

Bases de Datos Distribuidas y Teorema de CAP

Pablo Fonseca Moncada

pfonsecam@ucenfotec.ac.cr

Cenfotec Learning Center S.A

Bachillerato de Ingeniería de Software

Abstract

En este documento encontrará una explicación breve sobre los dos mejores IDE para programar en C++, dicha explicación contiene sus ventajas, sus desventajas y características adicionales, así como una comparación de ambos IDE al final. Es importante recalcar que los dos IDE seleccionados se consideran los mejores, y para obtener esta información se hizo una tabla que se fue llenando con datos obtenidos según las opiniones de diferentes páginas web. Los mejores IDE para programar según las votaciones son “Visual Studio Code” y “Eclipse”, sin embargo, bien se sabe que “Visual Studio Code” técnicamente no es un IDE, por lo que se elegirá otro de los mejores IDE, que no es tan votado como “Visual Studio Code” pero que se recomienda en 4 de las 7 páginas web analizadas, en este caso “NetBeans”. Al final de lo investigado, se dará un detalle del mejor de ambos IDE, eligiendo así uno de ellos como el mejor de todos.

Palabras Clave

IDE, C++, C, Programación, Software, Compilador, Compilación, Aplicaciones.

Introducción

Desde que el ser humano las creó iniciando con bolas y palitos de madera, han sido herramientas imprescindibles en nuestras vidas. Las computadoras son dispositivos electrónicos con grandes capacidades, podemos utilizarlas para automatizar nuestras tareas diarias y también para concluir nuestros quehaceres mediante el uso de programas digitales. Las computadoras son demasiado potentes, incluso lo son

aquellas que hoy en día tenemos abandonadas en el sótano por sus pocas capacidades electrónicas. Es normal que para un usuario común no exista una necesidad superior que usar un programa para trabajar, pero si hablamos de capacidad las computadoras pueden hacer mucho más que eso. Un usuario puede darle instrucciones a una computadora para realizar un conjunto de acciones específicas, provocando así la creación de una nueva aplicación o de un sistema. Esta tarea no es común, la mayoría de las veces la realizan profesionales dedicados a la computación con el fin de resolver un conjunto de problemas específicos. Es normal que dicha solución sea solicitada por una compañía de Software que necesita mejores herramientas para poder progresar.

El resolver este conjunto de problemas requiere que el programador escriba código fuente en un lenguaje específico, C++ es uno de ellos. Como C++ es uno de los lenguajes más complicados y poderosos que existen, se debe de conocer una buena herramienta para su programación, dicha herramienta por lo habitual se le denomina IDE, que en inglés significa Entorno de Desarrollo Integrado. En este escrito se detallará la información específica de 2 de los IDE más recomendados para programar en C++. Se mostrarán sus ventajas, sus

funcionalidades técnicas y sus desventajas.

Información sobre Eclipse

Eclipse es un IDE bastante poderoso, el cual tiene las siguientes características:

Funciona por medio de Plugins

Eclipse trabaja por conjuntos de Plugins, lo que le da la facilidad al programador de poder editar las funcionalidades y las capacidades de este de una manera más simple, en otras palabras, como el IDE trabaja según los módulos que se le configuran un programador puede elegir fácilmente si trabajar con lo mínimo del IDE o si quiere convertirlo en una súper herramienta con miles de funcionalidades. Esto es una gran ventaja, ya que existen múltiples IDE que le dan al programador todas las características y herramientas, sin la opción de quitar las que no ocupe.

Es un IDE de propósito general

Eclipse en realidad es un IDE que no se diseñó para un lenguaje específico, por lo que tiene varios tipos de versiones con propósitos específicos, pero en sí, todos en conjunto lo que intentan proveer es un entorno general para diferentes tipos de tecnologías, por lo que, si el programador trabaja en C++ o usa Java, podría utilizar "Eclipse IDE for C/C++ Developers" o bien "Eclipse IDE for Java Developers".

Filosofía basada en proyectos

Eclipse trabaja basado en proyectos, lo que significa que cada diseño realizado en él crea un paquete al que se le llama proyecto. La filosofía basada en proyectos tiene la ventaja de que ya trae varias configuraciones, permitiéndole al programador enfocarse en la lógica y no

tanto en la configuración compleja del diseño, como lo son las configuraciones específicas para los procesos de compilación y ejecución. Claramente esto trae una serie de precauciones, ya que no cualquier programador debería de utilizar una herramienta tan poderosa y también, porque no a todos los programadores les sirve que haya configuraciones por defecto desde que va a iniciar su proyecto.

