CCAMP working Group
Internet-Draft
Expires: January 9, 2006

G. Bernstein
Grotto Networking
D. Caviglia
Marconi
R. Rabbat
Fujitsu

July 8, 2005

Operating Virtual concatenation (VCAT) and the Link Capacity Adjustment Scheme (LCAS) with Generalized Multi-Protocol Label Switching (GMPLS)

draft-bernstein-ccamp-gmpls-vcat-lcas-00

Status of this Memo

Copyright Notice

Abstract

This document describes the use of the Generalized Multi-Protocol Label Switching (GMPLS) control plane in conjunction with the Virtual Concatenation (VCAT) layer 1 inverse multiplexing mechanism and its companion Link Capacity Adjustment Scheme (LCAS) which can be used for hitless dynamic resizing of the inverse multiplex group. These techniques apply to the Optical Transport Network (OTN), Synchronous Optical Network (SONET), Synchronous Digital Hierarchy (SDH) and Plesiochronous Digital Hierarchy (PDH) signals.

Table of Contents

1. Overview of VCAT and LCAS

Virtual Concatenation (VCAT) is a standardized layer 1 inverse multiplexing technique that can be applied to OTN [5], SONET [2], SDH [1], and PDH [4] component signals. By inverse multiplexing we mean a method that combines multiple links at a particular layer into an aggregate link to achieve a commensurate increase in available bandwidth on that aggregate link. More formally, VCAT essentially combines the payload bandwidth of multiple path layer network signals (or trails) to support a single client (e.g. Ethernet) layer link. Other well known standardized inverse multiplexing techniques include Multi Link PPP [6] and Ethernet's Link Aggregation mechanism as documented in chapter 43 of [7].

One of the main differences between VCAT and the other mentioned inverse multiplexing standards is that VCAT works at layer 1 rather than at the data link layer, i.e., VCAT works with "circuits" and the others with layer 2 packets. This can be important when considering its capabilities or limitations.

1.1 SDH/SONET VCAT signals and components

In the following we will use SDH terminology rather than both SONET and SDH terminology. In SDH Virtual Concatenation (VCAT) can be applied to the following component time division multiplex (TDM) signals referred to as Virtual Containers (VCs) (and not to be confused with virtual circuits):

VC type	VC bandwidth	VC payload
VC-11	1 664 kbit/s	1 600 kbit/s
VC-12	2 240 kbit/s	2 176 kbit/s
VC-2	6 848 kbit/s	6 784 kbit/s
VC-3	48 960 kbit/s	48 384 kbit/s
VC-4	150 336 kbit/s	149 760 kbit/s

Extracted from table 6-1 of [1].

Table 1: Permissible SDH VCAT components

Note that when reading the VCAT and LCAS references the term "frame" is generally used to describe the repetitive structure of TDM signals

and not to describe a layer 2 packet. To simplify high speed aggregation of these signals, only like component signals can be aggregated into a VCAT group. The aggregate signals are named and characterized in Table 2 extended from table 11-4 of [1].

VCAT group	X	Capacity	In steps of
VC-11-Xv	1 to 64	1600 Kbit/s to 102 400 Kbit/s	1 600 Kbit/s
VC-12-Xv	1 to 64	2176 Kbit/s to 139 264 Kbit/s	2 176 Kbit/s
VC-2-Xv	1 to 64	6784 Kbit/s to 434 176 Kbit/s	6 784 Kbit/s
VC-3-Xv	1 to 256	approx. 48 Mbit/s to 12.5 Gbit/s	48 384 Kbit/s
VC-4-Xv	1 to 256	approx. 149 Mbit/s to 38.3 Gbit/s	149 760 Kbit/s

Table 2: SDH VCAT Signals

Since VCAT is an inverse multiplexing technique, SONET/SDH transport network nodes do not need to support these VCAT signals explicitly since it is the job of the VCAT end systems to reassemble the aggregate signal. The only requirement on the SONET/SDH network is to be able to transport the individual component signals, i.e., the VCs of Table 1.

