**Министерство образования и науки Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа информационных технологий и робототехники

Отделение Информационных технологий

Профиль Геоинформационные системы

**ОТЧЕТ  
по лабораторной работе №1**

Создание векторного графического редактора

по дисциплине Компьютерная графика

Выполнил студент группы 8И5Б \_\_\_\_\_\_\_\_\_\_ Смирнов П.О.

\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ 2018 г.

Отчет принят:

Ассистент отд. ИТ \_\_\_\_\_\_\_\_\_\_\_ Перенос В.С.

\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ 2018 г.

Томск 2018 г.

# Цель работы

Разработать графический редактор, а также свой собственный формат хранения векторного рисунка.

# Задачи:

* Разработать векторный редактор и собственный формат хранения векторного рисунка.
* Векторный редактор должен позволять пользователю выбирать определенный примитив, выводить его на экран с заданными параметрами.
* Создать векторный рисунок
* Сохранить получившийся рисунок в собственном векторном формате и одном из существующих растровых форматов.
* Написать парсер, анализирующий предложенный формат, и отобразить рисунок на экране, используя классы пространства имен System.Drawing. Если формат xml‐подобный, то можно воспользоваться классами пространства имен System.Xml.
* В выводах проанализировать плюсы и минусы векторного и растрового способа хранения изображений.

**Ход работы**

Сохранение векторного рисунка происходит с помощью класса XmlSerializer, который сериализует и десериализует объекты в XML-документы и из них.

[XmlInclude(typeof(Line))]

[XmlInclude(typeof(Poliline))]

[XmlInclude(typeof(OurPoint))]

[XmlInclude(typeof(List<OurPoint>))]

[XmlInclude(typeof(Bizie))]

[XmlInclude(typeof(Text))]

[XmlInclude(typeof(Cirlcle))]

[XmlInclude(typeof(Ellipsis))]

[XmlInclude(typeof(Polygon))]

[XmlInclude(typeof(Rectangle))]

public abstract class AbstractObject

{

public int Width { get; set; }

public string Color { get; set; }

public abstract void Draw(object sender, PaintEventArgs e);

}

С помощью функции toolStripButton1\_Click() в классе Form1 происходит сохранение в формат bmp.

private void toolStripButton1\_Click(object sender, EventArgs e)

{

SaveFileDialog saveFileDialog = new SaveFileDialog();

saveFileDialog.Filter = "bitmap image (\*.bmp)|\*.bmp";

if (saveFileDialog.ShowDialog() == DialogResult.OK)

{

Bitmap bitmap = new Bitmap(pictureBox1.Width, pictureBox1.Height);

using (Graphics gr = Graphics.FromImage(bitmap))

{

for (int i = 0; i < ListOfObjects.Count; i++)

{

ListOfObjects[i].Draw(this, new PaintEventArgs(gr, new System.Drawing.Rectangle(0, 0, pictureBox1.Width, pictureBox1.Height)));

}

}

bitmap.Save(saveFileDialog.FileName);

