

# Altreonic NV

[www.altreonic.com](http://www.altreonic.com)

## UNIVERSAL SCALABILITY AND HIGH RELIABILITY WITH ALTREONIC



28/02/2010

Altreonic confidential

1

## Altreonic history

- History goes back to Eonic Systems NV
  - *Background in CSP and transputers*
  - *Developed parallel DSP Virtuoso RTOS*
  - *Acquired by Wind River Systems in 2001*
- Open License Society (R&D) 2004
  - *Developing a formalized systems engineering methodology*  
*Unified semantics + interacting entities*
  - *Formally developed network-centric OpenComRTOS.*
- Altreonic is a spin-off of OLS

28/02/2010

Altreonic confidential

2

## Products:



### OpenCookbook:

web portal for full project support, customer tailored.



### OpenVE:

visual modeling/development of embedded software



### OpenComRTOS:

formally developed and verified network-centric RTOS



### OpenTracer:

visual profiling of applications



### OpenHardWare

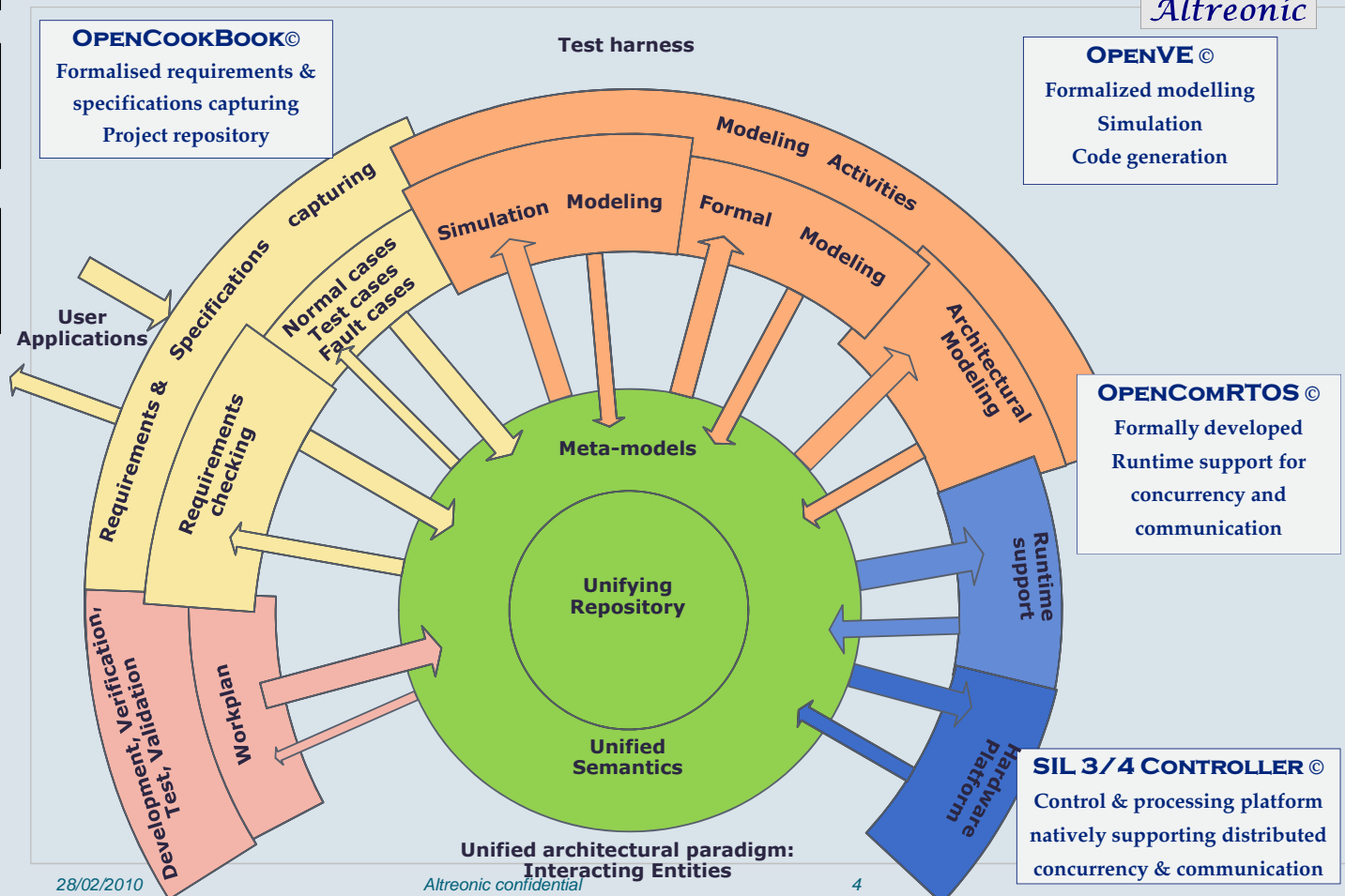
- SIL3/4 controllers under development

28/02/2010

Altreonic confidential

3

## Unified Systems/Software engineering



28/02/2010

Altreonic confidential

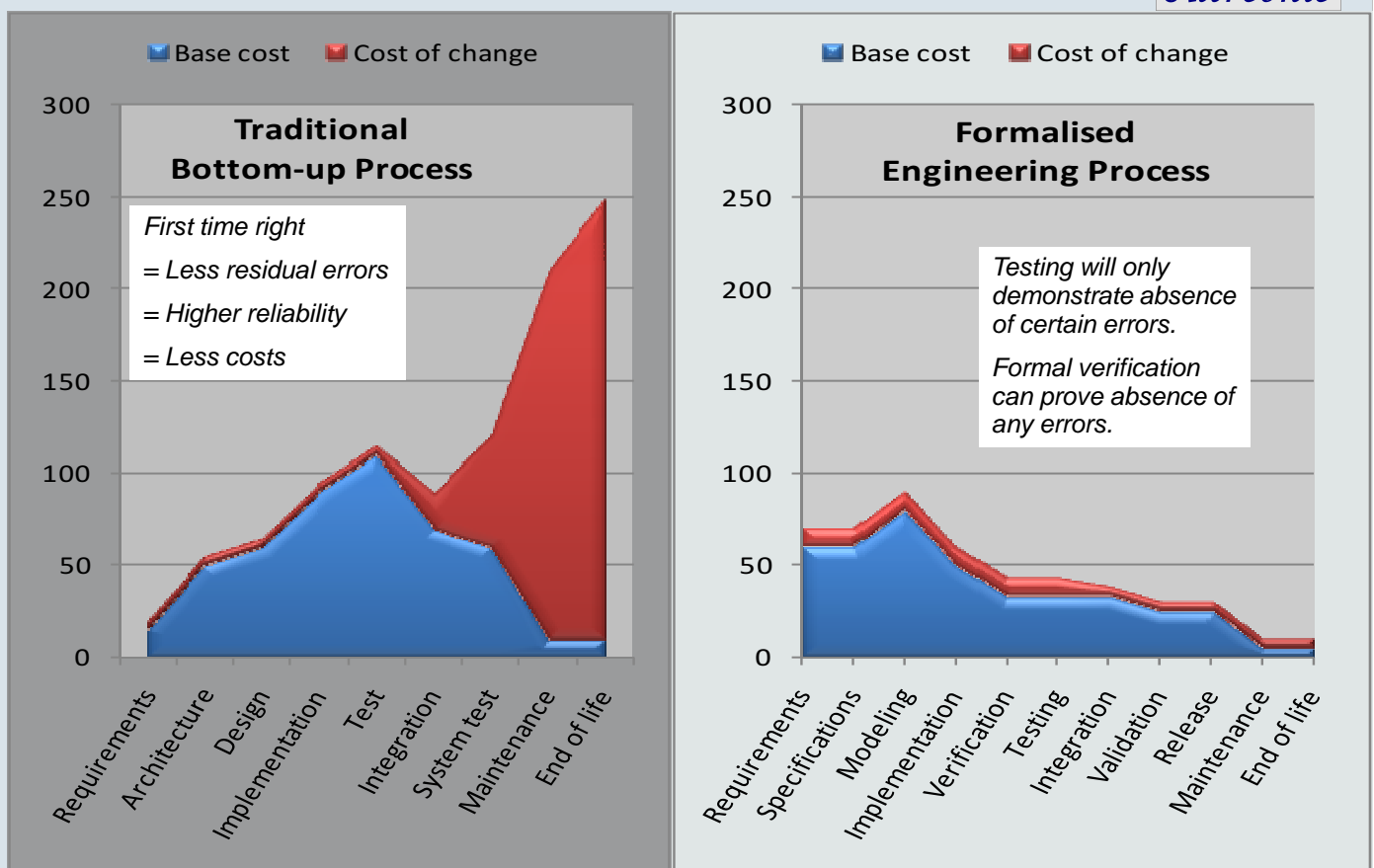
4

- Growing need for dependability of embedded systems

➡ **Trustworthy = Added Value.**

- High reliability can be cost-efficient
- Integrated approach from Altreonic
- Customer base are system integrators
- Application markets:
  - Automotive (e-car).
  - Distributed control (machines, house robots)
  - Next gen. mobility platforms (e.g. 4G netbook phone)

