SML Tutorial

Soar Workshop 33 – Nate Derbinsky

While waiting…

1. Make sure you have internet access
2. Download Soar Tutorial package

   web.eecs.umich.edu/~soar/workshop_tutorial
3. Download Eclipse (with at least Java)

   www.eclipse.org
4. Download tutorial support files

   web.eecs.umich.edu/~nlderbin/workshop33

Agenda

- Big picture
- System setup + Hello Soar
- Basic usage
- Additional resources

Big Picture: Soar Markup Language

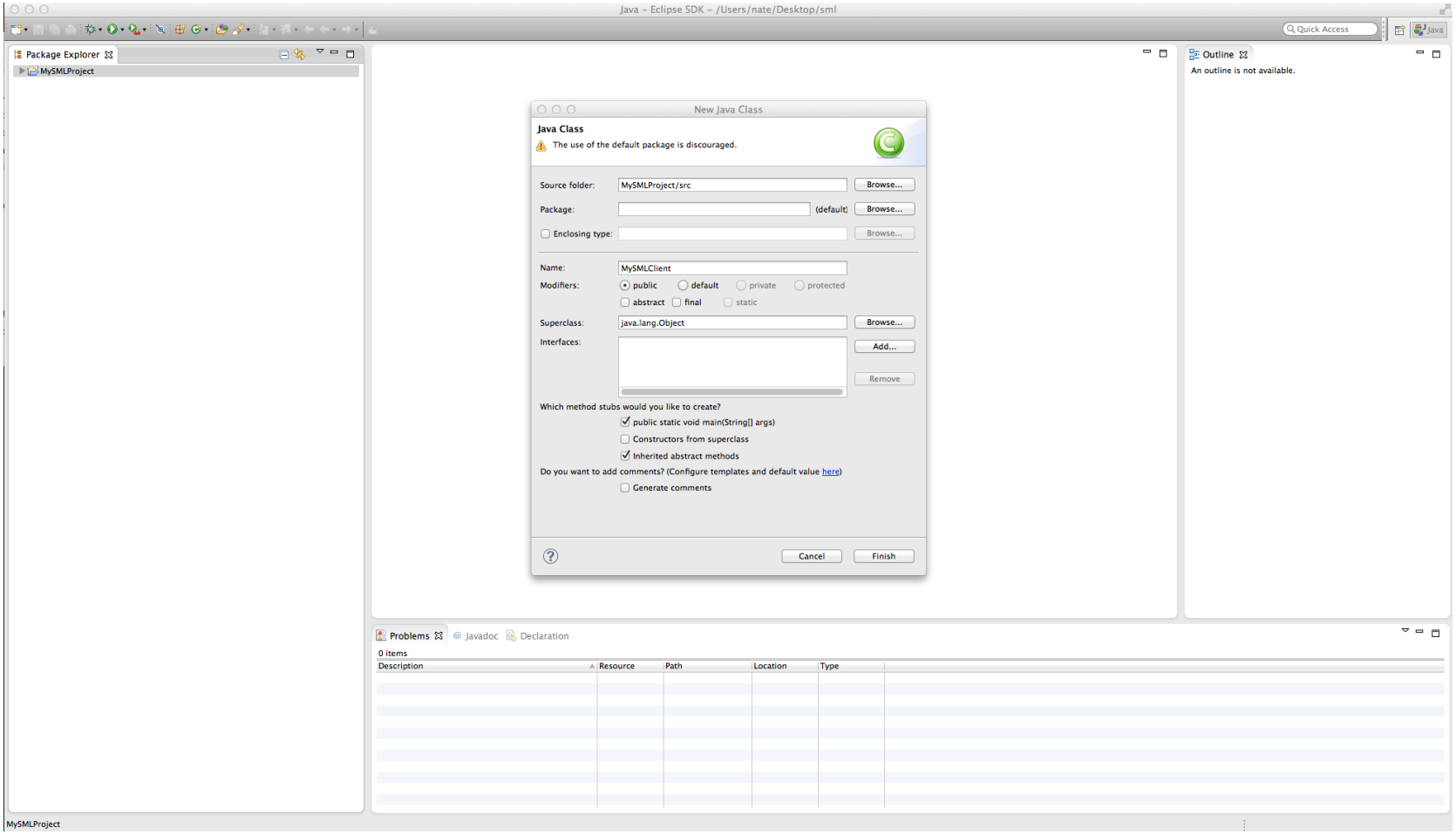
- SML provides a programmatic interface into Soar based around sending and receiving commands packaged as XML packets. Used for…
  - Environments
  - Debuggers
  - Automated experimentation

