



# Model-based Software Development

## Practice 3

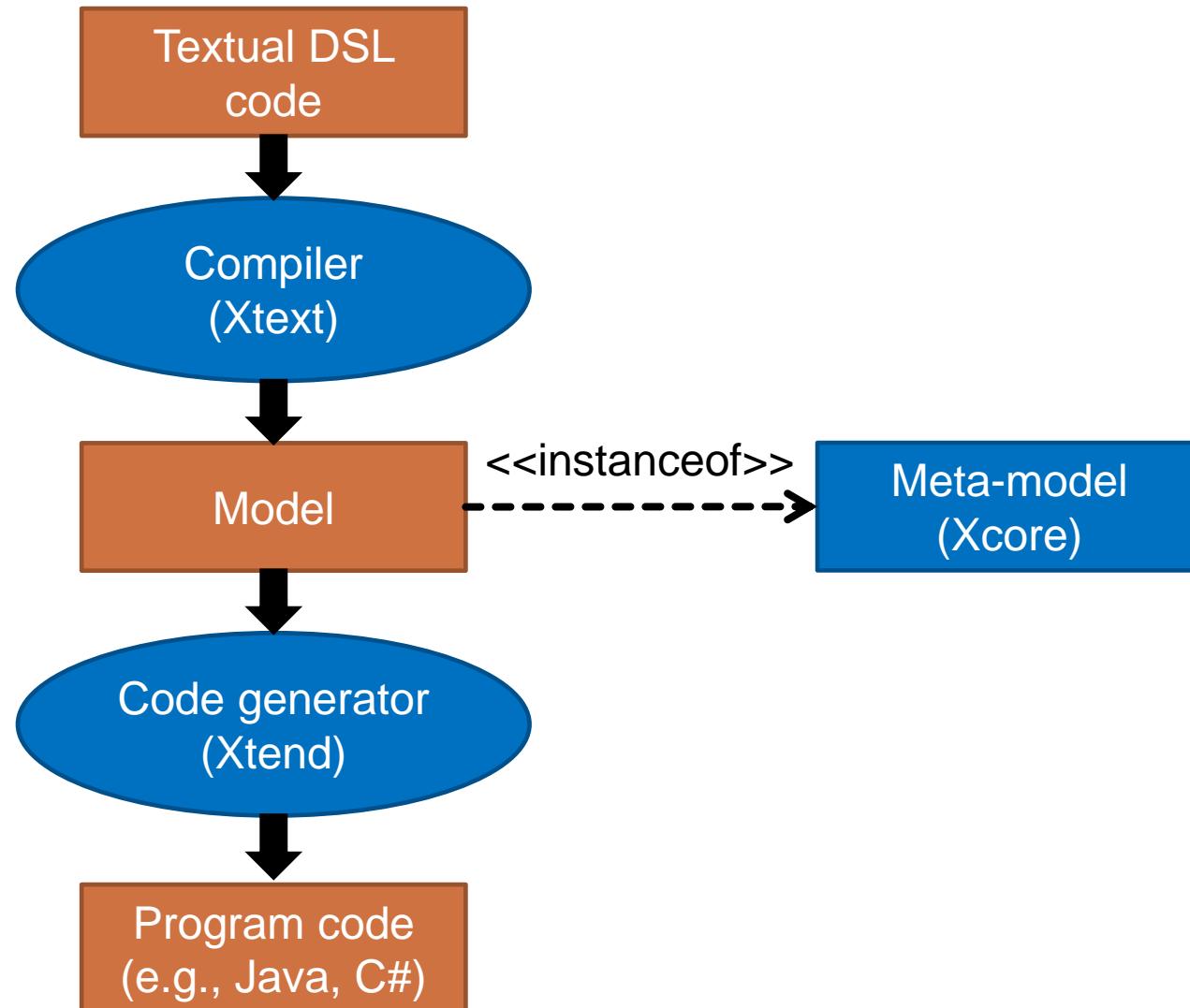
### IDE support for textual DSLs

Dr. Balázs Simon

# IDE support for textual DSLs

- Eclipse environment – Java language
- Xtext – describing and processing DSLs, IDE support
- Xcore – meta-modeling (AST description at the same time)
- Xtend – template-based code generation
- IDE functions:
  - > syntax highlighting, validation, error markers, content assist, hyperlinking, outline, automatic code formatting, automatic build / code generation

# Compilation and code generation



- Used for describing a meta-model
- Compiles to an EMF (Eclipse Modeling Framework) meta-model
- Essentially a MOF (MetaObject Facility) implementation
  - > simplified UML class diagram
- Used together with Xtext it is also an AST description

