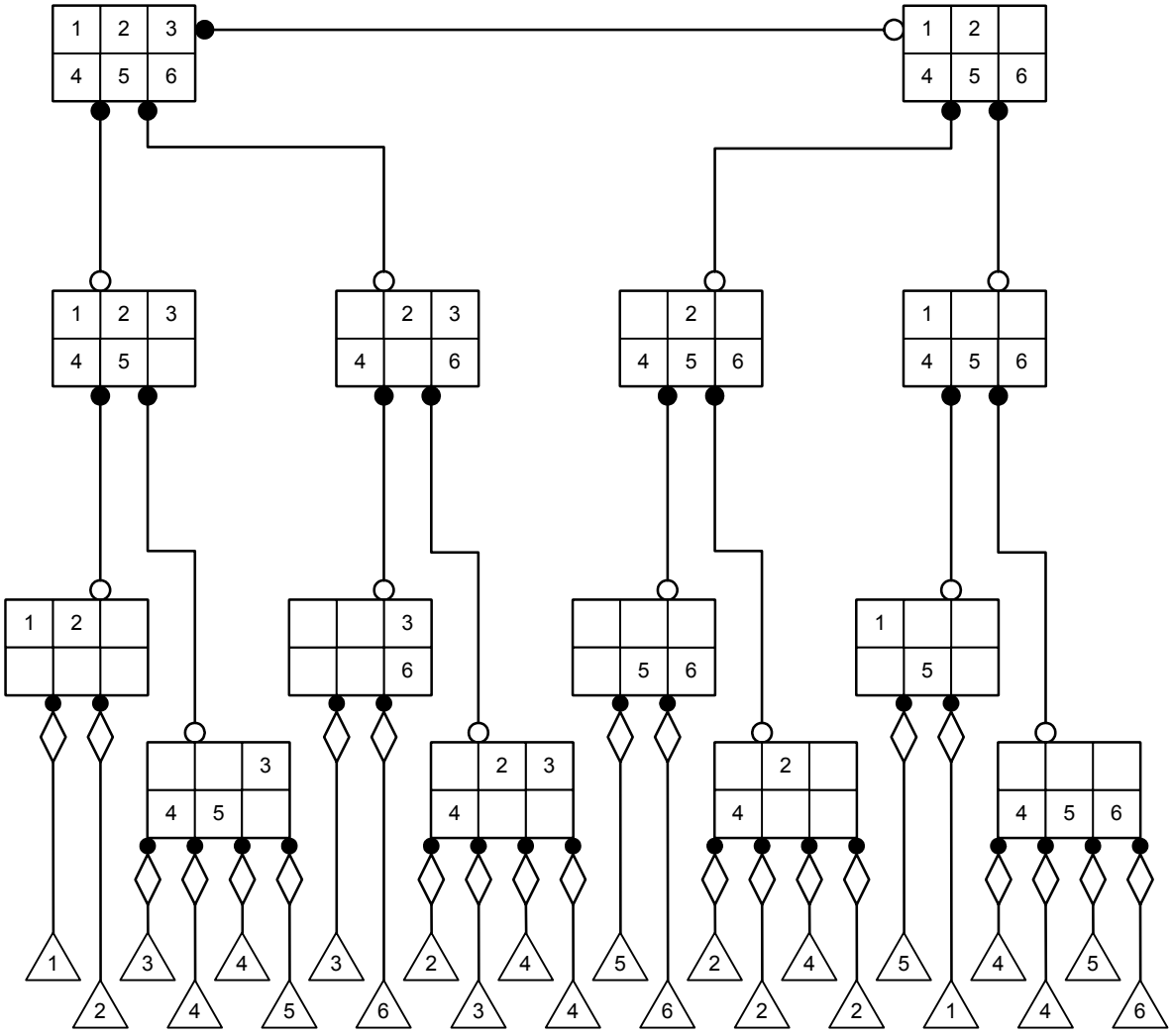

Figure 2 – Active topology of example structured network with VLANs

