

## Release Notes: Enterprise Objects Framework 2.0

This file contains the preliminary release notes for release 2.0 of Enterprise Objects Framework. For the latest release notes for any NeXT product, see NeXTanswer #2455. If you have access to the World Wide Web, you can look up the URL <http://www.next.com/NeXTanswers/HTMLFiles/2455.html#2455.html>.

The MacOS version of this release is 3-way fat; it can be used to develop software for NeXT, Intel, and SPARC. The OpenStep for Windows version of this release can be used to develop software for Intel machines running Windows NT.

### Notes Specific to Windows NT Developers

References to filenames in this document use UNIX format. On Windows NT, you can interpret these by reading backslashes for the slashes and adding the installation directory (**c:\NeXT\** by default) to the beginning. For example, **/NextLibrary/Documentation** on Windows NT is **c:\NeXT\NextLibrary\Documentation**.

In order to use Enterprise Objects Framework on Windows NT, you must purchase appropriate client libraries from your database vendor.

Here's what you need:

#### **Oracle**

Phone: (800) 542-1170

Ask for SQLNET V2.2 for PC/Windows NT

The Oracle adaptor on NT requires the Oracle 7.2 Client Library. It won't work with the 7.1 libraries.

#### **Sybase**

Phone: (800) 685-8225

Ask for: Open Client Runtime for PC/Windows NT

#### **Informix**

Phone: (800) (800)331-1763

Ask for: ESQL/C for Win32

## Converting Release 1.x Applications to 2.0 (Mach only)

Enterprise Objects Framework 2.0 for Mach includes **tops** scripts for performing source code conversion of your Enterprise Objects Framework 1.x applications to 2.0. Because of the magnitude of changes in the architecture of the 2.0 release, these scripts are by no means sufficient for automatically converting your application. However, once you've decided how the structure of your application will evolve to take advantage of the new architecture, these scripts can be useful in identifying and sometimes making specific API conversions. The scripts, along with instructions for running them, can be found in **/NextDeveloper/EOF1xTo2Conversion**.

For more information on the differences between Enterprise Objects Framework 1.x and Enterprise Objects Framework 2.0, see the document *Differences Between Enterprise Objects Framework 1x and 2.0*. You can find it on-line in **/NextLibrary/Documentation/EnterpriseObjects/1x\_To\_2**.

## Installing the Examples

This release provides on-line examples to help familiarize you with Enterprise Objects Framework 2.0. These examples are located in **/NextDeveloper/Examples/EnterpriseObjects**. Installing the examples involves these steps:

- Setting up users and databases on your database server for the example databases.
- Installing the example directory.
- Populating your database server with example data.

### Setting up Database Accounts

The Enterprise Objects Framework 2.0 examples use two sets of tables: Movies and Rentals. Some examples use just one of these databases, while others use both. The multi-database support in Enterprise Objects Framework 2.0 makes it possible for you to install these databases in three different configurations:

- Both sets of tables together in a single user/database.
- Each set of tables in its own user/database on the same database server.
- Each set of tables on its own database server (for example, Movies on Informix, Rentals on Oracle).

Depending on your desired setup, you use the tools available with your database server to set up one or two new user/databases. For example, on Sybase you might create a new database on your server called "Movies" and login with the user "sa". On Oracle you might create a new user with the name "Movies". Once you have set up these accounts,

you're ready to install the examples.

### Copying the Example Directory

To configure and build the examples you need to copy the example directory to a writable area in your file system. You can do this by copying the **/NextDeveloper/Examples/EnterpriseObjects** folder into your home directory (or any other directory writable by you).

### Configuring the Example Models

The model files used by the examples must be configured to use your adaptor and server. To configure the examples, run the **configure\_examples** program in your copy of the examples directory. It will ask you for the name of the adaptor you wish to use (Informix, Oracle, Sybase, and so on) and for the login information for your database. It will then convert the example models for your server.

### Populating the Databases

Now that the examples are configured, you can fill your example databases with sample data. The **install\_database** tool in the **DatabaseSetUp** directory will connect to your databases, add the example tables, and fill them with data. If you later wish to remove the data, simply run the **drop\_database** tool.

