A Multiple VLAN Registration Protocol (MVRP)

Mick Seaman

This note proposes a replacement protocol for GVRP, provisionally named MVRP. MVRP communicates topology changes for each VLAN independently of the spanning tree supporting the VLAN. This allows many VLANs to use a single spanning tree without requiring a Bridge to relearn addresses for a given VLAN if a topology change does not change the Bridge Ports used to reach stations receiving frames for that VLAN.[a]

MVRP also support declarations and withdrawals of many VLAN registrations efficiently, communicating the information required for all 4094 VLANs in a single PDU[b,c]. Efficient operation for stations that only need to register declarations for a few VLANS is retained.

Though it is proposed as a replacement for GVRP[d], MVRP uses the general architecture of GARP and much of the state table design. Plug and play migration from GVRP to MVRP is possible.

Essential Background

This note assumes that the reader is familiar with GVRP as documented in 802.1Q-2003, and with the origination and propagation of topology changes as documented for RSTP in P802.1D/D4[1] and for MSTP in 802.1Q-2003.

There are scaling benefits to supporting many VLANs with each spanning tree. The number of independent paths in a network is typically far fewer than the number of VLANs. Even in large networks the major sources and sinks of traffic are organized around relatively few major hubs[2]. Shortest path routing to and from each hubs lies along a spanning tree rooted at the hub. Multiplexing VLANs onto trees can reduce the quantity of routing information to be exchanged by a factor of a hundred or more. Currently MSTP limits the number of spanning trees to 64, while the maximum number of VLANs is 4094.

Localizing Topology Changes

While assigning many VLANs to each spanning tree is convenient for network scaling and operation, it is desirable that addresses learnt in the Filtering Database for a given VLAN are only removed following a change in the network that affects that portion of the active topology used by the VLAN and not removed following changes in the spanning tree in parts of the network that only support VLANs for other customers.

Bridges using MVRP achieve this by not flushing address entries on receipt on the spanning tree change, but by using that change to solicit fresh declarations from the sources of VLAN registrations and marking those declarations as 'change declarations' as they propagate through the port that originated the spanning tree change, and subsequently using the change declarations to flush address entries for their associated VLAN – in the same way that a bridge that is not MVRP capable would treat a spanning tree topology change. Figures 1 thru 4 provide an example, and are described below.

---

[a] This supports network scaling by ensuring that the chance of disruption of an individual service instance relates almost entirely to the resources directly used to support that instance, and not to the size of the network as a whole.

[b] No larger than the maximum 802.3 frame size limit permitted in all environments

[b] The advantages of this are explained. The approach used is similar to that proposed by Norm Finn, extended to accommodate the additional per VID attribute events

[d] Throughout this note the term GVRP refers to GVRP as specified in Clause 11 of 802.1Q-2003, without any other suggested modifications. Depending on the upgrade arrangements MVRP could be GVRP version 2.

[1] Available standards participants from the 802.1 website, P802.1D/D4 is technically identical to 802.1D-2004.

[2] The SF Bay Area can be served by organizing around fewer than 30 significant hub locations. Even providing alternate trees for traffic engineering allows the area to be covered by 64 trees.

