

Израда језика у Пајтону

Језици специфични за домен (*Domain Specific Languages*)

Проф. др Игор Дејановић ([igord на uns ac rs](mailto:igord@uns.ac.rs))

Креирано 2025-11-17 Mon 14:26, притисни ESC за мапу, Ctrl+Shift+F за претрагу, "?" за помоћ

Проблем и мотивација

Језици специфични за домен - ЈСД (*Domain-Specific Languages - DSL*)

- Језици специфични за домен (ЈСД, енг. *Domain-Specific Languages - DSL*) су језици прилагођени и ограничени на одређени домен проблема.
- За разлику од језика опште намене (ЈОН, енг. *General Purpose Language - GPL*), нуде повећање експресивности кроз употребу концепата и нотација прилагођених домену проблема и доменским експертима.
- Називају се још и *мали језици* (енг. *little languages*).
- Успешан ЈСД је фокусиран на узак, добро дефинисан домен и покрива га на одговарајући начин.
- Домен често има свој језик коришћен од стране доменских експерата иако не постоји његова имплементација на рачунару.



Примери

SQL

```
SELECT player, stadium  
FROM game JOIN goal ON (id=matchid)
```

JPA мапирање

```
@Entity
@Table(name="COURSES")
public class Course {

    private long courseId;
    private String courseName;

    public Course() {
    }

    public Course(String courseName) {
        this.courseName = courseName;
    }

    @Id
    @GeneratedValue
    @Column(name="COURSE_ID")
    public long getCourseId() {
        return this.courseId;
    }
}
```

Build језици (Ant/Maven/Gradle)

```

<project>
  <target name="clean">
    <delete dir="build"/>
  </target>
  <target name="compile">
    <mkdir dir="build">
    <javac srcdir="src" destdir="build" classpath="lib/*" >
  </target>
  <target name="jar">
    <mkdir dir="build">
    <jar destfile="build/app.jar" manifest="manifest.mf" >
    <attr name="Main-Class" value="com.mycompany.app.Main" />
  </jar>
  </target>
  <target name="run">
    <java jar="build/app.jar" classpath="lib/*" >
  </target>
</project>

```

```

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://maven.apache.org/POM/4.0.0"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd"
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.mycompany.app</groupId>
  <artifactId>my-app</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>Maven Quickstart</name>
  <url>http://maven.apache.org</url>

  <dependencies>
    <dependency>
      <groupId>org.apache.maven</groupId>
      <artifactId>maven-quickstart</artifactId>
      <version>1.0</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
</project>

