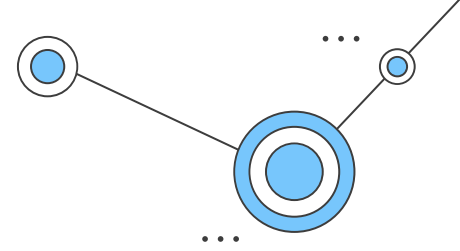


# Tutorial AspectJ

Kevin Leonardo Alvarez Mora  
Jose Francisco Lugo Nomesque  
Andres Yohany Velasquez Martinez  
Valentina Viafara Esteban

# Tabla de contenido



1. ¿Qué es?  
...

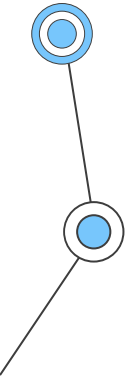
2. Instalacion de AspectJ  
...

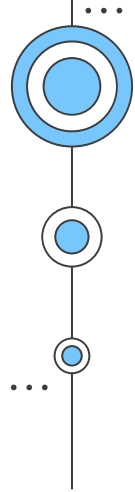
3. Conceptos básicos  
...

4. Ventajas y  
desventajas  
...

5. Características  
...

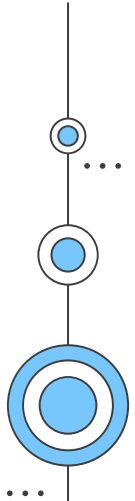
6. Ejemplos  
...





# 01

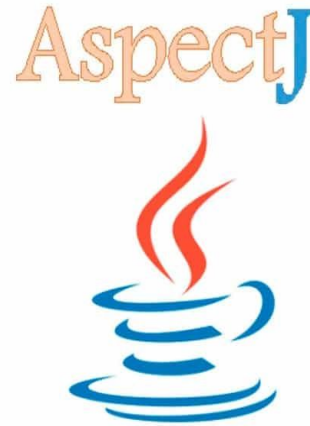
## ¿Qué es?



# AspectJ

AspectJ es una extensión de programación orientada a aspectos para Java, diseñada con la máxima compatibilidad posible en mente.

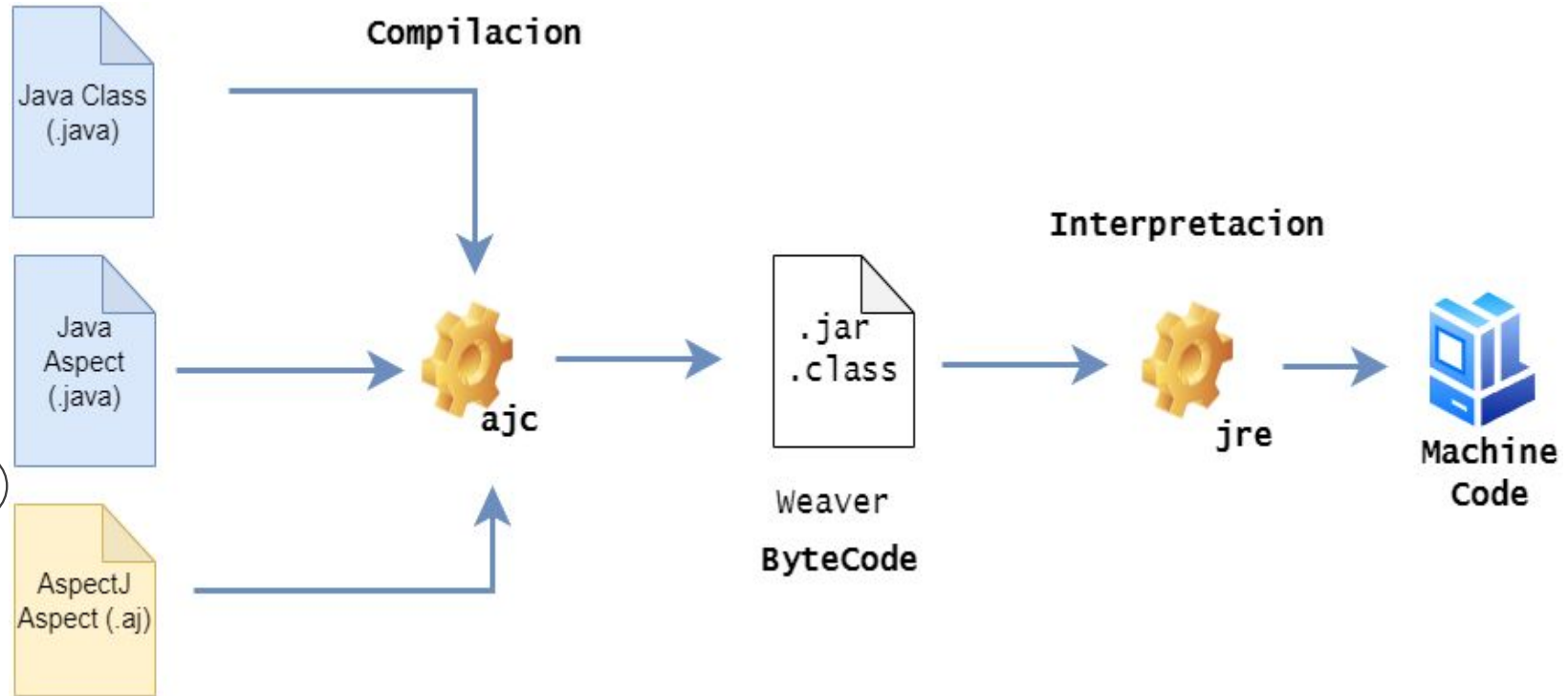
- Desarrollado por Xerox Parc (Palo Alto Research Center)
- Fue liberado al público en 1998, con su primera versión estable en el 2001
- Busca seguir la misma filosofía de Java (Write once, Run anywhere)
- Cuenta con su propio compilador /Weaver (AJC)



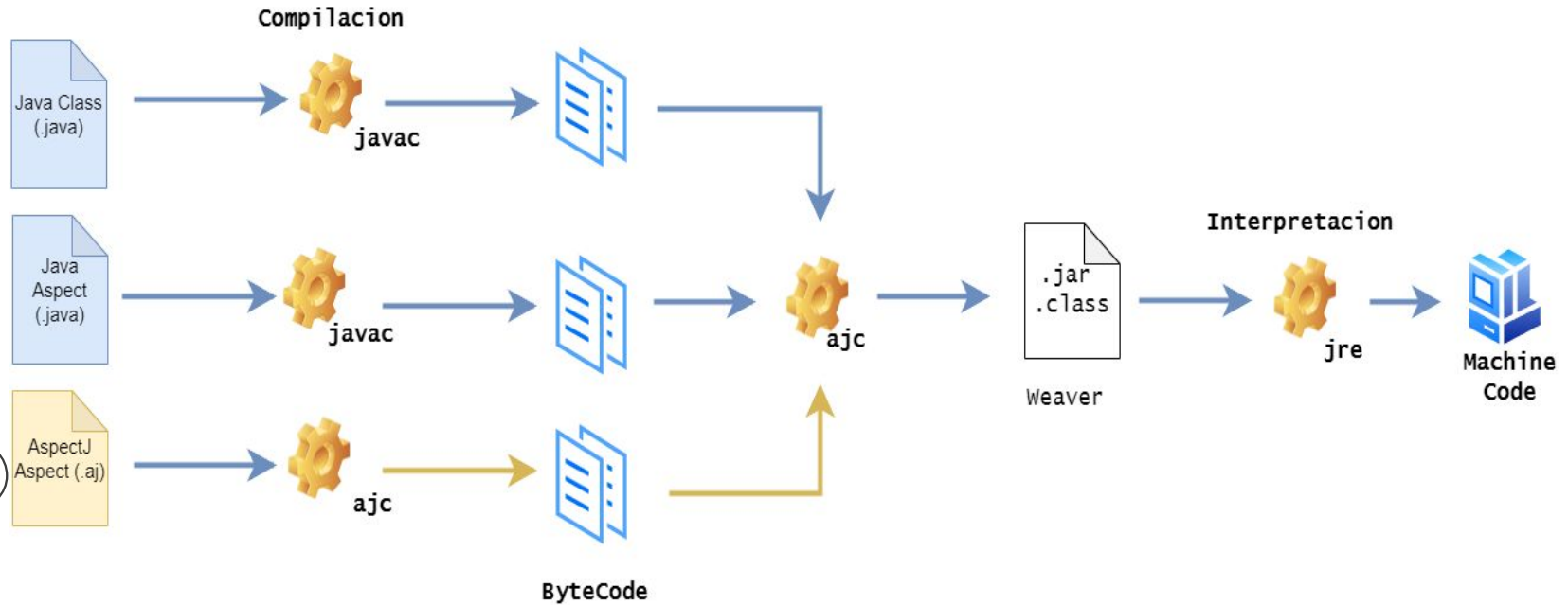
# AspectJ – Filosofía

- Compatibilidad Base: Todos los programas válidos de Java deben ser programas válidos de AspectJ
- Compatibilidad de plataforma: Todos los programas válidos de aspectJ deben correr en la maquina virtual estándar de Java
- Compatibilidad con el programador: Programar en AspectJ debe sentirse tan natural como programar en Java
- Compatibilidad con herramientas: debe ser posible mejorar herramientas para soportar AspectJ de manera natural (IDE's, herramientas de documentación y diseño)