## Why Formalized?



# Unique technology



- Formalized, straightforward approach
- Full integration of tools
- **OpenComRTOS** unique features :
  - Network-centric RTOS
  - Formally developed and verified
  - Scalable yet very small and complete
    - 5 to 10 Kbytes/node
  - Real-time communication support
  - Heterogeneous target support
- Affordable cost

*OpenComRTOS* was one of the three final nominees for the:



28/02/2010

Altreonic confidential

7

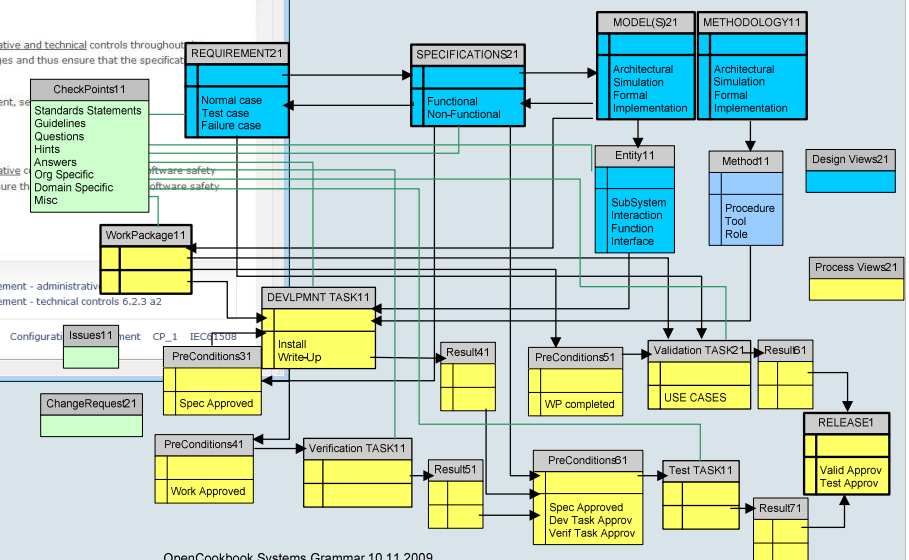
## Step1: Requirements & Specifications



The screenshot shows a web browser displaying the 'ASIL OpenCookbook' interface. The main content area shows details for 'CP\_1 Software Configuration Management - administrative & technical controls 6.2.3 a'. It includes a search bar, a sidebar with navigation links (Forums, ASIL OpenCookbook, Checkpoints, Requirements, Specification, Methodology, Models, Work Package, Release, Free Tagging, Manage Project, My account, Project release API compatibility, Projects, Project types, Recent posts, Standard, Create content, Administer, Log out), and a main text area with 'Status: In Work', 'Type: Standard related', and 'Description Text: Software configuration management should apply administrative and technical controls throughout the software safety lifecycle...'. Below the text are sections for 'CheckPoints1', 'WorkPackage11', 'Issues11', 'PreConditions31', 'Spec Approved', 'PreConditions41', 'Work Approved', 'Verification TASK11', 'Result51', 'PreConditions51', 'WP completed', 'Validation TASK21', 'USE CASES', 'Result61', 'Test TASK11', 'Spec Approved Dev Task Approv', 'Valid Task Approv', 'Result71', and 'RELEASE1'.

