Ocarina, an AADL-to-X generator: status & work in progress

Jérôme Hugues, TELECOM ParisTech

AADL at TELECOM ParisTech

- **Involved in AADL since 2004**
- **Used AADL as part of our research activities on middleware and Distributed Real-Time and Embedded systems**
- **Key idea: use AADL to configure and deploy applications**
  - Use a compiler approach to generate support code for distribution, concurrency, buffer allocation and **checks**
  - vs. relying on a huge framework a-la CORBA

- **Open source projects:**
  - Ocarina: toolbox for AADL
  - PolyORB-HI: C/RT-POSIX and Ada2005 runtimes for AADL
  - POK: Partitioned OS and runtime for AADL

Ocarina features

http://aadl.telecom-paristech.fr

- **Ocarina is a stand-alone tool for processing AADL models**
- **Fully supports both AADLv1 and AADLv2**
- **Prototype support for the Behavioral Annex (up to 2.9)**
- **Code generation facilities target AADL runtimes**
  - Ada HI integrity profiles, with Ada native and bare board runtimes
  - C POSIX or RTEMS, for RTOS & Embedded
  - User code can be C, C++, Ada, Esterel, Simulink , Lustre, SCADE
- **Model to model transformations**
  - WCET analysis of AADL runtime + user code: Bound-T for LEON
  - Model checking specifications using Colored or Timed Petri Nets
  - Constraint language to validate AADL model

Ocarina features (2/2)

- **Ocarina proposes an API to build your own AADL tools**
  - Like Ocarina itself, but also Cheddar (UBO), LabASSERT (ESA)
  - Parsers, printers, semantic checks, model transformation
  - Compiler-based approach, rather than model-to-text

- **Add-ons**
  - Emacs and vim modes
  - Eclipse plug-in for integration with OSATE

Ocarina visibility

- **Used in the IST-ASSERT (9/2004 -> 1/2008) projec**
  - Validated on industrial case studies
- **Ocarina & AADL used jointly in ANR Flex-eWare and MOSIC**
  - Evaluation of DRE models performance, code generation
  - In a CCM context, mapped onto AADL models
- **Ocarina is part of the TopCased project**
  - Part of the “model bus” philosophy of Eclipse
- **Ocarina featured on http://libre.adacore.com**
  - Open source projects hosted by AdaCore
  - Enhance visibility from the Ada community
  - Highlight benefits of AADL tools for the HI domain

Ocarina distributions

- http://aadl.telecom-paristech.fr/
- **Ocarina 2.0 wavefront, daily snapshots**
  - Binaries of Ocarina (release 1.2 and nightly builds)
    - For GNU/Linux, Windows, Solaris, Mac OS X, FreeBSD
  - Documentation and examples (30+ available)
  - Scientific papers on the use of AADL
  - Teaching materials for Master degree
- **PolyORB-HI AADL runtimes**
  - Ada 2005 and C/POSIX
- **POK runtime**
  - For IMA systems, using time and space partitioning