# AspectJ - ajc



# AspectJ - ajc



# AspectJ - Usos

Frameworks



Monitoreo



Glassbox

Servidores



spring<sup>®</sup>  
source

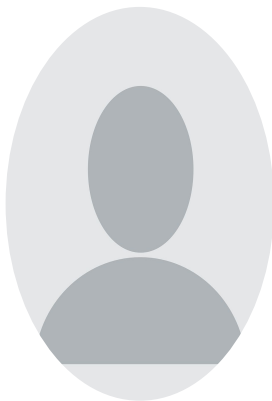


# AspectJ - Autores



Gregor Kiczales

Fundador y líder del equipo PARC que creó AspectJ, creador del término crosscutting



Chris Maeda

Creador del término AOP (Aspect Oriented Programming)  
Encargado de desarrollar prototipos funcionales.



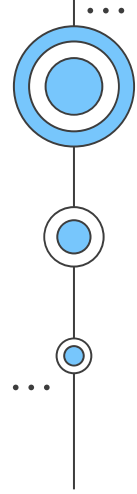
Jim Hugunin

Creador de Numeric (Predecesor de numpy) y Jython.  
Encargado de desarrollar el compilador y el Weaver de aspectJ



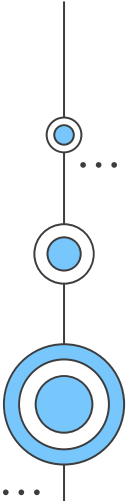
Mik Kersten

Desarrollador de las integraciones para IDE e inició el proyecto AJDT (AspectJ Development Tools)



# 02

## Instalación



# AspectJ - Instalacion

**Prerrequisitos:** Para el correcto funcionamiento es necesario contar con los siguientes productos instalados:

1. Java JDK y JRE previamente instalados (Últimas versiones recomendadas)
2. JDK y JRE Previamente asignadas en PATH
3. Un IDE con Plugins Compatibles (Se recomienda Netbeans, Eclipse O IntelliJ Community Edition)

# AspectJ - Instalacion

1. Dirigirse al sitio web: [https://www.eclipse.org/aspectj/downloads.php#most\\_recent](https://www.eclipse.org/aspectj/downloads.php#most_recent)  
Y ubicarse en la sección Latest Stable Release, de ahí descargar la última versión.

## Latest Stable Release

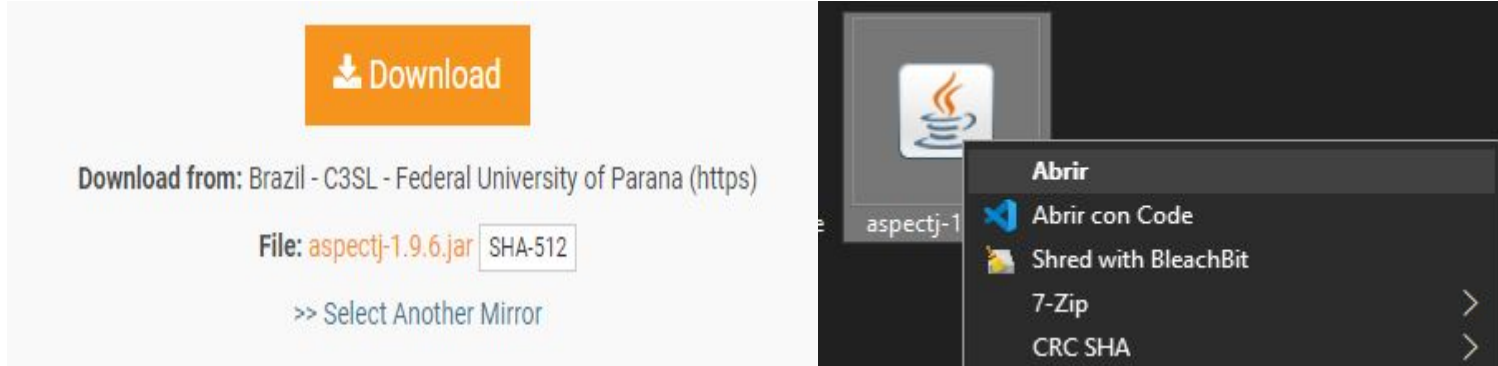
From 1.9.7 onwards, AspectJ releases are now available from GitHub.  
See the releases page here.

## AspectJ 9

- **AspectJ 1.9.6, Released 22 Jul 2020**  
AspectJ compiler, browser, documentation tool, Ant tasks, and documentation. [aspectj-1.9.6.jar](#) (~17M)  
[aspectj-1.9.6-src.jar](#)
- **AspectJ 1.9.5, Released 28 Nov 2019**  
AspectJ compiler, browser, documentation tool, Ant tasks, and documentation. [aspectj-1.9.5.jar](#) (~17M)  
[aspectj-1.9.5-src.jar](#)
- **AspectJ 1.9.4, Released 10 May 2019**  
AspectJ compiler, browser, documentation tool, Ant tasks, and documentation. [aspectj-1.9.4.jar](#) (~17M)  
[aspectj-1.9.4-src.jar](#)

# AspectJ - Instalacion

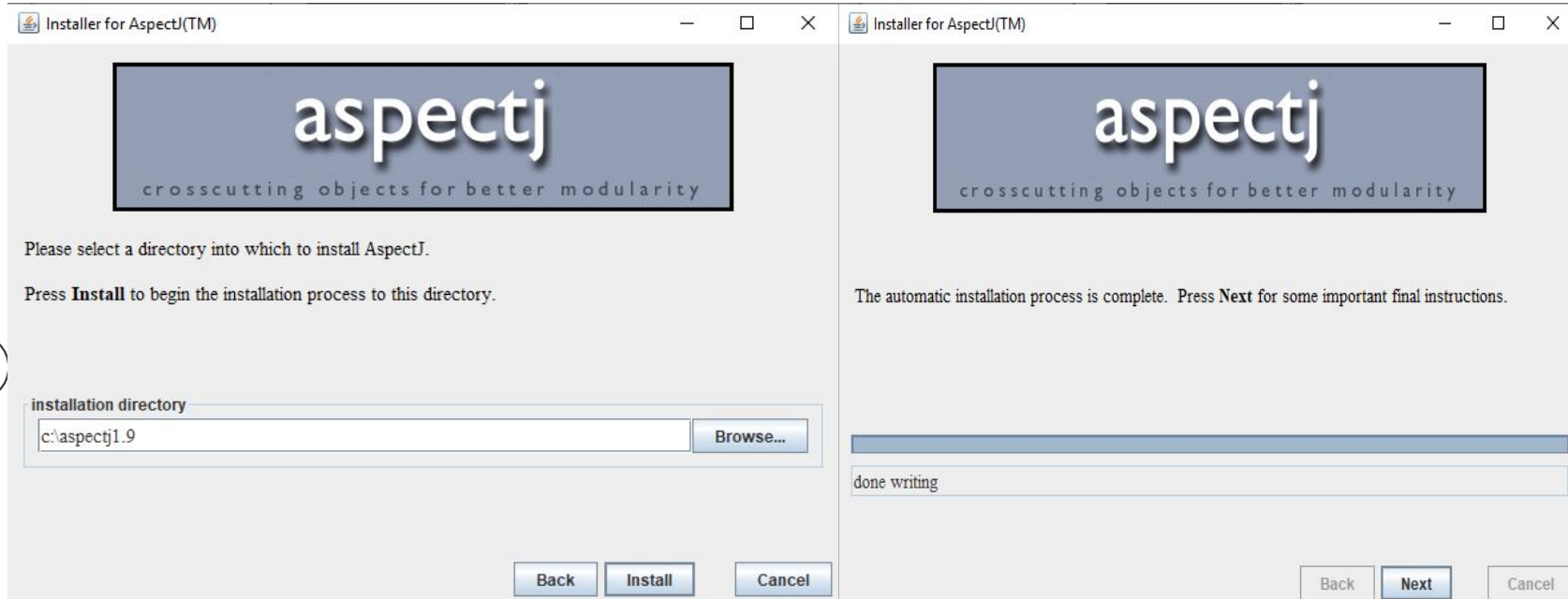
2. Hacer click en descargar y al terminar la descarga ejecutar el archivo .jar



The image shows two parts of the installation process. On the left, a screenshot of the AspectJ download page features a prominent orange 'Download' button with a downward arrow icon. Below the button, it states 'Download from: Brazil - C3SL - Federal University of Parana (https)'. The file name 'aspectj-1.9.6.jar' is displayed next to its SHA-512 hash, and a link '>> Select Another Mirror' is visible. On the right, a screenshot of a file explorer shows a context menu for the downloaded file 'aspectj-1.9.6.jar'. The menu options include 'Abrir', 'Abrir con Code', 'Shred with BleachBit', '7-Zip', and 'CRC SHA'.