```

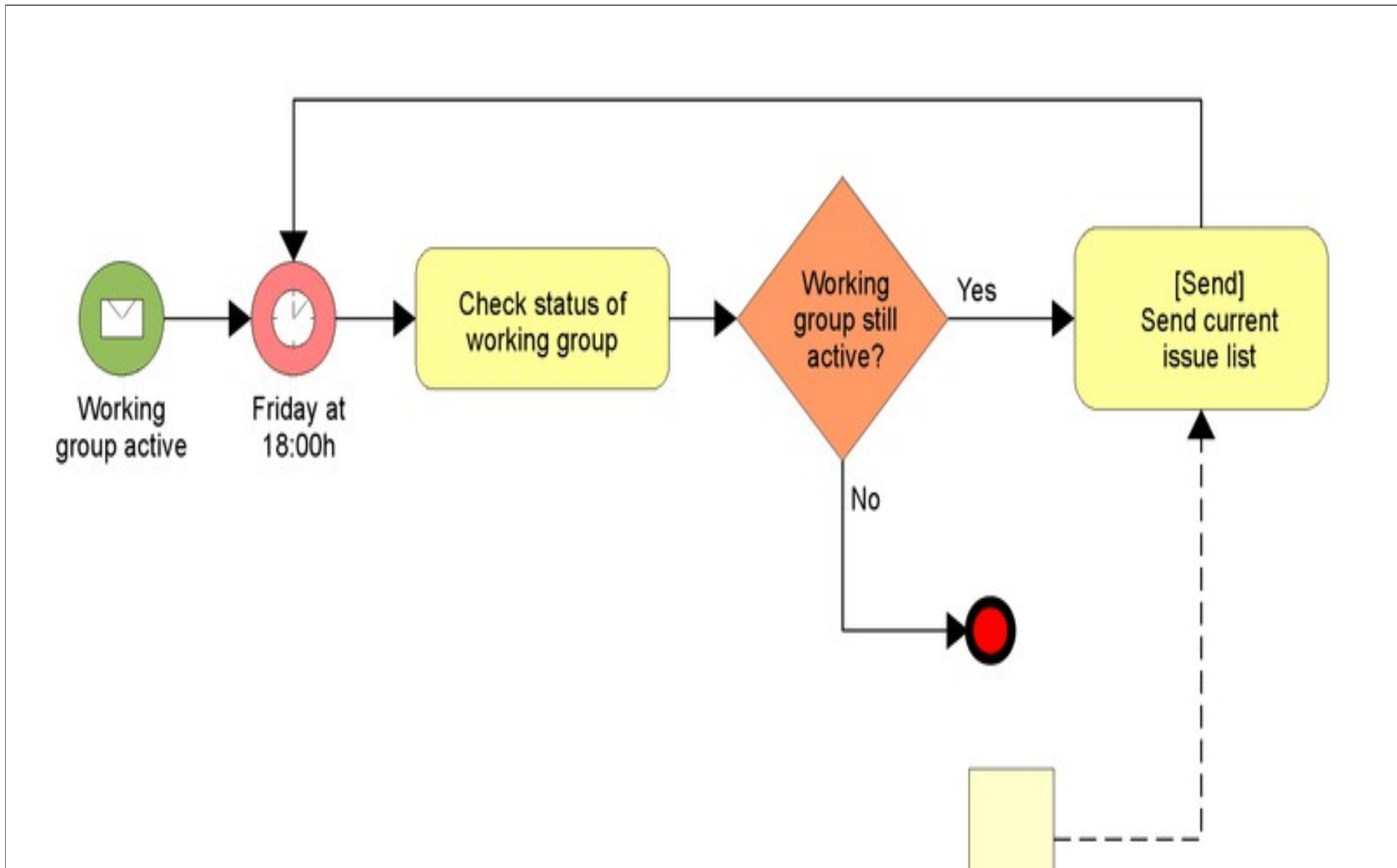
```

task transform {
  ext.srcFile = file('mountains.xml')
  ext.destDir = new File(buildDir, 'generated')
  doLast {
    println "Transforming source file."
    destDir.mkdirs()
    def mountains = new XmlParser().parse(srcFile)
    mountains.mountain.each { mountain ->
      def name = mountain.name[0].text()
      def height = mountain.height[0].text()
      def destFile = new File(destDir, "${name}.txt")
      destFile.text = "$name -> ${height}\n"
    }
  }
}

```

```
}  
}
```


Пословни процеси - BPMN





Issue list

OrgMode

Белешке, креирање садржаја, агенда, писмено програмирање (*literate programming*).

```
** PROJ
*** TODO Прегледати пријаву грешке #173
SCHEDULED: <2022-12-14 Wed>
1. [x] Неко парче кода:
  #+begin_src rust
  fn main() {
    // Statements here are executed when the compiled binary is called

    // Print text to the console
    println!("Hello World!");
  }
  #+end_src

  #+RESULTS:
  : Hello World!
```

2. [] Нека друга забелешка...

```
*** WAIT Предати пројектни извештај
```

Активност	Завршено	Проблеми
Прва активност	30%	Нема
Друга актиност	25%	Проблеми у снабдевању

Typst

Припрема за штампу (*typesetting*).

= Markup <markup>

With built-in syntax for the most common document elements, Typst markup is designed to be pleasant to write and read:

- ***Strong*** and *_normal_* emphasis
- A reference to @markup
- Math: $a, b \in \{1/2, \sqrt{4ab}\}$

But that's just the surface!

