



Equipo 2  
Desarrollo de Aplicaciones III

# Agenda

- Historia
- Definición de Json
- Versiones
- Estructura
- Formas de representar
  - Objetos
  - Arreglos
  - Tipos de valores
- Codificador && Decodificador

# Historia

- 1999 ECMAScript Third Edition
- 2001 State Software, Inc.
- 2002 JSON.org
- 2005 Ajax
- 2006 **RFC 4627**

# ¿Qué es Json?

- *Acrónimo JavaScript Object Notation*  
(Notación de Objetos JavaScript)
- Es un formato ligero para el intercambio de datos.
- Especialmente como alternativa a XML en [AJAX](#)
- Es más pequeño que XML, más rápido y más fácil de analizar.

# Ventajas

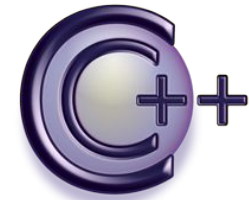
- No etiqueta de cierre
- Más rápida de leer y escribir
- Se puede analizar utilizando una función de JavaScript eval ()
- Utiliza matrices
- No hay palabras reservadas

# Versiones

- Json no tiene numero de versiones
- Json es muy estable

# Lista de lenguajes que soportan

- ActionScript
- C, C#,
- ColdFusion
- Common Lisp,
- Delphi
- E, Eiffel
- JavaScript,
- ML
- Objective-C, Objective CAML,
- Rebol
- Lua
- Visual FoxPro.



# Está constituido por dos estructuras

- Una colección de pares de nombre/valor son conocidos como.
  - *Objeto*
  - Registro
  - Estructura
  - Diccionario
  - Tabla hash
  - arreglo asociativo.
- Una lista ordenada de valores.
  - Arreglos
  - vectores
  - listas o secuencias.