# AspectJ - Instalacion

## 3. Seguir las instrucciones de instalación.



# AspectJ - Instalacion

## 3. Seguir las instrucciones de instalación.




# AspectJ - Instalacion

4.1. Añadir AspectJ a las variables de entorno (CLASSPATH y PATH) (Los pasos pueden variar de acuerdo a las versiones de windows) Podemos acceder al menú de sistema desde Panel de control > Sistema y Seguridad > Sistema



Ajustar la configuración del equipo



**Sistema y seguridad**  
Revisar el estado del equipo

Guardar copias de seguridad	Sistema y seguridad
Historial de archivos	Vea y cambie el estado del sistema
Copias de seguridad y	de seguridad, haga copias de



**Firewall de Windows Defender**  
Comprobar estado del firewall | Permitir una aplicación a través



**Sistema**  
Ver la cantidad de memoria RAM y la velocidad del procesador  
Iniciar asistencia remota | Mostrar el nombre de este equipo

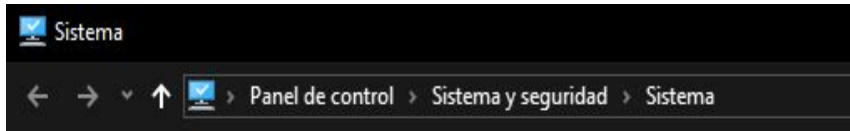


**Opciones de energía**  
Cambiar la configuración de batería | Cambiar las acciones de  
Cambiar la frecuencia con la que el equipo entra en estado de su



# AspectJ - Instalacion

4.2. Una vez dentro de sistema, procedemos a hacer click en Configuración Avanzada de Sistema > Variables de Entorno



Ventana principal del Panel de control

- Administrador de dispositivos
- Configuración de Acceso remoto
- Protección del sistema
- [Configuración avanzada del sistema](#)

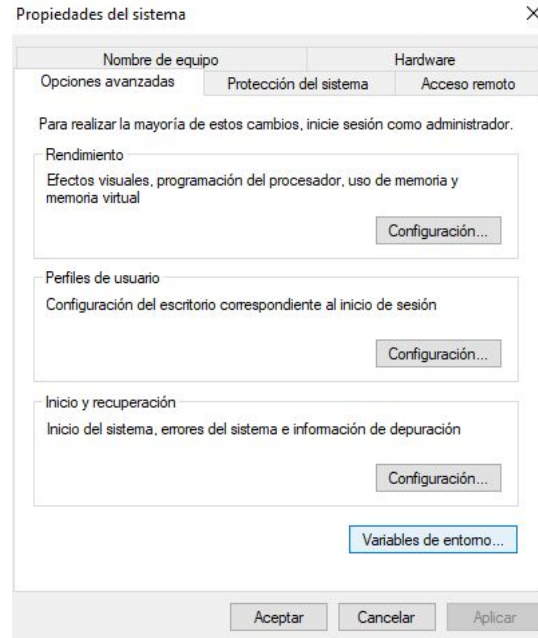
Ver información básica acerca del equipo

Edición de Windows

Windows 10 Enterprise LTSC

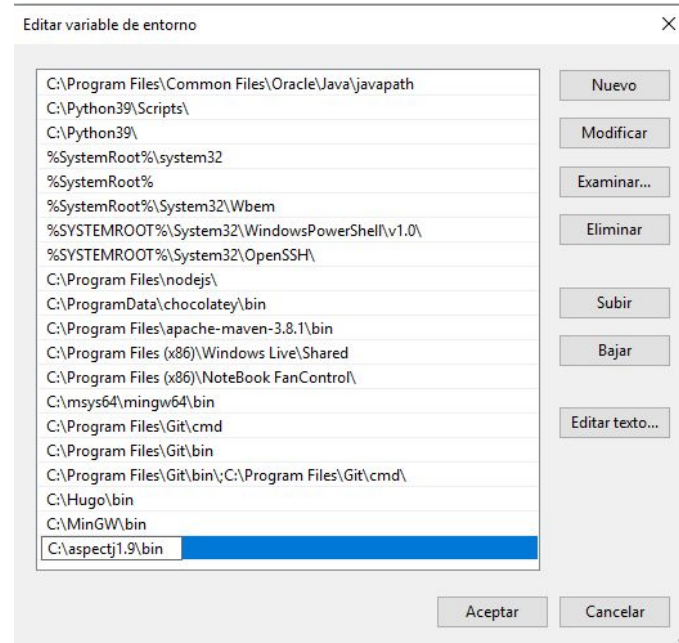
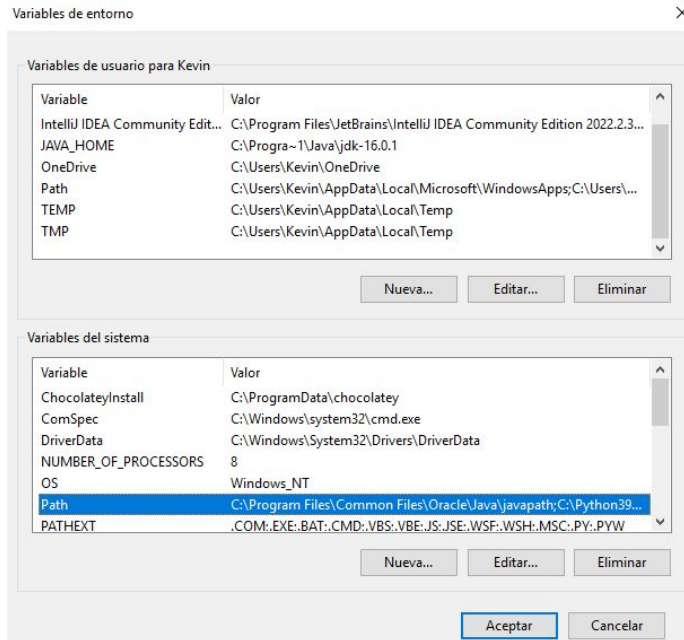
© 2018 Microsoft Corporation. Todos los derechos reservados.

Sistema



# AspectJ - Instalacion

4.3. Nos ubicamos en Variables del Sistema, y buscaremos la variable Path, daremos click en editar, una vez dentro, daremos click en nuevo, y anotaremos el directorio de instalación de AspectJ (Por defecto c:\aspectj1.9\bin) y daremos aceptar.



# AspectJ - Instalacion

4.4. Regresando al menú anterior, nos volveremos a ubicar en Variables del Sistema, daremos click en nuevo, en nombre de la variable colocaremos CLASSPATH, y en el valor el directorio de la librería de aspectJ (por defecto c:\aspectj1.9\lib\aspectjrt.jar) Finalmente dando click en aceptar en ambos menús.

