

FROM RESEARCH TO INDUSTRY



list

SYSTEM MODELING INTRODUCTION

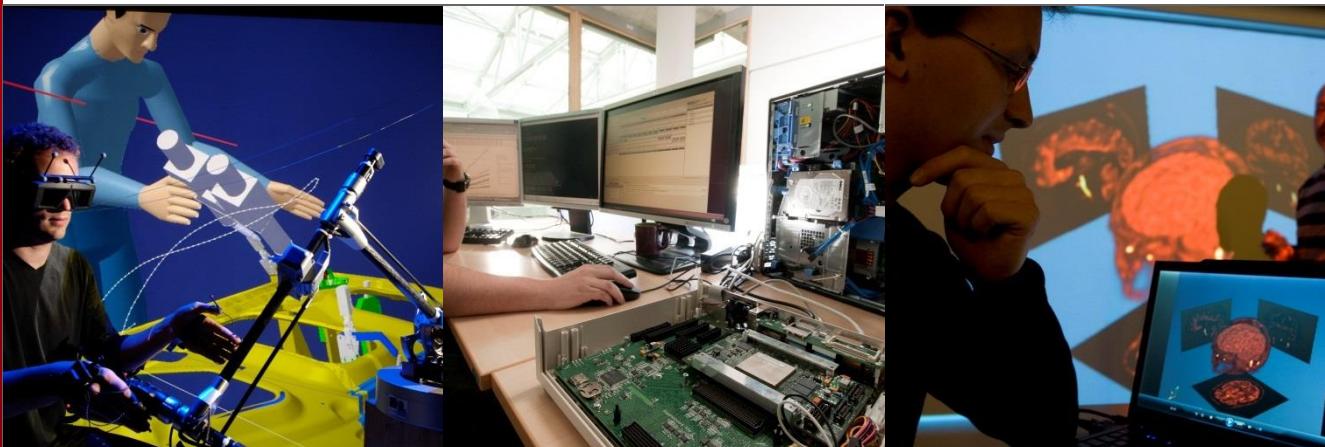
Introduction on UML for Industrial Systems

François Terrier, Shuai Li, Jérémie Tatibouët, Sébastien Gérard, [Asma Smaoui](#)

{first_name}.{last_name}@cea.fr

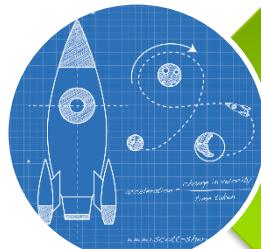
Copyright (c) 2015, CEA LIST, All rights reserved.

Redistribution and use (commercial or non-commercial), of this presentation, with or without modification, is strictly forbidden without explicit and official approval from CEA LIST





Introduction



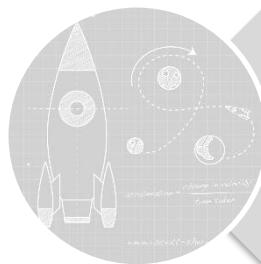
Basis is Modeling



Modeling with Which Language?



Introduction

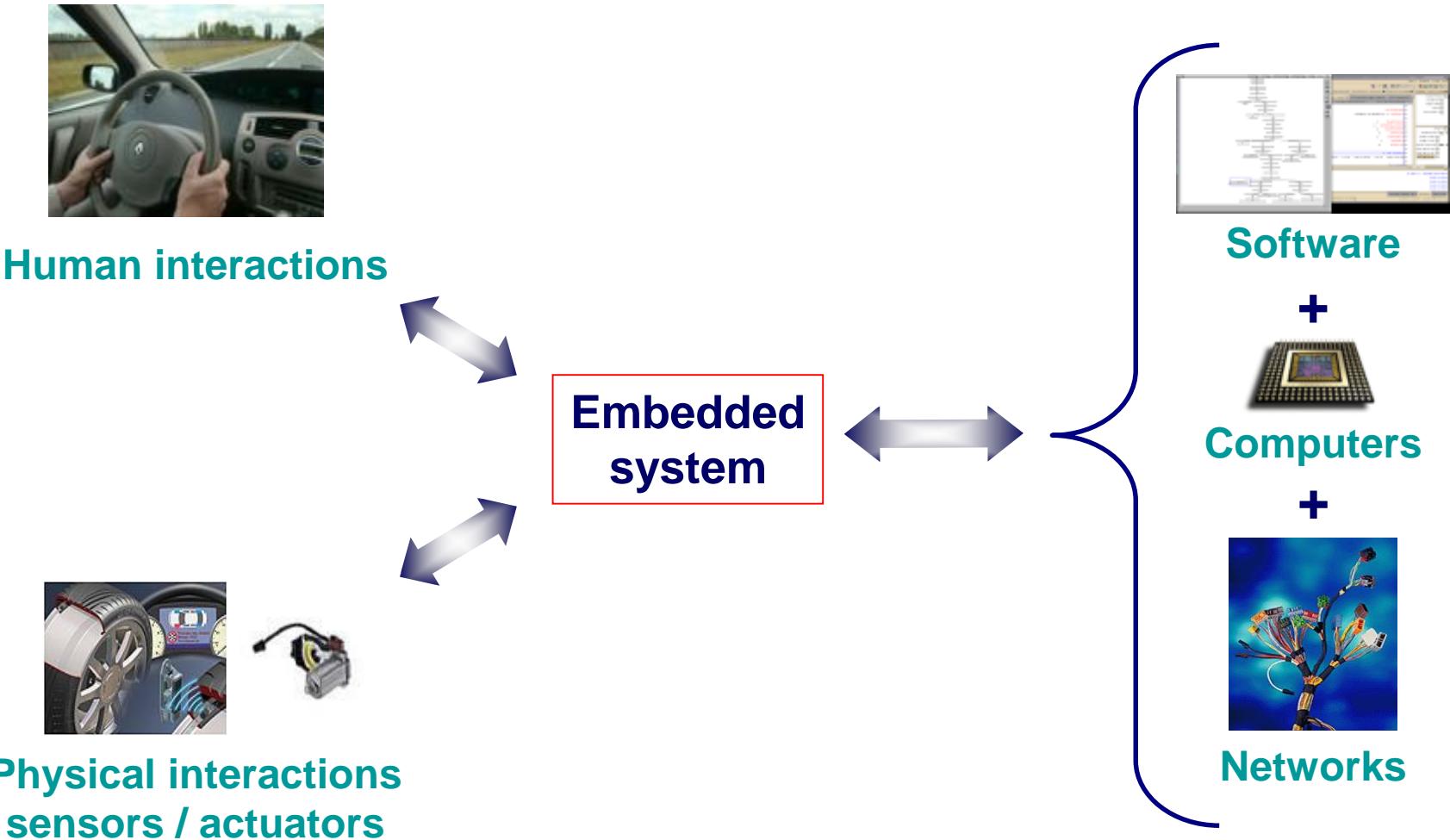


Basis is Modeling



Modeling with Which Language?