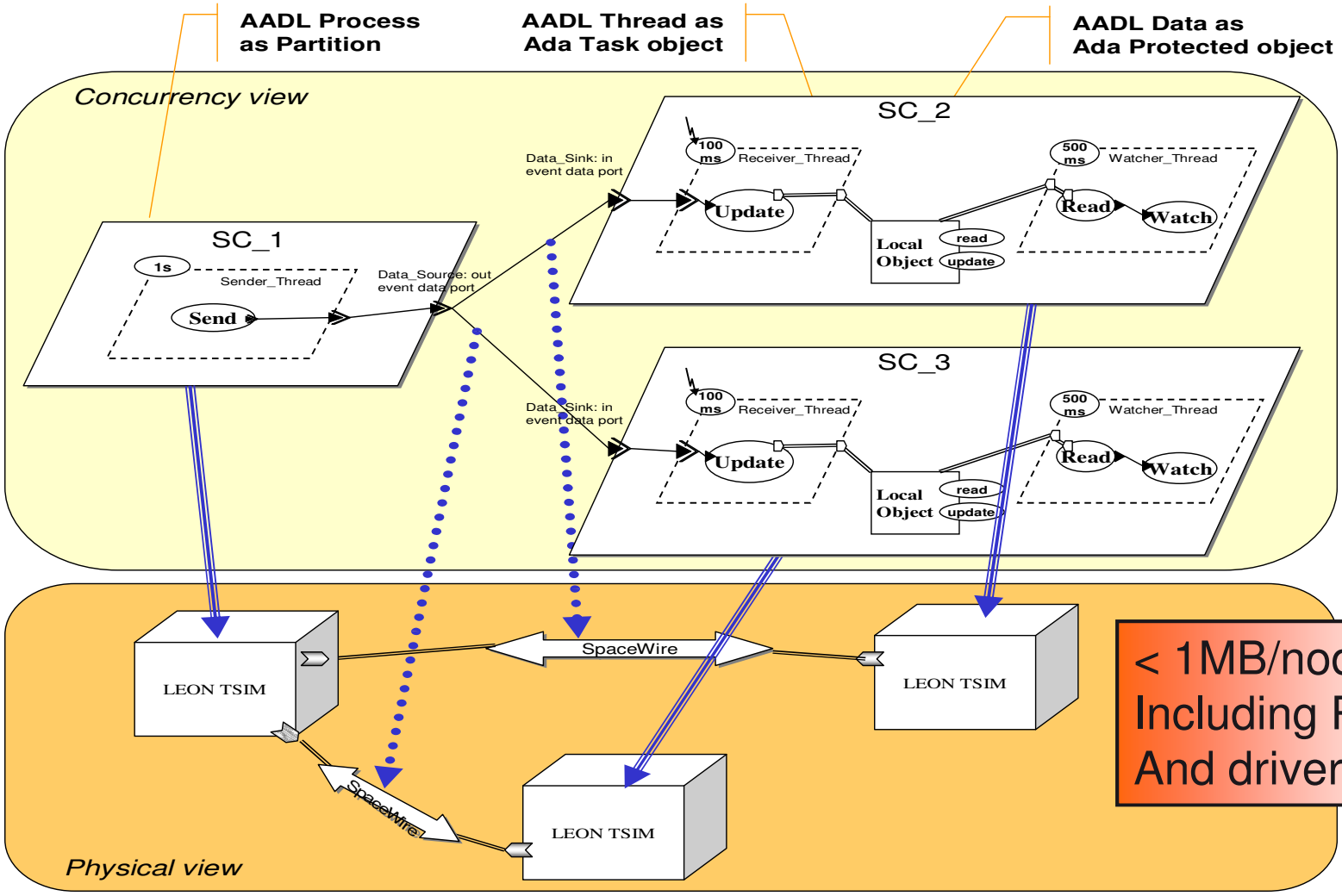
Ocarina's AADL runtimes 1/2

- **PolyORB-HI/Ada**
  - Target Ada Ravenscar and High-Integrity runtimes
  - Supports AADL semantics, v1 and v2
    - Need more tests to validate corner cases and extended use of AADL
  - Based on the Ravenscar & HI Ada profiles
    - Meets stringent requirements from ESA
  - Supports native, LEON2, ERC32 targets
    - With Ethernet or SpaceWire connections
    - Runtime can be configured to use other drivers
- **Validated in the context of IST-ASSERT with ESA**

Ocarina's AADL runtimes 2/2

- **PolyORB-HI/C**
  - Targets C/POSIX and C/RTEMS
    - Set of macros to support other RTOS
  - Tested on multiple operating systems
    - Native, GNU/Linux
    - Restricted libc: GNU/Linux on Nintendo DS and Nokia 770
    - POSIX RTOS: RTEMS
  - Tests demonstrated a limited subsystem of RT-POSIX & libc is enough to support AADL
  - Performance comparable to the Ada version
- **Used in the ANR Flex-eWare project by Thales**
  - 500+ downloads of the MyCCM-HI toolchain

