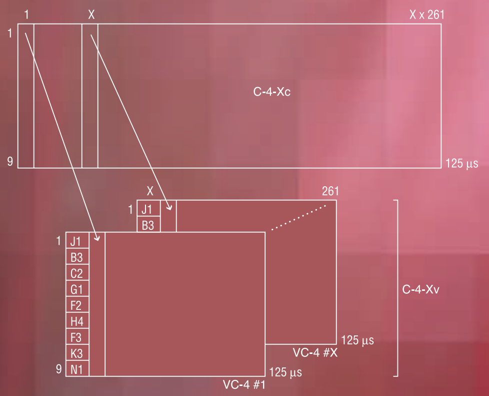
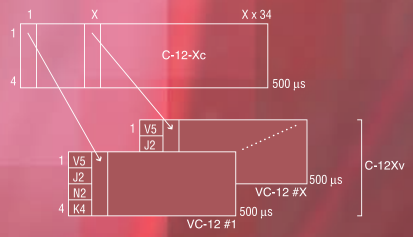
VCAT Virtual Concatenation, G.707

Two methods for concatenation are defined: contiguous and virtual concatenation. Both methods provide concatenated bandwidth of X times Container-N at the path termination. The difference is the transport between the path termination. Contiguous concatenation maintains the contiguous bandwidth through out the whole transport, while virtual concatenation breaks the contiguous bandwidth into individual VCs, transports the individual VCs and recombines these VCs to a contiguous bandwidth at the end point of the transmission. Virtual concatenation requires concatenation functionality only at the path termination equipment, while contiguous concatenation requires concatenation functionality at each network element.

VC-4-Xv structure



VC-12-Xv structure



H4 byte multiframe indicator for HO VCAT

H4 byte 1st multi-frame number 2nd multi-frame number								1st multi-frame no.	2nd multi-frame no.
Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8		
				1st multiframe indicator MFI1 (bits 1-4)					
Sequence indicator MSB (bits 1-4)				1	1	1	0	14	n-1
Sequence indicator LSB (bits 5-8)				1	1	1	1	15	
2nd multifr. indicator MFI2 MSB (bits 1-4)				0	0	0	0	0	n
2nd multifr. indicator MFI2 LSB (bits 5-8)				0	0	0	1	1	
Reserved ("0000")				0	0	1	0	2	
Reserved ("0000")				0	0	1	1	3	
Reserved ("0000")				0	1	0	0	4	
Reserved ("0000")				0	1	0	1	5	
Reserved ("0000")				0	1	1	0	6	
Reserved ("0000")				0	1	1	1	7	
Reserved ("0000")				1	0	0	0	8	
Reserved ("0000")				1	0	0	1	9	
Reserved ("0000")				1	0	1	0	10	
Reserved ("0000")				1	0	1	1	11	
Reserved ("0000")				1	1	0	0	12	
Reserved ("0000")				1	1	0	1	13	
Sequence indicator SQ MSB (bits 1-4)				1	1	1	0	14	
Sequence indicator SQ LSB (bits 5-8)				1	1	1	1	15	
2nd multifr. indicator MFI2 MSB (bits 1-4)				0	0	0	0	0	n+1
2nd multifr. indicator MFI2 LSB (bits 5-8)				0	0	0	1	1	
Reserved ("0000")				0	0	1	0	2	

K4[2] multiframe for LO VCAT

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Frame Count					Sequence Indicator						Reserved																				

Compensation and realignment of the payload

Each VC-3/4 of the VC-3/4-Xv is transported individually through the network. Due to different propagation delay of the VC-3/4s, a differential delay will occur between the individual VC-3/4s. This differential delay has to be compensated and the individual VC-3/4s have to be realigned for access to the contiguous payload area. The realignment process has to cover at least a differential delay of 125 µs.

