



A Design Methodology for the Development of a Complex SoC using UML and Executable System Models

Yves Vanderperren
yves.vanderperren@st.com

6th European SystemC User Group Meeting
October 22nd, 2002

Outline

- ❑ Stating the case: the OFDM Wireless LAN Project
- ❑ Methodology overview
 - Process
 - Requirements Capturing and Use-Case analysis
 - System Modelling
 - Matlab + SystemC + UC
 - A step further: UML modelling of architecture
- ❑ Conclusions and Summary



Outline

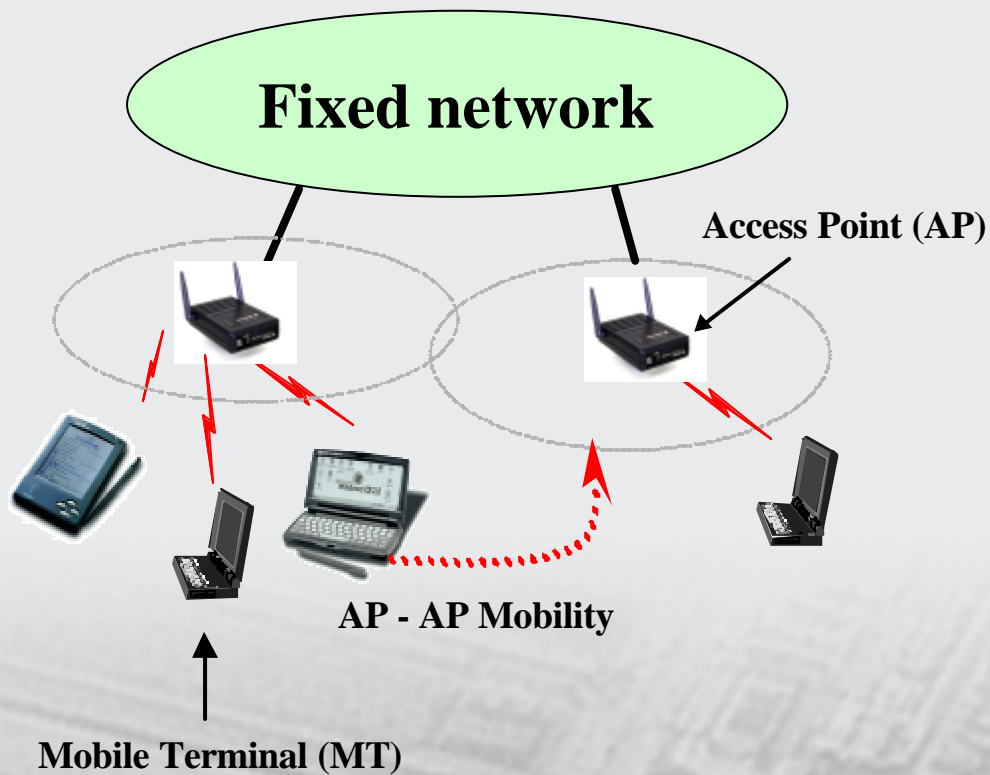
- ❑ Stating the case: the OFDM Wireless LAN Project
- ❑ Methodology overview
 - Process
 - Requirements Capturing and Use-Case analysis
 - System Modelling
 - Matlab + SystemC + UC
 - A step further: UML modelling of architecture
- ❑ Conclusions and Summary



Stating the case

Infrastructure based network:

Business Environment
Centralized Mode

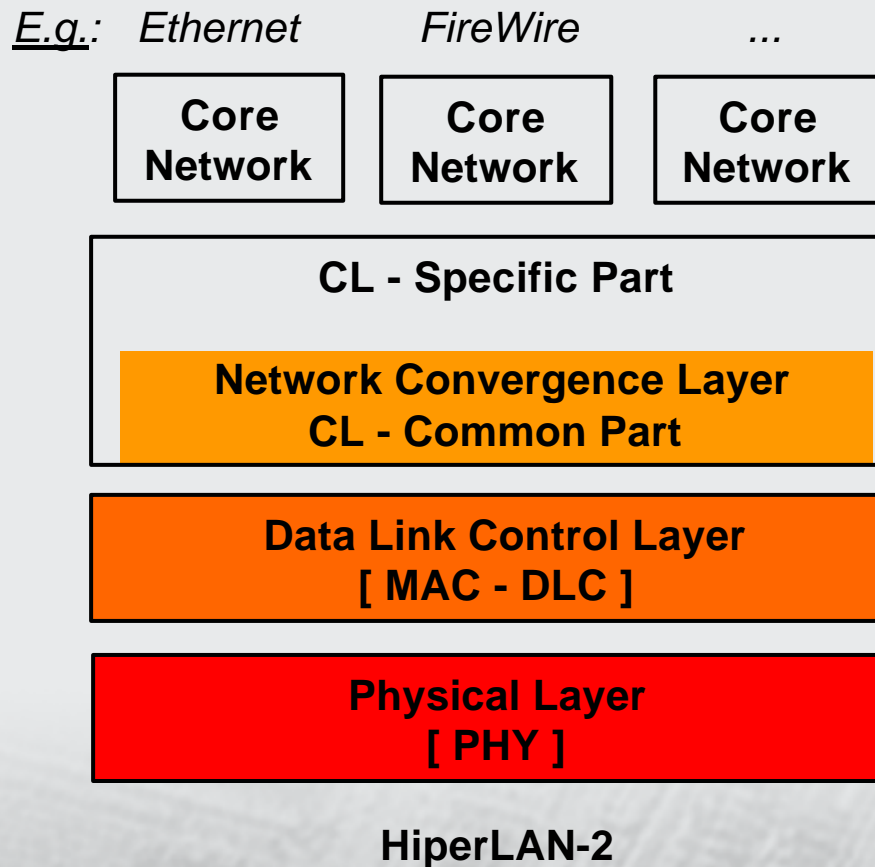


Ad-hoc network:

Home Environment
Direct Mode



Stating the case

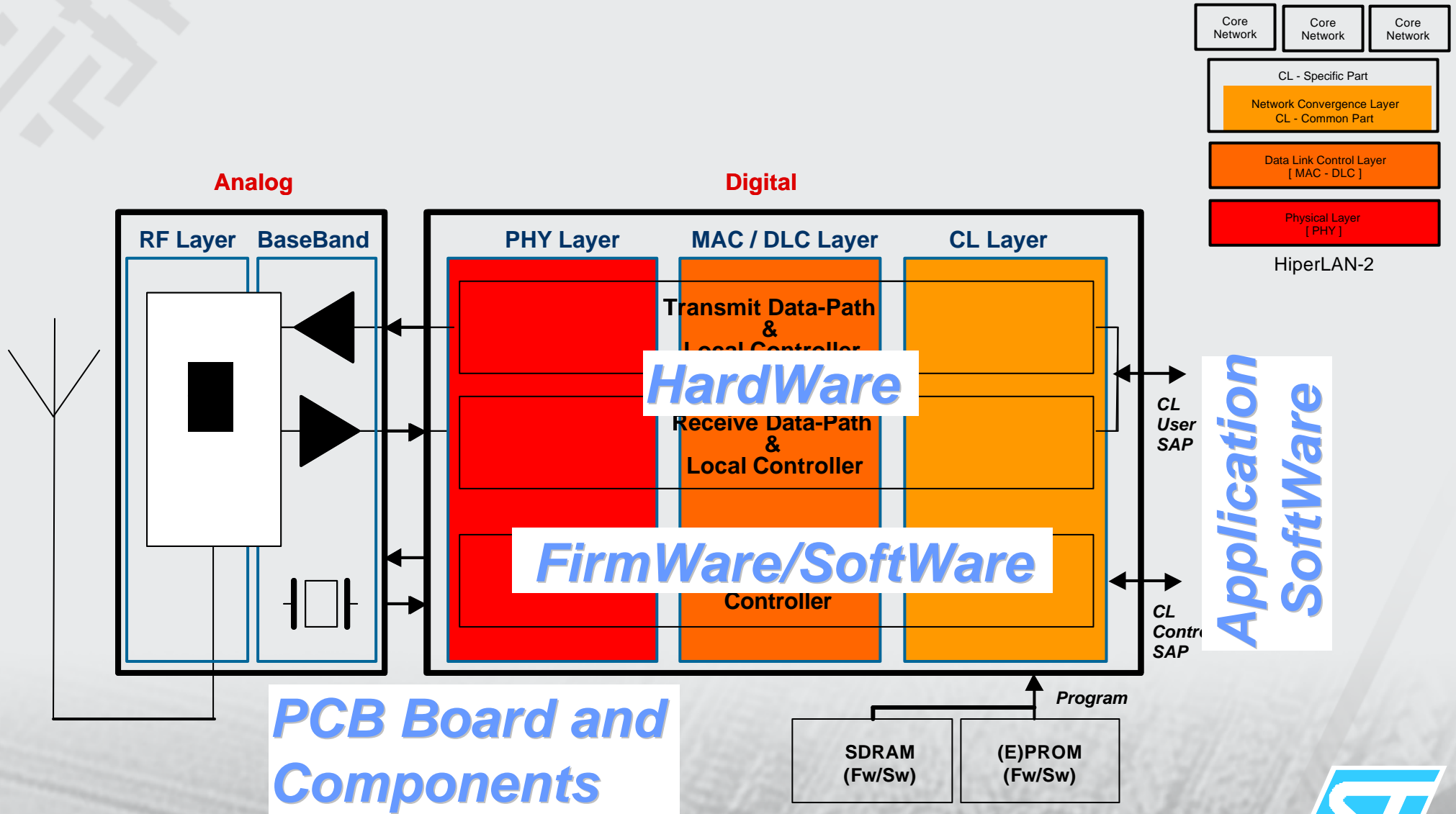


- Different flavours
 - Business-environment
 - Home-environment

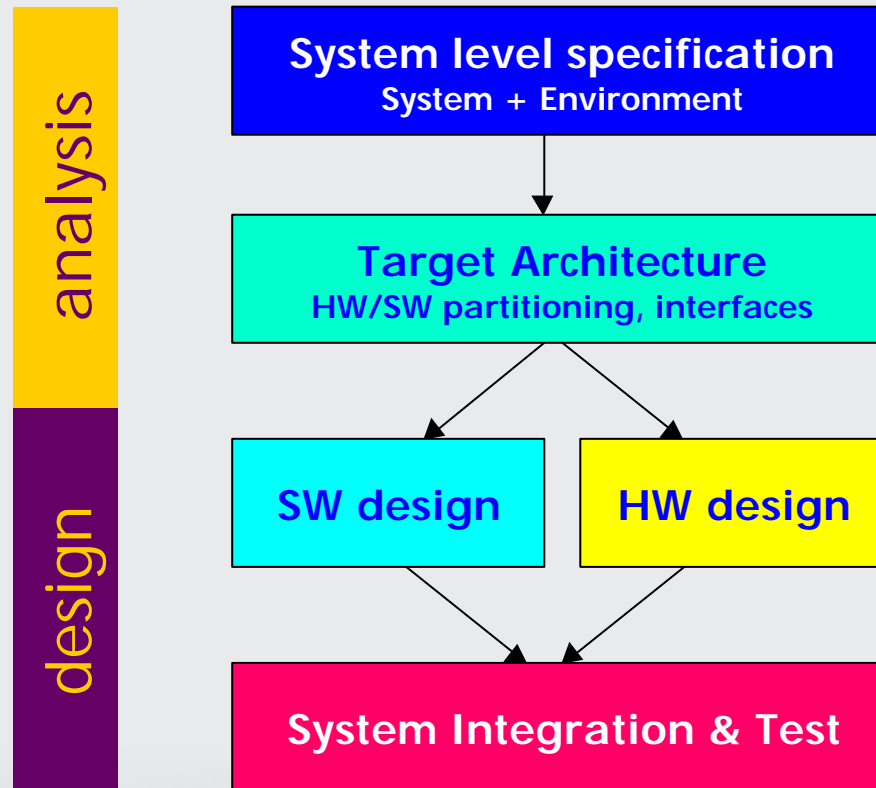
- ➔ Hardware
- ➔ Firmware
- ➔ Software



Stating the case



Stating the case



Easy said...



Outline

- ❑ Stating the case: the OFDM Wireless LAN Project
- ❑ Methodology overview
 - Process
 - Requirements Capturing and Use-Case analysis
 - System Modelling
 - Matlab + SystemC + UC
 - A step further: UML modelling of architecture
- ❑ Conclusions and Summary



Central RUP principles

❑ Iterative Development

- Refine system implementation to address risks early

❑ Use Case Driven Architecture

- Validate architectural design using Use Cases

❑ Visual Modelling (using UML)