### Building the Example Programs

With your example projects installed and your database filled with data, you are ready to build and run the examples. To do this, in a command shell **cd** to your example directory and type **make all**. This performs a **make install** on **BusinessLogic.framework** and **EOExtensions.framework** to put them in **/LocalDeveloper/Frameworks**, where they are shared by many of the other examples. It then makes all of the examples applications.

**Note:** The **/LocalDeveloper/Frameworks** directory must be created and writable by you in order to build the examples. On Mach this can be accomplished by using **su** to become the superuser and then executing the following commands:

```
mkdirs /LocalDeveloper/Frameworks
chmod a+w /LocalDeveloper/Frameworks
```

## New Features for 2.0

- A new class, **EOEditingContext**, that manages an object graph and coordinates change notification. This class is also responsible for change tracking and undo. One **EOEditingContext** can also be nested inside another, thereby supporting nested in-memory transactions.
- Support for multiple databases.
- **NSDecimalNumber**, a class that provides a lossless representation of decimal numbers.

- User interface formatting and validation.
- Updatable relationships: Your business logic may now update relationship pointers in the object graph and the framework will automatically generate the appropriate foreign keys changes to affect the change in the database.
- Referential integrity rules. You can explicitly add referential integrity rules in EOModeler. They will be enforced by Enterprise Objects Framework in your running application.
- Improved support for modeling inheritance.
- Enterprise object-based validation.
- Update support for flattened attributes.
- Schema creation. You can use the Generate SQL command in EOModeler to generate SQL for a selected entity or entities.
- Automatic primary key generation: The Framework can now automatically assign primary keys for newly inserted objects.
- Structured qualifiers—that is, qualifiers are now independent of database and SQL.
- Incremental model loading for improved performance. Models now consist of an **.eomodel** directory in which each entity has its own file. Entity descriptions are loaded into your application only as needed.
- Support for Informix servers.
- An ODBC adaptor (Windows NT only).
- Improved support for Oracle stored procedures.
- Reverse engineering of primary keys and relationships on all supported databases (Informix, Oracle, and Sybase). Enterprise Objects Framework can now read this information from the database and incorporate it into a model.
- The Sybase adaptor is now based on CT-Lib.
- The ability to edit the UserInfo dictionary in EOModeler.

For more discussion of these features, see the document *Differences Between Enterprise Objects Framework 1x and 2.0*.

## New Features Shared with Enterprise Objects Framework 1.2

- An Informix adaptor is included in this release.
- Adaptors are now frameworks. Just as with the old bundle-based adaptors, framework adaptors can be dynamically or statically linked. If you need to reference a class defined in a particular adaptor from your application code (for example, the `InformixChannel` class), you have to add the adaptor to the Frameworks list for your project.

If you have custom adaptors that you're converting to 2.0, you need to convert them into

frameworks. If you do this, note that you may encounter problems linking to the client libraries that weren't present when you were loading your adaptor as a bundle. For example, you can't use common symbols. If you use undefined symbols, you need to set a flag in the linker to tell it to ignore those symbols.

- EOAdaptor's use of **adaptor.info** has been removed. Instead, the **Info-platform.plist** file from the adaptor's bundle is consulted for the adaptor class name, login panel nib name, and adaptor major version number. The key names are as follows:

EOAdaptorMajorVersion

EOAdaptorClassName

To add key-value pairs to the infoDictionary, add an ASCII property list file named **CustomInfo.plist** to the supporting files category of your project. The project makefiles will automatically merge this into the **Info.plist** that is derived from your project.

- The following Enterprise Objects Framework 1.1 classes have been renamed and moved into the AppKit: NXTableView (renamed to NSTableView), and NXImageView (renamed to NSImageView).
- EOApplication is obsolete. NSApplication now manages the autorelease pool.

## New Methods and Behavior

For a description of how Enterprise Objects Framework 2.0 differs from the preceding releases, see the document *Differences Between Enterprise Objects Framework 1x and 2.0*. See **/NextLibrary/Documentation/EnterpriseObjects/1x\_To\_2**.