→ Complex and heterogeneous systems responding to real-world events

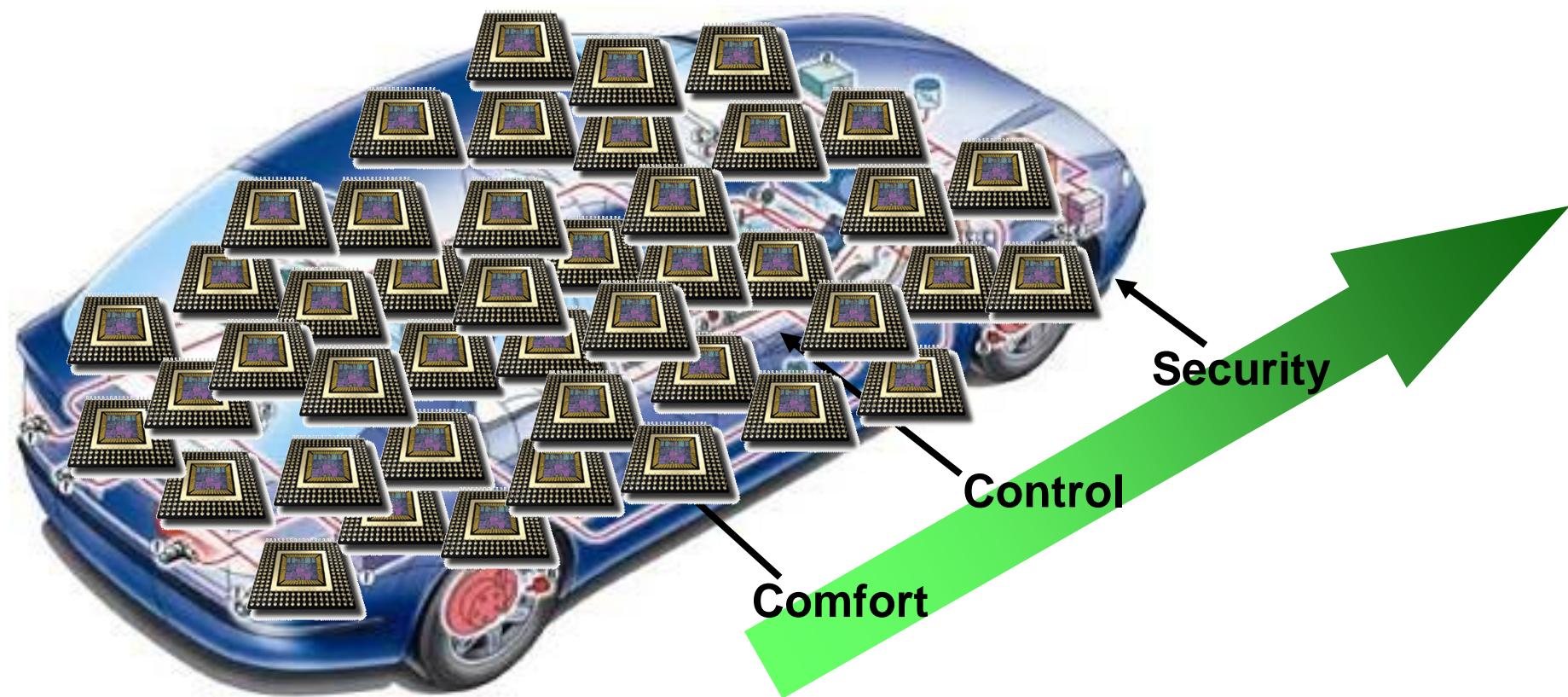




Systems are everywhere,
deeply involved in our daily life
and inter-connected.



Some numbers...



- *Tenth of interconnected processors*
- *Hundreds of processing in parallel*
- *Thousands of exchanged data*

GM Recalls 50,500 2011 Cadillac SRXs Over Airbag-related Software Glitch

By Robert Charette
Posted 22 Jun 2011 | 12:46 GMT



GM is recalling 50,500 Cadillac SRXs due to a software glitch that may not allow airbags for passengers sitting in the right side of the front seats in a crash, reports a [blog post](#) at Zacks Investors.

According to GM, the post says:

"...the front passenger-side airbag will not deploy if it detects an occupant in the seat, even though GM has been invited to update its airbag software to correct this issue, only to learn that the project is still in the works. GM's recall notice still alludes to this issue."

SPECIAL REPORT



The country's biggest phone company paid out \$2.7 million to some 47,000 customers who were overcharged after a software glitch meant people hit their data limits early, the antitrust regulator said in a statement. Telecom and the commission reached a settlement after the phone company acknowledged the fault and sought to compensate its customers. A fair trading act.

Released by:
Technical Operations
International Council on Systems Engineering (INCOSE)

1 SYSTEM FAILURE

Journal of SOFTWARE ENGINEERING VOL. 34 NO. 6 JUNE 2013

What Industry Needs from Architecture Languages: A Survey

Senior Member, IEEE, Henry Tang, Member, IEEE



CHAOS

Editorial concept, languages, and tools in use

What applies to software architecture, what applies to architectural languages has been introduced under [4]. If fully fill in the user's perceived needs in architectural description, the study analyzes practical issues. Early

Engineering Automotive Software

Abstract] the amount of software in cars grows exponentially, driving forces of its development are the availability of cheap and more powerful hardware, as well as the demand for innovation through new function-based functionality that are significant to various challenges in the automotive industry. These concern methods, tools, models, products and structures of labor, logistics, materials and longer term in division of labor: the paper pinpoints the diversification of the domain, this paper pinpoints the diversification of the domain, characterizes the challenges of automotive software engineering.