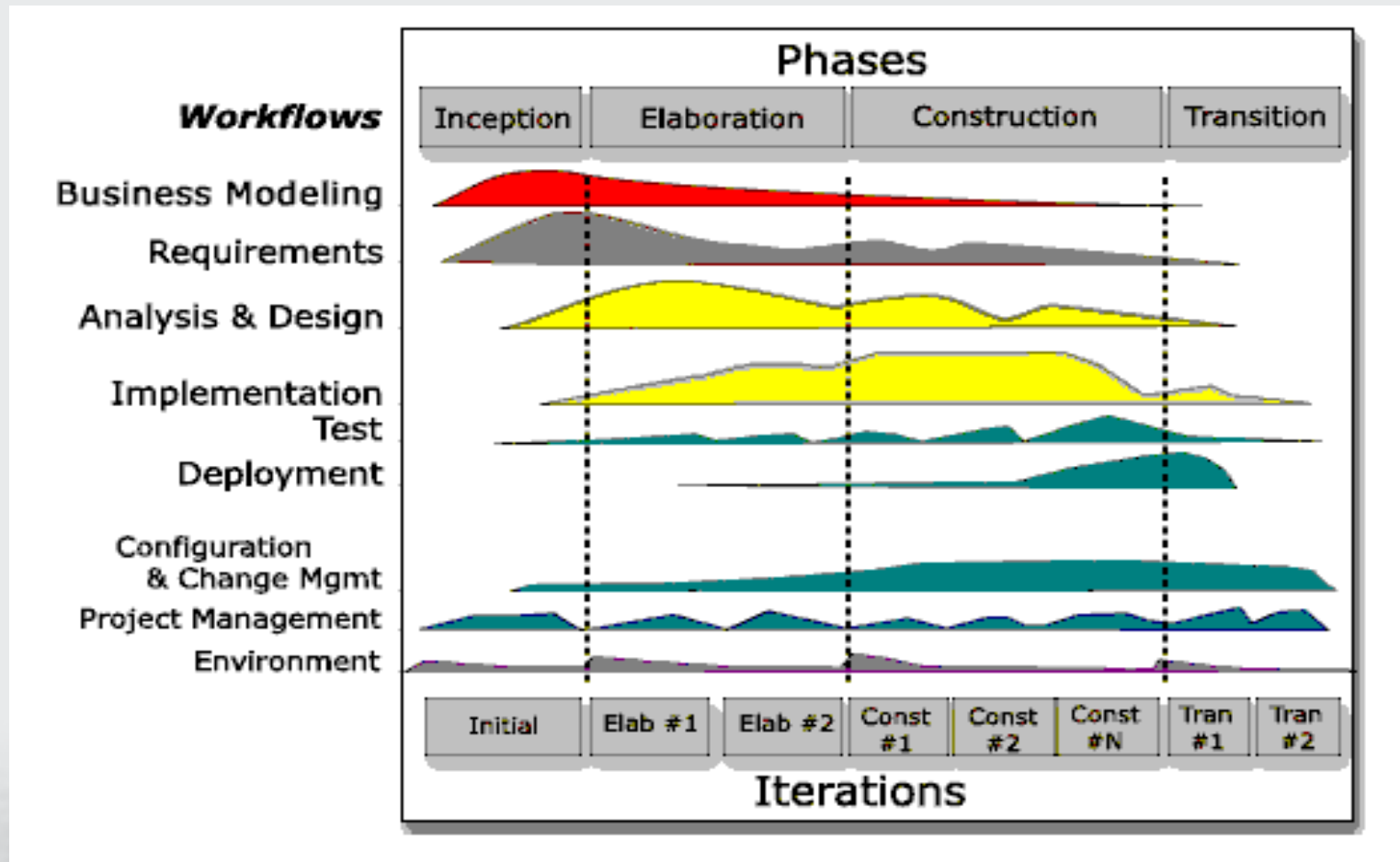
- Use well-defined visual language to communicate requirements and design

❑ Testing

- Evolves during each iteration in parallel with system
- Allows efficient regression testing of each iteration



