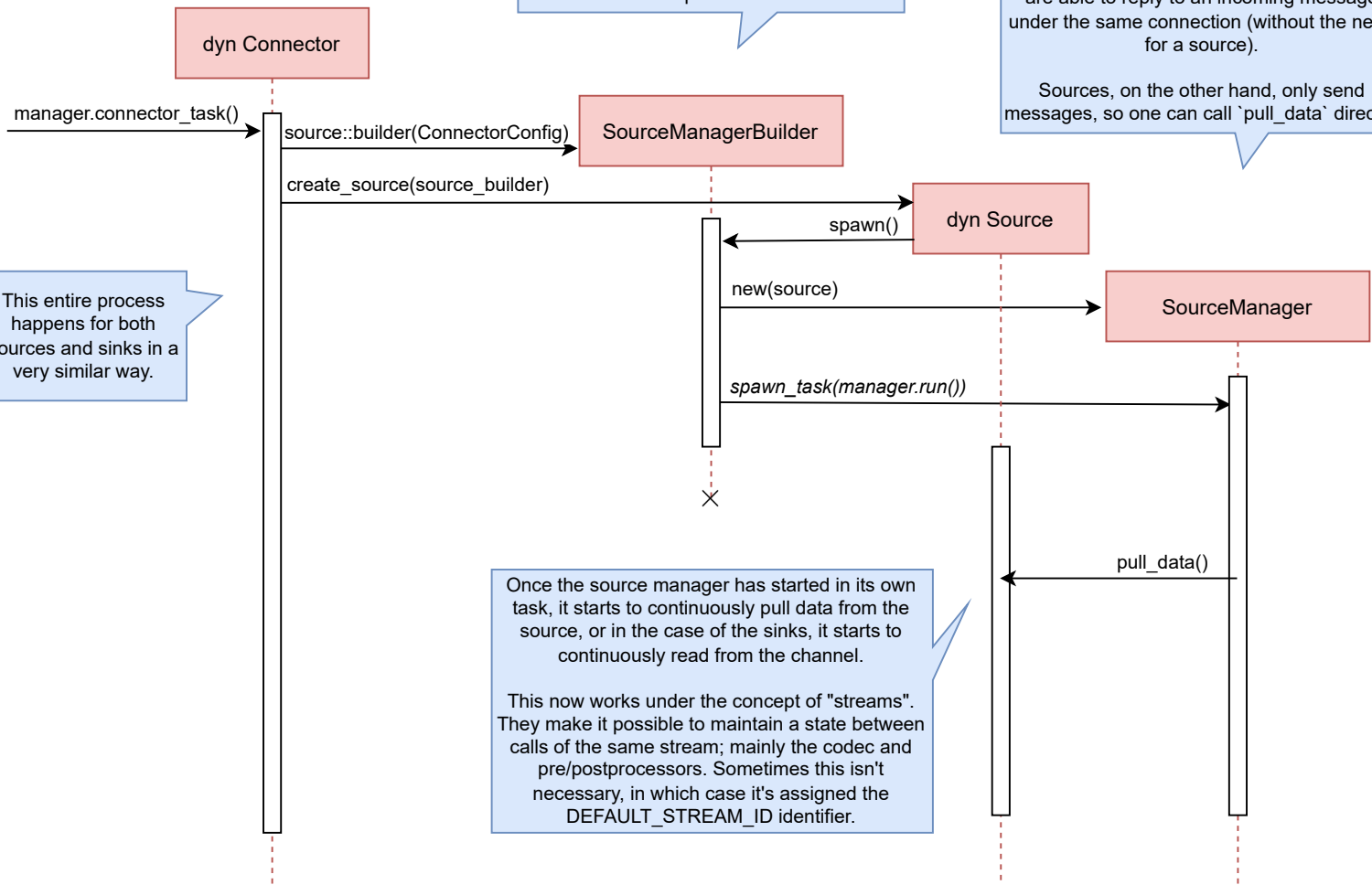


The source or sink builders initialize the source or sink, starting their manager in a new task. Unlike the connectors manager, a source or sink manager will only handle a single one of them, so they may communicate directly without channels.

Note that a connector may implement only a source, a sink or both, so it is possible that one of these will not be created in the first place.

The main difference here between sources and sinks is that the latter's channel combines both incoming and outgoing messages. This is because some sinks like TCP or WebSockets are able to reply to an incoming message under the same connection (without the need for a source).

Sources, on the other hand, only send messages, so one can call 'pull_data' directly.



This entire process happens for both sources and sinks in a very similar way.

Once the source manager has started in its own task, it starts to continuously pull data from the source, or in the case of the sinks, it starts to continuously read from the channel.

This now works under the concept of "streams". They make it possible to maintain a state between calls of the same stream; mainly the codec and pre/postprocessors. Sometimes this isn't necessary, in which case it's assigned the `DEFAULT_STREAM_ID` identifier.