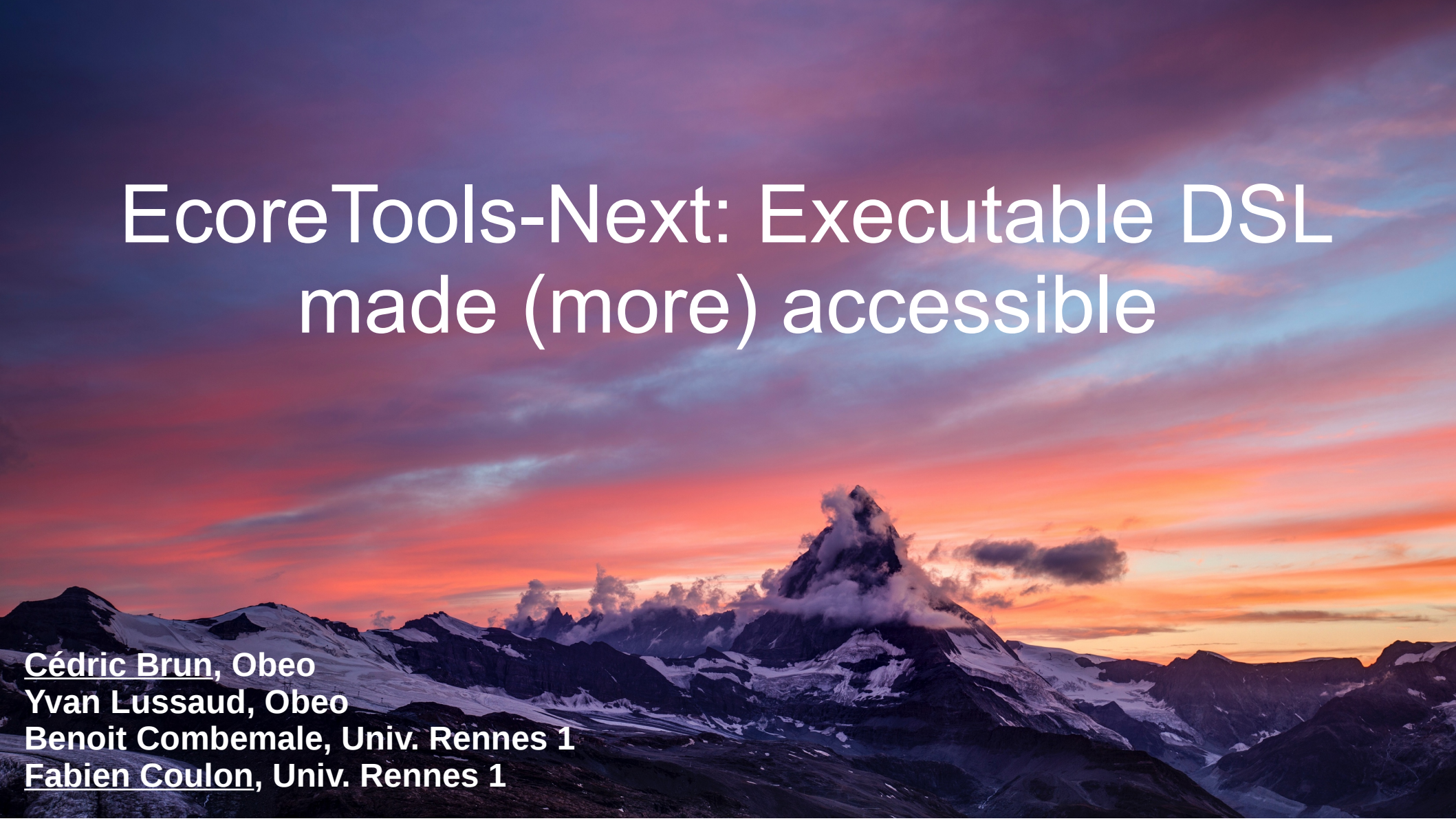
EcoreTools-Next: Executable DSL made (more) accessible



Cédric Brun, Obeo
Yvan Lussaud, Obeo
Benoit Combemale, Univ. Rennes 1
Fabien Coulon, Univ. Rennes 1

eclipsecon France 2017

OBEO

UNIVERSITÉ DE RENNES 1

Executable DSL

- Interpreter
- Compiler
- Model Transformation
- Static Model Analysis
- Etc ...