1. Markup

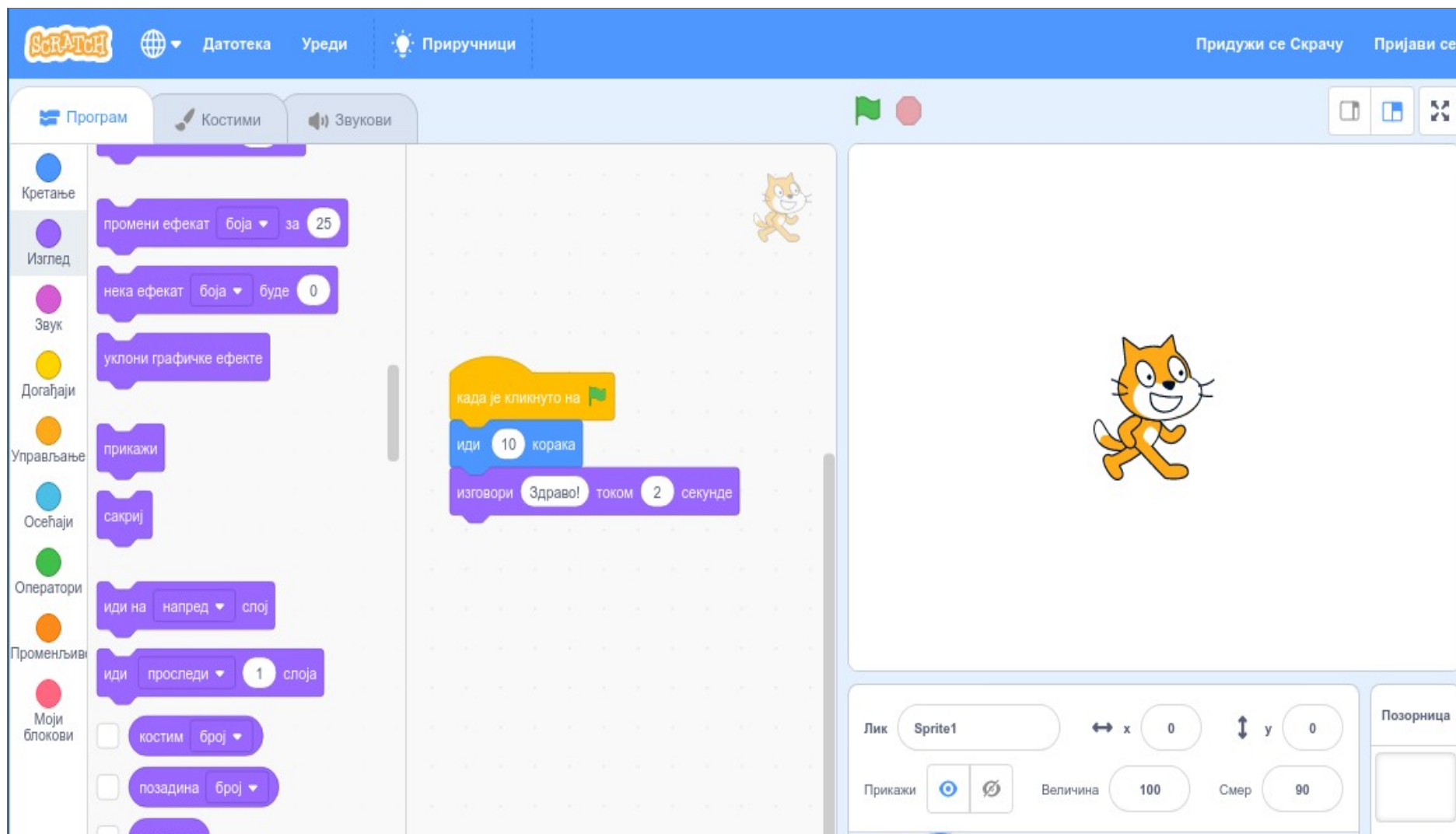
With built-in syntax for the most common document elements, Typst markup is designed to be pleasant to write and read:

