Consuming OData Services

Overview

- **Querying OData services**
- **Building .NET clients**
- **Additional clients**

System query options

Option	Description	Example
$filter	Filters the result set	/Movies?$filter=Title eq 'Star Wars'
$orderby	Sort the target resources	/Movies?$orderby=Title desc
$top	Return only the top n resources	/Movies?$top=10 /Movies?$orderby=Title&$top=5
$skip	Skip the first n resources	/Movies?$skip=100&$top=10
$select	Create a projection	/Movies(1)/'Reviews?$select=rating,reviewer
$inlinecount	Includes the total count	/Movies?$top=100&$inlinecount=allpages
$value	Retrieve a value without any surrounding metadata	/Movies(2)/Title/$value
$expand	Eager loading of specified elements	/Movies(1)?$expand=Reviews

Filter query expression operators

Logical operators	Arithmetic operators
and	add
or	sub
not	mul
eq	div
ne	mod
lt	
gt	
le	
ge	

Query expression functions

String functions	Date functions	Math functions
substring, substringof,	day	round
length, indexof	hour	floor
startswith, endswith	minute	ceiling
insert, remove, replace	month	
tolower, toupper	second	
Concat, trim	year	

- **No aggregate functions allowed**

Building clients

HTTP
- Any client library that supports HTTP
- XML / JSON can be programmed directly

AJAX
- Use jQuery to get/post JSON data
- AJAX library available – includes managed context

.NET
- Libraries included in .NET Framework
- Client data context
- "LINQ to REST"

Silverlight
- Client library available as download (Silverlight 2)
- Included in Silverlight 4
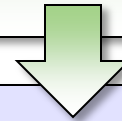
Response Formats

- **Negotiated via the HTTP Accept header**
- **Can choose from Atom (default) and JSON**

```
GET /movieservice.svc/Movies(1)
Accept: application/atom+xml
```

```
GET /movieservice.svc/Movies(1)
Accept: application/json
```

```xml
<?xml version="1.0" ?>
...
   <d:movie_id>1</d:movie_id>
   <d:title>
      Where the Wild Things Are
   </d:title>
...
```

```json
{ "d" : {
...
  "movie_id": 1,
  "title": "Where the Wild Things Are",
  ...
} }
```

Additional client SDKs available

Java

PHP

Windows Phone 7

IPhone – Objective C

Building clients

- **Services support metadata**
  - Using the $metadata query string parameter
- **Visual Studio 2008 SP 1 / 2010 supports Add Service Reference**
  - Creates client proxies based on System.Data.Services.Client library
- **Programming model varies slightly based on client**
  - Silverlight and AJAX are asynchronous

Consuming OData with existing clients

- **Syndication feed readers**
  - Support for ATOM provides simple read access
- **PowerPivot**
  - Excel / SQL Server 2008 R2 add-in
  - Native support for OData feeds
  - Perform analysis on data, manage relationships

Summary

- REST interface = reach
- Ajax or simple HTTP programming
- Client SDKs simplify programming model
- DataContext rather than proxy object