

# Práctica 3 Linda/Jess



ISBC

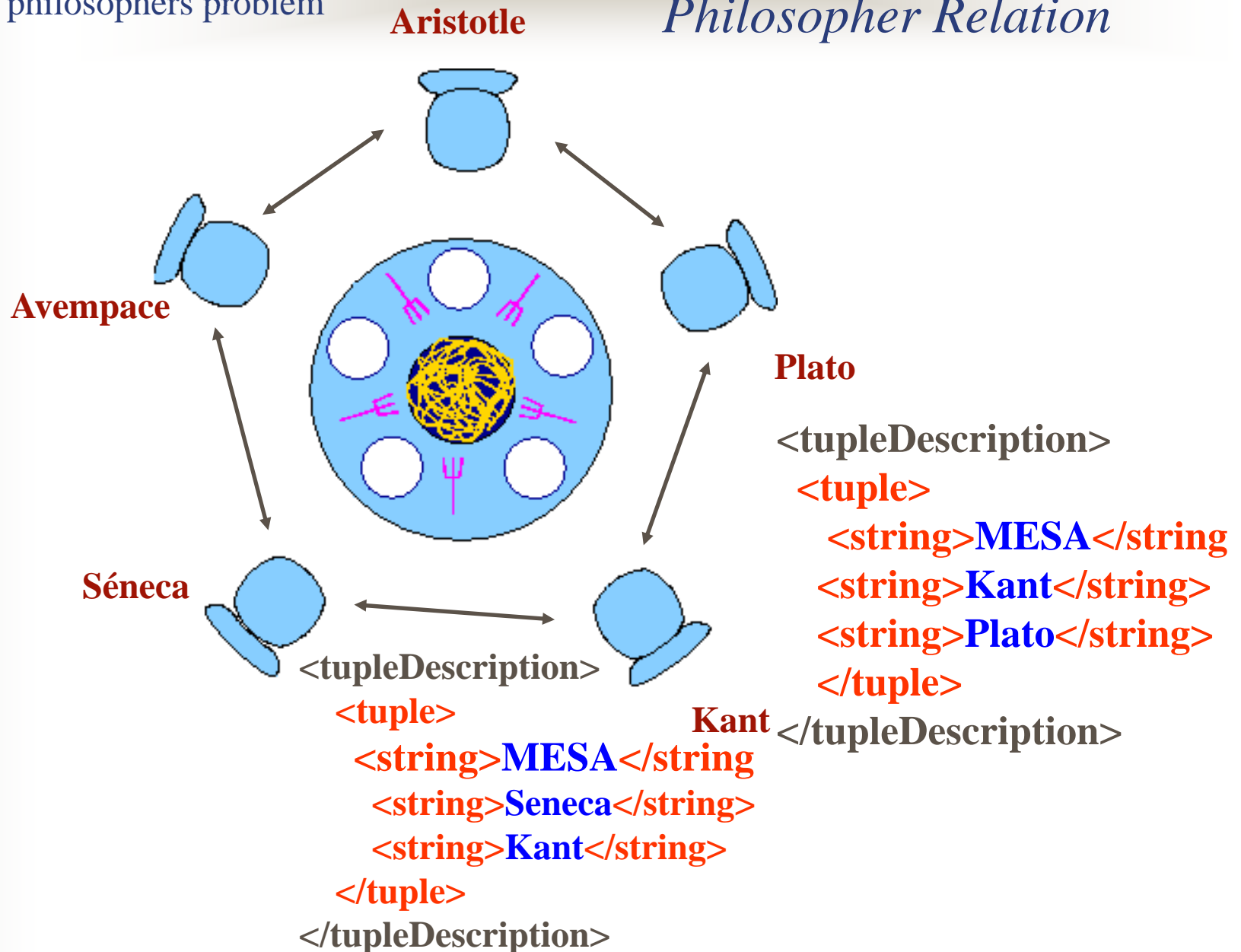
24/11/2008

# Classical Example

The dining philosophers problem

# Model Representation:

*Philosopher Relation*



# Problema

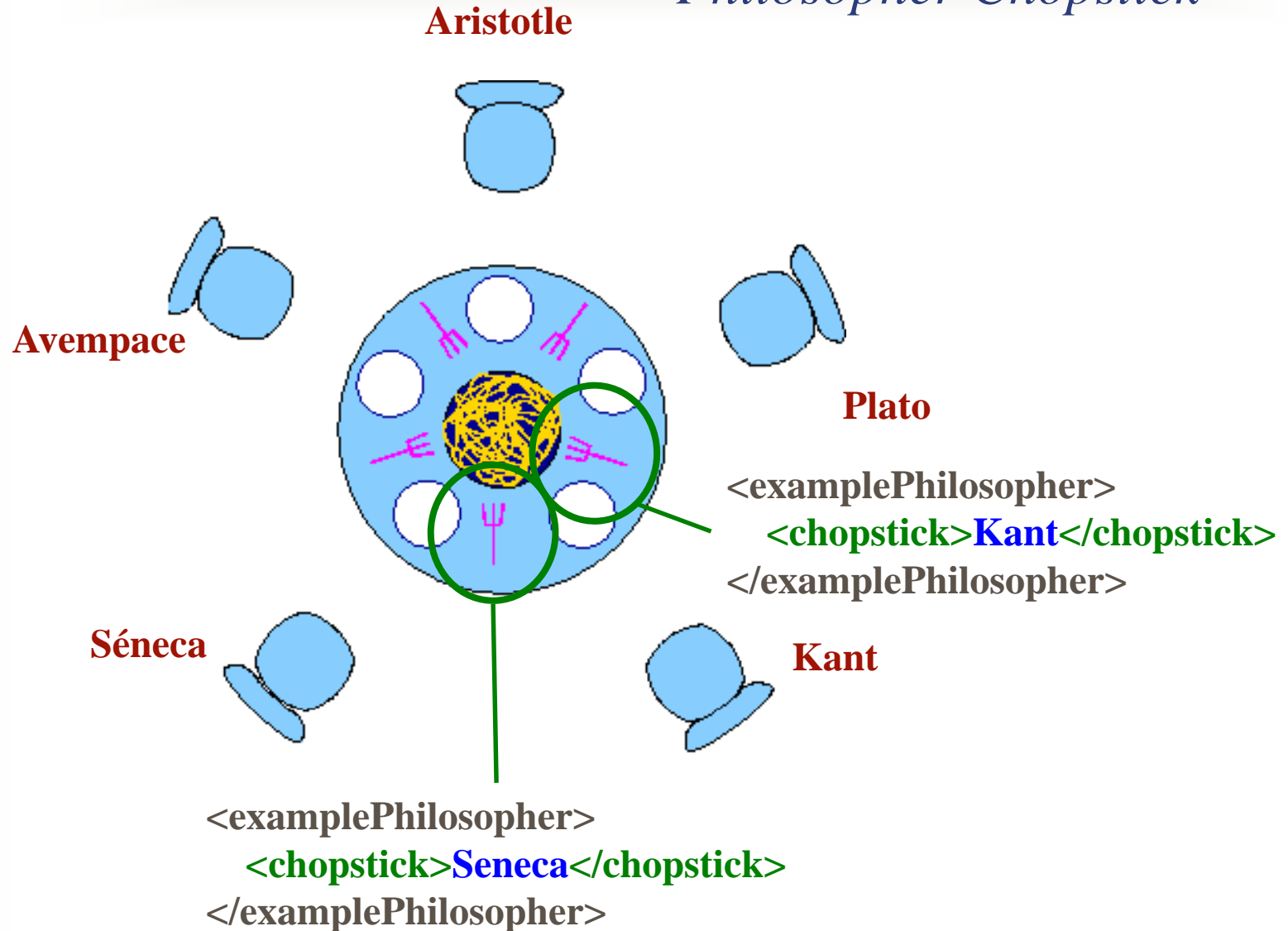
- Problema de los filósofos alrededor de una mesa
  - La mesa se representará en el espacio de tuplas con los comensales
  - El programa deberá introducirnos como comensales en la mesa.