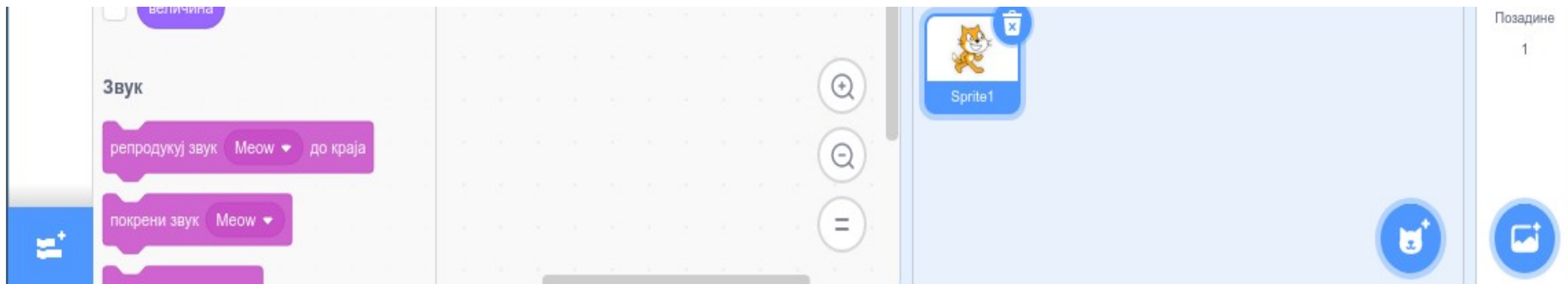
- **Strong** and *normal* emphasis
- A reference to Section 1
- Math: $a, b \in \left\{\frac{1}{2}, \sqrt{4ab}\right\}$

But that's just the surface!

Scratch

Учење програмирања за децу.





Inform

Inform је језик за креирање интерактивне фикције (текстуалних авантура) базиран на природном језику.

```
"Cactus Will Outlive Us All"
```

```
Death Valley is a room. Luckless Luke and Dead-Eye Pete are men in the Valley.  
A cactus is in the Valley. Persuasion rule: persuasion succeeds.
```

```
A person has an action called death knell. The death knell of Luckless Luke is  
pulling the cactus.
```

```
The death knell of Dead-Eye Pete is Luke trying dropping the cactus.
```

```
Before an actor doing something:
```

```
repeat with the victim running through people in the location:
```

```
let the DK be the death knell of the victim;
```

```
if the DK is not waiting and the current action is the DK:
```

```
say "It looks as if [the DK] was the death knell for [the victim], who  
looks startled,
```

```
then nonexistent.";
```

```
now the victim is nowhere.
```

Chuck

Језик за креирање музике и синтезу звука у реалном времену.

```
// this synchronizes to period
.5::second => dur T;
T - (now % T) => now;

// construct the patch
SndBuf buf => Gain g => dac;
// read in the file
"kick.wav" => buf.read;
// set the gain
.5 => g.gain;

// time loop
while( true )
{
    // set the play position to beginning
    0 => buf.pos;
    // randomize gain a bit
    Math.random2f(.8,.9) => buf.gain;

    // advance time
    1::T => now;
}
```


GritQL

GritQL је ЈСД за претрагу, линтинг и трансформацију кода.

```
`console.log($log)` => . where {  
  $log <: not within `try { $_ } catch { $_ }`  
}
```