1.2 PDH VCAT signals and components

VCAT can be applied to the following PDH signals as specified in reference [4]:

Common signal name	Signal Rate
DS1	1544 Kbit/s
E1	2048 Kbit/s
E3	34 368 Kbit/s
DS3	44 736 Kbit/s

Similar to the SONET/SDH case these component signals can only be combined with like signals to produce aggregates. For some reason the virtual concatenation groups of the PDH signals were not given unique designations in [4] so we shall adopt a similar notation to the SDH VCAT signals for the permitted PDH VCAT signals that follow.

pseudo name	X range	Approx. capacity
DS1-Xv	1 to 16	X*1533 Kbit/s
E1-Xv	1 to 16	X*1980 Kbit/s
E3-Xv	1 to 8	X*33856 Kbit/s
DS3-Xv	1 to 8	X*44134 Kbit/s

Table 4: Standardized PDH VCAT signals

1.3 OTN VCAT signals and components

Concatenation in the optical transport network (OTN) is realized by means of virtual concatenation of Optical Channel Payload Unit (OPU) signals. OPUk signals (k=1, 2, 3) can be concatenated into OPUk-Xv aggregates with X= 1,..., 256. The aggregate signals are named and characterized as follows (Table 5 is taken from Table 6-3 G.8012).

OPU Type	OPU payload (kbit/s)	In steps of (kbit/s)
OPU1	2488320	
OPU2	238/237x9953280 ~ 9995277	
OPU3	238/236x39813720~40150519	
OPU1-Xv	2488320 to 637009920	2488320
OPU2-Xv	~9995277 to ~2558709902	~9995277
OPU3-Xv	~40150519 to ~10278532946	~40150519

Table 5: Standardized OTN component VCAT signals

Note that the last row in Table 5 is not a mis-print. Reference [5] does indeed permit the virtual concatenation of up to 256, 40Gbps, ODU3 signals to produce an aggregate link, a ODU3-256v, with a capacity of over 10Tbps! At the time of this writing the authors do not currently know of any actual implementations, but it should be noted that the standard is quite "future proof".

1.4 VCAT Capabilities and Limitations

VCAT performs inverse multiplexing by octet/byte de-interleaving of the encapsulated client bit stream. As such it operates below the packet/frame level. Each frame/packet will therefore "travel" over all members of the VCAT group, and a fault in any of those members hits every Xth byte in each packet/frame. With LCAS the failed member is temporarily taken out of the service providing set of the VCAT group, until the fault is repaired. Due to this octet/byte de-interleaving VCAT introduces an insignificant processing delay into the transmission path. The propagation time for the aggregate signal will correspond to that of the longest component signal.

Figure 1 illustrates how incoming client traffic, in this case an Ethernet frame, is transported via VCAT in a transport network. The incoming Ethernet frame -for the sake of simplicity only six bytes of the frame are depicted- is inverse-multiplexed by VCAT into three different VCAT members. In Figure 1 the incoming Ethernet frame is spread across the three VCAT members, that is, bytes 1 and 4 are carried by VCAT member number 1, bytes 2 and 5 by member number 2 while bytes 3 and 6 by member number 3. In the case of a failure of VCAT member 2 bytes 2 and 5 are lost and thus it is not possible to rebuild the original incoming Ethernet frame.

```
                             Transport Network
                           <--------------------->
                         /|       b1    b4       |\
                        | |----------------------| |
                        | |    VCAT member #1    | |
                        | |                      | |
                        | |                      | |
       Eth Link         | |       b2    b5       | |       Eth Link
   =====================| |----------------------|
                                                   |=====================
    +--+--+--+--+--+--+ | |    VCAT member #2    | | +--+--+--+--+--+--+
    |b1|b2|b3|b4|b5|b6| | |                      | | |b1|b2|b3|b4|b5|b6|
    +--+--+--+--+--+--+ | |                      | | +--+--+--+--+--+--+
   Ethernet Pkt         | |       b3    b6       | | Ethernet Pkt
                        | |----------------------| |
                         \|    VCAT member #3    |/

                         VCAT                    VCAT
                        Ingress                 Egress
```

