

Exercises

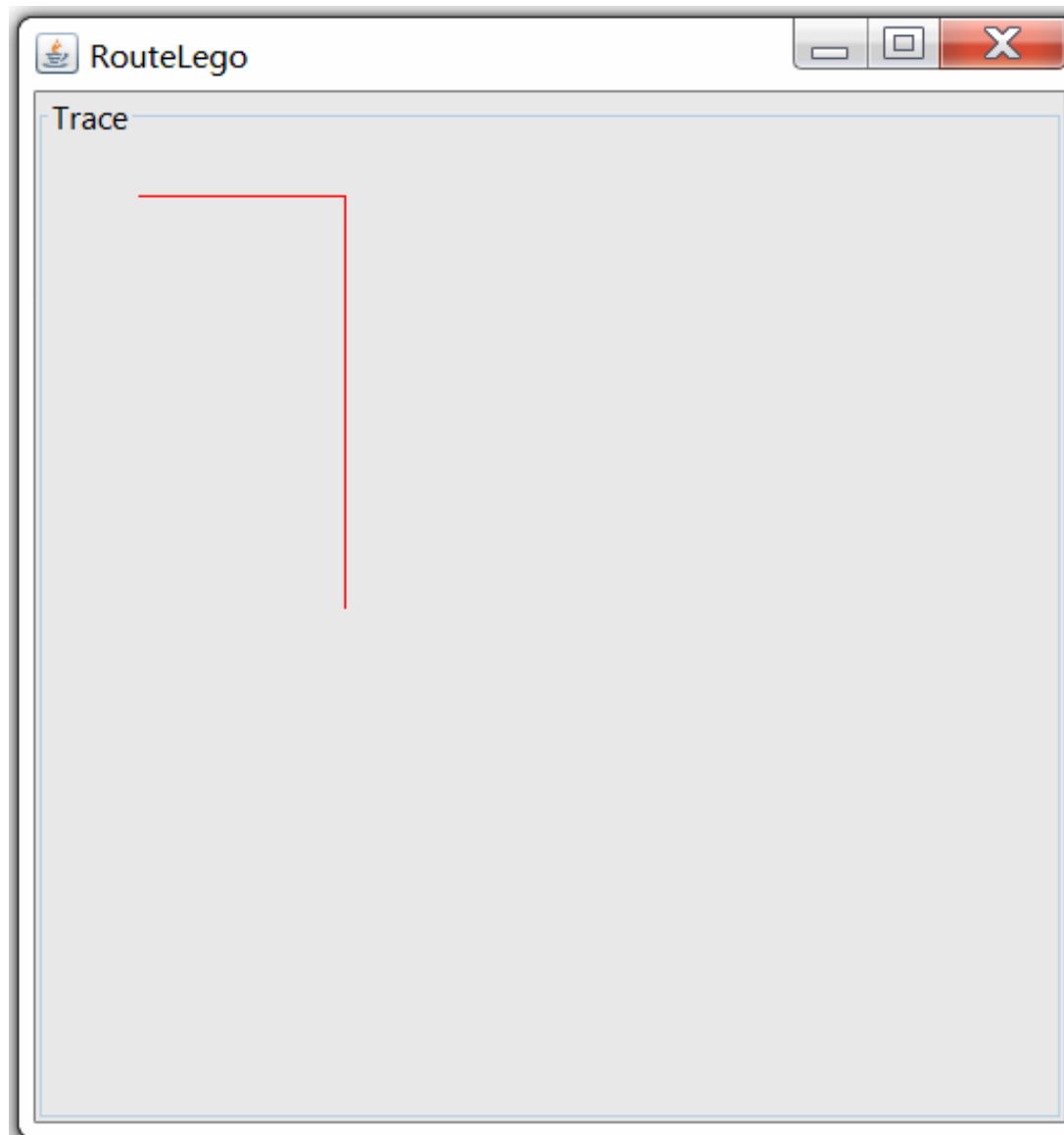
MPS / MetaMod / Xtext

Ana Maria Șutîi



- The steps to define a DSL
 - In a projectional language-workbench, MPS
 - In a parser-based language-workbench, Xtext + EMF
 - In an simple, modularity-focused meta-tool, MetaMod
- How to extend a DSL in these language workbenches
- How to embed existing DSLs into your own DSL

- STEP 1: Make a “Route” language for a Lego Mindstorm robot:
 - Boundary within which the robot can move
 - Commands that the robot can make
 - Forward commands on a certain distance (e.g. integer distance)
 - Turn commands (to the left and to the right)
- STEP 2: Extend the “Route” language, and create a separate language, called “RouteWithExpressions”:
 - Add a section with variables to the route:
 - The section contains a list of named variables that can be used in forward commands to specify the distance
 - Variable needs to have an initialization (e.g., an integer), and can be initialized only once
 - Embed arithmetic expressions in the forward command
 - HINT: take the expressions from the base language of MPS
 - Combine the variables with the expressions in the forward commands so that the DSL user can write “forward a + 100”, for instance



- Metamodel
 - What concepts play a role in the Route language?
 - What are the relationships among these concepts?
 - Involve the domain expert in the process 😊
- Constraints/Checking rules
 - What constraints do you need to define?
 - E.g., restrict some types of expressions from being available to the user of the DSL
- Extensions and embeddings
 - How do you implement the interplay between your DSL and the base language of MPS?
 - Structure
 - Generation
 - Type-system

- CTRL + R (search for concepts and import them)
- CTRL + L (search for languages and import them)
- CTRL + space (auto completion)
- CTRL + B (jump to the definition of the element)
- Alt + enter (show intentions)

- <https://confluence.jetbrains.com/display/MPSD34/MPS+User's+Guide>
- Shapes language from class:
<https://github.com/farcasia/ShapesLang.git>
- mbeddr blog posts
- Youtube videos