

Aggregations

Aggregation is the creation of a unique (*aggregate*) of the object corresponding to each non-**NULL** value of some *aggregated* property. Each such object is assumed to have properties that map this object to each aggregated property parameter, and, conversely, a property that maps the aggregated property parameters to this object.

The aggregated object and each aggregated property parameter must belong to a specified [class](#).

The aggregation mechanism is implemented using two [consequences](#) with automatic resolution and an [aggregation](#) operator. With the help of the aggregation operator, the first consequence creates an object when the aggregated property becomes non-**NULL**, and writes the necessary values to all its properties. The second consequence deletes the object when the aggregated property becomes **NULL**.

Language

To create aggregations, use the [operator](#) **AGGR**.

Examples

```
1 CLASS A; CLASS B; CLASS C;
2 f = DATA INTEGER (A, B);
3 c = AGGR C WHERE f(A a, B b) MATERIALIZED INDEXED;
4
5 CLASS AB;
6 ab = AGGR AB WHERE A a IS A AND B b IS B; // for each A B pair creates an object AB
7
8 CLASS Shipment 'Delivery';
9 date = ABSTRACT DATE (Shipment);
10 CLASS Invoice 'Invoice';
11 createShipment 'Create delivery' = DATA BOOLEAN (Invoice);
12 date 'Shipment date' = DATA DATE (Invoice);
13 CLASS ShipmentInvoice 'Delivery by invoice' : Shipment;
14 shipment(Invoice invoice) = AGGR ShipmentInvoice WHERE createShipment(invoice); // creating a
15 date(ShipmentInvoice si) += sum(date(invoice(si)),1); // delivery date = invoice date + 1
```