Figure 1: VCAT Inverse Multiplexing

With any inverse multiplexing technique two important issues come up: (a) how packet reordering is prevented, and (b) delay compensation limits. For example Ethernet's link aggregation scheme prevents reordering by restricting "conversations" to a single link. This means that the total aggregate bandwidth is not available to a single flow. MLPPP and VCAT prevent reordering in a way that imposes no limits on the bandwidth delivered to a single flow. Since VCAT works with circuits it doesn't have to deal with queueing induced differential delays between components. In fact, since most circuit switched technologies have very low switching latency most differential delays experienced by VCAT component signals are due to propagation. The maximum differential delays that can be accommodated by the standards is given in Table 6. Actual implementation can choose to provide much less differential delay compensation and frequently do so to save on memory requirements.

VCAT Signal	max diff. delay
VC-m-Xv	256 msec
DS1-Xv	384 msec
E1-Xv	256 msec
E3-Xv	255 msec
DS3-Xv	217 msec
ODU1-Xv	411 sec
ODU2-Xv	102 sec
ODU3-Xv	25.4 sec

Table 6: Differential Delay Limits

As mentioned in [8] the ability to compensate for 256msec of differential delay compares favorably with the circumference of the earth and some rather paranoid disjoint paths. The theoretical differential delay compensation limits for the OPUk, last three rows of Table 6, are in far excess of that needed for any terrestrial applications. However it is the natural outcome of future proofing the VCAT mechanism for OPUk signals via the allocation of the equivalent of a 24 bit frame counter which can also be used by future higher speed signals without modification to meet the need for 256 ms of delay compensation.

1.5 The LCAS Protocol

The Link Capacity Adjustment Scheme for VCAT signals is a protocol for dynamically and hitlessly changing (i.e., increasing and decreasing) the capacity of a VCAT group. LCAS also provides survivability capabilities, automatically decreasing the capacity if a member of the VCAT group experiences a failure in the network, and increasing the capacity when the network fault is repaired. LCAS, itself, provides a mechanism for interworking between LCAS and non-LCAS VCAT end points. VCAT does not require LCAS for its operation.

We find analogous mechanisms in other inverse multiplexing technology such as the Link Control Protocol (LCP) used in MLPPP [6] and the Link Aggregation Control Protocol (LACP) used in Ethernet Link Aggregation [7]. It needs to be emphasized that none of these

mechanisms are responsible for establishing the component links. Indeed, these protocols run over the component links themselves. Hence LCAS functionality does not overlap or conflict with GMPLS' routing or signaling functionality for the establishment of component links or entire VCAT groups. LCAS instead is used to control whether a particular component signal is actually put into service carrying traffic for the VCAT group.

Although we are used to PDH and SONET/SDH signals being bi-directional, LCAS actually works on unidirectional components in a VCAT group with the proviso that there is at least one return component for conveyance of LCAS messages. As viewed from LCAS' point of view the source end of each component can have the following states:

State	Explanation
IDLE	This member is not provisioned to participate in the concatenated group.
NORM	This member is provisioned to participate in the concatenated group and has a good path to the sink end.
DNU (Do Not Use)	This member is provisioned to participate in the concatenated group and has a failed path to the sink end.
ADD	This member is in the process of being added to the concatenated group.
REMOVE	This member is in the process of being deleted from the concatenated group.

Table 7: LCAS/VCAT component states

LCAS provides for graceful degradation of failed links by having the sink end report back the receive status of all member components. In the case of a reported member failure, the source end will stop using the component and the source end will send an LCAS message to the sink end that it is not transmitting data on that component. The

worst case notification times, not including propagation delays, for the different VCAT signals discussed here are given in Table 8. These values were obtained from [1] and [5], and reverse engineered from information in [4].

