Complex Systems and Microsystems Design: The Meet-in-the-Middle Approach

J.L. Boizard, N. Nasreddine,
D. Estève, JY. Fourniols
N2IS
Université de Toulouse, LAAS-CNRS
7 avenue du Colonel Roche, 31 077 Toulouse.

Outline

- **An example of complex system/microsystem: wireless sensors networks**
- Methodology and design flow
- Integrated development environment and tools
- Conclusions

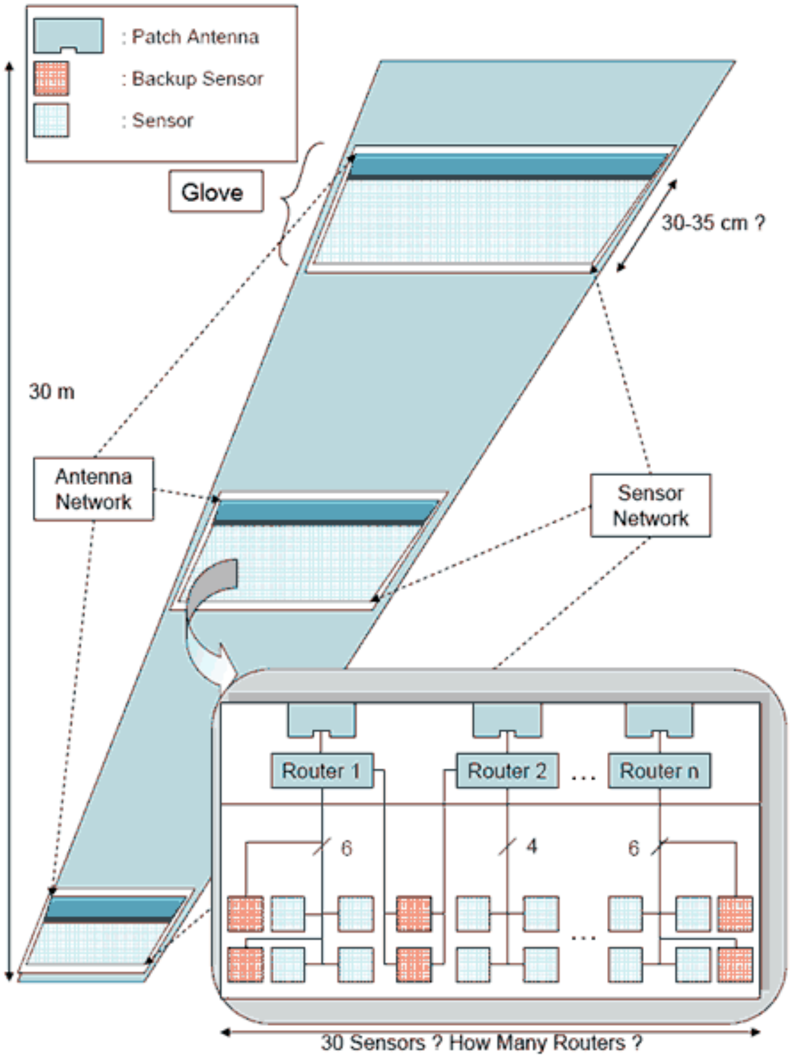
The sensors network

<u>Main goals:</u>

•Watch the behavior of the wings of a plane during its flying stage

•Watch the behavior of the structure of satellites during the experimental setup

The sensors network



The sensors network

<u>Our goals:</u>

•Design, check and validate a virtual prototype before manufacturing

•Elaborate behavioral models of the network in order to evaluate its performances for topological reconfigurations

The sensors network

Worst case sensors network :

•Hundreds of wireless sensors communicating with a supervisor inside the plane

• Strict requirements (useful bandwidth, sampling jitter, large data flow, no loss data, temperature range, …)

•Self powered (day and night), very small packages

• ….

