3.3 Release Notes: The Database Kit

This file contains notes for the 3.3 release of the Database Kit.   It also contains the release notes for the previous Release 3 Database Kit software (Release 3.1 and 3.0).

There are no changes for Release 3.3. However, the functionality of the Database Kit has been superceded by the Enterprise Object Framework. The Enterprise Object Framework product (available separately from NEXTSTEP) provides a way to develop object-oriented database applications. It provides tools for defining an object model and mapping it to a data model, which allows you to create objects that encapsulate both data and methods for operating on that data-and these objects can persist in a relational database accessible via the Framework's data services. You should use the Enterprise Objects Framework to develop database applications instead of the Database Kit.

New Features in Release 3.2

Sybase Connection Management

In previous releases, two connections contexts were established and maintained when an application connected to Sybase: One connection was used to select (and fetch) data, the other was used to update data. The select context was doggedly faithfulÐit stuck around for as long as an application was connected to the server. The other context was created just before each update and killed when the update was completed. During the time that data was being updated, therefore, an application held two contexts. For sites that were allowed a limited number of connections, this double connection was wasteful.

In Release 3.2, two new system default parameters let you control the constancy of Sybase connections. Both take boolean values and are owned by **SybaseAdaptor**:

- **SybaseHoldsSelectConnection** determines whether the select context is maintained when a select isn't in progress. The default is YES.

- **SybaseHoldsUpdateConnection** does the same for the update context. The default is NO.

The default values for these parameters emulate the connection behavior of previous releases. For a minimally context-consumptive application, you should set the parameters thus:

```
> dwrite SybaseAdaptor SybaseHoldsSelectConnection NO
> dwrite SybaseAdaptor SybaseHoldsUpdateConnection NO
```

Conversely, setting both parameters to YES ensures that the two contexts will always exist (while your application is connected to the server).

The Oracle Adaptor and ORACLE7

The Oracle adaptor supports the new ORACLE7 software (it also works with Oracle Version 6).   But note the following:

- To use ORACLE7, you (or your database administrator) must run the **CATALOG6.SQL** command file.

- The adaptor doesn't support ORACLE7 stored procedures (an upgrade of the adaptor that takes advantage of stored procedures is anticipated).

Bugs Fixed in Release 3.2

Interface Layer Bugs

Reference 36045

Problem Can't update a single row in a DBTableView.

Description When you identify a single row through the **rowsChangedFrom:to:** method, some number of surrounding rows are also updated.

Reference 35804

Problem Chained DBFetchGroups aren't always saved.

Description The **saveChanges:** DBModule methods stops saving when it finds a DBFetchGroup that doesn't have any modifications. For example, let's say you have three DBFetchGroups: A (root), B, and C. You modify the data that's "owned" by A and C, but you don't touch B. If you send **saveChanges:** to the DBModule that owns the chain, A's is saved, but C isn'tÐthe save stops when the DBModule determines that B doesn't have any changes to save.

Access Layer Bugs

Reference 35796

Problem Primary key update technique is faulty.

Description The 3.1 Release Notes recommended a primary-key-update technique in which the existing record is deleted and a new record added. Because of the order in which deletions and insertions are processed, this doesn't work.

Reference 34804

Problem Compound keys don't work correctly in a relationship.

Description If you traverse a compound key relationship, the results are often wrong (no records are retrieved from the destination table).

Adaptor Bugs

Reference 35637

Problem NULL values are written to the server as 0.

Description DBValue's **setNull** method effectively sets the object's value (and type) to NULL, but when the object is decoded by an adaptor, the value is incorrectly interpreted as zero.

Reference 36560

Problem Sybase Adaptor hangs seeing 256 characters.

Description A client application using the Sybase Adaptor will hang after text data containing more than 256 characters is fetched or saved.

Reference 35650

Problem Sybase Adaptor can hang during BEGIN TRANSACTION after an error.

Description The Sybase Adaptor doesn't follow the progress of transactions closely enoughĐafter an error, some server tables may remain locked. If you attempt a new transaction, the adaptor will hang.

Reference 35979

Problem Sybase Adaptor can crash if you use a ScrollView object.

Description Setup a DBTableView and a ScrollView such that the ScrollView displays RTF data (a "note") for the record that's selected in the DBTableView. If you edit a record's note, save the modification, select a different record, and then return to the first record, the Sybase Adaptor will cause your application to crash.

Reference 35953

Problem DBKit's Sybase **interfaces** file clobbers the existing **interfaces** file.

Description The Sybase interfaces file that's distributed as part of the Database Kit is simply a templateÐit's useless as released. When installing a release, an existing **interfaces** file is clobbered by this template file. *(The released file's name has been changed to **interfaces.template**)*.

Reference 35689

Problem Aborting a record limited fetch and Sybase Adaptor's alert panel.

Description If you hit the record limit during a fetch, Sybase puts up an alert panel that lets you abort or continue. If you abort, then on all subsequent fetches, an alert panel appears that complains that the result of a previous operation is pending.

Reference 30125

Problem Oracle Adaptor can't handle dates past 1999.

Description Dates that refer to days in or after the year 2000 aren't interpreted correcly.

Reference 30076

Problem Sybase Adaptor always maintains two connections to the server.

Description See the New Features description, above.

Known Problems

Listed below are known problems that were identified in the 3.0 and 3.1 releases. Problems that have been fixed since 3.1 are so noted.

