



Web-разработка на C# и платформе Microsoft .NET

Основы технологии .NET

Типы разрабатываемых приложений

Консольные

Настольные

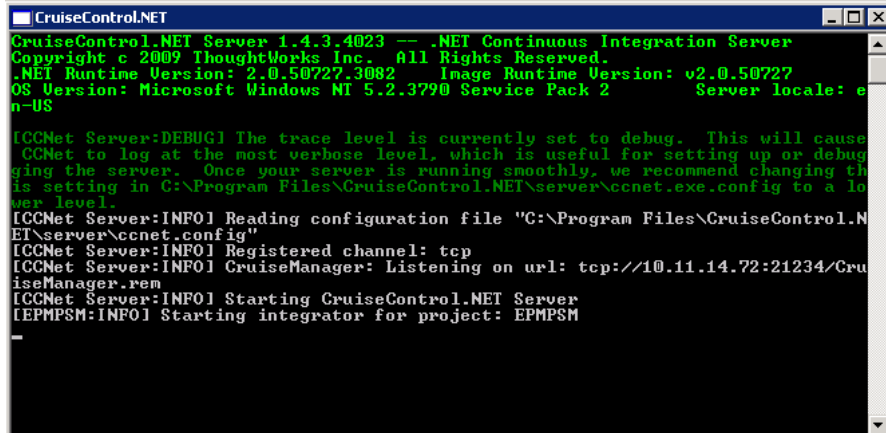
Веб

Сервисы

Библиотеки классов

Консольные приложения

- Вспомогательные утилиты
- Системы расширенных команд
- Автоматические системы



```
CruiseControl.NET
CruiseControl.NET Server 1.4.3.4023 -- .NET Continuous Integration Server
Copyright c 2009 ThoughtWorks Inc. All Rights Reserved.
.NET Runtime Version: 2.0.50727.3082 Image Runtime Version: v2.0.50727
OS Version: Microsoft Windows NT 5.2.3790 Service Pack 2 Server locale: e
n-US

[CCNet Server:DEBUG] The trace level is currently set to debug. This will cause
CCNet to log at the most verbose level, which is useful for setting up or debug
ging the server. Once your server is running smoothly, we recommend changing th
is setting in C:\Program Files\CruiseControl.NET\server\ccnet.exe.config to a lo
wer level.
[CCNet Server:INFO] Reading configuration file "C:\Program Files\CruiseControl.N
ET\server\ccnet.config"
[CCNet Server:INFO] Registered channel: tcp
[CCNet Server:INFO] CruiseManager: Listening on url: tcp://10.11.14.72:21234/Cru
iseManager.rem
[CCNet Server:INFO] Starting CruiseControl.NET Server
[EPMP5M:INFO] Starting integrator for project: EPMP5M
```

Настольные приложения

- Активно используемые приложения, установленные на компьютере пользователя

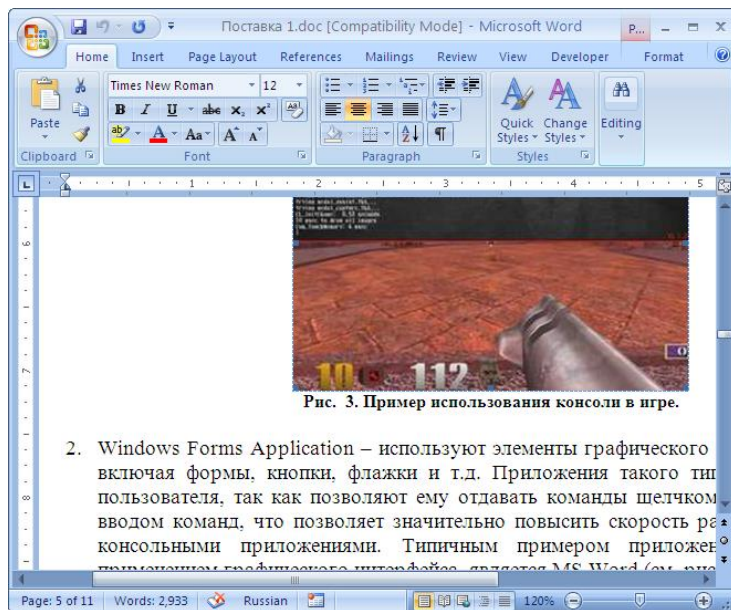
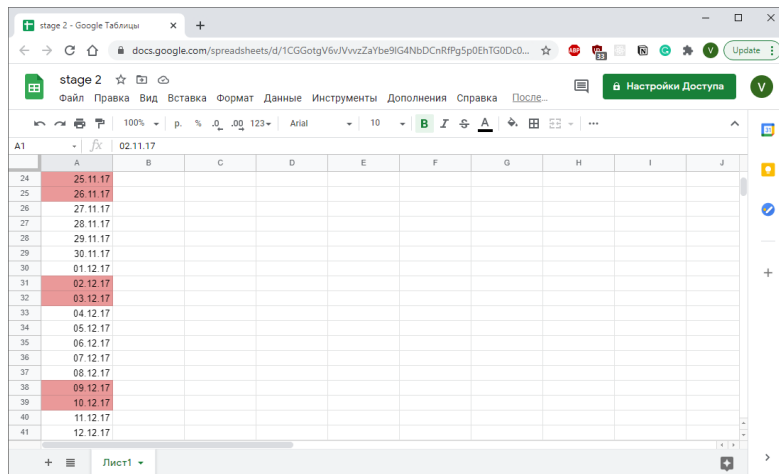


Рис. 3. Пример использования консоли в игре.