**<XML>**

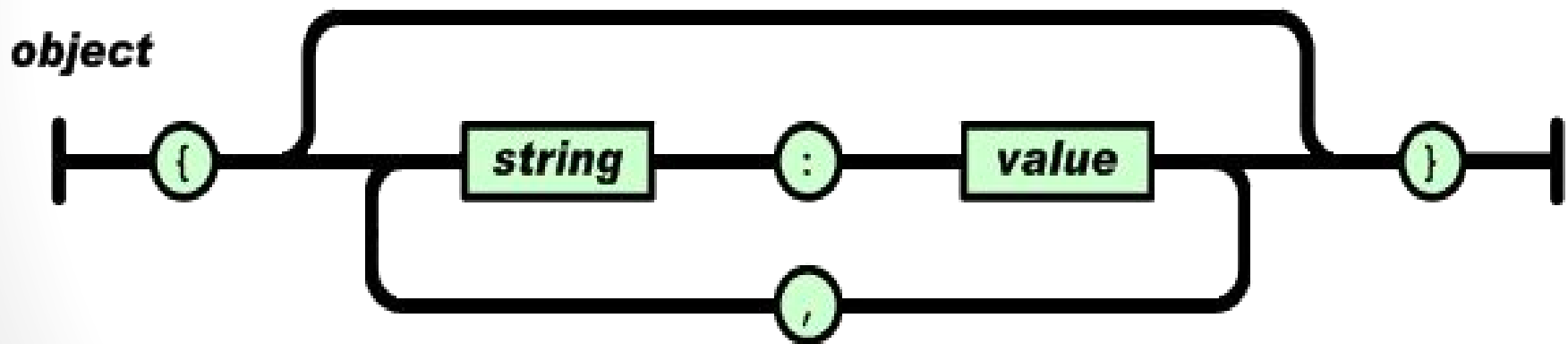
**{JSON}**

Formas de representar  
Serialization

Json

# Objeto

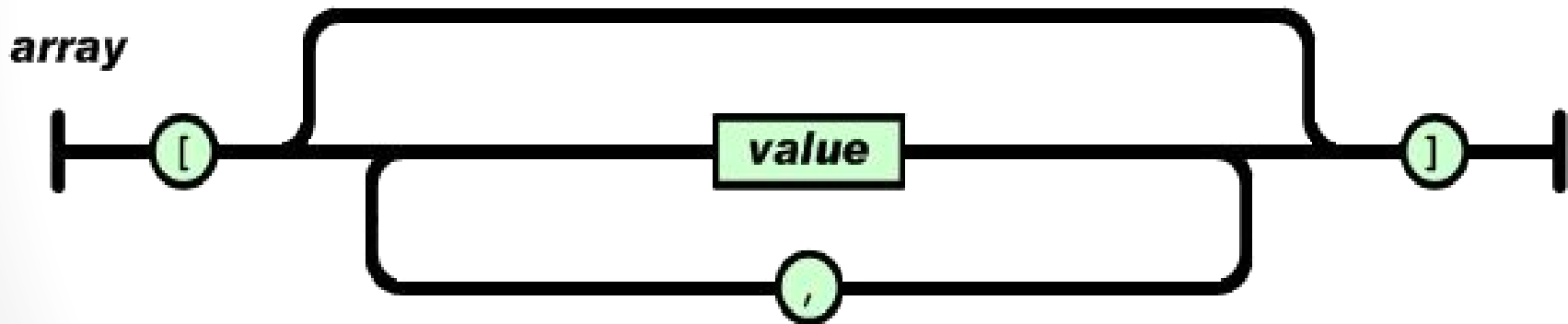
- Es un conjunto desordenado de pares nombre/valor.
- Un objeto comienza con { y termina con }.
- Cada nombre es seguido por :
- Los pares nombre/valor están separados por ,



```
{ "id" : 46, "nombre": "Miguel", "empresa": "Autentia" }
```

# Arreglo

- Es una colección de valores.
- Comienza con [ y termina con ]
- Los valores se separan por ,

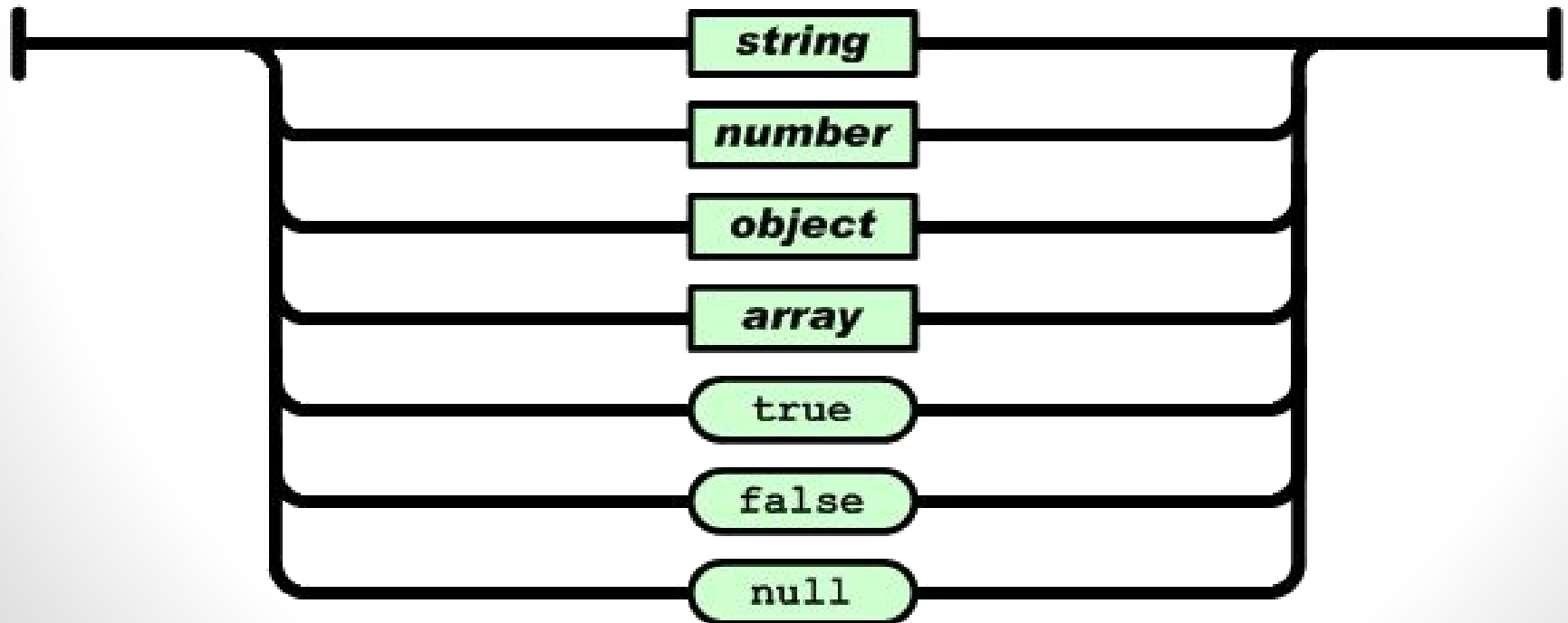


```
[  
  {"Titulo": "El señor de los anillos", "Autor": "J.R.R. Tolkien"},  
  {"Titulo": "Cancion de hielo y fuego", "Autor": "George RR Martin"},  
  {"Titulo": "Los Pilares de la Tierra", "Autor": "Ken Follett"}  
]
```