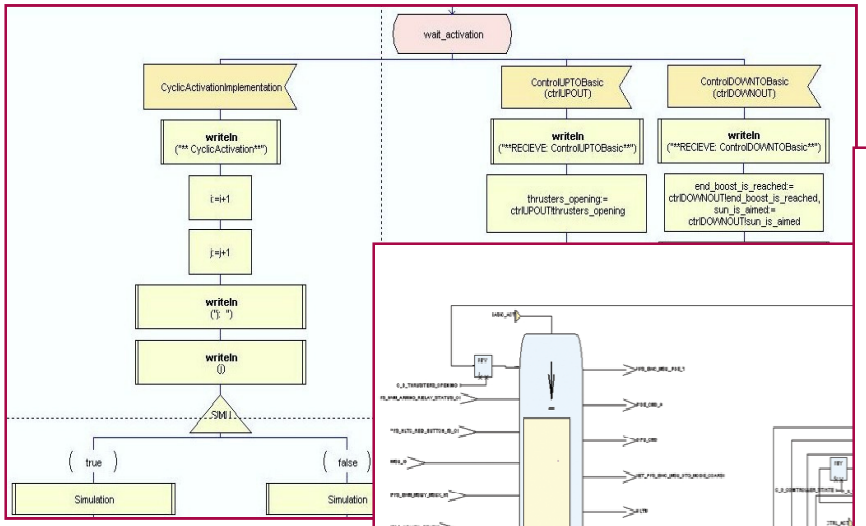
The ASSERT MPC V2 demonstrator (2007)

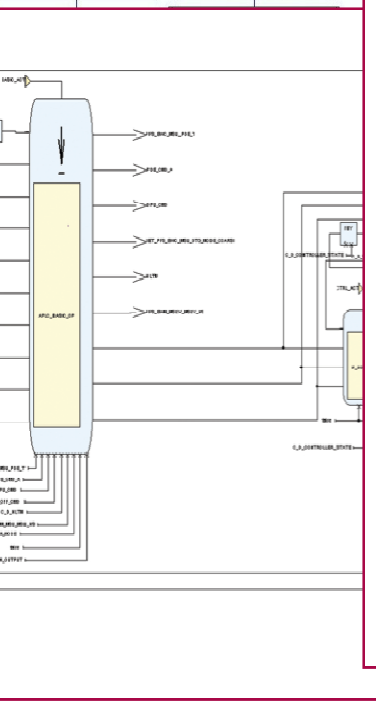


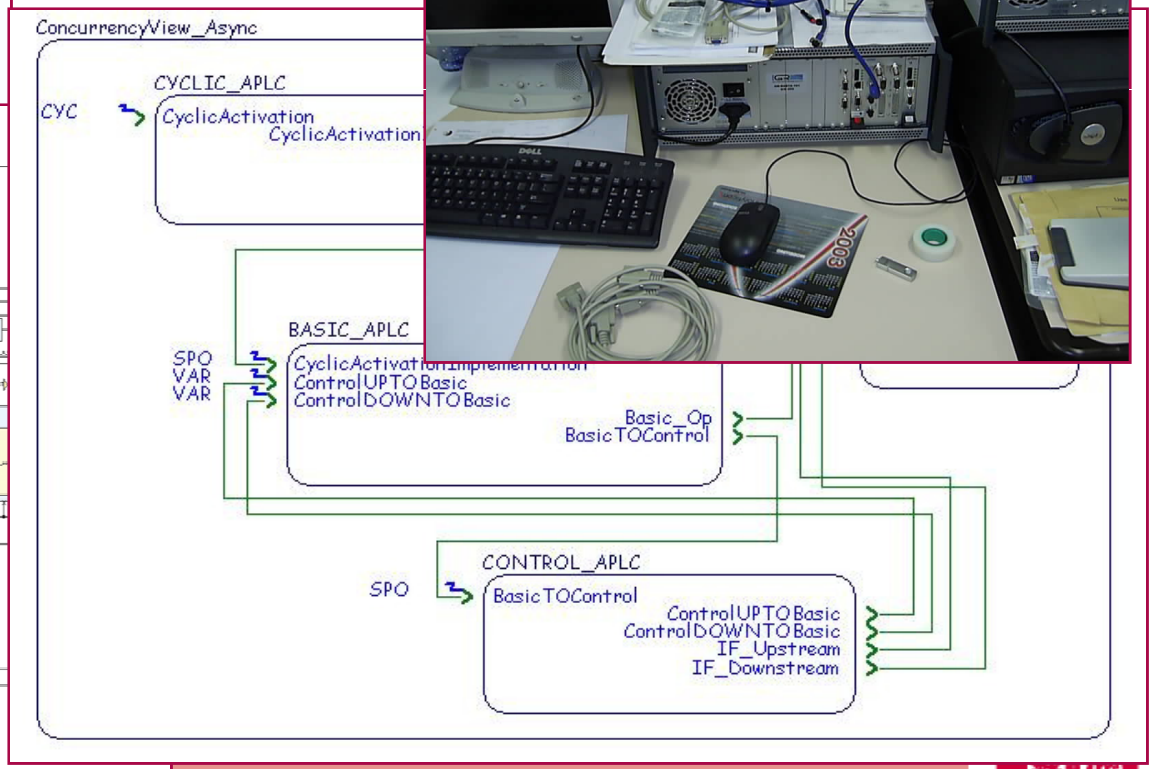
< 1MB/node,
Including RTOS
And drivers (60%)