A two-stage 512 ms multiframe is introduced to cover differential delays of 125 µs and above (up to 256 ms). The first stage uses H4, bits 5-8 for the 4-bit multiframe indicator (MFI1). MFI1 is incremented every basic frame and counts from 0 to 15. For the 8-bit multiframe indicator of the second stage (MFI2), H4, bits 1-4 in frame 0 (MFI2 bits 1-4) and 1 (MFI2 bits 5-8) of the first multiframe are used. MFI2 is incremented once every multiframe of the first stage and counts from 0 to 255. The resulting overall multiframe is 4096 frames (= 512 ms) long.

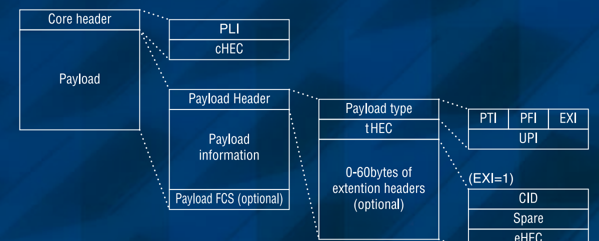
The sequence indicator SQ identifies the sequence/order in which the individual VC-3/4s of the VC -3/4-Xv are combined to form the contiguous container VC-3/4-Xc. Each VC-3/4 of a VC-3/4-Xv has a fixed unique sequence number in the range of 0 to (X–1). The VC-3/4 transporting the first time slot of the C-3/4 of the C-3/4-Xc has the sequence number 0, the VC-3/4 transporting the second C-3/4 of the C-3/4-Xc has the sequence number 1 and so on up to the VC-3/4 transporting time C-3/4 X of the C-3/4-Xc with the sequence number (X–1). For applications requiring fixed bandwidth the sequence number is fixed assigned and not configurable. This allows the constitution of the VC-3/4-Xv to be checked without using the trace. The 8-bit sequence number (which supports values of X up to 256) is transported in bits 1 to 4 of the H4 bytes, using frame 14 (SQ bits 1-4) and 15 (SQ bits 5-8) of the first multiframe stage.

HO VCAT multiframe and sequence indicator

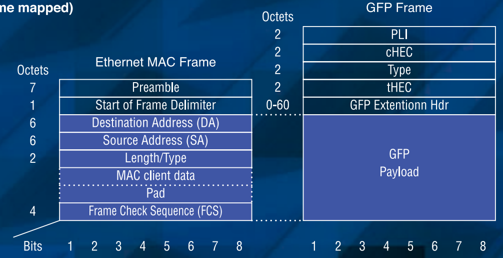
GFP Generic Framing Procedure, G.7041

GFP provides a generic mechanism to adapt traffic from higher-layer client signals over a transport network. Client signals may be PDU-oriented (such as IP/PPP or Ethernet MAC), block-code oriented constant bit rate stream (such as Fibre Channel or ESCON/SBCON).

GFP frame structure



Ethernet and GFP (frame mapped) Frame Relationships



GFP Core Header : The four octets of the GFP Core Header consist of a 16-bit PDU Length Indicator field and a 16-bit Core Header Error Check (cHEC) field. This header allows GFP frame delineation independent of the content of the higher layer PDUs.

PDU Length Indicator (PLI) field : The two-octet PLI field contains a binary number representing the number of octets in the GFP Payload Area. The absolute minimum value of the PLI field in a GFP client frame is 4 octets. PLI values 0-3 are reserved for GFP control frame usage.

Core HEC (cHEC) field : The two-octet Core Header Error Control field contains a CRC-16 error control code that protects the integrity of the contents of the Core Header by enabling both single-bit error correction and multi-bit error detection. The cHEC sequence is calculated over the octets of the Core Header.

GFP Payload Area : The GFP Payload Area, which consists of all octets in the GFP frame after the GFP Core Header, is used to convey higher layer specific protocol information. This variable length area may include from 4 to 65 535 octets. The GFP Payload Area consists of two common components: a Payload Header and a Payload Information field. An optional Payload FCS (pFCS) field is also supported.

