

ELS - Checkpoint 3

João Félix - up202008867
João Malva - up202006605
Sofia Teixeira - up201806629

Problem Statement

- **Objectives:**
 - An external DSL that configures and executes the semantic model
 - An expanded version of the semantic model which includes support for simple operations and types

Design Decisions

- **How the DSL Meets Objectives:**
 - User-friendly syntax.
 - Accurate and efficient data extraction.
 - Extensibility for additional operations.
- **Results:**
 - Successfully aggregated data from multiple formats into a single table.

Architecture Overview

- Input DSL Script
- Parser (Xtext-generated)
- Semantic Model
- Java Backend Execution

Tools Used:

- Xtext
- Java class integrating the semantic model with the DataBaseBuilder API

```
.nestedIn("Device", "Resources")

.loadYAML()
.from("assignment2Files/run1/decision_tree.yaml")
.into("decision_tree")

.loadYAML()
.from("assignment2Files/run1/decision_tree.yaml")
.into("decision_tree2")
.nestedIn("params")

.selectMax()
.onColumn("time%")
.onTable("time")
.toTable("timeNew")

.concatHorizontal()
.toTable("final")
.onTables("decision_tree", "decision_tree2", "time", "vitis")

.save()
.tables("final")
.to("assignment2Files/output.html")
.as("html")

.end();
```

Example of input file

Grammar Definition

```
1 grammar org.xtext.example.mydsl.MyDsl with org.eclipse.xtext.common.Terminals
2
3 generate myDsl "http://www.xtext.org/example/mydsl/MyDsl"
4
5@Model:
6     operations+=TopLevelOperation+;
7
8@TopLevelOperation:
9     LoadOperation | SelectOperation | ConcatOperation | SaveOperation |
10    PrintOperation | RenameOperation | LimitOperation |
11    ArgMaxOperation | ArgMinOperation | DropOperation | FilterOperation |
12    ProcessFoldersOperation | {TopLevelOperation} EndOperation;
13
14 LoadOperation:
15     LoadJSON | LoadXML | LoadYAML;
16
17 LoadJSON:
18     '.loadJSON()'
19     '.from(' file=STRING ')'
20     '.into(' table=STRING ')'
21     ('.withAttributes' '(' attributes+=STRING (',' attributes+=STRING)* ')')?
22     ('.nestedIn' '(' nested+=STRING (',' nested+=STRING)* ')')?;
23
24 LoadXML:
25     '.loadXML()'
26     '.from(' file=STRING ')'
27     '.into(' table=STRING ')'
28     ('.nestedIn' '(' nested+=STRING (',' nested+=STRING)* ')')?;
29
30 LoadYAML:
31     '.loadYAML()'
32     '.from(' file=STRING ')'
33     '.into(' table=STRING ')'
34     ('.nestedIn' '(' nested+=STRING (',' nested+=STRING)* ')')?;
```

```
35@ConcatOperation:
36     '.concatHorizontal()'
37     '.toTable(' target=STRING ')'
38     '.onTables(' tables+=STRING (',' tables+=STRING)* ')';
39
40@FilterOperation:
41     '.filter()'
42     '.onColumn(' column=STRING ')'
43     '.onTable(' table=STRING ')'
44     '.where(' condition=STRING 'value' value=STRING ('toTable' target=STRING)? ')';
45
46@SaveOperation:
47     '.save()'
48     '.tables(' tables+=STRING (',' tables+=STRING)* ')'
49     '.to(' file=STRING ')'
50     '.as(' format=STRING ')';
51
52@PrintOperation:
53     {PrintOperation} PrintAll | PrintTable;
54
55@PrintAll:
56     '.printAll()';
57
58@PrintTable:
59     '.printTable(' table=STRING ')';
60
61@RenameOperation:
62     '.renameColumn()'
63     '.from(' original=STRING ')'
64     '.to(' renamed=STRING ')';
65     '.onTable(' table=STRING ')';
66
67@LimitOperation:
68     '.limit()'
69     '.table(' table=STRING ')'
70     '.from(' start=INT ')';
71     '.to(' end=INT ')';
```

Backend Implementation

- Java Backend:
 - Parses the DSL file into an abstract syntax tree (AST).
 - Iterates over the AST and executes operations using DataBaseBuilder.

```
DataBaseBuilder dbBuilder = new DataBaseBuilder();

var treeIterator = resource.getAllContents();

while (treeIterator.hasNext()) {
    var element = treeIterator.next();
    System.out.println(element);

    if (element instanceof TopLevelOperation operation) {
        switch (operation.getClass().getSimpleName()) {
            case "ArgMaxOperation":
                dbBuilder.selectMax().end();
                break;
            case "ArgMinOperation":
                dbBuilder.selectMin().end();
                break;
        }
    }
}
```

Checkpoint 2 vs 3

```
db
  .processFolders()
  .folders("assignment2Files/run1", "assignment2Files/run2", "assignment2Files/run3") // Specify folders
  .operations(() -> {
    db.loadJSON()
      .from("profiling.json")
      .into("time")
      .withAttributes("time%", "seconds", "name")

    .loadXML()
      .from("vitis-report.xml")
      .into("vitis")
      .nestedIn("AreaEstimates", "Resources")

    .loadYAML()
      .from("decision_tree.yaml")
      .into("decision_tree")

    .loadYAML()
      .from("decision_tree.yaml")
      .into("decision_tree2")
      .nestedIn("params")

    .selectMax()
      .onColumn("time%")
      .onTable("time")
      .toTable("timeNew")

    .concatHorizontal()
      .toTable("final")
      .onTables("decision_tree", "decision_tree2", "time", "vitis")

    .end();
  })
  .printTable("final")
  .end();
```

```
.nestedIn("Device", "Resources")

.loadYAML()
.from("assignment2Files/run1/decision_tree.yaml")
.into("decision_tree")

.loadYAML()
.from("assignment2Files/run1/decision_tree.yaml")
.into("decision_tree2")
.nestedIn("params")

.selectMax()
.onColumn("time%")
.onTable("time")
.toTable("timeNew")

.concatHorizontal()
.toTable("final")
.onTables("decision_tree", "decision_tree2", "time", "vitis")

.save()
.tables("final")
.to("assignment2Files/output.html")
.as("html")

.end();
```

Obstacles

- Steep Learning Curve:
 - Eclipse-specific workflows can be non-intuitive for developers unfamiliar with the environment.
 - Error reporting in the DSL is often not user-friendly. Syntax errors or type mismatches are not always clearly communicated to end-users in an actionable way.
- Complexity of an External DSL:
 - Building an external DSL involves designing a new grammar, writing a parser, and maintaining compatibility with evolving use cases.
 - Managing changes to the DSL over time (e.g., adding features) can require non-trivial updates to the grammar and associated tooling.

Future Implementation

- More complete parsing
- Enable previous operations with this new DSL
- Implement new operations to solve Cal's problem

Demo





Questions?