- Written in C++
- Exported, via SWIG (swig.org) to…
  - Python
  - Java

System Setup (1)

1. Open Eclipse
2. New Java Project
   - Name="MySMLProject"
   - Finish
3. New Class
   - Name="MySMLClient"
   - Check: "public static void main…"
   - Finish

Status


Java – Eclipse SDK – /Users/nate/Desktop/sml
Quick Access
Java
Package Explorer
MySMLProject
Outline
An outline is not available.
New Java Class
Java Class
The use of the default package is discouraged.
Source folder:
MySMLProject/src
Browse...
Package:
(default)
Enclosing type:
Name:
MySMLClient
Modifiers:
public
default
private
protected
abstract
final
static
Superclass:
java.lang.Object
Interfaces:
Add...
Remove
Which method stubs would you like to create?
public static void main(String[] args)
Constructors from superclass
Inherited abstract methods
Do you want to add comments? (Configure templates and default value here)
Generate comments
Cancel
Finish
Problems
Javadoc
Declaration
0 items
Description
Resource
Path
Location
Type
MySMLProject

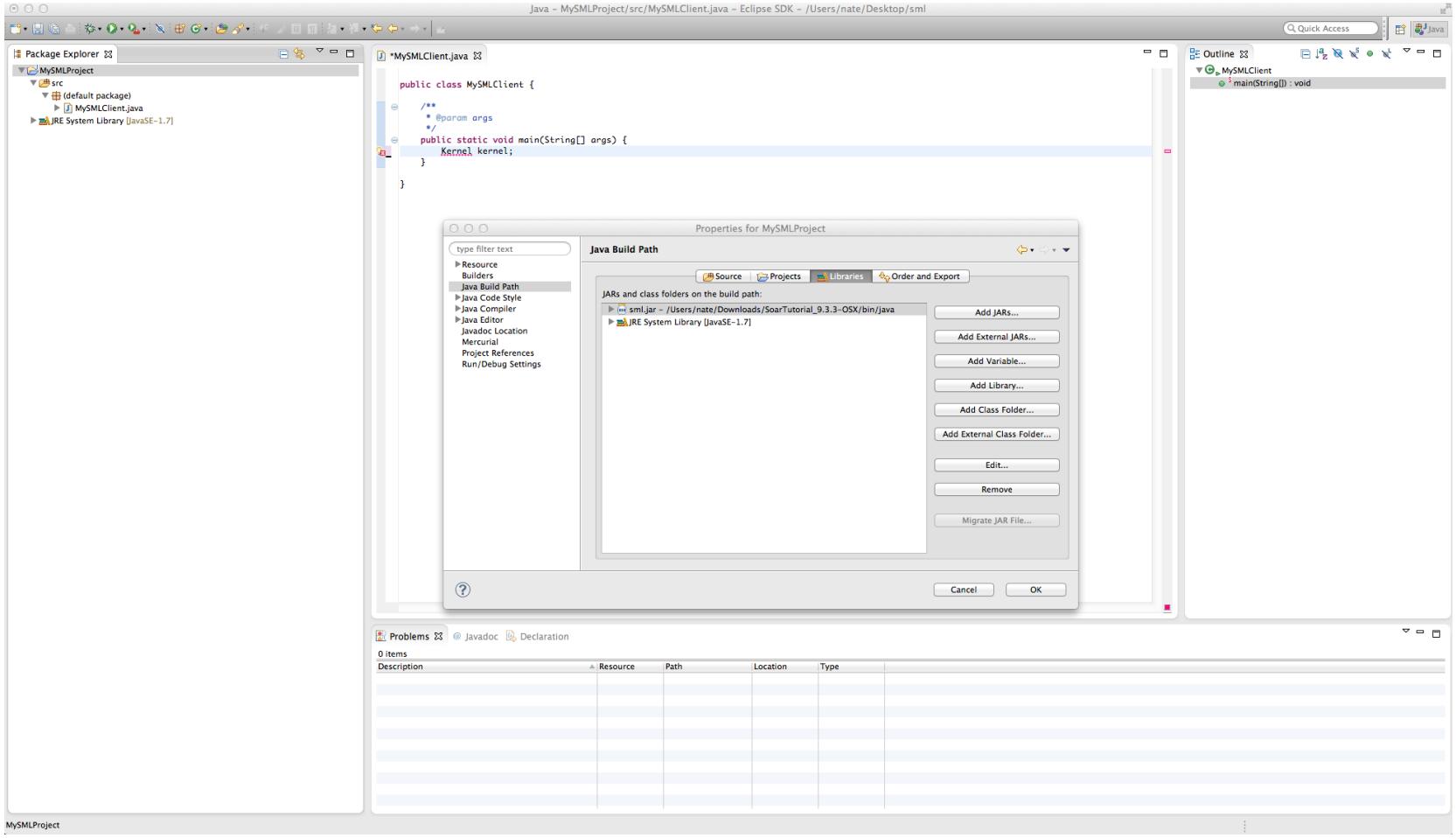
System Setup (2)