Payload Header : The Payload Header is a variable-length area, 4 to 64 octets long, intended to support data link management procedures specific to the higher-layer client signal. The area contains two mandatory fields, the Type and the tHEC fields, and a variable number of additional payload header fields. This group of additional payload header fields are referred to as the Extension Header. The presence of the Extension Header, and its format, and the presence of the optional Payload FCS are specified by the Type field. The tHEC protects the integrity of the Type field. An implementation shall support reception of a GFP frame with a Payload Header of any length in the range 4 to 64 octets.

GFP Type field : The GFP Type field is a mandatory two-octet field of the Payload Header that indicates the content and format of the GFP Payload Information field. The Type field distinguishes between GFP frame types and between different services in a multi-service environment. The Type field consists of a Payload Type Identifier (PTI), a Payload FCS Indicator (PFI), a Extension Header Identifier (EXI) and a User Payload Identifier (UPI).

Payload Type Identifier (PTI) : A 3-bit subfield of the Type field identifying the type of GFP client frame. Two kinds of client frames are currently defined, User Data frames (PTI = 000) and Client Management frames (PTI=100).

Payload FCS Indicator (PFI) : A one bit subfield of the Type field indicating the presence (PFI=1) or absence (PFI=0) of the Payload FCS field.

Extension Header Identifier (EXI) : A 4-bit subfield of the Type field identifying the type of Extension Header GFP. Three kinds of Extension Headers are currently defined, a Null Extension Header, a Linear Extension Header, and a Ring Extension Header.

User Payload Identifier (UPI) : An 8-bit field identifying the type of payload conveyed in the GFP Payload Information field. Interpretation of the UPI field is relative to the type of GFP client frame as indicated by the PTI subfield.

Type HEC (tHEC) field : The two-octet Type Header Error Control field contains a CRC-16 error control code that protects the integrity of the contents of the Type Field by enabling both single-bit error correction and multi-bit error detection.

Extension Header for a linear frame : The Payload Header for a Linear (Point-to-Point) frame with an Extension Header which is intended for scenarios where there are several independent links requiring aggregation onto a single transport path.

Channel ID (CID) field : The CID is an 8-bit binary number used to indicate one of 256 communications channels at a GFP termination point.

Spare field : The 8-bit spare field is reserved for future use.

Extension HEC (eHEC) field : The two-octet Extension Header Error Control field contains a CRC-16 error control code that protects the integrity of the contents of the extension headers by enabling both single-bit error correction (optional) and multi-bit error detection.

Payload Frame Check Sequence (pFCS) field : The GFP Payload FCS is an optional, four-octet long, frame check sequence. It contains a CRC-32 sequence that protects the contents of the GFP Payload Information field. A value of 1 in the PFI field identifies the presence of the payload FCS field.

GFP Payload Type Identifiers

Payload Type Identifiers Type Bits <15:13>	Usage
000	Client Data
100	Client Management
Others	Reserved

GFP Extension Header Identifiers

Extension Header Identifiers Type Bits <11:8>	Usage
0000	Null Extension Header
0001	Linear Frame
0010	Ring Frame
Others	Reserved

User Payload Identifiers for GFP Client Frames

PTI = 000	
User Payload Identifier (binary) TYPE Bits <7:0>	GFP Frame Payload Area
0000 0000 1111 1111	Reserved and not available
0000 0001	Frame-Mapped Ethernet
0000 0010	Frame-Mapped PPP
0000 0011	Transparent Fibre Channel
0000 0100	Transparent FICON
0000 0101	Transparent ESCON
0000 0110	Transparent Gb Ethernet
0000 0111	Reserved for future
0000 1000	Frame-Mapped Multiple Access Protocol over SDH (MAPOS)
0000 1001	Transparent DVB ASI
0000 1010	Framed Mapped IEEE 802.17 Resilient Packet Ring
0000 1011	Frame-Mapped Fibre Channel FC-BBW
0000 1100	Asynchronous Transparent Fibre Channel
0000 1101 through 1110 1111	Reserved for future standardization
1111 0000 through 1111 1110	Reserved for proprietary use