## Locating the Documentation for this Release

The documentation for this release is provided in RTF (readable on Mach only) and in PDF format to be readable (and printable) on all platforms. On Windows NT, documentation is available in the "OpenStep Books Online" item of the OPENSTEP Enterprise program group. You can also find the documentation in the following on-line locations:

- The *Differences Between Enterprise Objects Framework 1x and 2.0* document describes how the product has changed between the 1x and 2.0 releases. See **/NextLibrary/Documentation/EnterpriseObjects/1x\_To\_2**.
- The *Enterprise Objects Framework Developer's Guide* provides a conceptual overview of the product. In addition to the hard copy book, a pre-production version of the book is provided on-line in PDF format. See **/NextLibrary/Documentation/NextDev/EnterpriseObjects/Guide**.
- The *Enterprise Objects Framework Reference* includes class specifications for the Enterprise Objects Framework classes. Reference documentation is located in **/NextLibrary/Frameworks** in one of these locations:

**/EOAccess.framework/Resources/English.lproj/Documentation/Reference**  
**/EOInterface.framework/Resources/English.lproj/Documentation/Reference**  
**/EOControl.framework/Resources/English.lproj/Documentation/Reference**

#### **If You Need a PDF Reader...**

If you need a PDF Reader, you can download one from one of the following web locations:

- For Windows or Solaris: **<http://www.adobe.com/acrobat/>**
- For NEXTSTEP: **<http://www.omnigroup.com/Software/OmniPDF/>**

#### **Other Documentation**

There are other documents that may provide useful information to you, depending upon your application:

- The *OpenStep Conversion Guide* discusses how to use the automated conversion process to convert applications to OpenStep. See  
**/NextLibrary/Documentation/NextDev/Conversion/ConversionGuide.**

#### **Documentation Feedback**

Your comments on our documentation are especially valuable. Please send electronic mail with your comments and suggestions to **[techpubs\\_feedback@next.com](mailto:techpubs_feedback@next.com)**. Please include **EODoc:** as the first part of your message's subject.

## **Known Problems in Release 2.0**

### **Enterprise Objects Framework**

These problems exist in this release of Enterprise Objects Framework:

Reference: 46679

Problem: All private instance variables should be **@private** or begin with **\_**.

Description: All instance variables in Enterprise Objects Framework should be considered private.

Workaround: Don't access Enterprise Objects Framework instance variables in client programs.

Reference: 46967

Problem: Using delete propagation triggers in database can cause Enterprise Objects Framework to become out of sync.

Description: The use of delete propagation triggers in the database instead of the model may cause Enterprise Objects Framework to become out of sync. This becomes a bigger problem if the Framework is executing several operations as part of a

transaction and the transaction is rolled back by a trigger rule.

Workaround: See chapter 6, “Answers to Common Design Questions,” in the *Enterprise Objects Framework Developer’s Guide* for information on handling this problem.

Reference: 47202

Problem: Conversion from money to double doesn’t always work.

Description: Database number types are sometimes more precise than a C **float** or **double**. Consequently when these numbers are read in from the database, stored in a double, and then written back out to the database, rounding errors may appear. For instance, the value ‘2078927178.0000’ is converted as ‘2078927178.0000002384185791’.

Workaround: Use the `NSDecimalNumber` class for money.

Reference: 47832

Problem: Don’t mark as “used for locking” those attributes in the model with a custom data type unless the custom value class implements a value-based **isEqual:** method.

Workaround: Implement **isEqual:** on the custom data type and don’t mark it used for locking.

Reference: 51250

Problem: You can’t join on derived attributes.

Description: Invalid SQL is generated whenever an derived attribute is designated as a join attribute in a relationship.

Workaround: None.

Reference: 52557

Problem: Enterprise Objects Framework wrongly generates **t0.\*nil\***.

Description: Enterprise Objects Framework sometimes inappropriately generates SELECT statements that include text like “t0.\*nil\*”. For example, this can happen when you make a column with no column name, a column with a column name like “convert(varchar(12), 'Default')”, or derived attributes with no column name.

Workaround: None.

Reference: 53598

Problem: Enterprise Objects Framework doesn’t support to-one relationships joining on candidate (partial primary) keys in the destination.

