

# Enterprise Objects Examples Quick Reference

*For EOF Release 1.1*

## Overview

This Quick Reference gives a brief description of each example in /NextDeveloper/Examples/EnterpriseObjects to facilitate your searching for a particular topic as well as the level of difficulty for those of you who are new to EOF. The following examples are available:

*Archiving*

*Distributed Enterprise Objects \**

*EOController*

*FlatFileDataSource*

*ManyToMany Relationship*

*Mastering Details \**

*QueryByExample and Sorting <sup>2</sup>*

*Radio Matrix Association*

*SQL Execution*

*Unique Key Generation*

*Validation \**

*Misc*

*SHL Examples (All these examples require Sybase and the Pubs demo database)*

*DB Scripts to install the PEOPLE demo databases*

*\* New in EOF 1.1*

*<sup>2</sup> Significantly enhanced for 1.1*

## **Archiving In A Mixed World (Simple)**

### **Archiving a subclass of NSObject**

ArchivingNSObject (Owner.m)

### **Archiving a subclass of Object**

ArchivingObject (Owner.m)

## **Distributed Enterprise Objects (Advanced)**

**How to use the NSNotification protocol to propagate basic DB Operations like fetch, update, insert between a server and remote clients.**

**The example shows how to send enterprise objects over the wire by following certain conventions. This example supersedes the**

**Distributed Objects example in EOF 1.0 Release.**

## **EOController (Simple)**

**How to build a master/detail controller and master/detail tableview in IB**

See PeopleDemo.nib

**Simple validation and sorting**

See MainController.[hm]

## **FlatFileDataSource (Intermediate)**

**How to build a custom data source that reads a table of data from a flat file to generate EO generic records**

See TableDataSource.[hm]

## **ManyToMany (Intermediate)**

**How to perform Insert/Delete on the master and Add/Delete on the detail in a many-to-many relationship**

See Projects.[hm]

## **MasteringDetails (Advanced)**

**A generic approach to handle insertion and deletion in detail controllers without writing custom code to assign the keys used to join the records in the database.**

## **QueryByExample (Intermediate)**

**How to connect to an EOController to construct qualifiers built from the UI and execute them**

See QBE.[hm]

**How to perform either descending or ascending ordering on any entity and its associated attributes tied to any tableview**

See [SortOrderSetter.\[htm\]](#)

## **RadioMatrixAssociation (Simple)**

**How to write a custom association**

See [RadioAssociation.\[htm\]](#)

## **SQLExecutor (Intermediate)**

**How to evaluate an arbitrary SQL statement for Sybase or Oracle.  
Also executes stored procedures (Result sets can only be  
processed for Sybase adaptor).**

See [Evaluator.\[htm\]](#) and [SybaseDelegate.\[htm\]](#)

## **UniqueKey (Intermediate)**

**How to generate unique keys using custom code and a custom**

## **table .**

See UniqueKey.[hm] and UniqueKey.eomodel.

UniqueKey uses a separate table to hold the external entity name and current max reserved integer key. The UniqueKey objects share a database channel that is sure to be free to allow immediate reservation of a block of keys.

## **Validation (Intermediate)**

**A generic scheme to allow validation logic to be specified in EOs.**

See KeyValueValidation.[hm] and ValidatingDelegate.[hm] in the Validation subproject

## **SystemHouse (SHL) Examples**

### **Access Layer (Simple)**

**How to establish connection and fetch from the database using either the database level API or the adaptor level API**

See DatabaseLevel.[hm] and AccessLevel.[hm]

## **Buffering (Simple)**

**How the different buffering schemes of EOController work**

## **Data Source (Simple)**

**How to build a custom data source whose records are stored in memory**

See DataSource.[hm]

## **Delegation (Simple)**

**Tracing the various delegation methods from EOController, EODatabase, and EOAdaptor**

See EOFDelegateControllerCategory.[hm], EOFDelegateDatabaseCategory.[hm], and EOFDelegateAdaptorCategory.[hm]

## **Enterprise Object(Simple)**

**How to build custom enterprise objects**

## **KeyValueCoding (Simple)**

## **Illustrate the KeyValueCoding protocol**

### **Querying (Intermediate)**

**How to build a query and display its results dynamically in a tableview**

See QueryController.[hm]

### **Sorting (Intermediate)**

**How to apply ordering based on a list of attributes from an entity or columns in the tableview**

See SortController.[hm]

*Last updated: February 1995*

*Mai Nguyen*