

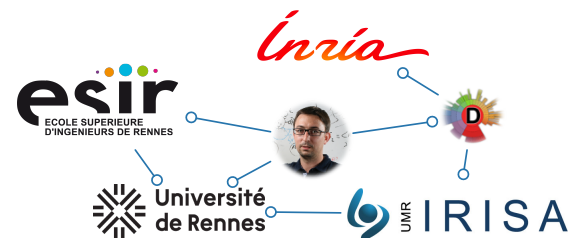
SOFTWARE ENGINEERING (AKA. OMD)

GENERAL INTRODUCTION

UNIV. RENNES 1, ESIR2, 2023-2024

BENOIT COMBEMALE
FULL PROFESSOR, UNIVERSITY OF RENNES, FRANCE

[HTTP://COMBEMALE.FR](http://combemale.fr)
[BENOIT.COMBEMALE@IRISA.FR](mailto:benoit.combemale@irisa.fr)
[@BCOMBEMALE](https://twitter.com/BCOMBEMALE)



WHO WE ARE?

The DiverSE team

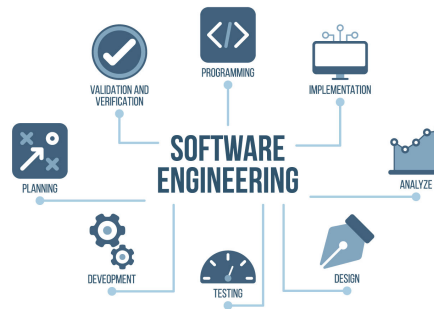
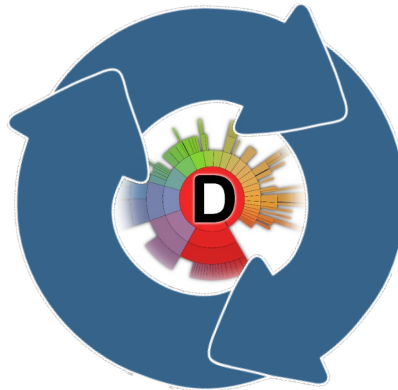


- Inria/IRISA project-team in **Software Engineering**
 - Strong background in Model-Driven Engineering
 - *(software)* architecture, testing, variability, reuse, continuous deployment, adaptation and languages
 - Applied to smart, heterogeneous, and distributed CPS (e.g., IoT, Industry 4.0, embedded systems)
- 12 Prof. and Inria/CNRS researchers, 1 Inria RSE, ~20 PhD, 3 Post-doc, 3 SE
- Deductive and empirical scientific approaches
- Open source software development
- Strong contractual activity (esp., EU and industrial projects)

The DiverSE team



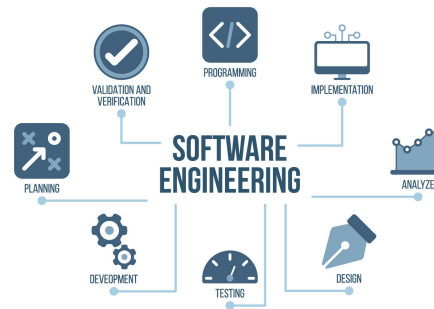
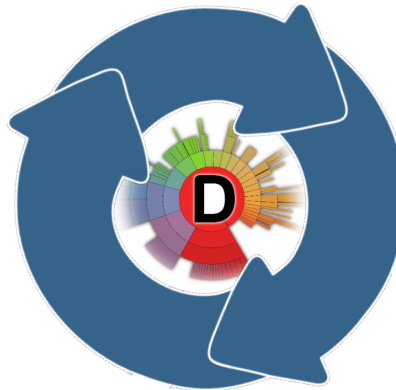
A Software Engineering Group



The DiverSE team

A Software Engineering Group

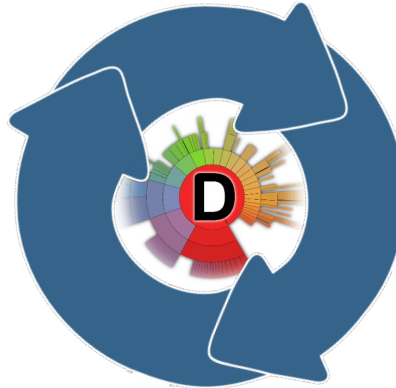
- Multi-engineering approach
- Domain-specific modelling
- High variability and customization
- Software as integration layer
- Openness and dynamicity