Description: Enterprise Objects Framework doesn’t support to-one relationships that join on

only part of a compound primary key of the destination entity. This is because the Framework bases its uniquing of fetched objects on their full primary keys. If Enterprise Objects Framework fetches an object with a to-one relationship, it must look that object up in the uniquing table to see if it is already in memory. If it isn't, Enterprise Objects Framework must create a fault for it. Because objects are uniqued by their full primary keys, a candidate key is insufficient to do the lookup. Even doing lookups by both full primary key and by candidate key doesn't work, because a fault created for one relationship joined on the full primary key can't be correlated with a fault for another relationship joined to the same row based on the candidate key.

Workaround: Use a to-many relationship and define accessor methods for the enterprise object that emulate the to-one relationship by getting and setting the first object of the relationship array.

Reference: 61475

Problem: Fetches to the top of an inheritance tree execute too many selects.

Description: Fetches to the top of an inheritance tree perform too many fetches (currently always one fetch is performed for each concrete subclass). When fetching a hierarchy mapped to in a single table, only one fetch should be performed.

Workaround: None.

Reference: 64084

Problem: Inserted enterprise objects don't get removed from EODisplayGroup after sending **revert** to EOEditingContext..

Description: If you fetch objects into an EODisplayGroup and then insert a few objects and delete a few objects, and update a few objects, telling the EOEditingContext to revert or refetch will back out the updates, but not insertions or deletions.

Workaround: The actual insertions and deletions have been reverted, but the EODisplayGroup doesn't know how to revert its object list (since it doesn't keep track of what its original object list was before the insertion or deletions). The easiest workaround is to programmatically tell all affected EODisplayGroups to refetch after telling the EOEditingContext to revert or refetch.

Reference: 64283

Problem: Enterprise Objects Framework 2.0 examples merge with obsolete Enterprise Objects Framework 1.1 examples.

Description: If a system has Enterprise Objects Framework 1.1 Developer installed on it (EODDeveloper.pkg) and then upgrades to 4.0, the EODDeveloper package (incorrectly) stays installed, and the examples stay in

### **/NextDeveloper/Examples/EnterpriseObjects.**

When the user later installs EO2Developer, the Enterprise Objects Framework 2.0 examples are merged with the Enterprise Objects Framework 1.1 example files already in **/NextDeveloper/Examples/EnterpriseObjects**.

Workaround: Uninstall 1.1 EODeveloper.pkg before installing EO2Developer.pkg.

Reference: 64968

Problem: Seemingly innocuous qualified fetch always causes exception.

Description: Some qualified fetches raise exceptions because values in the SQL have been formatted as strings when they should have been formatted as some other type.

This can happen whenever an external type has been entered erroneously that does not actually map to a column type recognized by the database. It may also occur when a Sybase model has no record of user defined types that have been used in the model.

Workaround: In the case of an erroneous external type definition, simply correct it. In the case of a user defined type, create a new model by reverse engineering the database, and copy the connection dictionary with the user defined type information from the new model to the original one.

Reference: 65165

Problem: **install\_database** doesn't work: Sybase: Can't find type 'decimal'.

Description: The **install\_database** script in the **EnterpriseObjects/DatabaseSetUp** directory doesn't work for Sybase 4.9 servers.

When the models in the examples directory were converted to Sybase, the server version wasn't taken into account. Some of the attributes were converted to "decimal" data type and the Sybase 4.x server doesn't know about these.

Workaround: Use EOModeler to find and change all the external types from decimal to float in each of the models in your examples directory.

Reference: 68146

Problem: Fault failure leaves EOInterfaceLayer unstable.

Description: If an entity being displayed has a to-one relationship and the row with the corresponding foreign key cannot be found in the database, when you try to access the source object your user interface gets into a corrupted state.

Workaround: Enterprise Objects Framework raises an exception when a to-one relationship cannot be resolved (because of a referential integrity problem in the database). If this exception is raised by user interface code (for example, in displaying a

master-detail relationship) the state of the user interface control objects can get corrupted. To prevent this you must prevent the exception. One way is to trap the exception by implementing the `EODatabaseContext` delegate method **`databaseContext:failedToFetchObject:globalID:`**. The other is to use a to-many relationship in your model instead of a to-one.

Reference: 68231

Problem: **make all** in the Enterprise Objects Framework examples fails sometimes.

Description: The Enterprise Objects Framework examples will fail to make if **`$(NEXT_ROOT)/LocalDeveloper`** is present on your system and you don't have write access to it. The makefile tries to install the BusinessLogic framework in the **`/LocalDeveloper/Frameworks`** directory.