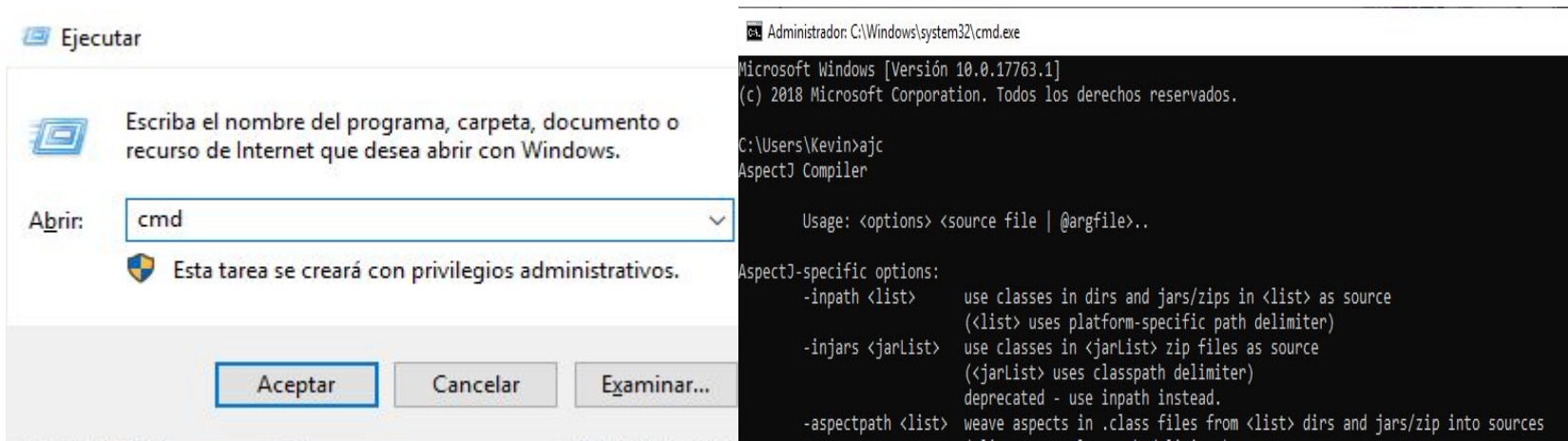
The image shows two overlapping windows from the Windows operating system. The background window is titled 'Variables de entorno' and is divided into two sections: 'Variables de usuario para Kevin' and 'Variables del sistema'. The 'Variables del sistema' section is active, showing a list of system variables. The 'Path' variable is selected and highlighted in blue. In the foreground, a smaller window titled 'Nueva variable del sistema' is open. It has two input fields: 'Nombre de la variable:' containing 'CLASSPATH' and 'Valor de la variable:' containing 'C:\aspectj1.9\lib\aspectjrt.jar'. At the bottom of this window, there are buttons for 'Examinar directorio...', 'Examinar archivo...', 'Aceptar', and 'Cancelar'. The 'Aceptar' button is highlighted with a blue border.

Variable	Valor
IntelliJ IDEA Community Edit...	C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.2.3...
JAVA_HOME	C:\Progra~1\Java\jdk-16.0.1
OneDrive	C:\Users\Kevin\OneDrive
Path	C:\Users\Kevin\AppData\Local\Microsoft\WindowsApps;C:\Users\...
TEMP	C:\Users\Kevin\AppData\Local\Temp
TMP	C:\Users\Kevin\AppData\Local\Temp

Variable	Valor
Chocolatey\Install	C:\ProgramData\chocolatey
ComSpec	C:\Windows\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
NUMBER_OF_PROCESSORS	8
OS	Windows_NT
Path	C:\Program Files\Common Files\Oracle\Java\javapath;C:\Python39...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC;.PY;.PYW

# AspectJ - Instalacion

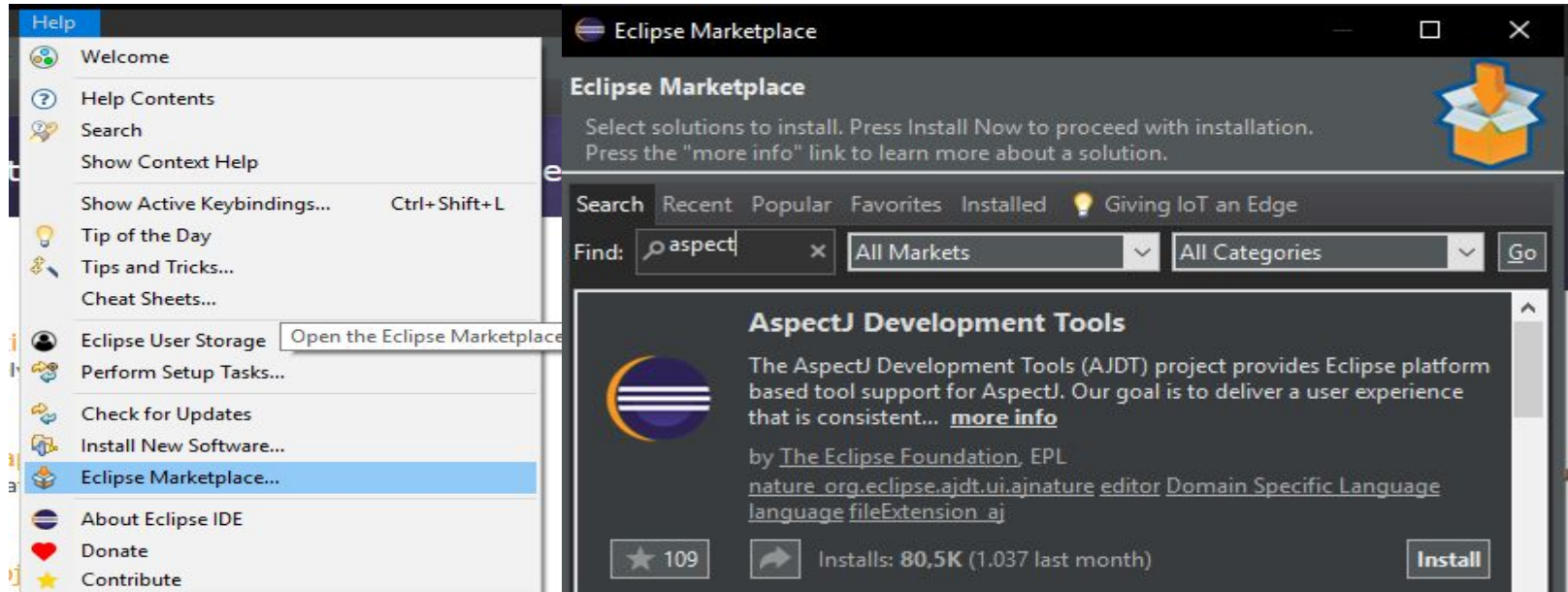
5. Finalmente comprobaremos la instalación abriendo la consola de windows (CTRL + R, escribir cmd y dar enter), en esta escribiremos el comando ajc, si la instalación fue correcta nos mostrará su información.



The image shows two overlapping windows. On the left is the Windows 'Ejecutar' (Run) dialog box. The title bar says 'Ejecutar'. The main text reads: 'Escriba el nombre del programa, carpeta, documento o recurso de Internet que desea abrir con Windows.' Below this, the 'Abrir:' field contains the text 'cmd'. A small shield icon with a checkmark and the text 'Esta tarea se creará con privilegios administrativos.' is visible below the input field. At the bottom, there are three buttons: 'Aceptar' (highlighted with a blue border), 'Cancelar', and 'Examinar...'. On the right is a terminal window titled 'Administrador: C:\Windows\system32\cmd.exe'. The terminal output shows: 'Microsoft Windows [Versión 10.0.17763.1] (c) 2018 Microsoft Corporation. Todos los derechos reservados. C:\Users\Kevin>ajc AspectJ Compiler Usage: <options> <source file | @argfile>.. AspectJ-specific options: -inpath <list> use classes in dirs and jars/zips in <list> as source (<list> uses platform-specific path delimiter) -injars <jarList> use classes in <jarList> zip files as source (<jarList> uses classpath delimiter) deprecated - use inpath instead. -aspectpath <list> weave aspects in .class files from <list> dirs and jars/zip into sources'

# AspectJ - Instalacion

6. Ahora procederemos a instalar extensiones para nuestro IDE, en este caso Eclipse, una vez abierto iremos a la opción help, abrimos el marketplace, instalamos la extensión y esperamos a que termine.



The screenshot displays the Eclipse IDE interface. On the left, the 'Help' menu is open, with 'Eclipse Marketplace...' selected. On the right, the 'Eclipse Marketplace' window is active, showing search results for 'aspectj'. The search results list 'AspectJ Development Tools' with a star rating of 109 and 80,5K installs (1,037 last month). An 'Install' button is visible next to the listing.

**Help**

- Welcome
- Help Contents
- Search
- Show Context Help
- Show Active Keybindings... Ctrl+Shift+L
- Tip of the Day
- Tips and Tricks...
- Cheat Sheets...
- Eclipse User Storage Open the Eclipse Marketplace
- Perform Setup Tasks...
- Check for Updates
- Install New Software...
- Eclipse Marketplace...**
- About Eclipse IDE
- Donate
- Contribute

**Eclipse Marketplace**

Select solutions to install. Press Install Now to proceed with installation. Press the "more info" link to learn more about a solution.