The DiverSE team

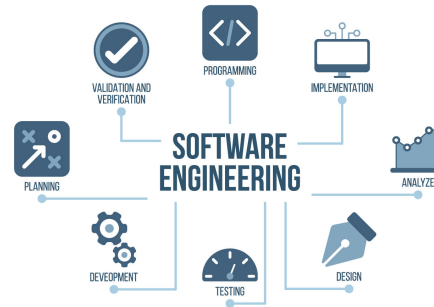
A Software Engineering Group

- Multi-engineering approach
- Domain-specific modelling
- High variability and customization
- Software as integration layer
- Openness and dynamicity



Diversity of...

- stakeholders
- requirements
- concerns
- configurations
- platforms
- environments...

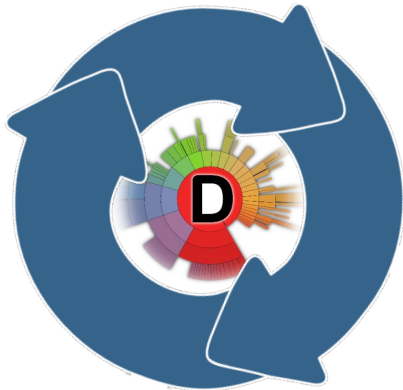


The DiverSE team



- Multi-engineering approach
- Domain-specific modelling
- High variability and customization
- Software as integration layer
- Openness and dynamicity

A Software Engineering Group



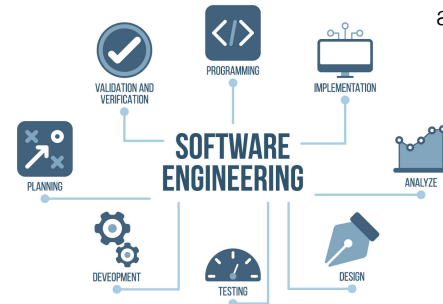
Diversity of...

- stakeholders
- requirements
- concerns
- configurations
- platforms
- environments...



Hyper-agility extends the agility of humans to technical and technological agility. Complex trade off among:

- configuration space,
- release frequency,
- execution platform,
- number of concerns,
- correctness space,
- correctness guarantee,
- Number of function points...



The DiverSE team



Software...

*modeling, architecture, testing, variability, reuse,
continuous deployment, adaptation and languages*



<https://www.diverse-team.fr>

Prof. Benoit Combemale

Agility and safety in the development of complex software-intensive systems



benoit.combemale@irisa.fr

<http://combemale.fr>

[@bcombemale](#)

Full Professor of Software Engineering @ University of Rennes
Computer Science Department (former head) @ Engineering School ESIR
Co-head of the DiverSE team @ IRISA & Inria

Research interest in Software Engineering, incl.: Model-Driven Engineering, Software Language Engineering, Domain-Specific Languages, Software-Product Lines, Software Validation & Verification, Resilience Engineering, Cyber-Physical Systems, Scientific Computing, Sustainable Digitalization, ICT for Sustainability.

Application domains: (smart) cyber-physical systems (transport, defense), internet of things (telecommunication, cities/farming, industry 4.0) and environmental sciences (climate change, sustainability).

Chief Science Advisor at TwiinIT

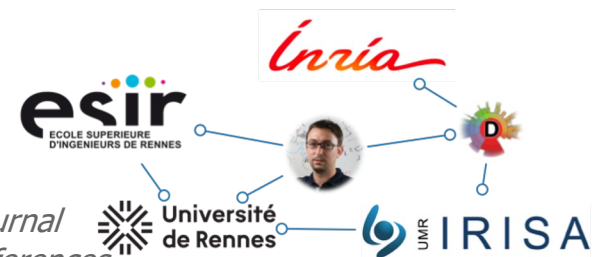
Scientific Advisor in Software and Systems Engineering

Collaborations with Airbus, Safran, Thales, Orange, CEA, DGA, Obeo, Akka...