2. Windows Forms Application – используют элементы графического включая формы, кнопки, флажки и т.д. Приложения такого типа позволяют пользователю отдавать команды щелчком мыши, что позволяет значительно повысить скорость работы с консольными приложениями. Типичным примером приложения такого типа является графический интерфейс администратора MS Word (см. рис. 4).

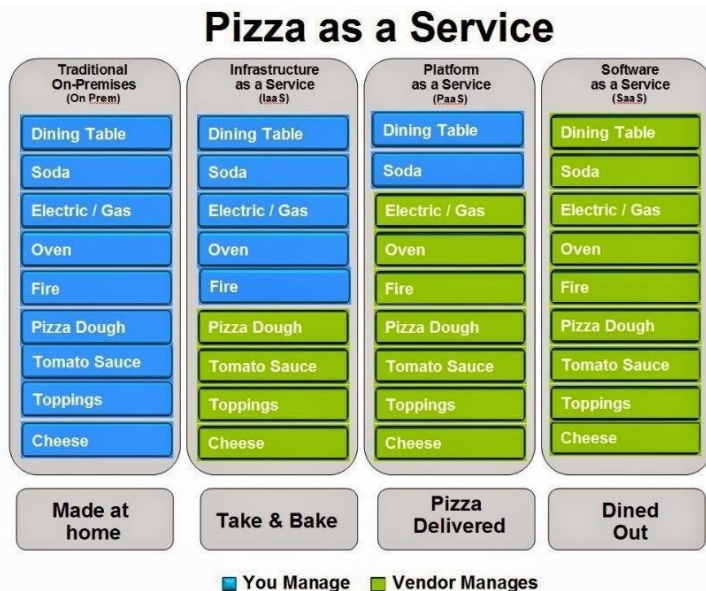
Веб приложения

- Активно используемые приложения, предназначенные для одновременной работы множества пользователей. Приложения не установлены на машине пользователя



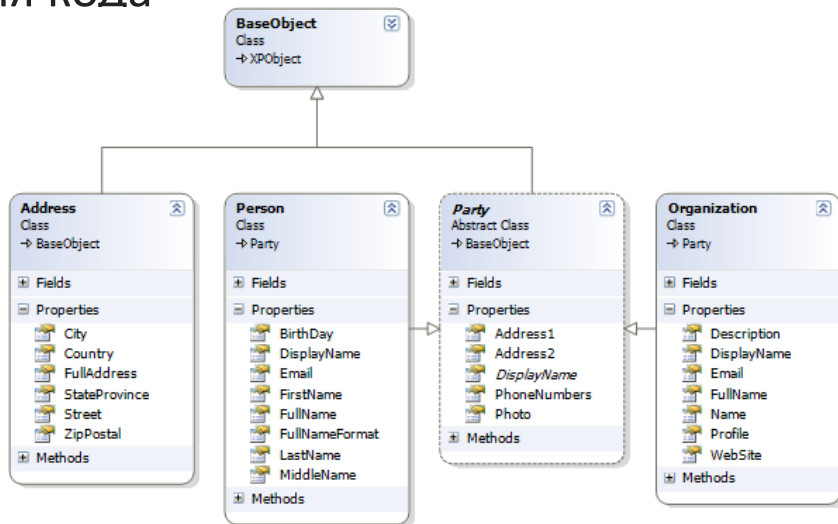
Сервисы

- Специальные web-приложения, предназначенные для предоставления другим приложениям доступа к своим данным



Библиотеки классов

- Код общего пользования (алгоритмы, удачные решения и т.д.)
- Логическая изоляция кода



