Serialization Fundamentals

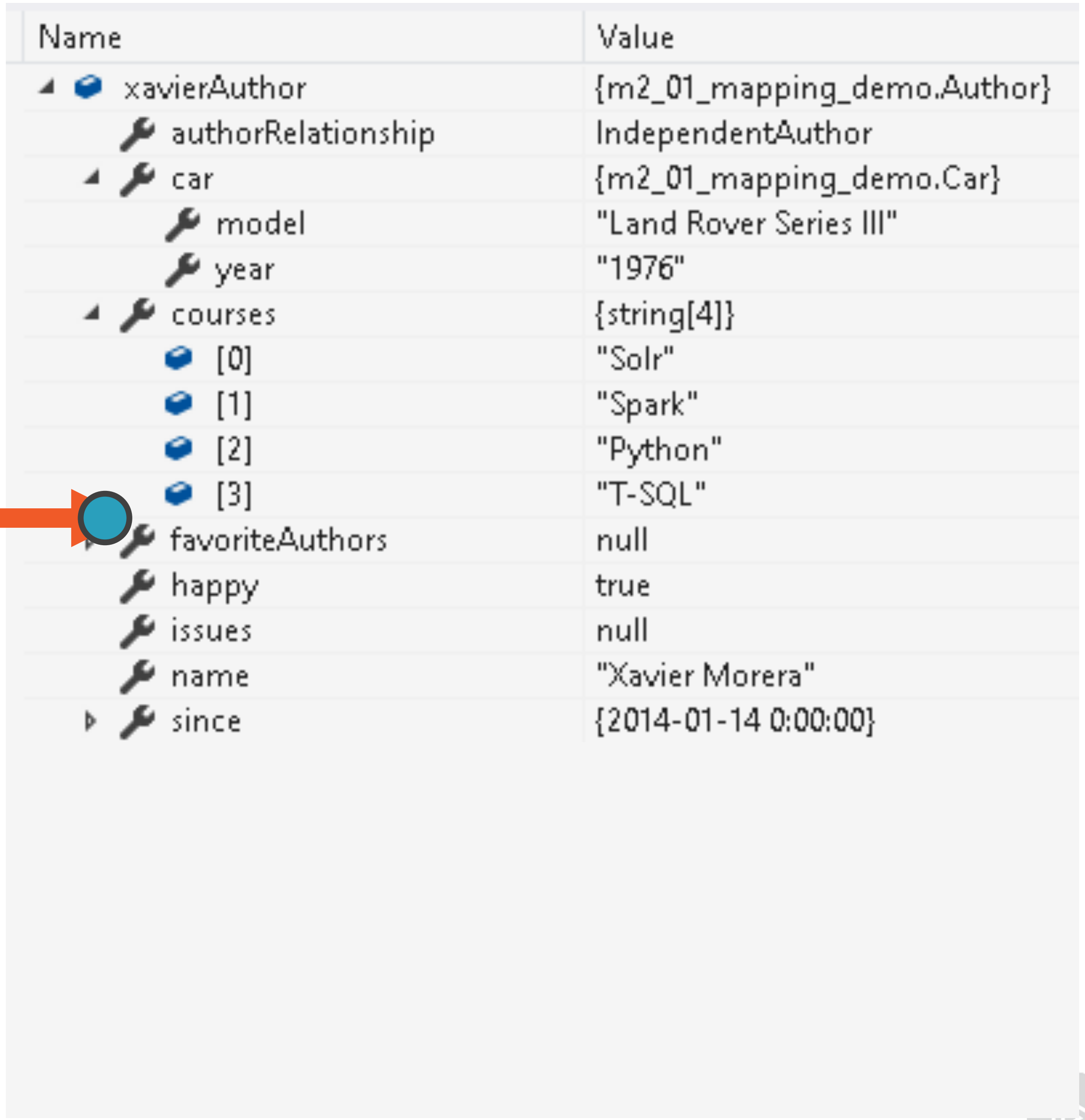
Xavier Morera
HELPING DEVELOPERS UNDERSTAND SEARCH & BIG DATA

@xmorera www.xaviermorera.com

Serialization

```
{
  "name": "Xavier Morera",
  "courses": [ "Solr", "Spark", "Python", "T-SQL" ],
  "since": "2014-01-14T00:00:00",
  "happy": true,
  "issues": null,
  "car": {
    "model": "Land Rover Series III",
    "year": 1976
  },
  "authorRelationship": 1
}
```

Deserialization



Name	Value
xavierAuthor	{m2_01_mapping_demo.Author}
authorRelationship	IndependentAuthor
car	{m2_01_mapping_demo.Car}
model	"Land Rover Series III"
year	"1976"
courses	{string[4]}
[0]	"Solr"
[1]	"Spark"
[2]	"Python"
[3]	"T-SQL"
favoriteAuthors	null
happy	true
issues	null
name	"Xavier Morera"
since	{2014-01-14 0:00:00}

Serialization & Deserialization

JsonSerializer

JsonReader

JsonWriter

```
JsonSerializer serializer = new JsonSerializer();
                            serializer.Serialize();
```

JsonSerializer

Class in charge of serialization and deserialization

Control & customization

```
string author = JsonConvert.SerializeObject(AuthorObject);
                            DeserializeObject<Author>(author);
```

JsonConvert

Easy to use

Wrapper over JsonSerializer

Settings & Attributes

```
DeserializeObject<Author>(author), new
    JsonSerializerSettings
    {
        Formatting = Formatting.Indented
    });
```

Settings & Attributes

```
 new JsonSerializerSettings
{
        Formatting = Formatting.Indented
});
```

Settings & Attributes

```
 new JsonSerializerSettings
{
            TypeNameHandling = TypeNameHandling.All,
});
```

Settings & Attributes

```
 new JsonSerializerSettings
{
            DateFormatString = "dd/MM/yyyy"
});
```

Settings & Attributes

```
 new JsonSerializerSettings
{
          Error = HandleDeserializationError
});
```

Demo

Mapping JSON to/from .NET Using JsonConvert

Demo

Object References

Demo

Dynamic Demo

Demo

Serializing Objects

Demo

Deserializing Objects

```
{
  "name": "Xavier Morera",
  "courses": ["Solr", "Spark", "Python",  "T-SQL"],
  "since": "2014-01-14T00:00:00",
  "happy": true,
  "issues": null,
  "car": {
    "model": "Land Rover Series III",
    "year": 1976
  }
}
```

```
public class Author
    {
        public string name { get; set; }
        public string[] courses { get; set; }
        public DateTime since { get; set; }
        public bool happy { get; set; }
        public object issues { get; set; }
        public Car car { get; set; }
        public List<Author> favoriteAuthors { get; set; }
}
```

One Size Fits All?

JsonReader & JsonWriter

Large objects and files

Non-cache, forward only

Reading JSON

JsonReader

Control and Performance

Non-cached, forward only

Creating JSON

JsonWriter

Demo

Using the JsonSerializer Class

Demo

Using the JsonTextReader Class

Demo

Using the JsonTextWriter Class

Dates in JSON

Dates are Tricky No Date Type	∅
Javascript Dates String or Number	‘Fri, 04 Sep 2015 01:21:31 GMT’ 1441329748
ISO 8601 Microsoft Format Custom	“date”: “2009-07-11T19:00:00-04:00” “date”: "\/Date(1247353200)\/"

Demo

Dates in JSON

Error Handling

Errors happen all the time

- Especially with JSON generated by a third party

Specify what to do with errors

- Handle
- Throw

Demo

Error Handling

Takeaway

Serialization

Deserialization

JsonSerializer

JsonConvert

Dates

Errors