

Chair of Software Engineering, ETH Zurich

Overview presentation, CIEL, Bordeaux, 9 June 2015

Bertrand Meyer

1

## ETH

Means “Eidgenössische Technische Hochschule” (German)

The only federal university in Switzerland (with a sister university, EPFL, in Lausanne, French-speaking part)

Created 1855

18,000 students, 400 professors

Budget \$1.5 billion

Around #8 worldwide (1<sup>st</sup> in continental Europe) in rankings

2



## ETH famous people

21 Nobel prizes (last one in 2003, chemistry)

This is where Einstein studied; also von Neumann

Bernays (logic)

Pauli (physics)

Clausius (physics)

Kalman (electronics)

Wirth (computer science)

4

## ETH Chair of Software Engineering

Professor (me)

Three senior researchers, several postdocs, PhD students

19 PhD theses completed since 2003 (several more in 2015)

Grants: SNF, Hasler, Gebert-Ruf, Microsoft

Industry collaboration: Eiffel Software, Zurich startups

ERC Advanced Investigator Grant, “Concurrency Made Easy”,  
2012-2017, 2.5 M €

5

## Scope

Help move software development to the next level through:

- Object-oriented development
- Software verification
- Support for concurrent programming

Eiffel as base technology

6

## Concurrency

Underlying model: SCOOP (Simple Concurrent Object-Oriented Programming)

Minimal extension to Eiffel (one new keyword)

Preserves sequential modes of reasoning

Guarantee of no data races

7

## Roboscoop

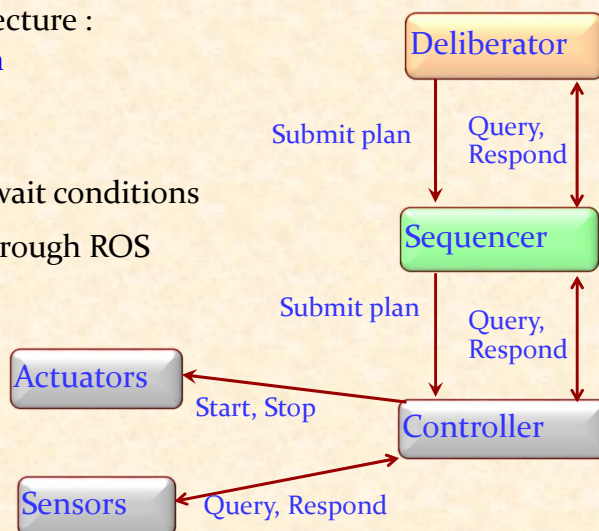
Coordination layer above SCOOP (and hence Eiffel)

Three-layer architecture :

- Deliberation
- Sequence
- Control

Synchronization: wait conditions

Interoperability through ROS  
(external calls)



8

## Robotics: SmartWalker

Smart assistant for elderly people

Hi-tech extension of the regular walker

Autonomous robot with sensors and actuators



Possible functionalities:

- Support while going uphill/downhill
- Navigation during shopping
- Finding a charging station
- Fall detection
- ...