Search Recent Popular Favorites Installed Giving IoT an Edge

Find: aspectj All Markets All Categories Go

**AspectJ Development Tools**

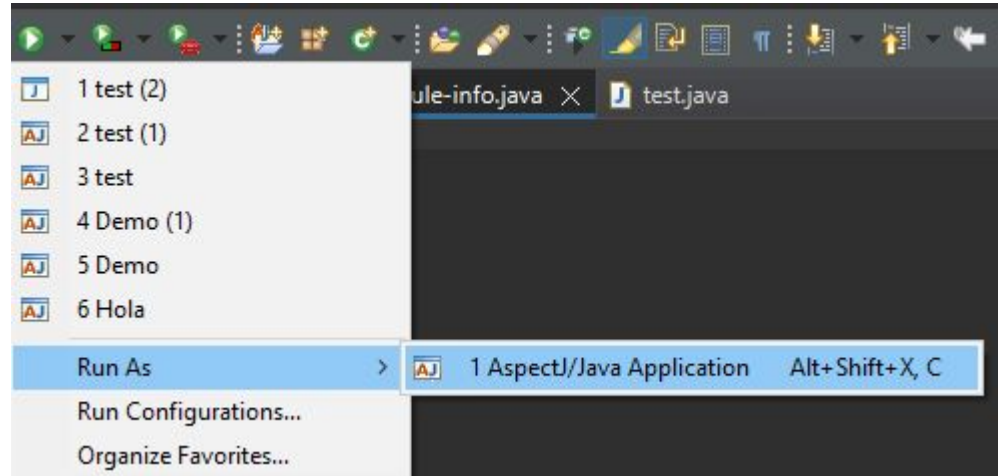
The AspectJ Development Tools (AJDT) project provides Eclipse platform based tool support for AspectJ. Our goal is to deliver a user experience that is consistent... [more info](#)

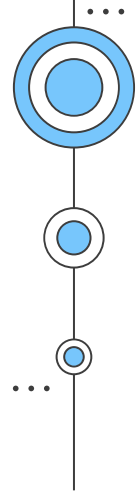
by The Eclipse Foundation, EPL  
[nature](#) [org.eclipse.ajdt.ui.ajnature](#) [editor](#) [Domain Specific Language](#) [language](#) [fileExtension](#) [aj](#)

★ 109 Installs: 80,5K (1.037 last month) **Install**

# AspectJ - Instalacion

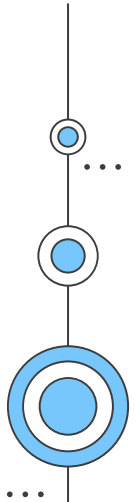
7. Finalmente, al ejecutar un programa, presionamos las opciones de run y damos click en Run As > AspectJ Application





# 03

## Conceptos básicos



# Conceptos básicos



Crosscutting concerns  
(Conceptos entrecruzados)



Aspects (Aspectos)



Joinpoints (Puntos  
de unión)



Pointcuts (Puntos  
de corte)

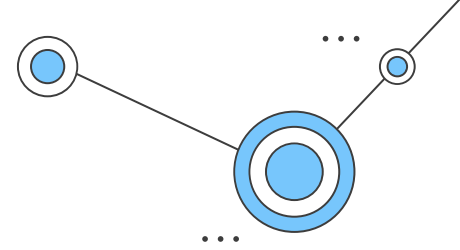


Advice (Avisos)

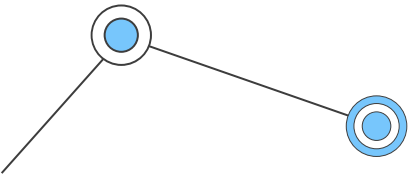
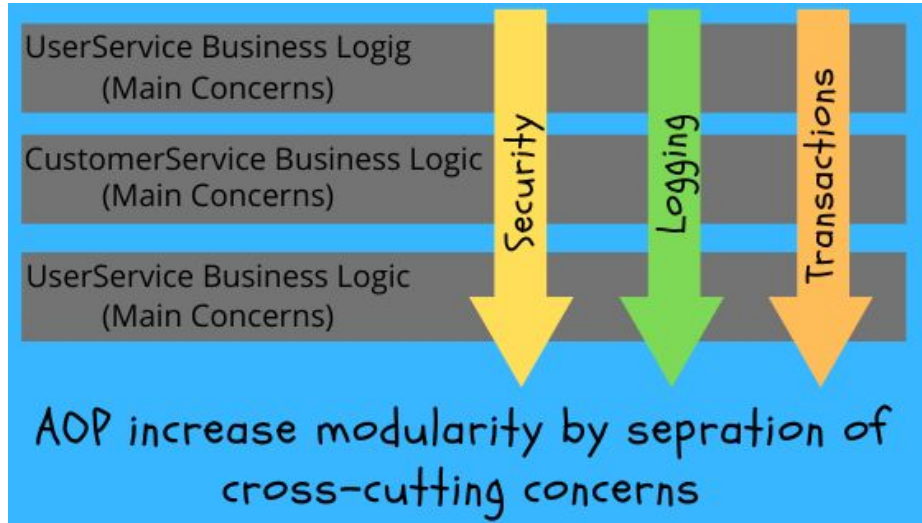




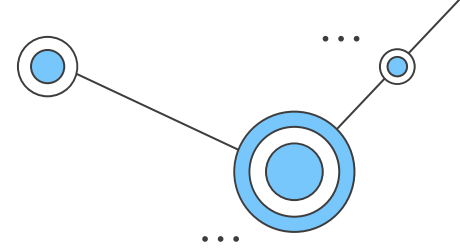
# Crosscutting concerns (Conceptos entrecruzados)



La mayoría de las aplicaciones contienen funcionalidad común que se esparce tanto verticalmente (en capas y tiers) como horizontalmente (entre módulos funcionales). Esta funcionalidad es generalmente descrita como crosscutting concerns porque afecta a la aplicación entera

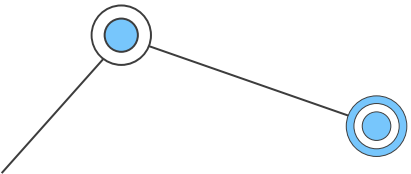
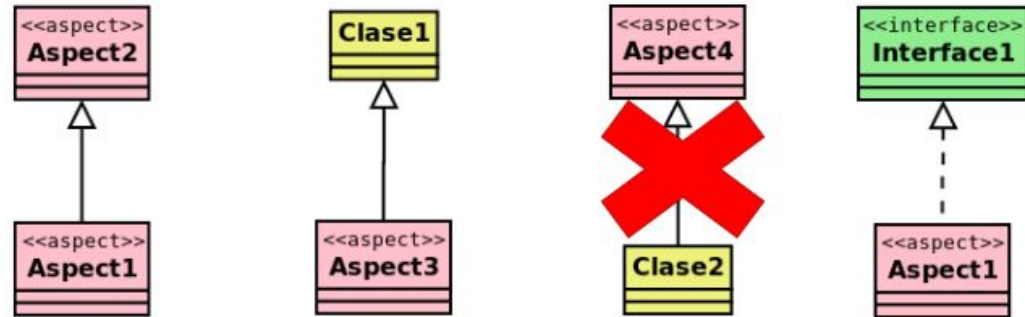


# Aspects (Aspectos)

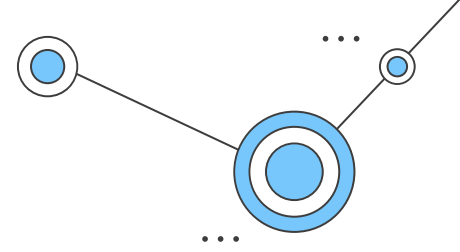


Es la unidad básica en la programación orientada a aspectos para implementar un crosscutting concern. Es muy parecido a una clase de programación orientada a objetos, con un nombre, unas variables y unos métodos.