GFP Client Management frame User Payload Identifier

PTI = 000	
UPI value	Usage
0000 0000 and 1111 1111	Reserved
0000 0001	Client Signal Fail (Loss of Client Signal)
0000 0010	Client Signal Fail (Loss of Character Synchronization)
0000 0011 thru 1111 1110	Reserved for future use

LCAS Link Capacity Adjustment Scheme, G.7042

The LCAS is a scheme that should be used to increase or decrease the capacity of a container that is transported in an SDH/OTN network using Virtual Concatenation. In addition, the scheme will automatically decrease the capacity if a member experiences a failure in the network, and increase the capacity when the network fault is repaired. The scheme is applicable to every member of the Virtual Concatenation group.

H4 byte multiframe indicator for HO LCAS

H4 byte				H4 byte				1st multi-frame no.	2nd multi-frame no.
Bit1	Bit 2	Bit3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8		
				1st multifr. indicator MFI1 (bits 1-4)					
Sequence indicator MSBs (bits 1-4)				1	1	1	0	14	n-1
Sequence indicator LSBs (bits 5-8)				1	1	1	1	15	
2nd multifr. indicator MFI2 MSBs (bits 1-4)				0	0	0	0	0	n
2nd multifr. indicator MFI2 LSBs (bits 5-8)				0	0	0	1	1	
CTRL				0	0	1	0	2	
GID ("000x")				0	0	1	1	3	
Reserved ("0000")				0	1	0	0	4	
Reserved ("0000")				0	1	0	1	5	
CRC-8				0	1	1	0	6	
CRC-8				0	1	1	1	7	
Member status MST				1	0	0	0	8	
Member status MST				1	0	0	1	9	
0	0	0	RS_Ack	1	0	1	0	10	
Reserved ("0000")				1	0	1	1	11	
Reserved ("0000")				1	1	0	0	12	
Reserved ("0000")				1	1	0	1	13	
Sequence indicator SQ MSBs (bits 1-4)				1	1	1	0	14	
Sequence indicator SQ LSBs (bits 5-8)				1	1	1	1	15	
2nd multifr. indicator MFI2 MSBs (bits 1-4)				0	0	0	0	0	n+1
2nd multifr. indicator MFI2 LSBs (bits 5-8)				0	0	0	1	1	
CTRL				0	0	1	0	2	
0	0	0	GID	0	0	1	1	3	
Reserved ("0000")				0	1	0	0	4	
Reserved ("0000")				0	1	0	1	5	
C_1	C_2	C_3	C_4	0	1	1	0	6	
C_5	C_6	C_7	C_8	0	1	1	1	7	

K4[2] multiframe for LO LCAS

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Frame Indicator					Sequence Indicator						CTRL				GID	Reserved "0000"				RS_Ack	Member Status								C1 C1 C3 / CRC-3		

H4 Frame to Member number relation

2nd multiframe frame number	Member number				
0, 32, 64, 96, 128, 160, 192, 224	0	1	2	3	MST- multiframe
	4	5	6	7	
1, 33, 65, 97, 129, 161, 193, 225	8	9	10	11	
	12	13	14	15	
⋮	⋮	⋮	⋮	⋮	
30, 62, 94, 126, 158, 190, 222, 254	240	241	242	243	
	244	245	246	247	
31, 63, 95, 127, 159, 191, 223, 255	248	249	250	251	
	252	253	254	255	