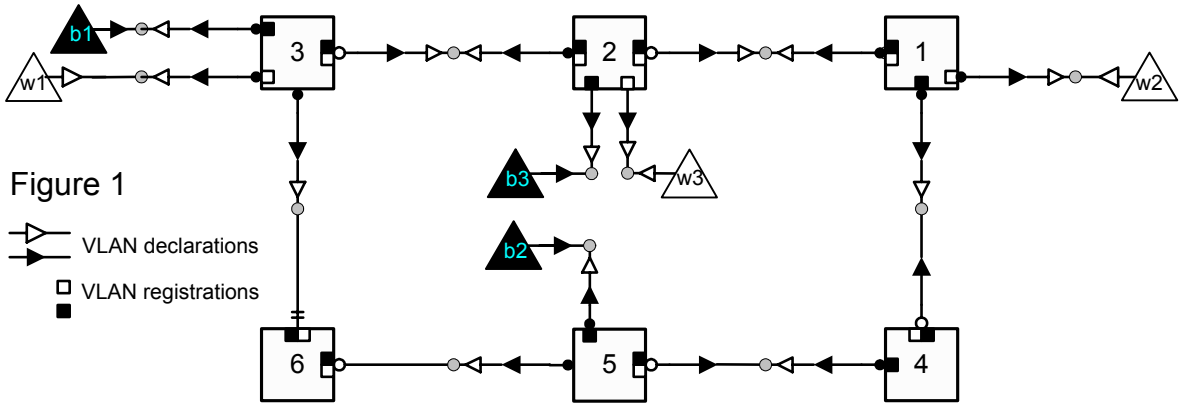

Figure 1

VLAN declarations

VLAN registrations

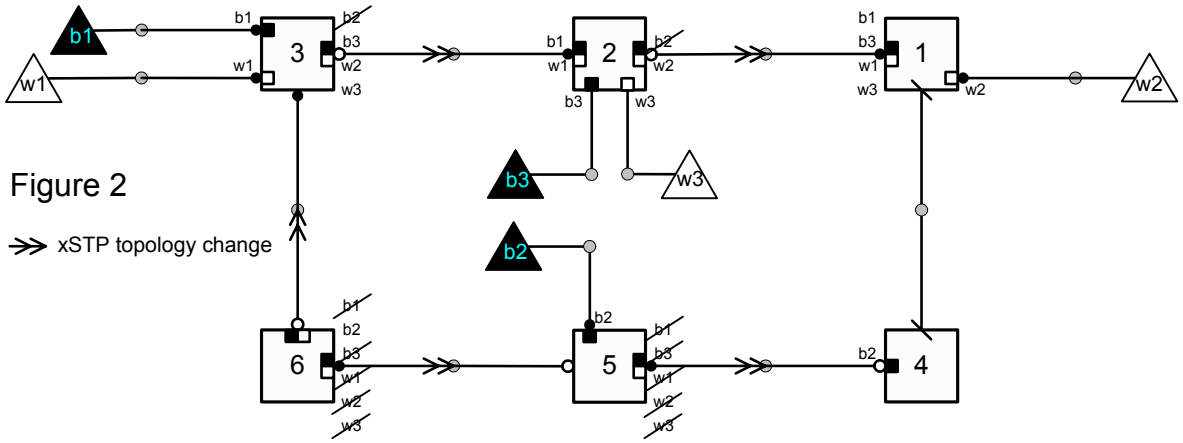

Figure 2

xSTP topology change

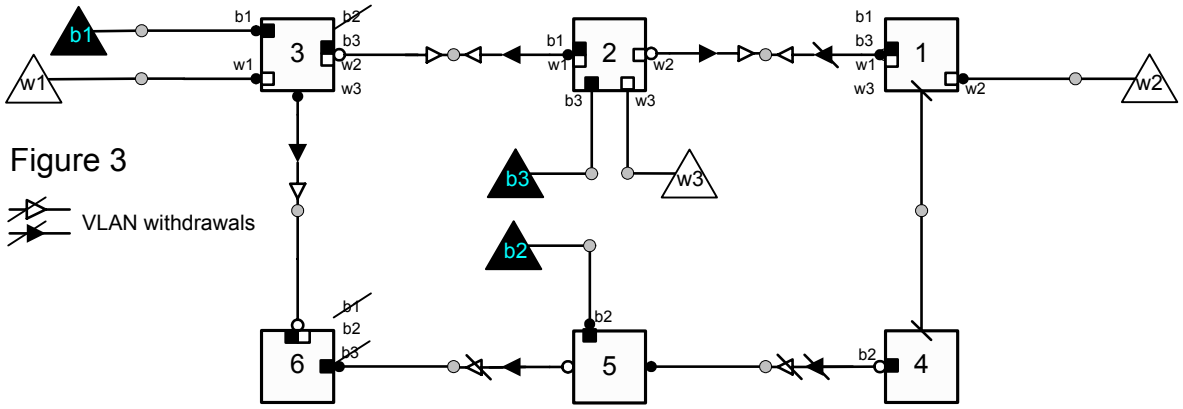

Figure 3

VLAN withdrawals

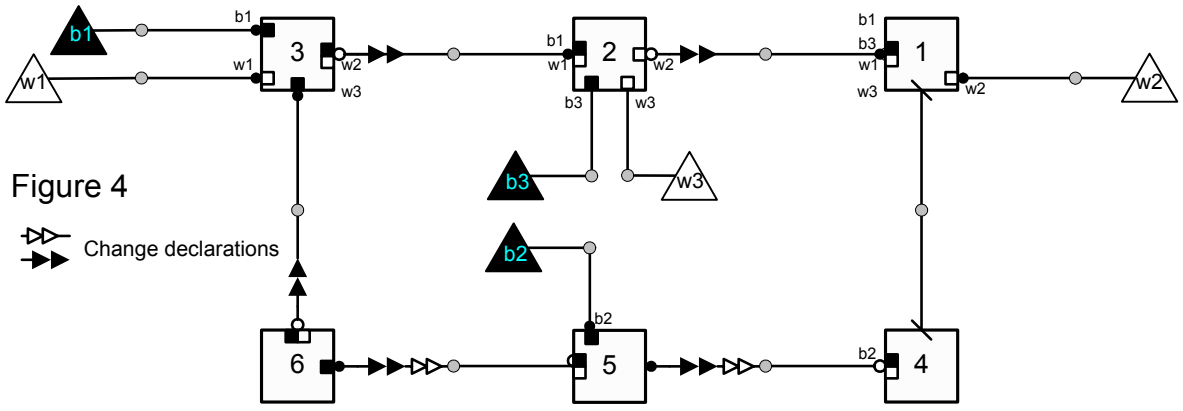

Figure 4

Change declarations

An Example

Figure 1 shows a simple network comprising a ring of bridges (1 thru 6) with groups of attached end stations or attached networks (b1, b2, b3, w1, w2, w3) for two VLANs, “black “ and “white”. The spanning tree port roles and states are shown (using the conventional notation) as is the propagation of VLAN declarations from each of the end stations, together with the resulting registrations at bridge ports[3].

Figure 2 shows the same network after the failure of link between bridges 1 and 4, together with the propagation of spanning tree topology changes[4]. While the port roles and states show the new active topology, the VLAN registrations and previously learnt address information are out of date at this point. The information that needs to be removed to restore full connectivity is struck through, while the rest of the learnt information is to be preserved if at all possible.

With MVRP each bridge does not flush learnt information directly on receipt of the spanning tree topology change. Rather the change is used to solicit fresh declarations from those bridges that have end station registrations or administratively Fixed registrations. Figure 3 shows those fresh declarations, together with the declaration withdrawals originating from bridge 4. When a VLAN is no longer registered on a port, addresses for that VLAN are no longer remembered on the port, so some of the now incorrect learnt information is removed.