1. INTRODUCTION

Just 30 years ago the automotive industry witnessed the first deployment of tiny little bits of software in can. It was installed in the engine and, in particular, the ignition, clutch, transmission, and brakes. These were simple logic control systems. Independent of single dedicated controllers, a typical node can be a few

SYSTEM TECHNOLOGY

America Europe Latin America Mid-East US & Canada Business Health SciEnvironment

Share +1

last updated at 21:44 GMT

Many users complain of fresh crash



Blackberry users have complained of a fresh crash hours after a company which makes the smartphones, RIM, said all services were "operating normally".

Twitter angry users reported renewed issues with their handsets due to an inability to send messages and email.

An initial blackout saw Blackberry services across Europe, the Middle East and Africa disrupted - but that has now spread to Latin America.

Motorola said the problems were caused by core and back-up switches.

A tweeter summed up the mood of many: "Blackberry server down AGAIN!!!! You have got to be kidding me!!!!"

Related Stories

Can the iPhone still scare rivals? Microsoft services fail

Android 'most popular' purchase

nodes. Only a minimum of abstraction was applied. The focus was on sufficient resource to consume the programming language C.

The basic architecture is can was developed by ECUs for dedicated take together with how they mainly connected to ECUs to set apart exchange the information. Given were the interaction, started to introduce ECU, connected by communication such functional such ECU to the network, custom in connection to the system integration.

ECU, connected by functional

communications, such ECUs to the system, custom in connection to the

system integration.

Given were the interaction, started to introduce ECU, connected by functional

communications, such ECUs to the system, custom in connection to the

system integration.

- Communication issues between numerous and various stakeholders.
- Time-to-market pressure vs. higher quality level.
- Ambiguous or incomplete descriptions of system.
- Non-availability of expertise for complex analysis.
- Manual-based methodologies.

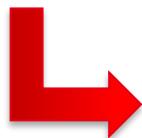
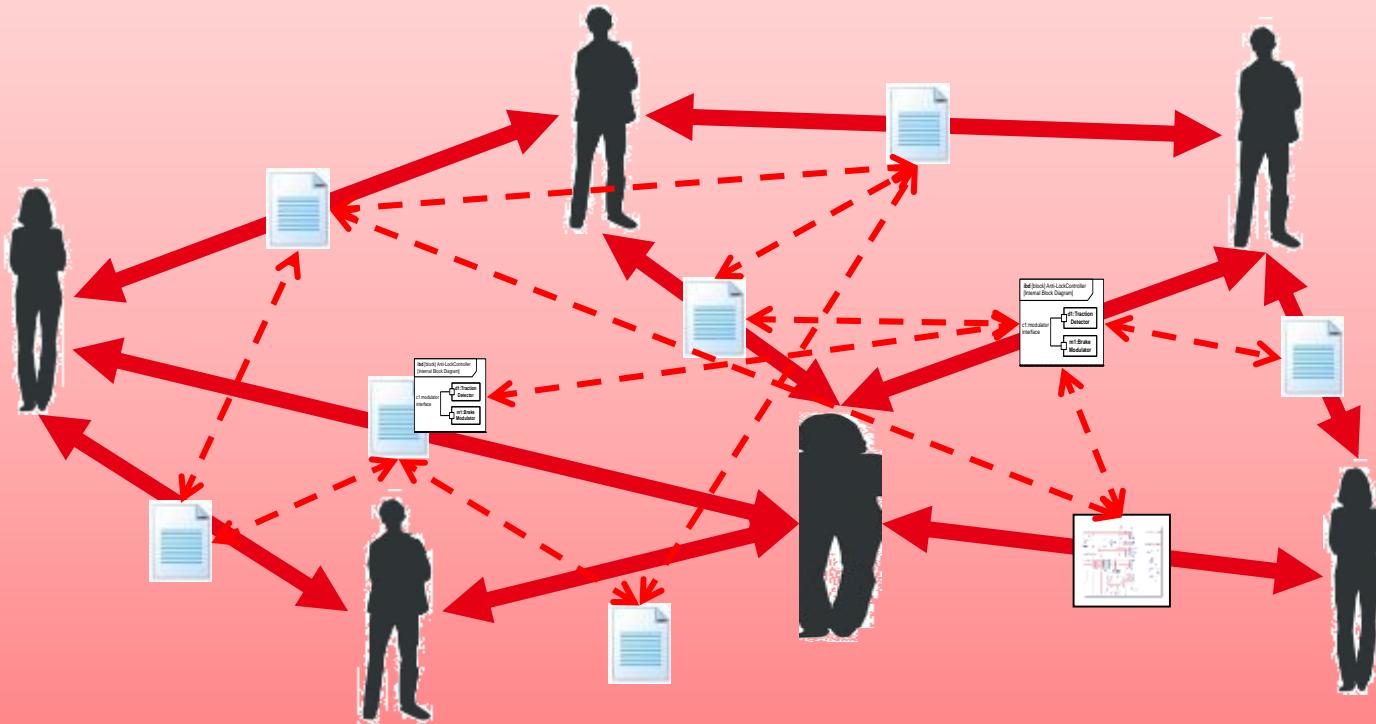
(Note: this list is of course not exhaustive)