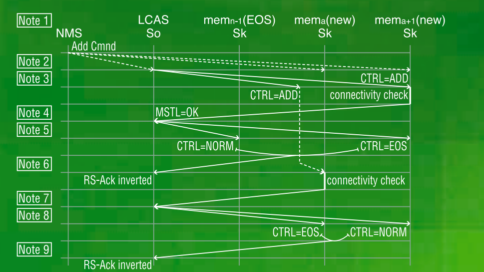
K4 Frame to Member number relation

Frame number	Member number								
0, 8, 16, 24	0	1	2	3	4	5	6	7	MST-multiframe
1, 9, 17, 25	8	9	10	11	12	13	14	15	
2, 10, 18, 26	16	17	18	19	20	21	22	23	
3, 11, 19, 27	24	25	26	27	28	29	30	31	
4, 12, 20, 28	32	33	34	35	36	37	38	39	
5, 13, 21, 29	40	41	42	43	44	45	46	47	
6, 14, 22, 30	48	49	50	51	52	53	54	55	
7, 15, 23, 31	56	57	58	59	60	61	62	NA	

LCAS CTRL words

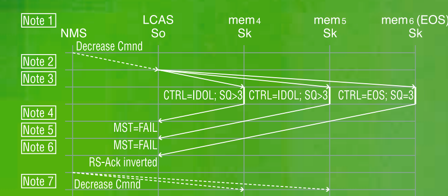
Value	Command	Remarks
0000	FIXED	This is an indication that this end uses fixed bandwidth (non-LCAS mode)
0001	ADD	This member is about to be added to the group
0010	NORM	Normal transmission
0011	EOS	End of Sequence indication and Normal transmission
0101	IDLE	This member is not part of the group or about to be removed
1111	DNU	Do Not Use (the payload) the Sk side reported FAIL status

Add two members after last one in the group of n



Note		Member n			Member a (new)			Member a+1(new)			RS-Ack
		CTRL	SQ	MST	CTRL	SQ	MST	CTRL	SQ	MST	
1	Initial Condition	EOS	n-1	OK	IDLE	FF	FAIL	IDLE	FF	FAIL	0
2	NMS issues Add command to So and Sk LCASC	EOS	n-1	OK	IDLE	FF	FAIL	IDLE	FF	FAIL	0
3	So (a) sends CTRL = ADD and SQ = n; So (a+1) sends CTRL = ADD and SQ =n+1	EOS	n-1	OK	ADD	n	FAIL	ADD	n+1	FAIL	0
4	Sk (a+1) sends MS=OK to So	EOS	n-1	OK	ADD	n	FAIL	ADD	n+1	OK	0
5	So (n-1) sends CTRL = NORM; So (a+1) sends CTRL = EOS and SQ = n	NORM	n-1	OK	ADD	n+1	FAIL	EOS	n	OK	0
6	RS-Ack bit inverted due to change in sequence	NORM	n-1	OK	ADD	n+1	FAIL	EOS	n	OK	1
7	Sk (a) sends MST=OK to So	NORM	n-1	OK	ADD	n+1	OK	EOS	n	OK	1
8	So (a) sends CTRL = EOS; So (a+1) sends CTRL = NORM	NORM	n-1	OK	EOS	n+1	OK	NORM	n	OK	1
9	RS-Ack bit inverted due to change in sequence	NORM	n-1	OK	EOS	n+1	OK	NORM	n	OK	0

Remove members 4 and 5 from a VGC with n = 6 members



Note		Member 4			Member 5			Member 6			RS-Ack
		CTRL	SQ	MST	CTRL	SQ	MST	CTRL	SQ	MST	
1	Initial Condition	NORM	3	OK	NORM	4	OK	EOS	5	OK	0
2	NMS issues Decrease command to So LCASC	NORM	3	OK	NORM	4	OK	EOS	5	OK	0
3	So (3) sends CTRL = IDLE, SQ > 3; So (4) sends CTRL = IDLE, SQ > 3; So (5) sends SQ = 3	IDLE	> 3	OK	IDLE	> 3	OK	EOS	3	OK	0
4	Sk (un-wanted) sends MST = FAIL to So	IDLE	> 3	FAIL	IDLE	> 3	OK	EOS	3	OK	1
5	Sk (un-wanted) sends MST = FAIL to So	IDLE	> 3	FAIL	IDLE	> 3	FAIL	EOS	3	OK	1
6	RS-Ack bit inverted due to change in sequence	IDLE	> 3	FAIL	IDLE	> 3	FAIL	EOS	3	OK	1
7	NMS issues Decrease command to Sk LCASC	IDLE	> 3	FAIL	IDLE	> 3	FAIL	EOS	3	OK	1

MultiFrame Indicator (MFI) field : At the So side the MFI is equal for all members of the VCG and it will be incremented each frame. At the Sk side the MFI shall be used to realign the payload for all the members in the group. The MFI is used to determine the differential delay between members of the same VCG.