ALE : Action Language for EMF

Complement an Ecore metamodel

- with Runtime Data
- by implementing EOperation


```
open class HelloWorld {

    String msg := "Hi!";

    override EString greeting () {
        result := self.msg;
    }
}
```

What is ALE ?

Ale is a type of beer brewed using a warm fermentation method, resulting in a sweet, full-bodied and fruity taste.

- *Wikipedia*

What is ALE ?

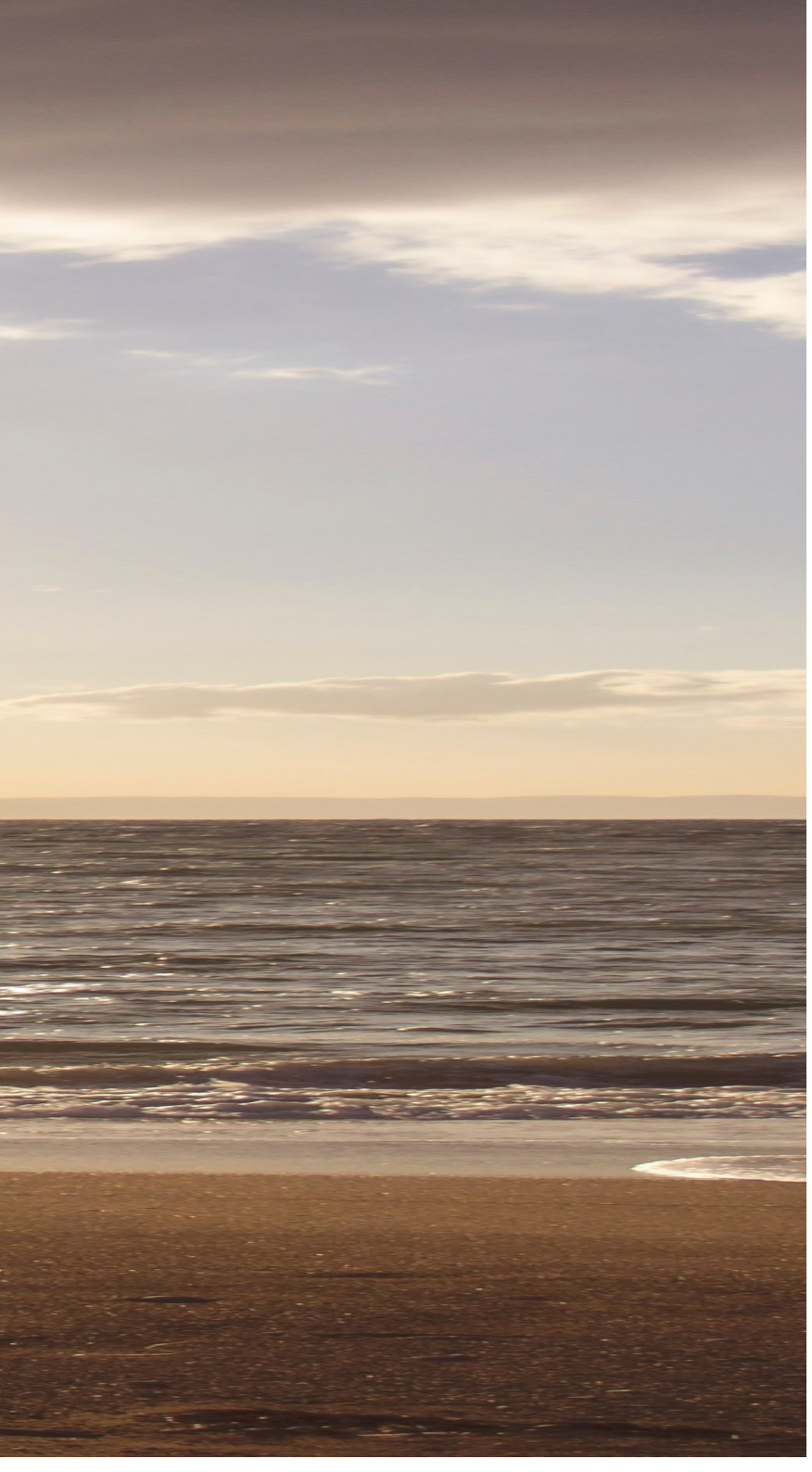
