Miguel tejada 8-868-1025

Programación orientada a objeto Python

Este artículo es una introducción a la programación orientada a objetos (POO, o OOP por sus siglas en inglés), uno de los paradigmas de programación estructurada más importante hoy en día.

El artículo usa el lenguaje [Python](http://www.python.org/) para los distintos ejemplos, y explica la implementación de la POO en ese lenguaje, aunque aborda conceptos generales que son fácilmente trasladables a otros lenguajes.

Los fragmentos de código se señalan usando un estilo de texto diferente (ancho fijo) y, cuando se utiliza el intérprete de python, se prefija cada línea con los símbolos ‘>>>’.

Esta introducción presupone conocimientos básicos de programación y del lenguaje python Es muy recomendable y productivo comprender el enfoque de distintos paradigmas, puesto que presentan estrategias alternativas, incrementando nuestras herramientas disponibles y haciéndono reflexionar sobre muchas de las tareas que realizamos al crear un programa. Además, a menudo ocurre que unos problemas se formulan de forma muy clara si se los analiza según una perspectiva determinada, mientras que producen una gran complejidad vistos de otra manera

Dirigida a eventos

La programación dirigida por eventos es un paradigma de programación en el que el flujo del programa está determinado por eventos o mensajes desde otros programas o hilos de ejecución.Las aplicaciones desarrolladas con programación dirigida por eventosimplementan un bucle principal o main loop donde se ejecutan las dos secciones principales de la aplicación: El selector de eventos y el manejador de eventos La mayoría de librerías para el desarrollo de aplicaciones con GUI como[GTK](http://www.gtk.org/) o [Qt](http://qt.nokia.com/) están diseñados para ser dirigidos por eventos, de ahí el famoso sistema de ranuras y señales de Qt que no es más que un patrón de diseño observer bastante currado.

Un poco de historiaA finales de los 70, los sistemas estaban pensados para trabajar como cadenas de ensamblaje donde un programa usaba una entrada y producía una salida que sería utilizada por otro programa como entrada para producir otra salida y así hasta finalizar el proceso. Este proceso mental de construir software es la base del desarrollo estructurado.El padre del desarrollo estructurado (que no de la programación estructurada ojo) fue Larry LeRoy Constantine bajo el ala del Instituto de Investigación de Sistemas de IBM. Uno de los mayores expertos y defensores de los métodos estructurados es Edward Yourdon, tanto que las expresiones “Yourdon” y “métodos de análisis y diseño estructurado” son sinónimos.

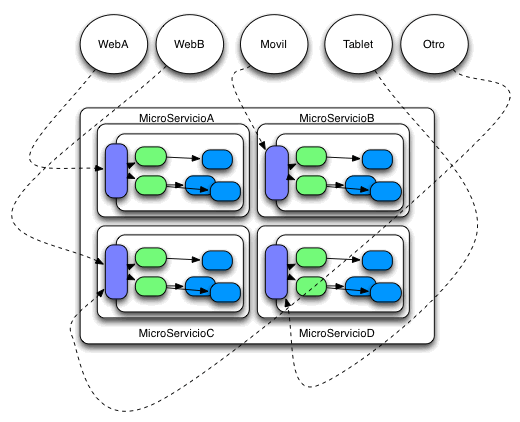
Constantine y Yourdon definieron nuevos modelos de control del flujo de datos implementando lo que llamaron transacciones que en realidad son un patrón de diseño de manejadores de evento

Orientada en espacio

La formación mediada por tecnología a través de espacios virtuales de trabajo colaborativo surge como una posibilidad de permitir la interacción de equipos de trabajo orientados a la realización de proyectos de desarrollo de software en asignaturas de grado donde sus integrantes no se encuentran en espacio físicos comunes. Las tecnologías de la información y la comunicación han permitido construir comunidades virtuales integradas donde surgen modelos colaborativos de trabajo. En este contexto, el trabajo formula una aproximación a la identificación de los elementos del espacio de trabajo y evalúa la viabilidad tecnológica de la virtualización de los elementos de dicho espacio.

Microservicio

 ser cada microservicio independiente tenemos varias ventajas. Primero es más fácil para los desarrolladores hacerse con ellos ya que son relativamente “pequeños” y fáciles de abordar. Segundo si un microservicio falla no afecta a los demás ya que son totalmente independientes. Por otro lado facilita el despliegue ya que no tenemos que desglegar aplicaciones gigantescas sino “microservicios” de tal forma que si solo tenemos que actualizar uno de ellos pues solo redesplegamos este. Por otro lado y una de las cosas más importantes podemos acceder a estos microservicios desde todo tipo de dispositivo ya que los tenemos publicados vía REST o algún tipo de RPC.

[](http://www.arquitecturajava.com/wp-content/uploads/008.gif)