Ключевые понятия

CLR – Common Language Runtime

IL – Intermediate Language

CTS – Common Type System

CLS – Common Language Specification

FCL – Framework Class Library

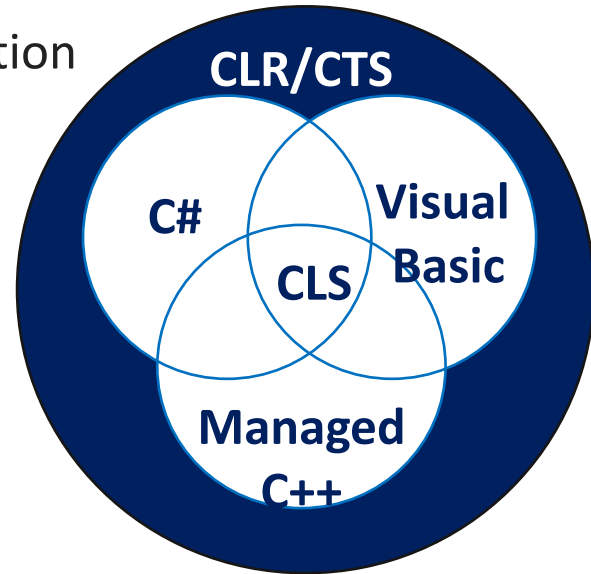


Схема трансляции приложений в .NET

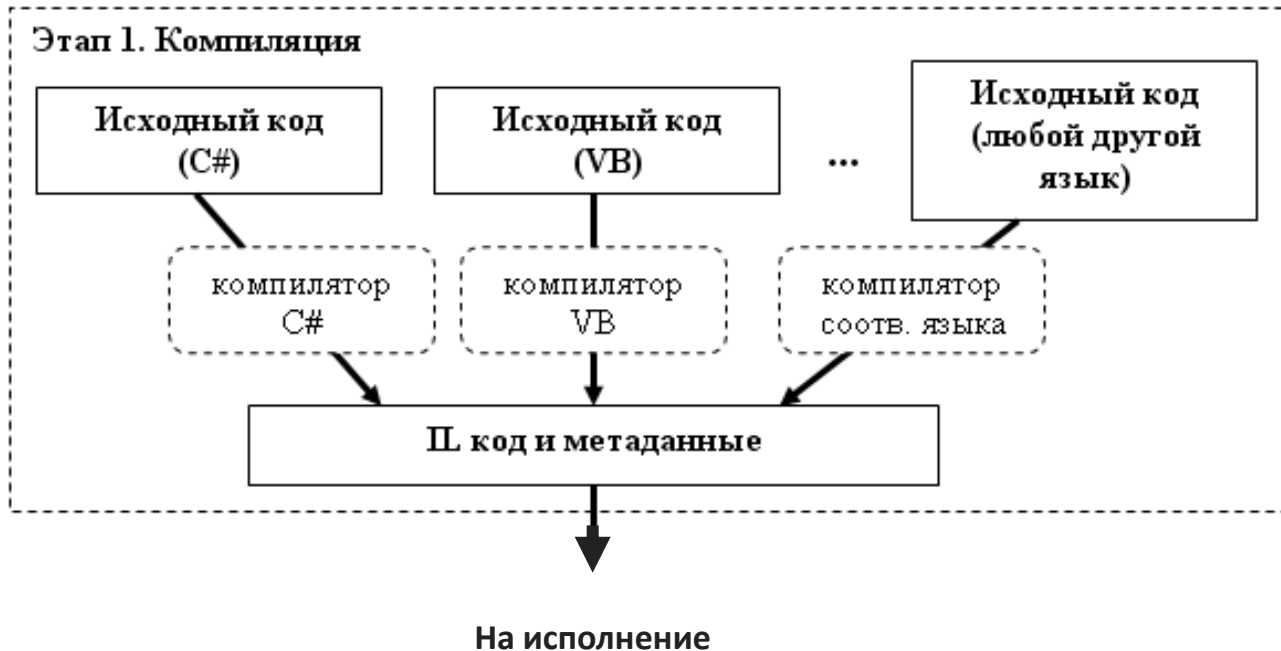
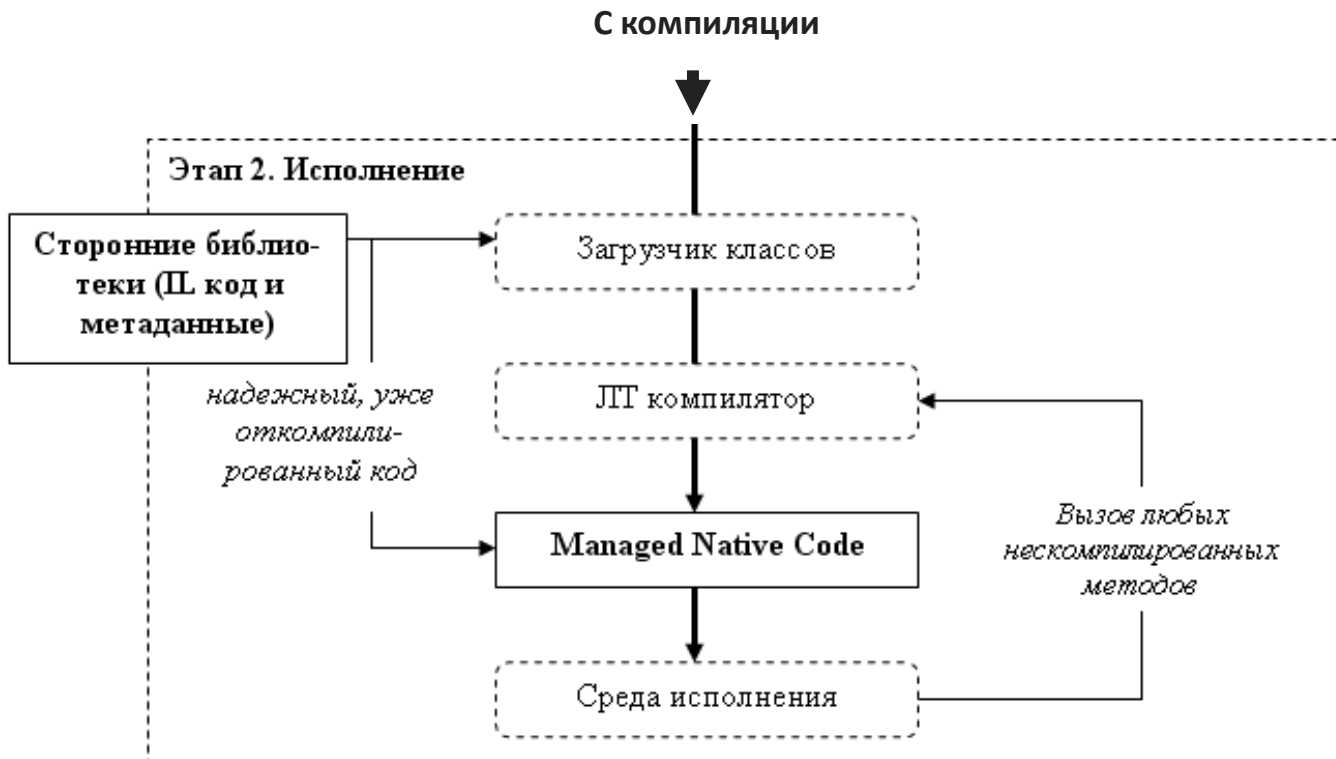
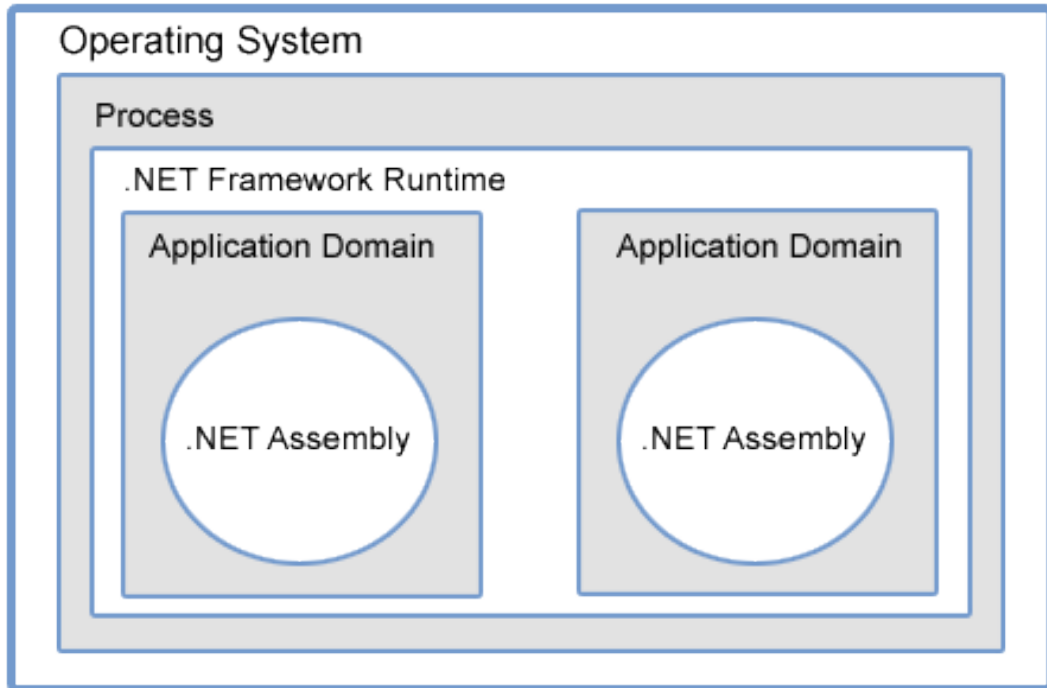