  - Deberemos mostrar la distribución en la mesa
  - En la segunda parte del ejercicio habrá que realizar algún tipo de razonamiento para elegir el lugar en el que nos sentamos, y o bien habrá que obtener permiso de un coordinador que decide a quienes les da permiso para comer evitando bloqueos.

# Classical Example

The dining philosophers problem

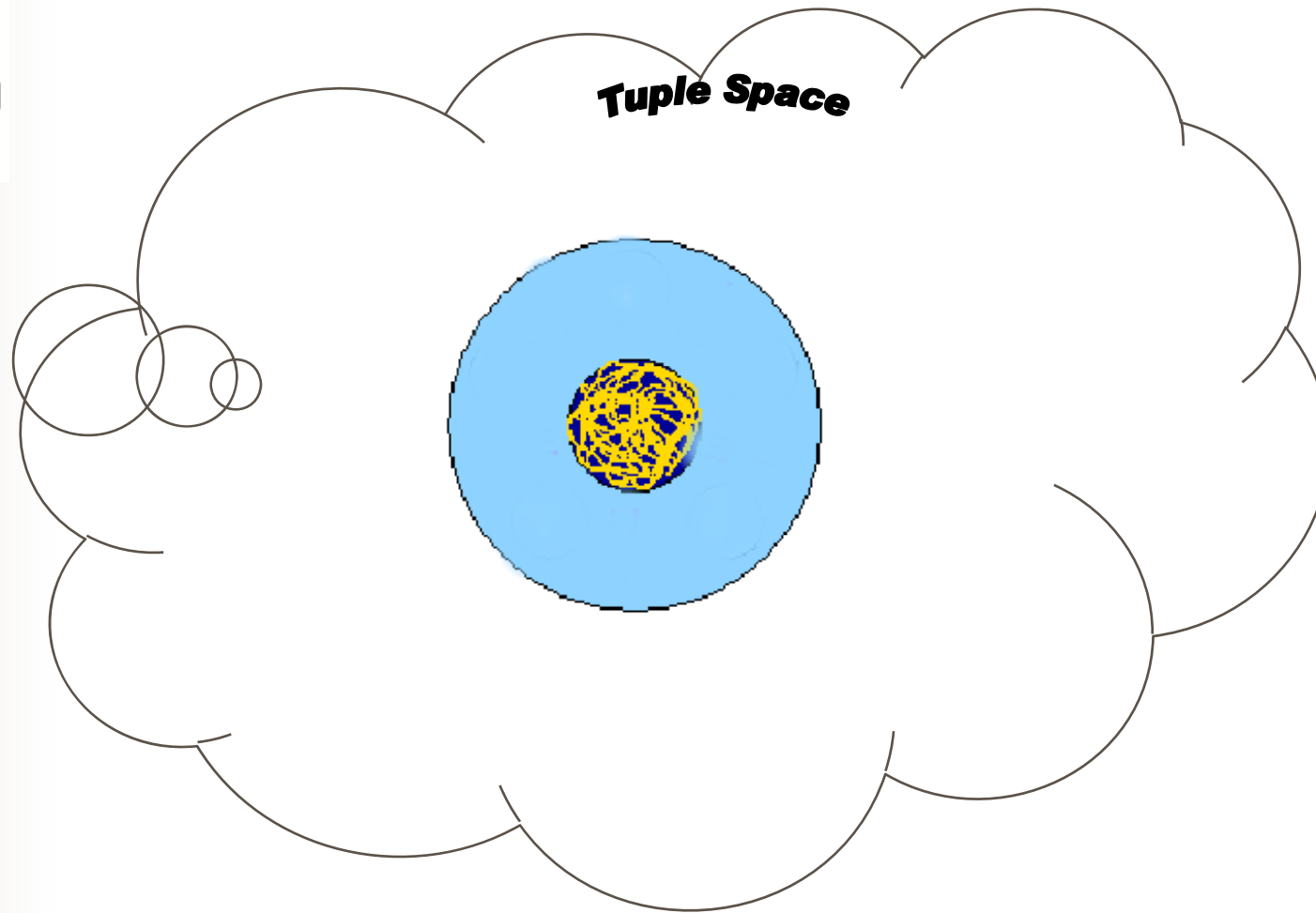
# Model Representation:

*Philosopher Chopstick*



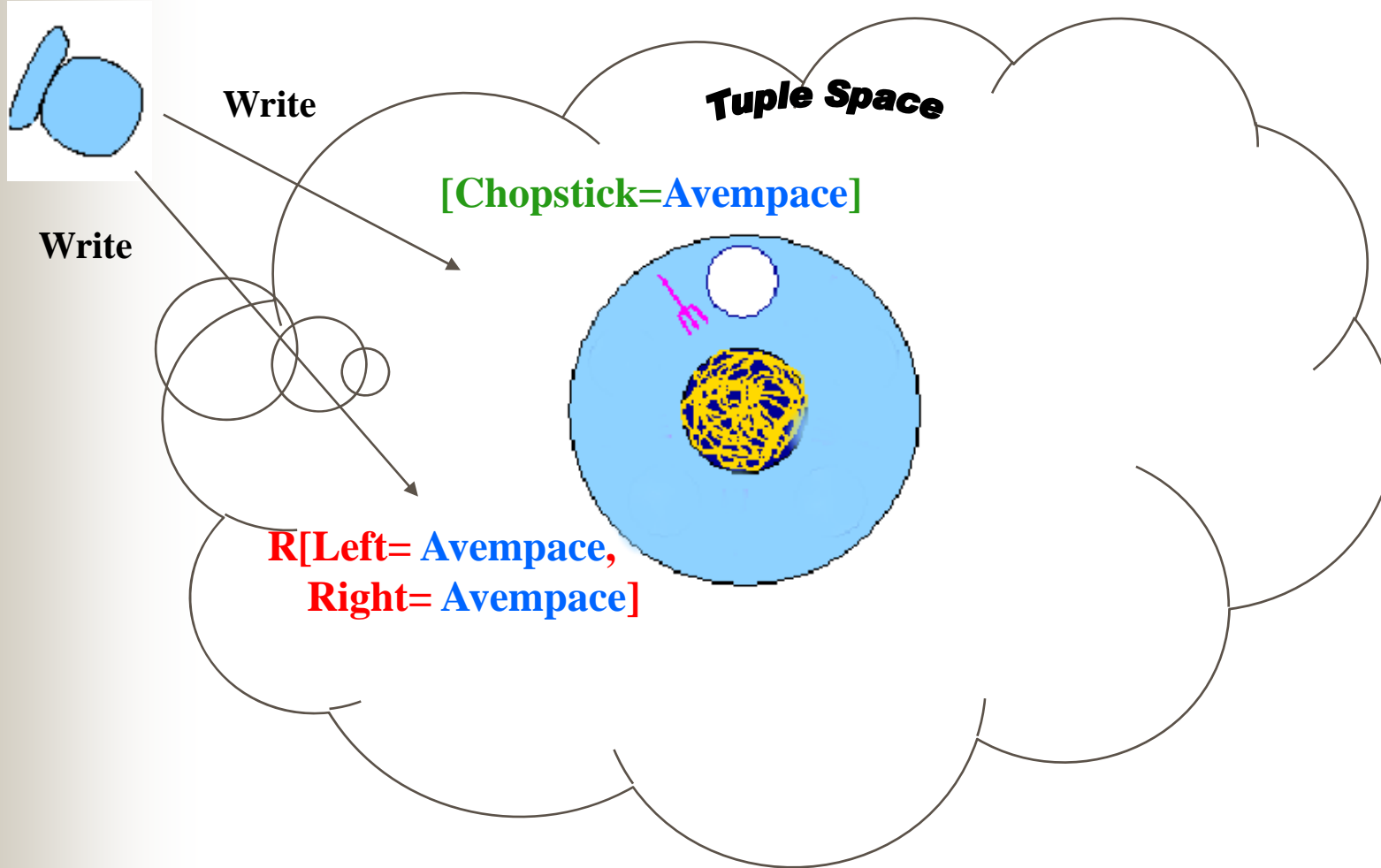
# Classical Example: Inserting the first philosopher

**Avempace**



# Classical Example: Inserting the first philosopher

**Avempace**



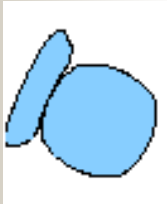






# Classical Example: Inserting another philosopher

**Avempace**



**Tuple Space**

[Chopstick=Avempace]

