For the case of virtual concatenated signals, Figures 1 and 2 present the port functional models; two versions are identified: (I) and (II).

Version (I) is a single port (Ia) application with pre-allocated VC-n termination functions (Sn_TT) as illustrated by the more specific functional model in (Ib). Note that not all X VC-n termination functions need to be in-service; typically a subset may be operational, which subset may be increased or decreased as required. If LCAS is supported this bandwidth increase/decrease can be performed hitless.

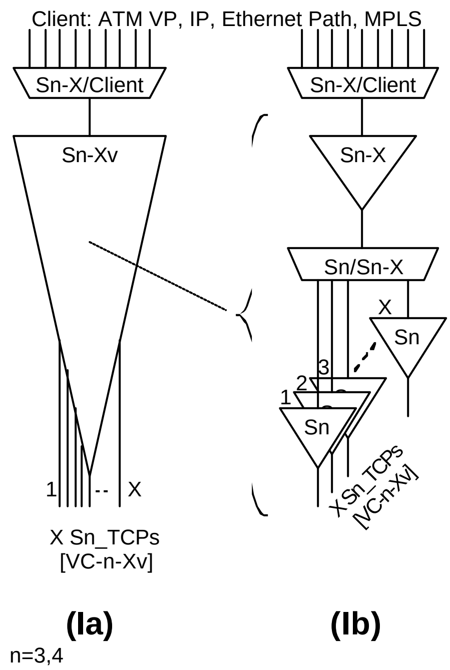

Figure 1 - single VC-n-Xv (n=3,4) port providing client layer topological link

Version (IIa) is a multi port application for which the VC-n termination functions (Sn_TT) are not pre-allocated to a particular port. When the bandwidth of the client link needs to be increased, an additional VC-n trail will be set-up, with as end point one of the unallocated VC-n termination functions. This VC-n termination function is connected to the client link via a matrix connection in connection function Y. Vice versa, when the bandwidth of a client link is to be decreased, the VC-n trail is released and the VC-n termination function can be de-allocated. "Y" is a temporary identifier for this new sublayer that supports this more flexible allocation of ports.

Version (IIb) is a variation of version IIa (or an extension of version (I)) with the flexibility provided by a VC-n connection function (Sn_C) instead of the new sublayer Y connection function. (Y_C).