Iterative development



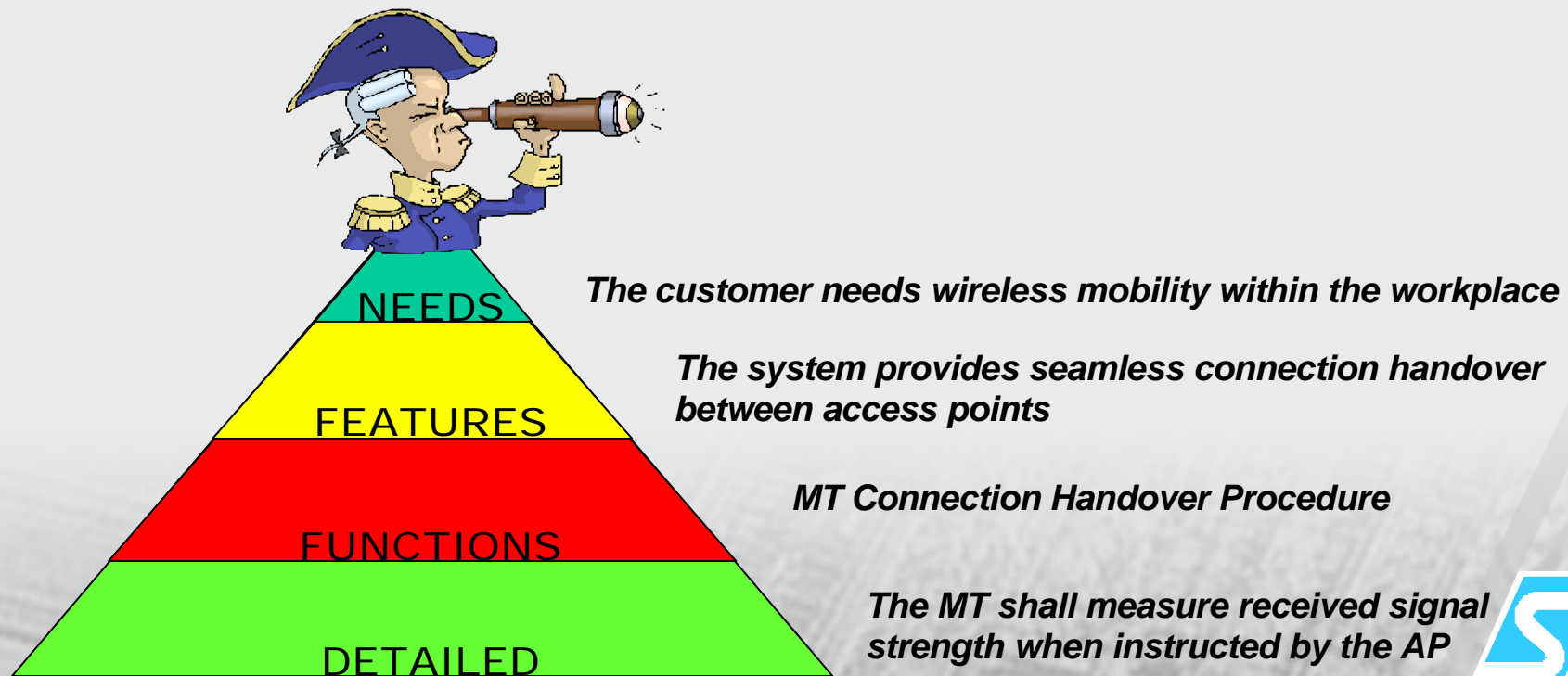
Outline

- ❑ Stating the case: the OFDM Wireless LAN Project
- ❑ Methodology overview
 - Process
 - Requirements Capturing and Use-Case analysis
 - System Modelling
 - Matlab + SystemC + UC
 - A step further: UML modelling of architecture
- ❑ Conclusions and Summary

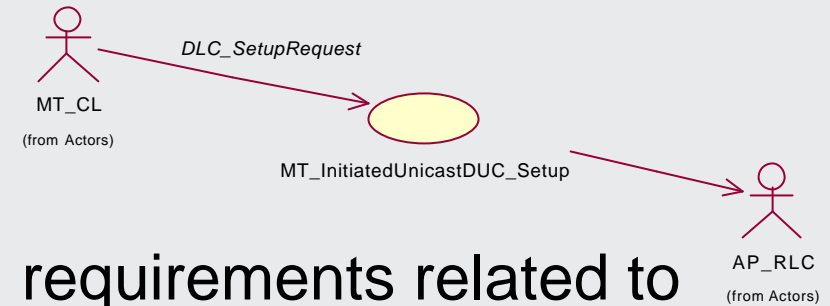


Vision document

- ❑ Focus on the Customers & Stakeholders
- ❑ Identify Success Criteria
- ❑ Top of Requirements Pyramid



Use Case Analysis



- ❑ A mechanism to specify functional requirements related to top-level view
 - UC = a service providing value to an external entity

- ❑ UC's are basically textual – a behavioural sequence
 - UML diagrams help to organise and document

- ❑ For each UC we document a.o.:
 - Primary (expected) system responses
 - Secondary system responses (e.g. error conditions)
 - Links to other UC's



