

# Procesamiento de Formatos en Aplicaciones Telemáticas: DOM

En esta unidad se trabajará con DOM (*Document Object Model*), un estándar del W3C que establece una interfaz para la representación de un documento en memoria mediante un árbol. Esta interfaz permite tanto consultar el contenido de un documento XML como modificarlo.

## Referencias básicas

Para una breve introducción a los analizadores de XML, y DOM en concreto, se recomienda consultar el capítulo 10 del libro *Learning XML, 2nd ed.*, Eirik T. Ray, O'Reilly (2003), disponible electrónicamente a través de la Biblioteca en el sistema *Safari Books Online*. En concreto, dentro del capítulo 10, son de interés la introducción y los subapartados *Streams and Events*, *Trees and Objects* y *DOM*.

Para profundizar en la API de DOM, se propone como referencia el [capítulo 5 \(DOM\) de Java and XML, 3rd ed.](#), Brett McLaughlin y Justin Edelson, O'Reilly (2006). Este libro también está disponible en Safari.

Otras referencias útiles son la [API Java de DOM](#) y la [API de analizadores \(parsers\) de XML](#).

## Sesión 1: grupo magistral

En esta sesión se realiza una introducción teórica al análisis de XML desde programas y los principales tipos de analizadores existentes, con sus ventajas y desventajas. Por otra parte, se profundizará en la interfaz DOM.

## Sesión 2: práctica

La práctica consiste en escribir un programa Java de línea de comandos que trabaje con ficheros XML de películas con la estructura desarrollada en las prácticas anteriores. Según el argumento de línea de comandos con que se invoque, el programa debe hacer lo siguiente:

- Argumentos `-l fichero.xml`: lista los títulos de todas las películas contenidas en el fichero.
- Argumentos `-L fichero.xml`: lista información más detallada acerca de todas las películas contenidas en el fichero (por ejemplo, título, director y año para cada película).
- Argumentos `-a fichero`: añade una nueva película al fichero. Pide por línea de comandos los datos necesarios de la película.

Puedes basarte en el programa [Films.java](#), que proporciona la parte del código más rutinaria (análisis de argumentos de línea de comandos, carga del árbol DOM de un fichero XML y lectura de una cadena desde entrada estándar).