Sequence Indicator (SQ) field : Contains the sequence number assigned to a specific member. Each member of the same VCG is assigned a unique sequence number, starting at 0, similar to the Recommendations for Virtual Concatenation in ITU-T Recs. G.707 [1] and G.709 [2]. The SQ is ignored at Sk end for members sending IDLE in the control field. The SQ of a member removed from the VCG shall be set to the highest possible value.

Control (CTRL) field : The control field is used to transfer information from So to Sk. It shall be used to synchronize the Sk with the So and provides the status of the individual member of the group.

Group Identification (GID) bit : Used for identification of the VCG. The GID bit of all members of the same VCG has the same value in the frames with the same MFI. The GID provides the receiver with a means of verifying that all the arriving members originated from one transmitter. The contents are pseudo-random, but the receiver is not required to synchronize with the incoming stream. The pseudo-random pattern used is $2^{15}-1$. *NOTE – The GID is not valid for members sending IDLE in the control field.*

CRC field : To simplify the validation of the changes in the virtual concatenation overhead, a CRC is used to protect each control packet. The CRC check is performed on every control packet after it has been received, and the contents rejected if the test fails. If the control packet passes the CRC test, then its contents are used immediately. To simplify MFI multi-framing it is allowed to disregard the result of the control packet CRC check for the MFI element checked by the CRC so that the multi-framing process may use the MFI element equivalently with the non-LCAS virtual concatenation processing case.

Member Status (MST) field : Information from Sk to So about the status of all members of the same VCG. It reports the member status from Sk to So with two states: OK or FAIL (1 status bit per member). OK = 0, FAIL = 1. Since each control packet contains only a limited number of bits for communicating the MST field, this information is spread across multiple control packets, an MST multiframe. The quantity of members in the VCG can be any number in the allocated range (e.g. 0-255 for High Order in SDH), and can be changed. For each member, the Sk uses the SQ number it receives from the So as the MST number for its response to the So. In this manner, the MST values received by the So will always correspond directly to the SQ values that it assigned. To allow the receiver to determine the number of members in the VCG, the following should be noted. The highest non-failed active member will be indicating end of sequence (EOS) in the control field. The VCG may have other members with a higher SQ value in the do-not-use (DNU) state. At initiation of a VCG sink all members shall report MST = FAIL. A transition to MST=OK occurs when a control packet is received for that member with a control field of ADD (or NORM or EOS after it has been added). All unused MST and members that have a control field of IDLE, shall be set to FAIL.

Re-Sequence Acknowledge (RS-Ack) bit : When a renumbering of the sequence numbers of the members sending in CTRL field NORM, DNU, EOS, or when a change of the number of these members is detected at the Sk, a notification to the So per VCG has to be performed by toggling (i.e. change from '0' to '1' or from '1' to '0') the RS-Ack bit. In particular, the causes that trigger the toggling of RS-Ack bit can be listed as follows:

· SQ change for any VC of the VCG (SQ change detected by Sk for members in DNU/NORM/EOS);

· CTRL="ADD"→CTRL="EOS" and/or CTRL="ADD" → CTRL="NORM" (Addition of one or more members);

· CTRL="NORM" (or "EOS") → CTRL="IDLE" (Decrease bandwidth);

· CTRL="DNU" → CTRL="IDLE".

The Next Generation SONET/SDH Network Technologies

ANRITSU CORPORATION
http://www.anritsu.com