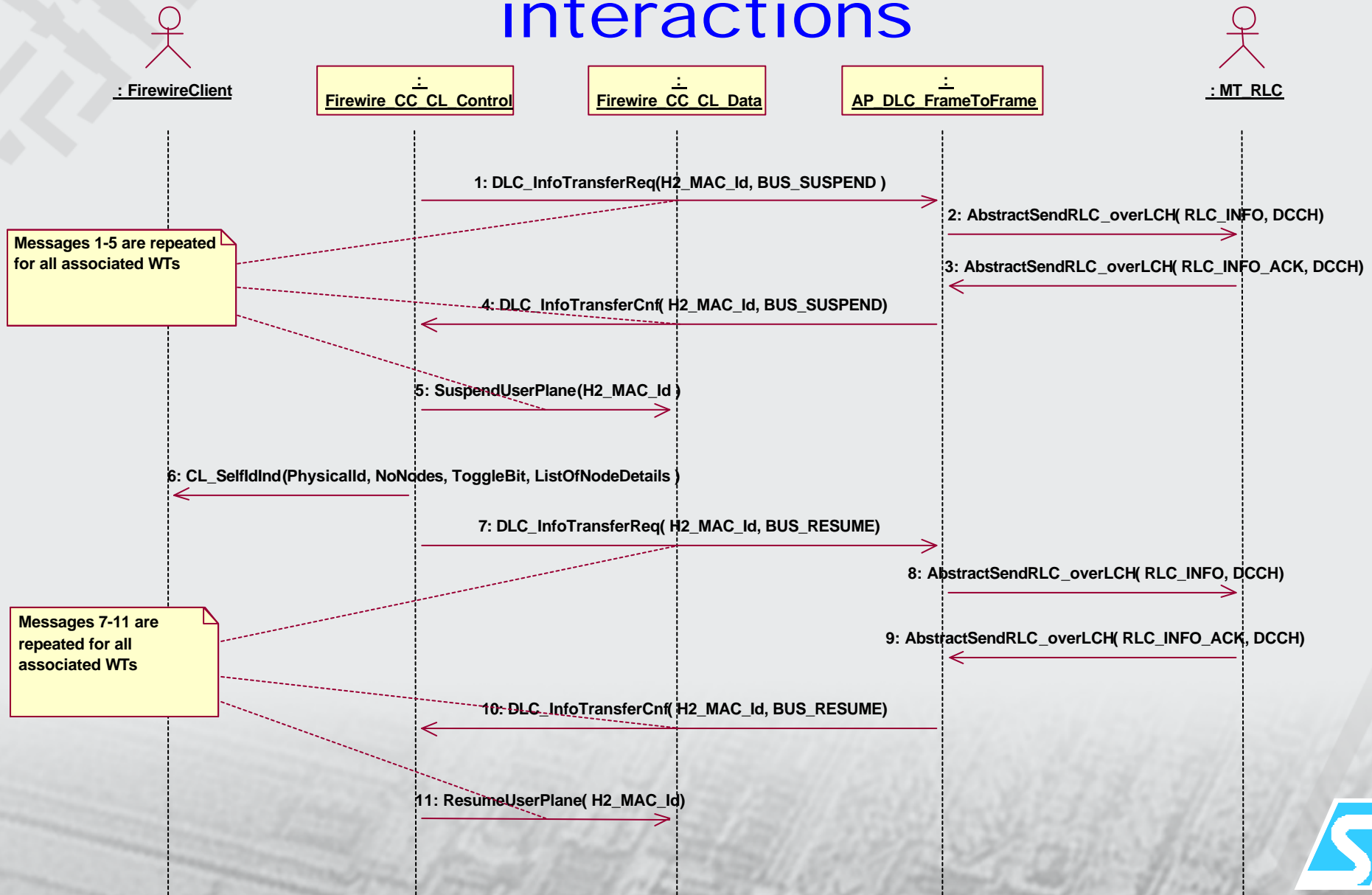
✓ scope of the system, user goals



✓ failure handling



Sequence diagrams of external interactions

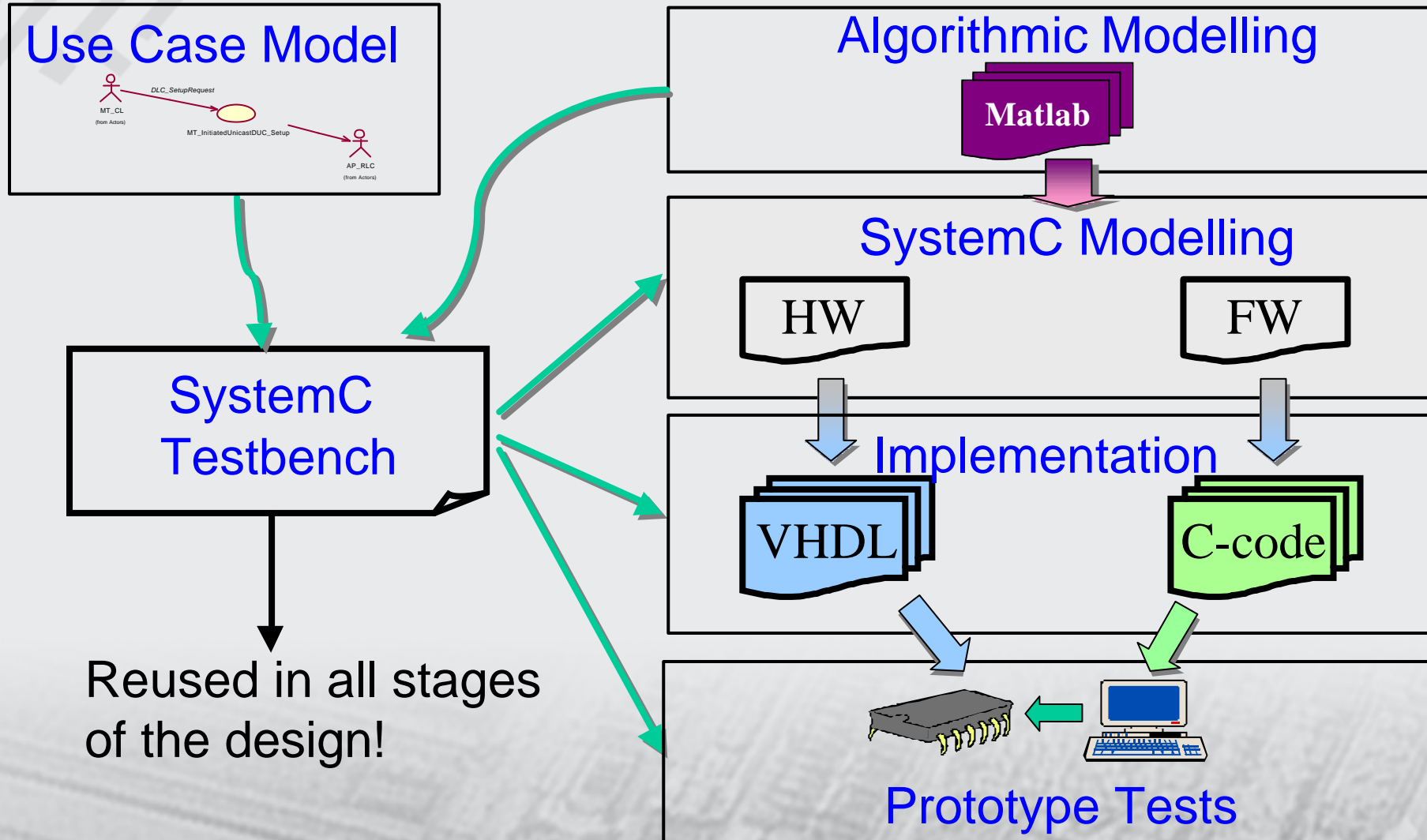


Outline

- ❑ Stating the case: the OFDM Wireless LAN Project
- ❑ Methodology overview
 - Process
 - Requirements Capturing and Use-Case analysis
 - System Modelling
 - Matlab + SystemC + UC
 - A step further: UML modelling of architecture
- ❑ Conclusions and Summary



Vertical Testbench Reusage



Objectives of the SystemC model

- ❑ **An Executable Specification** for communication with designers
 - Designers participate in writing the specification – organic knowledge transfer

- ❑ **A Tool** to
 - verify the overall system behavior and architecture
 - run test scenarios covering a global model of HW & FW
 - detail the architecture for critical blocks
 - do finite-precision design

- ❑ **A Reference**
 - test benches are the starting point for all other test benches and system test plans (Lab qualification)



Outline

- ❑ Stating the case: the OFDM Wireless LAN Project
- ❑ Methodology overview
 - Process
 - Requirements Capturing and Use-Case analysis
 - System Modelling
 - Matlab + SystemC + UC
 - A step further: UML modelling of architecture
- ❑ Conclusions and Summary



