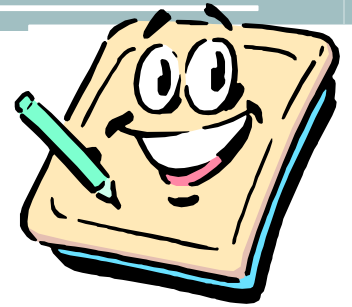


Inteligencia Artificial

CLIPS - Sesión 9

Entregas



Hoy:

TPO Entrega 5 Conceptualización

28/5:

TP7 Integración CLIPS

Objetivo y Contexto de la Sesión

- Conocer como se integra un programa CLIPs en una aplicación JAVA o .NET sobre una plataforma Windows.
- Definir requerimientos funcionales de y de usabilidad de la aplicación a desarrollar en el contexto del TPO.

















Integrando CLIPS con JAVA

Introducción

- CLIPSJNI (Clips Java Native Interface)
- Descargar ultima versión SourceForge del ZIP (Clips_jni_xxx.zip)
- El zip contiene el código fuente y las dlls compiladas para 32 y 64 bits.
- En Plataformas Windows:
 - Verificar la correcta instalación de la DLL (32/64 Bits)
- El paquete incluye aplicaciones de ejemplo para tomar como referencia.
 - Código JAVA
 - Código CLIPS

Integrando CLIPS con JAVA

Introducción

Name	Date modified	Type	Size
 bin	12/06/2015 12:21	File folder	
 java-src	12/06/2015 12:21	File folder	
 library-src	12/06/2015 12:21	File folder	
 AnimalDemo.jar	11/06/2015 18:26	Archivo WinRAR	50 KB
 AutoDemo.jar	11/06/2015 18:26	Archivo WinRAR	30 KB
 CLIPSJNI.dll	11/06/2015 17:46	Application extens...	1.448 KB
 CLIPSJNI.jar	11/06/2015 18:26	Archivo WinRAR	12 KB
 CLIPSJNI32.dll	11/06/2015 18:22	Application extens...	1.048 KB
 CLIPSJNI64.dll	11/06/2015 17:46	Application extens...	1.448 KB
 instructions.pdf	12/06/2015 12:58	Adobe Acrobat D...	476 KB
 libCLIPSJNI.jnilib	11/06/2015 17:19	JNILIB File	2.633 KB
 makefile.linux	01/06/2015 16:54	LINUX File	4 KB
 makefile.mac	04/06/2015 13:22	MAC File	4 KB
 makefile.win	04/06/2015 13:07	WIN File	5 KB
 SudokuDemo.jar	11/06/2015 18:26	Archivo WinRAR	30 KB
 WineDemo.jar	11/06/2015 18:26	Archivo WinRAR	27 KB

Integrando CLIPS con JAVA


Ejemplo

- **Importación:**
 - `import net.sf.clipsrules.jni.*;`
- **Definición de la variable**
 - `private Environment clips;`
- **Principales Métodos / Ejecución del programa clips**
 - `clips = new Environment();`
 - `clips.loadFromResource("/net/sf/clipsrules/jni/examples/programa.clp");`
 - `clips.reset();`
 - `String evalString = "(assert (hecho (slot1 " + valor1 + ") (slot2 " + valor2 + ")))";`
 - `clips.eval(evalString);`
 - `clips.run();`

Integrando CLIPS con JAVA

Ejemplo

- **Obtención de los Resultados:**



*Busca todos los
hechos “diagnostico”*

- `String evalstr= "(find-all-facts ((?J diagnostico)) TRUE)";`
- `MultifieldValue pv = (MultifieldValue) clips.eval(evalstr);`
- `FactAddressValue fv = (FactAddressValue) pv.get(0);`
- `String s = null;`
- `try {`
 - `s = fv.getFactSlot("resultado").toString();`
 - `} catch (Exception e1) {`
 - `e1.printStackTrace();`
 - `}`
- `ResultadoFinal = s;`

Integrando CLIPS con JAVA

Ejemplo

- **CLIPS (FIND-FACT y FIND-ALL-FACTS)**

```
(deftemplate myTemp (slot one) (slot second) (slot third) (slot fourth)

(assert (myTemp (one asd) (second jhg))
        (myTemp (one asd) (second kjh))
        (myTemp (one bvc) (second jhg))
        (myTemp (one bvc) (second jhg) (third qwe)))

(find-fact ((?p myTemp)) (or (eq ?p:one bvc) (eq ?p:third qwe)))
<Fact-3>
(find-fact ((?p myTemp)) (or (eq ?p:one bvc) (eq ?p:second jhg)))
<Fact-1>
(find-all-facts ((?p myTemp)) (or (eq ?p:one bvc) (eq ?p:second jhg)))
<Fact-1> <Fact-3> <Fact-4>
```