Working group and open-source project leader at the Eclipse Foundation

Editor-in-Chief of the SoSyM Journal, and Deputy Editor-in-Chief of the JOT Journal

Steering Committee member of the MODELS, SLE and ICT4S international conferences



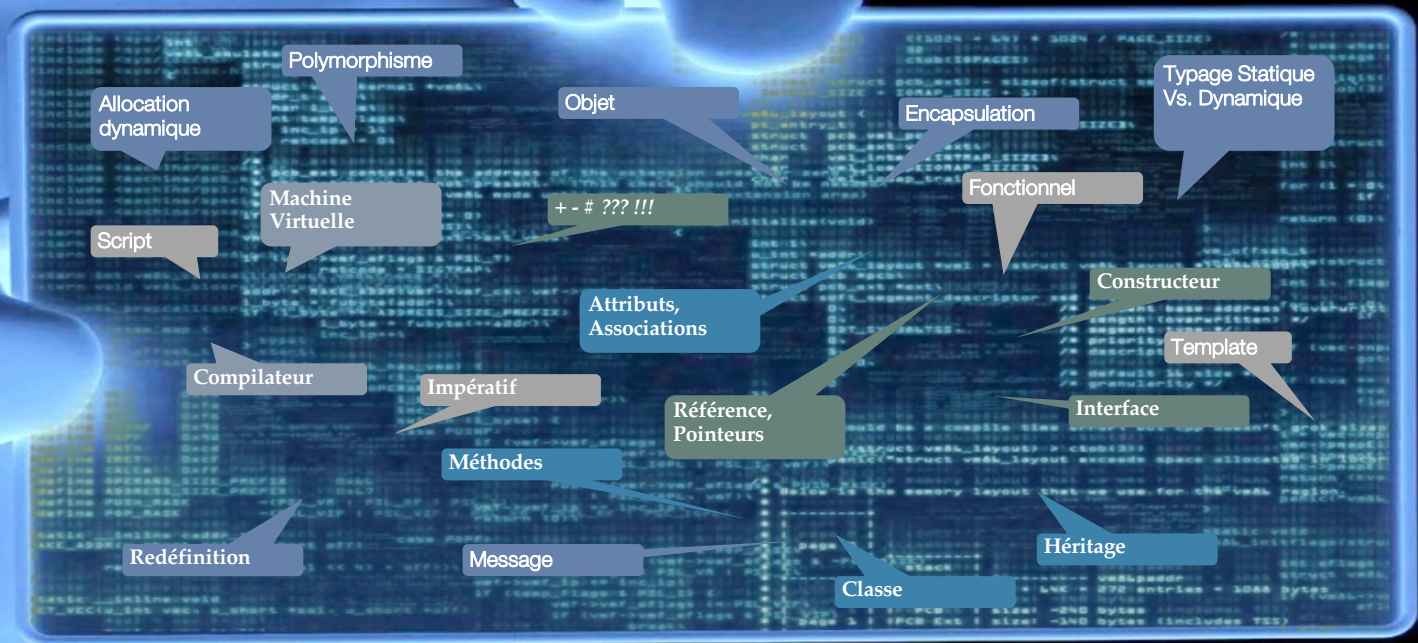
WHO ARE YOU?



OMD is about "Software Engineering"...