Workaround: Execute **make** as root or ask your system administrator to create a **`/LocalDeveloper/Frameworks`** directory and make it writable by you.

Reference: 68604

Problem: Repeated false alert that Enterprise Objects Framework 1x nib file is being converted to Enterprise Objects Framework 2.0.

Description: When you reopen a nib file after conversion you get the message: "This Enterprise Objects Framework 1x nib file has been translated to Enterprise Objects Framework 2.0. If you did not intend to convert to Enterprise Objects Framework 2.0, unload EOPalette, quit Interface Builder without saving this file, and reopen the nib."

Workaround: Quit Interface Builder, then reopen the nib file.

Reference: 69211

Problem: Link errors on Windows NT.

Description: Programs on Windows NT must add explicit references to at least one class in each framework in order to avoid link errors at run time.

Workaround: Add a function like that in the following code snippet, which refers to classes in each of Enterprise Objects Framework's layers. Though never invoked, it forces the appropriate linking to occur.

```
#i fdef WIN32
#i mport <EOControl/EOControl.h>
#i mport <EOAccess/EOAccess.h>
#i mport <EOInterface/EOInterface.h>

void _referenceAllEOFrameworks()
{
```

```

        [EODisplayGroup new];          // EOInterface
        [EOEntity new];                // EOAccess
        [EOEditingContext new];        // EOControl
    }
#end if

```

If you create your project with the type "EOF Application," this code is automatically added to your project main file.

Reference: 71554

Problem: EOPickTextAssociation doesn't buffer EOQualifier [...] wildcards.

Description: EOQualifier supports the use of [...] to match a single character from a set. EOPickTextAssociation evaluates its text as a qualifier after each character typed, however, so if the user types an opening bracket, an attention panel appears for each character typed until the closing bracket.

Workaround: None.

## Oracle Adaptor

These problems exist in the Oracle adaptor supplied with this release of Enterprise Objects Framework:

Reference: 62425

Problem: Oracle Adaptor doesn't read stored procedures inside of packages.

Description: There is no way to get the database to tell you the components (procedures and functions) that are inside a package definition. Clients can still create stored procedures in the model that will call into packages, it's just that model description using EOModeler won't create these at connect time.

Workaround: You can use EOModeler to create the stored procedure definitions in the model. Just set the external name of the stored procedure to *package-name.procedure-name*.

Reference: 63348

Problem: Converted models have no width information—Oracle fetches empty strings.

Description: The Enterprise Objects Framework 2.0 Oracle Adaptor has problems if you attempt to use an eomodel file from the Enterprise Objects Framework 1.0 release. These model files don't contain width information for string values, which results in all the strings being fetched with 0 length. If this happens for the column that is the primary key, you will also have problems with uniqueness since every row will appear to have the same key.

Workaround: Use EOModeler to add width information for VARCHAR2 and RAW columns.

Reference: 69421

Problem: If you're editing numeric fields and attempt to save data, you may receive the error:

```
Raise [NSStringWithGap doesn't respond to stringValue]
```

Description: This can happen when an enterprise object inadvertently gets NSString values for attributes that are expected to have NSNumbers or NSDecimalNumber values.

Workaround: If you encounter this problem, you can turn off the use of bind variables by setting the EOUseBindVariables default to no:

```
defaults write NSGlobalDomain EOAdaptorUseBindVariables NO
```

A better solution is to use a number formatter on all the display widgets that map to number values. In addition to all the benefits of the formatting, this will ensure that your enterprise objects' values will always be NSNumbers or NSDecimalNumbers.

## Sybase Adaptor

These problems exist in the Sybase adaptor supplied with this release of Enterprise Objects Framework:

Reference: 61481, 66164

Problem: When debugging, the new Sybase CT-Lib adaptor blocks if you try to step over a call that requires a trip to the Sybase server.

Description: GDB hangs when you use the **next** command to step over a line of code which eventually causes a call to the Sybase client library.

Workaround: Set a breakpoint after the line and use **continue** rather than **next**.

Reference: 61999

Problem: An update fails for text that disallows NULL and has an empty field.