- **Instanciación de múltiples entornos CLIPS.**

```
▪ clips = new Environment(); . . . clips.destroy();
```














Integrando CLIPS con .NET

Introducción

- CLIPSCLRWrapper.dll (Clips Common Language Runtime Wrapper)
- Descargar ultima versión de SourceForge del Zip (Clips_dotnet_###.zip)
- El zip contiene la solución de la DLL y versiones compiladas para 32 y 64 bits.
- El paquete incluye aplicaciones de ejemplo para tomar como referencia.
 - Código .NET
 - Código CLIPS

Integrando CLIPS con .NET

Introducción

Name	Date modified	Type	Size
 AnimalFormsExample	12/08/2015 15:05	File folder	
 AnimalWPFFExample	12/08/2015 15:05	File folder	
 AutoFormsExample	12/08/2015 15:05	File folder	
 AutoWPFFExample	12/08/2015 15:05	File folder	
 CLIPSCLRWrapper	12/08/2015 15:04	File folder	
 Executables	12/08/2015 15:23	File folder	
 RouterFormsExample	12/08/2015 15:04	File folder	
 RouterWPFFExample	12/08/2015 15:05	File folder	
 WineFormsExample	12/08/2015 15:05	File folder	
 WineWPFFExample	12/08/2015 15:05	File folder	
 CLIPSNET.sln	11/08/2015 14:32	SLN File	14 KB
 instructions.pdf	13/08/2015 02:21	Adobe Acrobat D...	68 KB

Integrando CLIPS con .NET

Ejemplo

- **Namespace:**
 - `using CLIPSNET;`
- **Definición de la variable**
 - `private CLIPSNET.Environment clips;`
- **Principales Métodos / Ejecución del programa clips**
 - `clips = new CLIPSNET.Environment();`
 - `clips.loadFromResource("programa.clp");`
 - `clips.Reset();`
 - `String evalStr = "(assert (hecho (slot1 " + valor1 + ") (slot2 " + valor2 + ")))";`
 - `clips.Eval(evalStr);`
 - `clips.Run();`

Integrando CLIPS con .NET

Ejemplo

- **Obtención de los Resultados:**

- `String evalStr= "(find-all-facts ((?J diagnostico)) TRUE)";`
- `FactAddressValue fv = (FactAddressValue) ((MultifieldValue) clips.Eval(evalStr))[0];`
- `String s = fv.GetFactSlot("resultado").ToString();`

- **Obtención de los Resultados Múltiples:**

- `String evalStr= "(find-all-facts ((?J diagnostico)) TRUE)";`
- `foreach (FactAddressValue fv in clips.Eval(evalStr) as MultifieldValue)`
- `{`
- `String s = fv.GetFactSlot("resultado").ToString();`
- `....`

Integrando CLIPS con .NET/java

Ejemplo

- **Otros Métodos:**
 - `clips.AssertString("(attribute (name preferred-color) (value red))");`
 - `clips.load("programa.clp");`
 - `String Respuesta = clips.getInputBuffer();`

Desarrollo de Aplicación para TPO

Premisas a Considerar para la Evaluación

- **Funcionalidad mínima:**
 - Ingreso de Parámetros (manualmente a través de la UI o a partir de archivos u otras interfaces en caso que corresponda).
 - Ejecución del Sistema Experto.
 - Obtención de Resultados.
- **Funcionalidad deseada para la entrega final (ver si corresponde)**
 - Brindar posibilidad de modificar algún parámetro (sin volver a cargar todo) y re ejecutar el sistema experto.
 - Permitir almacenar elementos (pacientes, arqueros, etc), seleccionarlo y ejecutar el sistema experto.
 - Dependiendo el dominio de conocimiento, puede tener más sentido la ejecución de varios casos a la vez en lugar de caso por caso.
 - Log de ejecución/pre análisis

Desarrollo de Aplicación para TPO

Premisas a Considerar para la Evaluación

- **Usabilidad mínima**
 - Navegabilidad Clara (Evitar ventanas superpuestas).
 - Coherencia de la UI
 - Mínimo Look & Feel (Evitar ventanas grises)
- **Usabilidad deseada para la entrega Final**
 - Look & Feel / Estilos
 - Menues

Ejemplo Aplicación

Funcionalidad / Usabilidad Mínima

CABA - Sistema Experto Clima

 **Buenos Aires Ciudad** **Sistema Experto Clima**

Carga | Histórico | Log

Fecha: domingo , 14 de mayo de 2017 ▼

Turno: MAÑANA ▼

Temperatura: NOSE ▼

Viento Dirección: NOSE ▼

Viento Velocidad: NOSE ▼

Presión Atmosférica en Aumetro: NOSE ▼

Guardar Diagnosticar

Resultado



Bibliografía

CLIPS – Advance Programming Guide

Section 10 – Clips Java Native Interface

Section 12 - Microsoft .NET Integration