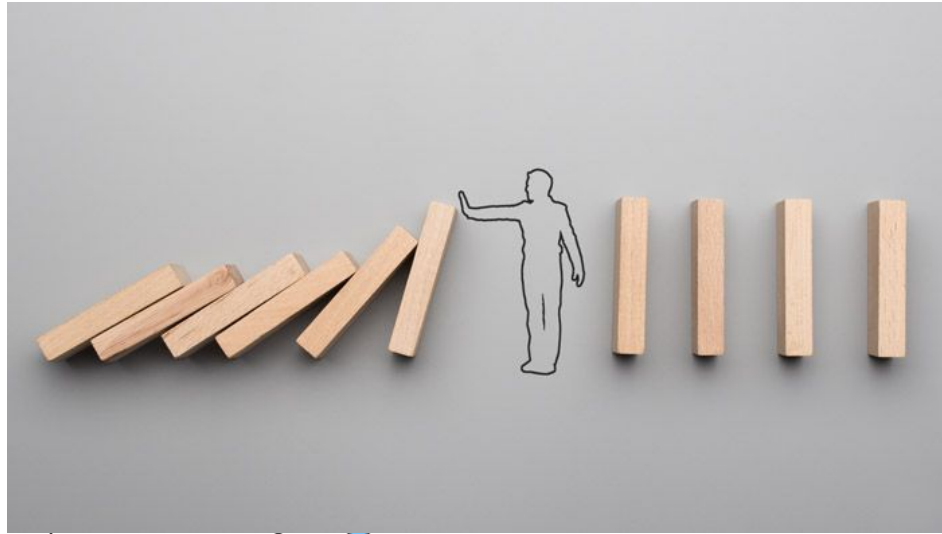
```
1 [privileged] aspect [ID] [extends][implements]{
2     body
3 }
```



# Aspectos



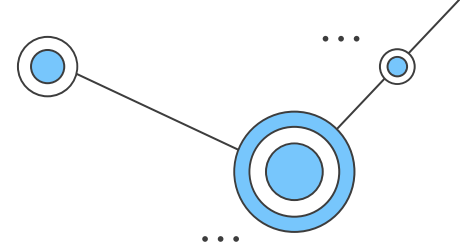
## Dinamico



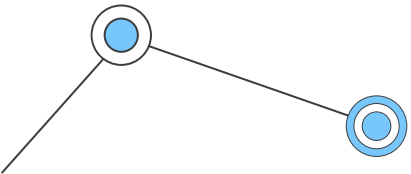
## Estatico



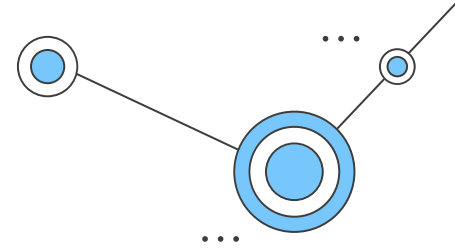
# Joinpoints (Puntos de unión)



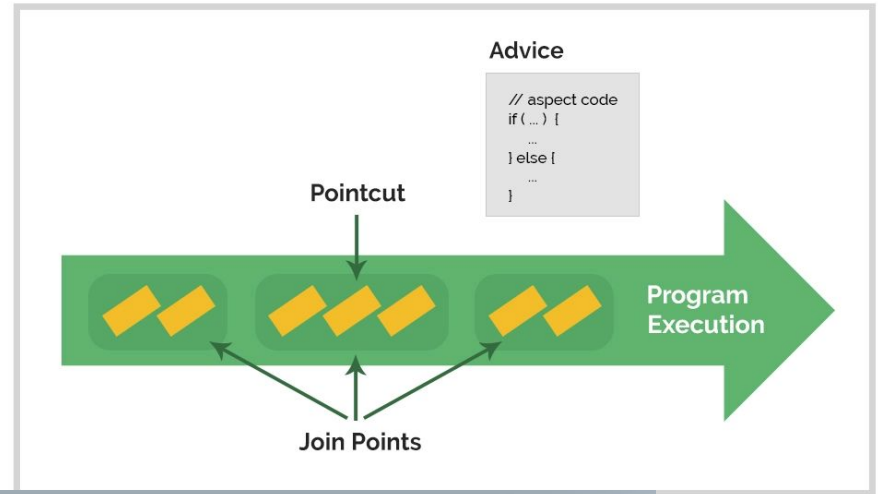
Son puntos bien definidos en la estructura de un programa que permiten añadir funcionalidad adicional. Los más comunes son llamadas a funciones



# Pointcuts (Puntos de corte)



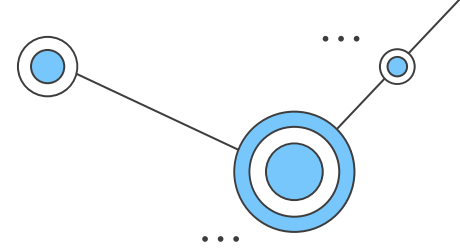
Describe un conjunto de joinpoints. Se trata de una herramienta muy útil para referenciar todos los joinpoints donde cierto código se quiere invocar, reduciendo el riesgo de invocar incorrectamente un aspecto



```
1 pointcut verifyAssurance():execution(void PetStore.makeAppointment(..)
2 || execution(void PetStore.UpdateLevel(..));
```



# Pointcuts (Puntos de corte)

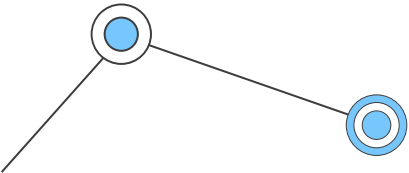


## Comodines

\* Sustituye a un conjunto de caracteres, excepto el punto.

.. Sustituye a un conjunto de parámetros.

+ Busca los joinpoint dentro del package o subpackage.



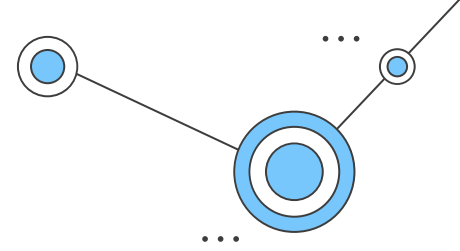
## Operadores

! Operador NOT. Utilizado para no seleccionar el joinpoint indicado.

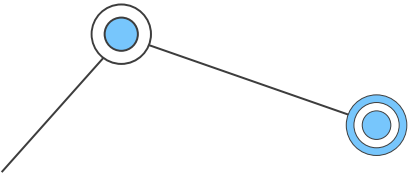
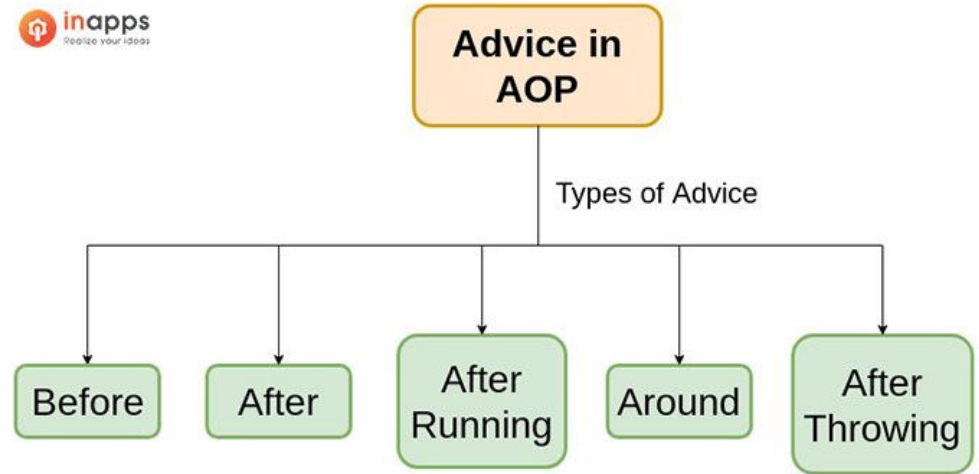
|| Operador OR. Se utiliza para unir un conjunto de pointcuts con la condición de uno u otro.



&& Operador AND. Se utiliza para unir un conjunto de pointcuts con la condición de uno y otro.

# Advice (Avisos)



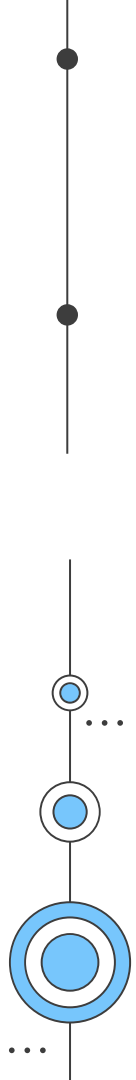
Un advice corresponde a las acciones que se ejecutan en cada joinpoint incluido en un pointcut.





