# Языки предметной области

## Domain Specific Languages

## для описания бизнес-правил

кто здесь?

Антон Архипов

Java-программист

~~Hansa~~Swedbank

# О чём это я?

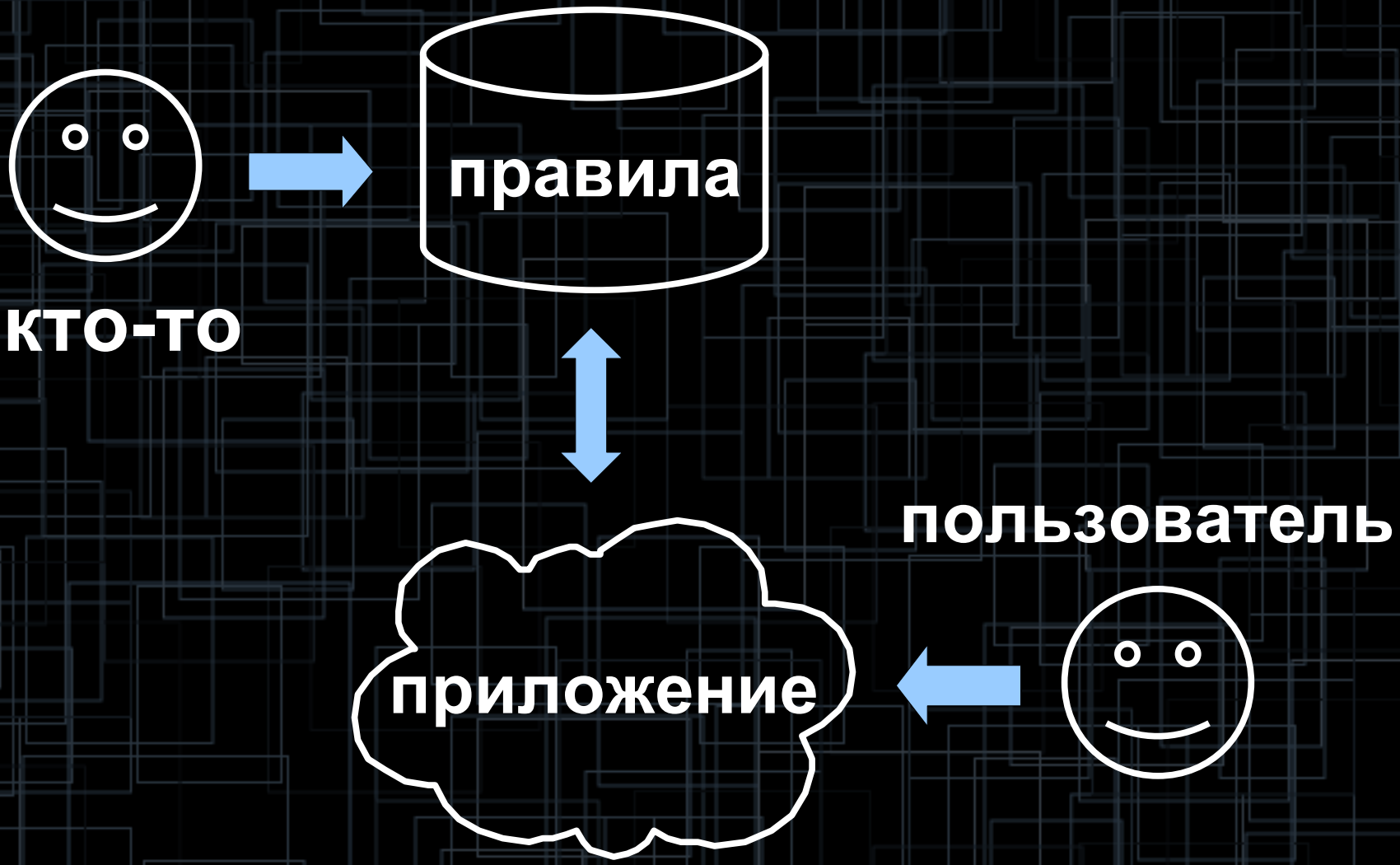
Как я до этого додумался?

Немного теории, что такое DSL

Проблемы выбора и реализации

# 30.01.2009 - “Экспертные системы для настоящих сибирских мужиков”

# Система управления бизнес-правилами *for dummies*



# Описание правил в Drools

```
rule "Hello World"
when
    m : Message( status == Message.HELLO,
                 myMessage : message )
then
    System.out.println( myMessage );
    m.setMessage( "Goodbye cruel world" );
    m.setStatus( Message.GOODBYE );
    update( m );
end
```

