

# sACN Project Synchronisation Mechanism

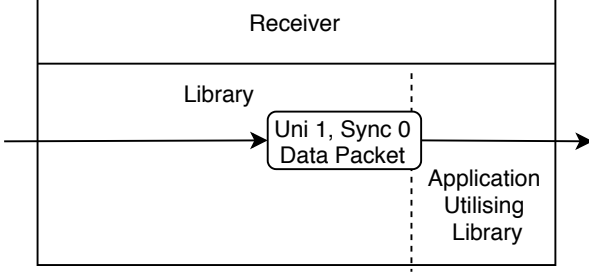
This diagram runs through an example scenario of packets being received to demonstrate how the project sACN receiver mechanism should behave.

**Key**

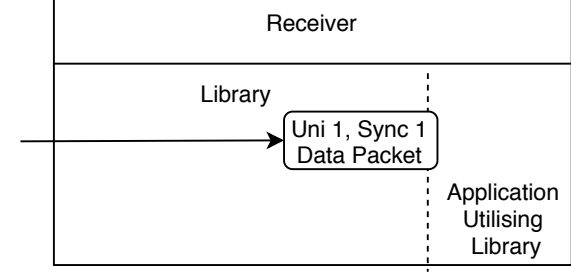
- Receiver**: The entire receiving program, split between the project Library (what is being created) and the Application code which is utilising the library.
- Uni 1, Sync 0 Data Packet**: An sACN packet which has been received from the network. Uni: refers to the data universe and Sync refers to the synchronisation address.
- Vertical Dotted Line**: The dotted line shows the separation between the library and the application code utilising the library. Once a packet has crossed this dotted line it is out of the scope of this project.
- Arrows**: The arrows show the movement of the packet information, so an arrow coming into the Library represents a packet being received from the network and an arrow moving out of the library represents the packet data being passed upto the application. No arrow on a packet shows that the packet has been stored and is waiting.
- Grey Arrows Indicate Explanation**
- Red indicates that the packet has been deleted**
- Orange indicates that a packet has been combined with another packet through a merge.**

The scenario below follows a storyboard style going from left to right and top to bottom. The number in **bold** indicates the current position in the scenario with the numbers increasing sequentially starting at 1.

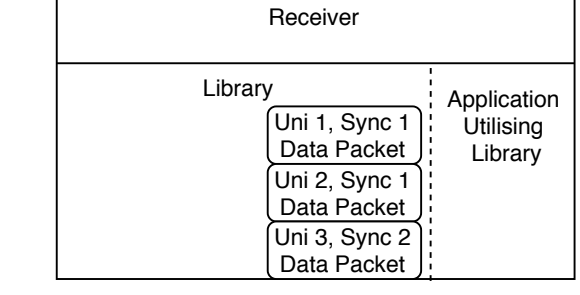
**1.** Data with a zero synchronisation address is unsynchronised and therefore is immediately passed from the library upto the application.



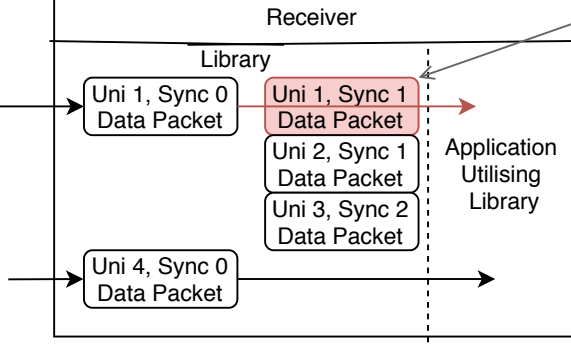
**2.** Data with a synchronisation address > 0 is synchronised and so therefore is held by the library.



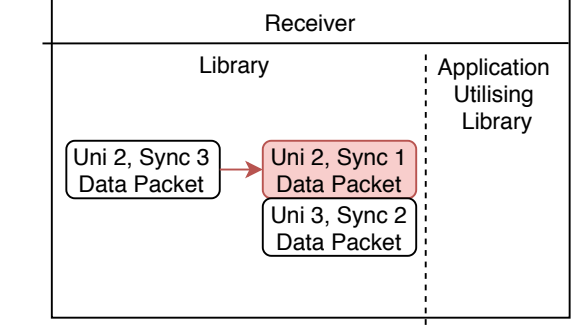
**3.** More data for other universe with sync address > 0 is received and therefore held by the library.



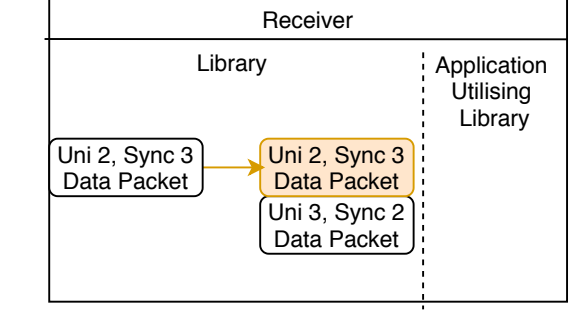
**4.** If data is received that is unsynchronised it is still passed through the Library as normal.



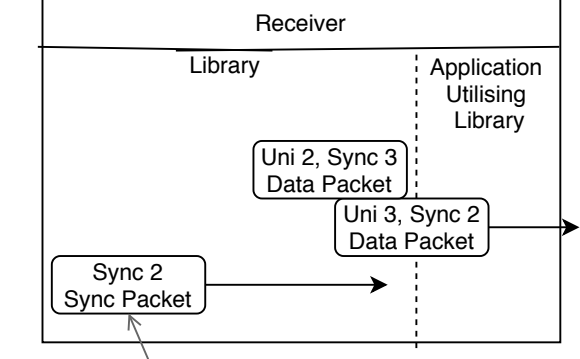
**5.** If data arrives for the same universe but a different synchronisation address then the previous data is discarded



**6.** If data arrives for the same universe and synchronisation address then the previous data is discarded or optionally merged using a user defined function.

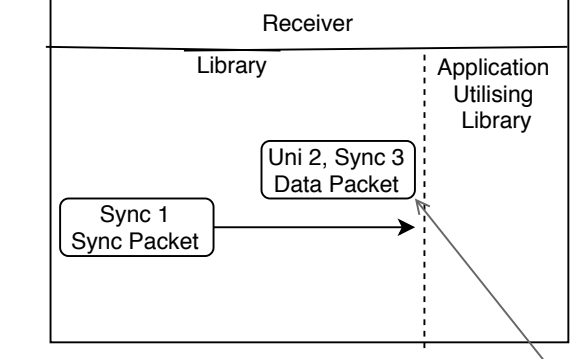


**7.** Synchronisation packets trigger all data packets waiting for that synchronisation address (note the sync packet itself isn't passed on).



Note that any source can trigger synchronisation not just the one that sent the data originally

**8.** Synchronisation packets may have no effect if no data is waiting for the corresponding synchronisation address



Keeps waiting for the corresponding synchronisation packet

**9.** Note that Universe discovery packets have no direct effect on this mechanism and cannot be synchronised

