

---

# EODatabaseOperation

<b>Inherits From:</b>	NSObject
<b>Conforms To:</b>	NSObject (NSObject)
<b>Declared In:</b>	EOAccess/EODatabaseOperation.h

## Class Description

An EODatabaseOperation object represents an operation—insert, update, or delete—to perform on an enterprise object and all the necessary information required to perform the operation. You don't ordinarily create instances of EOAdaptorOperation; rather, the Framework automatically creates an EODatabaseOperation object for each new, updated or deleted object in an EOEditingContext. An EODatabaseContext object analyzes a set of database operations and maps each operation to one or more adaptor operations. The adaptor operations are then performed by an EOAdaptorChannel object. You generally interact with EODatabaseOperation objects only if you need to specify the order in which a set of operations are carried out (see the description for the EODatabaseContext delegate method **databaseContext:willOrderAdaptorOperationsFromDatabaseOperations:**).

An EODatabaseOperation specifies an enterprise object (called “object”) on which the operation is performed, the global ID for the object, and the object's entity. In addition, the database operation has a snapshot containing the last known database values for the object and a **newRow** dictionary of new or updated values to save in the database. Finally, a database operation specifies one of the following operators (the type of operation represented by the database operation).

- EODatabaseNothingOperator
- EODatabaseInsertOperator
- EODatabaseUpdateOperator
- EODatabaseDeleteOperator

## Method Types

Creating a new EODatabaseOperation	– initWithGlobalID:object:entity:
Accessing the global ID	– globalID
Accessing the object	– object
Accessing the entity	– entity

---

Setting the operator	<ul style="list-style-type: none"> <li>– <code>setDatabaseOperator:</code></li> <li>– <code>databaseOperator</code></li> </ul>
Setting the database snapshot	<ul style="list-style-type: none"> <li>– <code>setDBSnapshot:</code></li> <li>– <code>dbSnapshot</code></li> </ul>
Setting the row	<ul style="list-style-type: none"> <li>– <code>setNewRow:</code></li> <li>– <code>newRow</code></li> </ul>
Setting the adaptor operations	<ul style="list-style-type: none"> <li>– <code>addAdaptorOperation:</code></li> <li>– <code>removeAdaptorOperation:</code></li> <li>– <code>adaptorOperations</code></li> </ul>
Comparing new row and snapshot values	<ul style="list-style-type: none"> <li>– <code>rowDiffs</code></li> <li>– <code>rowDiffsForAttributes:</code></li> </ul>

## Class Methods

### **adaptorOperations**

– (NSArray \*)**adaptorOperations**

Returns the EOAdaptorOperation objects that need to be performed to carry out the operation represented by the receiver.

**See also:** – `addAdaptorOperation:`, – `removeAdaptorOperation:`

### **addAdaptorOperation:**

– (void)**addAdaptorOperation:**(EOAdaptorOperation \*)*adaptorOperation*

Adds *adaptorOperation* to the receiver's list of adaptor operations.

**See also:** – `adaptorOperations`, – `removeAdaptorOperation:`

### **databaseOperator**

– (EODatabaseOperator)**databaseOperator**

Returns the receiver's database operator.

**See also:** – `setDatabaseOperator:`

---

## dbSnapshot

– (NSDictionary \*)**dbSnapshot**

Returns the database snapshot for the receiver’s enterprise object. The snapshot contains the last known database values for the enterprise object. The dictionary returned from this method will be empty if the receiver’s object has just been inserted into an EOEditingContext and has not yet been saved in persistent storage.

**See also:** – **setDBSnapshot:**

## entity

– (EOEntity \*)**entity**

Returns the entity that corresponds to the receiver’s enterprise object.

**See also:** – **initWithGlobalID:object:entity:**

## globalID

– (EOGlobalID \*)**globalID**

Returns the global ID that corresponds to the receiver’s enterprise object.

**See also:** – **initWithGlobalID:object:entity:**

## initWithGlobalID:object:entity:

– **initWithGlobalID:**(EOGlobalID \*)*globalID* **object:**(id)*object* **entity:**(EOEntity \*)*entity*

The designated initializer, this method initializes a new EODatabaseOperation instance. Sets the enterprise object to which the operation will be applied, the object’s global ID, and the object’s entity. Returns **self**.

**See also:** – **globalID**, – **object**, – **entity**

## newRow

– (NSMutableDictionary \*)**newRow**

Returns a dictionary representation of the receiver’s enterprise object. In addition to all the properties of the enterprise object that are stored in the database, the dictionary contains values for the non-derived attribute’s of the enterprise object’s entity that aren’t visible in the enterprise object. For example, primary and foreign keys aren’t ordinarily properties of an enterprise object but are attributes of the object’s entity.

---

The **newRow** dictionary is initialized with the values in the receiver's snapshot. New or updated values are added to the **newRow** dictionary (replacing out-of-date values) as the Framework maps changes in the object to an operation.

**See also:** – **setNewRow:**

## **object**

– (id)**object**

Returns the receiver's enterprise object.

**See also:** – **initWithGlobalID:object:entity:**

## **removeAdaptorOperation:**

– (void)**removeAdaptorOperation:**(EOAdaptorOperation \*)*adaptorOperation*

Removes *adaptorOperation* from the receiver's list of adaptor operations.

**See also:** – **adaptorOperations**, – **addAdaptorOperation:**

## **rowDiffs**

– (NSDictionary \*)**rowDiffs**

Returns values in the receiver's **newRow** dictionary that are different than the corresponding values its snapshot. The dictionary returned from this method contains the new values from the enterprise object.

**See also:** – **rowDiffsForAttributes:**, – **newRow**

## **rowDiffsForAttributes:**

– (NSDictionary \*)**rowDiffsForAttributes:**(NSArray \*)*attributes*

For the EOAttribute objects in *attributes*, this method returns values in the receiver's **newRow** dictionary that are different than the corresponding values its snapshot. The dictionary returned contains the new values from the enterprise object.

**See also:** – **rowDiffs**, – **newRow**

---

**setDBSnapshot:**

– (void)**setDBSnapshot:**(NSDictionary \*)*dbSnapshot*

Sets the snapshot for the receiver’s enterprise object. If the object has just been inserted into an EOEditingContext, it won’t have a snapshot. In this case, *dbSnapshot* should be an empty dictionary.

**See also:** – **dbSnapshot**

**setDatabaseOperator:**

– (void)**setDatabaseOperator:**(EODatabaseOperator)*dbOp*

Sets the receiver’s database operator.

**See also:** – **databaseOperator**

**setNewRow:**

– (void)**setNewRow:**(NSMutableDictionary \*)*newRow*

Sets the dictionary representation of the receiver’s enterprise object. *newRow* should contain values for all the properties of the enterprise object that are stored in the database and for the non-derived attribute’s of the enterprise object’s entity that aren’t visible in the enterprise object.

**See also:** – **newRow**