The sensors network

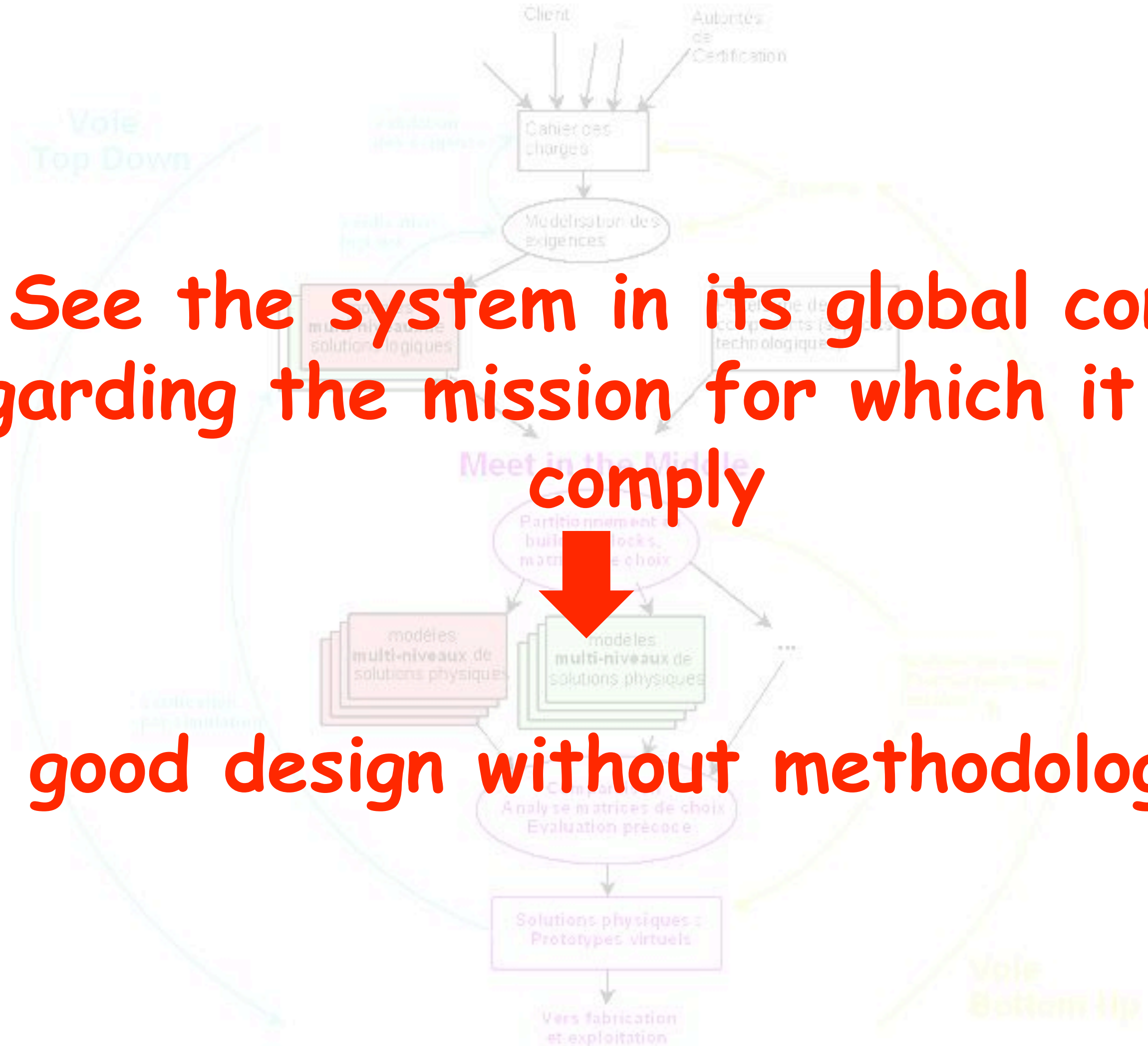
In the end the network represents:

- Many use cases
- Many functionalities

… under severe constraints

At the same time we have to:

- Design quickly, without errors and non expensive solutions
- Provide a well design quality process (re-use, …)
- Give a method to coordinate the engineers envolved in the design



=> See the system in its global context regarding the mission for which it has to comply

No good design without methodology

Outline

- An example of complex system/microsystem: wireless sensors networks
- **Methodology and design flow**
- Integrated development environment and tools
- Conclusions

Methodology and design flow

- Based on the EIA 632, from the Object Management Group.

- A standard to provide an integrated set of fundamental processes to aid a developer in the engineering of a system

The EIA 632 standard

- Logical solution
- Physical solution

Processes for Engineering a System

- Validation
- Checking

Acquisition and Supply *(Subclause 4.1)*
- Supply Process
- Acquisition Process

Technical Management *(Subclause 4.2)*
- Planning Process
- Assessment Process
- Control Process

System Design *(Subclause 4.3)*
- Requirements Definition Process
- Solution Definition Process

Product Realization *(Subclause 4.4)*
- Implementation Process
- Transition to Use Process

Technical Evaluation *(Subclause 4.5)*
- Systems Analysis Process
- Requirements Validation Process
- System Verification Process
- End Products Validation Process

Our approach

Methodology for the design of wireless sensors network (suitable for many embedded systems):

- The Top Down approach: high level analysis and simulation based on logical and/or behavioral models that take into account the system requirements.