Figure 3 only shows propagation of the fresh declarations up to the point that they pass through the port on bridge 6 that was previously Discarding and is now Forwarding. When MVRP propagates a declaration through such a port it marks it as a Change Declaration, as illustrated in Figure 4 by the double headed arrows for the black and white declarations. When a bridge receives a change declaration on a port, it removes address information for that VLAN from all the other ports. In other words the MVRP bridge treats the change declaration just as an existing bridge would treat a spanning tree topology change.

It needs to be noted that this example has been kept deliberately simple, and that a larger network would have additional spurs off the ring serving black VLAN (from any of bridges 1 thru 6) and the white VLAN (from any of the bridges 1 thru 3). As the figure stands, application of the enhanced filtering utility criteria (802.1ad/D2) further enhanced to special case attached networks could be used to remove the learning requirement on all ports except those for the attached networks.

[3] Anyone puzzled by the registrations shown on bridge 6, should recall that registration is unaffected by the spanning tree state of a port, but propagation only occurs between Forwarding Ports.

[4] As specified for

Encoding

MVRP communicates all the possible information, for all 4094 VLANs if necessary, from a protocol participant in a single legal size PDU. It is worth examining the argument for such information packing.

GVRP[5] takes 4 octets[6] per VLAN for which an attribute event[7] is to be communicated. If events are to be signaled for all 4094 VLANs, 16376 octets are required – 11 frames[8]. The more frames that are required to carry information, the greater the chance that a participant will propagate only part of the information to another bridge port. In a network comprising a number of bridges with a large number of ports, this effect can fragment the packing of information hop by hop. Implementations that delay or backoff on subsequent transmissions can reduce the fragmentation effect, but will slow network reconfiguration after failure[9]. Reducing the number of frames does not solve the problem, but helps a great deal.