9

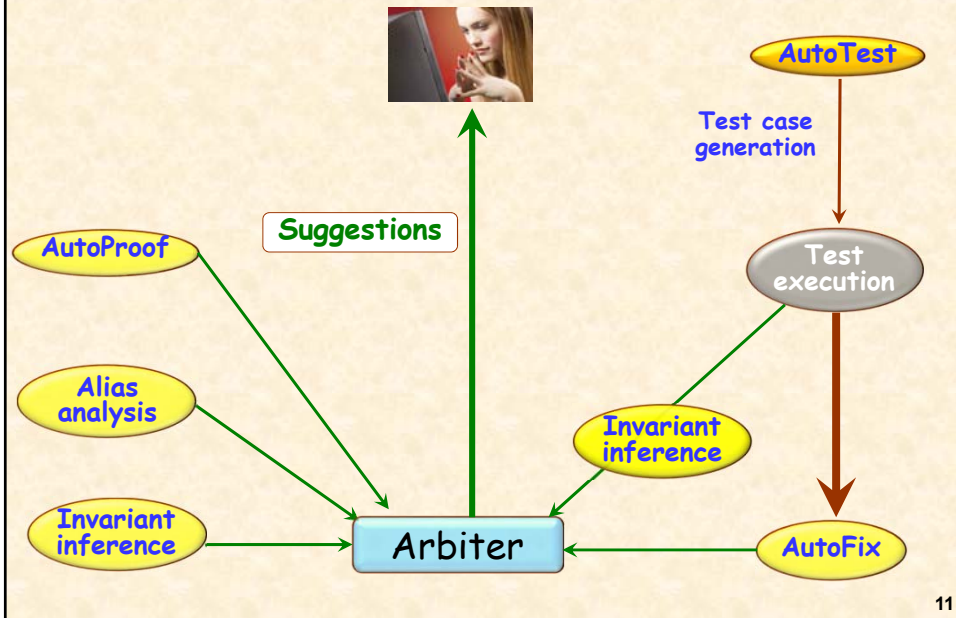
## Verification: EVE

Eiffel Verification Environment

- Open source
- Developed at ETH, others' contributions welcome
- Continuous integration (weekly) with EiffelStudio
- The platform for trying new ideas
- All ETH and other contributions included

10

## EVE: Verification As a Matter Of Course



## AutoProof

(Nadia Polikarpova, Julian Tschannen, Carlo Furia, Martin Nodio)

A practical environment for verifying programs

Supports most of Eiffel, including advanced features (agents, exceptions...)

Full proof of data structure library (EiffelBase 2)

Based on MSR's Boogie and Z<sub>3</sub>

Ownership methodology

"Semantic collaboration" for invariants

Framing

Available on the Web (google for "AutoProof tutorial")

12

## Alias calculus

Rules for determining the aliases of an expression

$a \gg \text{skip}$	$= a$
$a \gg (\text{then } p \text{ else } q \text{ end})$	$= (a \gg p) \cup (a \gg q)$
$a \gg (p ; q)$	$= (a \gg p) \gg q$
$a \gg (\text{forget } x)$	$= a - \{x\}$
$a \gg (\text{create } x)$	$= a - \{x\}$
$a \gg \text{cut } x, y$	$= a - \{[x, y]\}$
$a \gg p^0$	$= a$
$a \gg p^{n+1}$	$= (a \gg p^n) \gg p$
$a \gg (\text{loop } p \text{ end})$	$= \bigcup (a \gg p^n)$
$a \gg \text{call } r (v)$	$= (a [x.r^{\bullet}: v]) \gg  r $
$a \gg \text{call } x.r (v)$	$= x . (x' . (\text{call } r (v)))$

13

## Frame inference: the change calculus

$$l + p ; q = (l + p) + q$$

$$\text{then } p \text{ else } q \text{ end} = p \cup q$$

$$\text{loop } p \text{ end} = \text{Body of } r \quad p \cup p^2 \cup p^3 \cup \dots$$

$$\text{call } r (l) = [r \downarrow [l : r^{\bullet}]] \quad \text{Back-pointer}$$

$$\text{call } x.r (l) = x . (\text{call } r (x'.l))$$

$$t := s = a (\text{Current}).t$$

Image by the alias relation

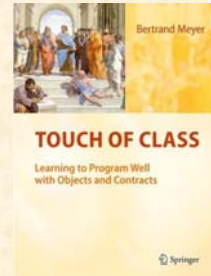
14



## ETH Introduction to Programming course

### “Inverted Curriculum” (outside-in)

- See and reuse lots of software: inspiration, imitation, abstraction
- Reuse through interfaces & **contracts**
- Interesting examples from day one
- Combination of principles and practices
- Programming/design language is Eiffel



[touch.ethz.ch](http://touch.ethz.ch)

Traditional topics too: algorithms, control structures, basic data structures, recursion, syntax & BNF, ...