**R[Left= Avempace,  
Right= Avempace]**

**Take**



**Seneca**

```
http://bubu.cps.unizar.es:/CoordinationServlet?  
REQUEST=<?xml version="1.0"?>  
<CoordinationService>  
  <function>take</function>  
  <clienttype>java</clienttype>  
  <tuple>  
    <relation>  
      <Left> ? </Left>  
      <Right> ? </Right>  
    </relation>  
  </tuple>  
</CoordinationService>
```

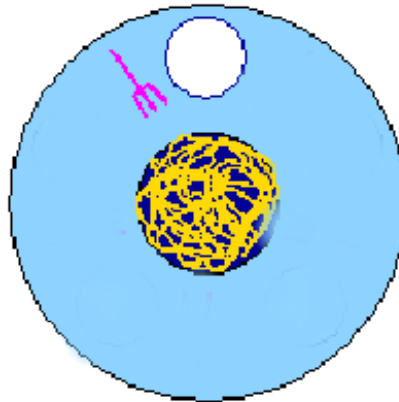
# Classical Example: Inserting another philosopher

**Avempace**



**Tuple Space**

[Chopstick=Avempace]



**Seneca**

**R[Left= Avempace,  
Right= Avempace]**

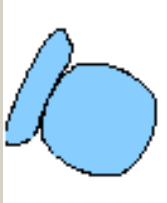


**R[Left= Avempace, Right= Seneca]**

**R[Left= Seneca, Right= Avempace]**

# Classical Example: Inserting another philosopher

**Avempace**



**Tuple Space**

[Chopstick=Avempace]

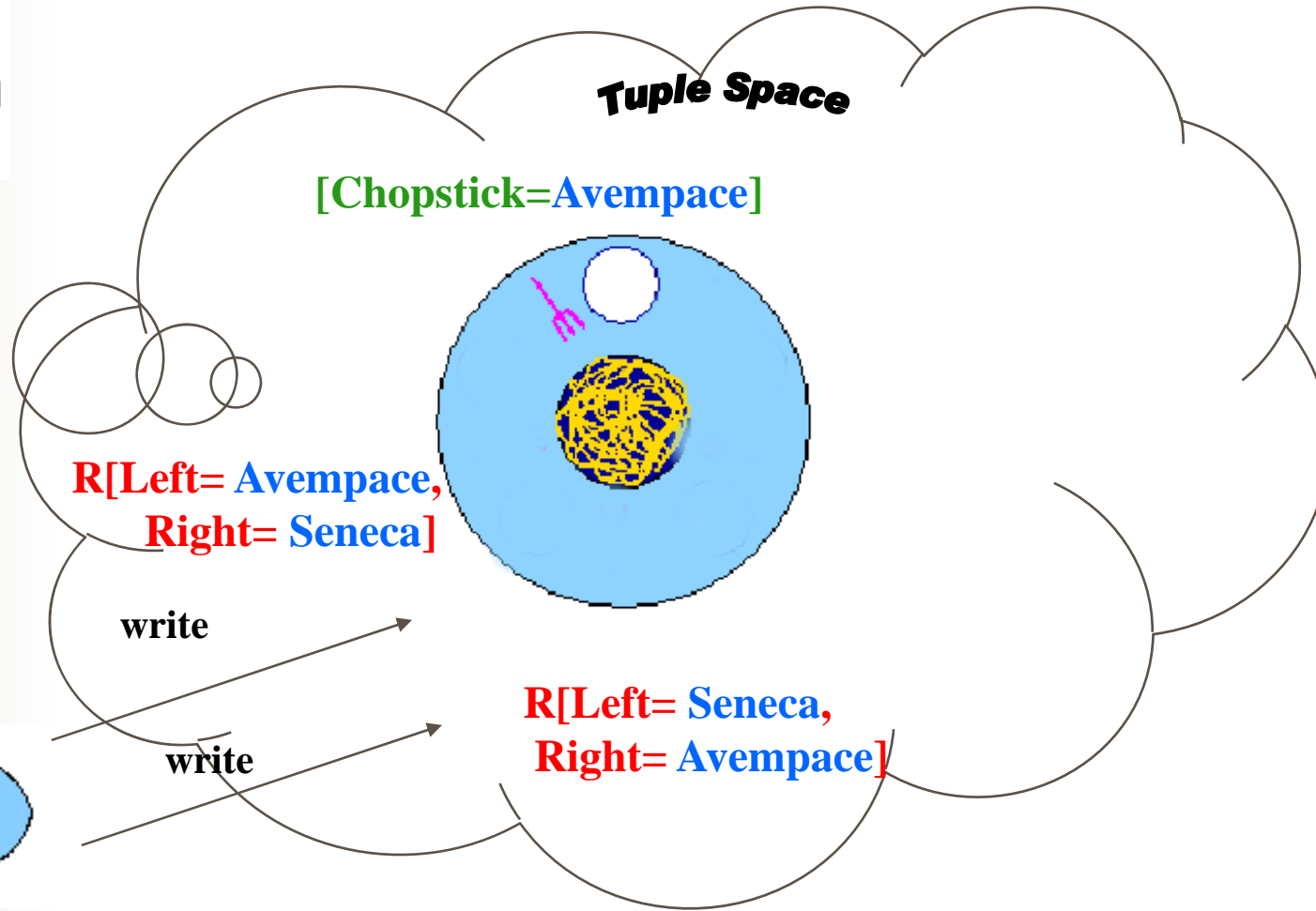
R[Left= Avempace,  
Right= Seneca]

write

write

R[Left= Seneca,  
Right= Avempace]