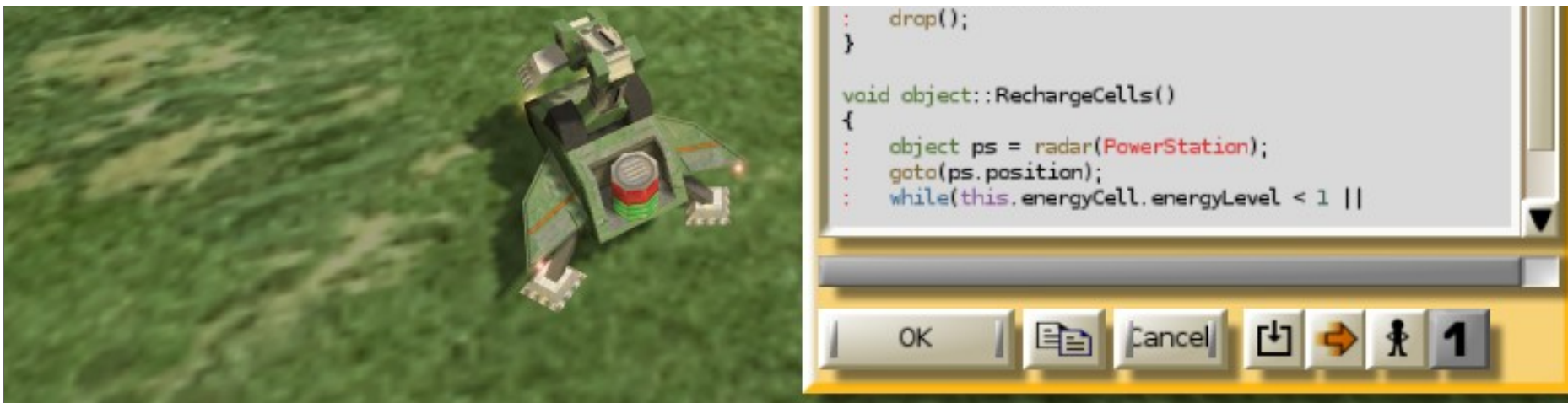
Colobot

- Colobot је едукативна real-time стратегија.
- Играч програмира роботе у виртуелном окружењу ЈСД-ом налик на C++ и Java.

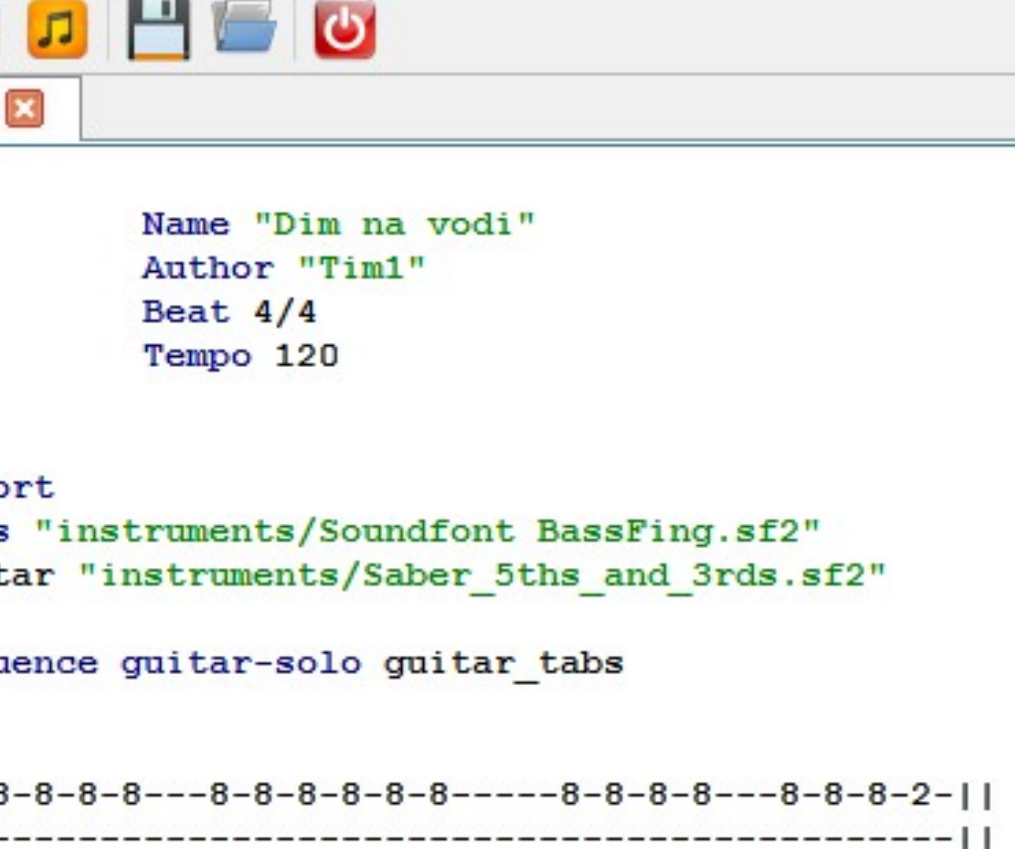
A screenshot of the Colobot program editor. The window has a yellow title bar with the text "Program editor" and standard window controls. Below the title bar is a toolbar with icons for file operations (new, open, save, print), editing (undo, redo, cut, copy, paste), and other functions (a stack of papers, a person icon, and a robot icon). The main area contains C++ code:

```
extern void object::ChangeChargeCell()
{
:   object cell = radar(PowerCell);
:   point cellpos = cell.position;
:   ChangeCell(cellpos);
:   cell = radar(PowerCell);
:   goto(cell.position);
:   grab();
:   RechargeCells();
:   goto(cellpos);
:   drop();
}

void object::ChangeCell(point p)
{
:   object ws = radar(WingedShooter);
:   goto(ws.position);
:   grab();
:   turn(-90);
:   move(2);
:   drop();
:   goto(p);
:   grab();
:   goto(ws.position);
}
```