# Valor

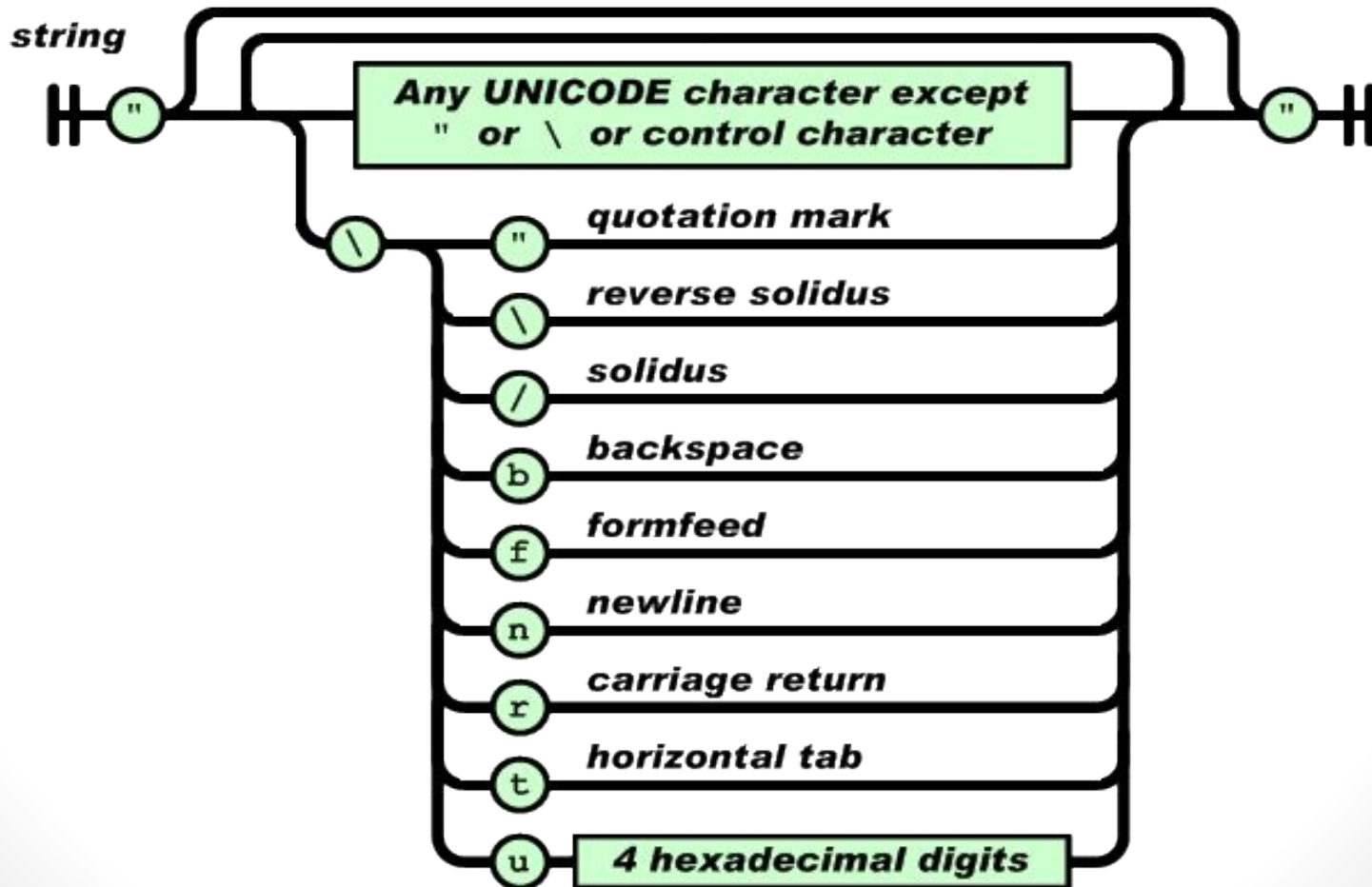
- Puede ser una *cadena de caracteres* con comillas dobles
- Un *número*
- true, false o null,
- Un *objeto* o un *arreglo*

