

F.22 Reader Data Lifecycle

[Return to DDS Quality of Service](#)

The `READER_DATA_LIFECYCLE` policy controls the behavior of the `DataReader` with regards to the lifecycle of the data-[instances](#) it manages, that is, the data-[instances](#) that have been received and for which the `DataReader` maintains some internal resources.

The `DataReader` internally maintains the [samples](#) that have not been taken by the [application](#), subject to the constraints imposed by other [QoS](#) policies such as [HISTORY](#) and `RESOURCE_LIMIT`.

The `DataReader` also maintains information regarding the identity, `view_state` and `instance_state` of data-[instances](#) even after all samples have been ‘taken.’ This is needed to properly compute the states when future samples arrive.

Under normal circumstances the `DataReader` can only reclaim all resources for instances for which there are no writers and for which all samples have been ‘taken.’ The last sample the `DataReader` will have taken for that instance will have an `instance_state` of either `NOT_ALIVE_NO_WRITERS` or `NOT_ALIVE_DISPOSED` depending on whether the last writer that had ownership of the instance disposed it or not. The following figure provides a state chart describing the transitions possible for the `instance_state`.

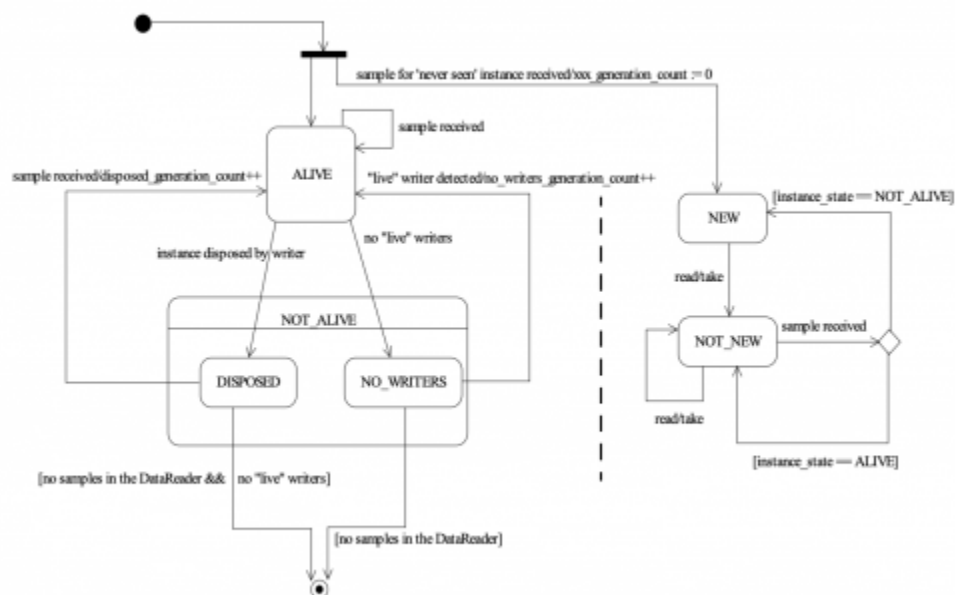


Figure 1: Statechart of the `instance_state` and `view_state` for a single instance.

In the absence of the `READER_DATA_LIFECYCLE` QoS this behavior could cause problems if the application “forgets” to ‘take’ those samples. The ‘untaken’ samples will prevent the `DataReader` from reclaiming the resources and they would remain in the `DataReader` indefinitely.

The `autopurge_nowriter_samples_delay` defines the maximum duration for which the `DataReader` will maintain information regarding an instance once its `instance_state` becomes

NOT_ALIVE_NO_WRITERS. After this time elapses, the DataReader will purge all internal information regarding the instance, any untaken samples will also be lost.

The `autopurge_disposed_samples_delay` defines the maximum duration for which the DataReader will maintain samples for an instance once its `instance_state` becomes NOT_ALIVE_DISPOSED. After this time elapses, the DataReader will purge all samples for the instance.

Source: [DDS 1.4 Spec](#)

From: <https://www.omgwiki.org/dido/> - **DIDO Wiki**

Permanent link: https://www.omgwiki.org/dido/doku.php?id=dido:public:ra:xapend:xapend.f_qos:reader_data_lifecycle&rev=1628274561

Last update: **2021/08/06 14:29**

