NEXTSTEP

Title: Trailing spaces in fixed length non null character fields

Entry Number:
Last Updated: 22 November 1995
Document Revision:

Keywords: space, Sybase, character, adaptor

Question

After installing EOF Release 1.1, I noticed that my column attributes which are defined as fixed length non-null character fields are now blank padded when fetched. How can I correct this behavior?

Answer

With the EOF 1.1 Sybase adaptor, non-null fixed length CHAR fields (for example CHAR(8)) end up space padded when fetched. This behavior is peculiar to Sybase, even though EOF Release 1.0 automatically removes these spaces.

You can return to the same behavior as in EOF 1.0 by adding the following code to your application. Note that you must statically link the Sybase adaptor with your application in order to use this workaround with the following Makefile.preamble option:

```
OTHER_OFILES: /NextLibrary/Adaptors/Sybase.dbadaptor/Sybase
```

```
// You may freely copy, distribute, and reuse the code in this example.
// NeXT disclaims any warranty of any kind, expressed or  implied, as to its
// fitness for any particular use.
//
// This code uses private API of the Sybase adaptor and will likely need
// to be changed for future releases of EOF.
//
// Below, we create a subclass of a private part of the Sybase adaptor
// (SybaseStringColumn) so that we can strip trailing blanks on columns
// declared "char not null".  Char columns which can be null and varchar
// columns automatically have trailing spaces removed.  If we were to let
// these extended values slip through to the database layer of EOF all kinds
// of problems would ensue because "hello" != "hello " in C, but they would
// be compared as equal in the db.

#import <eoaccess/eoaccess.h>

#define SYBASE_CHAR_TYPE 47

@interface SybaseColumn : NSObject
{
    void *_channel;
    void *_dbProcess;
    void *_attribute;
    int _srcType;
}
@end

@interface SybaseStringColumn:SybaseColumn
{
}
- valueFromBytes:(void *)bytes length:(int)length zone:(NSZone *)zone;
@end

@interface HackedSybaseStringColumn:SybaseStringColumn
{
}
+ load;
- valueFromBytes:(void *)bytes length:(int)length zone:(NSZone *)zone;
@end
```

```
@implementation HackedSybaseStringColumn

+ load
{
    [self poseAsClass:[SybaseStringColumn class]];
    return self;
}

- valueFromBytes:(void *)bytes length:(int)length zone:(NSZone *)zone
{
    const char *ptr;

    if (_srcType == SYBASE_CHAR_TYPE) {
        ptr = (const char *)bytes + length - 1;
        while (ptr >= (const char *)bytes && *ptr == ' ')
            ptr--;
        length = ptr - (const char *)bytes + 1;
    }

    return [super valueFromBytes:bytes length:length zone:zone];
}

@end
```

Valid for: EOF 1.1, NEXTSTEP 3.2 Developer, NEXTSTEP 3.3 Developer

See Also: