setFirst

setLast
setNext
setPrevious
setKey:andLength:
getKey:andLength:
isMatch

readRange:ofLength:atOffset:
removeValue

(IXBTree *)btree

Returns the IXBTree that the IXBTreeCursor accesses. This is useful if your code didn't initialize
since it allows your code to reconfigure the IXBTree, or empty or free it.

initWithBTree:

(BOOL)getKey:(void **)aKey
andLength:(unsigned int *)aLength
withHint:(unsigned int *)aHint

Returns by reference the key defining the IXBTreeCursor's position in its IXBTree, along with the
hint that your code can use to speed up a subsequent key search for the same key using setKey:and
hint is guaranteed to remain useful as long as no insertions or removals are performed however, th
changes, the less useful the hint becomes.

getKey:andLength: (IXCursorPositioning protocol), setKey:andLength:withHint:

initWithBTree:(IXBTree *)aBTree

Initializes the IXBTreeCursor to work with aBTree. It will use that IXBTree's comparator or com
position itself at keys. This is the designated initializer for IXBTreeCursors. Returns self.

btree

(unsigned int)readRange:(void **)aRange
ofLength:(unsigned int)aLength
atOffset:(unsigned int)anOffset

Copies a portion of the value in the IXBTree at the IXBTreeCursor's position, and returns the leng
may be less than the length requested). If there is no key/value pair at the IXBTreeCursor's positi
IX_NotFoundError exception is raised.

readValue:, writeValue:andLength:, writeRange:ofLength:atOffset:, openRange:ofLength:atOff
removeValue

writeValue:, removeValue, readRange:ofLength:atOffset:, writeRange:ofLength:atOffset:

removeValue

Removes the key and the associated value at the IXBTreeCursor's position. This method raises IX
there is no key at the IXBTreeCursor's position. Returns self.

readValue:, writeValue:, readRange:ofLength:atOffset:, writeRange:ofLength:atOffset:

(BOOL)setKey:(void *)aKey
andLength:(unsigned int)aLength
withHint:(unsigned int)aHint

Positions the IXBTreeCursor at aKey if aKey is stored in the IXBTree otherwise, positions the IXI
aKey would be (which may be between two keys, or off either end of the key space). aLength is th
Returns YES if aKey is in the IXBTree (that is, if the IXBTreeCursor finds the key), and NO if it's

getKey:andLength:withHint:, getKey:andLength: (IXCursorPositioning protocol), setKey:andLe
(IXCursorPositioning protocol)

writeRange:(void *)aRange
ofLength:(unsigned int)aLength
atOffset:(unsigned int)anOffset

Writes aRange over a portion of the value in the IXBTree at the IXBTreeCursor's position. Data s
within the IXBTree's value is overwritten for aLength bytes. If the range would extend past the en
value is enlarged to hold the new amount. Returns self.

readRange:ofLength:atOffset:, readValue:, writeValue:andLength:, removeValue

(BOOL)writeValue:(void *)aValue andLength:(unsigned int)aLength

Writes aLength bytes from aValue as the value in the IXBTree at the IXBTreeCursor's position, p
previously stored value. Returns YES if the write resulted in an insertion, and NO if the write ove
stored value.