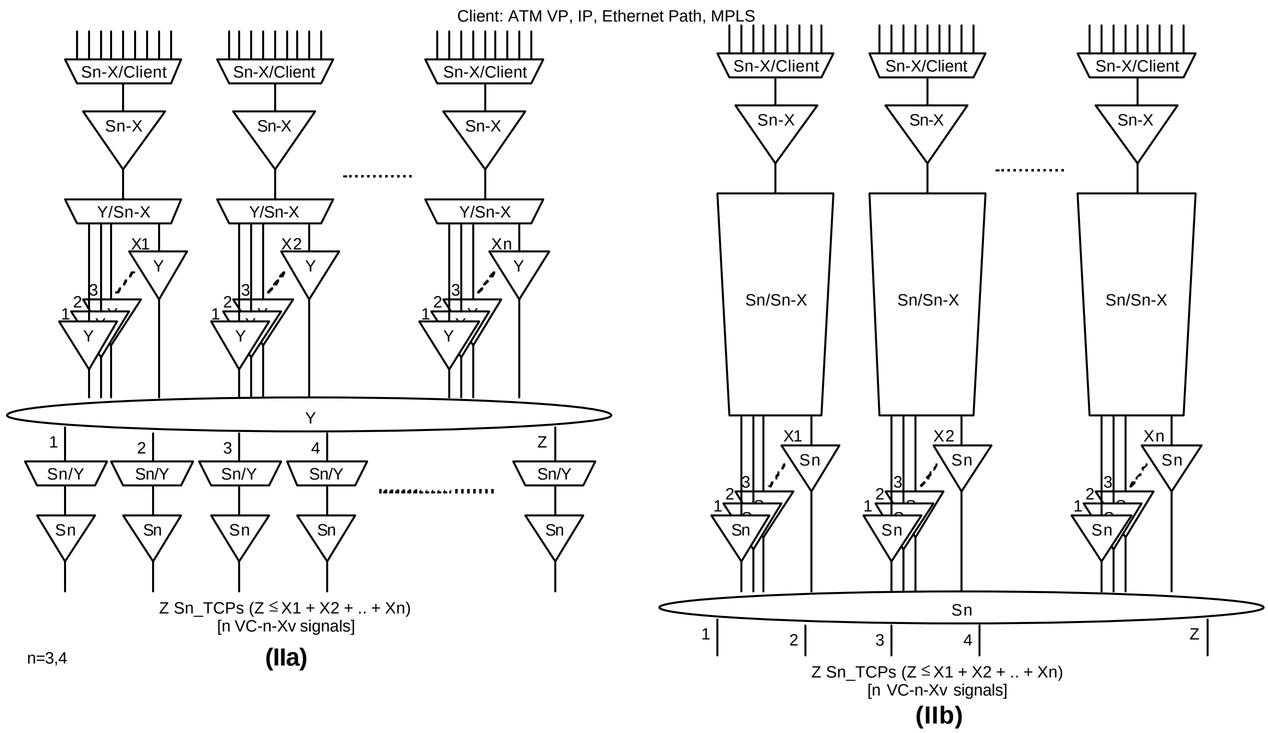

Figure 2 - multiple VC-n-Xv (n=3,4) ports with over-subscription providing client layer topological links

The Sn/Sn-X_A (or Y/Sn-X_A) function has “X” Access Points, each connecting to one Sn_TT (or Y_TT) function. This is a fixed relationship. The Sn/Sn-X_A function has a selector, which is used to connect “C-n-timeslots of the C-n-Xc” to the individual VC-n trails.

Figure 3 illustrates the case in which a C-n-3c is active of which the C-n #0 is transported via the VC-n trail starting at Access Point #2, C-n #1 via AP #X and C-n #2 via AP #1.

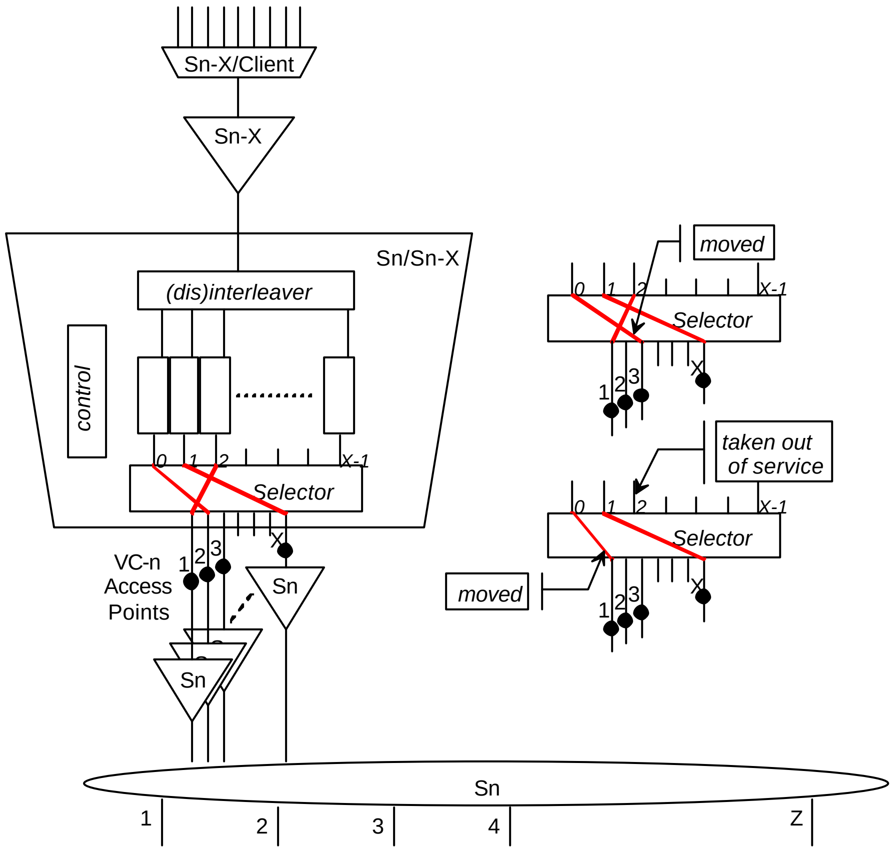

Figure 3

If the VC-n trail via AP #2 would fail, LCAS would stop forwarding information over this VC-n trail (temporarily (non-permanent) bandwidth reduction). If the failed VC-n trail can't be repaired quickly enough, one may

- either reroute that VC-n trail (endpoints stay the same; case IIb in figure 2 and figure 3) by changing matrix connections in VC-n connection functions,
- or if rerouting is not possible then a new VC-n trail can be setup (e.g. via AP #3) and the C-n #0 is to be moved from AP#2 to AP#3 (figure 3, top right),
- or the VC-n trail via AP #2 is taken out of service and released (permanent bandwidth reduction). In this case C-n #2 (via AP #1) is to be taken out of service and C-n #0 is to be routed over VC-n trail with AP #1 (figure 3, middle right).