Compilador incorporado

Uno de los procesos más complicados que se deben de realizar a la hora de programar en C++ es configurar un Debugger para depurar nuestro código cuando haya errores, en el caso de Eclipse nos podríamos olvidar de esto, ya que trae listo un depurador de código fuente, el cual solo se ejecuta iniciando en un botón y marcando un punto de parada o también llamado breakpoint.

Multiplataforma

Una de características más importantes de Eclipse es que es multiplataforma, esto quiere decir que el ambiente de desarrollo puede ser instalado en diferentes sistemas operativos: Puede usarse en Darwin, Linux y también Windows.

Autocompletado de Código

Eclipse trae consigo un conjunto de configuraciones por defecto y algunos paquetes instalados según la versión Eclipse que uno instala. En el caso de "Eclipse for C/C++ Developers" el ambiente integrado trae configuraciones para autocompletado. Por lo que es mucho más simple poder codificar, ya que tenemos una vista completa de los

métodos, objetos, variables estáticas y propiedades que podemos utilizar. Su implementación permite la realización de grandes operaciones. Como una gran parte del código fuente de Eclipse es escrita en lenguajes cercanos al lenguaje máquina y también, como eclipse permite el uso de módulos de terceros desarrollados en lenguajes muy potentes, da la capacidad de poder ejecutar dentro de él grandes funcionalidades, como por ejemplo una aplicación de red como Telnet o bien un sistema de gestión de base de datos.

Trabaja por medio de Wizards

Eclipse utiliza un conjunto de wizards o aplicaciones gráficas de ayuda para poder inicializar fácilmente los proyectos.

Compilación en tiempo real

Eclipse tiene la ventaja de tener JIT (Just in Time Compilation) y en las palabras más sencillas eso significa que el código Bytecode de Java puede ser interpretado en tiempo real a código nativo de la máquina, esta característica es especialmente popular en lenguajes que trabajan con Bytecode, como por ejemplo Java y C#. Just in Time Compilation también es conocida como Dynamic Compilation, en español Compilación Dinámica. Las ventajas de JIT se encuentran en el hecho que, desde que la compilación toma lugar en tiempo de ejecución, el compilador JIT tiene acceso a la información de rutinas dinámicas habilitándolas para realizar mejores optimizaciones (por ejemplo, convirtiendo las funciones en funciones en línea). Lo que es importante de entender acerca de la compilación JIT es que compilará el Bytecode en

instrucciones de código máquina según el tipo de arquitectura que tenga nuestra computadora (ya que existen diferentes tipos de Assembly). Por último y no menos importante, una de las funcionalidades más asombrosas de JIT es que no compila todo el código Bytecode, solamente compilará aquellos fragmentos de código fuente que se cambian constantemente, dichos fragmentos se llaman Cached Bytecode.

Desventajas

Su principal inconveniente es que a como es de poderoso consume una gran cantidad de recursos de la computadora. Sin embargo, esto es habitual, y todo usuario experimentado en el campo computacional debería de saberlo, si hay un buen software debe haber un precio.

Información sobre NetBeans

Uno de los IDE más completos diseñados por la compañía de Oracle con el fin de poder tener una herramienta para programar en el lenguaje Java, se convirtió en uno de los entornos más poderosos para una gran variedad de desarrolladores de software. Esta gran herramienta tiene las siguientes características:

Orientado especialmente a un lenguaje

Este IDE tiene la característica de que es muy útil para programar en el lenguaje de programación Java, ya que fue diseñado pensando en ello, sin embargo, su poder no decrementa si es escogido para la utilización de otra tecnología.

Soporte excepcional

NetBeans se considera uno de los primeros IDE para programar en Java, por lo que es habitual que tenga soporte incluso para las versiones más recientes de Java JDK.

Funciona por medio de Plugins

Al igual que otros ambientes, NetBeans funciona por medio de Plugins, no solo tiene Plugins propios del entorno de desarrollo, sino que también tiene una gran librería donde diariamente se van agregando Plugins de terceros, lo que da unas grandes capacidades de modificación.