Језик за гитарске таблатуре.



Spike pyTabs

Text

```
[
    Name "Dim na vodi"
    Author "Tim1"
    Beat 4/4
    Tempo 120
]

import
bass "instruments/Soundfont BassFing.sf2"
guitar "instruments/Saber_5ths_and_3rds.sf2"

sequence guitar-solo guitar_tabs
{
R|-8-8-8-8---8-8-8-8-8-8---8-8-8-8---8-8-8-2-||
e|-----||
B|-----||
G|-----||
D|-----||
}
```

```
D |-----| |
A |-----0-----1-0-----| |
E |-0--3-----0--3-----0--3-----3--0-| |

}
```

Tekst Neki Selection Label

PyFlies

Језик за израду психолошких тестова.

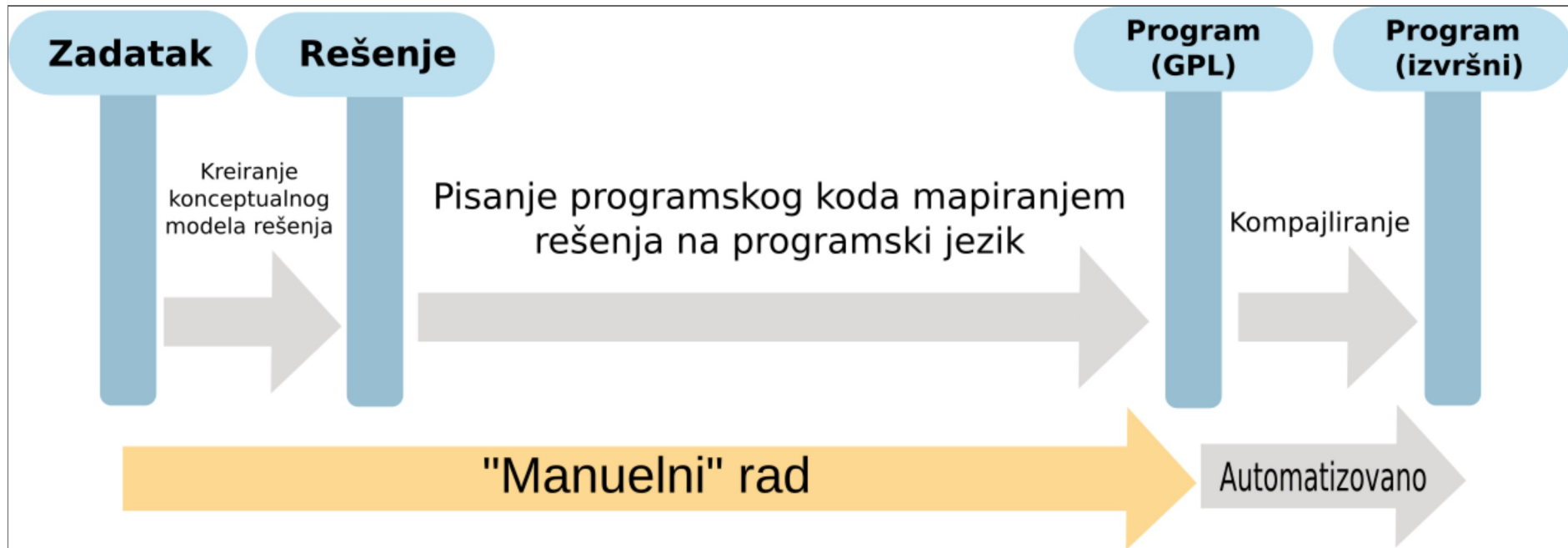
Предности употребе

Утицај на продуктивност

- Поједине студије показују да повећање продуктивности иде и до 1000%¹.
- Шта је основни разлог за повећање продуктивности?