**Seneca**



# Classical Example: Inserting another philosopher

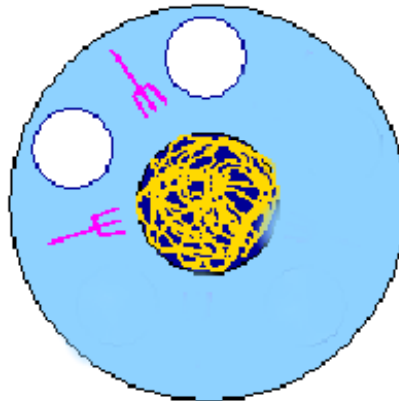
**Avempace**



**Tuple Space**

[Chopstick=Avempace]

R[Left= Avempace,  
Right= Seneca]



[Chopstick=Seneca]

R[Left= Seneca,  
Right= Avempace]

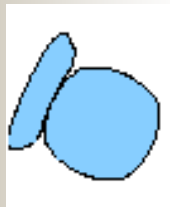
write



**Seneca**

# Classical Example: Before eating

**Avempace**



**Tuple Space**

[Chopstick=Avempace]

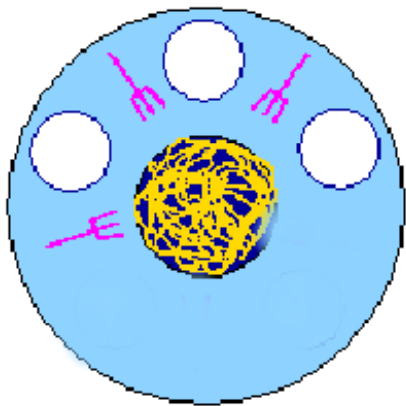
R[Left= Plato,  
Right= Avempace]

**Plato**



[Chopstick=Plato]

R[Left= Avempace,  
Right= Seneca]



[Chopstick=Seneca]

R[Left= Seneca,  
Right= Plato]

**Seneca**



# Classical Example: Eating

**Avempace**



Take[R[Left= Avempace, Right= ? ]]

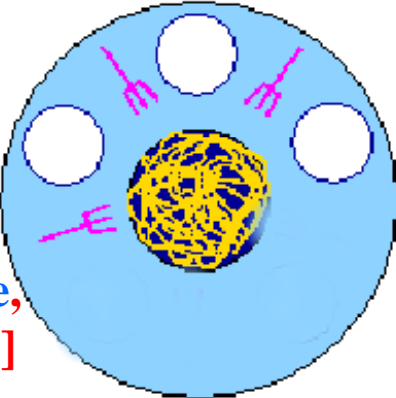
**Tuple Space**



**Plato**

[Chopstick=Avempace]

R[Left= Plato,  
Right= Avempace]



[Chopstick=Plato]

R[Left= Avempace,  
Right= Seneca]

[Chopstick=Seneca]

R[Left= Seneca,  
Right= Plato]



**Seneca**

Take[R[Left= Seneca, Right= ? ]]