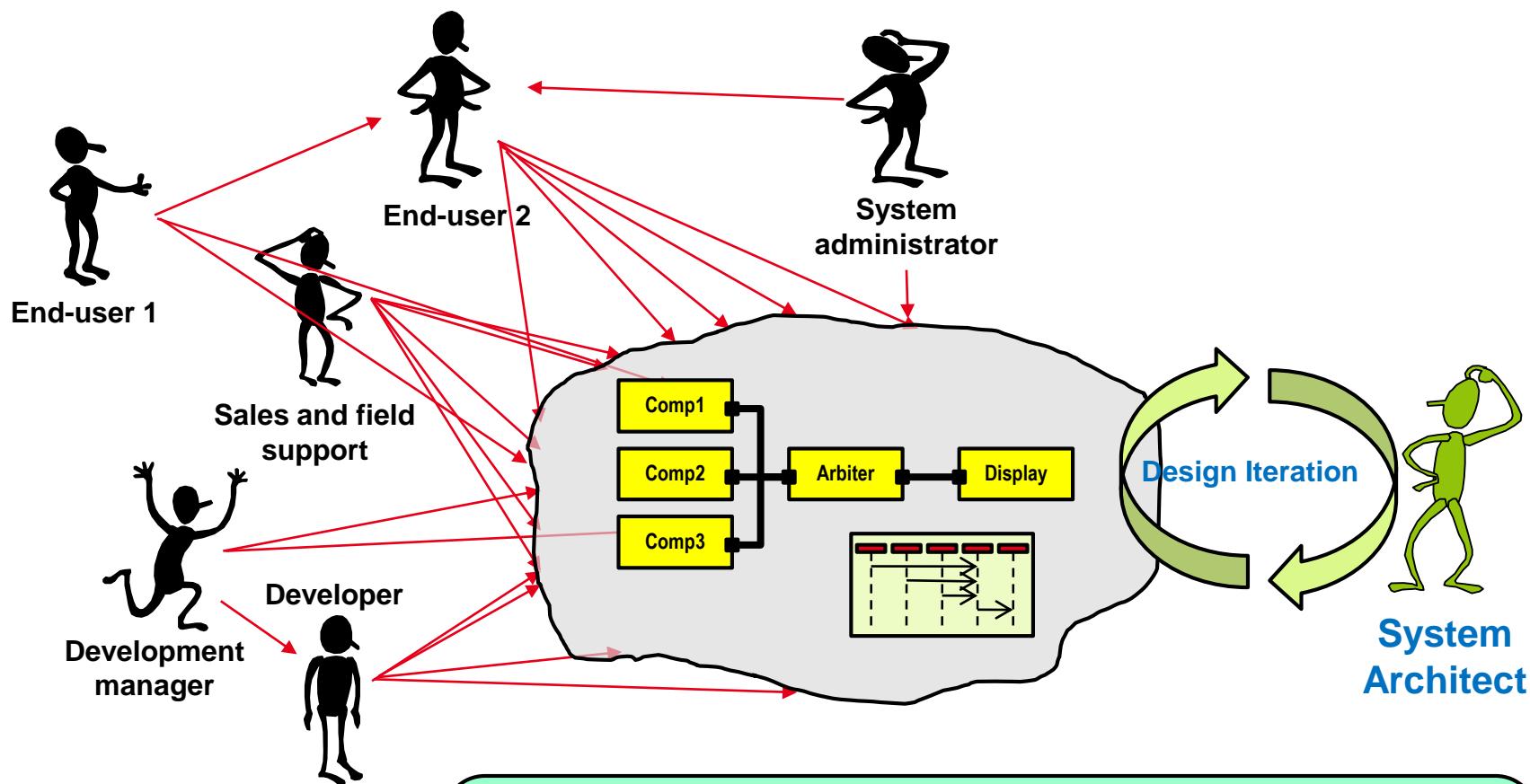
Complexity, Complexity, Complexity, Complexity, Complexity . . .



Traditional Development Approach



Unreliable / Inefficient / Non-scalable



Many requirements conflicts and necessary tradeoffs are only detected through analysis of candidate architectures.

Design the system as a whole rather than as an aggregate of separately designed sub-systems

- Provides possibility to ensure system integrity
- Requires a “big picture” approach
- → An architecture specification

One definition of Architecture [IEEE Standard 1471]:

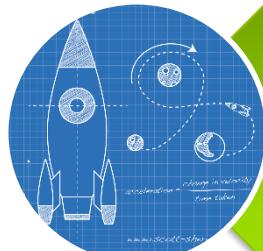
- “The fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution



“ To architect is to model ”