Схема трансляции приложений в .NET



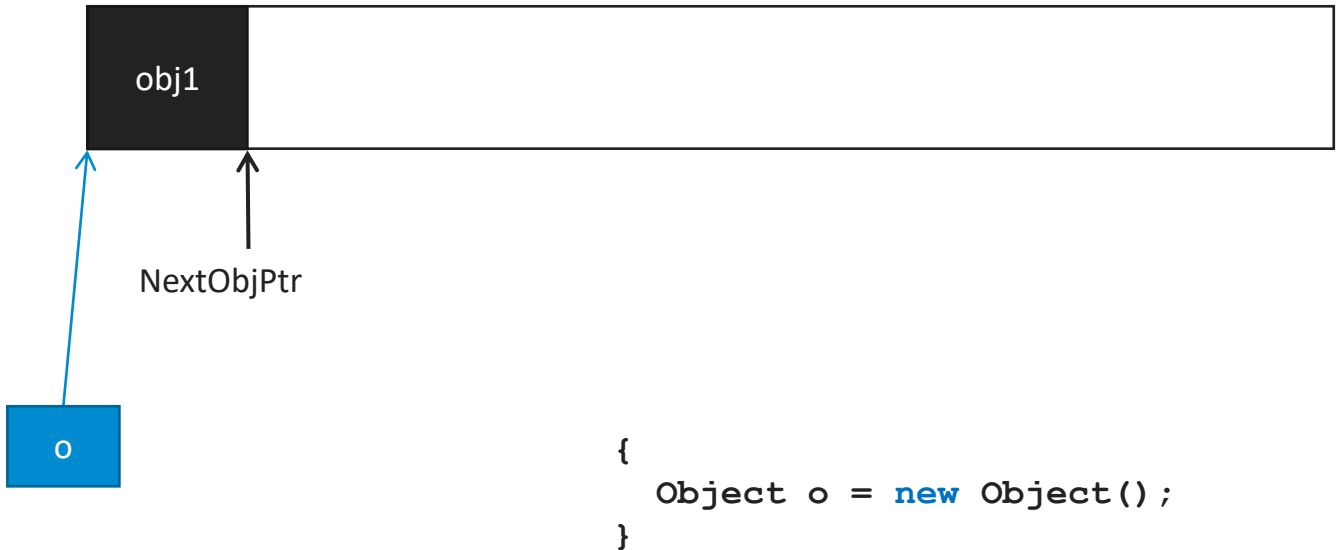
Application domain

Изолированный контейнер приложения

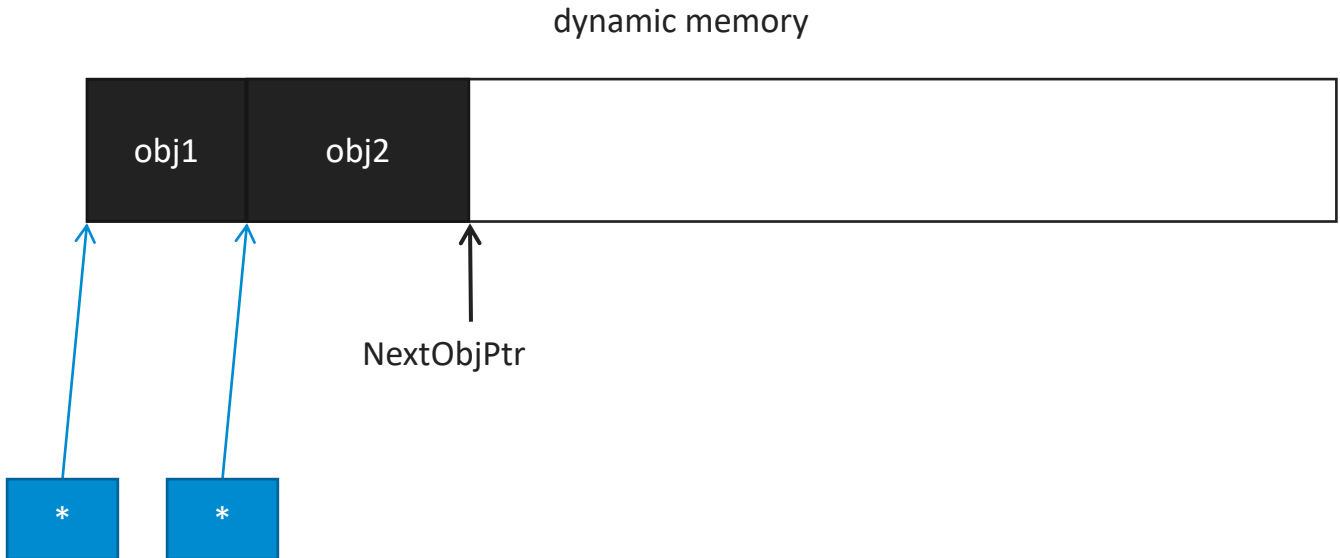


Выделение памяти

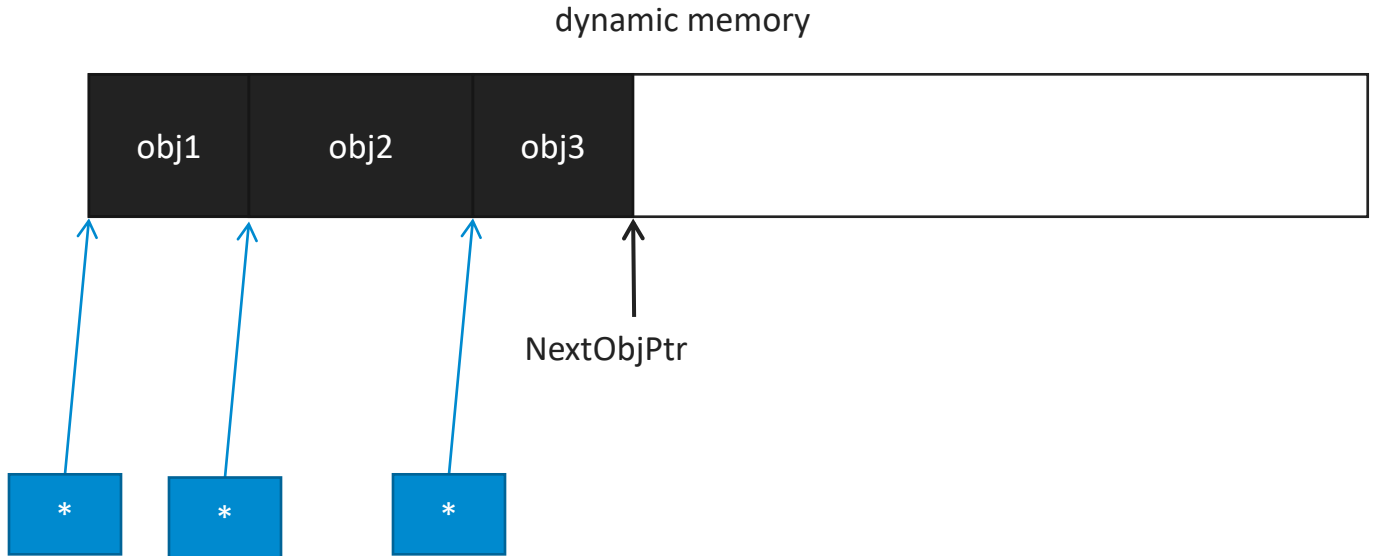
dynamic memory



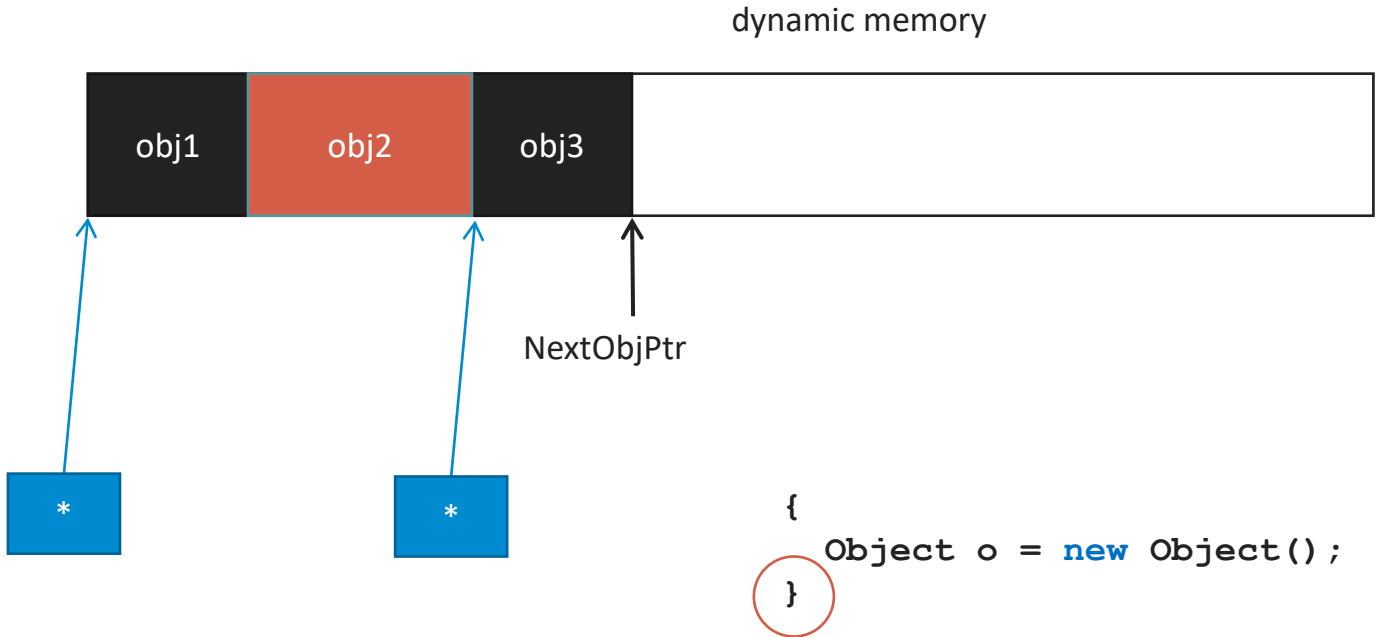
Выделение памяти



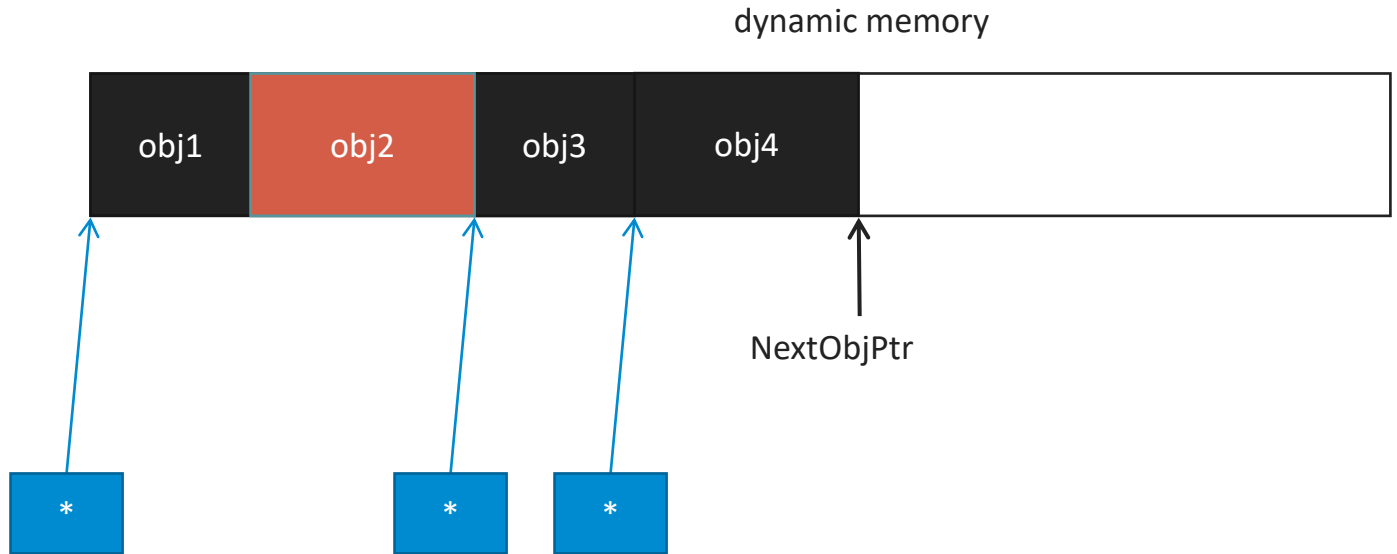
Выделение памяти



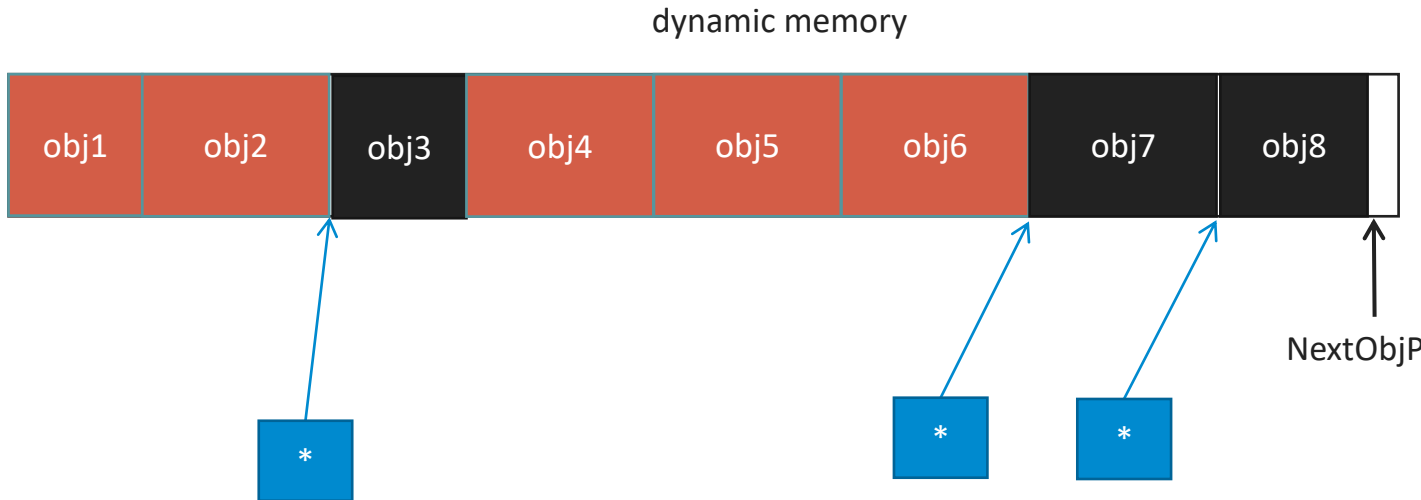
Выделение памяти



Выделение памяти



Garbage collector



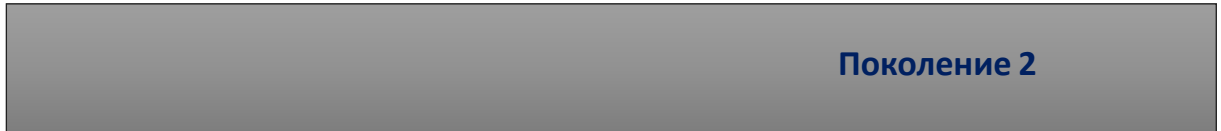
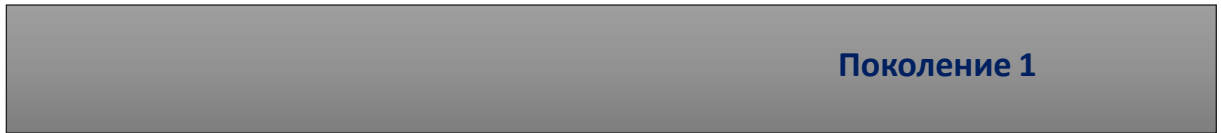