4. Inside main…

```
Kernel kernel;
```

5. Add `sml.jar` to class path
   - Right click project -> Properties
   - Java Build Path
   - Libraries -> Add External Jar
   - Locate `sml.jar` in `bin/java`, Open, OK
6. Hover over “Kernel” (red underline)
   - Click “Import ‘Kernel’ (sml)”

Status


Java – MySMLProject/src/MySMLClient.java – Eclipse SDK – /Users/nate/Desktop/sml
Package Explorer
MySMLProject
src
(default package)
MySMLClient.java
JRE System Library [JavaSE-1.7]
*MySMLClient.java
public class MySMLClient {
/**
* @param args
*/
public static void main(String[] args) {
Kernel kernel;
}
}
Outline
MySMLClient
main(String[]) : void
Properties for MySMLProject
type filter text
Resource
Builders
Java Build Path
Java Code Style
Java Compiler
Java Editor
Javadoc Location
Mercurial
Project References
Run/Debug Settings
Java Build Path
Source
Projects
Libraries
Order and Export
JARs and class folders on the build path:
sml.jar – /Users/nate/Downloads/SoarTutorial_9.3.3-OSX/bin/java
JRE System Library [JavaSE-1.7]
Add JARs...
Add External JARs...
Add Variable...
Add Library...
Add Class Folder...
Add External Class Folder...
Edit...
Remove
Migrate JAR File...
Cancel
OK
Problems
Javadoc
Declaration
0 items
Description
Resource
Path
Location
Type
MySMLProject

System Setup (3)

