

Programación en Capas.

Ricardo J. Vargas Del Valle

Universidad de Costa Rica, Ciencias de Computación e Informática,
San José, Costa Rica, 506
ricvargas@gmail.com

Juan P. Maltés Granados

Universidad de Costa Rica, Ciencias de Computación e Informática,
San José, Costa Rica, 506
maltes29@gmail.com

Abstract

This paper is about the last trends used in software engineering for complex program development made by object oriented programming (OOP).

In the layer model objects are divided by their functionality. There are three principal layers: Interface Layer, composed by objects that relate with the user, like application forms and screens, Business Layer made by objects that make most part of the internal work in the program, in this part resides most of the logic of the application and serves as link between the other layers; finally the Data Layer, which is made by objects that send and receive information when communicate with data bases or other external systems that work simultaneously with the main program.

There are many ways to implement this model, depending on the trends and technologies used by the project development team; its important to say that the layer model is complex, so it must be done efficiently in order to obtain a consistent final application that satisfies user needs.

Keywords: Object Oriented Programming (OOP), Capsulation, Coupling, Cohesion, Modularity, Framework.

Resumen

Este artículo trata sobre una de las últimas tendencias utilizadas en la ingeniería del software para la construcción de aplicaciones complejas desarrolladas mediante la programación orientada a objetos (POO).

En la programación en capas los objetos se dividen según su funcionalidad. Destacan tres principales: la Capa de Interfaz o Frontera, compuesta por los objetos encargados de interactuar con el usuario, como lo son los formularios e interfaces de la aplicación; por otra parte está la Capa de Lógica del Negocio o Control, en donde se encuentran los objetos que realizan la mayor parte del trabajo interno del programa, en esta etapa destaca la lógica de la aplicación así como la funcionalidad de servir de enlace entre las otras capas; por último se encuentra la Capa de Datos, integrada por los objetos que envían y obtienen información al comunicarse con bases de datos u otros sistemas de información que colaboran con el programa.

Este modelo puede ser implementado de muchas formas, dependiendo de las tendencias y tecnologías que siga el equipo de desarrollo del proyecto; es importante recalcar que el modelo de programación por capas es complejo y por lo tanto debe realizarse de forma eficiente para así obtener una aplicación final consistente que satisfaga los requerimientos del usuario.

Palabras clave: Programación Orientada a Objetos (POO), Encapsulamiento, Acoplamiento, Cohesión, Modularidad, Framework.

1 Introducción

Cuando se construye software como producto empresarial o comercial, se llevan a cabo varias técnicas de manera que el desarrollo se haga en forma ordenada y así poder asegurar un avance continuo del proyecto, un producto final de calidad, y además realizar posteriores mejoras sea una tarea más fácil.

Existen muchas prácticas de programación, dependiendo del tipo de software que se va a desarrollar y de la disciplina o disciplinas de programación que se utilicen en el desarrollo del producto. Una de las más utilizadas se llama la programación por capas, que consiste en dividir el código fuente según su funcionalidad principal.

La programación para lograr sacarle el mayor provecho a la programación por capas se necesita seguir una serie de pasos complejos los cuales primeramente deben ser definidos para cada proyecto en específico, luego deben ser revisados para asegurarse de que el modelo adoptado cumpla con las normas necesarias para que la aplicación sea del agrado del usuario, y por último debe ser implementado por el grupo de desarrollo encargado para tal fin, los cuales siguiendo el modelo propuesto obtienen una herramienta útil para facilitar la labor de programación dividiendo la aplicación en módulos y capas fáciles de pulir.

2 Características de la Programación en Capas.

La programación por capas es una técnica de ingeniería de software propia de la programación por objetos, éstos se organizan principalmente en 3 capas: la capa de presentación o frontera, la capa de lógica de negocio o control, y la capa de datos.

Siguiendo el modelo, el desarrollador se asegura avanzar en la programación del proyecto de una forma ordenada, lo cual beneficia en cuanto a reducción de costos por tiempo, debido a que se podrá avanzar de manera más segura en el desarrollo, al ser dividida la aplicación general en varios módulos y capas que pueden ser tratados de manera independiente y hasta en forma paralela.

Por otra parte, otra característica importante de recalcar es la facilidad para las actualizaciones de la aplicación. En este aspecto, la programación en capas juega un papel de suma importancia ya que sigue un estándar conocido en el ambiente de desarrollo de aplicaciones, lo cual da al programador una guía para hacer mejoras a la aplicación sin que esto sea una tarea tediosa y desgastante, siguiendo el estándar establecido para tal fin y dividiendo las tareas en partes específicas para cada capa del proyecto.