Description: Whenever a text or image column is being updated, the Sybase adaptor attempts to ensure that the column has been initialized correctly by assigning NULL to it. In cases where null is not allowed this does not work.

Workaround: Modify the table definition to allow nulls.

Reference: 62634

Problem: The Sybase adaptor is using **CS\_CONVERT()** to convert numeric data into NSDecimalNumber.

Description: Fetching decimal numbers in locales that use a characters other than ‘.’ for the decimal may not work.

Workaround: None.

## Informix Adaptor

These problems exist in the Informix adaptor supplied with this release of Enterprise Objects Framework:

Reference: 63225

Problem: Informix adaptor does nothing for  
**returnValuesForLastStoredProcedureInvocation.**

Description: The Informix Adaptor will not return variable values for stored procedures.

Workaround: Don’t use stored procedures. If you need to access their logic, use an intermediate table to hold return values and then SELECT them back to your client.

Reference: 64031

Problem: UserDefaults different between Enterprise Objects Framework1.2 and Enterprise Objects Framework2.0 InformixAdaptors.

Description: In Enterprise Objects Framework 1.2, the InformixAdaptor stores defaults in the following domain EOFInformixAdaptor and uses the following keys: INFORMIXDIR, DBDATE, DBLANG, DBMONEY, InformixLogErrors, ShowSystables, ShowTableOwner, Beautify, DefaultIsolationLevel, GlobalOptimization, GlobalExplain, GlobalLockMode, GlobalPDQPriority, GlobalDataSkip, GlobalConstraints, DatabaseExclusive

Enterprise Objects Framework 2.0 stores defaults in the standard NSGlobalDomain and prefixed all keys with “Informix”, that is: InformixINFORMIXDIR, InformixDBDATE, InformixDBLANG, InformixDBMONEY, InformixLogErrors, InformixShowSystables, InformixShowTableOwner, InformixBeautify, InformixDefaultIsolationLevel, InformixGlobalOptimization, InformixGlobalExplain, InformixGlobalLockMode, InformixGlobalPDQPriority, InformixGlobalDataSkip, InformixGlobalConstraints, InformixDatabaseExclusive.

Workaround: None.

## EOModeler

These problems exist in EOModeler with this release of Enterprise Objects Framework:

Reference: 67804

Problem: EOModeler can't find model from another framework.

Description: Problem 1: When you load a nib file into your application, it can't find the model you have in your framework.

Problem 2: When you save changes to a model file in your framework, the change isn't reflected in open nib files.

Enterprise Objects Framework 2.0 finds models for an application by looking for all models in the application and in any frameworks used by the application. By default Enterprise Objects Framework looks for referenced framework in their *installation* location. Thus if you don't have the framework that contains the model installed, Enterprise Objects Framework won't find the model. Similarly, a change saved to a model in the framework project isn't noticed by Interface Builder since it's looking at the *installed* version, not the source version.

Workaround: You can tell Enterprise Objects Framework to look at the source for your framework projects by using the following defaults command (executed in a shell):

```
defaults write NSGlobalDomain EOProjectSourceSearchPath  
"($(HOME)/myProjectsDirectory1, /myOtherProjectsDirectory)"
```

Then, when Interface Builder or EOModeler is looking for models contained in one of your frameworks, it will first search all project directories within **\$(HOME)/myProjectsDirectory1** and **/myOtherProjectsDirectory** before searching for the built versions.

Reference: 57842

Problem: EODisplayGroups are missing their keys.

Description: Enterprise Objects Framework was probably unable to find the model file containing the entity used by the EODisplayGroup. To verify this, inspect the data source of the EODisplayGroup (click on the EODisplayGroup, bring up the Inspector panel, and selected the DataSource panel from the Inspector popup). If the "Model" field is blank, then Enterprise Objects Framework was unable to find your model.

Workaround: Quit Interface Builder and verify that the model containing the entity referenced in the EODisplayGroup is added as a resource in the project containing the nib file.

Reference: 59472

Problem: Derived attributes are limited and don't offer full SQL as advertised

Description: Placing a string or a numeric constant in the definition field of a derived attribute will cause invalid SQL to be generated. Definitions such as "title" and "0.0" will

not work correctly. However, definitions such as “att1 + 5” should work correctly when “att1” specifies another attribute.

Workaround: None.

