

# Unidad 9

Entornos de Desarrollo

# Contenido

- Introducción
- Entornos de Desarrollo
- Propiedades de un Entorno

# Introducción

- En muchas ocasiones se usan los términos de Entornos de Programación y Entorno de Desarrollo como sinónimos
- Sin embargo hay una diferencia entre estos dos conceptos

# Entorno de Programación

- Un Entorno de Programación es una aplicación que facilita las tareas relacionadas con la codificación

# Entorno de Desarrollo

- Un Entorno de Desarrollo es aquel que facilita o automatiza varias de las tareas involucradas en el desarrollo de software:
  - Diseño
  - Codificación
  - Configuración
  - Pruebas
  - Instalación
  - Documentación

# Herramientas CASE

- Siglas de *Computer-Aided Software Engineering* (Ingeniería de Software Asistida por Computadora)
- Se denominan herramientas CASE a los productos que apoyan en el desarrollo, en particular a las tareas de análisis: diseño, codificación, pruebas y ejecución.
- Pueden auxiliar en actividades como la documentación o la configuración

# Productos CASE

- Entre la gran variedad de productos CASE que se pueden encontrar se tienen:
- Herramientas de Análisis y Diseño
- Editores de Código
- Herramientas de Documentación
- Depuradores

# Clasificación de Productos CASE

- Los Productos CASE se pueden clasificar en base a las funciones que realizan:
  - Servicio. Aquellos que realizan una determinada operación, por ejemplo la compilación
  - Herramientas. Ofrecen soporte a una tarea determinada, por ejemplo el código fuente
  - *Workbench*. Soporte a las actividades de diversos perfiles de usuarios, por ejemplo herramientas de diseño
  - Entorno. Ofrece soporte a todo el proceso de desarrollo

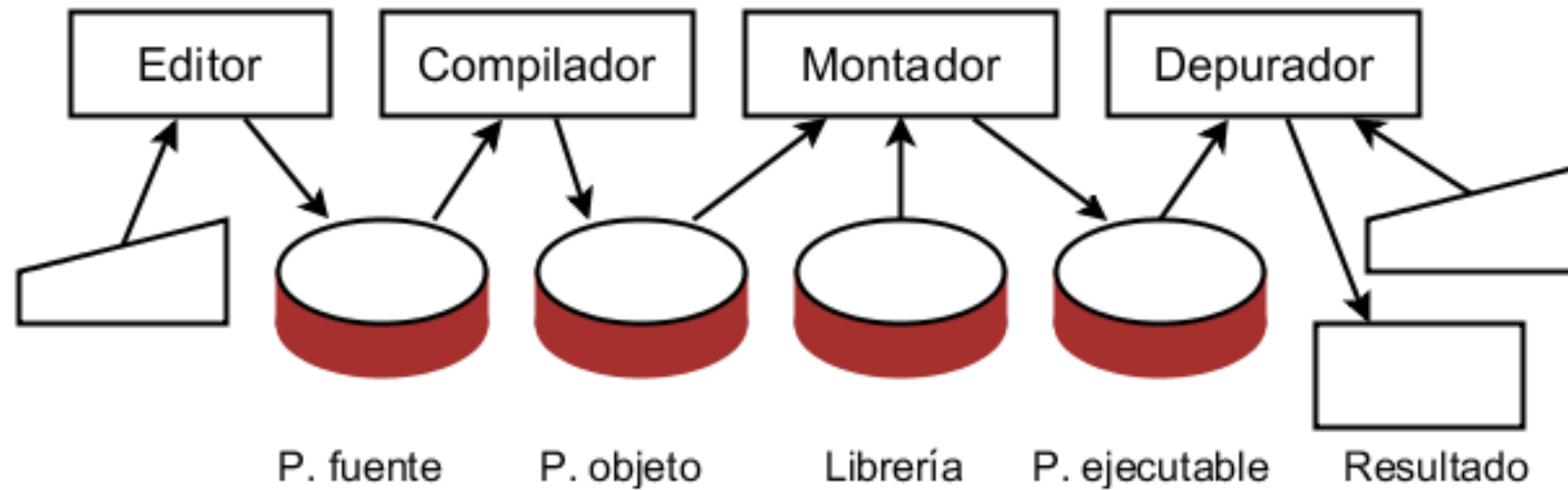


# Entornos de Desarrollo

# Entornos de Desarrollo

- Las dos actividades que más se manejan en los Entornos de Desarrollo son la codificación y las pruebas
- A las herramientas que soportan estas actividades se les conoce como IDE (*Integrated Development Environment*), aunque no forman parte un entorno de desarrollo, sino una parte de él