}

}

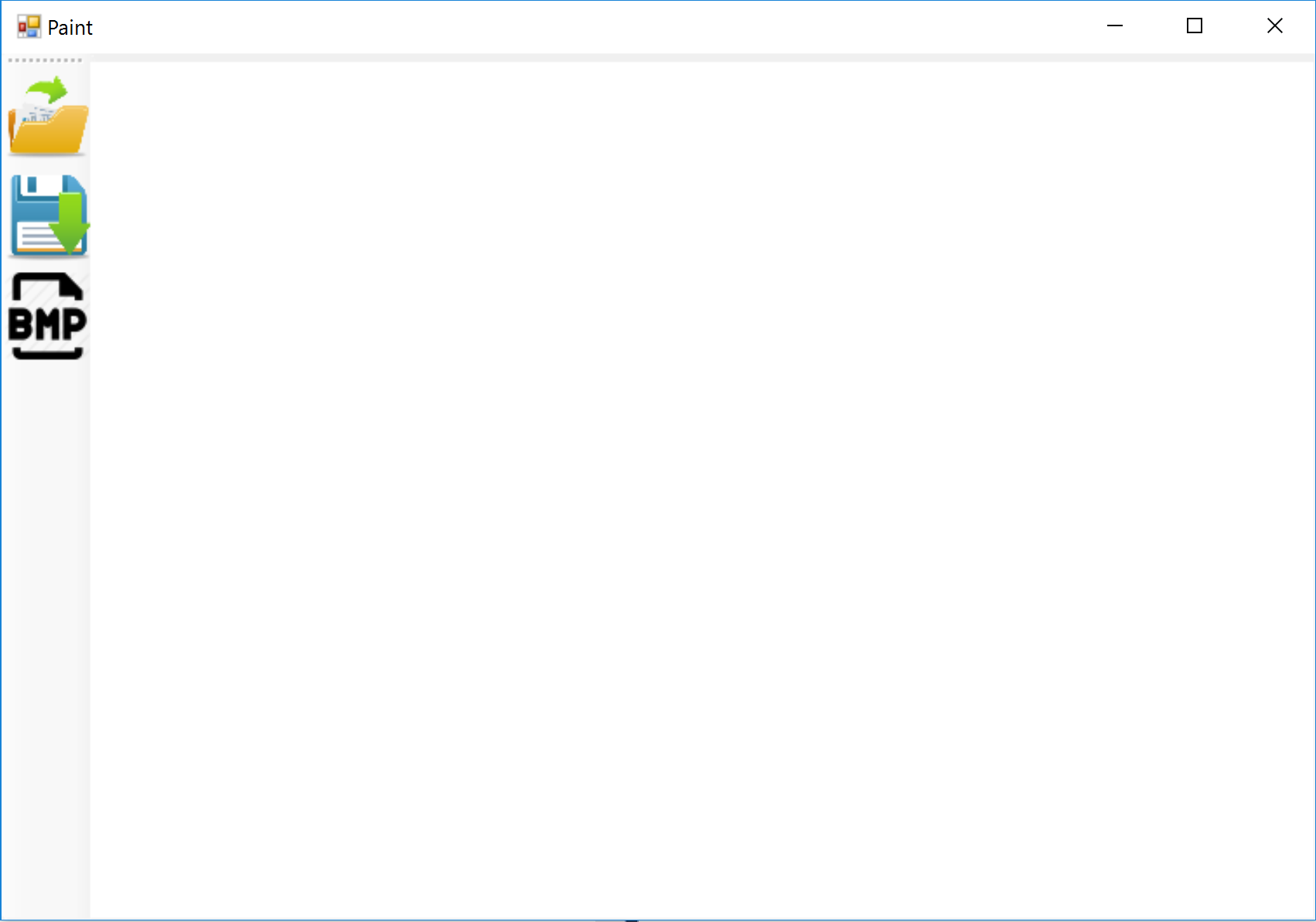


Рисунок 1 – Пользовательский интерфейс

Сбоку окна расположен элемент формы ToolStrip, содержащий 3 кнопки. Первая кнопка открывает файл с разрешением xml и отображает его содержимое в pictureBox. Вторая сохраняет объекты, изображенные на pictureBox в файл xml. Третья сохраняет изображение в формат bmp.