Norm Finn has suggested an encoding where information for every VLAN is present, in order. The number of distinct attribute events for each VLAN in this scheme adds one, for ‘nothing to be said’, to the essential set for shared media[10], so GVRP requires five code points for each VID. Since 5^{4094} is less than $2^{8.1500}$ all possible combinations of VID attribute events can be represented in one PDU. Successively dividing a 1500 octet number by 5 to extract the remainder, and thus decoding the event for the next VID is a little tedious, so a slightly less efficient packing of information for N VIDs in M octets is used. Since $5^3 < 2^8$, and $5^{13} < 2^{32}$ we can encode the events for 3 VIDs in a single octet, or for 13 VIDs in a 32 bit word. The former seems preferable, and allows us to pack all 4094 VID events into 1365 octets. An obvious encoding multiplies the event code for the first VID of a three VID sequence by 5^0 (i.e. 1), the second by 5, and the third by 25, and adds the results to give an unsigned number that is encoded in the octet in the ordinary way.

A similar encoding lets us pack 6 code points for each of 3 VIDs into an octet, or 7 code points for each of 11 VIDs into a 32 bit word. The latter uses 1492 octets to encode the MVRP information for 4094 VLANs.

Different bridge implementations playing different roles in a provider network will of course have different scaling concerns. It is highly desirable that MVRP encoding not unduly

[6] GARPs encoding rules are flexible enough to allow less efficient representations of this data.

[7] Attribute events are defined so that only one occurs at a time.

[8] 802.3 frames for all environments.

[9] Each would like to have all the information from all its ports before it transmits, but that requires the next hop to wait even longer, and so on.

[10] Empty, JoinEmpty, JoinIn, Leave

burden[11] bridges or networks that only need to encode information for small numbers of VLANs. One way to do this is to encode non-null VID events in blocks, each prefaced by the first VID. Rather than using a length count, which has to be retroactively filled as the potential events are scanned, the compacted encodings allow space for escape values to signal 'no more'. A next block can then be encoded, prefaced with the first VID with a non-null event[12,13].

This note proposes the 7 code point/11 VID/4 octet encoding with the escape value 0xFFFF to indicate 'end of VID event sequence'[14]. The end of sequence marker is followed by the VID, in two octets, that starts the next sequence. If those two octets are both zero the end of the PDU has been reached. Five code points are required for GARP events[15], with an additional two to support topology change notification as it is necessary to signal both JoinTc and JoinEmptyTc.

Point-to-point media

The foregoing assumes that MVRP is operating on shared media, with the accompanying challenges of efficiently determining when all declarations of an attribute have been withdrawn, and of ensuring that a participant's own applicant does not interfere with its registrar. Things are really much simpler on point to point media. This raise two questions. First, is there any point in specifying the shared media solution. Second, if the shared media solution is specified, how different should the point-to-point solution be.

There are few, if any, true shared media remaining, so the need for such a solution arises from the potential requirement to run MVRP over a point to point infrastructure that simulates shared media, and does not itself run GVRP/MVRP. The reasons for the latter vary widely. Since, in the real world, upgrades to the multiple products that form a system cannot be synchronized, simply deploying MVRP would seem to demand a shared media solution.

When an applicant withdraws a declaration on a point-to-point link, the peer registrar can remove the registration immediately. There is no need to wait for a timer. A simpler set of states can be used, and applicants could just send Join or Leave events. Registrars don't have to say anything, apart from sending the occasional LeaveAll[16] to recover from lost messages.

An earlier draft of this note considered using a different encoding on point-to-point links. However 4 distinct code[17] points are required in any case, the obvious improvement in code point packing leaves us without an escape code. The suggestions is to use the same event codes for point-to-point as for shared media, while the state tables are changed to permit instant withdrawal of attribute registrations. A further advantage is that if MVRP discovers that the media is not point-to-point, but really shared, or alternatively that the number of participants has dropped to two, then the behavior can be changed on the fly while retaining the state information. There are no messy decisions to be made about receiving a point-to-point format PDU on a port thought to be attached to shared media, or vice versa[18].

[11] Because someone will invent a private protocol that is "simpler".

[12] Obviously a little look ahead is required to check there are enough null event VIDs to permit a block to be ended and the next started without increasing the frame size.

[13] I don't know whether Norm was thinking of this or not.

[14] There may be advantages to allowing all numbers above 7^{11}-1 to be treated as end of sequence.

[15] See above.

[16] In one version, applicants send state for all possible VLANs all the time, so no LeaveAll polling is required. On balance this is an unnecessary load on bridges away from the core, and could prompt private protocol development.

[17] Join, JoinWithTopologyChange, Leave, and Null.

[18] Of course it would be possible to extend the state tables to accommodate both p-to-p and shared formats and their codes, with state tables based on what the receiver chooses to believe about the media. However that is probably the worst of all worlds, and to high a price to pay for the simplified p-to-p format.