Rápido desarrollo de aplicaciones

Este IDE tiene la característica de que contiene RAD (Rapid Application Development), esto se utiliza principalmente en Java, para crear formularios complejos con Layouts en tecnologías de Java para interfaz gráfica, por ejemplo, Java Swing. El desarrollo RAD no es aconsejable para principiantes, de hecho, un programador avanzado no lo usaría si conoce bien Java Swing. Sin embargo, desarrollar las vistas en RAD ciertamente es una ventaja, lo que hace que se incremente la productividad notablemente.

IDE Multiplataforma

Al igual que otros ambientes, NetBeans es muy conocido por su gran flexibilidad para poder ser instalado y ejecutado en Windows, Darwin y Linux.

Soporta varios lenguajes

Sin duda alguna, NetBeans es un IDE que fue pensado para programar en Java, sin embargo, su arquitectura se modificó para

que pudiera ser en realidad un IDE de propósito general y así tener diferentes características según la versión que se instale. Esto significa que NetBeans es un IDE que funciona de una manera muy potente dependiendo del lenguaje que utilicemos, simplemente tendremos que instalar en nuestra máquina la versión específica para el lenguaje que queremos utilizar.

Desventajas de NetBeans

Como todo IDE, NetBeans también tiene sus defectos, los cuales son bastante considerables a criterio del autor de este escrito:

- Poca existencia de Plugins.
- Hace falta documentación del Rich Client Platform (RPC).
- No posee un editor de código HTML.

Conclusión

Es impresionante como el mundo tecnológico va cambiando constantemente, cada segundo que pasa en el mundo va creciendo una bola de nieve hacia la innovación y también todas aquellas creaciones recientes van pasando velozmente al pasado. Sin duda alguna para poder innovar a tal velocidad, requerimos de las herramientas correctas y como una gran parte de las invenciones de esta era se encuentra en el material digital, igual debemos disponer de las mejores herramientas digitales.

Luego de las investigaciones realizadas, se pudo llegar a la cima del entendimiento de las características de mejores ambientes de desarrollo integrado, Eclipse y NetBeans. Gracias a horas de análisis, se pudo cumplir con los objetivos esperados, y se mencionaron las

características de cada IDE, que también vienen a ser parte de sus ventajas, sus desventajas al igual que también se dieron algunas explicaciones más profundas de conceptos importantes, como por ejemplo la compilación dinámica.

cofundadores de compañías de Costa Rica como Grupo Cinética C.R en el área de desarrollo de Software.

Referencias

GeeksforGeeks. (2020, August 25). Top 5 IDEs for C++ That You Should Try Once.

<https://www.geeksforgeeks.org/top-5-ides-for-c-that-you-should-try-once/>

GeeksforGeeks. (2021, March 18). 10 Best IDEs for C or C++ Developers in 2021.

<https://www.geeksforgeeks.org/10-best-ides-for-c-or-cpp-developers-in-2021/>

K. (2021a, March 9). 27 Best C & C++ IDE & Editors In 2021 (With Infographics). Web Development & Technology Resources.

<https://codecondo.com/top-10-ide-for-c-and-cplusplus-for-programmers/>

K. (2021b, April 30). Top IDEs for C or C++ Developers in 2021 & Beyond! Eduonix Blog.

<https://blog.eduonix.com/software-development/top-10-ides-c-c-developers/>

L., J., C., O., Rodriguez, D., J., Manivannan, K., Points, V. S., & Abdal-Aziz, Y. (2021c, February 4). Best C++ IDEs or Source Code Editors for Programming in 2021 [Updated]. Hackr.io.

<https://hackr.io/blog/cpp-ide>

Rungta, K. (2021, March 25). 15 Best C++ IDE: Free Windows Editor | Compiler | 2021. Guru99.

<https://www.guru99.com/best-cpp-ide-editor-free.html>

Welch, S. (2020, May 19). Best C++ IDEs. Udacity.

<https://www.udacity.com/blog/2020/05/best-c-ides.html>

Pablo Fonseca Moncada es un estudiante de la universidad Cenfotec, que actualmente está cursando la carrera de bachillerato de ingeniería de Software, ha trabajado un año con