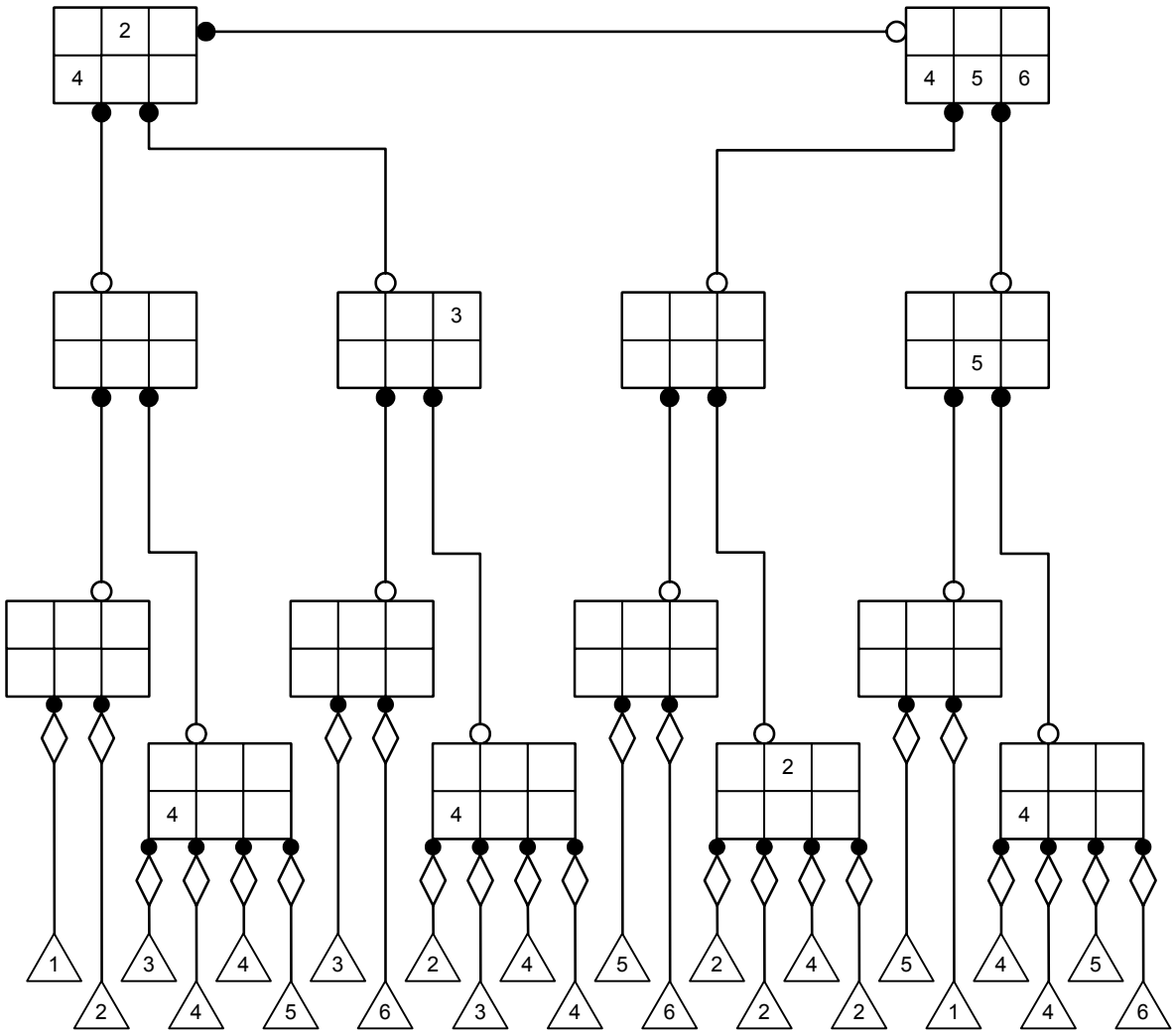

Figure 3 – Scalable learning requirement in the example structured network

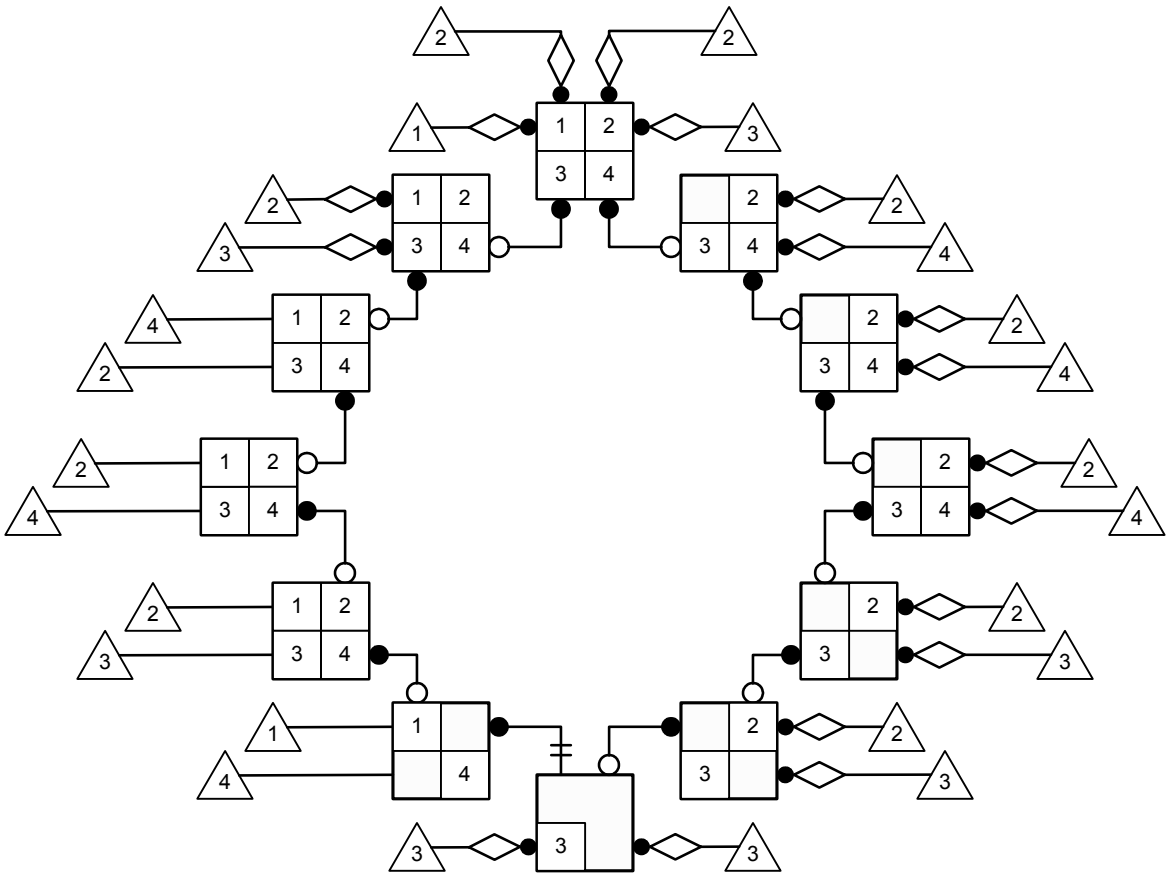