VCAT signal type	Notification Time
VC-11-Xv, VC-12-Xv, VC-2-XV	128 msec
VC-3-Xv, VC-4-Xv	64 msec
DS1-Xv	96 msec
E1-Xv	64 msec
E3-Xv	2 msec
DS3-Xv	1.7024 msec
OPU1-Xv	1.567 msec
OPU2-Xv	390 usec
OPU3-Xv	97 usec

Table 8: LCAS Notification times for Member Status

2. VCAT/LCAS Scenarios

In this section we list a number of VCAT/LCAS usage scenarios. We will evaluate the applicability of GMPLS to these scenarios in Section 3 and for those scenarios that GMPLS does not currently support we describe possible GMPLS extensions in Section 4. These scenarios can easily form the basis for formal requirements, however at this point in time we list the scenarios and can later evaluate which of these must be supported. Note the term "component" signal in the text is used as a simplified notion to the more formal concepts of VC-n, ODUk, and PDH termination function as well as VC-n, ODUk and PDH path/trail.

2.1 Discovery of Enabled End Systems

Discovering VCAT: VCAT sources can only communicate with VCAT capable sinks. Hence the VCAT capabilities of a PDH, SDH, or OTN path termination points need to be known.

Discovering LCAS: LCAS offers additional functionality between VCAT capable sources and sinks. Hence the LCAS capabilities of VCAT enabled path termination points can be useful to know in advance of component signal setup.

2.2 Client to End Point Mappings

Fixed Client to End point Mapping: Per client signal there is a VC-n-Xv circuit in which the X VC-n termination points are dedicated to this client signal. At any point in time, Y out of X VC-n termination points may be set up to carry client traffic. For example when VCAT is deployed on a Router, the VCAT group connects directly to one STM-N interface port (in absence of a HO or LO switch fabric in the router). The transport network will then split the VCAT group into two or more subgroups of components, each being routed via diverse routes.

Variable Client to End point Mapping: For a set of M client signals there are M VC-n-Xv VCAT endpoints sharing a set of N (N>M) VC-n termination points. Typically MxX > N (example: M=10, X=7, N=64); i.e. there is a kind of overbooking. Implication: must be able to accommodate multiple different sized VCAT groups at an "interface". For example an STM-64 interface can support many different VC-4-Xv groups.

2.3 VCAT configuration without LCAS

1. Sink end needs to be informed of how many components are in the VCAT group. It has no other way of knowing if it is currently receiving all components intended to be in the group.

2. Additions or removals of components from the VCAT group are not hitless, that is data loss will occur while the source and sink become synchronized as to the number of members in the group. With each addition or removal the sink end point needs to be told the expected number of components in the group.

3. Failure of a component is detected external to VCAT system. Entire group is rendered inoperable until source takes the failed component out of service and sink end is notified to take component out of service.

2.4 VCAT configuration with LCAS

1. Sink end (and source end) are first configured with the value of "Y" (the number of components), and more specifically which of the X (e.g. VC-n) access points (and thus (VC-n) termination functions) are allocated to the VCAT group with Y (VC-n) components. LCAS then detects automatically which of those Y (VC-n) components is carrying actual traffic and puts them into service for the group.

2. When a new component signal has to be added to a VCAT group the following procedure applies.

    1. Configure the adaptation source/sink functions and change the number of components, Y, to Y+1 by identifying which of the X-Y (e.g. VC-n) access points currently outside the group is added to the group;

    2. The new component is created, e.g., the cross-connections are establish along the components path.

    3. As soon as LCAS protocol information exchange is finished, i.e., the state NORM is reached, client traffic is sent on the added component.

    This procedure does not affect the already established LCAS members, that is, client traffic is not sent on the new component until the LCAS procedure is complete;

3. When a component is removed the following procedure applies:

    1. LCAS protocol is used to remove the component from the group, that is, incoming traffic client data is transmitted on the other VCAT component(s) to assure that the procedure is not traffic affecting

    2. Configure the adaptation source/sink functions and change the number of components Y to Y-1; i.e. remove the VC-n access point from the group.

