

# Seminario de Herramientas

- # Instalación y uso básico de una herramienta UML: BOUML
- # Otras herramientas del laboratorio



Laura Barros  
Computadores y Tiempo Real



## Características

---

- # Herramienta UML 2.0
- # Software libre. Puede ser redistribuida y/o modificada bajo los términos de la Licencia Pública General (*General Public License, GNU*).
- # Permite especificar y generar código C++, Java, IDL, PHP y Python.
- # Multiplataforma: Linux, Windows, MacOS
- # Es rápida y no consume mucha memoria
  
- # Instalación : Descargar el programa: <http://bouml.free.fr/download.html>
  - Windows: Bouml\_4.15\_setup.exe
    - Ejecutar el Setup y seguir las instrucciones.
  - Linux: existen paquetes para las distribuciones principales.
- # Manual de usuario: <http://bouml.free.fr/doc/index.html>



## Crear un proyecto nuevo

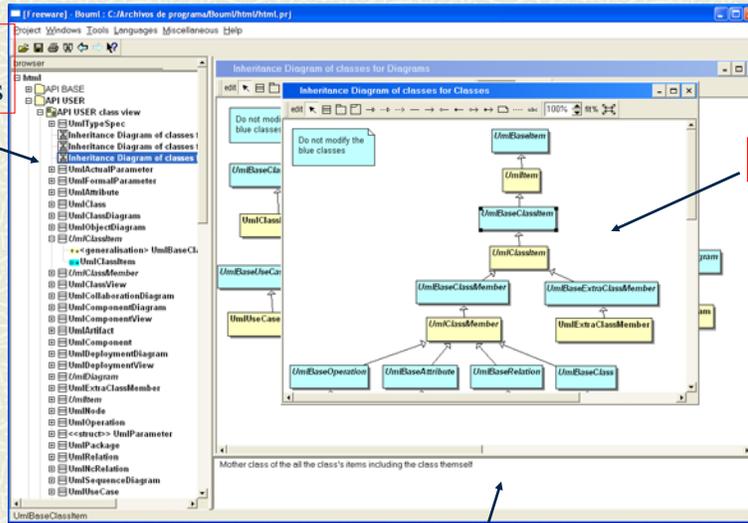
---

- # Al iniciar BOUML nos pide un identificador del desarrollador (BOUML\_ID): si varios desarrolladores van a trabajar con el mismo modelo, se debe especificar esta variable de entorno
  - Windows: start > settings > control panel > system > advanced > environmentvariables.
  
- # Crear el proyecto: *Project => New*
- # Guardar el proyecto: se indica el directorio donde se guardará el proyecto.
  - Nota: la carpeta debe llamarse como el fichero *.prj*
- # Indicar lenguaje de desarrollo:
  - Pestaña *Languages=> Java default definition*



## Pantalla principal

Elementos estructurales



Diagramas

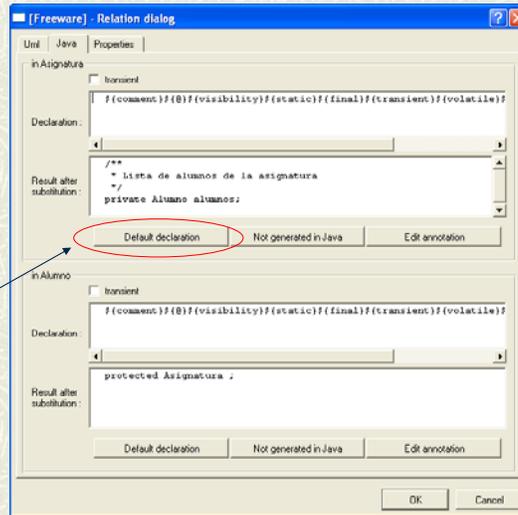
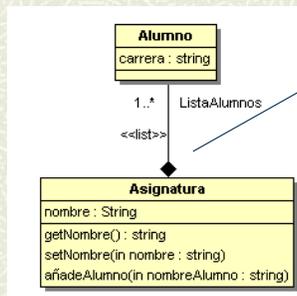


## Ejemplo de uso: Creación de un diagrama de clases

- # Generar una vista de clases:
  - *Botón derecho en el proyecto=>New class view*
- # Generar diagrama de clases:
  - *Botón derecho en la vista de clases => New class diagram*
- # Cambiar las opciones del dibujo (muestra de atributos, signatura completa de operaciones, etc):
  - *Botón derecho en el diagrama de clases => Edit drawing Settings*
- # Crear clases y asociaciones:
  - *Elegir el icono correspondiente en la paleta y pinchar en el área del diagrama*
- # Editar propiedades de las clases (añadir operaciones, atributos, etc):
  - *Doble click sobre la clase en el diagrama o en el modelo*

## Modificación del modo de generación de código I

- Para ver como se va a generar el código de una clase o asociación: Botón derecho en una clase o asociación => Edit => Java





## Generación de código

- # Generamos una vista de desarrollo:
  - *Botón derecho en el proyecto => New Deployment View*
- # La asociamos a la vista de clases:
  - *Botón derecho sobre la vista de clases => Edit => deployment view*
- # Creamos un artefacto (.java) para cada clase:
  - *Botón derecho en la clase => Create source artifact*
- # Elegimos el directorio en que se va a generar:
  - *Botón derecho en el proyecto => Edit generation settings => Directory*
- # Generamos el código :
  - *Botón derecho en el proyecto => Generate => Java*
  - Aparece una ventana que informa de los errores y warnings encontrados*



### # 1. Java Runtime Environment (JRE):

- Descripción : Entorno de ejecución de Java.
- Descargar el programa:  
<http://java.sun.com/javase/downloads/index.jsp>
- Windows: Java Runtime Environment (JRE) 6 update 2

### # 2. Eclipse IDE for Java Developers

- Descripción: Framework para el desarrollo de software escrito en Java.
- Descargar el programa:  
<http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/ganymede/R/eclipse-jee-ganymede-win32.zip>
- Windows: eclipse-jee-ganymede-win32.zip



## Intercambio de Archivos con Eclipse

---

Estos son los pasos que se deben seguir para exportar/importar un proyecto que hayamos desarrollado en Eclipse:

- # Exportar el proyecto: en el menú Package Explorer:
  - *Export=>General=>Archive File=>To Archive File (.zip)*
  
- # Importar el proyecto:
  - *Import=>Existing Projects into Workspace=>Select archive file*