- The Bottom Up approach: can be viewed as an exploration of libraries containing models of physical solutions in order to build an architecture (virtual prototype) able to meet all the requirements.

- **The « meet in the middle » approach: can be considered as a successive refinement method going alternatively from the top down to the bottom up in order to converge to a physical solution.**

Our approach

➡ The Top Down approach:

Advantages:

- Identify the main objects of the system by analyse and successive refinements
- High level constraints can be associated to the objects (execution delay, comsumption, packaging, …)
- The models are logical or behavioral ones: the simulation time is greatly reduced, the architectural exploration is easier.

Disadvantages:

- The method may converge to solutions which will be rejected because no physical solutions are available.

Our approach

➾ The Bottom up approach:

<u>Advantages:</u>

- based on the exploration of physical models of solutions which guarantees the practicability of the system.

<u>Disadvantages:</u>

- the increasing of the combinatorial solutions that are linked to the importance of the libraries
- the complexity of the models increases the simulation time (not convenient for architectural exploration)

Our approach

➡ The Meet in the Middle approach:

Well suited for the design of complex and heterogeneous embedded systems

1) Converge to a logical/behavioral solution of the global system (i.e. all the functions are identified and checked by simulation by the way of high level or logical models). At this step physical solutions are available *a priori*.

Our approach

➾ The Meet in the Middle approach:

Well suited for the design of complex and heterogeneous embedded systems

2) Libraries of physical models are explored. Parameters linked to the technological solutions (energy consumption, packaging, execution time, …) are extracted and re-introduced to the high level simulation.

Our approach

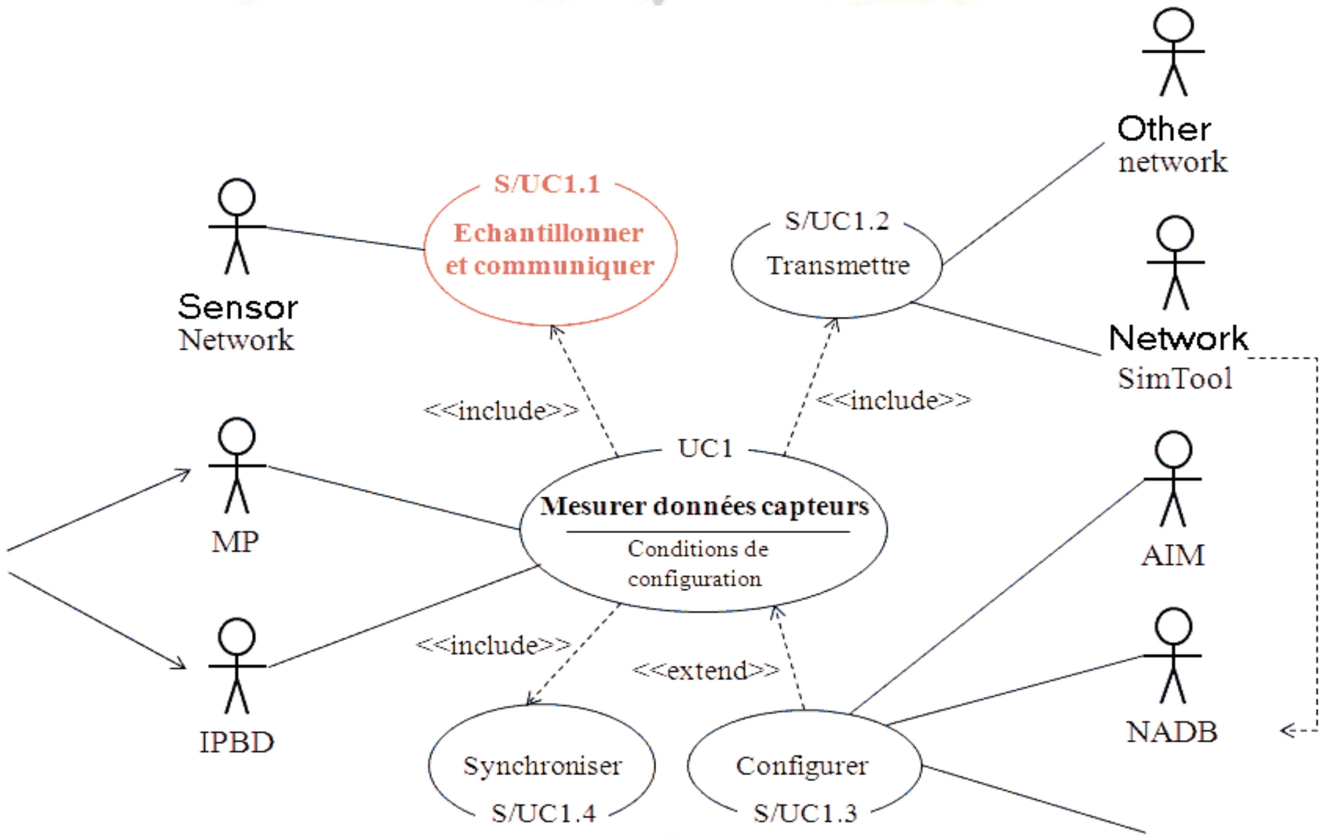
➾ The Meet in the Middle approach:

Well suited for the design of complex and heterogeneous embedded systems

3) This process is iterated until all the physical parameters meet the constraints requirements.

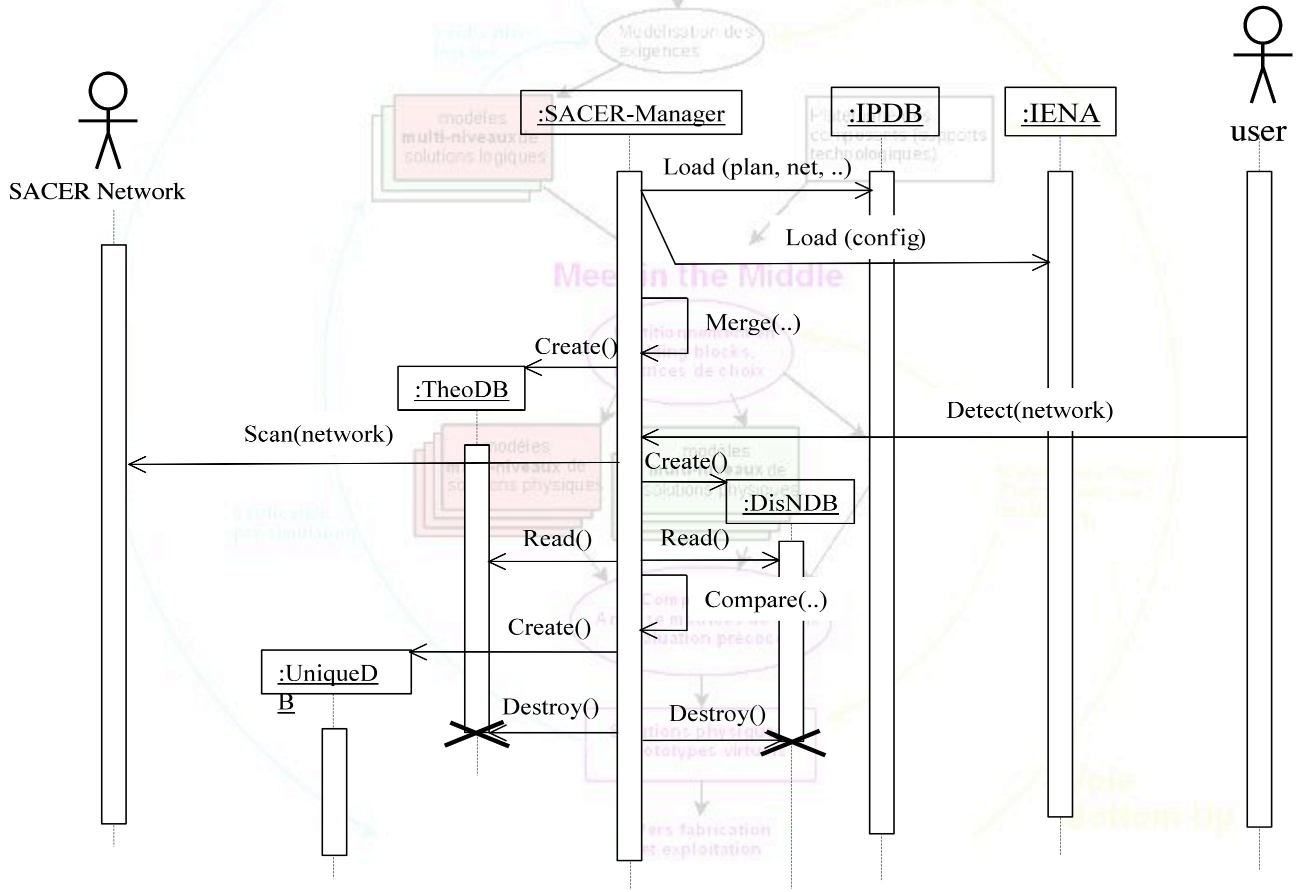
The Top Down approach

An example of the Top Down approach: the use case diagrams (first level) with objects involved in the system.



