initEntity:
setName:

Reporting current context name
module
entity
recordList
currentRecord
recordCount

Controlling current selection setAutoSelect:
doesAutoSelect
setCurrentRecord:
clearCurrentRecord
selectedRowAfter:
redisplayEverything

Manipulating contents deleteCurrentSelection
insertNewRecordAt:
fetchContentsOf:usingQualifier:

Dealing with changes hasUnsavedChanges
validateCurrentRecord
saveChanges
discardChanges

Using associations addExpression:
makeAssociationFrom:to:
takeValueFromAssociation:
addAssociation:
removeAssociation:

Using a delegate delegate

addExpression:newExpression

Adds the DBProperties-conforming object anExpression to the list of expressions that the DBFetch
the database. Typically, you add instances of DBExpression, although it's also possible to add pro
gotten from a database model.

When you add an expression object to a DBFetchGroup, the DBFetchGroup ªownsº the object and
freeing it. If you don't want to surrender control of your expression objects, you should create a co
the copies in invocations of this method). If you're adding a model property, then you must copy i
DBExpression to cover it).

Returns self.

clearCurrentRecord

Deselects the currently selected record (or records), so that there is no selected record. DBAssocia
been involved in the formerly selected records are notified of the change. Returns self. However,
permission to change the rows of the DBFetchGroup's DBRecordList, the method has no effect an

(unsigned int)currentRecord

Returns the position (index number) of the current record in the DBFetchGroup's DBRecordList.

delegate

Returns the DBFetchGroup's delegate.

setDelegate

deleteCurrentSelection

Deletes the currently selected row (or rows) from the DBFetchGroup's DBRecordList. Following
selected. All DBAssociations are notified of the change.

Returns self. However, if no rows were selected, or there is no permission to change the rows of th
DBRecordList, the method has no effect and returns nil.

discardChanges

Terminates any editing changes currently in progress for this DBFetchGroup and recursively for ar
DBFetchGroups. All the DBAssociations involved are notified so that they can update the display
self.

Returns the DBEntity to which the DBFetchGroup belongs.

fetchContentsOf:aSource usingQualifier:aQualifier

Replaces the content of the current DBRecordList by records fetched from the database. Any editi
fetch group is terminated and changes are lost. The argument aSource may be nil, in which case al
DBFetchGroup's entity are fetched. If aSource is a DBValue containing NULL, the effect is to cle
without fetching any new records.

Alternatively, aSource may be a DBValue that specifies a relationship. For example, suppose the r
entity called Department to the entity called Employees, containing the employees belonging to ea
DBValue may contain a specific value for the property ªDepartment Numberº and also the entity t
(Employees). Records will be fetched for all employees in the indicated department, using the key
Number as a foreign key that qualifies the selection of records from Employees.

The argument aQualifier is a DBQualifier that further restricts the records that will be fetched.

If the parent DBModule's delegate responds to fetchGroupWillFetch:, it is notified. Similarly, afte
DBModule's delegate responds to fetchGroupDidFetch:, it is a notified. Provided the fetch is succ
DBAssociations are notified that the contents of their views has changed, so they can redraw thems
record index is set to 0 (the index of the first record). Returns self.

(BOOL)hasUnsavedChanges

Returns YES if there are unsaved changes in this DBFetchGroups's DBRecordList, or in any of its
DBRecordLists, and NO otherwise.

initEntity:anEntity

Initializes an instance of DBFetchGroup. The fetch group thus initialized will coordinate fetches f
DBModule from the DBEntity named anEntity. Returns self.

(BOOL)insertNewRecordAt:(unsigned int)index

Instructs the DBFetchGroup's DBRecordList to insert a new record at the position indicated by ind
negative, the method appends the new record (that is, inserts it at the end of the DBRecordList inst

Returns YES if the DBRecordList is able to comply, and NO otherwise. A NO return may arise if
authorization to modify rows, if no records have been fetched, or if for any reason the DBRecordL

If the DBFetchGroup has appointed a delegate and the delegate implements the method fetchGroup
the method insertNewRecordAt: notifies the delegate. The delegate may then fill in default values

makeAssociationFrom:anExpr to:aView

(const char *)name

Returns the name of the DBFetchGroup. Fetch groups that are created automatically are given nam
names assigned in the model. Fetch groups that were created by the application and initialized (for
initEntity:) remain unnamed until explicitly named by setName:.

setName:

(unsigned int)recordCount

Returns the number of records in the DBFetchGroup's DBRecordList.

recordList

Returns the DBRecordList instance that the receiving DBFetchGroup serves.

