

# СЕТЕВОЕ ВЗАИМОДЕЙСТВИЕ

**Михаил Черномордиков**

Developer Evangelist, Microsoft Россия

[mikcher@microsoft.com](mailto:mikcher@microsoft.com)

<http://blogs.msdn.com/mikcher>

# Содержание

- Взаимодействие JavaScript
- Класс WebClient
- Классы HttpRequest/HttpResponse
- Подписка на WCF Web Services
- Подписка SOAP Web Services
- Сокеты

# Класс WebClient

- Похож на Downloader
- Обеспечивает асинхронный кросс-доменный доступ к сетевым ресурсам
- Простой в использовании:

```
System.Net.WebClient client = new System.Net.WebClient();  
client.DownloadStringCompleted+=new  
DownloadStringCompletedEventHandler(client_DownloadStringCompleted);  
client.DownloadStringAsync(myURI);
```

# Web Client



*demo*

# HttpWebRequest / HttpWebResponse

- Обеспечивает асинхронный кросс-доменный доступ к сетевым ресурсам
- Позволяет получить контроль над сообщением HTTP
- Очень полезный при обращении к существующим основанным на формах (т.е. HTTP-POST) ресурсам, которые активно используются в таких технологиях как PHP

# Разработка для HTTP Post

```
Uri uri = new Uri("http://localhost:8002/Sample2Web/GetData.ashx");  
HttpWebRequest request = (HttpWebRequest)HttpWebRequest.Create(uri);  
request.Method = "POST";  
request.ContentType = "application/x-www-form-urlencoded";  
request.BeginGetRequestStream(new AsyncCallback(RequestProceed), request);
```

```
void RequestProceed(IAsyncResult asyncResult)  
{  
    ...  
}
```

```
void ResponseProceed(IAsyncResult asyncResult)  
{  
    ...  
}
```

# HttpWebRequest and HttpWebResponse *demo*



# Подписка на WCF Web Services

- Определение сервисов с помощью стандартных контрактов WCF (т.е. [DataMember] и так далее)
- Использование Visual Studio для создание прокси сервисов (based on HttpWebRequest/Resp)
- Silverlight использует этот прокси **асинхронно**



# WCF и Silverlight



# *demo*

# Подписка на SOAP Web Services

- Веб-сервисы должны предоставлять файл WSDL и сервер должен иметь crossdomain.xml или clientaccesspolicy.xml, правильно настроенный и расположенный
- Использование Visual Studio для создания прокси сервиса (основано на HttpWebRequest/Resp)
- Silverlight использует этот прокси **асинхронно**

# Поддержка сокетов

- В Silverlight возможно сетевое взаимодействие на базе сокетов
- Ограничения
  - Поддерживаются только порты из диапазона 4502-4534
  - Коммуникация возможна только с тем же доменом, на котором находится элемент управления
  - Необходим полиси-сервер, расположенный на порту 943
  - До инициализации соединения Silverlight опрашивает порт 943 для получения информации о полиси, которая содержит разрешенные порты и протоколы

# Sockets



*demo*

# СЕТЕВОЕ ВЗАИМОДЕЙСТВИЕ

**Михаил Черномордиков**

Developer Evangelist, Microsoft Россия

[mikcher@microsoft.com](mailto:mikcher@microsoft.com)

<http://blogs.msdn.com/mikcher>