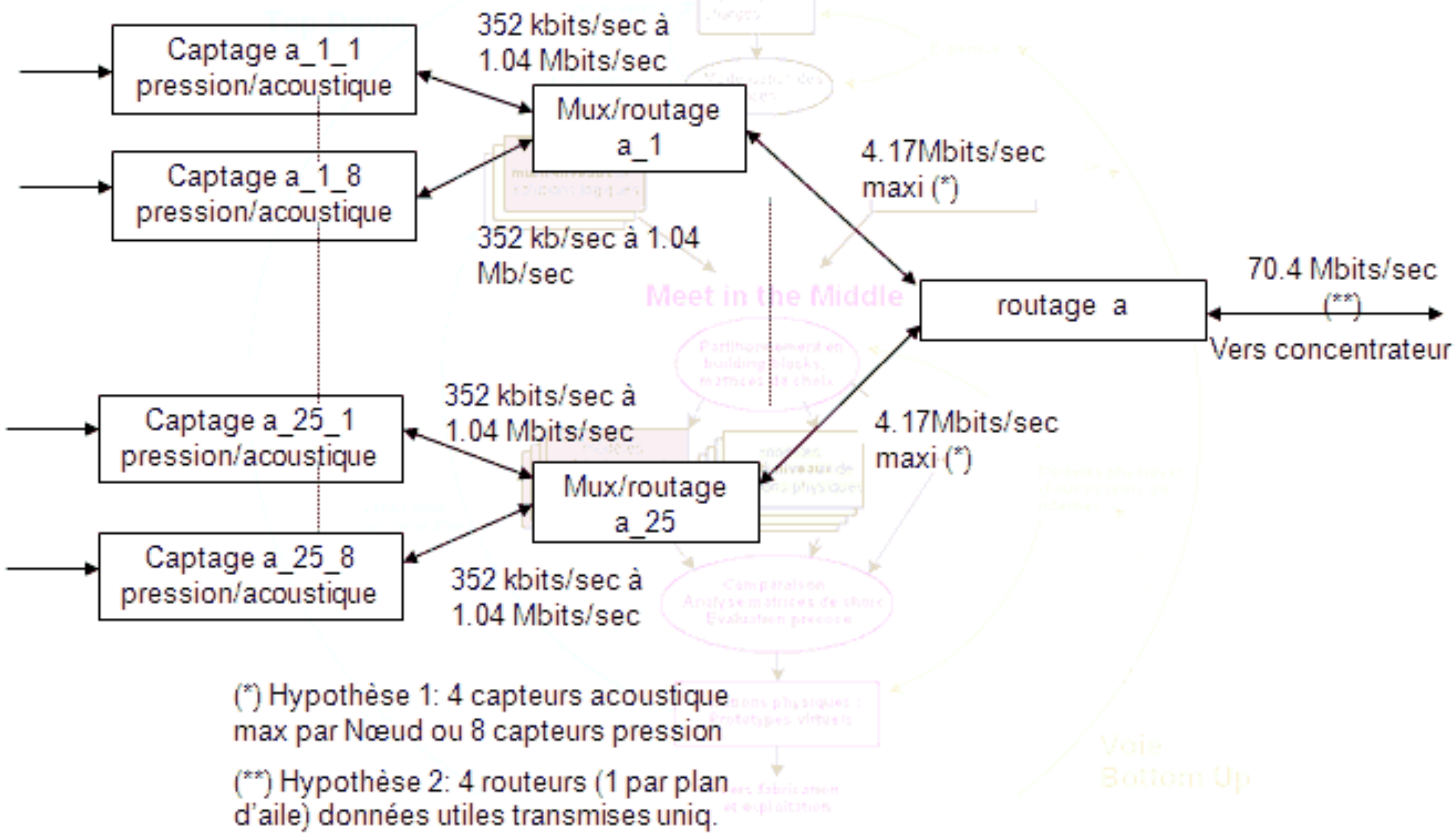
The Top Down approach

An example of the Top Down approach: the sequence diagrams (scheduling of the tasks)