7. Finish main …

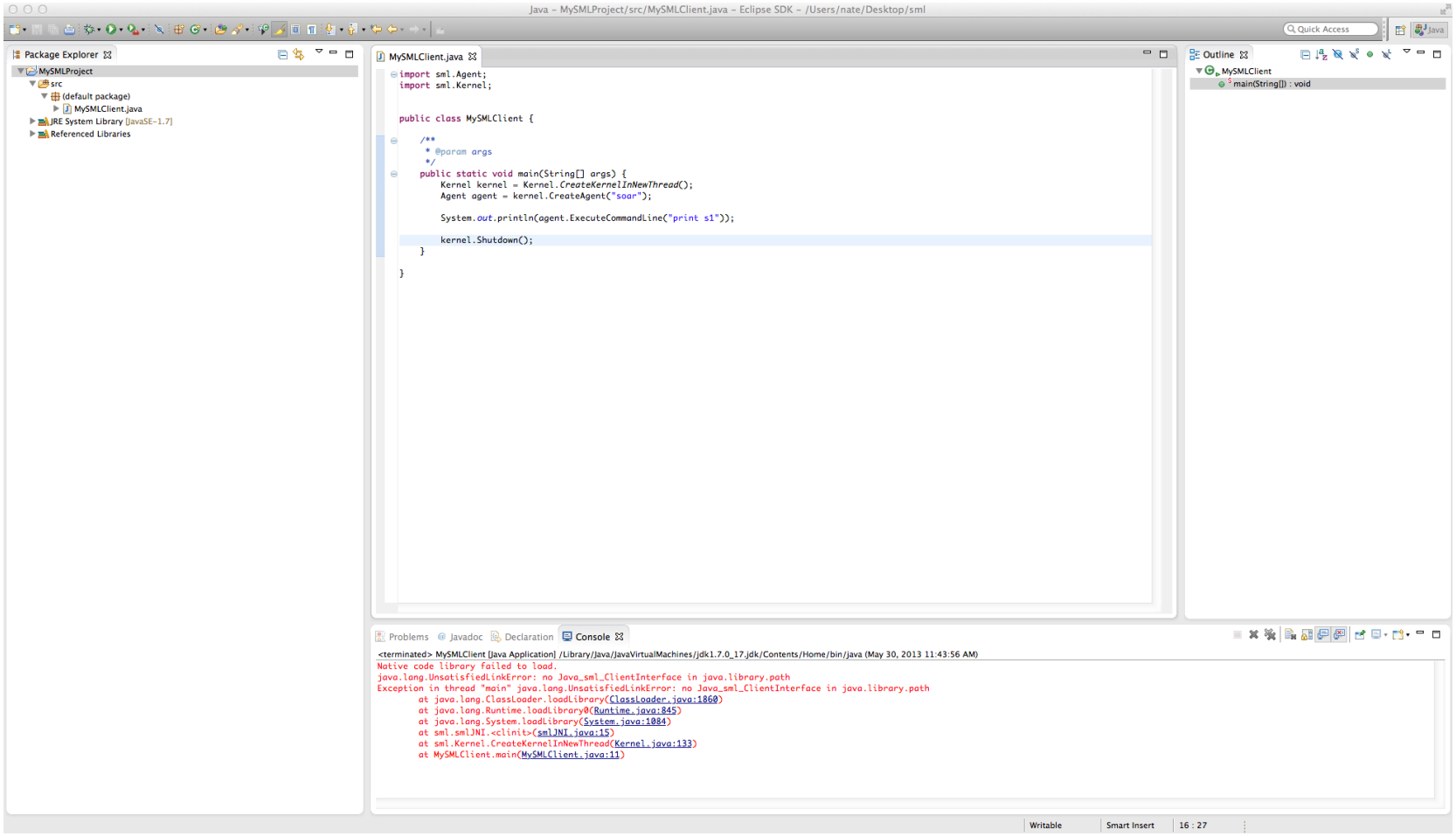
```
Kernel kernel = Kernel.CreateKernelInNewThread();
Agent agent = kernel.CreateAgent("soar");

System.out.println(agent.ExecuteCommandLine("print s1"));

kernel.Shutdown();
```

