WriteOperationOrdering

Relational databases often use Referential Integrity (RI) constraints to maintain database consistency. When RI is enabled, database operations must be performed in a specific order that prevents invalid references. If this order is not adhered to then the database will throw an exception rather than let the write succeed.

When an application makes modifications to a data graph and then asks the RDB DAS to "apply changes", the DAS will create a "batch" of database writes that must be sorted and then executed in the correct order to avoid RI Constraint exceptions. A simple example will help here. Assume a database with two tables CUSTOMER and ORDER. Also assume that the ORDER table has a foreign key reference to the CUSTOMER table. Now assume that the application has made modifications to a data graph that was retrieved by the DAS using a SELECT statement like the following:

```
SELECT * FROM CUSTOMER LEFT JOIN ORDER ON CUSTOMER.ID = ORDER.CUSTOMER_ID WHERE CUSTOMER.NAME = ?
```

The resulting data graph will contain all matching customers as well as their related orders. Now assume the application uses the SDO API's to delete some customer from the graph and then also deletes the related orders. Although the application deleted the customer first, the DAS must delete the orders first from the database since deleting the customer, with existing referencing orders, would violate the RI constraints and result in an exception.