A Multiple VLAN Registration Protocol (MVRP)

Mick Seaman

This note proposes a replacement protocol for GVRP, provisionally named MVRP. The goal is to support declarations and withdrawals of many VLAN registrations efficiently. The information required for all 4094 VLANs can be communicated in a single PDU[a]. MVRP also communicates topology changes for each VLAN independently of the spanning tree supporting the VLAN[b]. Though it replaces GVRP, MVRP continues to use the general architecture of GARP and much of the state table design.

The first part of this note is a general discussion of the design decisions that contributed to MVRP. The second part is, or is intended to be, a proposed new 802.1Q clause specifying MVRP. Experience has shown that a complete standards grade specification is required to fully debug a proposal of this sort . However it is anticipated that many changes and much input will be required before a new clause could be agreed.

One frame says it all

The desire to ensure that the information for all 4094 VLANs be communicated in a single legal size PDU by a protocol participant has a very significant effect on the MVRP design. It is worth examining the argument for information packing.

GVRP[1] takes 4 octets[2] per VLAN for which an attribute event[3] is to be communicated. If events are to be signaled for all 4094 VLANs, 16376 octets are required – 11 frames[4]. 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[5]. 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[6], so five code points are required 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 all 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 burden[7] 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

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

[b] The advantages of this are explained.

[1] Throughout this note the term GVRP refers to GVRP as specified in Clause 11 of 802.1Q-2003, without any other suggested modifications.

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

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

[4] 802.3 frames for all environments.

[5] 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.

[6] Empty, JoinEmpty, JoinIn, Leave

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

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[8,9].

This note proposes the 6 code point/3 VID/1 octet encoding with the escape value 0xFF to indicate 'end of VID event sequence'[10]. 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[11], the sixth is used to support topology change notification.

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. They include operational aversion, relative priorities, intellectual disagreement, the timing of product cycles, and ignorance[12]. 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[13] 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[14] points are required in any case, and any improvement in code point packing into the PDU beyond the 3 VLANs per octet would leave 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[15].

Topology Changes

The number of independent paths in a provider bridge network is typically much less than the number of customer VLANs. The current specification of MSTP limits the number of spanning trees to 64, while the maximum number of service VLANs is 4094.

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

Accordingly, when MVRP is used, the change in the spanning tree topology is not used directly to trigger removal of learnt information in a bridge receiving a topology change notification. Rather the spanning tree protocol topology change notification inhibits the maintenance of VLAN declarations by bridges that are simply passing the declaration along remote from its original source while at the same time soliciting declarations or withdrawals of declarations from the original sources. This ensures that declarations that have been propagated from sources that no longer lie along the same path in the active topology will be removed.

As each of the solicited declarations reaches a bridge the inhibition on propagation and maintenance is removed so the declaration is propagated once more. If no declarations are received but declarations are withdrawn on all

[8] 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.

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

[10] There may be advantages to allowing all numbers above 6^3-1 to be treated as end of sequence.

[11] See above.

[12] Not necessarily in that order.

[13] 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.

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

[15] 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.

ports other than a given port a withdrawal of the declaration is transmitted on said given port. By this means the declarations propagated in the network reflects the reachability of stations using the current active topology of the network rather than the topology prior to the spanning tree change. When a propagated declaration passes through a bridge port whose role in the active topology has changed from Discarding to Forwarding it is marked as a change declaration. When a change declaration for a VLAN is received by a bridge port the addresses learnt for the other ports for that VLAN are removed. This ensures that communications to stations on that VLAN are not inhibited by information learnt prior to the change. If all the stations receiving frames for a given VLAN are on the same side of any prior cut in the active topology introduced by the spanning tree to prevent loops and are also on the same side of a new cut in the active topology then the mechanisms described ensures that bridges connecting those stations do not remove learnt addresses for that VLAN.