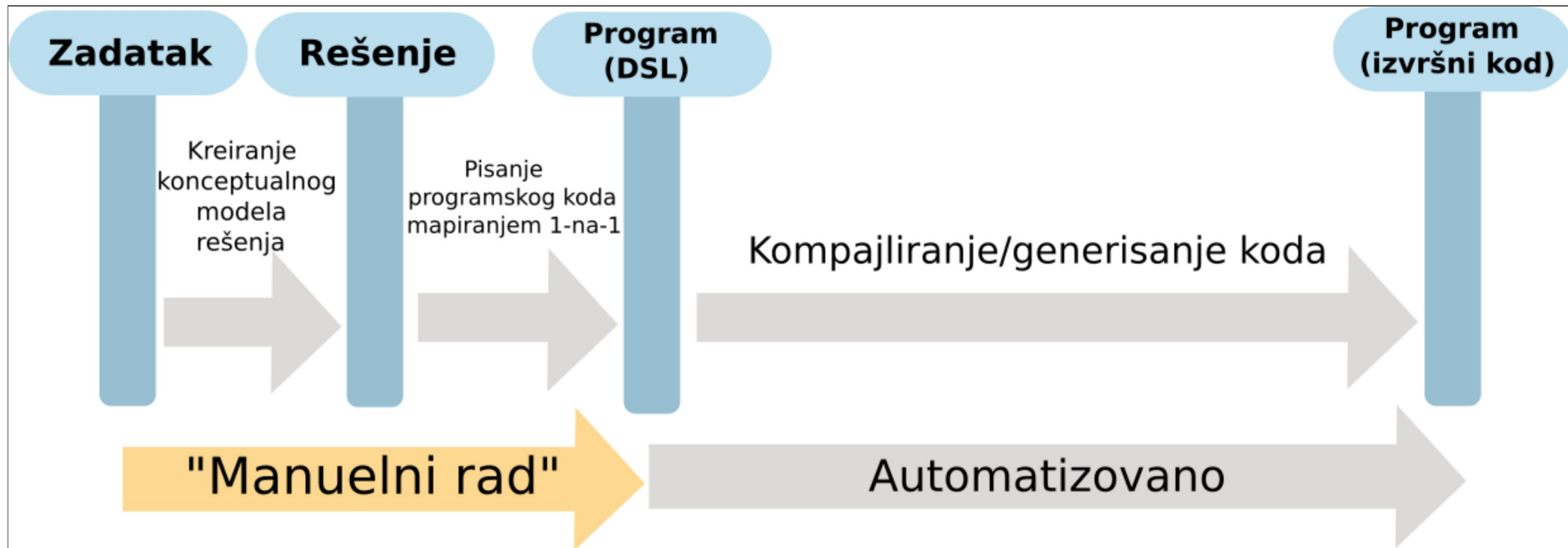
1. MetaCase, *Nokia case study*, tech. rep., MetaCase, 2007

Проблем менталног мапирања



Dmitriev, S. *Language oriented programming: The next programming paradigm*, JetBrains onBoard, 2004.

Решење употребом ЈСД



Dmitriev, S. *Language oriented programming: The next programming paradigm*, JetBrains onBoard, 2004.

Када језик сматрамо ЈСД-ом?

- Зависи од тога шта нам је домен.
- Језик може бити више или мање прилагођен неком домену.
- У екстремном случају и општи језик као што је Јава можемо сматрати ЈСД-ом ако нам је домен "развој софтвера". Наравно, иако тачно у теоријском смислу, у практичном губимо све предности ЈСД.
- Добар ЈСД покрива узак, добро дефинисан домен (домен проблема). Користи само концепте циљног домена, ограничен је на дати домен и самим тим је исказивање решења језгровитије и јасније доменским експертима.
- Чест је случај да језик настане као ЈСД али се временом прошири до те мере да га можемо сматрати ЈОН.

Како правити ЈСД-ове - textX

Speaker notes