

# Expert Tables in CoCoViLa

Pavel Grigorenko

MBJSDT seminar  
29.10.2013

# Intro

An **Knowledge representation** is the area of Artificial Intelligence concerned with how **knowledge** is represented and manipulated.

An **expert system** (a.k.a. Knowledge-based System) is a computer program that simulates the judgement of a human that has **expert knowledge** and experience in a particular field. Typically, such a system contains a **knowledge base** containing accumulated experience and a **set of rules** for applying the knowledge base to each particular situation that is described to the program.

# Rule-based Approach

A **Production Rule** is a pair of the form

*(condition, action)*

with the meaning “If the *condition* is satisfied, then the *action* can (or must) be taken”.

Two methods of execution:

- Forward Chaining – “data-driven”
- Backward Chaining – “goal-driven”

# Example

**when**

an honest Politician exists

**then**

logically assert Hope

**when**

Hope exists

**then**

print "Hurrah!!! Democracy Lives"

**when**

Hope does not exist

**then**

print "Democracy is Doomed"

# Textbook Example

if  $h = 8$  and  $s_z admN = 5$  and  $FGr = 1$  and  $\sigma \leq 180$  and  $IM = VK6$

then  $s_z adm = 0.28$

if  $h = 8$  and  $s_z admN = 5$  and  $FGr = 1$  and  $\sigma \leq 1800$  and  $IM = VK6$

then  $s_z adm = 0.24$

if  $h=8$  and  $s_z admN=5$  and  $FGr \leq 3$  and  $\sigma \leq 180$  and  $IM \neq VK6$

then  $s_z adm = 0.76$

...

$h = 8, s_z admN = 5$

FGr	IM = VK6		IM $\neq$ VK6	
	$\sigma \leq 180$	$\sigma \leq 1800$	$\sigma \leq 180$	$\sigma \leq 1800$
1	0.28	0.24	0.38	0.32
$\leq 3$	0.56	0.48	0.76	0.64



# Textbook Example cont'd

$h = 8, s_{z \text{ adm}} N = 5$

FGr	IM = VK6		IM $\neq$ VK6	
	$\sigma \leq 180$	$\sigma \leq 1800$	$\sigma \leq 180$	$\sigma \leq 1800$
1	0.28	0.24	0.38	0.32
$\leq 3$	0.56	0.48	0.76	0.64

- Corresponding structural table

				X		X		$\sigma \leq 180$
					X		X	$\sigma \leq 1800$
$h = 8$	FGr = 1	FGr $\leq 3$	$s_{z \text{ adm}} N = 5$	X	X			IM = VK6
X	X		X	0.28	0.24	0.38	0.32	
X		X	X	0.56	0.48	0.76	0.64	

# Implementation in CoCoViLa

drive\_constr (drive.tbl) - Expert Table Visual Editor

File Edit View Help

Horizontal

+Rule +Row

-Rule -Row

< > ^ v

Vertical

+Rule +Column

-Rule -Column

^ v < >

road = dry	road = wet	road = icy	visibility > 100	visibility ≤ 100			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	tech condition = good
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	tech condition = bad tires
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	experience = beginner
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	90		
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	80		
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	90		
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	70		
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		stay at home	
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	30		



# Calling from Specification

```
class TableTest {  
    /*@ specification TableTest {  
        int x, y, value;  
        const String tableId = "test";  
        tableId, x, y -> value{@table};  
    }@*/  
}
```

# .tbl (=XML) format

```
<?xml version='1.0' encoding='utf-8'?>
<tables xmlns="cocovila"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="cocovila_file:./table.xsd"
        xmlns:cocovila="cocovila">
  <table id="smcomplex-gses">
    <input>
      <var id="mgn" type="String"/>
      ...
    </input>
    <output>
      <var id="ml" type="int"/>
    </output>
    <hrules>
      <rule var="mgn" cond="eq" value="User_training">
        <entry id="0"/>
      </rule>
      ...
    </hrules>
    <vrules>...</vrules>
    <data>
      <row id="0">
        <cell id="0">0</cell>
      ...
    </data>
  </table>
</tables>
```

# Interactive Consultant

Expert System Consultant: drive\_constr

What are the road conditions?

dry ▾

Cancel Back Next Finish

Expert System Consultant: drive\_constr

What is the value of an input (int) visibility?

50

Range constraint:  $0 \leq \text{visibility}$

Cancel Back Next Finish

Expert System Consultant: drive\_constr

In what technical condition is the car?

good ▾

good  
bad\_tires

Cancel Back Next Finish

Expert System Consultant: drive\_constr

The result is:

stay at home

for the given input values:

What are the road conditions?  
Answer: icy

What is the value of an input (int) visibility?  
Answer: 101

In what technical condition is the car?  
Answer: bad\_tires

What is your driving experience?  
Answer: beginner

Cancel Back Next Finish

Thanks!