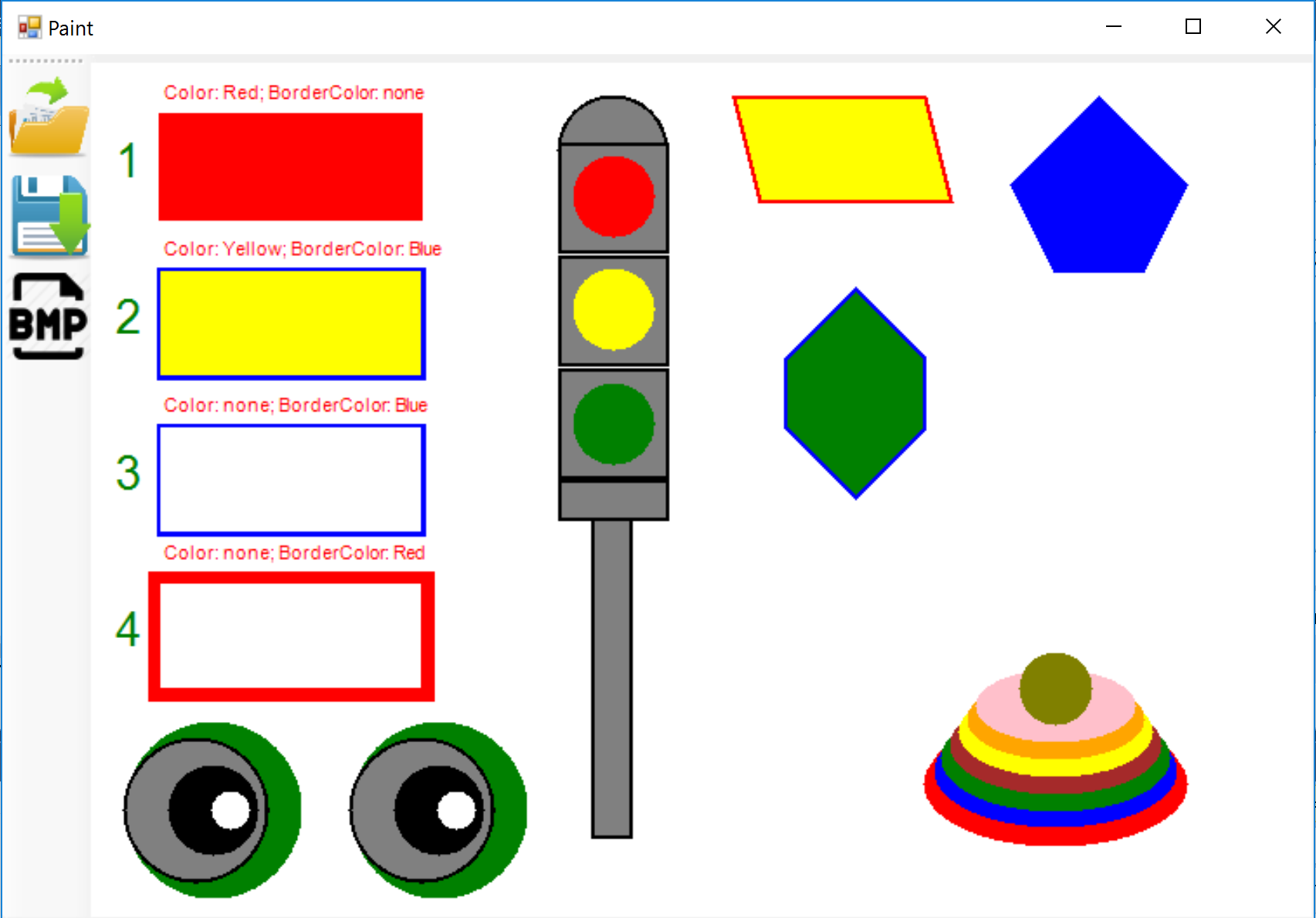


Рисунок 2 – Созданный рисунок

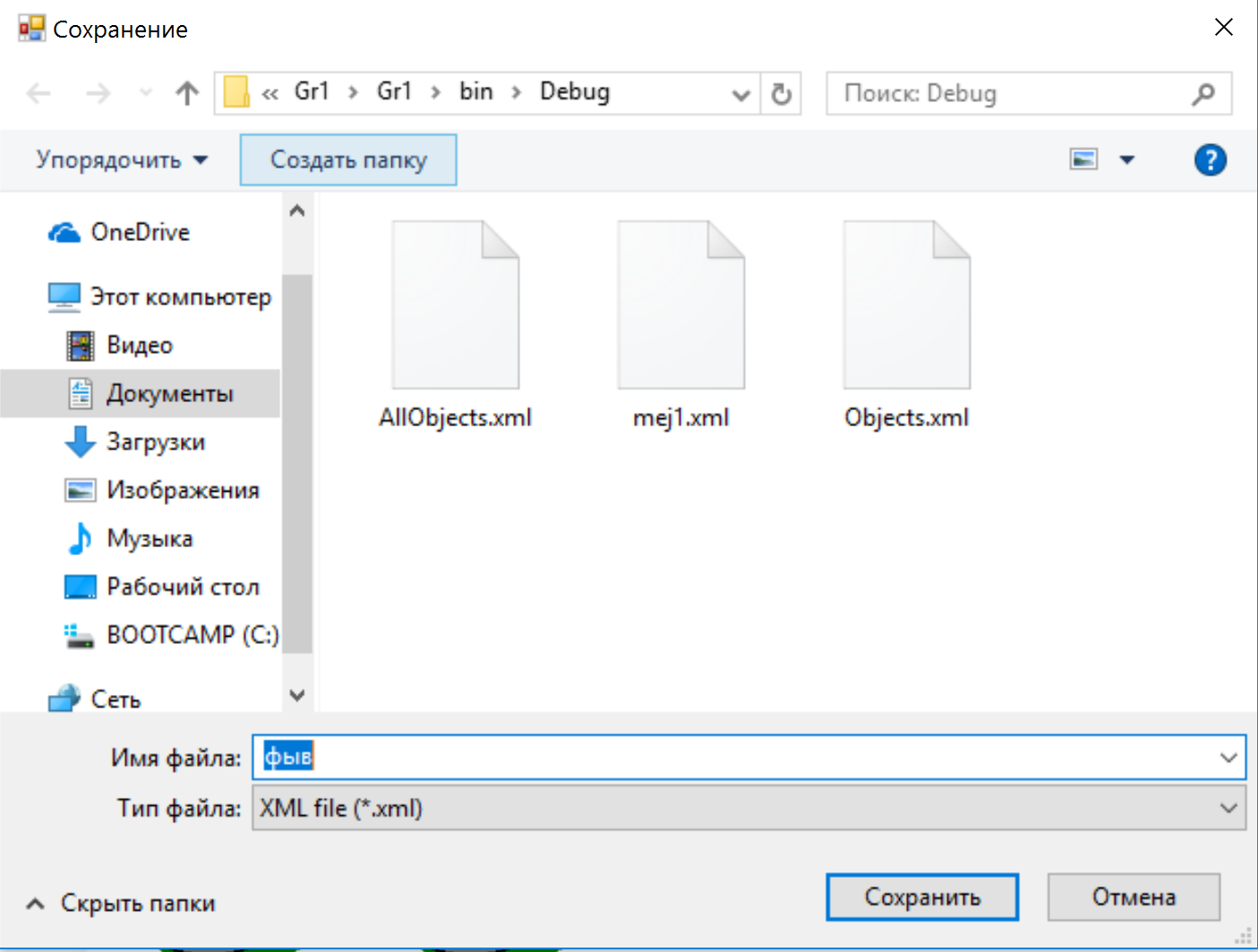


Рисунок 3 – Сохранение рисунка в формате xml

С помощью класс Parser осуществляется запись в xml документ и его чтение.

class Parser

{

public void SaveInFile(string path, List<AbstractObject> objects)

{

XmlSerializer OurSerializer = new XmlSerializer(typeof(List<AbstractObject>));

TextWriter tw = new StreamWriter(@path);

OurSerializer.Serialize(tw, objects);

tw.Close();

}

public List<AbstractObject> ReadFromFile(string path)

{

List<AbstractObject> objects = new List<AbstractObject>();

XmlSerializer OurSerializer = new XmlSerializer(typeof(List<AbstractObject>));

StreamReader or = new StreamReader(@path);

objects = (List<AbstractObject>)OurSerializer.Deserialize(or);

return objects;

}

}

# Вывод:

В ходе лабораторной работы был разработан свои графический формат и графический редактор. Каждый из форматов имеет свои плюсы и минусы. Растровый формат хранит более фотореалистичные изображения, более сглаженные. Но векторный формат занимает меньший объем памяти, также легко масштабируется без потерь в качестве.