If you want to set a thread's apartment model via Thread.CurrentThread.ApartmentState, you need to act quickly

devblogs.microsoft.com/oldnewthing/20140811-00

August 11, 2014

Raymond Chen

Welcome to CLR Week 2014. Don't worry, it'll be over in a few days.

A customer wanted to know why their `FolderBrowserDialog` was displaying the infamous *Current thread must be set to single thread apartment (STA) mode before OLE calls can be made* error.

```
private void btnBrowseFolder_Click(object sender, System.EventArgs e)
{
  Thread.CurrentThread.ApartmentState = ApartmentState.STA;
  FolderBrowserDialog fbd = new FolderBrowserDialog {
    RootFolder = System.Environment.SpecialFolder.MyComputer,
    ShowNewFolderButton = true,
    Description = "Select the awesome folder..."
  };
  DialogResult dr = fbd.ShowDialog();
  ...
}
```

"Even though we set the `ApartmentState` to `STA`, the apartment state is still `MTA`. Curiously, if we put the above code in a standalone test program, it works fine."

The problem is that the customer is changing the apartment state too late.

> On the first call to unmanaged code, the runtime calls **CoInitializeEx** to initialize the COM apartment as either an MTA or an STA apartment. You can control the type of apartment created by setting the System.Threading.ApartmentState property on the thread to **MTA**, **STA**, or **Unknown**.

Notice that the value you specify in `CurrentThread.ApartmentState` is consulted at the point the runtime initializes the COM apartment (which occurs on the first call to unmanaged code). If you change it after the COM apartment has been initialized, you're revising the blueprints of a house after it has been built.

The standard way to avoid this problem is to attach the `[STAThread]` attribute to your `Main` function, or if you need to set the apartment model of a thread you created yourself, call the `Thread.SetApartmentState` method before the thread starts.

Raymond Chen

Follow