Figure 4 – Example ring network with VLANs

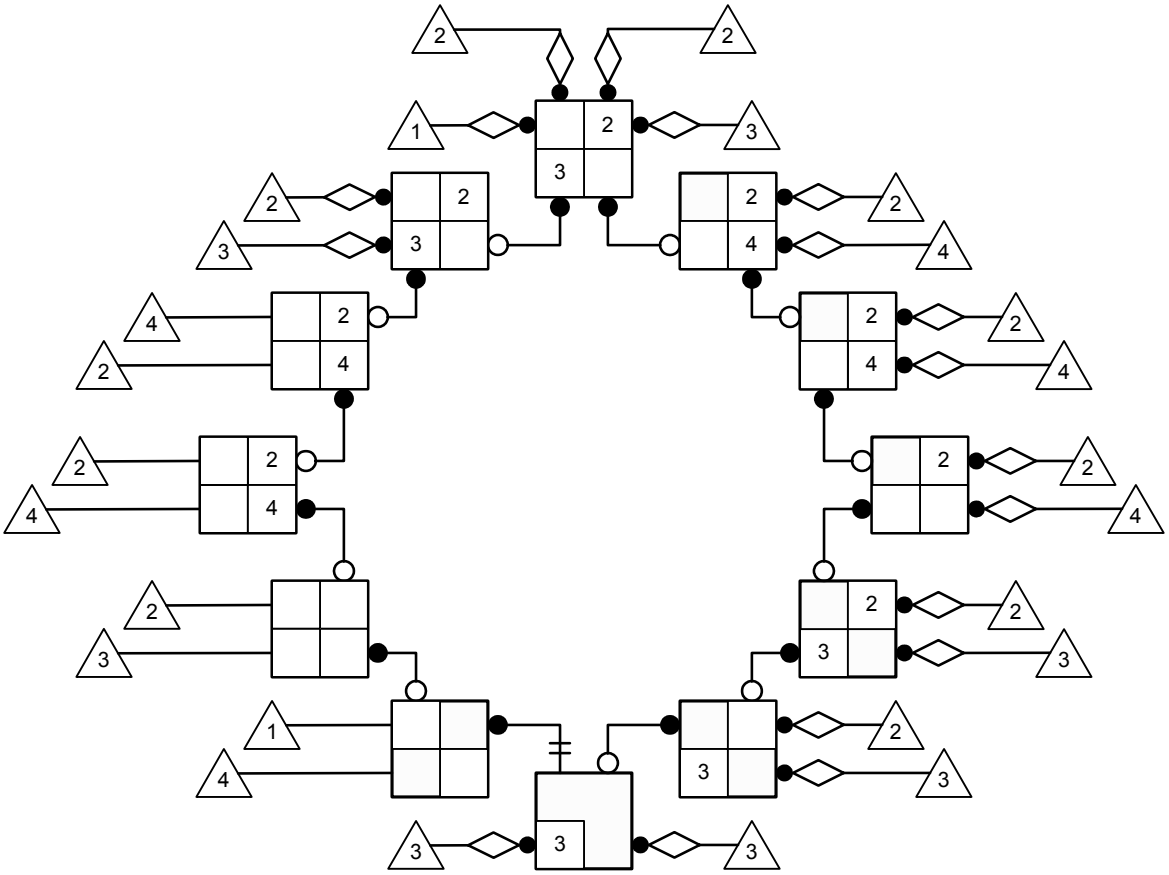

Figure 5 – Scalable learning