

Создание приложения .NET на базе GDAL/OGR: загрузка слоя

[Обсудить в форуме](#) Комментариев — 13

Эта страница опубликована в основном списке статей сайта по адресу <http://gis-lab.info/qa/sharpmap-opentab.html>

Пример кода на примере данных Mapinfo TAB.

Рассмотрим пример приложения .NET, использующего GDAL/ORG для доступа к данным Mapinfo TAB.

Файлы таблиц MapInfo часто распространяются в формате *.TAB. Здесь и далее, под форматом TAB понимается целый набор файлов с расширениями *.TAB, *.MAP, *.DAT и *.ID. Если с первым из них можно посмотреть в блокноте и при желании разобраться что к чему, то с остальными тремя это не пройдет, поскольку они двоичные. А разобраться с двоичными файлами невероятно сложно. Таким образом, вариант с написанием кода обработки этих файлов вручную отпадает. На помощь приходит GDAL/OGR.

Библиотека абстракций [GDAL/OGR](#) позволяет открывать и модифицировать большое количество различных форматов. Распространяется она под лицензией [MIT](#). Это означает, что вы можете без каких-либо ограничений использовать её в своих проектах. Запрещено лишь использование доброго имени держателя авторских прав в рекламе.

Следует отметить, что если вы большой поклонник .NET, то, скорее всего, отделаться малой кровью у вас не получится, поскольку в состав GDAL/OGR входят лишь [unmanaged dll](#), которые содержат обычный машинный код. Это чревато тем, что вам придётся иметь дело с [маршалингом](#) для вызова неуправляемого (unmanaged) кода в ваших управляемых (managed) .NET-сборках. Всё это выливается в написание дополнительного кода. В идеале хочется, чтобы после добавления ссылок на все необходимые библиотеки слой открывался одной строчкой, например, так:

```
mapComponent.Map = MapOpener.Open(@"C:\Layer.TAB");
```

Решение вышеобозначенной проблемы - [SharpMap](#). SharpMap – фреймворк для .NET, который является надстройкой над GDAL/OGR и некоторыми другими библиотеками для работы с данными ГИС. Он позволяет открывать большое количество форматов файлов, содержащих ГИС-данные. Распространяется SharpMap по лицензии [LGPL](#), поэтому если вы ничего не собираетесь менять в исходном коде, то можете быть спокойны, ограничения лицензии не должны вас коснуться.

Перейдём от слов к делу и по шагам опишем, что нужно сделать для того, чтобы получить на форме вашего приложения изображение карты, имея в распоряжении установленную Visual Studio 2005/2008 и несколько TAB-файлов:

1. Загружаем [FWTools](#) (текущая версия на момент написания статьи 2.2.8). GDAL/OGR входит в этот набор программ. Устанавливаем.
2. Загружаем [последний change set](#) (набор изменений) проекта [SharpMap](#) (на данный момент последним является change set №40903). Откройте файл *.sln из скачанного архива и посмотрите на структуру решения. Изучите его и попытайтесь скомпилировать.
3. Создаём в Visual Studio пустой Solution (Blank Solution).
4. Копируем папки с проектами SharpMap, SharpMap.UI, SharpMap.Extensions и папку ExternalReferences в папку свеже созданного Solution. После чего добавляем их в среде Visual Studio. Попробуйте всё скомпилировать – должно получиться.
5. Создайте в Solution новый Window Forms проект. И добавьте в него ссылки (References) на проекты SharpMap, SharpMap.UI и SharpMap.Extensions. Сделайте Build. После билда в режиме дизайнера главной формы приложения откройте тулбокс – там должны появиться компоненты MapImage и FeatureDataSet. Расположите MapImage на форме.
6. Скопируйте все dll-файлы из папки bin, где вы установили [FWTools](#), в папку bin/Debug (или bin/Release)

вашего решения. Их там около 30Мб. Возможно, что нужны не все, но лучше перестраховаться и скопировать полный набор.

7. Исследуйте скачанный change set фреймворка [SharpMap](#) на предмет наличия примеров кода по открытию ТАВ-файлов. Используйте их в своём приложении.