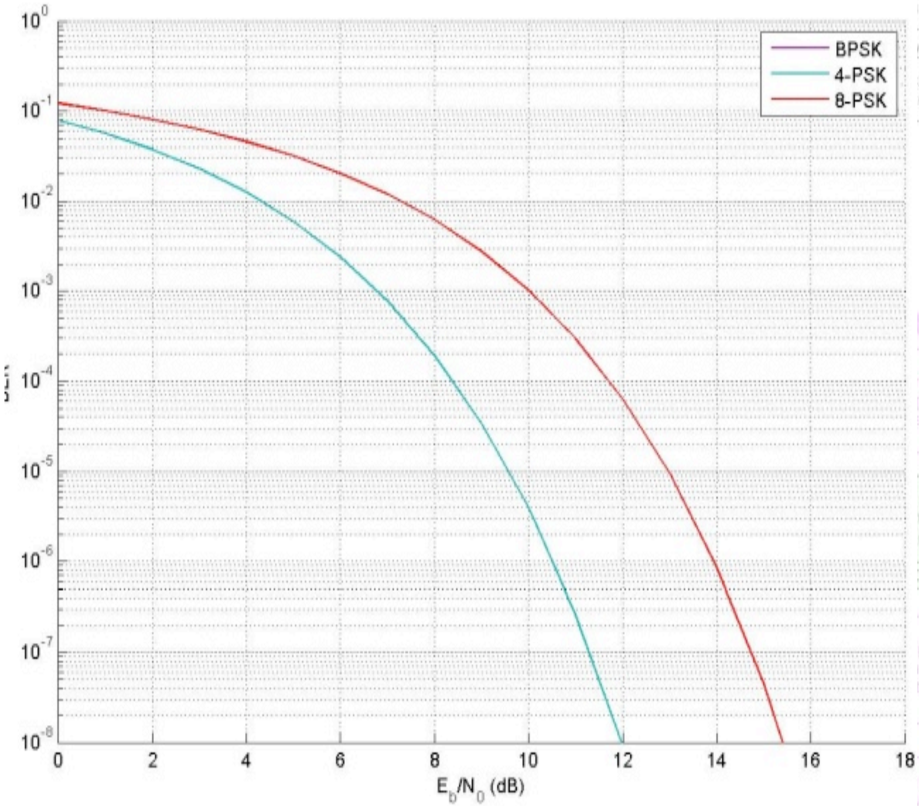
Architectural exploration

Start with an initial architecture that meets the **nominal functionnal requirements** and proceed to its refinement.



Architectural exploration

Starting from physical models, find behavioral ones for high level simulation



Ex : RF transmission channel

Table 1, BER (Eb/No)

$BER = erfc\left(\sqrt{s(x)} \cdot \sin\left(\frac{\pi}{8}\right)\right)$	(1)	8-PSK
$BER = erfc\left(\sqrt{s(x)} \cdot \sin\left(\frac{\pi}{4}\right)\right)$	(2)	4-PSK
$BER = \frac{1}{2} \cdot erfc\left(\sqrt{s(x)}\right)$	(3)	BPSK

BER = Bit Error Rate
E_b = Energy per bit
N_0= Noise power spectral density

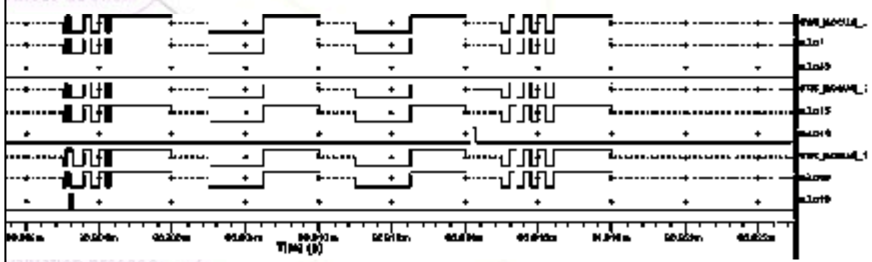
Architectural exploration

Starting from physical models, find behavioral ones for high level simulation

```
\$1I29\ : entity WORK.MUX2_1(IDEAL)
   port map ( SEL => ERR_CODE,
              D0 => \$1N31\,
              D1 => \$1N6\,
              OUTPUT => \$1N48\ );

INVERTER2 : entity EDULIB.INVERTER
   port map ( INPUT => \$1N31\,
              OUTPUT => \$1N6\ );

INVERTER0 : entity EDULIB.INVERTER
   port map ( INPUT => \$1N37\,
              OUTPUT => \$1N5\ );

\$1I41\ : entity WORK.STATE_RAND(IDEAL)
   generic map ( BER => BER )
   port map ( OUT_RAND => \$1N40\ );

\$1I42\ : entity WORK.DEGROUPE_10(IDEAL)
   port map ( D0 => \$1N37\,
              D1 => \$1N36\,
              D2 => \$1N38\,
              D3 => \$1N4\,
              D4 => \$1N35\,
              D5 => \$1N34\,
              D6 => \$1N33\,
              D7 => \$1N32\,
              D8 => \$1N55\,
              D9 => \$1N31\,
              BUS_OUT => OUT_PROP(9 downto 0) );

\$1I43\ : entity WORK.MUX2_1(IDEAL)
   port map ( SEL => ERR_DATE,
              D0 => \$1N37\,
```