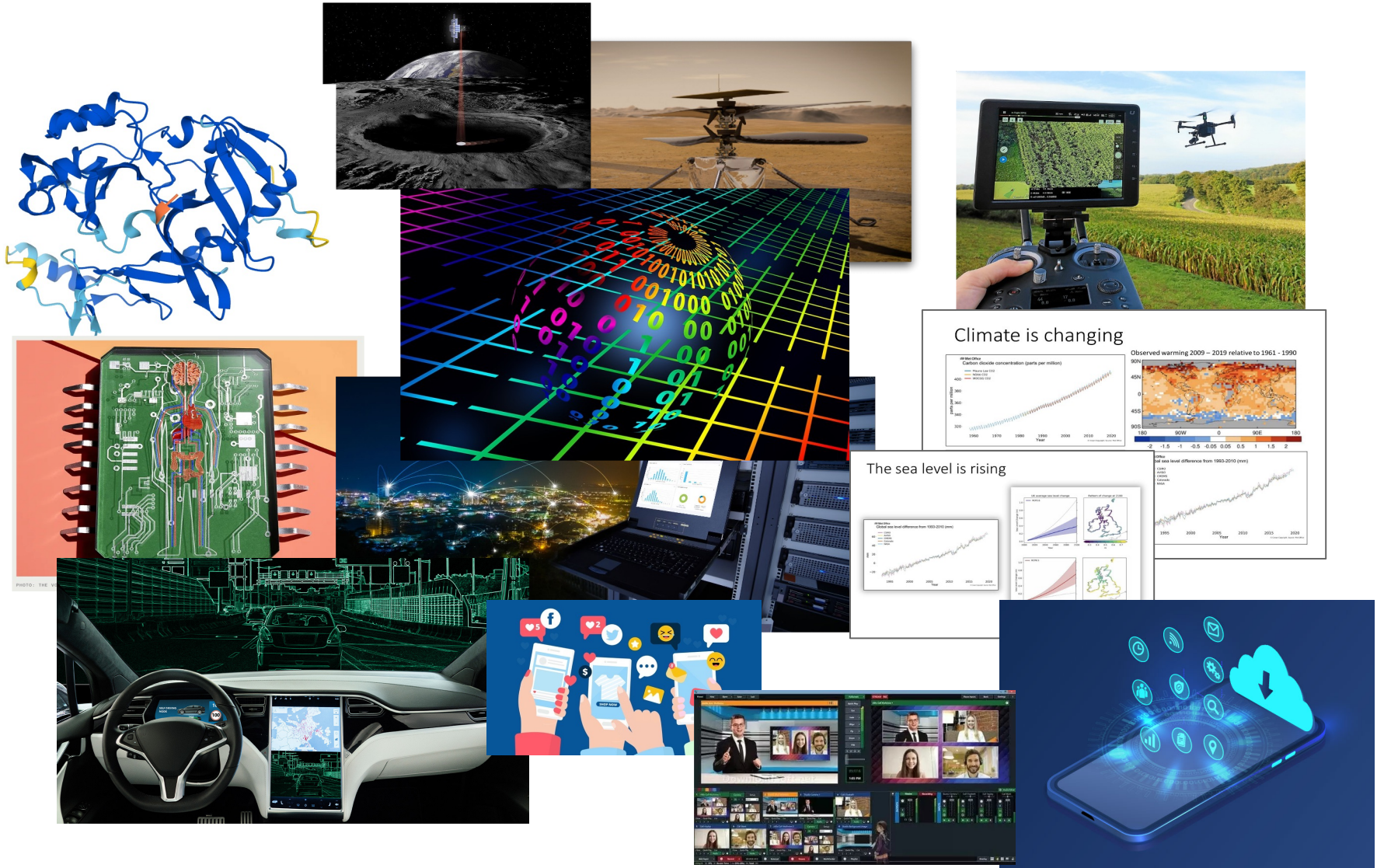
WHY SOFTWARE ENGINEERING?

Vous maîtrisez la programmation...



... orientée objet !

Software is Eating the World



SOFTWARE INTENSIVE SYSTEMS

*Autonomic
Computing*

*Internet of
Services*

*Pervasive
Computing*

*Internet of
Things*

*Ultra Large
System*

etc.

*Embedded
Systems*

*Real-Time
Systems*

*Critical
Systems*

```
order1 = ...
order2 = ...

#ifdef TRACE
if (!fopen("ppenc.doc", "wb"))
{
fprintf(stderr, "Error: Can't
exit(2);
}
#endif

/* allocate 'order+1' elements
... is used to stop
... the t
```

Et vous souhaitez passer
de la technique à l'ingénierie...

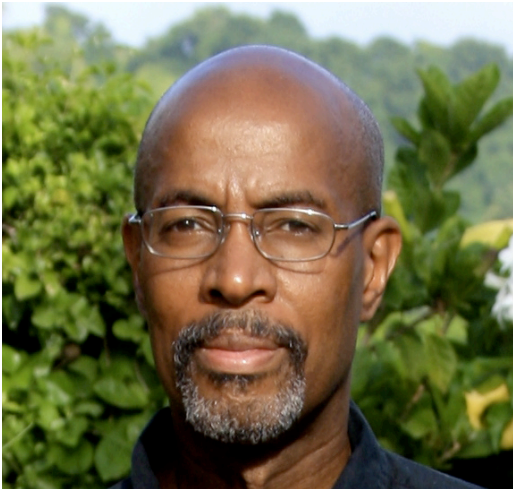
...