Reference: 62444

Problem: When you change a text field in an EOModeler Inspector and don't press Return, you can lose your changes.

Workaround: Always use the Return key.

Reference: 62447

Problem: Last column in EOModeler table mode isn't handled correctly when you resize.

Description: The last column in a table view can get clipped. If you widen the second-to-last column by dragging its right edge, the last column becomes narrower and the scroll knob doesn't appear until the mouse approaches the edge of the table view. If you save a model with a clipped column, then when you open it again the column remains clipped even if the browser is big enough to handle it.

Workaround: Widen the last column by dragging its rightmost edge before widening the second-to-last column.

Reference: 62669

Problem: In EOModeler, the external types field isn't case insensitive for word-completion

Description: In the External Types field of the Attribute table view, word-completion needs to be case-insensitive. For example, typing 'v' should complete to 'VARCHAR' just like typing 'V' does.

Workaround: None.

Reference: 62727

Problem: Editing in Inspector with editing still active in table doesn't work.

Description: If you edit something in table mode, then, without tabbing out of the edited column, make a change in the Inspector, the change made in the table column is lost.

Workaround: None.

Reference: 63118

Problem: Schema generation may produce identifiers that are too long.

Description: If you have database table names that are long, the names generated for

constraints may be too long.

Workaround: Keep your table names shorter, or edit the SQL EOModeler generates to keep the constraint names shorter.

Reference: 63119

Problem: EOModeler generates bogus foreign key constraints for compound keys.

Description: Schema generation in EOModeler can produce some confused constraint definitions for Oracle and Informix if you have non-normalized data (such as the TALENT and TALENT\_PHOTO tables).

Workaround: You can work around this by editing the SQL generation scripts, or just not asking EOModeler to generate the constraint information.

Reference: 63169

Problem: Models created from version 4.9 Sybase servers do not include stored procedures.

Workaround: You need to create the the stored procedures by hand.

Reference: 63183

Problem: Trying to generate schema SQL will raise if any of the primary key attributes don't have a column name.

Workaround: None.

Reference: 64339

Problem: Derived attributes don't work properly.

Description: Sometimes when you're trying to create a derived attribute, you may encounter seemingly random errors—the pop-up list in the Inspector reverting to Column, the text you typed in the field being erased or reverted, error panel telling you “Unable to modify the attribute,” and so on.

Workaround: Always add a definition immediately after changing the attribute to from Column to Derived. Once you add a definition for a derived attribute, it can't be deleted—you can change it, but a definition must be present.

Reference: 69386

Problem: Trying to drag a model or an entity from EOModeler into a nib file sometimes fails.

Description: If you try to drag a model or an entity from EOModeler into a nib file, Interface Builder asks if you would like to add this model to the project. Sometimes the

attempt to add the model to the project will fail.

Workaround: Relaunch Interface Builder by double-clicking on the nib icon in Project Builder. Try dragging the EOModel or EOEntity now—everything should work.

## On-line Examples

Reference: 69199

Problem: Error running **configure\_examples** script on Windows NT.

Description: Running **configure\_examples** will fail unless your path is properly specified.

Workaround: In order to install and run examples on NT, you need to add the following to your system path (in the System Control Panel):

```
. ; c:\NeXT\LocalDeveloper\Executables
```

(The above assumes that \$NEXT\_ROOT is defined as **c:\NeXT**).

Reference: 69419

Problem: Example data for Movie and Rental models must be installed in different databases

Description: When installing data for more than one model in a database, the **eo\_sequence\_table** is only properly initialized for the first model installed.

Workaround: You can either install the data for each model in a different database or change the last line in the **install\_database** script located in the DatabaseSetUp directory of your examples directory as follows:

### Change this:

```
${EOUTIL} dump ../BusinessLogic/Rentals.eomodeld -source plist  
database -schemaCreate -postInstall < RentalData.plist
```

### To this:

```
${EOUTIL} dump ../BusinessLogic/Rentals.eomodeld -source plist  
database -schemaCreate -postInstall -force < RentalData.plist
```

When you use the **-force** option as a workaround, exceptions are logged because the **eo\_sequence\_table** table and **eo\_pk\_for\_table** procedure already exist when you execute **install\_databases**. You can just ignore these exceptions.