



Project Bolognese

Solving the Bolognese process scheduling problem

Alon Dolev, Joey Ezechiëls, Volker Lanting
November 16th 2012

Outline

- 1 The Bologna process scheduling problem
- 2 Our approach
 - User interface with Flapjax
 - Constraint programming with OscaR
- 3 What we learned
- 4 Demo
- 5 Questions

Outline

- 1 The Bologna process scheduling problem
- 2 Our approach
 - User interface with Flapjax
 - Constraint programming with OscaR
- 3 What we learned
- 4 Demo
- 5 Questions

The Bolognese process scheduling problem

Parts of the problem:

- Modules
- Categories
- total ECTS
- (optimization goal)



Outline

- 1 The Bologna process scheduling problem
- 2 Our approach
 - User interface with Flapjax
 - Constraint programming with OscaR
- 3 What we learned
- 4 Demo
- 5 Questions

Outline

- 1 The Bologna process scheduling problem
- 2 Our approach
 - User interface with Flapjax
 - Constraint programming with OscaR
- 3 What we learned
- 4 Demo
- 5 Questions

User interface with Flapjax



Outline

- 1 The Bologna process scheduling problem
- 2 Our approach
 - User interface with Flapjax
 - Constraint programming with OscaR
- 3 What we learned
- 4 Demo
- 5 Questions

Constraint programming with OscaR

- Integer Programming
 - ▶ decision variables per possible booking
 - ▶ never double book a Module
 - ▶ book at least `Category.min` ECTS
 - ▶ obtain enough *creditable* ECTS
- Encapsulating side-effects
 - ▶ AST as constraint model
 - ▶ the OscaR monad



Outline

- 1 The Bologna process scheduling problem
- 2 Our approach
 - User interface with Flapjax
 - Constraint programming with OscaR
- 3 What we learned
- 4 Demo
- 5 Questions

What we learned



Outline

- 1 The Bologna process scheduling problem
- 2 Our approach
 - User interface with Flapjax
 - Constraint programming with OscaR
- 3 What we learned
- 4 Demo
- 5 Questions

Outline

- 1 The Bologna process scheduling problem
- 2 Our approach
 - User interface with Flapjax
 - Constraint programming with OscaR
- 3 What we learned
- 4 Demo
- 5 Questions