Вот и всё. Осталось лишь показать работающий код.

```
public class MapOpener
{
    public static Map InitializeMap(params string layerFileNames)
    {
        Map map = new Map();

        foreach (string fileName in layerFileNames)
        {
            string layerName;

            Ogr ogr = new Ogr(fileName, out layerName);
            VectorLayer layer = new VectorLayer(layerName, ogr);

            map.Layers.Add(layer);
        }

        map.BackColor = Color.White;
        map.ZoomToExtents();

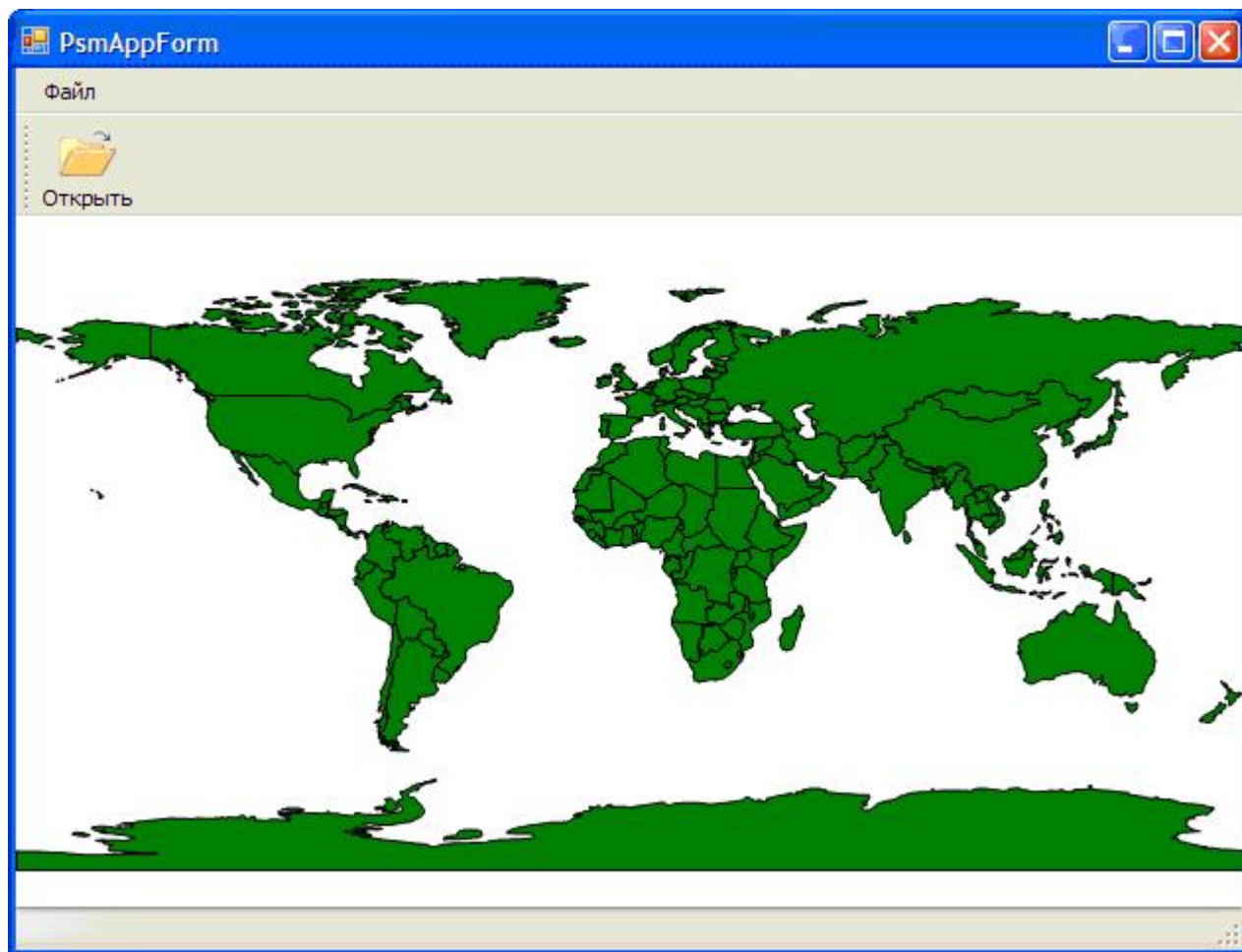
        return map;
    }
}

public partial class PsmForm : Form
{
    public PsmForm()
    {
        InitializeComponent();
    }

    private void OpenMap()
    {
        try
        {
            if (openFileDialog.ShowDialog() == DialogResult.OK)
            {
                mapImage.Map = MapOpener.InitializeMap(openFileDialog.FileNames);
            }
        }
        catch (Exception e)
        {
            MessageBox.Show(this, e.Message, "Ошибка");
        }
    }

    private void btnOpen_Click(object sender, EventArgs e)
    {
        OpenMap();
    }
}
```

Результирующее приложение может выглядеть например таким образом:



Ссылки по теме

- [SharpMap - Geospatial application framework for the CLR](#)

[Обсудить в форуме](#) Комментариев — 13

Последнее обновление: 2014-05-15 01:37

Дата создания: 12.11.2008 Автор(ы): [s1ay3r](#)