**OpenCookbook**

*Structured team work  
over the internet*

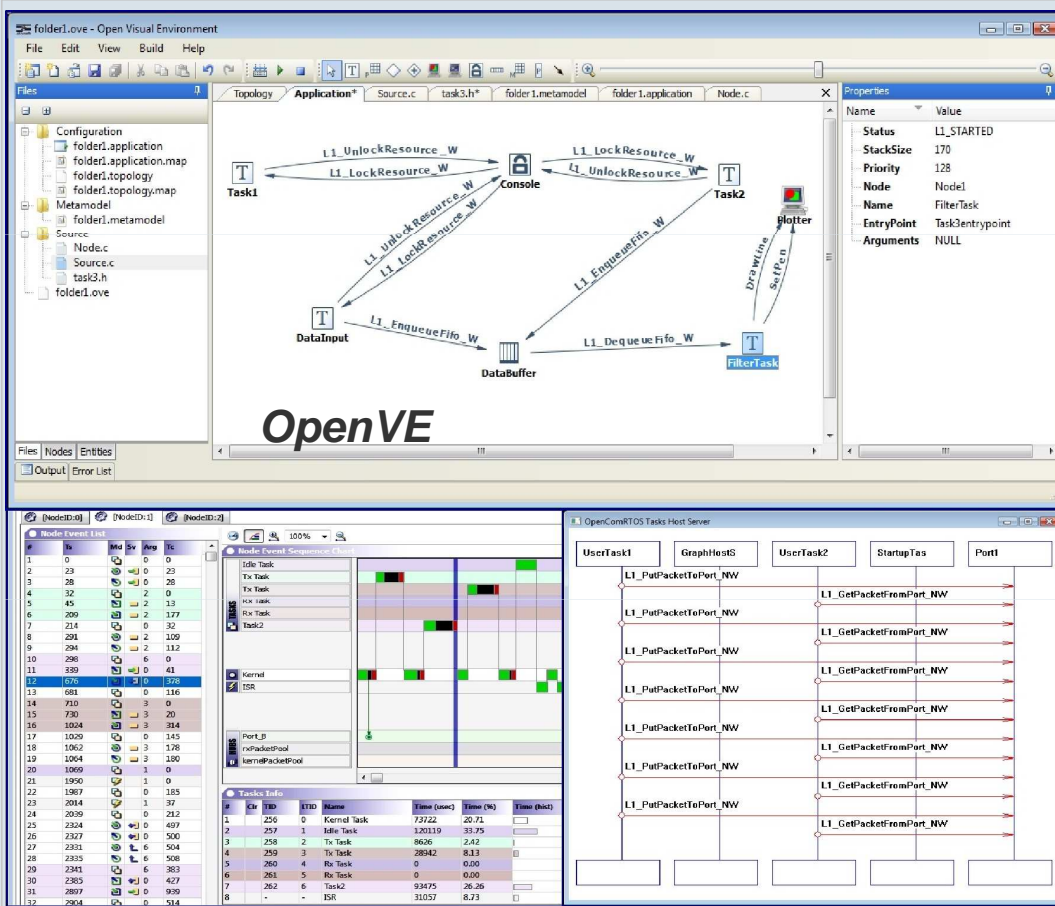


28/02/2010

Altreonic confidential

8

## Step2: Implementation Modeling



**After simulation and model checking, select the application architecture and start development**

28/02/2010

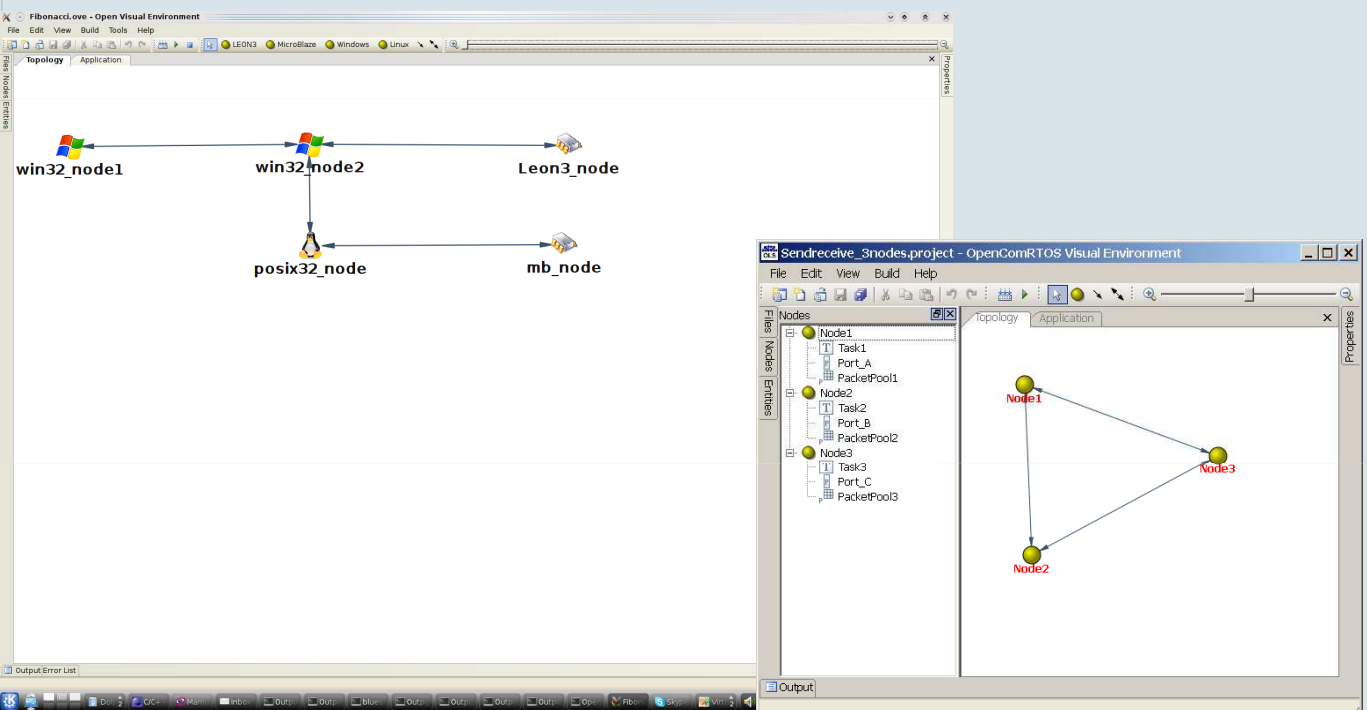
Altreonic confidential

9

## Step3: Select processing modules

**OpenVE:**

**How are processors connected ?**



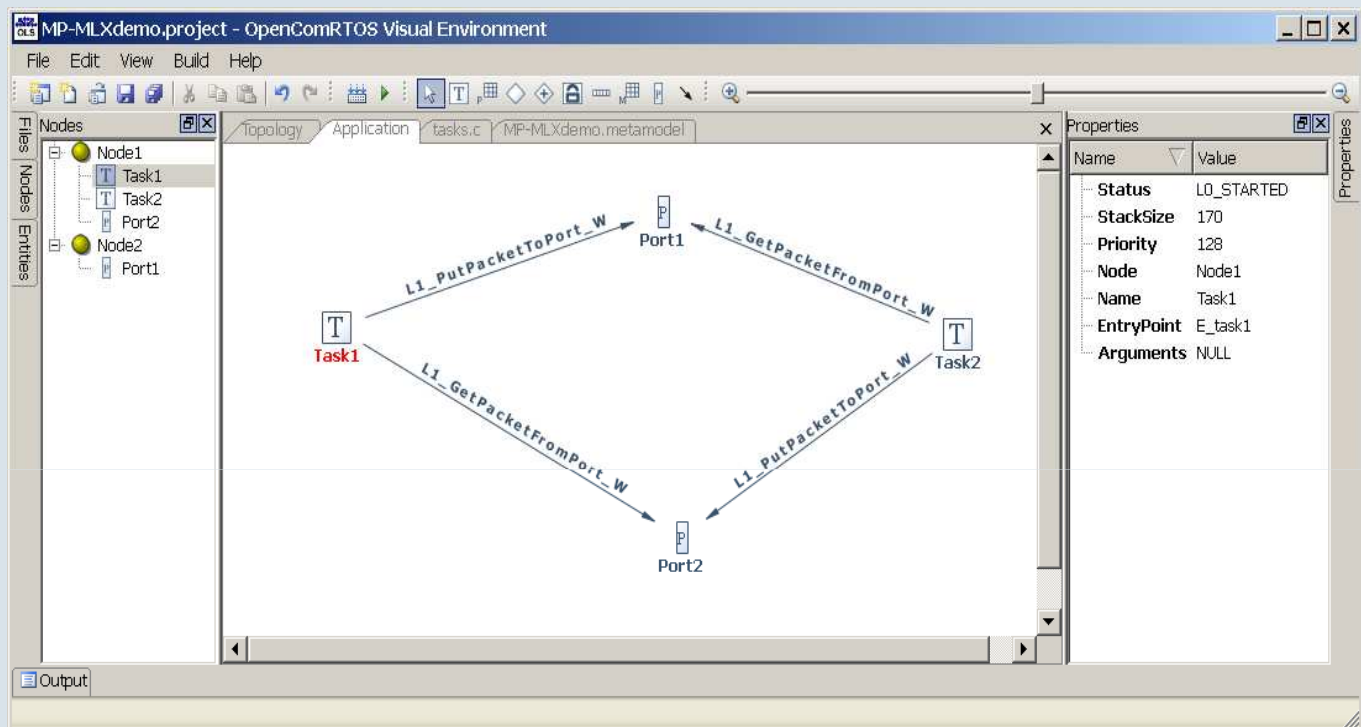