requirement in the example ring network

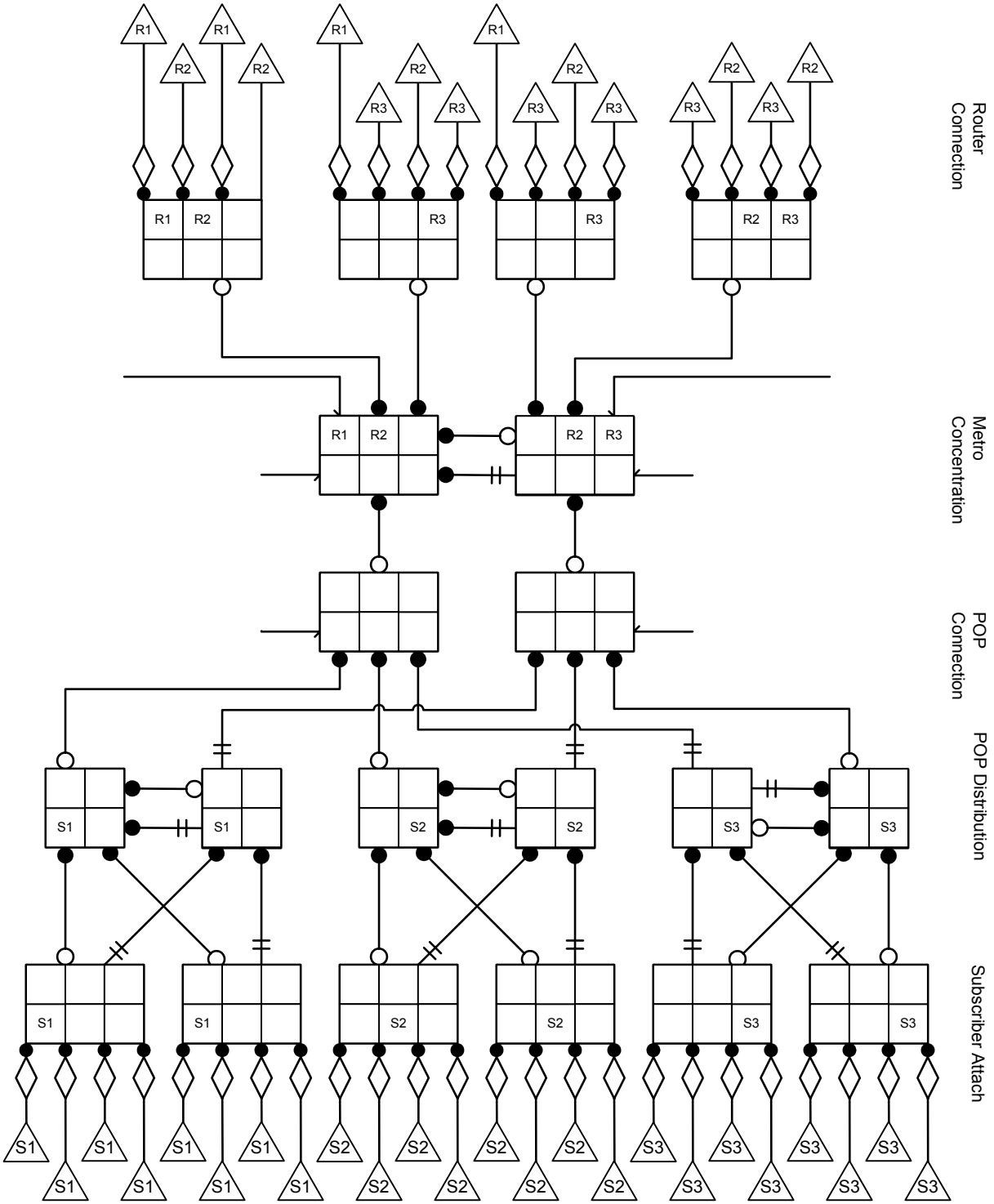

Figure 6 – Large scale subscriber network (part)