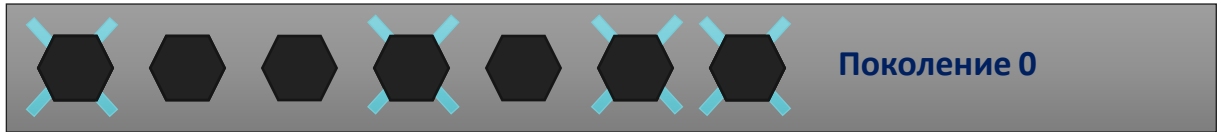
http://ru.wikipedia.org/wiki/Сборка_мусора

[http://msdn.microsoft.com/ru-ru/library/f144e03t\(v=vs.100\).aspx](http://msdn.microsoft.com/ru-ru/library/f144e03t(v=vs.100).aspx)

[http://msdn.microsoft.com/ru-ru/library/0xy59wtx\(v=vs.100\).aspx](http://msdn.microsoft.com/ru-ru/library/0xy59wtx(v=vs.100).aspx)

[http://msdn.microsoft.com/ru-ru/library/system.gc\(v=vs.100\).aspx](http://msdn.microsoft.com/ru-ru/library/system.gc(v=vs.100).aspx)

Принцип работы сборщика мусора



<http://www.rsdn.ru/article/dotnet/GC.xml>

ToString()

```
class Vector
{
    public int X { get; set; }
    public int Y { get; set; }
    public int Z { get; set; }

    public override string ToString()
    {
        return string.Format("{0}, {1}, {2}",
                               X, Y, Z);
    }
}
```

Equals()

```
class Vector
{
    public int X { get; set; }
    public int Y { get; set; }
    public int Z { get; set; }

    public override bool Equals(Vector obj)
    {
        return X==obj.X && Y==obj.Y && Z==obj.Z;
    }
}
```