UML Modelling of Architecture

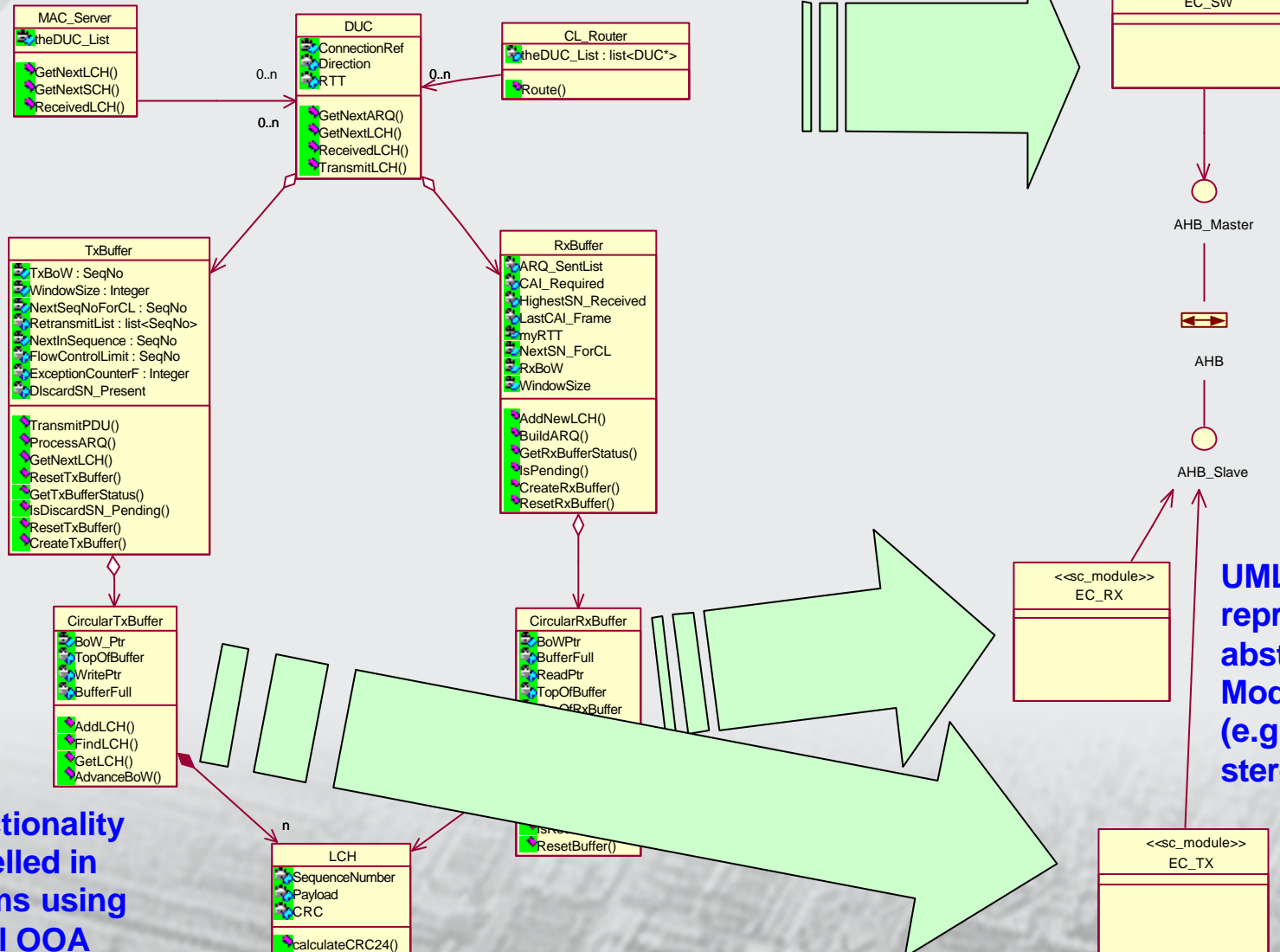
- ❑ Applied to analysis of requirements and specification of higher layer SW, using RRT

- ❑ Motivation
 - Capture essential information about system architecture and interfaces in a single, common format

- ❑ Why UML?
 - UML is the notation of choice for object-oriented analysis and design
 - A SystemC model is an object-based representation of the system elements written in an OO language
 - Tools are available for reasonable costs
 - UML Modelling Environment
 - Document Generation



From Function to Architecture



System functionality can be modelled in abstract terms using conventional OOA approach

UML can be tailored to represent SystemC abstractions such as Module, Port, Channel (e.g. using stereotypes)



But...

- ❑ UML tools don't support function to architecture mapping
 - Re-mapping functions results in manual changes to interfaces, collaborations etc.

- ❑ UML tools don't support SystemC directly
 - Manual synchronization between SystemC and UML
 - Standard profile for SystemC?

- ❑ UML standardized mainly for SW systems development
 - Different profiles for real-time embedded systems are available



Outline

- ❑ Stating the case: the OFDM Wireless LAN Project
- ❑ Methodology overview
 - Process
 - Requirements Capturing and Use-Case analysis
 - System Modelling
 - **Matlab + SystemC + UC**
 - **A step further: UML modelling of architecture**
- ❑ **Conclusions and Summary**



Conclusions

- ❑ Use Case Driven approach drives design
- ❑ Unified view, interdisciplinary communication, cross-functional teamworking
- ❑ UML-based approaches complement SystemC v2.0 well
- ❑ Methodology required to complement language capabilities



- ❑ Need to address SystemC - UML synchronisation
- ❑ Learning curve for UML and modelling concepts
- ❑ How does SoC domain fit with general model of UML for Systems Engineering