Action Language for EMF

Integrated to EcoreTools

Define behavior
over metamodel

- Interpreted Language
- Static typing, type inference
- Open class support

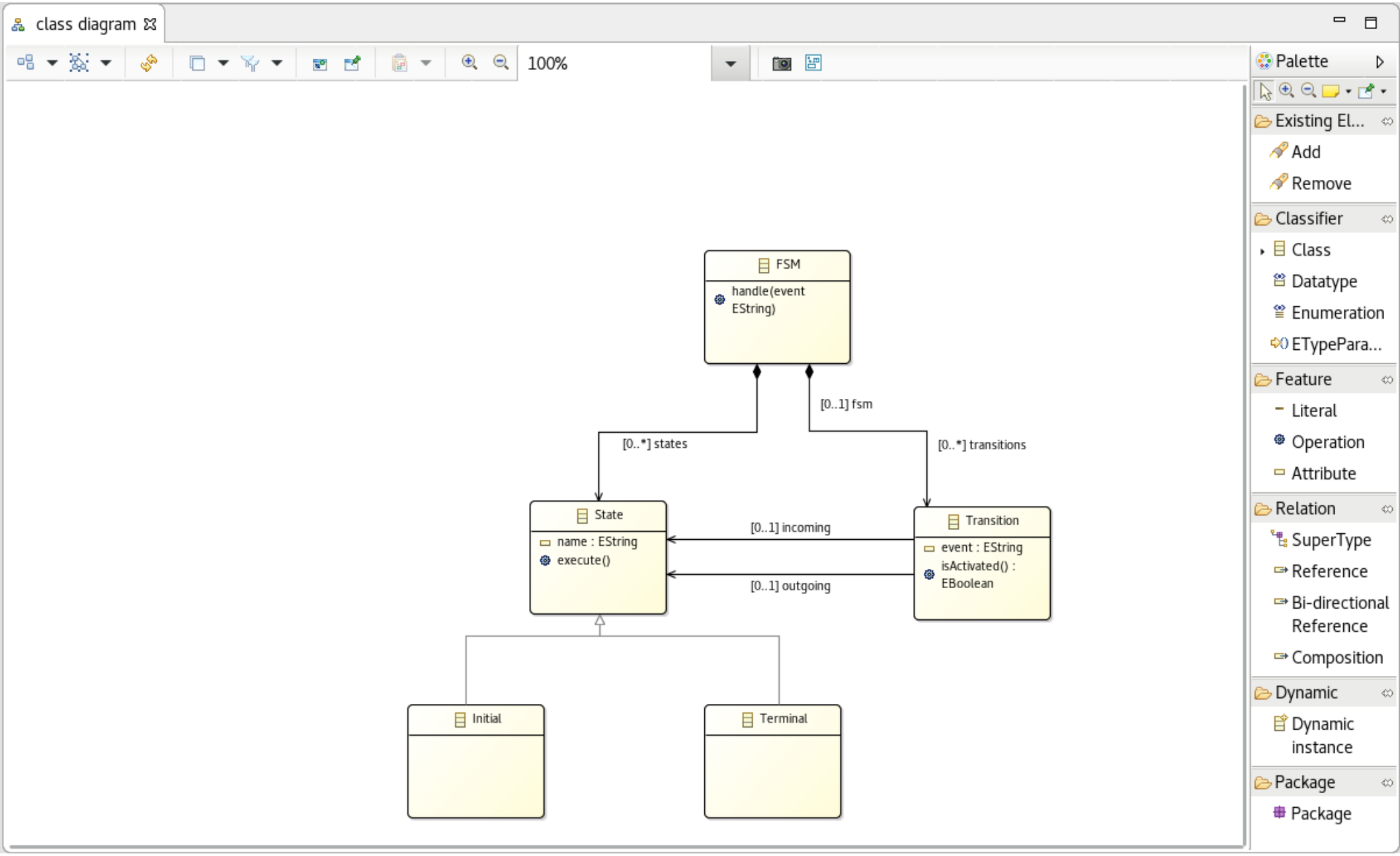
```
open class FSM {
    def void foo() {
        State currentState := … ;
        if (currentState.oclIsKindOf(Initial)) {
            Initial state := currentState; //valid statement
        }
    }
}
```



Design Ecore metamodel is easy

Thanks EcoreTools !

