**Empezando con Jess: Primeros Pasos**

0.- Instalación:

- Desempaqueta el Jess71p2.zip en el directorio que quieras

(mejor donde tienes instalados el resto de tus programas)

*(Para 1ª Entrega Práctica 2: Uso de Jess con el interprete)*

1.- Ejemplo básico de sintaxis deftemplate, deeffacts, defrule : <ejemploTrozoJuegoJess.txt>

2.- Funciones básicas del intérprete: Una ejecución desde shell <runJessEnShellUnix.txt>

- Ejemplos de la distribución en \Jess71p2\examples\jess

- Ejemplo interactivo: <sticks.clp>

3.- Hacer ejercicios Hoja 5 : 3 y 4

*( Para 2ª Entrega Práctica 2: Uso de Jess desde una aplicación Java )*

4.- Pasos para esta parte

- Instalar el plugin de Jess para eclipse (JessDE) siguiendo la *Seccción A*

- Para ver cómo funciona usaremos un ejemplo de la distribución de Jess (pricing\_engine):

Sigue todos los pasos indicados en la *Sección A* para crear un proyecto nuevo PreciosJess.

(javax.rules es la API para llamar a Jess, dir. JSR94

- Información del funcionamiento de la API Java- Jess con otro ejemplo:

<tutorialBreveJessConJava.docx>

- Documentación interesante con muchos ejemplos: ver *Sección C*

**SECCION A**:

**A.1.- Instalación paso a paso para la 2ª entrega**

Cómo instalar todo y probar con un ejemplo que tra la distribución de Jess, usando el plugin JessDE para eclipse. El ejemplo “pricing\_engine” es java que llama a archivo de Jess.

-- Instalar java (si tienes un ordenador nuevo puede que no esté el jdk):

el jre y jdk siguiendo los pasos indicados en:

<http://java.sun.com/javase/downloads/index.jsp>

qué versión de Java tengo?: para JRE java -version y para la de JDK : javac -version

-- Instalo una versión de eclipse: descomprimo y ya está

Puedes bajar varias versiones, por ejemplo: eclipse-java-galileo-SR2-win32.zip

Eclipse Java EE IDE for Web Developers: eclipse-jee-galileo-SR1-win32.zip

-- Instalar el plugin JessDE en eclipse:

Descomprimo lo que hay en \Jess71p2\eclipse y lo coloco en dir. *features* y en dir. *plugins* del dir raiz de eclipse. Ver

<http://www.jessrules.com/doc/70/eclipse.html> (opcion del menu izq)

-- Eclipse:

- Crear un proyecto nuevo(File + New Project): PreciosJess

- Importar Ficheros (File + Import) de C:\Program Files\Jess71p2\examples\pricing\_engine

Marcar "select all" ...Into Folder : seleccionas el proyecto nuevo PreciosJess

- Incluir las librerías de Jess: jess.jar y jsr94.jar

- Ratón encima del proyecto PreciosJess + botón dcho + Build Path + Configure Path

(también en la barra de menús “Project” opción “Properties”)

- Abre ventana “Properties for PreciosJess” en columna izq “Java Build Path” :

- Pestaña “Libraries” + columna dcha “add external JARs” :

Buscar la distribución de Jess \Jess71p2\lib y marcar las dos jess.jar y jsr94.jar

**A.2.- Más detalles útiles (no imprescindibles, para manitas)**

-- *Facilidades para usar Jess desde el interprete (el prompt):*

Como No tiene edición en la línea de comandos, hay dos modos de facilitar el uso:

a)- Usar una shell en emacs (si no conoces emacs antes: NO te lo recomiendo)

b)- Instalar (fácil) Jline, ver <http://jline.sourceforge.net/>

-- *Puedes ver el diagrama del rete desde eclipse:*

<http://www.jessrules.com/doc/70/eclipse.html#install_details>

Para usar el Rete Network View, necesitas instalar Eclipse Graph Editing Framework (GEF) . Puedes conseguirlo desde la página de Eclipse: <http://www.eclipse.org/gef/>

O también instalarlo mediante el Eclipse's built-in update manager.

Después, para visualizar el rete: está en el grupo "Jess Debugger" en el diálogo "Show view" de Eclipse. Cuando el cursor esté dentro de una regla en la ventana del editor Jess, la Rete Network View mostrará la red compilada para esa regla.