# 04

## Ventajas y desventajas





# Ventajas

Facilidad para modularizar

Sistema mantenible

Extiende las  
capacidades de P00



# Desventajas

Pueden existir conflictos entre las combinaciones de aspectos y clases

Mala elección de delegaciones

Poca documentación





05

Características



# Forma de entrelazamiento

- **Tejido estático**

En esta forma se modifica el código fuente insertando sentencias en los joinpoints, es decir, se introducen los aspectos en el código fuente.

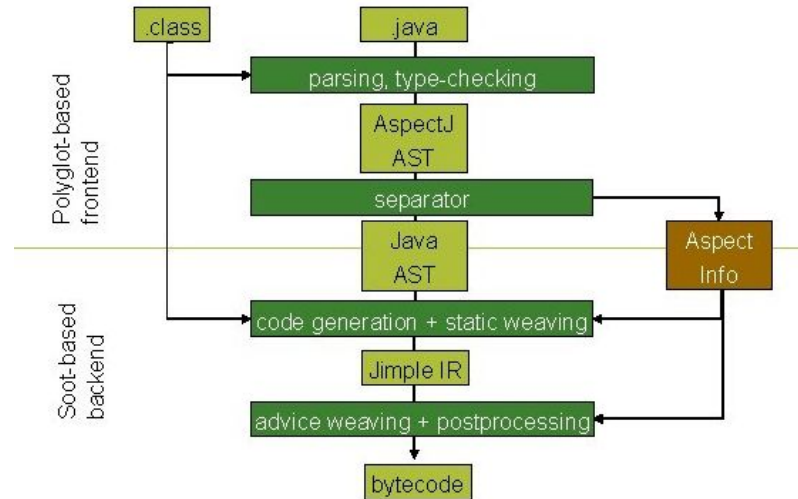
- **Tejido dinámico**

Requiere que los aspectos estén presentes de forma explícita tanto en tiempo de compilación como de ejecución, para poder modificarse de manera dinámica.

# Otro compilador diseñado para AspectJ

- **ABC**

Compilador optimizado que es producido por aspectbench, pero no tiene un IDE



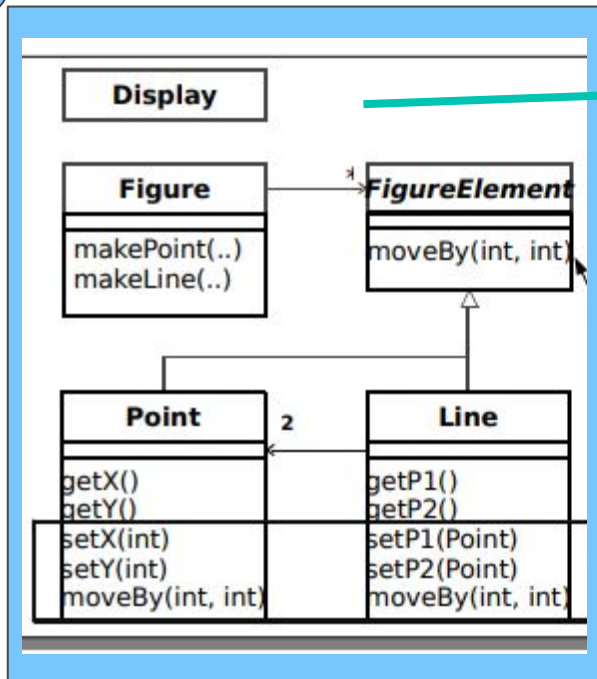


06

Ejemplos



# Ejemplos - Utilidad



Elemento que se actualiza siempre un elemento actualiza su estado (Setters)

```
public class Point {
    int x;
    int y;

    public void setX(int x) {
        this.x = x;
        Display.update(this);
    }

    public void setY(int y) {
        this.y = y;
        Display.update(this);
    }
}
```

```
public class Line {
    Point p1;
    Point p2;

    public void setP1(Point p1) {
        this.p1 = p1;
        Display.update(this);
    }




    public void setP2(Point p2) {
        this.p2 = p2;
        Display.update(this);
    }
}
```

```
public class Display {
    public static void update(Line line) {
        System.out.println("Se actualizo elemento gráfico");
    }

    public static void update(Point point) {
        System.out.println("Se actualizo elemento gráfico");
    }
}
```

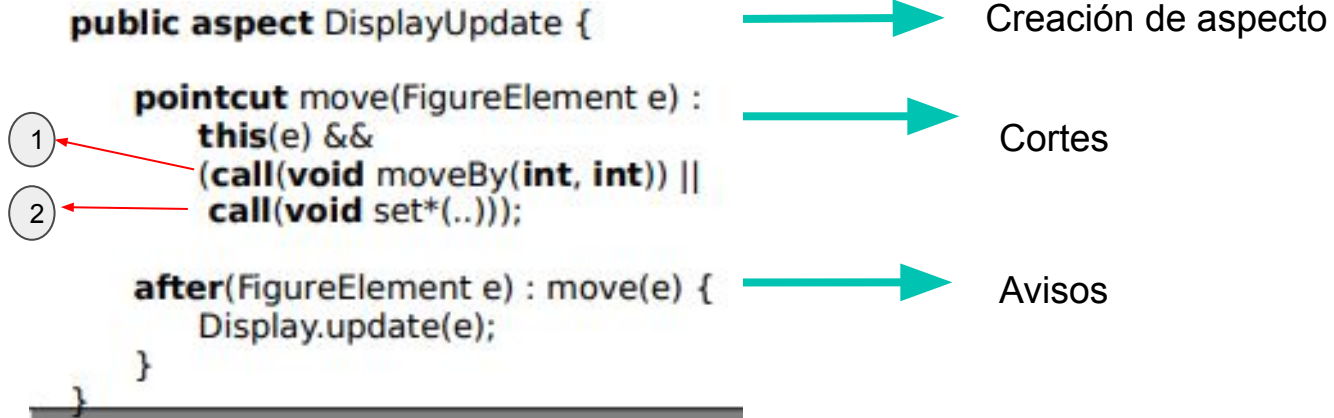
# Recordando...

## Join Points

<pre>public class Point {     int x;     int y;</pre>		Instancia clase
<pre>    public void setX(int x) {         this.x = x;         Display.update(this);     }</pre>		Llamado de un método
<pre>    public void setY(int y) {         this.y = y;         Display.update(this);     }</pre>		Acceso a un campo



# Creación de Aspect



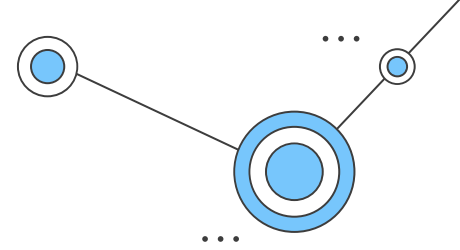
# Un poco más complicado

```
public aspect ObserverInstance {  
    private Vector<Display> Point.observers = new Vector<Display>();  
  
    public static void addObserver(Point p, Display d) { p.observers.add(d); }  
    public static void removeObserver(Point p, Display d) { ... }  
  
    pointcut changes(Point p) :  
        target(p) && ( call (void Point.set*(int)) ||  
                      call (void Point.moveBy(int, int)) );  
  
    after(Point p) : changes(p) {  
        for (Display d : p.observers) {  
            updateObserver(p, d);  
        }  
    }  
    static void updateObserver(Point p, Display d) {  
        d.update(p);  
    }  
}
```

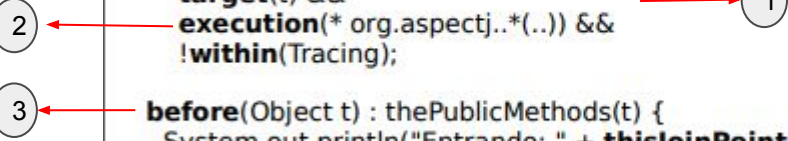
# Ejemplos PointCuts

PUNTO DE ENLACE	DESCRIPCIÓN
<code>call (void Clase.set*(..))</code>	Se ejecuta cuando se llama a cualquiera de los métodos de <i>Clase</i> que empiezan por <i>set</i> , retornan <i>void</i> y tienen cero o muchos parámetros
<code>initialization (public Objeto.new(..))</code>	Se ejecuta cuando se crea una instancia de la clase <i>Objeto</i> , especificando el constructor
<code>get (public * Clase.*)</code>	Se ejecuta cuando se accede a un atributo público de <i>Clase</i>
<code>Handler (IOException)</code>	Se ejecuta cuando se lanza o dispara el manejador de excepciones de <i>IOException</i>
<code>staticinitialization (Logger)</code>	Se ejecuta cuando se crea una instancia estática de la clase <i>Logger</i>

