JavaScript Object Notation

AKA: JSON

Who Created JSON

Doug Crockford popularized it in April 2001

Who used JSON

- Cartoon Network was the first (for chatting)

Who used JSON

- Cartoon Network was the first (for chatting)
- Sun Microsystems

Who used JSON

- Cartoon Network was the first (for chatting)
- Sun Microsystems
- Amazon.com

Who used JSON

- Cartoon Network was the first (for chatting)
- Sun Microsystems
- Amazon.com
- Electronic Data Systems (EDS)

What is JSON

Language-independent data format

Where/When JSON Is Used

- Talking to a server from the browser

Where/When JSON Is Used

- Talking to a server from the browser
- Moving data around inside of Javascript

Why use JSON

Moving data

Why use JSON

Moving data

- Browser to server

Why use JSON

Moving data

- Browser to server
- From one part of your JS app to another

Noticing the Pattern

JSON ***is*** storage for communication

JSON 6 Datatypes

1. Number - Signed decimal (with E notation)

JSON 6 Datatypes

1. Number - Signed decimal (with E notation)
2. String - Zero or more unicode characters

JSON 6 Datatypes

1. Number - Signed decimal (with E notation)
2. String - Zero or more unicode characters
3. Boolean - true or false

JSON 6 Datatypes

1. Number - Signed decimal (with E notation)
2. String - Zero or more unicode characters
3. Boolean - true or false
4. Array - Ordered list of zero or more values

JSON 6 Datatypes

1. Number - Signed decimal (with E notation)
2. String - Zero or more unicode characters
3. Boolean - true or false
4. Array - Ordered list of zero or more values
5. null - An empty value or the word null

JSON 6 Datatypes (too much text here)

1. Number - Signed decimal (with E notation)
2. String - Zero or more unicode characters
3. Boolean - true or false
4. Array - Ordered list of zero or more values
5. null - An empty value or the word null
6. Object - Unordered list of zero or more associative pairs

JSON Example

JSON describing a person:

```json
{
    "firstName": "John",
    "lastName": "Smith",
    "isAlive": true,
    "age": 25,
    "height_cm": 167.64,
    "address": {
        "streetAddress": "21 2nd Street",
        "city": "New York",
        "state": "NY",
        "postalCode": "10021-3100"
    },
    "phoneNumbers": [
        { "type": "home", "number": "212 555-1234" },
        { "type": "fax", "number": "646 555-4567" }
    ]
}
```

Using JSON in JavaScript

JSON is derived (mostly) from JavaScript

Using JSON in JavaScript

JSON is derived (mostly) from JavaScript

JSON can be natively used (mostly) in JavaScript

Using JSON in JavaScript

JSON is derived (mostly) from JavaScript

JSON can be natively used (mostly) in JavaScript

Since 2010, JSON is supported in most browsers

Using JSON in JavaScript

JSON parsing example:

```
var contact = '{"firstName":"John", "lastName":"Smith", "isAlive":true,
    "age":25, "height_cm":167.64, "address":{"streetAddress":"21 2nd Street",
    "city":"New York", "state":"NY", "postalCode":"10021-3100"},
    "phoneNumbers":[{"type":"home", "number":"212555-1234"}, {"type":"fax",
    "number":"646555-4567"}]}';

var p = JSON.parse(contact);
```

Using JSON in JavaScript

JSON parsing example:

```
var contact = '{"firstName":"John", "lastName":"Smith", "isAlive":true,
    "age":25, "height_cm":167.64, "address":{"streetAddress":"21 2nd Street",
    "city":"New York", "state":"NY", "postalCode":"10021-3100"},
    "phoneNumbers":[{"type":"home", "number":"212555-1234"}, {"type":"fax",
    "number":"646555-4567"}]}';

var p = JSON.parse(contact);
```

This shows turning a string into JSON

XML vs JSON (When Moving Data)

XML (143 Characters)

```xml
<animals>
    <dog>
        <name>Rufus</name>
        <breed>labrador</breed>
    </dog>
    <dog>
        <name>Marty</name>
        <breed>whippet</breed>
    </dog>
    <cat name="Matilda"/>
</animals>
```

Very verbose

JSON (117 Characters)

```json
{
    "animals": {
        "dog": [{
            "name": "Rufus",
            "breed": "labrador"
        }, {
            "name": "Marty",
            "breed": "whippet"
        }],
        "cat": {
            "name": "Matilda"
        }
    }
}
```

More efficient

Conclusion

JSON used for moving and storing data

Get These Slides

http://tinyurl.com/HBJSON