

Práctica 03. *Serialización en XML*

Iñigo Aguas Ardaiz
Iñaki García Munarriz
Álvaro Gil Escalada

Gestión de Sistemas de Información
7º semestre. Curso 2014-2015
Grado en Ingeniería Informática
Universidad Pública de Navarra

1. Informe relativo al ejercicio 1

Para este ejercicio, hemos decidido representar cada una de las instancias con toda la información ya que podrían darse diferentes circunstancias en las que con sólo una parte de los objetos asociados no fuera suficiente para recuperar todos los datos de la clase que estamos *serializando*. Con representar toda la información asociada nos referimos a una asociación de contenido. Esto es, en una clase como `periodista` no existe ninguna, ya que las opciones que le llegan de autoría o revisión no son inherentes al periodista. No necesitas escribir o revisar noticias para ser un periodista. En cambio, al revés no ocurre lo mismo, ya que un documento necesariamente tiene que tener un autor para existir.

Por esta razón, la clase `Journalist` será guardada con los datos que la componen. En el caso de la clase `Photographer` se incluirá, además, todas la fotos que este fotógrafo haya tomado. Las clase `Picture` únicamente guardará los datos relativos a la propia fotografía. La clase `Teletype` dispondrá de sus datos así como los del periodista que lo creó. En el caso de la clase `PrintableNews` guardaremos los datos además de las fotos a las que hace referencia y los periodista que son autores y revisores así como los premios. En el caso de `WebNews` volcaremos al archivo XML las fotos, premios y el autor además de sus datos. Por último, los periódicos incluirán su fecha así como todas las noticias que contienen. En el caso de todas las clases padre de las heredadas que aquí se presentan, al no ser instanciables tampoco serán representables en formato XML.

Para cumplir con la explicación anterior, la implementación del método dispone de un modo en el cual esto se lleva a cabo, el modo *full*. Adicionalmente, y para poder utilizarlo más tarde, se ha implementado, también, el modo *relational* con el cual, en el momento que se refiere a la información externa (por ejemplo, a las fotografías de un fotógrafo) se guarda la información sintéticamente necesaria para su correcta identificación.

2. Informe relativo al ejercicio 3

En este caso, se ha elegido implementar la opción b), por la cual se utiliza un constructor en cada una de las clases del paquete `B.Model` al que se le pasa un objeto tipo `string`. Este `string` será la representación XML del objeto que se está incluyendo. Se elige este método básicamente porque se entiende que en este momento se está incluyendo un objeto nuevo dentro del sistema y es justamente para esto para lo que JAVA dispone de los constructores.

Se debe añadir que ciertamente, por el hecho de ser constructores, que la lógica que estos contengan sea la menor posible. Esto se facilita evitando las excepciones que son relativas al fichero, por eso se debe pasar directamente el `string` donde se encuentra la información del nuevo objeto. Además, es claro que cuando el *parser* de XML no funcione bien, los objetos no se crearán por lo que no será posible asignarlos al sistema, lo que evitará problemas de coherencia de los datos.