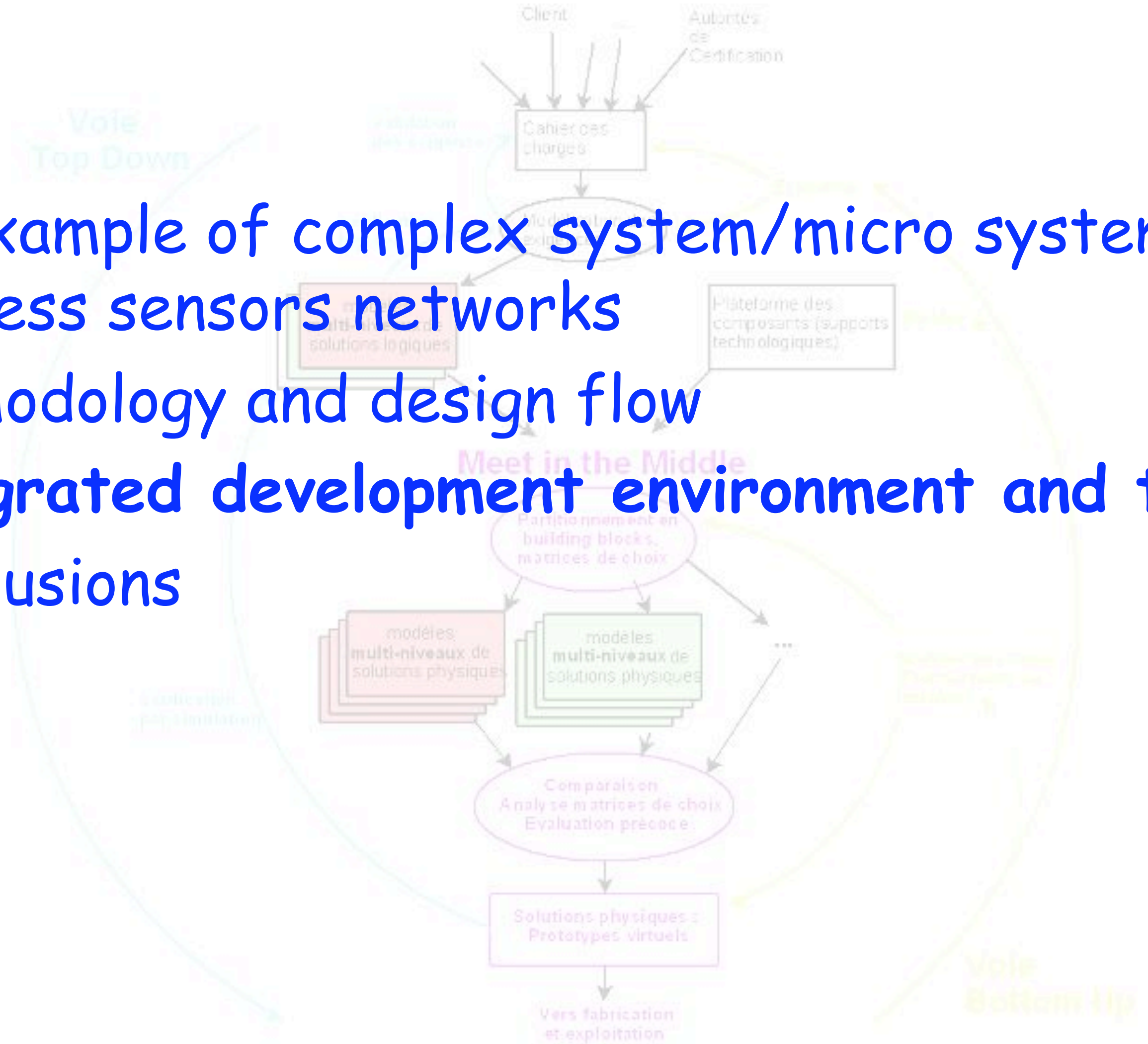
Ex : RF transmission channel with a BER of 10^{-3}