**SECCION B** : Información y Capturas del proyecto **Pricing** (ejemplo de la distribucion jess)

Este ejemplo está explicado con detalle en el capitulo 11 del manual, "Embedding Jess in a Java Application" .

🡪 Hay tres paquetes Java:

------ gov.sandia.jess.example.pricing:

La clase gov.sandia.jess.example.pricing.PricingEngine

Muestra un modo de usar la API de Jess para Java.

- Crea el motor con Rete, carga las reglas del fichero "pricing.clp"

- Carga la base de datos en memoria de trabajo

- Añade cada orden, cliente e items a la memoria de trabajo del motor

La clase gov.sandia.jess.example.pricing.Database

- Interface de la base de datos simulada:

implementada con datos en paquete demo con la clase DemoDatabase

------ gov.sandia.jess.example.pricing.model

- Tiene las clases del modelo para una aplicacion simple de e-commerce

------ gov.sandia.jess.example.pricing.demo

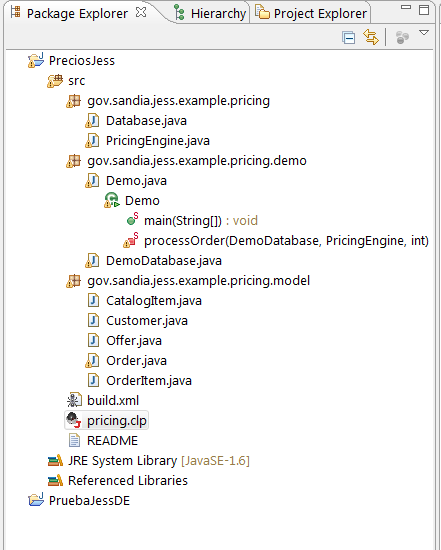
- Tiene la clase driver para ejecutar todo y algunos datos simulando una base de datos

- Las reglas están en el fichero pricing.clp .

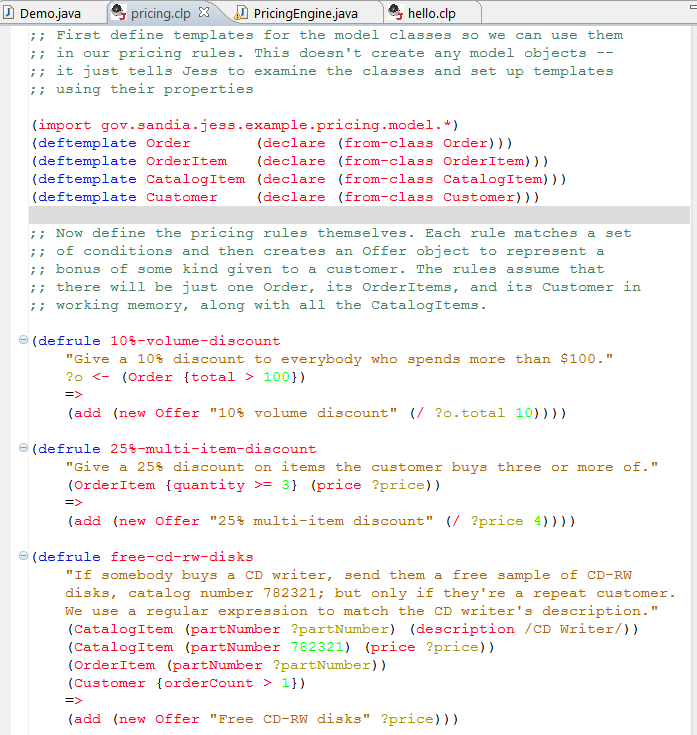
- Tambien hay un fichero ant para compilar todo: build.xml (desde eclipse es automático)

🡪 Para ejecutar : compilar todo y ejecutar gov.sandia.jess.example.pricing.demo.Demo

LA JERARQUIA DE CLASES



EL FICHERO DE REGLAS:



**SECCION C: DOCUMENTACION UTIL**

(los “\” para linux son “/” para Windows)

Documentación dentro de la distribución \Jess71p2\docs\index.html

/Jess71p2/docs/intro.html

/Jess70p2/docs/rules.html

/Jess70p2/docs/api.html: Jess - the Rule Engine for the Java Platform v70p2

The JSR94 API: **javax.rules, javax.rules.admin**

Ejemplos : \Jess71p2\examples

En la Web la Wiki de Jess <http://www.jessrules.com/jesswiki/view>

<http://www.jessrules.com/jess/mailing_list.shtml>