Introduction

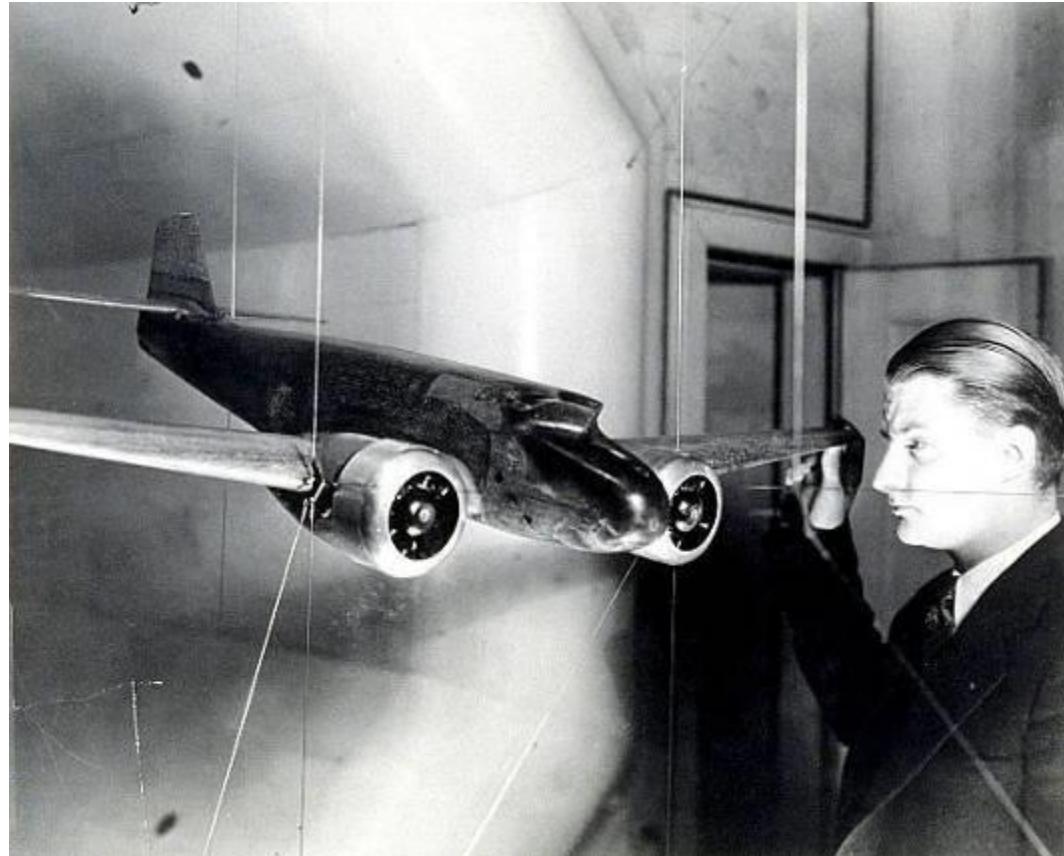


Basis is Modeling



Modeling with Which Language?

Probably as old as engineering



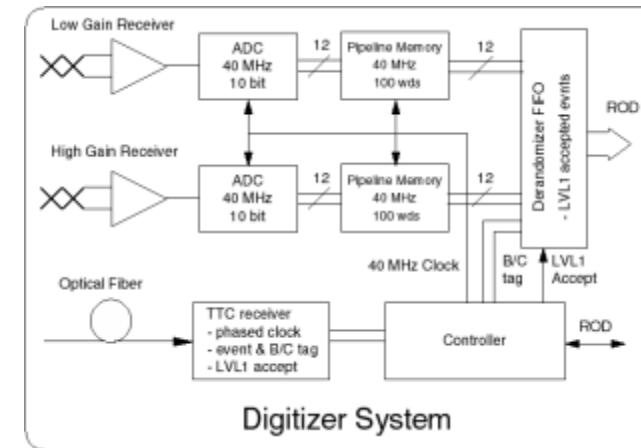
Extracted from B. Selic presentation

Engineering model

- A reduced representation of some system that highlights the properties of interest from a given viewpoint



Modeled system



Functional Model

- We don't see everything at once
- We use a representation (notation) that is easily understood for the purpose on hand

```

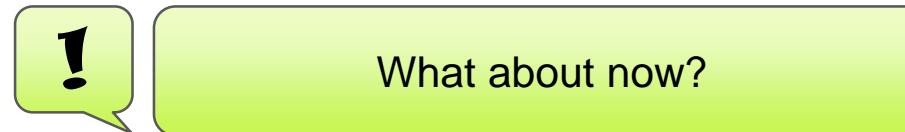
SC_MODULE(producer)
{
sc_outmaster<int> out1;
sc_in<bool> start; // kick-start
void generate_data ()
{
for(int i =0; i <10; i++) {
out1 =i ; //to invoke slave;}
}
SC_CTOR(producer)
{
SC_METHOD(generate_data);
sensitive << start;}};
SC_MODULE(consumer)
{
sc_inslave<int> in1;
int sum; // state variable
void accumulate (){
sum += in1;
cout << "Sum = " << sum << endl;}
```

```

SC_CTOR(consumer)
{
SC_SLAVE(accumulate, in1);
sum = 0; // initialize
};
SC_MODULE(top) // container
{
producer *A1;
consumer *B1;
sc_link_mp<int> link1;
SC_CTOR(top)
{
A1 = new producer("A1");
A1.out1(link1);
B1 = new consumer("B1");
B1.in1(link1);}}
```