28/02/2010

Altreonic confidential

10

## OpenVE:

### How is the application structured ?



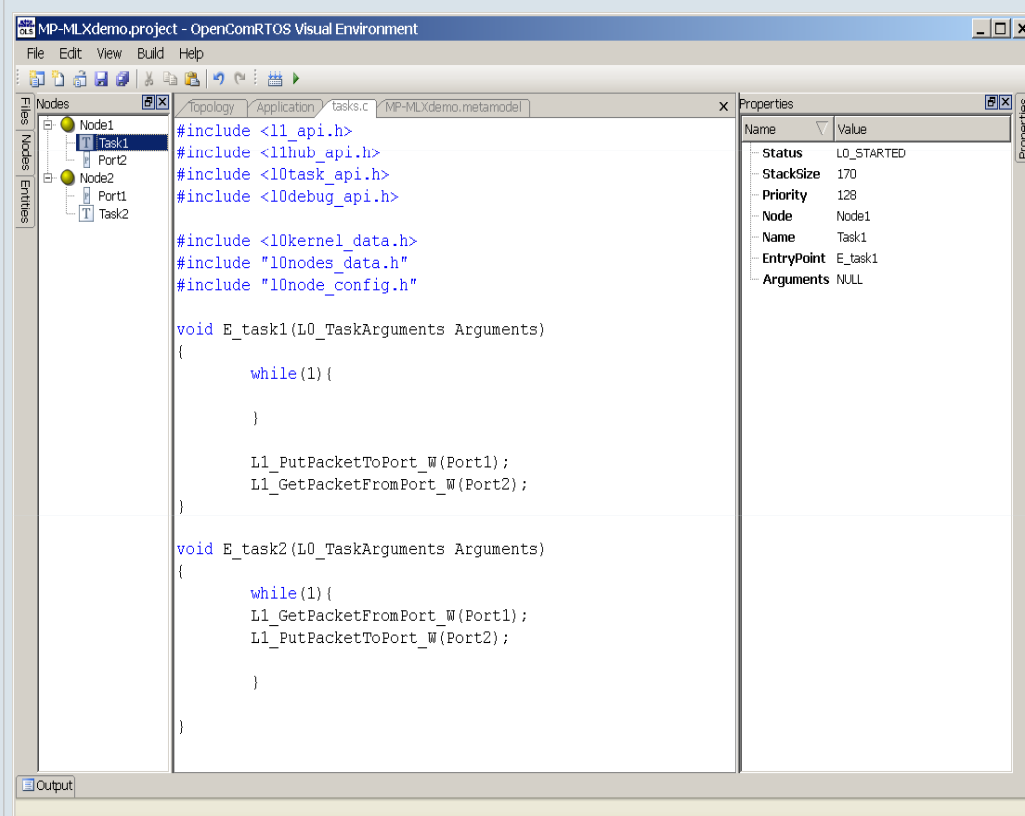
28/02/2010

Altreonic confidential

11

## Step4: Generate C for OpenComRTOS

### OpenVE: Code generation



```
#include <l1_api.h>
#include <l1hub_api.h>
#include <l0task_api.h>
#include <l0debug_api.h>

#include <l0kernel_data.h>
#include "l0nodes_data.h"
#include "l0node_config.h"

void E_task1(L0_TaskArguments Arguments)
{
    while(1) {

        L1_PutPacketToPort_W(Port1);
        L1_GetPacketFromPort_W(Port2);
    }
}

void E_task2(L0_TaskArguments Arguments)
{
    while(1) {
        L1_GetPacketFromPort_W(Port1);
        L1_PutPacketToPort_W(Port2);
    }
}
```

*Auto generated  
code is ~ 10  
times more  
reliable than  
human  
generated code.*

