

```
//
// FILENAME: eTDocInfoUI.m
// SUMMARY: Inspector/Controller for eText-based eTDocInfos
// SUPERCLASS: Object:eTDocInfoUI
// PROTOCOLS: <Inspectable>
// INTERFACE: eTDocInfoUI.nib
// AUTHOR: Rohit Khare
// COPYRIGHT: ©1993,94 California Institute of Technology, eText Project
//
// Implementation Comments
// This is a shared inspector (once copy of the nibs, can inspect any
// eText DocInfo). It is also used as an "implicit" annotation in eText docs.
//
// HISTORY
// 10/31/94: Changed fontwell setup completely.
// 10/04/94: Revamped for eText5; renamed eTDocInfoUI
// 01/15/94: Created for eText4.0. Borrows from DocInspector in eText3.0
//
// Imported Interfaces
//
// #import "eTDocInfoUI.h"

@implementation eTDocInfoUI
//
// Class Management
//
+ new {
    static eTDocInfoUI *ui = nil;

    if (!ui) {
        ui = [[eTDocInfoUI alloc] init];
    }
    return ui;
}

- free {
    return self;
}

- init {
    char buf[MAXPATHLEN];
    NXBundle *bundle;

    [super init];
    bundle = [NXBundle bundleForClass:[eTDocInfoUI class]];
    if ([bundle getPath:buf forResource:"eTDocInfoUI" ofType:"nib"] ) {
        [NXApp loadNibFile:buf owner:self withNames:NO];
    } else {
        NXLogError("NIB not found: eTDocInfoUI");
    }
}
```

[illegible]

[illegible]

```
[theDocInfo setRTFComments:NXUniqueString(data)];  
free(data);
```

```
[navigator touch];
```

```
} else if ((oldInspector == taggingView) && [theDocInfo etDoc]){  
    [theDocInfo setTag:H1 to:[heading1Well font]];  
    [theDocInfo setTag:H2 to:[heading2Well font]];  
    [theDocInfo setTag:H3 to:[heading3Well font]];  
    [theDocInfo setTag:H4 to:[heading4Well font]];  
    [theDocInfo setTag:H5 to:[heading5Well font]];  
    [theDocInfo setTag:H6 to:[heading6Well font]];  
    [theDocInfo setTag:QT to:[quotationWell font]];  
    [theDocInfo setTag:BODY to:[bodyWell font]];  
}  
return self;
```

```
- activateInspector: (View *) newInspector ofType: (const char *) type {  
    if (newInspector == docInfoView) {  
        NXStream *stream;  
        char *buf;  
        BOOL editOK;  
  
        [docIDField setStringValue:[theDocInfo docIDStr]];  
        [dateField setStringValue:[theDocInfo lastModifyDate]];  
        [title setStringValue:[theDocInfo docTitle]];  
        [keywords setStringValue:[theDocInfo keywords]];  
        [author setStringValue:[theDocInfo author]];  
        [parent setStringValue:[theDocInfo parent]];  
        [peers setStringValue:[theDocInfo peers]];  
        // Note that this call is in the eText category of theDocInfo; eDraw?  
        [trailerSw setState:[theDocInfo emitTrailer]];  
        buf = (char *)[theDocInfo rtfComments];  
        if (buf) {  
            stream = NXOpenMemory(buf, strlen(buf), NX_READONLY);  
            [comments readRichText:stream];  
            NXClose(stream);  
        } else {  
            [comments setSel:0 :[comments textLength]];  
            [comments setText:@""];  
        }  
        editOK = ([theDocInfo etDoc] != nil);  
        [docIDField setEditable:editOK];  
        [dateField setEditable:editOK];  
        [title setEditable:editOK];  
        [keywords setEditable:editOK];  
        [author setEditable:editOK];
```

```

[parent      setEditable:editOK];
[peers      setEditable:editOK];
[trailerSw  setEnabled:editOK];
[comments  setEditable:editOK];
} else if (newInspector == taggingView) {
    BOOL editOK = ([theDocInfo etDoc] != nil);

    [heading1Well  setFont: [theDocInfo tagFont:H1]];
    [heading1Well  setEnabled:editOK];
    [heading2Well  setFont: [theDocInfo tagFont:H2]];
    [heading2Well  setEnabled:editOK];
    [heading3Well  setFont: [theDocInfo tagFont:H3]];
    [heading3Well  setEnabled:editOK];
    [heading4Well  setFont: [theDocInfo tagFont:H4]];
    [heading4Well  setEnabled:editOK];
    [heading5Well  setFont: [theDocInfo tagFont:H5]];
    [heading5Well  setEnabled:editOK];
    [heading6Well  setFont: [theDocInfo tagFont:H6]];
    [heading6Well  setEnabled:editOK];
    [quotationWell setFont: [theDocInfo tagFont:QT]];
    [quotationWell setEnabled:editOK];
    [bodyWell      setFont: [theDocInfo tagFont:BODY]];
    [bodyWell      setEnabled:editOK];
}
return self;

```

[illegible]

```

[heading1Well setFont:[theDocInfo defaultTagFont:H1]];
[heading2Well setFont:[theDocInfo defaultTagFont:H2]];
[heading3Well setFont:[theDocInfo defaultTagFont:H3]];
[heading4Well setFont:[theDocInfo defaultTagFont:H4]];
[heading5Well setFont:[theDocInfo defaultTagFont:H5]];
[heading6Well setFont:[theDocInfo defaultTagFont:H6]];
[quotationWell setFont:[theDocInfo defaultTagFont:QT]];
[bodyWell setFont:[theDocInfo defaultTagFont:BODY]];      return [self
touch];
}

- changeTagFont:sender {
    if ([sender isKindOfClass:[FontWell class]]) {
        int choice = NXRunAlertPanel("Change Tagging Font",
            "Would you like to replace all instances of this tag "
            "font in your document? This can't be undone.",
            "Replace", "Don't Replace", "Cancel");
        if (choice == NX_ALERTOTHER) {return nil;}
        else if (choice == NX_ALERTDEFAULT) {
            // OK, so now we want to message eTextObj to replace fonts
            [[[theDocInfo etDoc] docUI] eTextObj]
                replaceFont:[sender font] with:[sender proposedFont]];
        }
        [self touch];
    }
    return self;}

- (Font *)heading1 {
    return [heading1Well font];}
- (Font *)heading2 {
    return [heading2Well font];}
- (Font *)heading3 {
    return [heading3Well font];}
- (Font *)heading4 {
    return [heading4Well font];}
- (Font *)heading5 {
    return [heading5Well font];}
- (Font *)heading6 {
    return [heading6Well font];}
- (Font *)quotation {
    return [quotationWell font];}
- (Font *)body {
    return [bodyWell font];}

```

@end