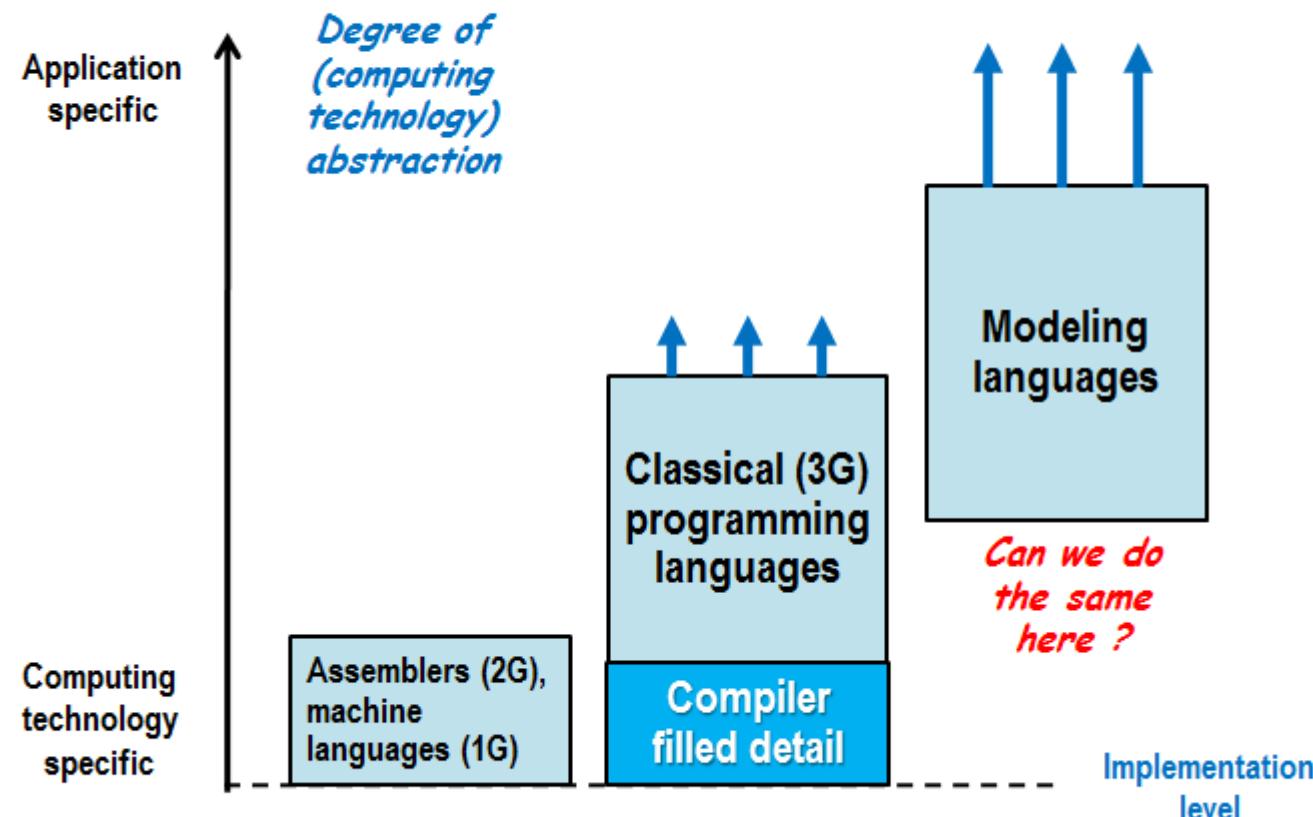


Can you spot the architecture?



Extracted from B. Selic presentation

- Much of the evolution of computer languages is motivated by the need to be more human-centric (i.e., descriptive)

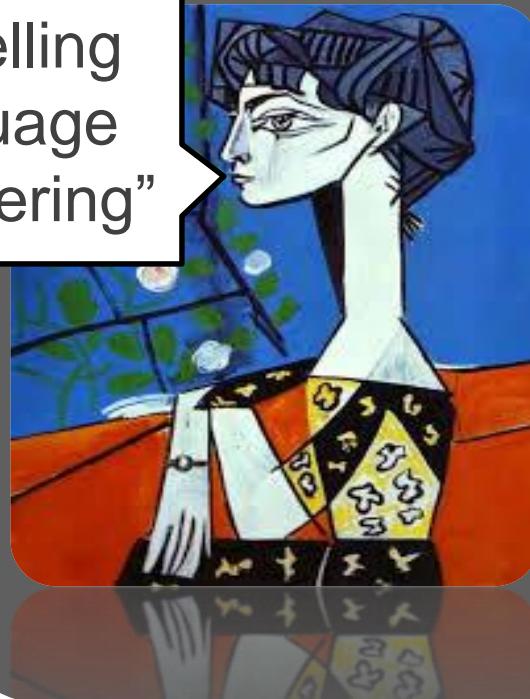


Is code a model?

© Copyright Malina Software

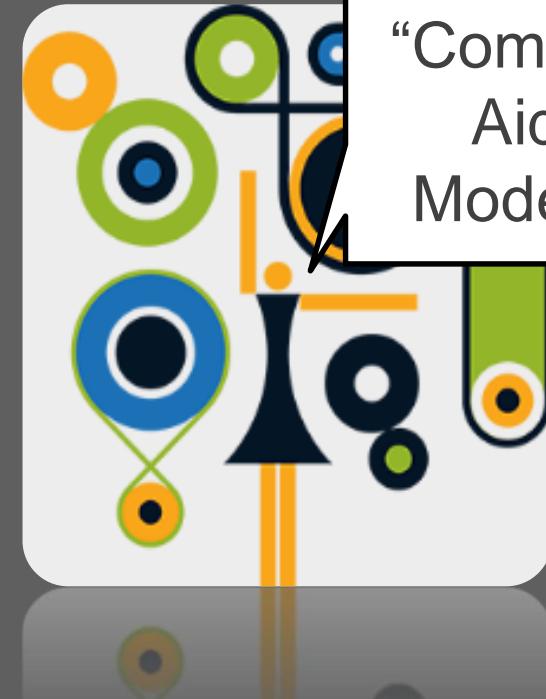
Abstraction

“Modelling
Language
Engineering”

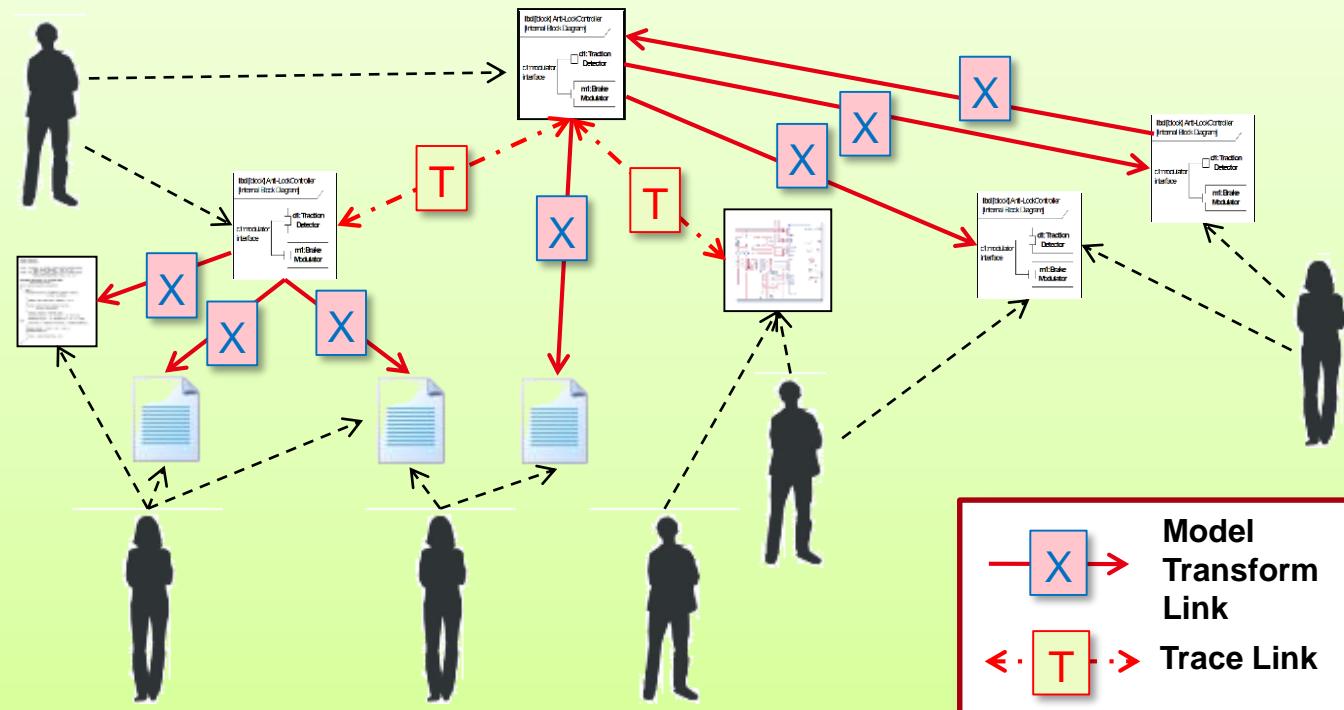


Automation

“Computer-
Aided
Modeling”