8. Run menu -> Run

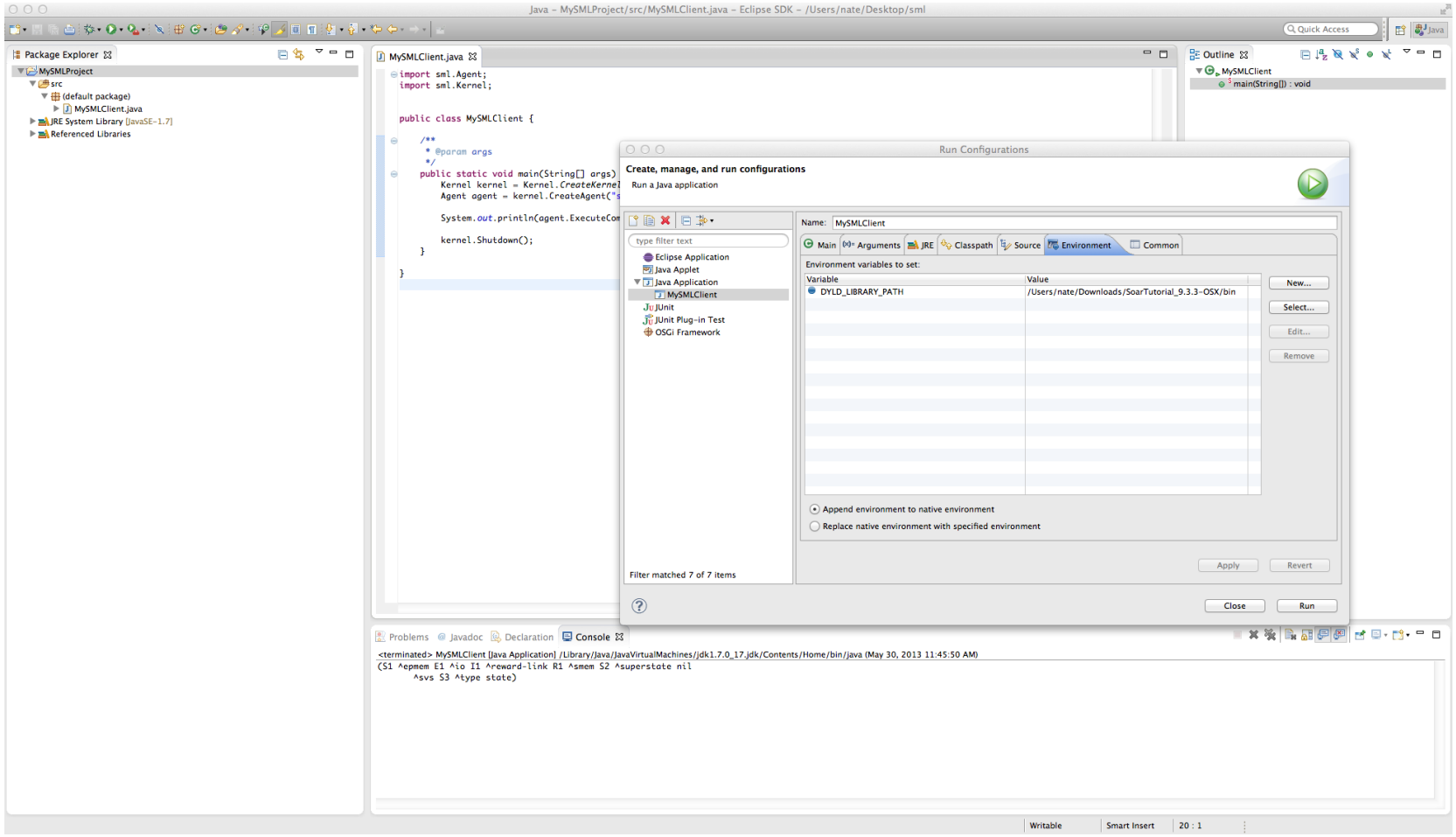
Status


Java – MySMLProject/src/MySMLClient.java – Eclipse SDK – /Users/nate/Desktop/sml
Package Explorer
MySMLProject
src
(default package)
MySMLClient.java
JRE System Library [JavaSE-1.7]
Referenced Libraries
MySMLClient.java
import sml.Agent;
import sml.Kernel;
public class MySMLClient {
/**
* @param args
*/
public static void main(String[] args) {
Kernel kernel = Kernel.CreateKernelInNewThread();
Agent agent = kernel.CreateAgent("soar");
System.out.println(agent.ExecuteCommandLine("print s1"));
kernel.Shutdown();
}
}
Outline
MySMLClient
main(String[]) : void
Problems
Javadoc
Declaration
Console
<terminated> MySMLClient [Java Application] /Library/Java/JavaVirtualMachines/jdk1.7.0_17.jdk/Contents/Home/bin/java (May 30, 2013 11:43:56 AM)
Native code library failed to load.
java.lang.UnsatisfiedLinkError: no Java_sml_ClientInterface in java.library.path
Exception in thread "main" java.lang.UnsatisfiedLinkError: no Java_sml_ClientInterface in java.library.path
at java.lang.ClassLoader.loadLibrary(ClassLoader.java:1860)
at java.lang.Runtime.loadLibrary0(Runtime.java:845)
at java.lang.System.loadLibrary(System.java:1084)
at sml.smlJNI.<clinit>(smlJNI.java:15)
at sml.Kernel.CreateKernelInNewThread(Kernel.java:133)
at MySMLClient.main(MySMLClient.java:11)
Writable
Smart Insert
16 : 27

