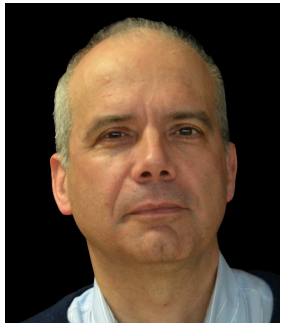
Algebraic and Coalgebraic Methods in Software Development

Alexandre Madeira, UM (coord.)
Dirk Hoffman, UA
Manuel A. Martins, UA
Luís S. Barbosa, UM
José Nuno Oliveira, UM

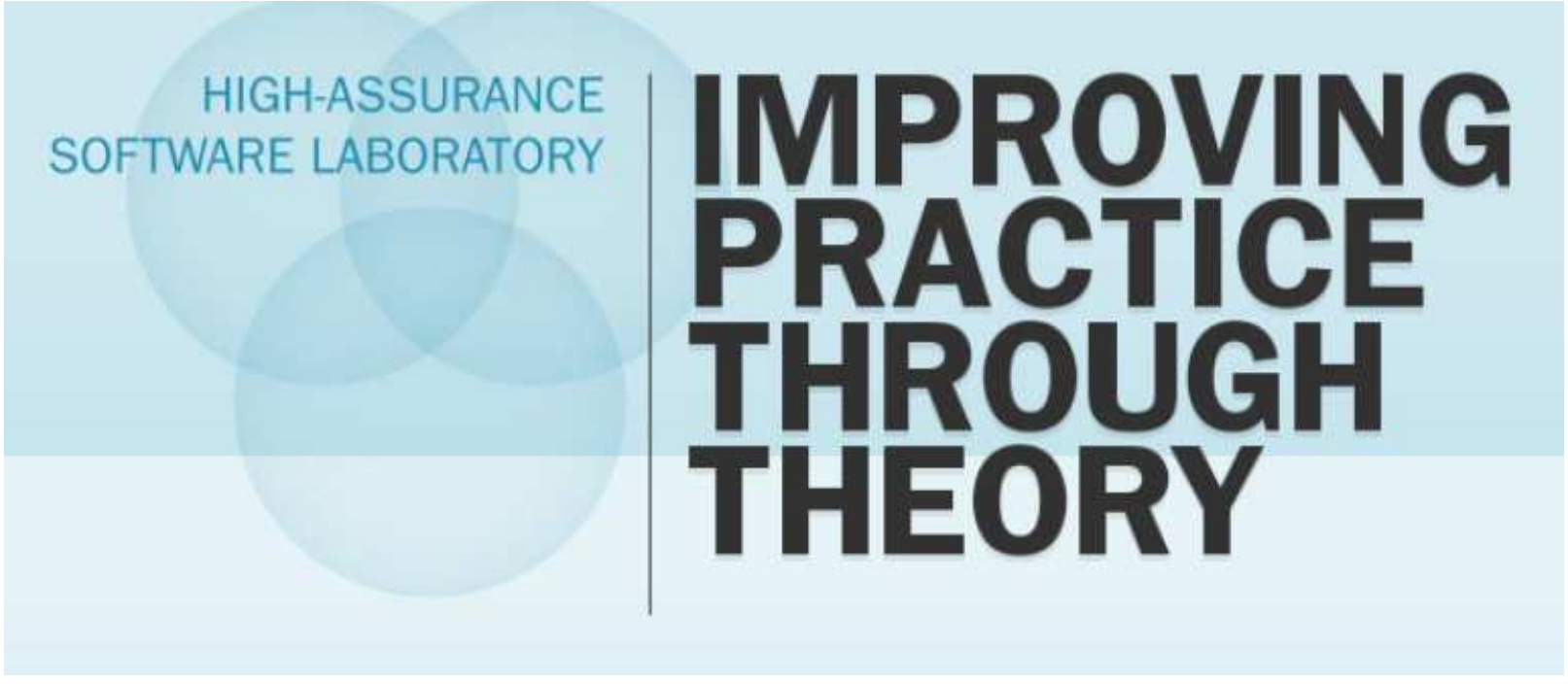
15 September, 2017

Our group (UA/UM)

ACMSD
17/18



Our motto


HIGH-ASSURANCE
SOFTWARE LABORATORY
IMPROVING
PRACTICE
THROUGH
THEORY

Our times

Industry 4.0 to rely on highly **sophisticated** software on an **unprecedented scale**.

Billions, not thousands, of lines of code required to

> **for** *all* **do** { *human* := *robot* }

This big? Yes — complex, therefore risky software.

Software **correctness** and **robustness** thus essential.

Our past

Industrial revolutions made possible by advances in Physics.

Industry 4.0 will rely on **software** on an unprecedented scale (CPS, data-mining, robotics...)

However

- Physics is a **truly scientific** discipline
- Software is a **pre-scientific** discipline

Uupps!

> *As happened in physics, the software sciences need to find a* **unified theory**.

Our aim

This course is about such **unifying theory**.

This will provide you with **better** knowledge of what **software** is all about.

Abstraction essential to better knowledge,

"The purpose of **abstraction** *is not to be* **vague**, *but to create a new semantic level in which one can be* **absolutely precise**." *(E. Dijkstra)*

E.W. Dijkstra (1920-2002)

Our course

Intimidated by

> *Algebraic and Coalgebraic Methods in Software Development ?*

Don't worry — you've seen this before without noticing:

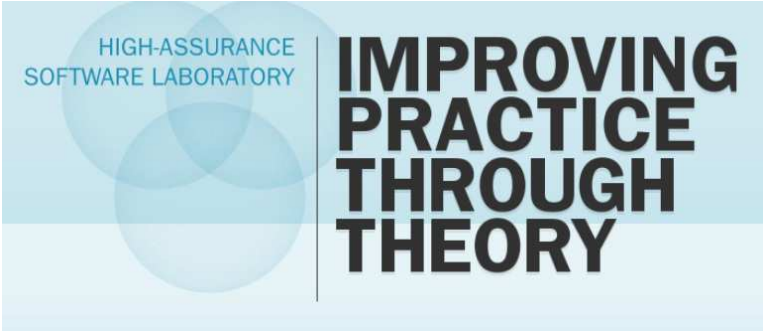
Programs = Algebras + Coalgebras

Examples:

- a **programming language** forms an **algebra**
- an **automaton** forms a **coalgebra**

Want to know why? — come to the course :-)

Our motto, again



Which **practice**?

- data mining
- cyber-physical systems
- software architecture
- risk analysis
- probabilistic programming
- ....

Note, however, that the course is a theoretical — i.e. **foundational** — one.

After all, *"There is nothing more practical than a good theory"* — as Kurt Lewin (1890-1947) once write it!

Course structure

Course plan:

1. **Category theory for computer science** (JNO)
2. **Advanced Category theory** (DH)
3. **Coalgebras and coalgebraic modelling** (LSB)
4. **Algebras and algebraic specification** (MAM)
5. **Modal logics – a logics on-demand approach** (AM)

where

- AM — Alexandre Madeira, UMinho
- DH — Dirk Hoffman, UAveiro
- MAM — Manuel Antonio Martins, UAveiro
- LSB — Luís Soares Barbosa, UMinho
- JNO — José Nuno Oliveira, UMinho

Course structure

Operating mode

- Organized in 5 modules (one per lecturer)
- One lecture per afternoon (with breaks!)

Bibliography

- Papers / books suggested by lecturers

Grading

- Paper recitation + paper resumé (individual assessment).

Last year's website: http://wiki.di.uminho.pt/twiki/bin/view/Education/ACMSD/WebHome