# Elementos Comunes



# Funciones de un IDE

- Las funciones principales que debe soportar un IDE son:
- Creación del código fuente
- Modificación del código fuente
- Ejecución del programa
  - Interpretación del código
  - Compilación a código máquina
  - Compilación a código interpretado

# Otras Funciones

- Otras de las funciones que soporta son:
  - Depuración
  - Control de versiones
  - Generación de documentación

# Origen de los Entornos de Programación

- Los primeros entornos de programación estaban formados por una cadena de operaciones que se invocaban de manera manual, por lo que no se consideraban como tal un entorno de programación
- Las herramientas que se utilizaban eran:
  - Un editor de texto sencillo
  - Un compilador que generaba un código objeto
  - Un cargador que combinaba varios registros objetos
  - Un depurador que ayudaba a la revisión del funcionamiento

# La Evolución al IDE

- Un Editor está orientado al lenguaje de programación, resaltando sintaxis especial del código fuente
- El Depurador ya presenta información no solo en código máquina, sino también en lenguaje simbólico
- La integración entre Editor y otras herramientas es muy completa, por ejemplo resaltando errores de compilación
- Los IDE se pueden clasificar en:
  - Centrados en un Lenguaje
  - Orientados a Estructura
  - Colección de Herramientas
  - Entornos para Ingeniería de Software

# Centrados en Lenguaje

- Sus características generales son:
- Se utilizan para un lenguaje de programación en específico
- El editor está muy enfocado al lenguaje
- Son sencillos de utilizar
- No es sencillo que interactúen con otros productos o funciones
- Se pueden encontrar entornos para los siguientes lenguajes:
  - LISP, Haskell – Inter
  - Java – BlueJ
  - C# - Visual Studio .NET



# Orientados a Estructura

- También se pueden considerar como específicos para lenguajes de programación
- El editor no es de texto, sino sintáctico
- Representan el código fuente como una estructura
- Se utilizan principalmente para desarrollo personal, no en equipo
- Actualmente este tipo de entornos de desarrollo están desactualizados

# Colección de Herramientas

- Son un conjunto de herramientas que trabajan entre sí
- Entre sus características se encuentran:
- Son sencillos de ampliar o adaptar nuevas herramientas
- La parte de interacción con el usuario puede ser configurable e invocar a herramientas externas
- Algunos de estos editores incluyen:
  - Emacs
  - Vim
  - Gvim
  - Eclipse

# Entornos para Ingeniería de Software

- Son entornos que apoyan a las actividades relacionadas con el desarrollo de software, en especial el trabajo colaborativo a partir de entornos de integración
- También ofrecen herramientas para la etapa de diseño ya que cuentan con herramientas para el modelado y el análisis y el diseño
- Ofrecen herramientas de gestión de proyectos

# Propiedades de un Entorno

# Facilidades Ofrecidas

- Con el paso del tiempo, los entornos han incrementado su orientación hacia los lenguajes ofreciendo las siguientes características:
  - Resaltado de sintaxis
  - Compresión o expansión del código fuente
  - Acceso directo a elementos relacionados
  - Vistas múltiples
  - Opciones de auto completado y generación automática de código
  - Formato de estilo

# Generación de Documentación

- Es común, y muchas veces deseable, que se incluyan comentarios en el código fuente
- Varios entornos de desarrollo ofrecen funcionalidad para generar documentación a partir del código fuente y los comentarios

# Análisis Estático

- Se realizan comprobaciones al momento de ir escribiendo el código fuente:
  - Errores de compilación
  - Errores de dependencia
  - Código inalcanzable
  - Comprobación de Estilo