BER = Bit Error Rate



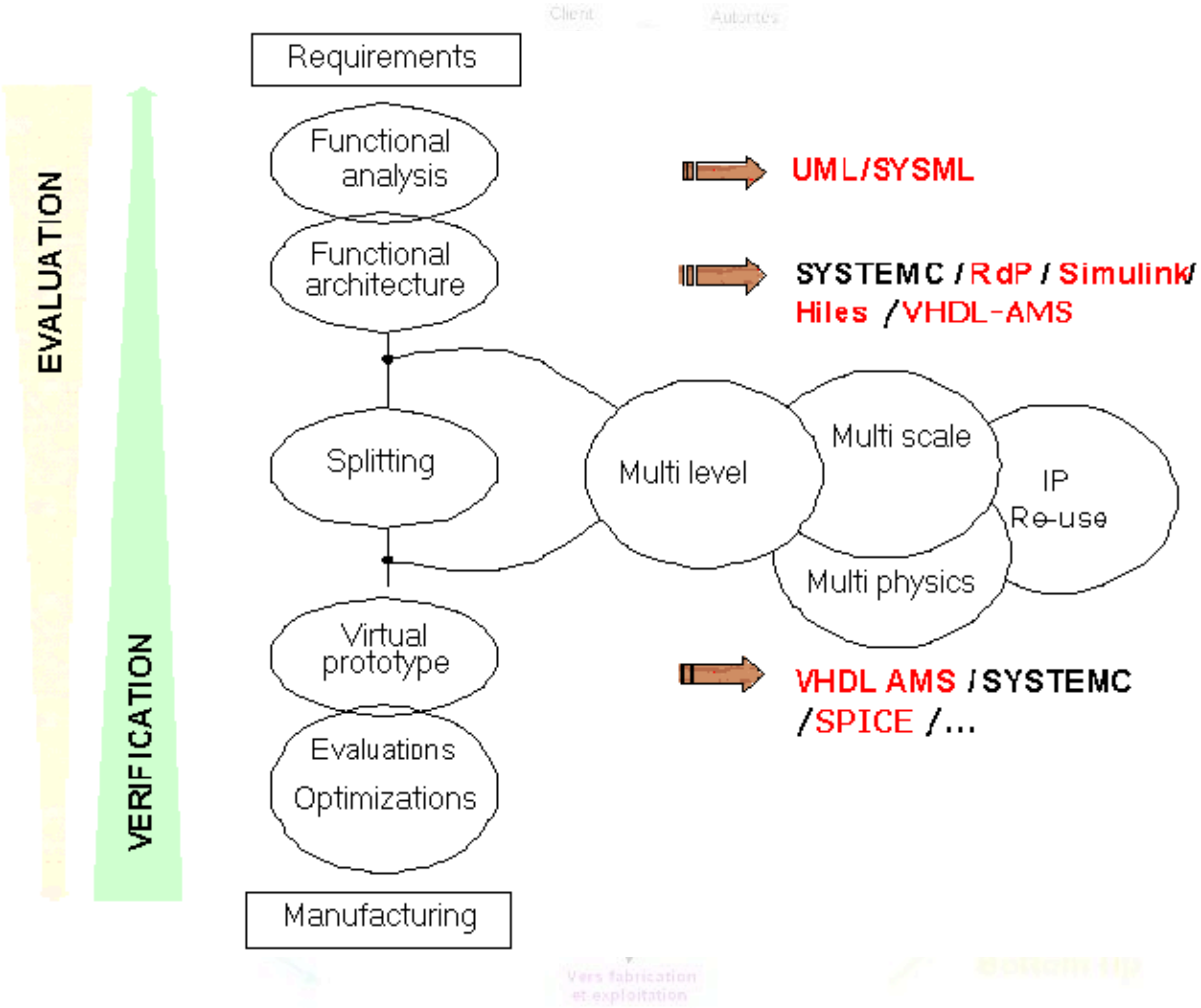
Introducing errors in the data flow

Outline



- An example of complex system/micro system: wireless sensors networks
- Methodology and design flow
- **Integrated development environment and tools**
- Conclusions

Environment and tools



Conclusion

The meet in the middle approach:

=> well suited to design complex and heterogeneous embedded systems.

=> based on a refinement process that starts with the top down approach and allows:

* To define a high level logical simulable architecture that meets all the requirements

Conclusion

* To facilitate the architectural exploration by using high level models which ones decrease the simulation time

* To propagate constraints in order to help the designers to explore the libraries of physical solutions more quickly.

* To focus on the high level functions for which physical solutions are available.