The ASSERT ESA demonstrator (2008)

- Stood + Ocarina + ASN.1 tools demo
- Seamless integration of SDL, SCADE, Simulink, C, Ada, ASN.1 and AADL

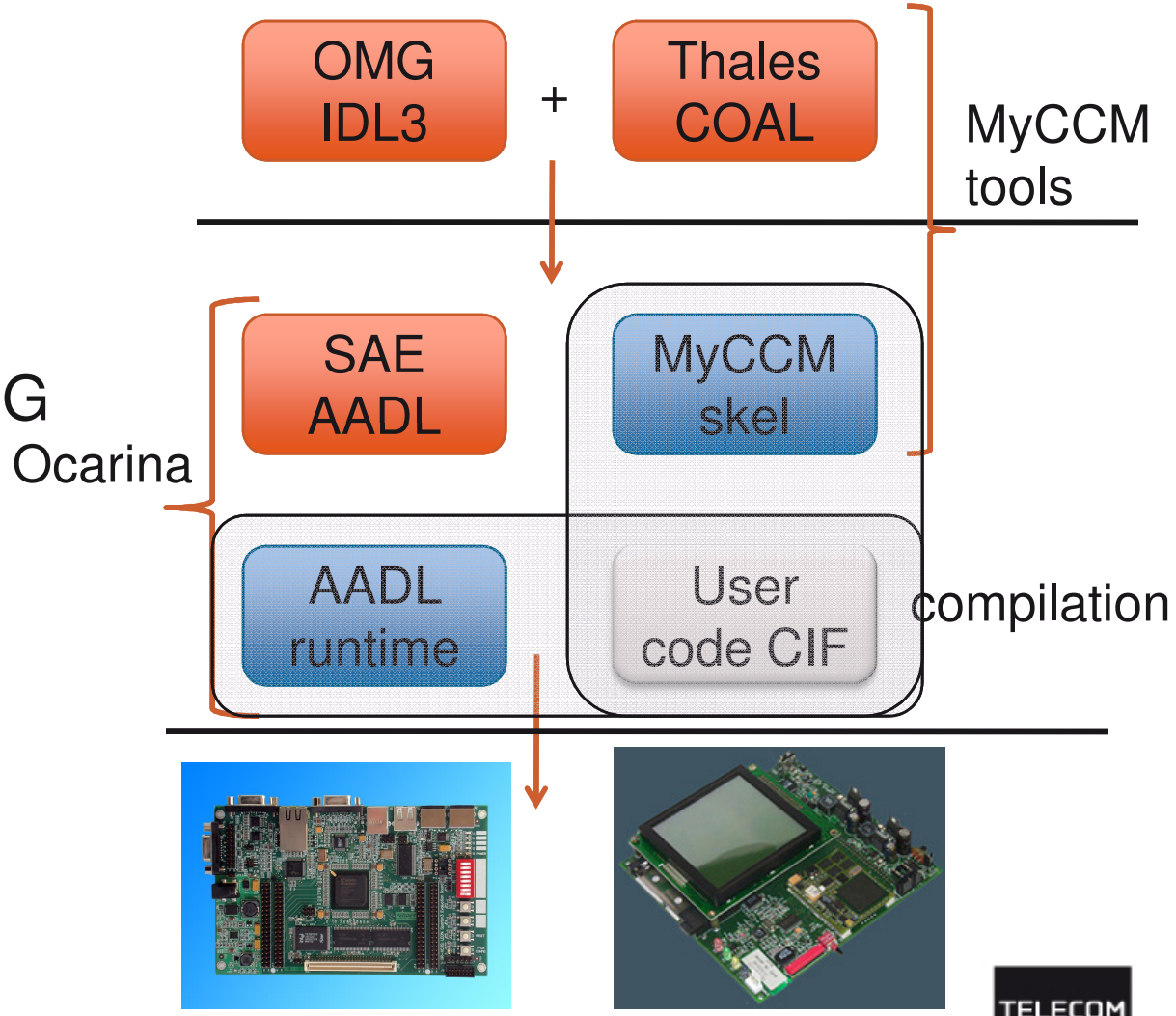






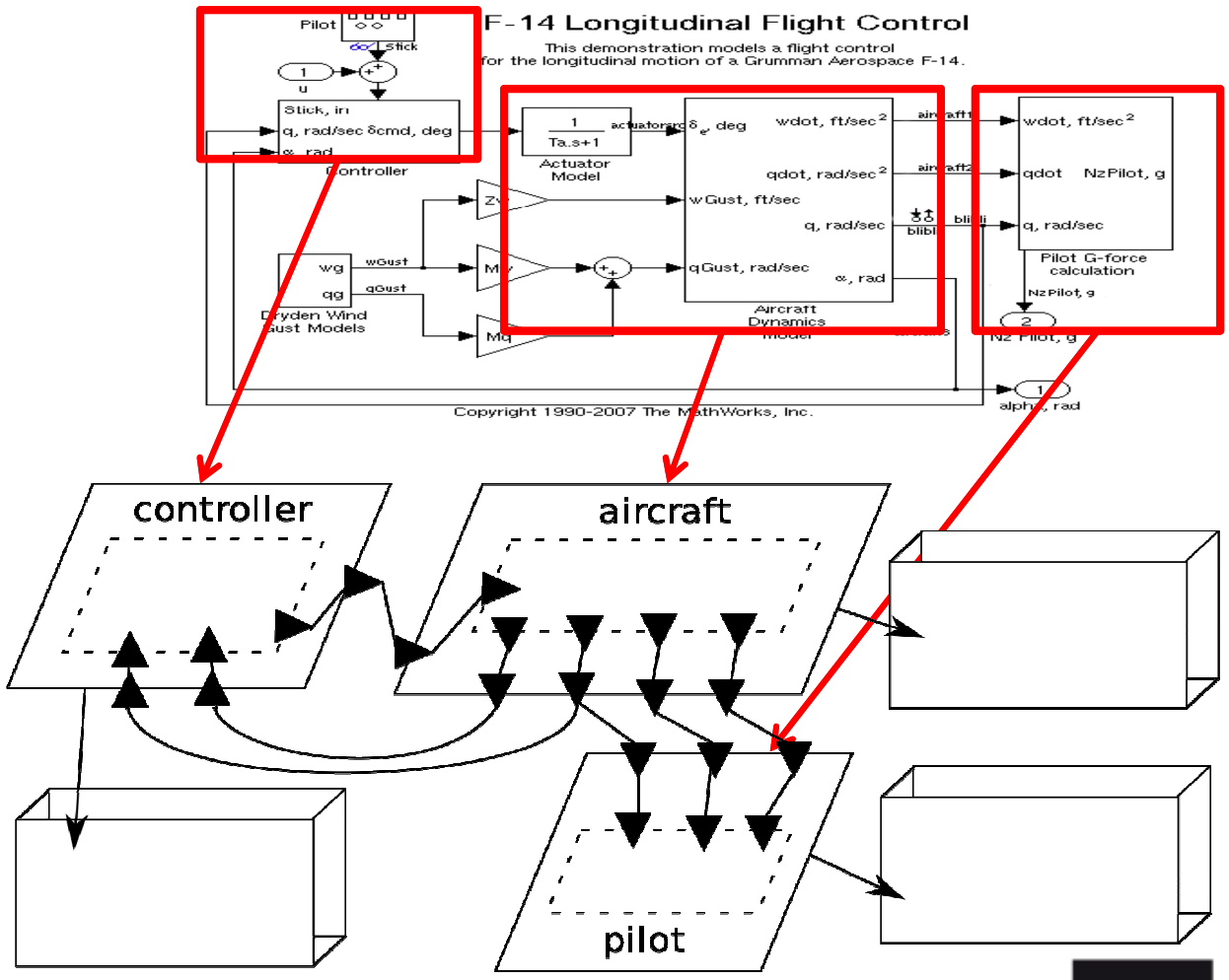
Flex-eWare project (2009)
Merging CCM and AADL