From "Teaching Programming Students how to Model: Challenges & Opportunities"
Prof. Robert B. France, EduSymp @ MoDELS, Oct. 2011



“Use of modeling techniques distinguishes a software engineer from a software developer (or programmer)”



“The earlier you start to code the longer it takes to complete the program”

“A good modeler is a good programmer; a good programmer is not always a good modeler”

“Learning a programming language is easy, learning how to program is difficult”

Prof. Robert B. France

Colorado State University

<http://www.cs.colostate.edu/~france/>

Students at work ...



From "Teaching Programming Students how to Model: Challenges & Opportunities"
Prof. Robert B. France, EduSymp @ MoDELS, Oct. 2011

Principles, Techniques, Methods, and Tools for Software Engineering

Basics!



Software Engineering

The production of operational software satisfying defined standards of quality...

... includes programming, but is more than programming!

The five components of Software Engineering [Meyer]:

- **Describe:** *requirements, design, specification, documentation...*
- **Implement:** *modeling, programming*
- **Assess:** *testing and other V&V techniques*
- **Manage:** *plans, schedules, communication, reviews*
- **Operate:** *deployment, installation...*

Software Engineering @ ESIR

Techniques

Methods

Approche générative, ingénierie des langages, sécurité...

Principles and Tools

ESIR3

Analyse et conception, modélisation, patrons de conception & implémentation, architecture logicielle, test, programmation web et mobile, devops...