MiniFSM


class diagram
100%
FSM
handle(event EString)
[0..1] fsm
[0..*] states
[0..*] transitions
State
name : EString
execute()
[0..1] incoming
[0..1] outgoing
Transition
event : EString
isActivated() : EBoolean
Initial
Terminal
Palette
Existing El...
Add
Remove
Classifier
Class
Datatype
Enumeration
ETypePara...
Feature
Literal
Operation
Attribute
Relation
SuperType
Reference
Bi-directional Reference
Composition
Dynamic
Dynamic instance
Package
Package

Write implementation is not that easy

Generate code and then:

- Modify generated code
- Configure the code generation
- Use delegation mechanism
- Use the generated Switch
- Use other tools

Write implementation is not that easy

Generate code and then:

- Modify generated code
- Configure the code generation
- Use delegation mechanism
- Use the generated Switch
- Use other tools

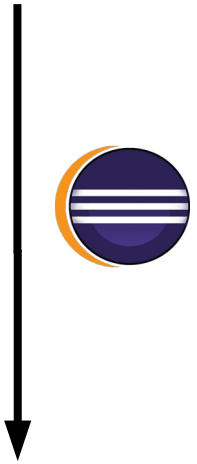
You have to master the compilation chain

Code generation step

Can’t design .ecore & test implementation in parallel

Code generation step

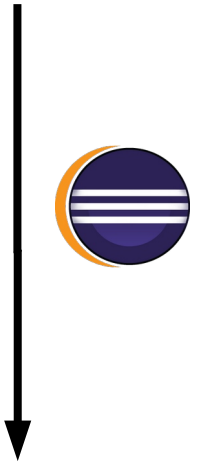
Can’t design .ecore & test implementation in parallel



Run another Eclipse

Code generation step

Can’t design .ecore & test implementation in parallel



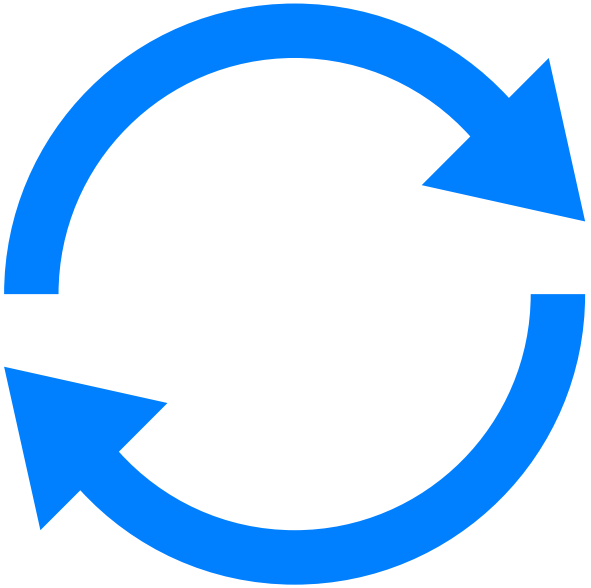
Run another Eclipse

… 20 times a day

Quick Design-Run cycle with ALE

Interpreted behavior

- Just run it !



No code Generation needed

No plugin deployment needed

Demo#1

- Implement an EOperation
- Run an xDSL

ALE is two things:

- **Open Class**
- Action Language

Separate concerns

- Abstract syntax
- Semantic (Runtime data & Behavior)

Ecore

Open Class

```
open class HelloWorld {

    String msg := "Hi!";

    override EString greeting () {
        result := self.msg;
    }
}
```

Demo#2

- Re-open **FSM** EClass
- Add **currentState** feature

ALE is two things:

- Open Class
- **Action Language**

AQL

Acceleo Query Language

- EMF Language in Eclipse Acceleo
  - Navigate & Query model
  - Lightning speed evaluation, strong validation

- Expression

```
(x.tokens > 0) or ( x.tokens = -1)
```

- Lambda

```
myCollection.select(elem | elem.tokens = -1)
```

- Easy extension through services mechanism

```
obj.myNewService()
```

- Interpreted
  - No code generation needed

ALE extends AQL

AQL is an expression language

- No side effects

ALE provides :

- Assignements
- More control flow
- Multi-inheritance
- Runtime data
- Type inference

Demo #3: make executable FSM

in 10 minutes

Summary

- Made an Executable FSM from scratch

Beyound this presentation

- Website

  http://gemoc.org/ale-lang/

- Animated execution & Debugger

  https://www.eclipse.org/sirius/lab.html

Perspectives

- Maturate to industrial grade

  Contribution to EcoreTools

- Compiler: Open class to Java code

Thank you!

Evaluate the Sessions