- **Using ASSERT philosophy: combining notations**
- **LwCCM is interesting for system designers**
  - Comfortable with the OMG
- **Map onto AADL for consolidation**
- **Generate code using Ocarina**
- **Uses AADLv2**



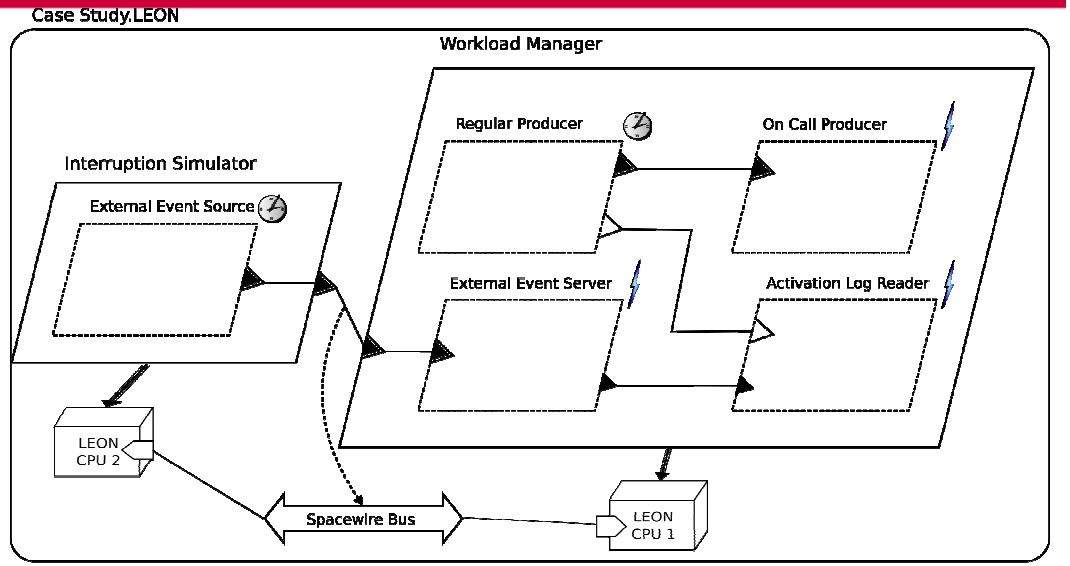
AADL + Simulink or SCADE (2009)

- Binding AADL and functional blocks seamlessly
- Complementary to ASSERT, without need for interoperability
- Achieve zero coding, mapping between simulation space and execution space



AADL vs. manual coding (2008)

- **Example from the "Guide for the use of the Ada Ravenscar Profile in high integrity systems »**
  - Model a pump system, typical example for RT systems
  - AADL generated code vs. Ada hand-coded
- **Same functional model**
  - Both are analyzable with RMA and RTA
  - Shares same code quality enforced by Ada compiler