# Classical Example: Eating

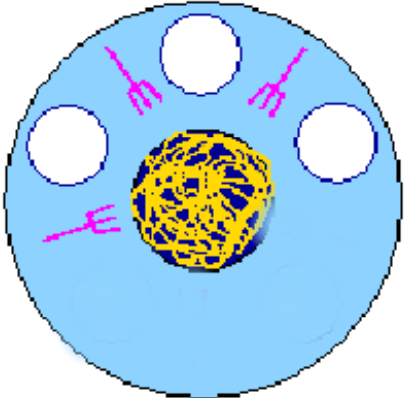
**Avempace**



**R[Left= Avempace,  
Right= Seneca]**

**Tuple Space**

**[Chopstick=Avempace]**



**R[Left= Plato,  
Right= Avempace]**

**Plato**



**[Chopstick=Plato]**

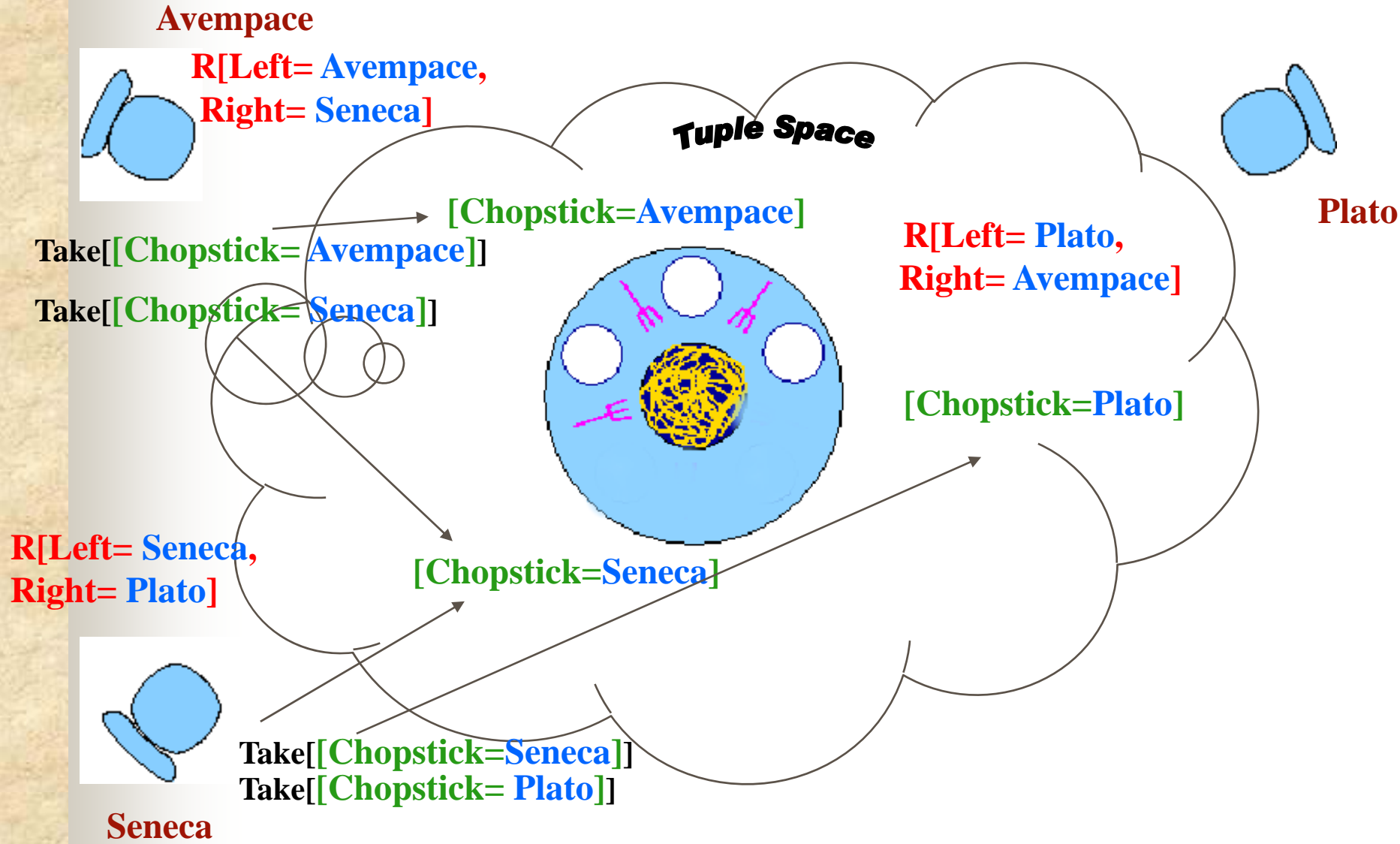
**R[Left= Seneca,  
Right= Plato ]**

**[Chopstick=Seneca]**



**Seneca**

# Classical Example: Eating





# Classical Example: Eating

[Chopstick=Avempace]  
Avempace



R[Left= Avempace,  
Right= Seneca]

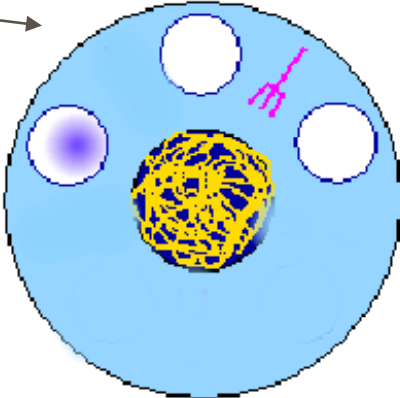
Take[[Chopstick=Seneca]]

**Tuple Space**



**Plato**

R[Left= Plato,  
Right= Avempace]



R[Left= Seneca,  
Right= Plato ]



**Seneca**

[Chopstick=Seneca]  
[Chopstick=Plato]

# Linda Server en UZ

- Cliente Web para probar insertar, sacar o leer tuplas
  - <http://luna1.cps.unizar.es:8080/misServlets/RLinda.html>

RLinda Operations - Windows Internet Explorer

http://luna1.cps.unizar.es:8080/misServlets/RLinda.html

Google Ir Marcadores 474 bloqueados Corrector ortográfico Configuración

Gma... R... http... Goo... Y:\a... Rep... WS... Aso... Página Herramientas

[in/out/rd](#) [Ver repositorio](#) [Estadísticas](#)

<b>Tupla en lenguaje natural:</b>	<input type="text"/>
<b>Tupla en XML:</b>	<input type="text"/>
<input type="button" value="Insertar"/>	

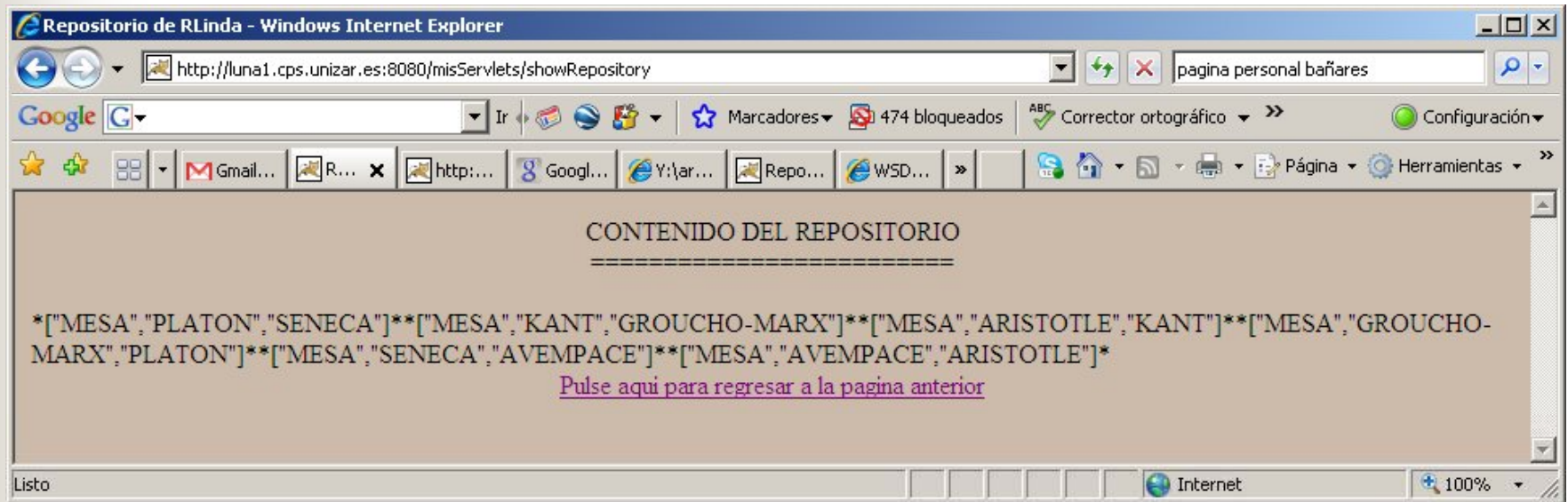
<b>Tupla en lenguaje natural:</b>	<input type="text"/>
<b>Tupla en XML:</b>	<input type="text"/>
<input type="button" value="Tomar"/>	

