Q: I've created a palette object with subviews.   These subviews are created within the **init:** method for my object.   I store the **id's** for these subviews in instance variables in the object.   When I instantiate one of these objects by dragging it from the palette, the subviews are thereÐthey are visibleÐyet none of my code for resizing or accessing these subviews is working.   Why?

Q: When I instantiate and initialize one of my palette objects, the instance variables do not contain the values to which they were initialized.   What's going on?

A: If you debug the object that you have written within the context of InterfaceBuilder (try using **fprintf(stderr, ...)** statements, if nothing else), you will discover that all of the instance variables and outlets are nil, even though at one point all of the instance variables were initialized and the objects to which the outlets once pointed still exist.   This is because the object is being archived and unarchived.   It is your responsibility to provide the specific archiving and unarchiving code for each of your objects.   This is accomplished within the **read:** and **write:** methods for your object.   Within those methods, you must read and write *all* of the instance variables for your objectÐeven the **id's**.   If you do not do this, then instance variables contain bogus values, and outlets become disconnected even though the objects themselves exist.   (View archiving automatically handles archiving and unarchiving subviews.)

Here is a code snippet which illustrates what you must do.   The object in question contains six instance variables: 1 string, 2 booleans, and 3 id's which point to subviews of the object.

```
- read:(NXTypedStream*)stream
{
        [super read:stream];
        NXReadTypes(stream,"cii", myTitle, &bordered, &gridEnabled);
```

```
        /* read in the id's for the outlets */
                myTitleField = NXReadObject(stream);
                grapherView = NXReadObject(stream);
                nameField = NXReadObject(stream);
                return self;
        }

        - write:(NXTypedStream*)stream
        {
            [super write:stream];
                NXWriteTypes(stream,"cii", myTitle, &bordered, &gridEnabled);

        /* write the id's for the outlets */
                NXWriteObjectReference(stream, myTitleField);
                NXWriteObjectReference(stream, grapherView);
                NXWriteObjectReference(stream, nameField);
                return self;
        }
```

QA754

Valid for 2.0, 3.0