28/02/2010

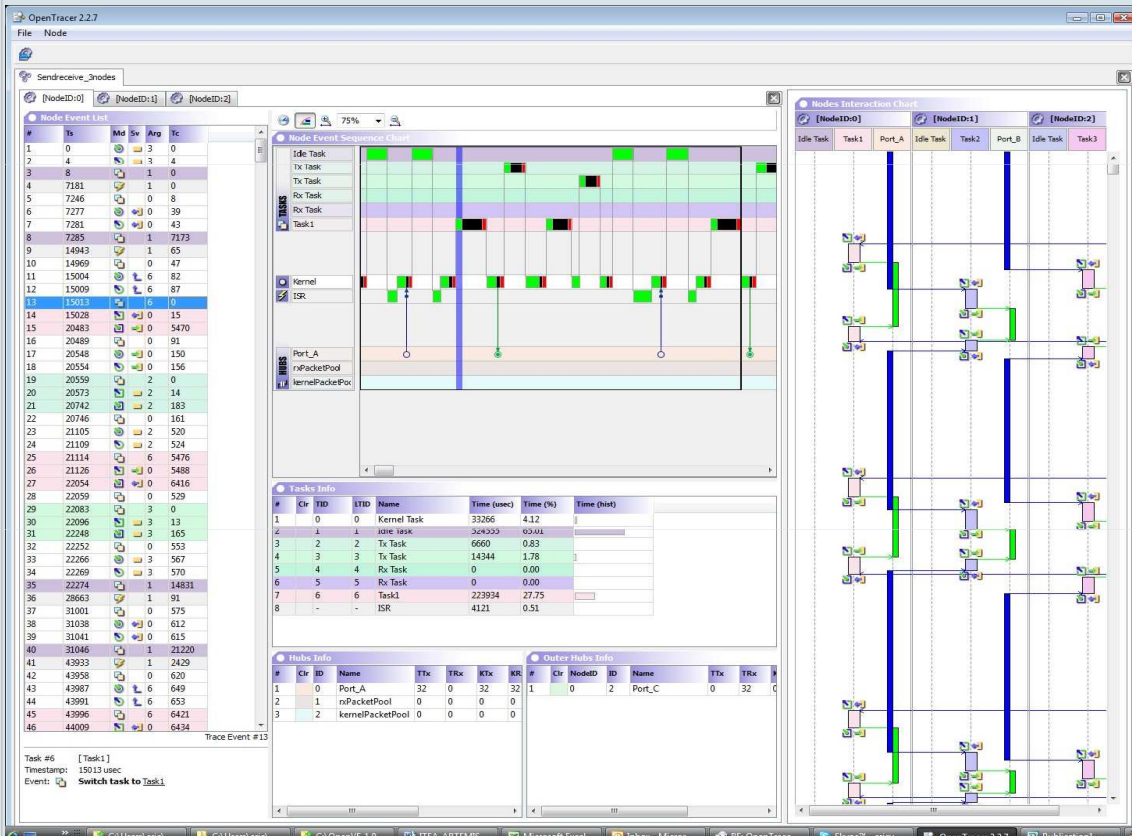
Altreonic confidential

12



# Step5: Run and verify

## OpenTracer



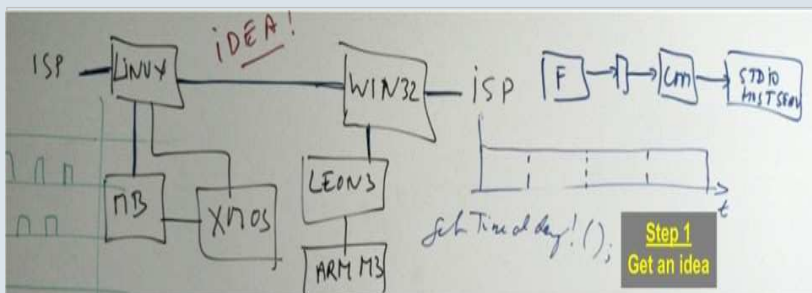
Verification and testing is needed to confirm the work was well done