System Setup (4)

9. Run menu -> Run Configurations
   - Environment tab
   - New
     - Name=
       - Mac: DYLD_LIBRARY_PATH
       - Linux: LD_LIBRARY_PATH
       - Windows: PATH
     - Value="/path/to/soar/release/bin" (no quotes)
     - Run

Status


Java – MySMLProject/src/MySMLClient.java – Eclipse SDK – /Users/nate/Desktop/sml
Package Explorer
MySMLProject
src
(default package)
MySMLClient.java
JRE System Library [JavaSE-1.7]
Referenced Libraries
MySMLClient.java
import sml.Agent;
import sml.Kernel;
public class MySMLClient {
/**
* @param args
*/
public static void main(String[] args)
Kernel kernel = Kernel.CreateKernel
Agent agent = kernel.CreateAgent("s
System.out.println(agent.ExecuteCom
kernel.Shutdown();
}
}
Outline
MySMLClient
main(String[]) : void
Run Configurations
Create, manage, and run configurations
Run a Java application
type filter text
Eclipse Application
Java Applet
Java Application
MySMLClient
JUnit
JUnit Plug-in Test
OSGi Framework
Filter matched 7 of 7 items
Name: MySMLClient
Main
Arguments
JRE
Classpath
Source
Environment
Common
Environment variables to set:
Variable
Value
DYLD_LIBRARY_PATH
/Users/nate/Downloads/SoarTutorial_9.3.3-OSX/bin
New...
Select...
Edit...
Remove
Append environment to native environment
Replace native environment with specified environment
Apply
Revert
Close
Run
Problems
Javadoc
Declaration
Console
<terminated> MySMLClient [Java Application] /Library/Java/JavaVirtualMachines/jdk1.7.0_17.jdk/Contents/Home/bin/java (May 30, 2013 11:45:50 AM)
(S1 ^epmem E1 ^io I1 ^reward-link R1 ^smem S2 ^superstate nil
^svs S3 ^type state)
Writable
Smart Insert
20 : 1

Basic Usage

Part 1: Automation
- Command execution
- Loading rules
- Synchronous run control
- Capturing trace output

Part 2: Basic IO
- Managing WMEs on input-link
- Read output-link WMEs + feedback

Part 3: Event-Driven Environment
- Output handler

Command Execution

Syntax

"result"=agent.ExecuteCommandLine("command");

Try

- "stats"
- "epmem --stats"
- "sp {test (state <s> ^superstate nil) --> (<s> ^foo bar)}"
- "print test"

Loading Rules

Syntax

True/False = Agent.LoadProductions("location");

Try

1. Load: `goodbye.soar`
2. Execute: "print goodbye"

Synchronous Run Control

Syntax

- Agent.RunSelf(numberSteps,stepSize = Decision);
- Agent.RunSelfForever();
- Agent.RunSelfTilOutput();
- Agent.ExecuteCommandLine("run…");

Try

1. Load: `goodbye.soar`
2. Run: forever
3. Execute: "print s1"

Capturing Trace Output

print.java

1. Create a PrintEventInterface (event handler)

```
public static final PrintEventInterface myPrinter = new PrintEventInterface() {
        public void printEventHandler(int eventID, Object data, Agent agent, String message) {
                System.out.println("Soar said: <" + message + ">");
        }
};
```

2. Register for Event

Syntax

```
Agent.RegisterForPrintEvent(eventId, handler, extraData);
```

Try

```
agent.RegisterForPrintEvent(smlPrintEventId.smlEVENT_PRINT, myPrinter, null);
```

Managing WMEs on `input-link`

input.java

Syntax

- Identifier = Agent.GetInputLink();
- Identifier = Identifier.CreateIdWME("attr");
- FloatElement = Identifier.CreateFloatWME("attr", value);
- IntElement = Identifier.CreateIntWME("attr", value);
- WMElement.DestroyWME();

Task. Add and remove WMEs of differently typed values to the input-link. Use execution and run-control to verify via System printing.