# JBoss Drools DSL

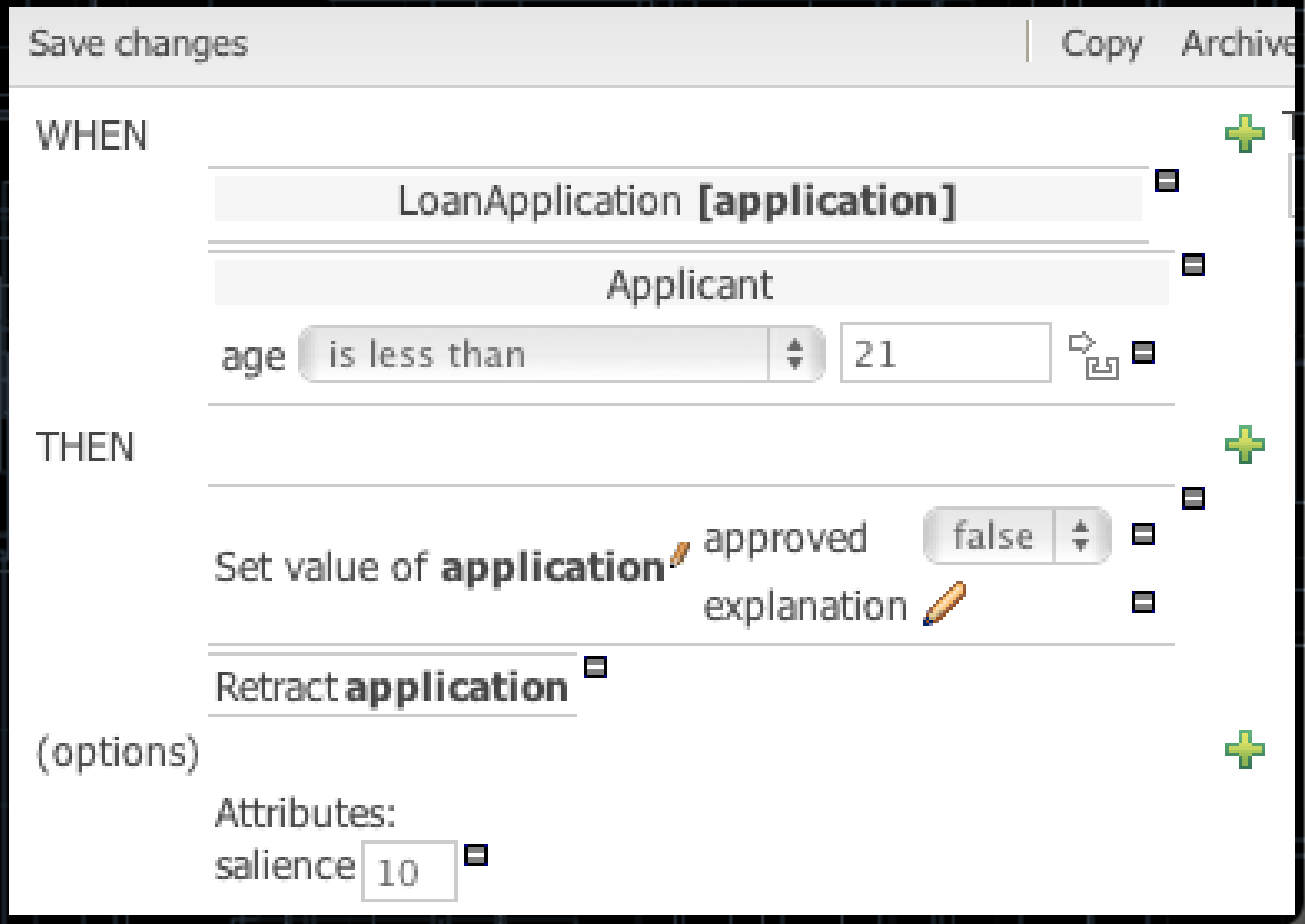
```
[when]There is a Person with name of "{name}"=
                                    Person(name=="{name}")

[then]Log "{message}"=System.out.println("{message}");

[when]And = and
```