Sybase client library incompatibility *(fixed in 3.2)*

Reference: 34043

The Sybase client library header files distributed in Release 3.1 were version 4.0; the Sybase library (**dblib**) in that release was version 4.6. The two releases weren't totally compatible. Release 3.2 fixes this incompatibility by including *both* versions (4.0 and 4.6) of the header files, and *both* versions of **dblib**. By default, the SybaseAdaptor uses the 4.6 version of the header files and **dblib**. To change this, you must redirect the appropriate links in **/usr/sybase**.

Date restriction in the Oracle adaptor *(fixed in 3.2)*

Reference: 30125

You can't modify primary key data

Reference: 28310

You can't modify a record's primary key; attempting to save after doing so brings up the ªError occurred during Transactionº Alert Panel and the save operation fails. To work around this, you must insert a new record with the modified primary key value, delete the old record, and then save.

In 3.1, the insert-and-delete technique didn't work. Now it does.

DBModeler's list of Oracle database entities is limited

When it asks for a list of entities from the Oracle adaptor, DBModeler only gets those that are owned by the user that has logged into the Oracle server (more accurately, the Oracle adaptor only provides DBModeler with entities created from the tables and views that are owned by the user). To extend the list of entities so you can see other users' entities, you must **dwrite** the OracleTableOwners system default as as set of Oracle user names.

For example, consider the following default written by a user who's known to UNIX as **scott** and to Oracle as **SCOTT**:

```
> dwrite OracleAdaptor OracleTableOwners "'SCOTT','JOE','TONY'"
```

The next time **scott** runs DBModeler, he'll be able to see the entities that are owned by the Oracle users **SCOTT**, **JOE**, and **TONY**. The important points here are

- The names in the list must be quoted as shown.
- You must include your own Oracle user name if you want to see your own entities.

Note that this *only* applies to DBModeler's entity-retrieval mechanism. Your model file can refer to entities that you don't own (as long as you can read them) even though DBModeler doesn't display them in its list of entities.

DBRecordList doesn't do pessimistic locking

DBRecordList supports multi-user concurrency through a ªcheck-before-updateº strategy, also known as *optimistic locking*. Although this is a good strategy for a wide array of uses, there are situations that are best served by *pessimistic locking,* in which a record is locked before the user is allowed to modify it. DBRecordList doesn't currently support pessimistic locking.

Image size limit with the Sybase adaptor

Reference: 30071

The Sybase adaptor can't store images that are larger than 64k bytes. A possible workaround is to load the images into the server directly, through a C program that makes calls to the Sybase client library.

Some stored procedures crash the Sybase adaptor

The Sybase adaptor crashes when asked to execute a stored procedure (through the **evaluateString:** method) that includes two (or more) identical property names, or that involves derived data that doesn't correspond to an actual column. You can work around the first situation by using aliases for the property names; the second situation can be avoided only through abstention.

DBDatabase's connectUsingString: method doesn't work properly

Reference: 33079

Use **connectUsingAdaptor:andString:** instead.

Model filenames mustn't have embedded periods

Reference: 32428

The name of a model file mustn't include a period (the character '.') other than that which distinguishes the ªdbmodelº or ªdbmodelaº extension. For example, the model filename **FloorModel.parquet.dbmodel** is illegal.

AddressBook example phone number truncation

Reference: 34039

The phone number text field in the AddressBook example application is too short. To fix it, you can resize the text field in the application's **AddressBook.nib** file.

Slider associations can crash your application

If you associate a Slider object with an attributeÐwhether through Interface Builder or programmaticallyÐand then delete all the records that contain that attribute, the mere fact of the Slider association will cause your application (or Interface Builder running in Test mode) to crash.

Sybase client library incompatibility

Reference: 34043

The 4.6 release of the Sybase client library that's part of the NEXTSTEP 3.1 release isn't completely compatible with Sybase 4.0 (which was part of NEXTSTEP 3.0).

Oracle requires a connection before setting properties

Reference: 34070

If you're using the Oracle adaptor, you must connect to the server *before* you set the properties of a DBRecordStream or DBRecordList. Otherwise, the "rowID" primary key attribute won't be found by the automatic-key-finding mechanism.

NULL values not handled correctly

Reference: 34033

If you set a NULL value into a DBValue object, set the object into a DBRecordList or DBRecordStream and then save, the NULL value is written to the server as 0 or "". Similarly, a NULL retrieved from the server is interpreted as 0 or "".

Oracle time and GMT

Reference: 33710

Using a binder to evaluate a string that retrieves a timestamp returns GMT time, whereas the same statement executed from the Oracle SQLDBA utility returns local time.

DBTableView doesn't always get endEditing message

Reference: 33022

The **endEditing** message isn't sent to a DBTableView that uses an arbitrary data source (an object of your own DBTableDataSources-conforming class).

Long record qualifications for Oracle can crash your application

Reference: 32750

Applying a DBQualifier that uses a fairly long string (something over 1000 charactersÐthe exact threshhold hasn't been determined) or range list (around 256 elements) will crash an application based on the Oracle adaptor.

DBTableView passes inconsistent values to tableView:movedColumnFrom:to:

Reference: 32626

If you move a DBTableView column anywhere *except* to the last position in the table, the object's delegate is (correctly) sent the **tableView:movedRowFrom:to:** method, with 0-based indices indentifying the positions of the "from" and "to" columns. If you move the column so it's the new "last" (rightmost) column, the argument to the **to:** keyword is too large by one. For example, if the DBTableView contains 3 columns, moving a column to each of the column positions, left-to-right, would send delegate messages that would identify the "to" column (by index) as 0, 1, and (incorrectly) 3.

Bad MultiBinder Example

Reference: 38067

The **MultiBinder** example in **/NextDeveloper/Examples/DatabaseKit** is broken and should be ignored.