Read `output-link` WMEs + Feedback

output.java

Syntax

- Int = Agent.GetNumberCommands();
- Identifier = Agent.GetCommand(Int);
- String = WMElement.GetAttribute();
- Int = Identifier.GetNumberChildren();
- WMElement = Identifier.GetChild(Int);
- WMElement = Identifier.FindByAttribute(String, Int)
- *Element = WMElement.ConvertTo*Element();
- Identifier.AddStatus<< Complete Error >>();

Task. Have an agent produce output. Parse via SML and provide feedback to the agent. Verify via agent action and working memory inspection.

Output Handler

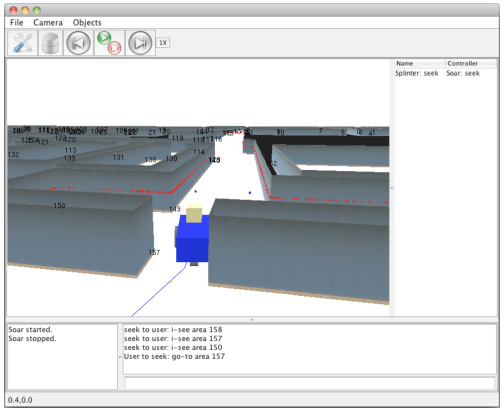
handler.java

Syntax

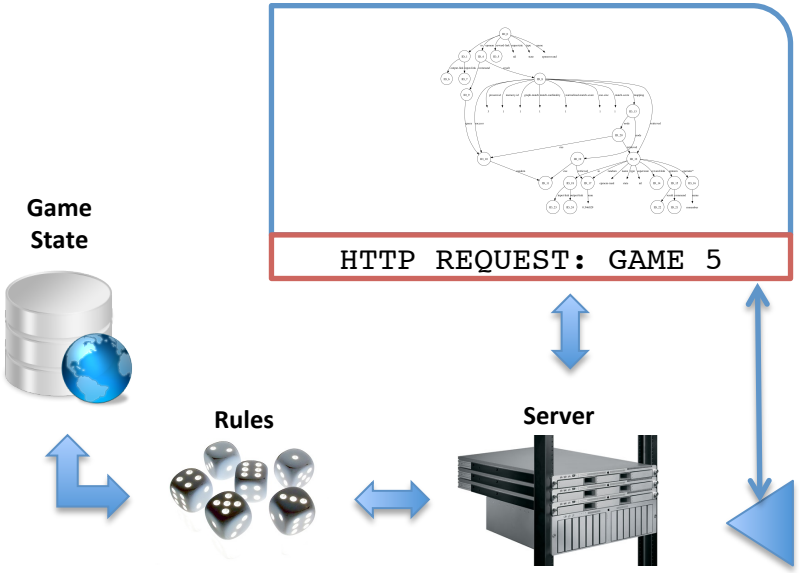
- Agent.AddOutputHandler("cmd", handler, data);

Task. Choose a secret number. Have a Soar agent guess the value via output commands until correct.

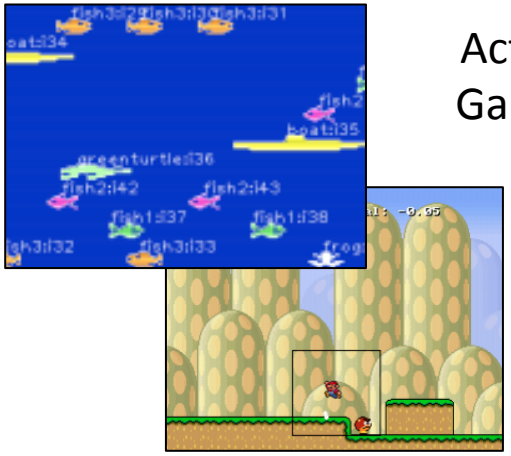
SML Example Environments



Cognitive Robotics



Web Gaming



Action Games

Interactive Mobile Music Generation

Additional Resources

- Quick Start Guide

  code.google.com/p/soar/wiki/SMLQuickStartGuide

- Threads in SML

  code.google.com/p/soar/wiki/ThreadsInSML