If the VC group (VCX) is to be increased, C-n #3 will be taken into service. C-n #3 can be connected to any available AP; e.g. AP #3. If the VCX is to be decreased, then the highest number C-n is to be taken out of service.

NOTE – the Y/Sn-X_A function (see figure 2) doesn't need the selector function of the Sn/Sn-X_A function. This selector functionality is available via the Y_C function.

If you have an implementation without selector (or Y_C) (see figure 4), then the VC-n trail and the C-n process block in the Sn/Sn-X_A functions are locked. If you have to take out VC-n trail via AP#2, then you can't use C-n processing block "B" anylonger, and you must renumber the C-n to process block allocation (right in figure 4); i.e. process block A associated with C-n #0, process block B associated with C-n #1 can't be used and C-n #1 is to be associated with process block C, C-n #2 associated with C moves now to D, etc.

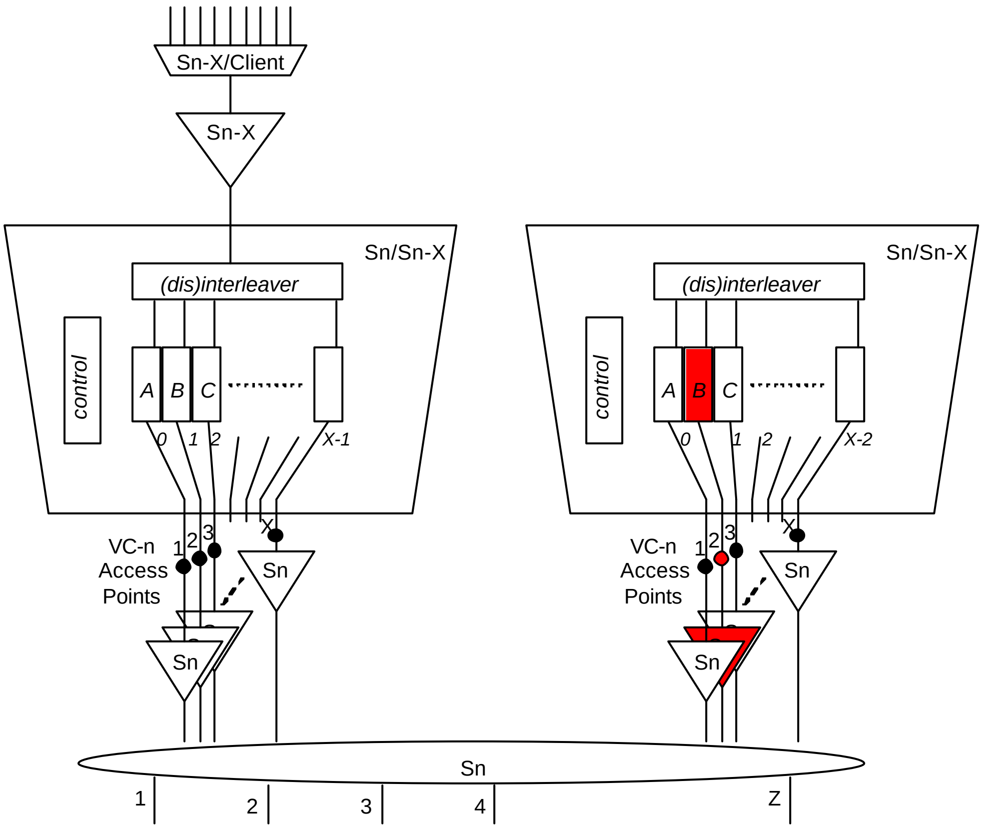
Sn-X/Client
Sn-X
Sn/Sn-X
(dis)interleaver
control
A
B
C
0
1
2
X-1
VC-n
Access
Points
1
2
3
X
Sn
Sn
Sn/Sn-X
(dis)interleaver
control
A
B
C
0
1
2
X-2
VC-n
Access
Points
1
2
3
X
Sn
Sn
Sn
1
2
3
4
Z

Figure 4