

Práctica 01.

Sistema de Información Básico

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. Respuesta a las preguntas planteadas

1. ¿Sería posible mantener la clase `BusinessSystem` y reemplazar todo el almacenamiento de información por una base de datos? ¿Qué efectos tendría esto en el funcionamiento de los programas que hacen uso de `BusinessSystem`?

Una de las mejores características de las que se dispone en JAVA es, justamente, poder hacer cambios en la implementación de las clases sin que aquellos que las utilicen deban modificar su código para ello. Esto se consigue por medio de la interfaz, que es el único conocimiento del que dispone una persona que utiliza la clase. Esto es, por ejemplo, si cogemos el método `findJournalist()`, para usarlo deberemos introducir un ID y nos devuelve el objeto `Journalist` correspondiente, aunque no sabríamos cómo lo hace internamente.

Más a detalle, con la implementación que se presenta, se hace una búsqueda entre los objetos que guardan los datos. En realidad, para hacerlo con una Base de Datos simplemente habría que utilizar un *driver* de conexión como puede ser `jdbc`. Este se encargaría de conectar y pasar al motor de la base las consultas. Todo esto implicaría cambiar de forma total la forma en la que está implementada el método (haciendo las consultas y convirtiendo la información a objetos), pero para quien lo utiliza, sería invisible ya que seguiría introduciendo un ID y obteniendo un `Journalist`.

En conclusión, si la interfaz de la clase `BusinessSystem` está pensada de forma adecuada, el cambiar la forma de almacenamiento no producirá ningún cambio perceptible en el exterior. Por tanto, podríamos llevar a cabo de forma sencilla y rápida el cambio solicitado y seguir utilizando la clase `BusinessSystem` sin modificar más código que el de la propia clase. Añadido a todo esto, creemos, sería una implementación más rápida dada la optimización con la que están programados los motores actuales de las bases de datos frente al uso de los objetos.

2. ¿Puede predecir problemas en relación a la consistencia de la información? ¿Se solucionarían estos empleando una base de datos?

Es claro que se podrían producir problemas de inconsistencia debido a que los objetos "referencia" son aquellos que almacena el sistema. Si en un cierto momento dos usuarios modificaran su versión del objeto de forma diferente, este sería inconsistente, ya que nadie conocería los cambios hasta que se almacenasen de nuevo. Además, en concreto en este ejemplo, si se modificaran los mismos campos con diferente información, sólo perduraría la copia que se haya guardado más tarde. Todo esto hace ver que el utilizar los objetos como forma de almacén de datos no sea, quizá, lo más adecuado.

Una solución podría ser utilizar la potencia de las bases de datos para evitar esto. En primer lugar, podrían guardarse los objetos directamente en la base (si es que estamos interesados en ello) si preferimos utilizar bases de datos no relacionales. En segundo, si hacemos que las operaciones sean transaccionales, evitaremos la inconsistencia de la información, ya que utilizaremos de forma correcta las propiedades *ACID* y por tanto no podrá haber dos copias diferentes de un objeto.

En definitiva, sería una buena elección el uso de bases de datos (tanto relacionales como no) para guardar la información contenida en el sistema. Este hecho

ayudaría a evitar su inconsistencia y a conseguir un manejo más rápido y sencillo de la información gracias a las transacciones que pueden hacerse sobre ellas. aunque la complejidad de la implementación aumentaría de cara al programador.