
EOKeyGlobalID

Inherits From:	EOGlobalID : NSObject
Conforms To:	NSCoding NSCopying (EOGlobalID) NSObject (NSObject)
Declared In:	EOAccess/EOKeyGlobalID.h

Class Description

EOKeyGlobalID is a concrete subclass of EOGlobalID for persistent IDs based on EOModel information: an entity and the primary key values for the object being identified. When creating an EOKeyGlobalID, the key values must be supplied following alphabetical order for their attribute names. EOKeyGlobalID defines the **globalIDWithEntityName:keys:keyCount:zone:** for creating instances, but it's much more convenient to create instances from fetched rows using EOEntity's **globalIDForRow:** method.

Adopted Protocols

NSCoding	– encodeWithCoder: – initWithCoder:
----------	--

Method Types

Creating instances	+ globalIDWithEntityName:keys:keyCount:zone:
Getting the entity name	– entityName
Getting the key values	– keyValues – keyCount – keyValuesArray
Comparison and identification	– hash – isEqual:

Class Methods

globalIDWithEntityName:keys:keyCount:zone:

+ (id)**globalIDWithEntityName:**(NSString *)*entityName*
 keys:(id *)*keyValues*
 keyCount:(unsigned int)*count*
 zone:(NSZone *)*zone*

Returns an EOKeyGlobalID based on *entityName* and *keyValues*. For performance reasons, the key values are given as a C array of **id**; *count* indicates how many key values there are. The object returned is allocated from *zone*.

EOKeyGlobalIDs are more conveniently created using EOEntity's **globalIDForRow:** method.

Instance Methods

entityName

– (NSString *)**entityName**

Returns the name of the entity governing the object identified by the receiver. This is used by EODatabaseContexts to identify an EOEntity in methods such as **faultForGlobalID:editingContext:**.

hash

@protocol NSObject
– (unsigned int)**hash**

Returns an integer that can be used as a table address in a hash table structure. If two objects are equal (as determined by **isEqual:**), they must have the same hash value. For more information, see the descriptions of this method in the NSObject class and protocol specifications of the Foundation Framework.

isEqual:

@protocol NSObject
– (BOOL)**isEqual:**(id)*anObject*

Returns YES if the receiver and *anObject* share the same entity name and key values, NO if they don't. For more information, see the descriptions of this method in the NSObject class and protocol specifications of the Foundation Framework.

See also: – **entityName**, – **keyValues**

keyCount

– (unsigned int)**keyCount**

Returns the number of key values in the receiver.

See also: – **keyValues**

keyValues

– (id *)**keyValues**

Returns the receiver's key values as a C array of **id** (for performance reasons).

See also: – **keyValuesArray**, – **keyCount**

keyValuesArray

– (NSArray *)**keyValuesArray**

Returns the receiver's key values as an NSArray.

See also: – **keyValues**