    3. The component connection can be, if needed, removed from the transport network.

4. When a component fails, the LCAS sink detects the failure (how this is done is outside the scope of this ID) and informs the source of this failure via the member status (MST) information. The source then:

    1. Takes the failed component out of service and if necessary rearranges the sequencing of the VCAT group.

    2. Informs the sink about the component removed from service and any re-arranging of the VCAT group.

    When the failed component is repaired, LCAS can automatically add the repaired component back to the group, or alternatively a new component can be added to bring the group back to its original size. Note that component failure is not hitless, but note the fast notification times of Table 8

2.5 Component Signal Configuration Scenarios

Here we use the term "group" to refer to the entire VCAT group and the terminology "set" and "subset" to refer to the collection of potential VCAT group member signals.

1. A fixed bandwidth VCAT group, transported over a co-routed set of member signals. This is the case where the intended bandwidth of the VCAT group does not change and all member signals follow a similar route. The intent here is the capability to allocate the "right" amount of bandwidth.

2. A fixed bandwidth VCAT group, transported over at least two disjointly routed subsets of member signals. The intent here is additional resilience and graceful degradation in the case of failure. Implications: either LCAS needs to be supported by both

source and sink or we need two-way communications of some type between source and sink to coordinate which members are to be used in the group in a failure scenario. Protection or restoration may be applied in order to restore the original group size in case of failure of either one of the subsets.

3. A dynamic VCAT group (bandwidth can be increased or decreased via the addition or removal of member signals), transported over a co-routed set of members. Intent here is dynamic sizing of bandwidth. Implications: LCAS is needed for hitless resizing. Note before LCAS can do its part of getting traffic over the modified VCAT group, the two VCAT/LCAS endpoints need to be configured (Y -> Y+1 or Y -> Y-1); this requires either "communication" between the two endpoints (when one of the endpoints is configured by call/connection controller, or simple communication of the call/connection controller with both endpoints. Without LCAS we still need two way communications between source and sink to coordinate which members are used in the group and changes will not be hitless. Of course, if all the members of the group are co-routed a single failure may destroy the entire group and cause interruption of traffic even if LCAS is enabled.

4. A dynamic VCAT group, transported over at least two disjointly routed subsets of member signals. Intent here is dynamic resizing and resilience. Implications similar to cases 2 and 3 above.

5. Two or more VCAT groups between the same source and sink who desire to share a pool of component signals between them. Each VCAT group may have a dedicated set of members, and may also obtain additional members from a "common pool" of components. Note that at any given point in time a component signal can belong to at most one VCAT group. The intent here is to allow dynamic resizing of VCAT groups via the sharing of a pool of established component signals without requiring complete circuit provisioning, i.e., only the group membership of the component signal would change. Implications: a communications mechanism between source and sink to indicate during a "change" which group a component should now belong. Similar dynamics and resilience implications as cases 2 and 3 above. (This is Adrian's scenario).

3. Current Support for VCAT group provisioning with GMPLS

Here we see how well we can satisfy the scenarios of Section 2.

3.1 Discovery of VCAT/LCAS

Currently no support for discovery of VCAT or LCAS apriori, i.e., via routing information. Support for "discovery" of VCAT capability at connection establishment time via signaling, i.e., we can request VCAT connection and if the end system cannot support it,it would refuse the connection. TBD -- is there a specific error code concerning "VCAT not supported".

Currently there is no mechanism to ask for an LCAS enabled end point nor is there a way to find out if the other end is LCAS enabled until after the connection is established. This is a problem if we specifically want hitless dynamic resizing or fast graceful degradation for a VCAT group.

3.2 Support for Multiple Client to End Point Mappings

This is where we can have more than one VCAT group on an "interface" (port, etc...) and we need to tell them apart. Currently there is no "VCAT group identifier" in GMPLS.

3.3 Support for VCAT configuration without LCAS

Fixed sized co-routed groups are supported with current GMPLS signaling. For disjointly routed components we would need a small amount of signaling between the VCAT source and sink to make up for the lack of LCAS. In particular, each side (source and sink) needs to know and be in agreement on the components in the group. It is TBD whether GMPLS's existing Admin-Status object can provide sufficient information to achieve this purpose. Note that we cannot achieve hitless resizing this way but we can be fairly prompt and keep the management systems from having to do this. Main items that we need to know are: (a) which component has failed (sink to source), (b) the which components should be in the group (source to sink).