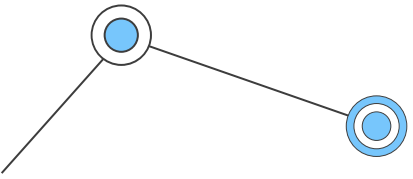
# Ejemplo 2 -



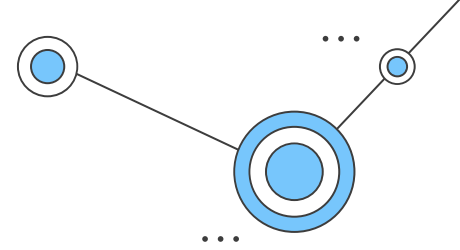
```
public aspect Tracing {  
    pointcut thePublicMethods(Object t) :  
        target(t) &&  
        execution(* org.aspectj..*(..)) &&  
        !within(Tracing);  
  
    before(Object t) : thePublicMethods(t) {  
        System.out.println("Entrando: " + thisJoinPoint.getSignature());  
    }  
}
```



```
Entrando: void org.aspectj.Point.setX(int)  
Entrando: void org.aspectj.Point.setY(int)  
Entrando: void org.aspectj.Line.moveBy(int, int)  
Entrando: void org.aspectj.Point.moveBy(int, int)  
Entrando: int org.aspectj.Point.getX()  
Entrando: void org.aspectj.Point.setX(int)  
Entrando: int org.aspectj.Point.getY()  
Entrando: void org.aspectj.Point.setY(int)  
Entrando: void org.aspectj.Point.moveBy(int, int)  
Entrando: int org.aspectj.Point.getX()  
Entrando: void org.aspectj.Point.setX(int)  
Entrando: int org.aspectj.Point.getY()  
Entrando: void org.aspectj.Point.setY(int)  
Entrando: void org.aspectj.Point.setY(int)
```

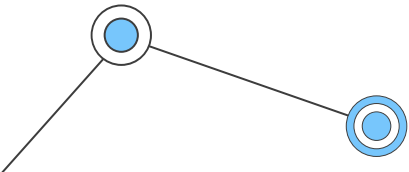


# Ejemplo 3 -

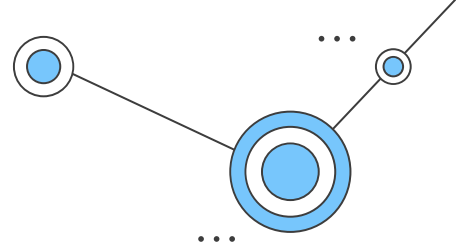


```
public aspect PointBoundsChecking {  
    1 ← pointcut setX(int x) :  
        ((call(void Point.moveBy(int,int)) && args(x, *))  
        || (call(void Point.setX(int)) && args(x))); → 1  
  
    3 ← pointcut setY(int y) :  
        ((call(void Point.moveBy(int,int)) && args(*, y))  
        || (call(void Point.setY(int)) && args(y))); → 4  
  
    5 ← before(int x) : setX(x) {  
        if (x < MIN_X || x > MAX_X)  
            throw new IllegalArgumentException("x está fuera de rango.");  
    }  
  
    6 ← before(int y) : setY(y) {  
        if (y < MIN_Y || y > MAX_Y)  
            throw new IllegalArgumentException("y está fuera de rango.");  
    }  
}
```

```
Problems Console Debug Progress  
<terminated> Main (104) [AspectJ]java Application] /usr/lib/jvm/java-6-sun-1.6.0.03/bin/java [28/03/2008 09:42:08]  
Entrando: int org.aspectj.Point.getX()  
Entrando: void org.aspectj.Point.setX(int)  
Entrando: int org.aspectj.Point.getY()  
Entrando: void org.aspectj.Point.setY(int)  
Exception in thread "main" java.lang.IllegalArgumentException y está fuera de rango.  
    at org.aspectj.examples.PointBoundsChecking.ajc$before$org_aspectj_examples_PointBo  
    at org.aspectj.Main.main(Main.java:22)
```

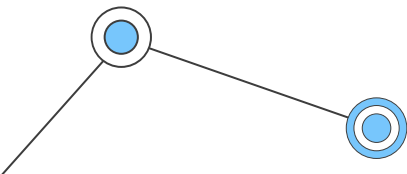


# Ejemplo 4 -



```
public aspect SystemOutMonitor {  
  
    pointcut syserrAccess() : get(* System.err); → 1  
  
    pointcut sysoutAccess() : get(* System.out); → 2  
  
    declare warning : syserrAccess() || sysoutAccess() :  
        "Please don't write messages to System out or err."  
}
```

```
System.out.println("Probando...");  
System.out.println("Probando...");  
Please don't write messages to System out or err;  
System.out.println("Probando...");  
System.out.println("Probando...");
```



# Ejemplo 5 -

```
public class DVD extends Product {
    private String title;
    ...
}

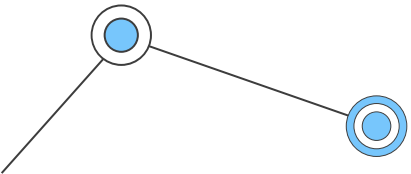
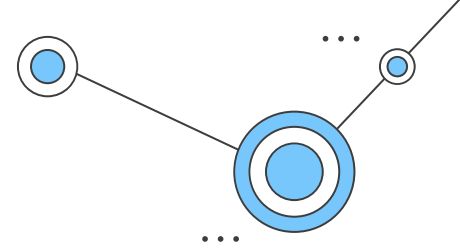
public aspect OutputType {
    pointcut callToDVDConstructor(): call((DVD).new(..));

    before(): callToDVDConstructor() {
        SourceLocation sl = thisJoinPoint.getSourceLocation();
        Class theClass = (Class) sl.getWithinType();
        System.out.println(theClass.toString());
    }
}
```

Output: class DVD

## ■ Methods

- `getThis()`
- `getTarget()`
- `getArgs()`
- `getSignature()`
- `getSourceLocation()`
- `getKind()`
- `toString()`
- `toShortString()`
- `toLongString()`



# Métodos para aplicar

```
public static void makeAppointment(){
    int id = Integer.valueOf(readConsole("Id:"));
    String date = readConsole("Date: ");
    users.get(id-1).setAppointment(date);
}

public static void UpdateLevel(){
    int id = Integer.valueOf(readConsole("Key: "));
    String assurance = readConsole("Membership level: ");
    users.get(id-1).setLevelAssurance(assurance);
}
```



# Ejemplo 6

```
public aspect CheckAssurance{  
    private static Scanner input = new Scanner(System.in);  
    pointcut verifyAssurance():call(void PetStore.*(..))
```

```
    around(): verifyAssurance(){  
        System.out.println("Executing method");  
    }  
}
```

# Ejemplo 7

```
public aspect CheckAssurance{
    private static Scanner input = new Scanner(System.in);
    pointcut verifyAssurance():call(void PetStore.*(..))
        || execution(void User.Update*(..));

    after(): verifyAssurance(){
        System.out.println("Request was executed successfully");
        readConsole("Press another key to keep using: ");
    }
}
```

# Ejemplo 8

```
public aspect CheckAssurance{
    private static Scanner input = new Scanner(System.in);
    pointcut verifyAssurance():execution(void PetStore.makeAppointment(..)
        || execution(void PetStore.UpdateLevel(..));

    before() : verifyAssurance(){
        System.out.println("Verifying assurance...");
        String name = readConsole("Name of your pet: ");
        System.out.println("Welcome "+ name);
        System.out.println("You are able to use this service");
    }
}
```



**GRACIAS!**

# Referencias

- <https://www.ida.liu.se/~chrke55/courses/SWE2017-CBSE/F07-AspectJ.pdf>
- <https://www.baeldung.com/aspectj>
- <https://es.wikipedia.org/wiki/AspectJ>
- <https://www.eclipse.org/aspectj/>
- <https://www.eclipse.org/aspectj/doc/released/progguide/starting.html>
- <https://ferestrepoca.github.io/paradigmas-de-programacion/poa/tutoriales/aspectJ/index.html>
- <http://ferestrepoca.github.io/paradigmas-de-programacion/poa/tutoriales/aspectJ/assets/presentaciones/presentacion2020-2.pdf>
- <http://ferestrepoca.github.io/paradigmas-de-programacion/poa/tutoriales/aspectJ/assets/presentaciones/presentacion2022-1.pdf>
-