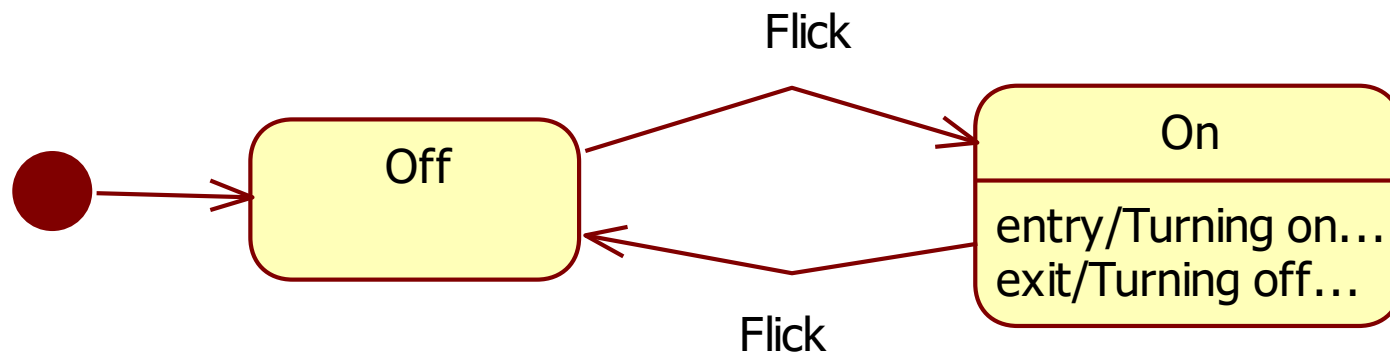
# Xtext

- Compiler front-end for textual DSLs
- Web page: <https://www.eclipse.org/Xtext/>
- Documentation:
  - > <https://www.eclipse.org/Xtext/documentation/index.html>
- One of the most popular lexer-parser generator and IDE support under Eclipse
- Can only be programmed in Java (and Xtend)
- Can work conveniently together with Java, EMF and Xcore

# Xtend

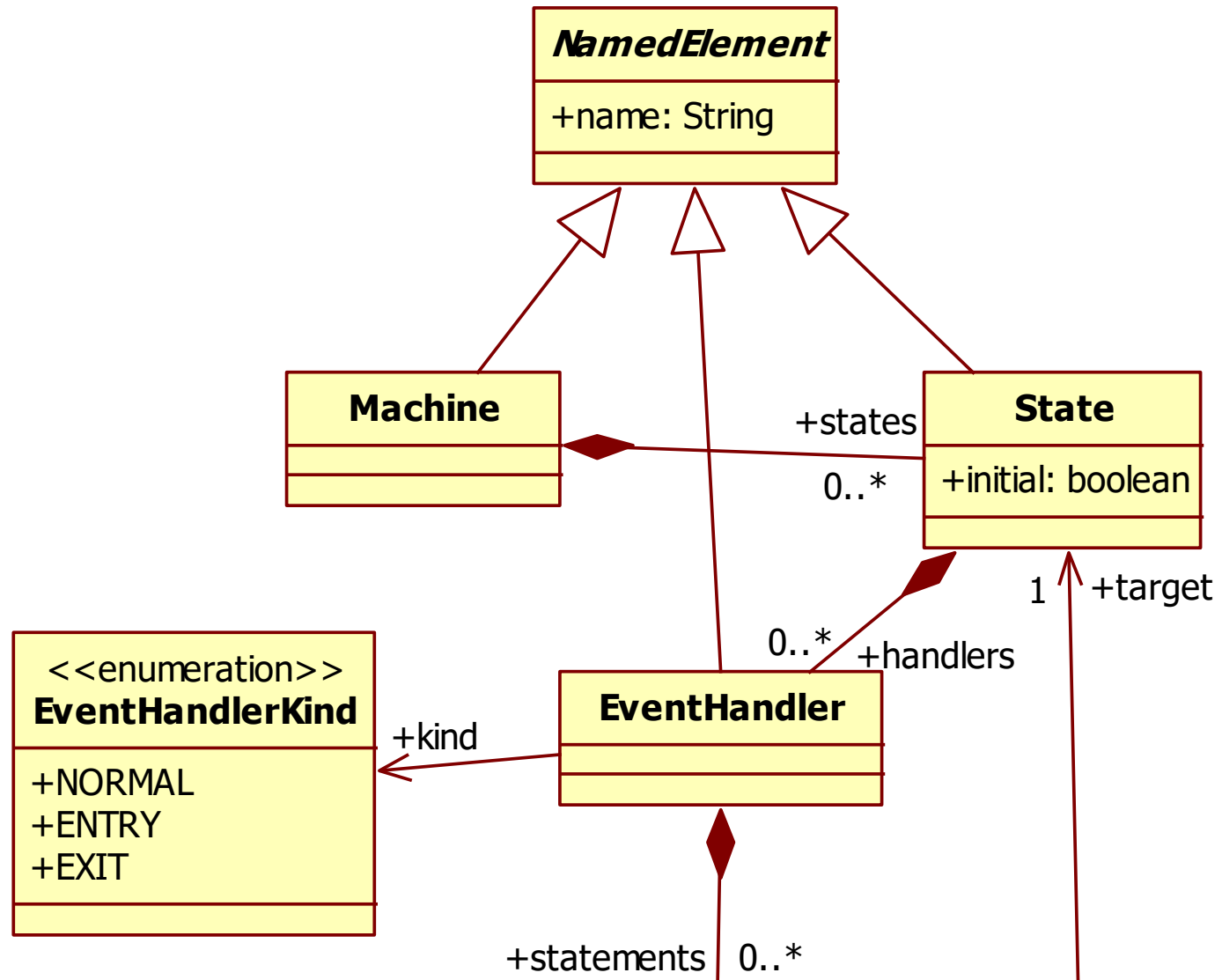
- Simplified Java
- Large expressive power
- Built-in template-based code generator
  - > code generation: traversing the model while printing text in a controlled manner
- Compiles to Java 8 in the background
- Can work conveniently together with Xtext and Xcore