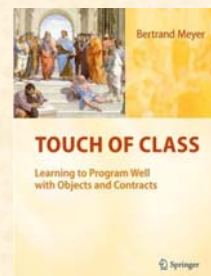
Advanced topics: closures & lambda-calculus, some design patterns, event-driven programming, intro to software engineering...

15

## ETH Introduction to Programming course

### “Inverted Curriculum” (outside-in)

- See and reuse lots of software: inspiration, imitation, abstraction
- Reuse through interfaces & **contracts**
- Interesting examples from day one
- Combination of principles and practices
- Programming/design language is Eiffel



[touch.ethz.ch](http://touch.ethz.ch)

Traditional topics too: algorithms, control structures, basic data structures, recursion, syntax & BNF, ...

Advanced topics: closures & lambda-calculus, some design patterns, event-driven programming, intro to software engineering...

16



# MOOC (available through EdX)

The screenshot shows the EdX website interface for a MOOC. At the top, there is a navigation bar with the EdX logo, links for 'HOW IT WORKS', 'FIND COURSES', and 'SCHOOLS & PARTNERS', and buttons for 'REGISTER' and 'LOG IN'. The main content area features a large header image of two stylized characters, a woman with green hair and a man with glasses, looking at a laptop. Below the image, the course title 'Computing: Art, Magic, Science' is displayed, followed by the course ID '1497x1050'. A brief description states: 'Learn the principles and techniques behind modern Information Technology.' The 'About this Course' section explains that computing is both an art and magic, and that the course will help students see beyond the apparent magic. A sidebar on the right contains a video player for 'Watch the Course Intro Video', course details (School: ETHx, Course Code: CAMSx, Classes Start: 30 Sep 2014, Course Length: 7 weeks), an 'Enroll in CAMSx' button, and a 'Prerequisites' section showing 'None'. At the bottom of the sidebar, there are social media icons and a 'Student Reviews' section with a star rating and the number '411'.

17

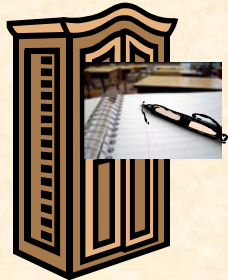
# Distributed development

Update  
(check-out)

Edit

Commit  
(check-in)

Reconcile



18



## CodeBoard

(Christian Estler, Martin Nordio)

Learn to program on the Web!

Used today by an increasing number of MOOCs

21

## Other work

Object-oriented persistence & schema evolution


Development of Eiffel (ECMA committee)

Software Process

Agile methods

22

## Summer school: LASER, September, Elba



2015  
laser  
software  
Elba, Italy



# Concurrency: the next frontiers

12th Summer School on Software Engineering  
September 6 - 12, 2015  
Elba, Italy

lecturers:  
Manfred Broy, TU München  
Maurice Herlihy, Brown University  
Jeff Kramer, Imperial College London  
Bertrand Meyer, ETH Zürich  
Jayadev Misra, University of Texas  
David Parnas, Middle Road Software, Inc.  
Jeannette Wing, Microsoft Research

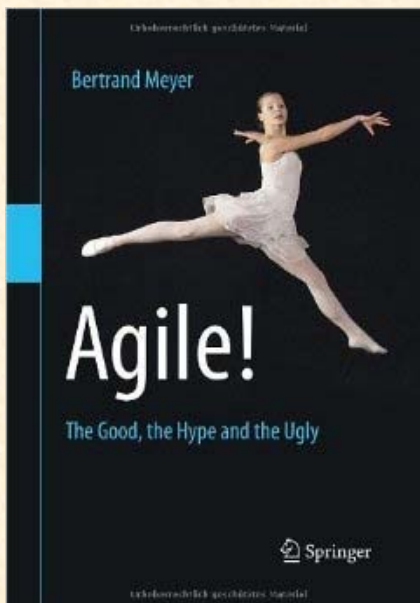
Visit [laser.inf.ethz.ch](http://laser.inf.ethz.ch)  
to register and for  
additional information

organized by:  
Chair of  
Software Engineering



23

## Agile book



Springer, 2014

Tutorial presentation &  
critical analysis of agile  
methods  
from a software  
engineering perspective

24