<b>Tupla en lenguaje natural:</b>	<input type="text"/>
<b>Tupla en XML:</b>	<input type="text"/>
<input type="button" value="Leer"/>	

Internet 100%

# Repositorio



# Ejemplos Tuplas/Templates

## Ejemplo de OUT:

**Lenguaje Tupla:** [[“esto”,”es”,[1],”prueba”]]

**Lenguaje XML:**

```
<tupleDescription><tuple><tuple><string>esto</string><string>es</string>
<tuple>1</tuple><string>prueba</string></tuple></tuple></tupleDescription>
```

## Ejemplo de IN/RD:

**Lenguaje Tupla:** [[“?”,”es”,[“?”],”prueba”]]

**Lenguaje XML:**

```
<tupleDescription><tuple><tuple><wildcard /> <string>es</string>
<tuple> <wildcard /></tuple> <string>prueba</string></tuple></tuple></tupleDescription>
```

# Cientes SOAP

- <http://luna1.cps.unizar.es:8080/axis/services/RLindaWS?wsdl>
- Generación Clientes
  - Lisp
    - (require :soap)
    - (use-package :net.xmp.soap)
    - (decode-wsdl-namespaces :file "RLindaW.wsdl")
    - (setf \*wsdl\* (decode-wsdl-file "RLindaW.wsdl"))
    - make-client-interface \*wsdl\* 0 "RLClient.cl")

# Ejemplo Clientes Lisp

```
(require :soap)
(require :pxml)
(load "RLClient.cl")
;; IN = TAKE, OUT = WR, RD = RD.
CG-USER(19): (common-lisp-user::OUT :in0
  "<tupleDescription><tuple><string>MESA</string><string>PLATON</string><string>SENECA</string></tuple></tupleDescription>")
(WSDL-2::|RLindaOUTResponse|
 (:|RLindaOUTReturn|

  "<tupleDescription><tuple><string>MESA</string><string>PLATON</string><string>SENECA</string></tuple></tupleDescription>"))
NIL
#<SOAP-ASERVE-CLIENT-STRING-IN-OUT-CONNECTOR @ #x2104f8a2>
```

# Cliente Lisp

```
CG-USER(20): (common-lisp-user::RD :in0
  "<tupleDescription><tuple><string>MESA</string><string>?</string>
  <string>?</string></tuple></tupleDescription>")
```

```
(WSDL-2::|RLindaRDResponse|
 (:|RLindaRDReturn| "<?xml version=\"1.0\" encoding=\"UTF-8\"?>
  <tupleDescription> <tuple> <string>MESA</string>
  <string>AVEMPACE</string> <string>ARISTOTLE</string> </tuple>
  </tupleDescription> "))
```

NIL

```
#<SOAP-ASERVE-CLIENT-STRING-IN-OUT-CONNECTOR @ #x217c4e62>
```



# Cliente Lisp

CL-USER(178): (RD :in0

```
"<tupleDescription><tuple><string>MESA</string><wildcard />
<wildcard /></tuple></tupleDescription>")
```

(WSDL-2::|RLindaRDResponse|

```
(:|RLindaRDReturn| "<?xml version=\"1.0\" encoding=\"UTF-8\"?>
<tupleDescription> <tuple> <string>MESA</string>
<string>KANT</string> <string>GROUCHO-MARX</string> </tuple>
</tupleDescription> ")
```

NIL

#<SOAP-ASERVE-CLIENT-STRING-IN-OUT-CONNECTOR @ #x22224ee2>

# Cliente Lisp

```
(defun RD-LST (template)
  (MAPCAR #'SECOND
    (remove-if-not #'listp
      (CDDAR
        (CDDADR
          (net.xml.parser:PARSE-XML
            (car
              (cdaDR (MULTIPLE-VALUE-BIND (X Y)
                (RD :in0 template) X))))))))))
```

CL-USER(179):

RD-LST hace un RD con un template en XML y devuelve una lista

CL-USER(182): (RD-LST

```
"<tupleDescription><tuple><string>MESA</string><string>?</string>
<string>?</string></tuple></tupleDescription>")
```

```
("MESA" "KANT" "GROUCHO-MARX")
```

