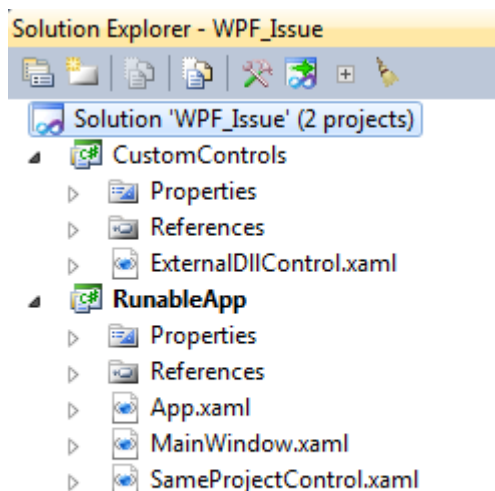


Eloquera and WPF

Во время экспериментов с объектной базой Eloquera, я столкнулся с одним очень неприятным эффектом, который меня напрягал в течение месяца наверно или даже больше. За это время я успел замучать самих создателей базы, отписался на официальном форуме DevExpress, даже вовлек в эту котовасию Дона Смита после конференции Patterns'n'Practices и его коллег. В итоге, после многочисленных писем со стэк-трейсами и примерами кода, проблема была решена. Но обо всем по порядку.

Частой практикой при разработке сколько-нибудь сложного приложения является вынос всех визуальных компонентов в отдельную библиотеку. Т.е. существует исполняемый проект с основной формой и проект с пользовательскими экранами (UserControls), которые отображаются на основной форме. В простейшем случае все это выглядит примерно так в обозревателе проектов:



Используя простейший код для отображения пользовательских компонентов, можно убедиться что все хорошо, компоненты отображаются и приложение работает как ожидалось.

```
public partial class MainWindow : Window {
    public MainWindow() {
        InitializeComponent();

        content.Children.Clear();
        content.Children.Add(new ExternalDllControl());
    }
}
```

Здесь за переменной content скрывается StackPanel, так что приводить код основной формы не стоит.

Далее начинаются волшебные вещи, которые можно было охарактеризовать фразой из старинного анекдота: «Если я тебя по голове ударю, у тебя шнурки развяжутся?», так вот шнурки развязывались. При создании экземпляра класса базы данных Eloquera, выкидывалось сообщение

о невозможности найти визуальный компонент в библиотеке CustomControl. Такое поведение характерно только для работы базы в конфигурации Desktop.

The component 'CustomControls.ExternalDllControl' does not have a resource identified by the URI '/CustomControls;component/externaldllcontrol.xaml'.

```
public partial class MainWindow : Window {
    public MainWindow() {
        InitializeComponent();

        var db = new DB("server=(local);options=none");

        content.Children.Clear();
        content.Children.Add(new ExternalDllControl());
    }
}
```

Дальнейшая работа была невозможна. Ошибка происходила в независимости от того, в каком потоке создавалось соединение с базой данных.

Проблему можно было обойти перенеся все компоненты в исполняемую сборку, но это было частичное решение и работало только если не использовались визуальные компоненты третьих сторон, с которыми могли произойти те же самые сложности.

О причинах такого поведения

После долгих расспросов и копаний обнаружилось, что проблема была в механизме обнаружения хранимых процедур, которые появились в версии 4.1, но задел к которым был наверно уже с версии 3.4.1. Данный механизм автоматически сканирует библиотеки в директории по умолчанию и загружает их в пространство памяти сервера. Таким образом, сервер пытался отъесть все библиотеки, и если остальному коду было все равно, на то какой у них codebase, то с WPF такой фокус не прокатил.

Решение

Решение как всегда оказалось простое и эффективное. Если не планируется использовать хранимые процедуры для базы Eloquera, то надо указать директорию для поиска библиотек с хранимыми процедурами на какую-нибудь папку, где вашего приложения точно не окажется.

Изменить путь до хранимых процедур можно декларативно в коде, причем сделать это желательно как можно раньше, до загрузки визуальных компонентов. Я поместил код в App.

```
public partial class App : Application {
    public App() {
        var db = new DB("server=(local);options=none");
        DB.Configuration.ServerSettings.StoredProceduresPath = "c:\\temp";
    }
}
```

После этого проблема исчезла и можно спокойно продолжать работу над проектом, используя в качестве базы данных Eloquera.

Hard'n'Heavy!

[Violet Tape](#)