3.4 Support for VCAT configuration with LCAS

Currently both co-routed and disjointly routed connections can be supported. Detailed analysis TBD. For hitless resizing some reasonable default behaviors for controlling LCAS should be followed: (a) After GMPLS has successfully established a potential new component, LCAS should be told (local to source end) to add it to the group, (b) Before GMPLS tears down a component, LCAS should be told (local to source end) to remove it from the group.

3.5 Component Signal Configuration Support

Rough analysis of the list of Section 2.5.

1. Fixed bandwidth, Co-routed: Yes, already in the signaling RFC.

2. Fixed bandwidth, Diversely routed component subsets: TBD if admin-status object will suffice in the non-LCAS case.

3. Dynamic Bandwidth, Co-routed: TBD if admin-status object will suffice in the non-LCAS case.

4. Dynamic Bandwidth, Diversely routed: Similar requirements as above.

5. Adrian's scenario: Currently not supported. Need to be able to signal that we want a potential component to be used in a new VCAT group. Note that the source end would first remove it from its old group. However we need to tell the VCAT group to add it to. The sink end really can't tell this itself. The LCAS group id is just a 1 bit psuedo-random sequence that is used to avoid adding the wrong component to a group.

4. Possible Extensions to GMPLS to support additional VCAT/LCAS scenarios

Here we look at what might be reasonable to add to GMPLS to support the interest scenarios of Section 2 that were not covered under Section 3 .

4.1 Mechanisms for Discovery of VCAT/LCAS

Would like to get both VCAT and LCAS capability of end systems via routing...

Would like to be able to specifically ask for LCAS capability via signaling...

4.2 Mechanism to Support Multiple Client to End Point Mappings

This is a very important capability and it is very similar to one that is being proposed in the end-to-end signaling for recovery I-D. In particular the ASSOCIATION object. Note, however, since there is a rather high probability that at some point we might use VCAT/LCAS with GMPLS based protection we would really need an ASSOCIATION object type specific to VCAT. Association objects are not unique and therefore adding a new type to the Association object would make it a good candidate to support this requirement.

4.3 Support for Component Signal Configuration Scenarios

TBD based on analysis of use of admin-status object. If the admin-status object is sufficient we will detail its use in this application since it is currently an optional object.

5. References

[1] International Telecommunications Union, "Network node interface for the synchronous digital hierarchy (SDH)", ITU-T Recommendation G.707, December 2003.

[2] American National Standards Institute, "Synchronous Optical Network (SONET) - Basic Description including Multiplex Structure, Rates, and Formats", ANSI T1.105-2001, 2001.

[3] "Link capacity adjustment scheme (LCAS) for virtual concatenated signals", ITU-T Recommendation G.7042, February 2004.

[4] "Virtual concatenation of plesiochronous digital hierarchy (PDH) signals", ITU-T Recommendation G.7043, July 2004.

[5] "Interfaces for the Optical Transport Network (OTN)", ITU-T Recommendation G.709, March 2003.

[6] Sklower, K., Lloyd, B., McGregor, G., Carr, D., and T. Coradetti, "The PPP Multilink Protocol (MP)", RFC 1990, August 1996.

[7] "Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications", IEEE Standard 802.3, March 2002.

[8] Bernstein, G., Rajagopalan, B., and D. Saha, "Optical Network Control: Archtecture, Protocols", Addison-Wesley, 2004.

Authors' Addresses

Greg Bernstein
Grotto Networking

Phone: +1 510 573 2237
Email: gregb@grotto-networking.com

Diego Caviglia
Marconi

Email: Diego.Caviglia@marconi.com

Richard Rabbat
Fujitsu

Phone: +1 408 530 4537
Email: richard@us.fujitsu.com

Appendix A. Acknowledgements

The authors would like to thank Maarten Vissers for extensive reviews and contributions to this draft.

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.