# Auxiliares

CL-USER(181): (**list2xml** '("MESA" 1 ("a" "b") "c"))

```
"<tupleDescription><tuple><string>MESA</string><integer>1</integer><tuple><string>a</string><string>b</string></tuple><string>c</string></tuple></tupleDescription>"
```

```
(defun 2xml (lista &optional cadena)
```

```
  (cond
```

```
    ((stringp (car lista))
```

```
     (2xml (cdr lista) (concatenate 'string cadena "<string>" (car lista) "</string>")))
```

```
    ((integerp (car lista))
```

```
     (2xml (cdr lista) (concatenate 'string cadena "<integer>" (format nil "~A" (car lista)) "</integer>")))
```

```
    ((eq '? (car lista))
```

```
     (2xml (cdr lista) (concatenate 'string cadena "<string>" "?" "</string>")))
```

```
    ((and (listp (car lista)) (not (endp lista)))
```

```
     (2xml (cdr lista) (concatenate 'string cadena "<tuple>" (2xml (car lista)) "</tuple>")))
```

```
    ((endp lista) cadena)))
```

```
(defun list2xml (lista)
```

```
  (concatenate 'string "<tupleDescription><tuple>" (2xml lista) "</tuple></tupleDescription>"))
```

# Inicia mesa

```
(defun inicia-mesa ()
  (OUT :in0
    "<tupleDescription><tuple><string>MESA</string><string>PLATON</string><string>SENECA</string></tuple>
    </tupleDescription>")
  (OUT :in0
    "<tupleDescription><tuple><string>MESA</string><string>SENECA</string><string>AVEMPACE</string></tu
    ple></tupleDescription>")
  (OUT :in0
    "<tupleDescription><tuple><string>MESA</string><string>AVEMPACE</string><string>ARISTOTLE</string>
    </tuple></tupleDescription>")
  (OUT :in0
    "<tupleDescription><tuple><string>MESA</string><string>ARISTOTLE</string><string>KANT</string></tuple
    ></tupleDescription>")
  (OUT :in0 "<tupleDescription><tuple><string>MESA</string><string>KANT</string><string>GROUCHO-
    MARX</string></tuple></tupleDescription>")
  (OUT :in0 "<tupleDescription><tuple><string>MESA</string><string>GROUCHO-
    MARX</string><string>PLATON</string></tuple></tupleDescription>")
)
```

# Lisp example Estilo Rest (old)

```
CG-USER(14):(net.aseve.client:do-http-request
http://bubu.cps.unizar.es:8080/CoordinationServlet :method :post
:content-type "application/x-www-form-urlencoded"
:query '(("REQUEST" .
"<?xml version=\"1.0\"?>
<CoordinationService>
  <function>write</function>
  <tuple>
    <examplePhilosopher><chopstick>Kant</chopstick></examplePhilosopher>
  </tuple>
</CoordinationService>"))
" "
200
NIL
#<URI http://bubu.cps.unizar.es:8080/CoordinationServlet>
```

# Cliente Java

//

Para compilar y ejecutar este código Java **es necesario incluir las librerías de axis1 en el classpath**. Disponibles en la URL [http://www.apache.org/dyn/closer.cgi/ws/axis/1\\_4](http://www.apache.org/dyn/closer.cgi/ws/axis/1_4)

## Ejemplo de cliente que accede al metodo OUT

```
import org.apache.axis.client.Call;
import org.apache.axis.client.Service;
import javax.xml.namespace.QName;

public class client {
public static void main(String [] args) {
    try {
        String endpoint = "http://luna1.cps.unizar.es:8080/axis/services/RLindaWS";
        Service service = new Service();
        Call call = (Call) service.createCall();
        call.setTargetEndpointAddress( new java.net.URL(endpoint) );
        call.setOperationName(new QName("http://soapinterop.org/", "RLindaOUT"));
        String value = (String) call.invoke( new Object[] { "[\esto\",[1],\prueba\"]" } );
        System.out.println(value);
    } catch (Exception e) {
        System.err.println(e.toString());
    }
}
}
```

# Cliente Java

// Para compilar y ejecutar este código Java es necesario incluir las librerías de axis1 en el classpath.  
// Disponibles en la URL [http://www.apache.org/dyn/closer.cgi/ws/axis/1\\_4](http://www.apache.org/dyn/closer.cgi/ws/axis/1_4)

## Ejemplo de cliente que accede al metodo OUT

```
import org.apache.axis.client.Call;
import org.apache.axis.client.Service;
import javax.xml.namespace.QName;

public class client {
public static void main(String [] args) {
    try {
        String endpoint = "http://luna1.cps.unizar.es:8080/axis/services/RLindaWS";
        Service service = new Service();
        Call call = (Call) service.createCall();
        call.setTargetEndpointAddress( new java.net.URL(endpoint) );
        call.setOperationName(new QName("http://soapinterop.org/", "RLindaOUT"));
        String value = (String) call.invoke( new Object[] { "[\"esto\",[1],\"prueba\"]" }
);
        System.out.println(value);
    } catch (Exception e) {
        System.err.println(e.toString());
    }
}
}
```

# Cliente Java

## Ejemplo de cliente que accede al metodo IN

```
import org.apache.axis.client.Call;
import org.apache.axis.client.Service;
import javax.xml.namespace.QName;

public class client {
public static void main(String [] args) {
    try {
        String endpoint = "http://luna1.cps.unizar.es:8080/axis/services/RLindaWS";
        Service service = new Service();
        Call call = (Call) service.createCall();
        call.setTargetEndpointAddress( new java.net.URL(endpoint) );
        call.setOperationName(new QName("http://soapinterop.org/", "RLindaIN"));
        String value = (String) call.invoke( new Object[] { ["esto\","?"]} );
        System.out.println(value);
    } catch (Exception e) {
        System.err.println(e.toString());
    }
}
}
```