Las principales capas que siempre deben estar en este modelo son:

2.1 Capa de Presentación o Frontera:

La presentación del programa ante el usuario, debe manejar interfaces que cumplan con el objetivo principal de este componente, el cual es facilitar al usuario la interacción con la aplicación. Para esto se utilizan patrones predefinidos para cada tipo de aplicación y para cada necesidad del usuario. La interfaz debe ser amigable y fácil de utilizar, ya que el usuario final es el que se va a encargar de utilizar el sistema y de dar retroalimentación al equipo de desarrollo en caso de que haya algo que mejorar.

Las interfaces deben ser consistentes con la información que se requiere, no se deben utilizar más campos de los necesarios, así como la información requerida tiene que ser especificada de manera clara y concisa, no debe haber más que lo necesario en cada formulario y por último, las interfaces deben satisfacer los requerimientos del usuario, por lo cual no se debe excluir información solicitada por el usuario final y no se debe incluir información no solicitada por el mismo.

Dentro de la parte técnica, la capa de presentación contiene los objetos encargados de comunicar al usuario con el sistema mediante el intercambio de información, capturando y desplegando los datos necesarios para realizar alguna tarea. En esta capa los datos se procesan de manera superficial por ejemplo, para determinar la validez de su formato o para darles algún orden específico.

Esta capa se comunica únicamente con la capa de Reglas de Negocio o Control. [1]

2.2 Capa de Lógica de Negocio o Control:

Es llamada capa de reglas de negocio porque en esta se definen todas las reglas que se deben cumplir para una correcta ejecución del programa. [2]

Es aquí donde se encuentra toda la lógica del programa, así como las estructuras de datos y objetos encargados para la manipulación de los datos existentes, así como el procesamiento de la información ingresada o solicitada por el usuario en la capa de presentación.

Representa el corazón de la aplicación ya que se comunica con todas las demás capas para poder llevar a cabo las tareas. Por ejemplo, mediante la capa de presentación obtiene la información ingresada por el usuario, y despliega los resultados. Si la aplicación se comunica con otros sistemas que actúan en conjunto, lo hace mediante esta capa. También se comunica con la capa de datos para obtener información existente o ingresar nuevos datos.

Recibe los datos que ingresó el usuario del sistema mediante la capa de presentación, luego los procesa y crea objetos según lo que se necesite hacer con estos datos; esta acción se denomina *encapsulamiento*. [3]

Al encapsular los datos, el programa asegura mantener la consistencia de los mismos, así como obtener información precisa de las bases de datos e ingresar en las mismas, solamente la información necesaria, asegurando así no tener datos duplicados ni en las bases de datos, ni en los reportes solicitados por el usuario.

2.2 Capa de Datos:

Es la encargada de realizar transacciones con bases de datos y con otros sistemas para obtener o ingresar información al sistema.

El manejo de los datos debe realizarse de forma tal que haya consistencia en los mismos, de tal forma los datos que se ingresan así como los que se extraen de las bases de datos, deben ser consistentes y precisos.

Es en esta capa donde se definen las consultas a realizar en la base de datos, tanto las consultas simples como las consultas complejas para la generación de reportes más específicos.

Esta capa envía la información directamente a la capa de reglas de negocio para que sea procesada e ingresada en objetos según se necesite, esta acción se denomina *encapsulamiento*.

3 Ventajas y Desventajas

La programación en capas no es una técnica rígida que debe implementarse solamente de una forma, sino que los desarrolladores de proyectos tienen múltiples maneras de implementarla según las tecnologías y tendencias que se utilicen.

La satisfacción de los requerimientos del usuario es la base para escoger el modelo de implementación a seguir. La tendencia a utilizar el modelo de programación en capas es grande cuando se trata principalmente de aplicaciones empresariales donde se deben manejar gran cantidad de subsistemas y módulos, así como generar reportes lo suficientemente complejos como para necesitar un orden estricto a la hora de desarrollar el proyecto.

Dentro del concepto de programación en capas, existen dos términos esenciales para el mejor entendimiento de los conceptos relativos a esta metodología, es aquí donde radica la importancia de la cohesión y el acoplamiento dentro de una aplicación generada mediante este método.

Cohesión:

Este termino es utilizado para describir el comportamiento que deben tener los módulos y objetos de un sistema o subsistema, comportamiento que describe la forma en que deben trabajar los objetos y módulos entre si, con alta cohesión para que trabajando en conjunto los módulos y objetos puedan alcanzar un solo propósito de manera más eficaz y rápida.

Determina que las operaciones de un objeto deben trabajar en conjunto para alcanzar un propósito común. Es deseable que haya alta cohesión.