# Пользовательский интерфейс JBoss Drools

**Малоинтуитивен**

**Неудобен**

**:-(**



B Mapping: date to kp A

B Mapping: date to revalKp A

A Rule name: bid-oHind

Type: DSL DRL

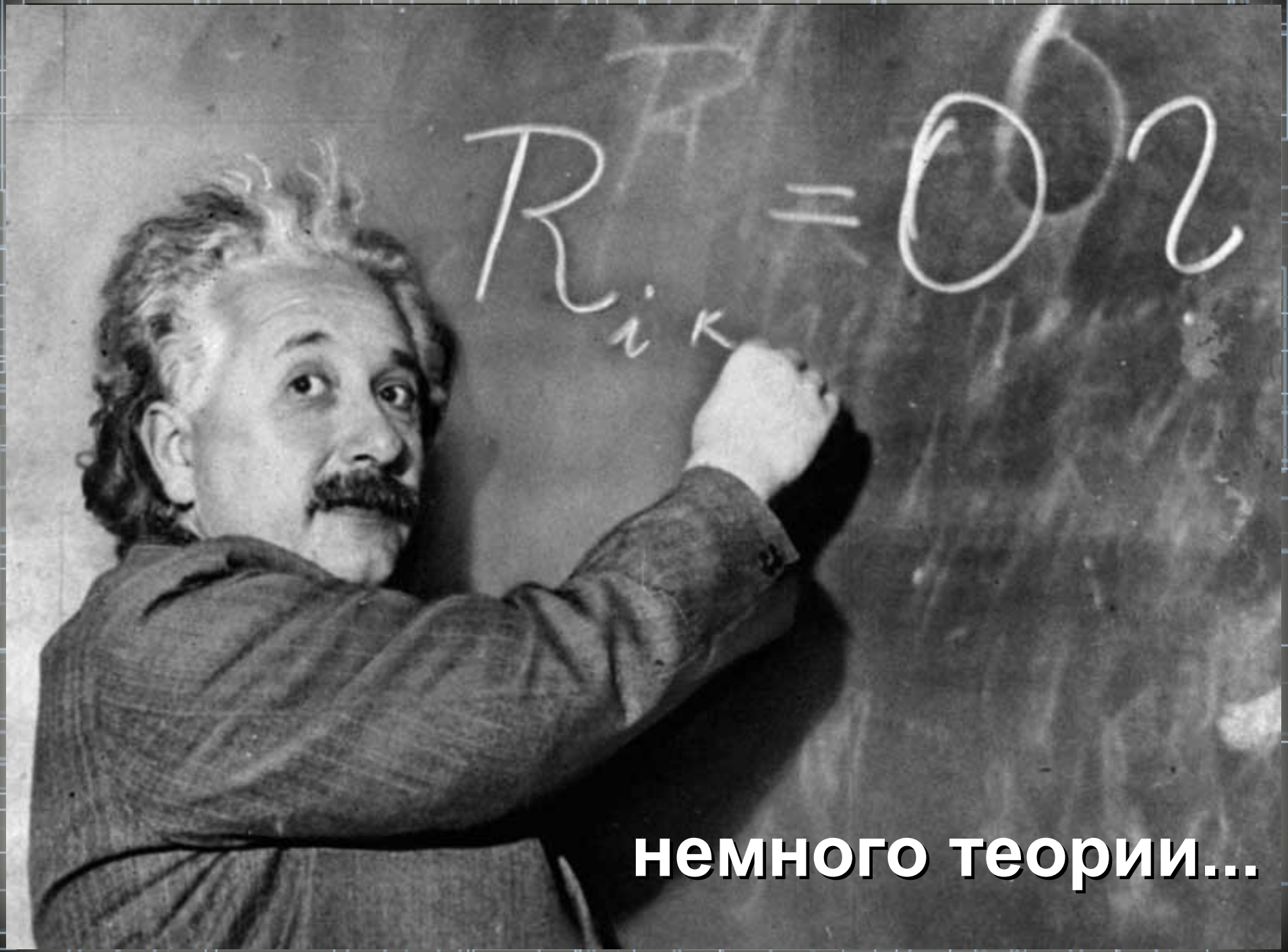
Conditions:

| When |
| --- |
| if has field bid |
| if value of field bidGrade equals 0 |

Actions:

| Then |
| --- |
| put field bid to result as oHind |

Add Basic rule | Add Advanced rule | View generated DRL and DSL texts


немного теории...

# Языки предметной области

External

Internal (Embedded)

Language Workbench

# External DSL

| Хорошо | Плохо |
| --- | --- |
| Гибкость реализации | Нет поддержки IDE |
| Выбор платформы | Нужно потрудиться |

# Internal DSL

## Хорошо

Легко реализовать

Не нужен парсер

Поддержка IDE

## Плохо

Ограничен базовым ЯП

# Примеры

# Ant

```
<project name="MyProject" default="dist" basedir=".">

  <target name="init">
    <mkdir dir="${build}"/>
  </target>

  <target name="compile" depends="init">
    <javac srcdir="${src}" destdir="${build}"/>
  </target>

  <target name="dist" depends="compile">
    <jar jarfile="${dist}/my.jar" basedir="${build}"/>
  </target>

  <target name="clean">
    <delete dir="${build}"/>
    <delete dir="${dist}"/>
  </target>
</project>
```

# Cucumber

```
Scenario: Add two numbers
  Given I have entered 50 into the calculator
  And I have entered 70 into the calculator
  When I press add
  Then the result should be 128 on the screen
```

```
Given /I have entered (.*) into the calculator/ do |n|
  calculator = Calculator.new
  calculator.push(n.to_i)
end
```

# Apache Camel

```
from(“data?fileName=*xml”).
  choice().
    xpath("//foo").to("queue:bar").
      otherwise().to("queue:others");
```

# Groovy Builder

```
def f = framesize:[300,300], text:'My Window' {
    labelbounds:[10,10,290,30], text:'Save changes'
    panelbounds:[10,40,290,290] {
        buttontext:'OK', action:{ save close }
        buttontext:'Cancel', action:{ close }
    }
}
```

# Проблемы выбора

# Проблемы реализации

# JRuby & Groovy

Groovy – the dynamic Java

http://groovy.codehaus.org

JRuby - Ruby для JVM

http://jruby.org

# DEMO

ant.arhipov@gmail.com

Twitter: antonarhipov

http://arhipov.blogspot.com

http://anton-arhipov.livejournal.com