**value**



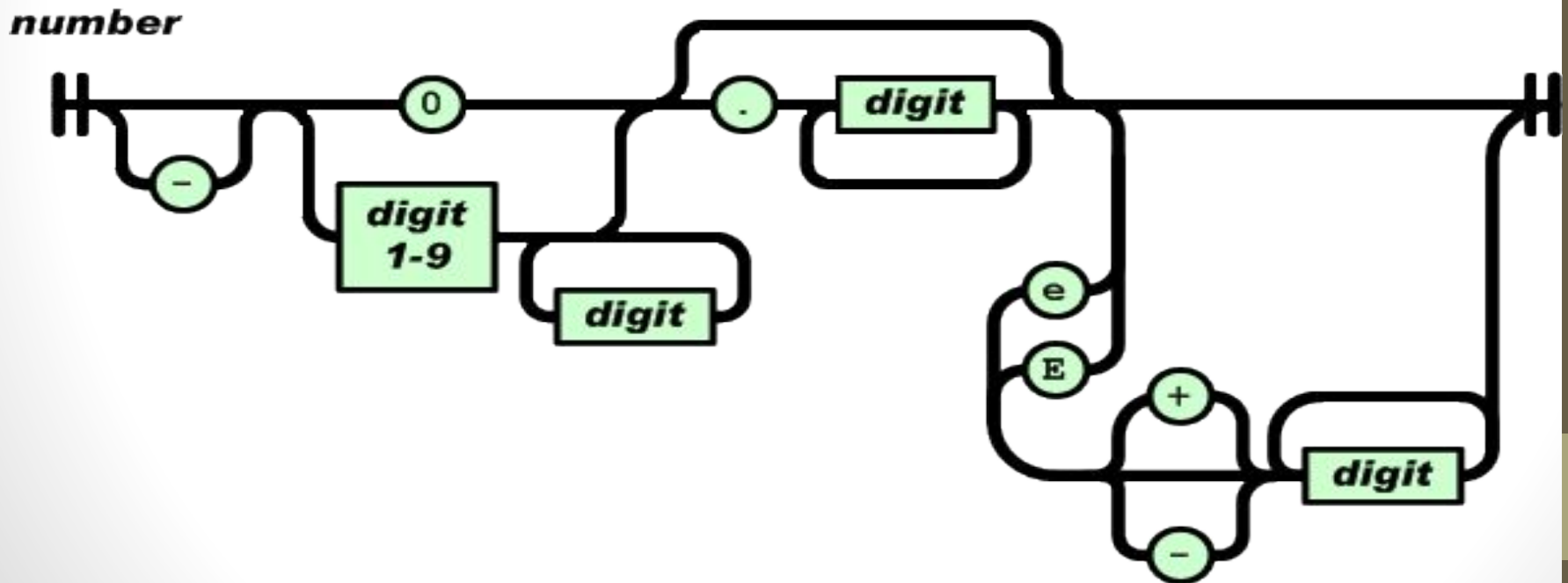
# Cadena de Caracteres

- Es una colección de cero o más caracteres Unicode, encerrados entre comillas dobles



# Numero

- Un *número* es similar a un número C o Java, excepto que no se usan los formatos octales y hexadecimales.



# Codificador y Decodificador

```
>>> import json

>>> data = {"algo": ["asi", 1, None]}
>>> json.dumps(data)
'{"algo": ["asi", 1, null]}'

>>> string = """{"probemos": 1.0}"""
>>> data = json.loads(string)
>>> data
{u'probemos': 1.0}

>>> data["probemos"]
1.0
```

También hay dos funciones para leer y escribir archivos:

- `json.load(fp)`
- `json.dump(obj, fp)`

# Referencias

- Wikipedia
- <http://es.wikipedia.org/wiki/JSON>
- Json
- <http://www.json.org/json-es.html>