# Task: IDE support for a textual state machine DSL



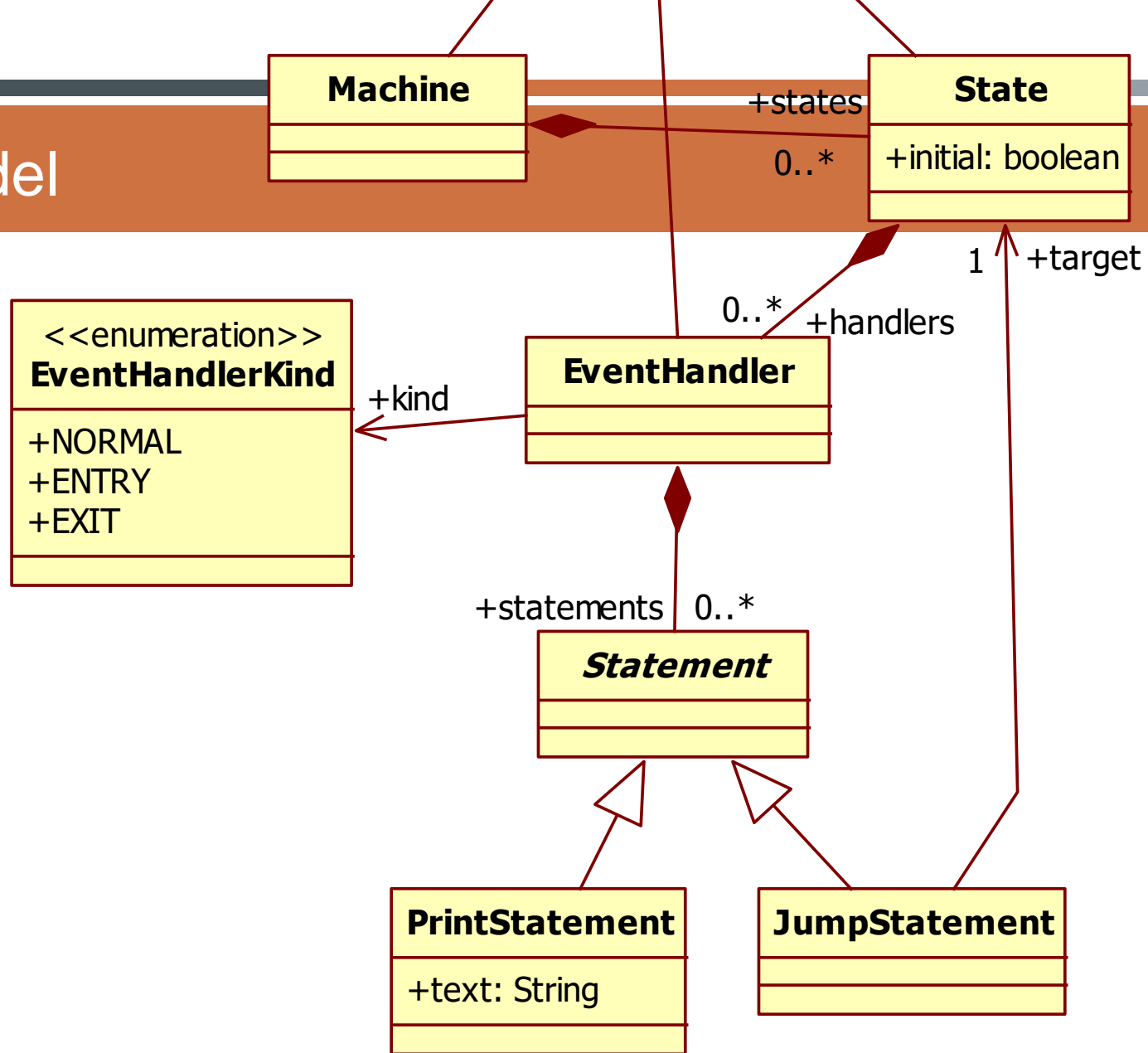
```
machine Lamp
{
  initial state Off
  {
    event Flick
    {
      jump On;
    }
  }
  state On
  {
    entry
    {
      print "Turning on...";
    }
    event Flick
    {
      jump Off;
    }
    exit
    {
      print "Turning off...";
    }
  }
}
```

# Meta-model





# Meta-model





Thank you!