Principles and Tools

ESIR2

Programmation: du procédural... à l'objet

Principles and Tools

ESIR1

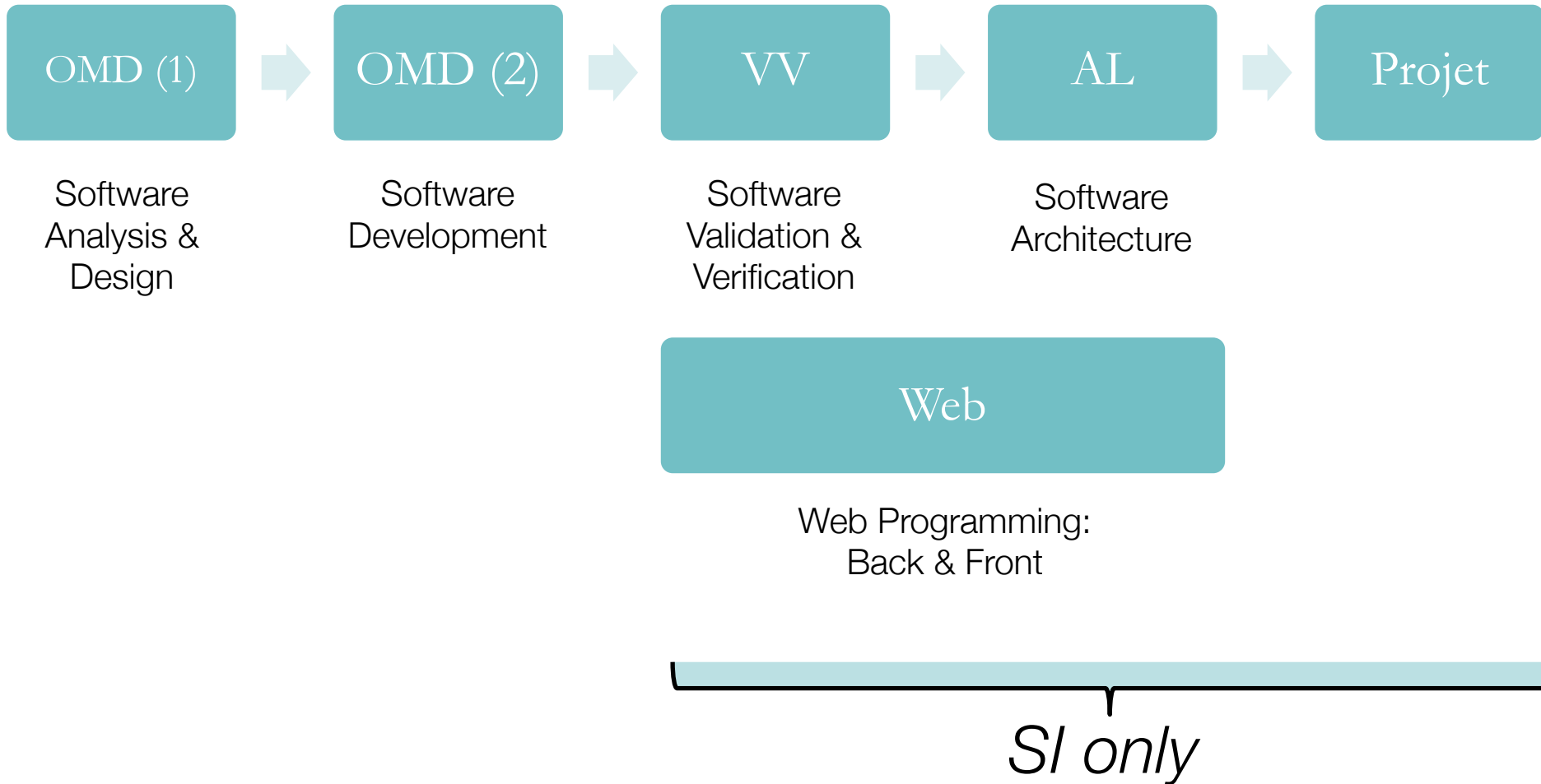


Software Engineering @ ESIR 2 & 3

	Option SI	Option IN	Option IoT
Semester 7	<ul style="list-style-type: none"> VV (24) AL (24) Web (48) Proj1 (24) 	<ul style="list-style-type: none"> AC: Algorithms and Complexity (48h) -- Resp.: P. Maurel TLC: Language Theory and Compilation (48h) -- Resp.: F. Lamarche OMD: Development Tools & Methods (48h) -- Resp.: B. Combemale BINP (48) IGI (24) TIA (48) 	<ul style="list-style-type: none"> TWeb (48) SRIO (48) FabLab1 (48)
Semester 8	<ul style="list-style-type: none"> ProgM (36) and DevOps (36) AA (48) and DE (24) SPP (36) and SDE (36) Proj2 (24) 	<ul style="list-style-type: none"> SI (48) AA (48) ACV (36) MATI (48) ANIM1 (48) V3D (36) 	<ul style="list-style-type: none"> SCP1 (28) SEM (36) ProgM (36) and DevOps (36) SNRF (48) FabLab2 (48)
Semester 9	<ul style="list-style-type: none"> ASE (48) and SSD (24) DS (24) and HPC (24) AI (24) and DMB (24) Sem (24) 	<ul style="list-style-type: none"> COV (24) HPC (24) Vrob 1/2 (24+24) IMEDx 1/2 (24+24) AMM (24) DLV (24) ANIM2 (24) <p>choix de 8/9 modules > 15 étudiants</p>	<ul style="list-style-type: none"> SCP2 (28) CloudIoT (60) MIoT (48) RSM (36) ResIoT (36)



Software Engineering @ ESIR2 / S7



Objectifs de OMD (1)



- Appréhender la totalité d'un projet et son processus de développement dans le domaine du numérique
- Comprendre les fondamentaux de la modélisation, sa continuité tout au long du processus (exigences, analyse, conception, développement, test et vérification)
- Acquérir les connaissances techniques pour mettre en œuvre une modélisation efficace dans un projet de système informatique
- Acquérir les connaissances spécifiques à la gestion de projet pour les systèmes informatiques

Organisation de OMD (1)



- CM avec B. Combemale
- TP avec H. Feuillâtre et P.-V. Besson
- Page du cours : <http://combemale.fr/course/esir/esir2>

Evaluation de OMD (1)



- 2 travaux pratiques en binôme (50%) :
 - TP #1 : analyse/conception (25%)
 - TP #2 : conception / implémentation (75%)
 - Un conducteur donnant des éléments de solution sera distribué le jours de chaque *deadline*.
- 1 partiel : première partie du partiel final d'OMD (50%)

Have fun!

How to contact professors

- Preferably during teaching hours
- Otherwise by sending an email with:
 - Your **academic email** (or a comprehensible one)
 - An **explicit object** starting with “[ESIR2-OMD] ...”
 - A **full signature** including your name, specialty, group/partners (for issues related to the lab sessions), etc.
 - A **comprehensible description** of your issues
 - The **related files** (diagram, workspace...)
 - Ex: to export an Eclipse project, use the dedicated facility:
 - » Project’s contextual menu (by right clicking) > export > Archive File...

Compliance with these rules implies a guaranteed response
compliance_with_these_rules => a_guaranteed_response



Key takeaways?