Model-based Development Approach

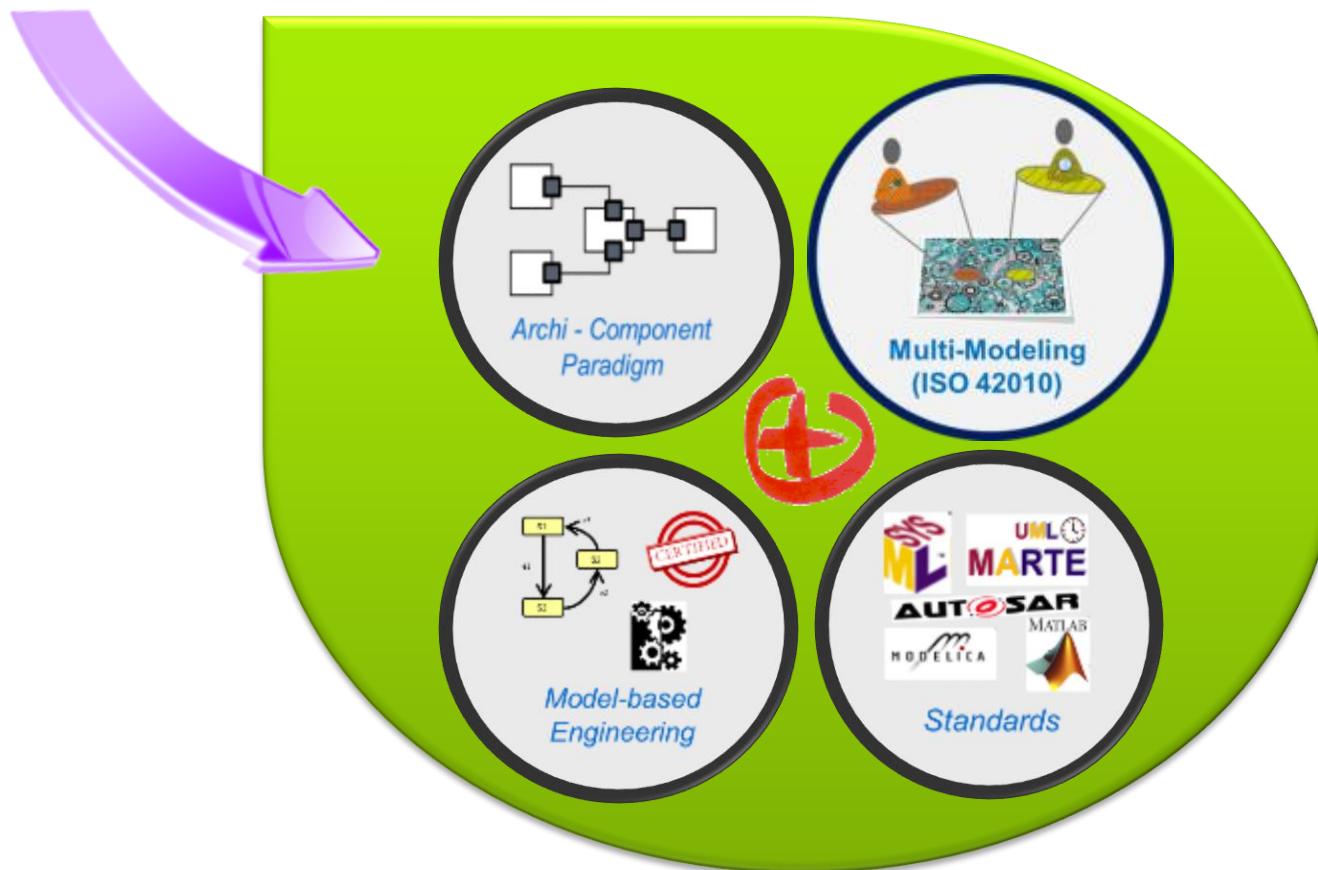


Model Transform Link
X → Model Transform Link
T ← Trace Link →

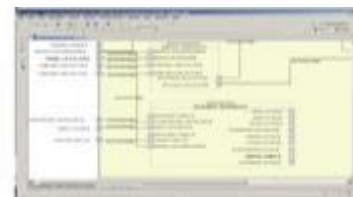
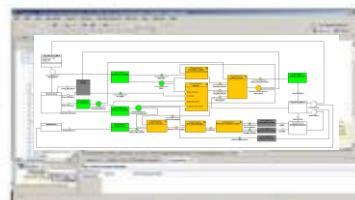
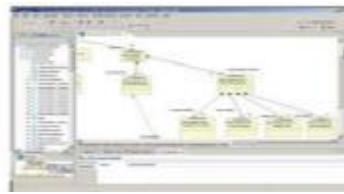


 More efficient, More reliable,
and More scalable.

