EOQualifiedAssociation

Inherits From:	EOAssociation : NSObject
Conforms To:	EOAssociationNotification (EOAssociation)
Declared In:	eointerface/EOQualifiedAssociation.h

Class Description

The EOQualifiedAssociation class implements associations between EOControllers. Often several elements in the user interface are linked through a master-detail relationship. EOQualifiedAssociations establish these links. When an EOController receives a **saveToObjects**, **saveToDataSource**, or **redisplay** message, it not only propagates that action to its next controller, but it also checks each association to see if its destination can perform that action, in which case it also propagates the action to the association's destination. The destination doesn't have to be linked through an EOQualifiedAssociation, though; it merely has to respond to the appropriate messages and have some kind of association that works properly with it.

When an EOQualifiedAssociation receives a **contentsDidChange** or **selectionDidChange** from its controller (the master), it needs to requalify its destination controller (the detail). It does this by getting the selected object from the master controller and using **qualifyWithRelationshipKey:ofObject:** (described in the EOQualifiableDataSources protocol specification) to qualify the data source of its destination controller with the association key and the selected object, and then having the destination controller fetch to update its display. A detail controller can meaningfully display values only when exactly one object is selected in the master; if the master controller doesn't have exactly one object selected, the association passes **nil** as the object to qualify for, which disables the detail controller.

It's possible for an EOQualifiedAssociation to be set up with a detail controller that doesn't yet have a data source. In this case the association will get a detail data source from the master by sending it a **dataSourceQualifiedByKey:** message (described in the EOMasterDataSources protocol specification). It then assigns the new data source to the detail controller by sending it a **setDataSource:** message.

Instance Variables

None declared in this class.

Instance Methods

None declared in this class.