

Parsear XML con ES6

Antonio Espín Herranz

Contenidos

- Objeto **DOMParser**
 - Objeto principal para parsear contenido XML
 - El objeto está disponible se lo ejecutamos desde un navegador.
 - Si se intenta acceder al objeto desde un fichero js será hay que instalar dependencias npm.
 - Si vamos a cargar un fichero XML desde el navegador se combinará con una llamada a fetch para acceder a la url.

DOMParse

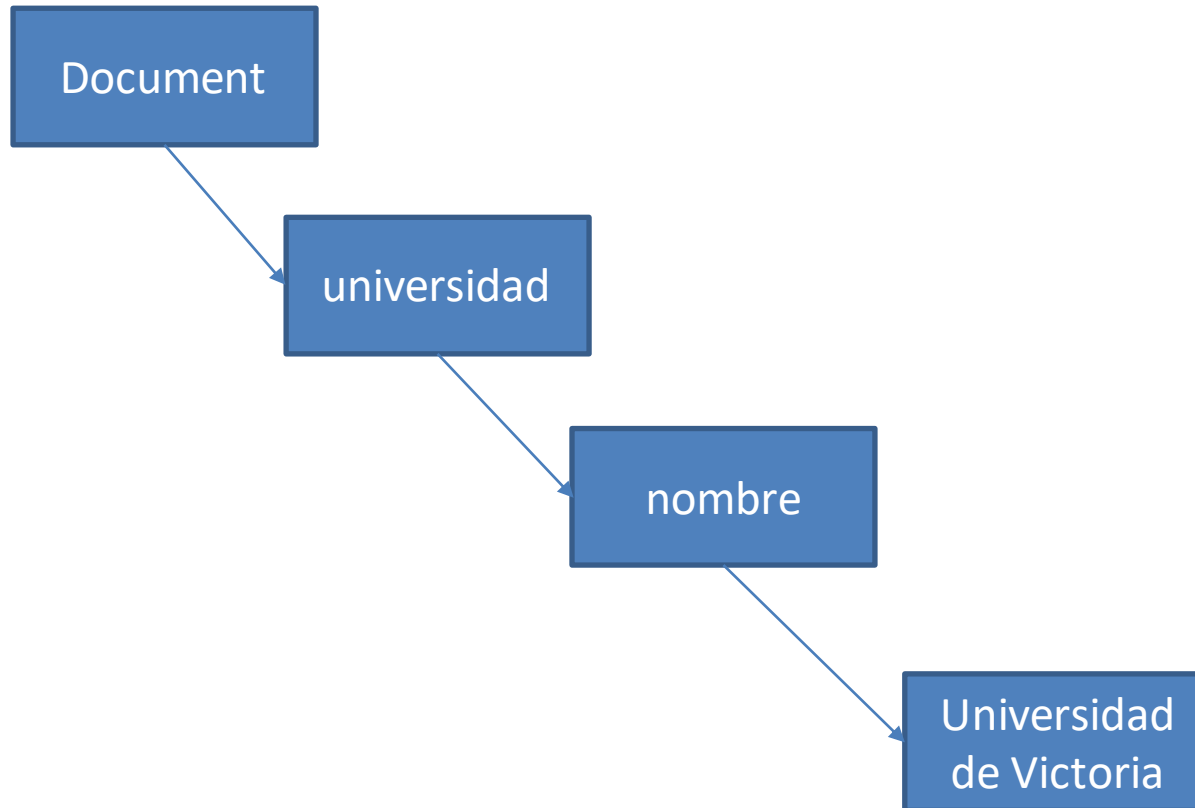
- `domParse = new DOMParse().parseFromString(texto, "application/xml")`
- La variable `texto` se habrá cargado desde el servidor.
 - Se carga en modo texto y luego se parsea a XML.

Ejemplo con un string

- Disponemos de un fragmento XML almacenado en una cadena y lo parseamos con DOMParse.

```
addEventListener('load', ()=>{  
  let xml = '<universidad><nombre id="1">Universidad de Victoria</nombre></universidad>'  
  
  let domParse = new DOMParser().parseFromString(xml, "application/xml")  
  
  let doc = domParse.documentElement  
  
  console.log(doc)  
  
  console.log('Nombre Uni: '+doc.firstChild.firstChild.nodeValue)  
}, false)
```

Representación en el DOM



```
▼<universidad>  
  <nombre id="1">Universidad de Victoria</nombre>  
</universidad>
```

Ejemplo con fetch

```
var root=null
```

```
fetch('./fichero.xml')  
  then(response=>response.text())  
  then(texto=>new DOMParser().parseFromString(texto,  
    "application/xml"))  
  then(dom=>{  
    // Capturar el nodo raíz del fichero XML  
    root = dom.documentElement  
  })  
  catch(err=>console.log(err.message))
```

document

- Una vez hemos capturado el nodo raíz :
 - Disponemos de métodos:
 - getElementByTagName
 - Las propiedades son las del DOM
 - Colecciones:
 - childNodes: Para las etiquetas hija
 - nodeNameMap : Para los atributos
 - Aplicar lo visto en el DOM