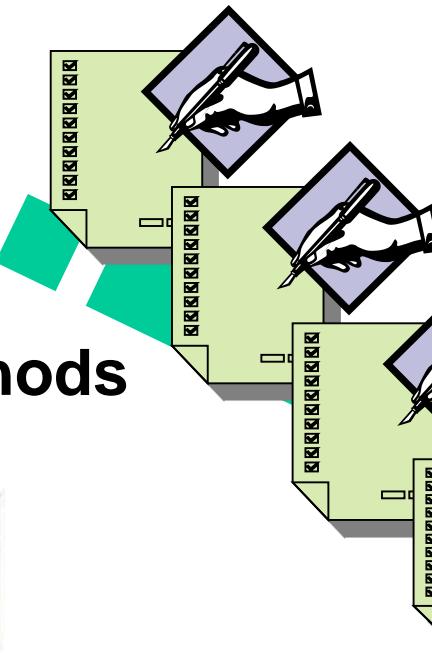
Going further for developing modern complex systems & software requires new advanced and innovative methods and tools



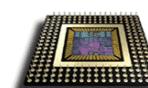
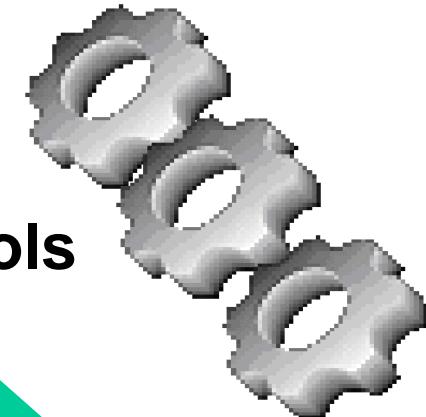
From requirement document: *Hundreds of pages*



... to code: *millions of lines*

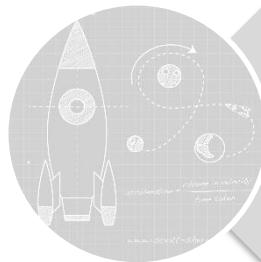


& tools

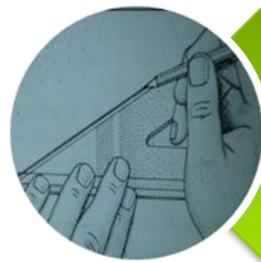




Introduction



Basis is Modeling



Modeling with Which Language?

Building



Worker



Architect

Computer science



Developer



Designer, Architect

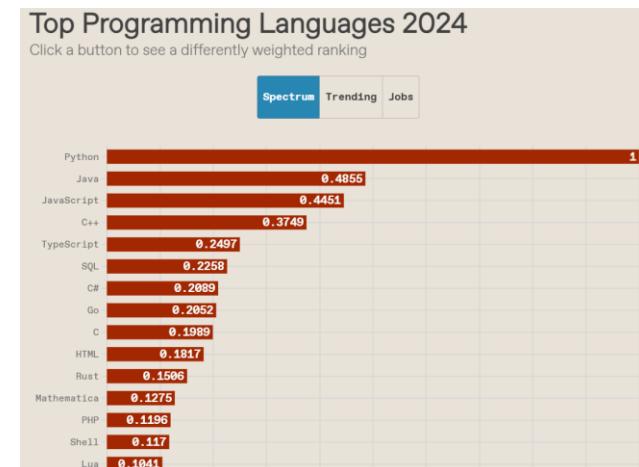
Language Rank Types Spectrum Ranking

1. Python		100.0
2. C		99.7
3. Java		99.5
4. C++		97.1
5. C#		87.7
6. R		87.7
7. JavaScript		85.6
8. PHP		81.2
9. Go		75.1
10. Swift		73.7

Rank	Language	Type	Score
1	Python v		100.0
2	Java v		95.4
3	C v		94.7
4	C++ v		92.4
5	JavaScript v		88.1
6	C# v		82.4
7	R v		81.7
8	Go v		77.7
9	HTML v		75.4
10	Swift v		70.4

<https://spectrum.ieee.org/computing/software/the-2017-top-programming-languages>

<https://spectrum.ieee.org/computing/software/the-2021-top-programming-languages>



Standards have traditionally provided major boosts to technological progress !

But standards enable also vendor independence

- Users have a choice of different vendors (no vendor “tie-in”)
- Forces vendors into competing and improving their products

The Object Management Group (OMG) has created the Model-Driven Architecture initiative

- A comprehensive set of standards in support of MBE including standard modeling languages



UML2, SysML and MARTE

Mature modeling language

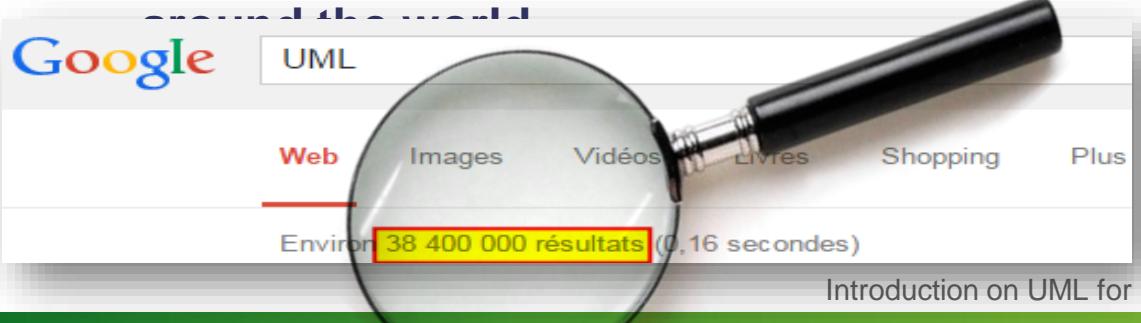
- Initially based on very experienced modeling language designers: the three amigos, Booch, Jacobson and Rumbaugh but also Coleman, Desfray, Embley, Gamma, Harel, Meyer, Odell, Selic, Shaer-Mellor, Wirfs-Brock, etc...
- A more than 20 year old modeling languages (current version:2.5) continually maintained and updated by very advanced experts coming from various origin: end users, tool providers and academics

A rich modeling language covering a large set of concerns

- E.g. architecture, automata, data-flow, scenario, and use-case

Internationally popular and in-use

- UML is widely educated, disseminated, and implemented... all around the world



Originally intended for modeling software-intensive systems

- UML models capture different views of a software system (information model, run-time structure/behavior, packaging, deployment, etc.)
- Inspired primarily by the concepts from object-oriented languages (class, operation, object, etc.)

However, the general nature and large scope of its concepts made UML suitable for extensions to other domains

Domain Specific Modeling by profiling the UML2!