28/02/2010

Altreonic confidential

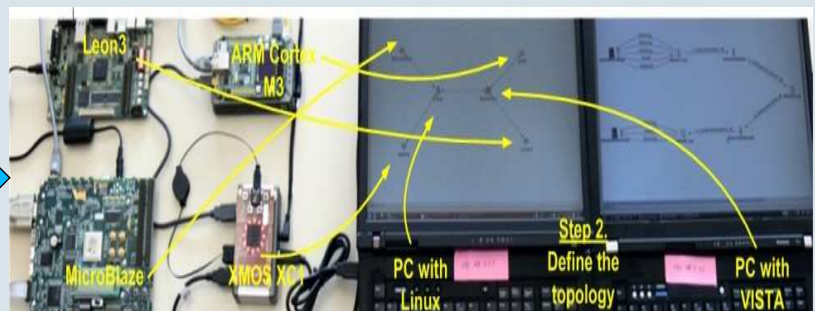
13

## What happened?



Initial idea

Processor topology



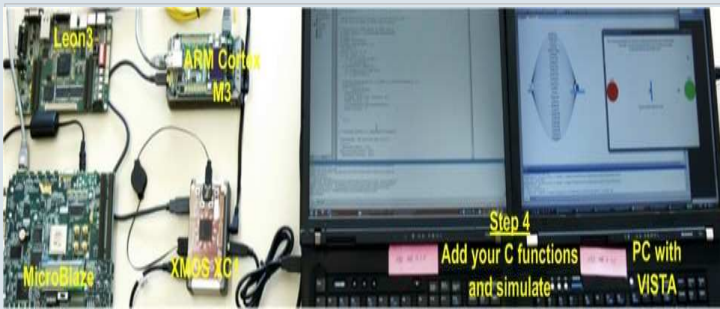
Application architecture

28/02/2010

Altreonic confidential

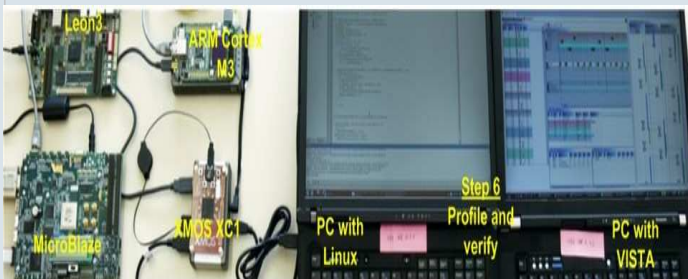
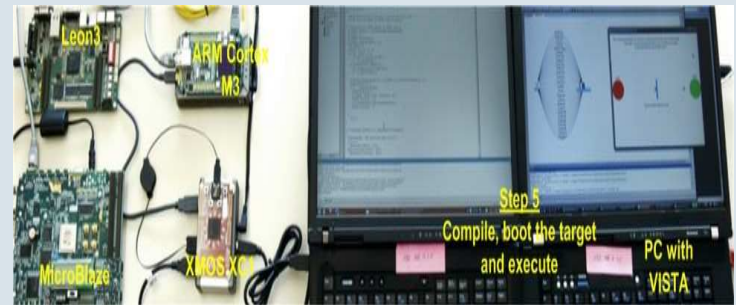
14

# Transparent and processor independent!



← C code generation

Compile, boot, execute →



← Profile and verify

28/02/2010

Altreonic confidential

15

## Use case 1: mobility aids



**Future of transport is consumer-friendly**

- Elderly customer base
- Seamlessly Indoors ↔ Outdoors
- Active safety
- Optimal use of road network



**Intelligent Transport Systems ITS, using cooperative Embedded Systems**

- 100% trust-worthy
- Fault Tolerance
- Heterogeneous network support
- Scalability
- Cost-Efficiency

28/02/2010

Altreonic confidential

16



## Use case 2: e-wheel control algorithm

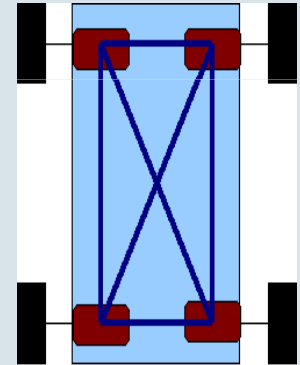


### Key characteristics :

➤ High Reliability (SIL3) → Fault Tolerance (SIL4)

All-in:

- Traction
- Braking
- Anti-slip
- Stability control
- Active suspension



➤ Exploits transparent distributed operation of **OpenComRTOS**.

➤ Software and Hardware redundancy enables **fault-tolerant controllers**

1-, 2-, 3-, 4-, n-wheel platforms



## Altreonic SIL4 Controller project

### • Key characteristics :

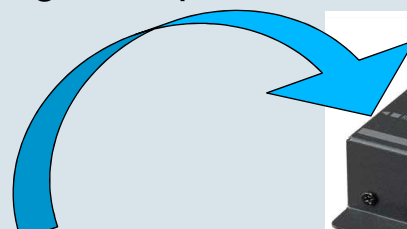
- High Reliability (SIL3) → Fault Tolerance (SIL4)

### • Target market :

- Robotics, Automotive, Transport, Aerospace, Machine Control.

### • Technological competencies/partners sought:

- Input from Use cases / Application scenarios
- Control systems design competence
- System Simulation



# What are we working on?



- Safety standards awareness in OpenCookbook
- Asynchronous 2-phase services for OpenComRTOS (feedback at application level)
- Protocol hubs (protocol composition OpenVE)
- Virtual C-machine (20 Kbytes)
- Dynamic resource scheduling
- SIL3/4 embedded controller

⇒ **Enabling technologies for wide-spread use of safety engineering**

⇒ Open to partnerships and joint projects

## Conclusion



- Altreonic's know-how; **20 years experience**

⇒ Trustworthy partner

- **Unique** products for **high added value**

⇒ Trustworthy products and systems

- Open Licensing scheme = **no risk**

**[www.altreonic.com](http://www.altreonic.com)**