

Service call type: AQ_Aggregate_Read

What is it for?

AQ_Aggregate_Read is used for loading a stored [Aggregate](#) into the profile based on a (known) Aggregate Id

This service will read an aggregate based on an aggregate type and Id. This service consults the database of stored aggregates. It will load the entry with the provided Id. If the Id is not present in the database, the `IDError` exit is taken. Otherwise, the aggregate entry is loaded into the profile. In case of versioned aggregates, this service will read the aggregate based on aggregate Id in combination with version Id.

Parameters

Name	Direction	Description	Type	Occurrence	Required
Aggregate Type	Input	Select the aggregate you want to read. This is a list of the aggregates definitions.	Aggregate	Single valued	Yes
AggregateId	Input	The attribute that holds the Id of the aggregate you want to read.	Expression resulting in an integer	Single valued	Yes
VersionIdInput	Bidirectional	The Id of the version to read.	Expression resulting in an integer	Single valued	No
VersionIdOutput	Output	The attribute on which the Id of the version that is to be read will be stored.	Attribute	Single valued	No
TargetInstance	Output	The instance on which the Id of the version that is to be read will be stored in the attribute mentioned in VersionIdOutput.	Expression resulting in an instance	Single valued	No

How to - find the AggregateId

The AggregateId of the desired Aggregate can be found for example using a [Service call type: AQ_Aggregate_Search](#) or by displaying a [Container type: AQ_Aggregate_List](#) containing a [Button](#).

Existing instances

When a singleton instance is present in the profile and an aggregate is read which holds this instance, the instances are merged. The merging rules are as follows:

- When an attribute is filled in the profile and the same attribute is empty in the loaded aggregate, the value of attribute in the profile is kept.
- When an attribute is empty in the profile and the same attribute is filled in the loaded aggregate, the value of read attribute is used.
- When an attribute is filled in both the profile as well as the aggregate, the read aggregate entry overrides the profile entry.

For each instance of a multiton, an instance is created. This means that instances of multitons are *not* merged, which has consequences when reading aggregates with multitons. Most likely, the profile must be cleared before reading such an aggregate.

Static instances have the same merge behavior.

Events

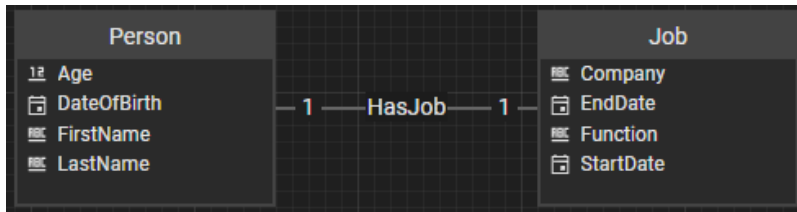
This service uses default events for a commit and for an exception. The `IDError` exit should be mapped to a custom event and will be triggered when the given AggregateId cannot be found.

The `Unauthorized` exit is triggered when the user does not have one of the roles that is set on the [aggregate definition](#).

Example

Below an example is given for a read service with versioning.

Entity Relationship Diagram (ERD)



Service call definition

Service call

Service type: **AQ_Aggregate_Read**

Parameters

→ AggregateType	ⓘ ×	Aggregate Person
→ AggregateId	ⓘ ×	Expression <code>Control.AggregateID</code>
↔ VersionIDInput	ⓘ ×	Expression <code>Control.VersionID</code>
← VersionIDOutput	ⓘ ⊕	
← TargetInstance	ⓘ ⊕	

Aggregate definition

Aggregate

Entity	Included relation ⓘ	To entity
Person	<input checked="" type="checkbox"/> HasJob	Job

+ Add entity

Custom metadata

Name	Data type	Multivalued	Value list	Value
+ Add custom metadata				

Visual overview of aggregate concepts

Aggregate