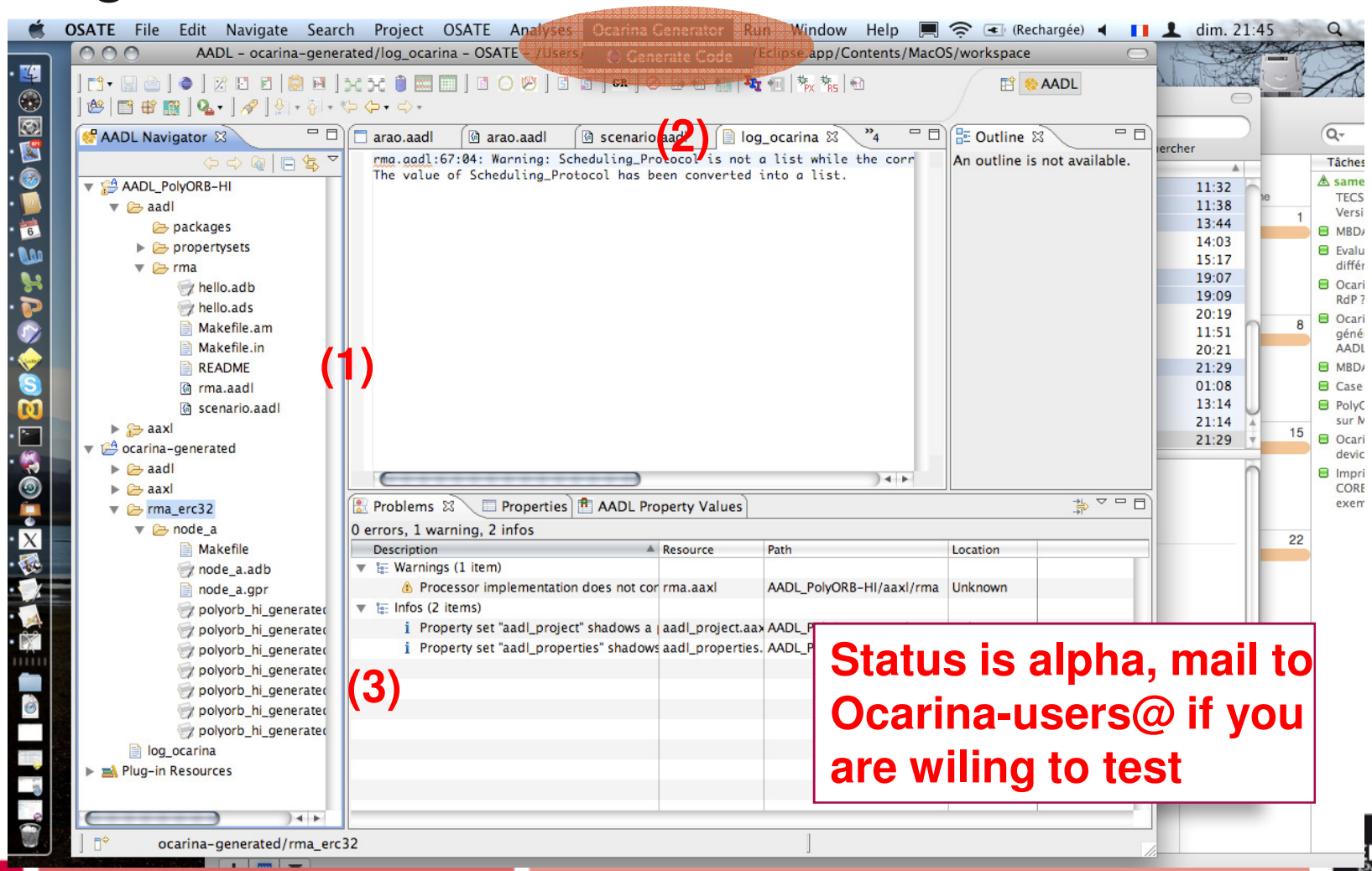
- **For LEON2 targets**
  - Penalty of 6% in memory size, equivalent WCET
- **Big improvement in analysis**
- **No coding**

Ocarina examples

- **A set of pre-built Ada generated examples available at http://aadl.enst.fr**
  - Examples from CMU/SEI, ASSERT, internal
  - For Linux, LEON and ERC32 platforms
  - Can be compiled for other native platforms
- **A set of educational material is available**
  - Build your own lab session using AADL
    - Then perform schedulability analysis, code generation, test
  - For master degree, or in-house tutorials

Ocarina's Eclipse plug-in

- Better integration with OSATE



Conclusion and Ongoing Work

- **AADL proved it is interesting for our partners to build and generate code**
  - IST-ASSERT, Flex-eWare, AdaCore, Thales, SAGEM, MBDA
- **Ocarina is now available as both source and binaries packages**
  - Use it, test it, report bugs to Ocarina's mailing lists
- **Some case studies are available, need more**
  - Do not hesitate to send us models !