redisplayEverything

Causes redisplay of all the fields governed by the DBFetchGroup's DBAssociations. (The redispla
sending all the DBAssociations a notification that the contents changed, and they respond in the sa
other change to their contents.) As a side effect, this method checks the value of the current record
of range, sets it to the index of the last record. Returns self.

removeAssociation:anAssociation

Removes the indicated association from the DBFetchGroup's list of associations. Returns self.

saveChanges

Saves changes made to any of the records governed by the receiving DBFetchGroup and any subor
DBFetchGroups. Before saving, the method terminates any editing that may have been in progress
DBFetchGroups. After saving, notifies the DBModule's delegate that the save took place. Return

(unsigned int)selectedRowAfter:(unsigned int)previousRow

Returns the index of the first selected row that is located after the row specified by previousRow.
selected row, also known as the current row. But under some conditions the user may select multip
the return is the index of the first of them.)

If no row is selected, or the only selected rows occur earlier than previousRow, returns DB_NoInd
methods interpret to mean ªafter the last recordº).

setCurrentRecord:(unsigned int)newIndex

Sets the index of the current record to newIndex. However, if the proposed value is less than the i
sets it to the first record if the proposed value is greater than the last record, sets it the last record.
method changes the current record index, the DBFetchGroup's DBAssociations are notified that th
(and can update the display accordingly). Returns self.

setDelegate:anObject

Makes anObject the DBFetchGroup's delegate. Returns self.

delegate

setName:(const char *)aName

Sets the name of the DBFetchGroup. This method is invoked automatically when the fetch group
application will need to call it explicitly only if you explicitly create a new fetch group. Returns se

name

takeValueFromAssociation:anAssociation

Takes a value from the part of the display governed by anAssociation, and inserts it in the corresp
DBFetchGroup's DBRecordList. The method then updates the display of other displayed fields th
other DBAssociations belonging to the same DBFetchGroup. Returns self.

(BOOL)validateCurrentRecord

Returns YES if changes that have been proposed for the current record are valid (or if there is no c

The validation is done in two stages. If there is a TextField editor for the field that changed, it revi
If the TextField editor says NO, that's the return. If there is no text field editor, or the editor raises
change, the task of validation is passed to the DBModule's delegate. (Each DBFetchGroup is own
Whatever the delegate returns becomes the return for this method.

fetchGroup:fetchGroup didInsertRecordAt:(int)index

Notification that the DBFetchGroup fetchGroup has inserted a record in its DBRecordList at the p
index.

fetchGroup:fetchGroup willDeleteRecordAt:(int)index

(DBFailureResponse)fetchGroup:fetchGroup willFailForReason:(DBFailureCode)code

Invoked when a failure is reported from the DBRecordList owned by fetchGroup. The reason for f
one of the following DBFailureCodes:

DB_ReasonUnknown = 0
DB_RecordBusy
DB_RecordStreamNotReady
DB_RecordHasChanged
DB_RecordLimitReached
DB_NoRecordKey
DB_RecordKeyNotUnique
DB_NoAdaptor
DB_AdaptorError
DB_TransactionError

The failure response that is returned must be one of the following constants, declared as type DBFa
header file dbkit/enums.h:

DB_NotHandledDisplays a default attention panel but takes no other action

DB_AbortTerminates the operation that encountered the error in its present state, and displays a

DB_ContinueIgnores the problem permits the action to continue if possible.

If the delegate does not implement this method, the effect is the same as returning DB_NotHandle

(BOOL)fetchGroup:fetchGroup willValidateRecordAt:(int)index

Notification that the DBFetchGroup fetchGroup, while preparing to save its DBRecordList, has rea
which it would be appropriate to insert validity checks on the record indicated by index. These mi
internal consistency between fields, or even checks that require a separate query to the database. V
fields (ªIs this a valid phone number?º or ªIs this in a valid format for a telephone number?º) shou
field editor notices that the user has changed a field's display (see the DBModule delegate method

If your implementation of fetchGroup:willValidateRecordAt: returns YES (or if your delegate doe
method), the record is treated as valid. If it returns NO, the record is treated as invalid, the attempt
and the user is notified by an attention panel.

fetchGroupDidFetch:fetchGroup

Invoked when fetchGroup has completed a fetch from the database.

fetchGroupDidSave:fetchGroup

Invoked when fetchGroup has completed a save to the database.

fetchGroupWillChange:fetchGroup

Invoked when fetchGroup is about to record change based on input from the user interface.

(BOOL)fetchGroupWillSave:fetchGroup

Invoked when fetchGroup is about to save the contents of the fetch group to the database.