Методы сравнения объектов

- Equals
- ReferenceEquals
- operator ==

GetHashCode()

```
class Vector
{
    public int X { get; set; }
    public int Y { get; set; }
    public int Z { get; set; }

    public override int GetHashCode()
    {
        return X ^ Y ^ Z;
    }
}
```

Partial классы

Файл Circle.cs

```
partial class Circle
{
    .....
}
```

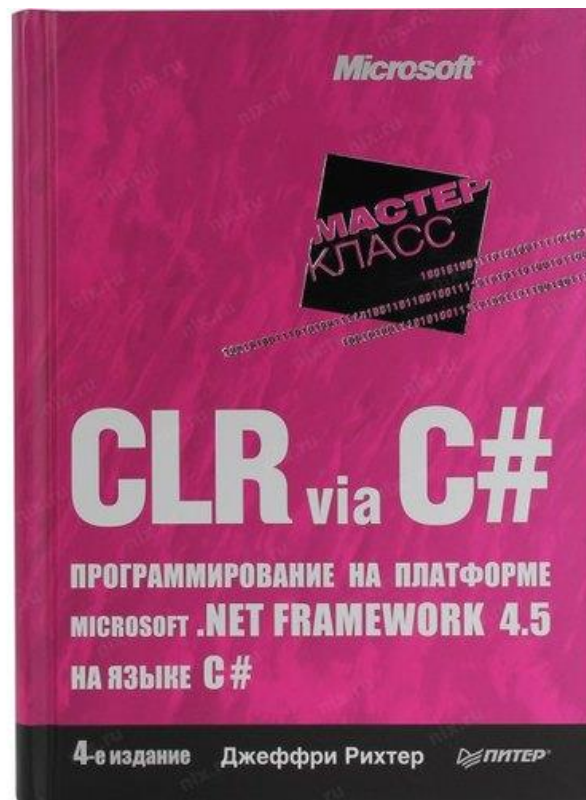
Файл Circle1.cs

```
partial class Circle
{
    .....
}
```

Что почитать

CLR via C# (Developer Reference)

by [Jeffrey Richter](#)



СПАСИБО ЗА ВНИМАНИЕ