Acoplamiento:

Se refiere al grado de dependencia que existe entre los módulos. Este grado de dependencia debe ser considerablemente bajo ya que el trabajo se divide en módulos para que cada uno tenga un funcionamiento específico y puede ser más factible la implementación por separado de cada uno. En caso de haber alto acoplamiento entre módulos no se estaría alcanzando el principal objetivo de este modelo, el cual es dividir una tarea grande en varias pequeñas, ya que los módulos actuarían como uno solo al estar altamente acoplados entre sí y se perdería el objetivo primordial de dividir el proyecto. [3]

3.1 Ventajas

Al implementar este modelo de programación, se asegura un trabajo de forma ordenada y separada, debido a que sigue el principio de “divide y vencerás”. [3]

Cada capa está dividida según su funcionalidad cuando se quiere modificar el sistema basta con cambiar un objeto o conjunto de objetos de una capa. Esto se llama *modularidad*.

3.2 Desventajas

Cuando se implementa un modelo de programación en capas, se debe llegar a un balance entre el número de capas y subcapas que componen el programa. Este debe ser el necesario y suficiente para realizar un trabajo específico con eficiencia y ser lo más modular posible.

De lo contrario se tiene una serie de desventajas como: pérdida de eficiencia, realización de trabajo innecesario o redundante entre capas, gasto de espacio de la aplicación debido a la expansión de las capas, o bien una alta dependencia entre los objetos y capas que contradice el objetivo principal del modelo.

4. Algunas tecnologías que implementaciones la Programación en Capas.

En la actualidad, la mayoría de las programas comerciales web que se desarrollan se hacen usando dos tecnologías: la herramienta Visual Studio .Net de Microsoft y tecnologías Java como lo son Spring y Hibernate.

4.1 Microsoft Visual Studio.NET

Esta plataforma o *framework* implementa la capa de presentación usando su propia tecnología de HTML llamada ASPX, que provee una gran variedad de herramientas para crear interfaces gráficas visibles en de la mayoría de los navegadores de Internet

La capa de datos es accesada mediante herramientas conocidas como Data Set y Data Reader, que fueron creadas con el fin de poder mapear con cierta facilidad los objetos y entidades del lenguaje con las tablas que hay en las bases de datos del programa. [4]

4.2 Tecnologías Java: Spring y Hibernate.

La máquina virtual Java también ha implementado su forma de programación en capas usando dos plataformas llamadas Spring y Hibernate.

La primera es una herramienta que facilita la creación de aplicaciones Java creada con el fin de facilitar el menor acoplamiento entre los objetos, ya que el programa se construye por medio de interfaces y no clases, se puede entonces cambiar en el futuro la implementación de un objeto sin alterar la funcionalidad general del programa. [5]

La capa de presentación se implementa por mediante las herramientas JSP, Servlets y Scriptlets con las que se construyen y conectan las interfaces gráficas con la capa de lógica del negocio.

La otra herramienta Hibernate, funciona para mapear las tablas de las bases de datos con los objetos creados en Java, esta herramienta sin embargo es más especializada y eficiente que la de Microsoft debido a que es la técnica más cercana a mapear realmente los objetos con las tablas de la base de datos. [6]

5. Conclusiones

La programación en capas ha sido una de las últimas tendencias en cuanto a software comercial se refiere, es una tendencia que bien aplicada puede resultar en un desarrollo de software eficiente.

Sin embargo, no todos los equipos ni empresas desarrolladoras usan un sistema rígido, ni existe una forma estricta en la que tenga que implementarse el modelo de capas, cada quién debe hacerlo según sus necesidades, alcances y lo más importante; las tendencias y nuevas tecnologías que vayan surgiendo.

Es importante tener en cuenta que no importa el lenguaje de programación usado, o el tipo de implementación que se le dé al modelo; se debe buscar una alta cohesión y un bajo acoplamiento dentro de los objetos y capas para lograr que la aplicación sea fácilmente desarmable y sea más sencillo realizar mejoras y actualizaciones al sistema.

References

- [1] http://es.wikipedia.org/wiki/Programaci%C3%B3n_por_capas, 1 Junio 2007.
- [2] <http://phpsenior.blogspot.com/2006/07/conceptos-separar-el-codigo-de-la-capa.html>, 2006.
- [3] Ingeniería del Software, Un enfoque práctico. Sexta Edición. Roger S Presuman. 2005.
- [4] http://www.informatizate.net/articulos/el_acceso_a_datos_en_asp_net_whidbey_parte_02_20050219.html
- [5] <http://sentidoweb.com/2006/12/26/spring-framework-de-java.php>, 26 diciembre 2006.
- [6] http://www.programacion.com/java/articulo/jap_persis_hib, 2007.