Ellidiss Technologies

www.ellidiss.fr

Methods and Tools for Critical Software development

Ellidiss Software
www.ellidiss.com

UK based company
aka.TNI Europe Ltd
Tools sales office

Ellidiss Technologies
www.ellidiss.fr

Fr based company
New tools development
R&D center

HOOD

AADL

LMP

20 years + support to major industrial projects:

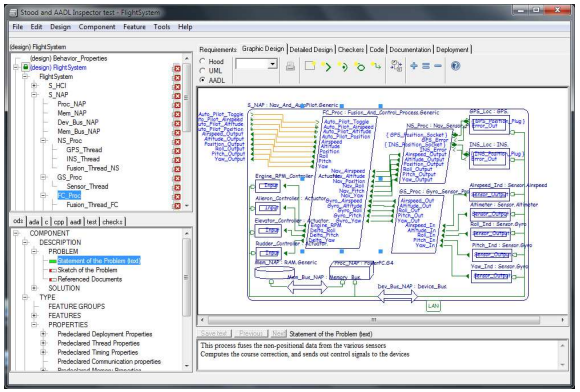
- HOOD Software design tools for Ada and C
- Eurofighter Typhoon
- Airbus A340, A380, A400M, A350
- Tiger Helicopter (mission calculator)
- Rafale (engine control)

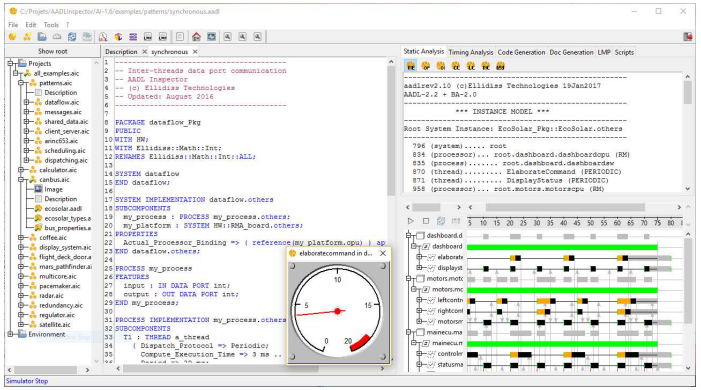
10 years + investement in new technology:

- SAE AS-5506: Architecture Analysis & Design Language
- AADL graphical modeling tools: Stood for AADL
- AADL analysis framework: AADL Inspector
- European Space Agency: TASTE Editors (Space SW development tools)
- European Commission: ERGO Project (Space Robotics)
- Generic model processing technologies: GMP, LMP, LMP Dev Kit

COTS Tools for AADL

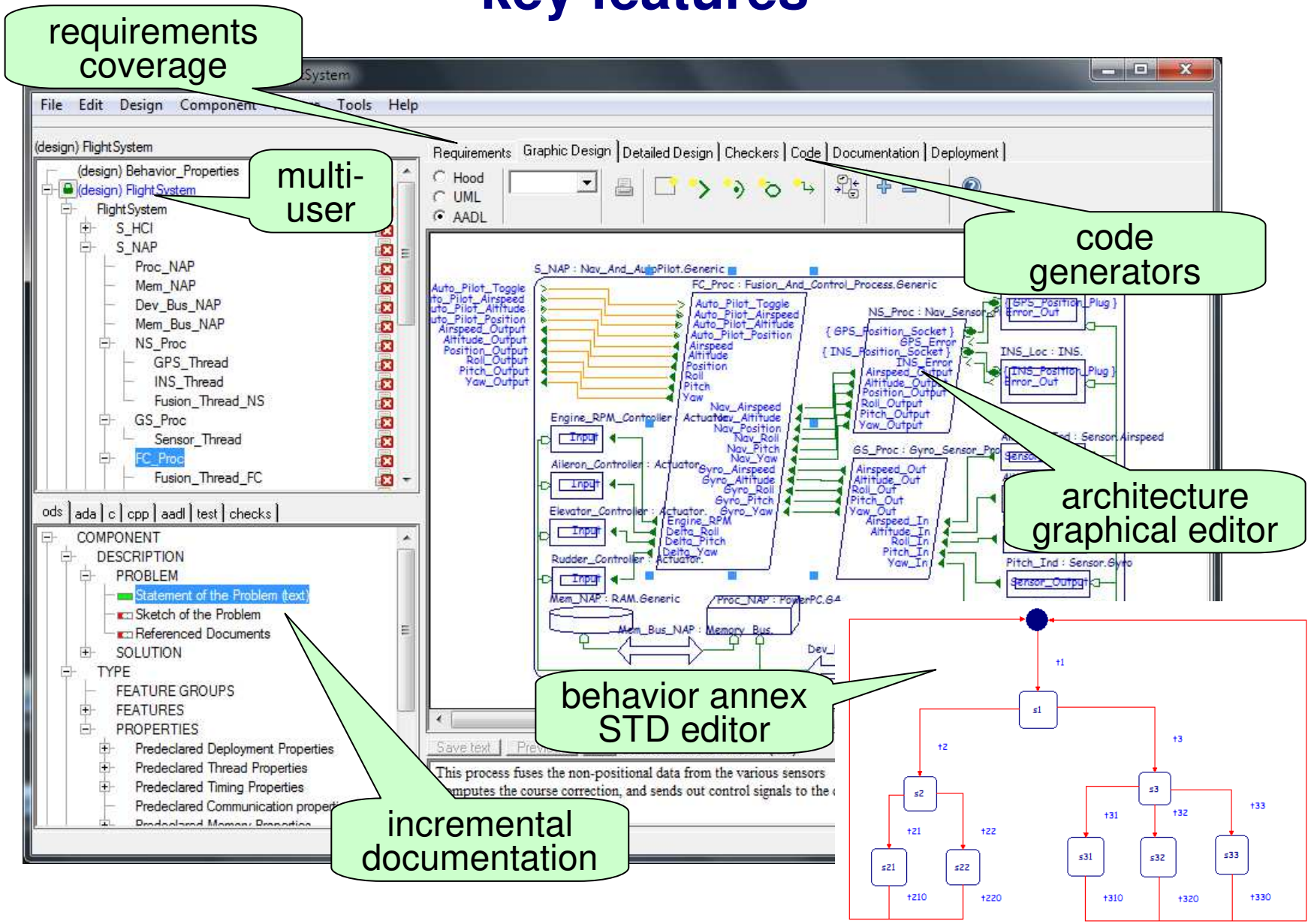
- **Design: Stood for AADL**
  - AADL project management
  - AADL Instance Model graphical editor
  - Requirement traceability
  - Documentation generator
  - Export textual AADL
- **Verification: AADL Inspector**
  - Import textual AADL
  - Model processing plugins
    - Static rules checkers
    - Scheduling analysis (Cheddar)
  - Simulation (Marzhin)
  - Pre-processors:
    - Import UML profiles (MARTE, SysML, …)
    - Import Domain Specific Models (XML)
- **Model Processing Toolbox: LMP**
  - Supported languages: AADL, Ada, C, XM* (XML, XMI, ECore)
  - Implementation: parsers + prolog engine and libraries





Ellidiss Technologies
www.ellidiss.fr

Stood for AADL key features

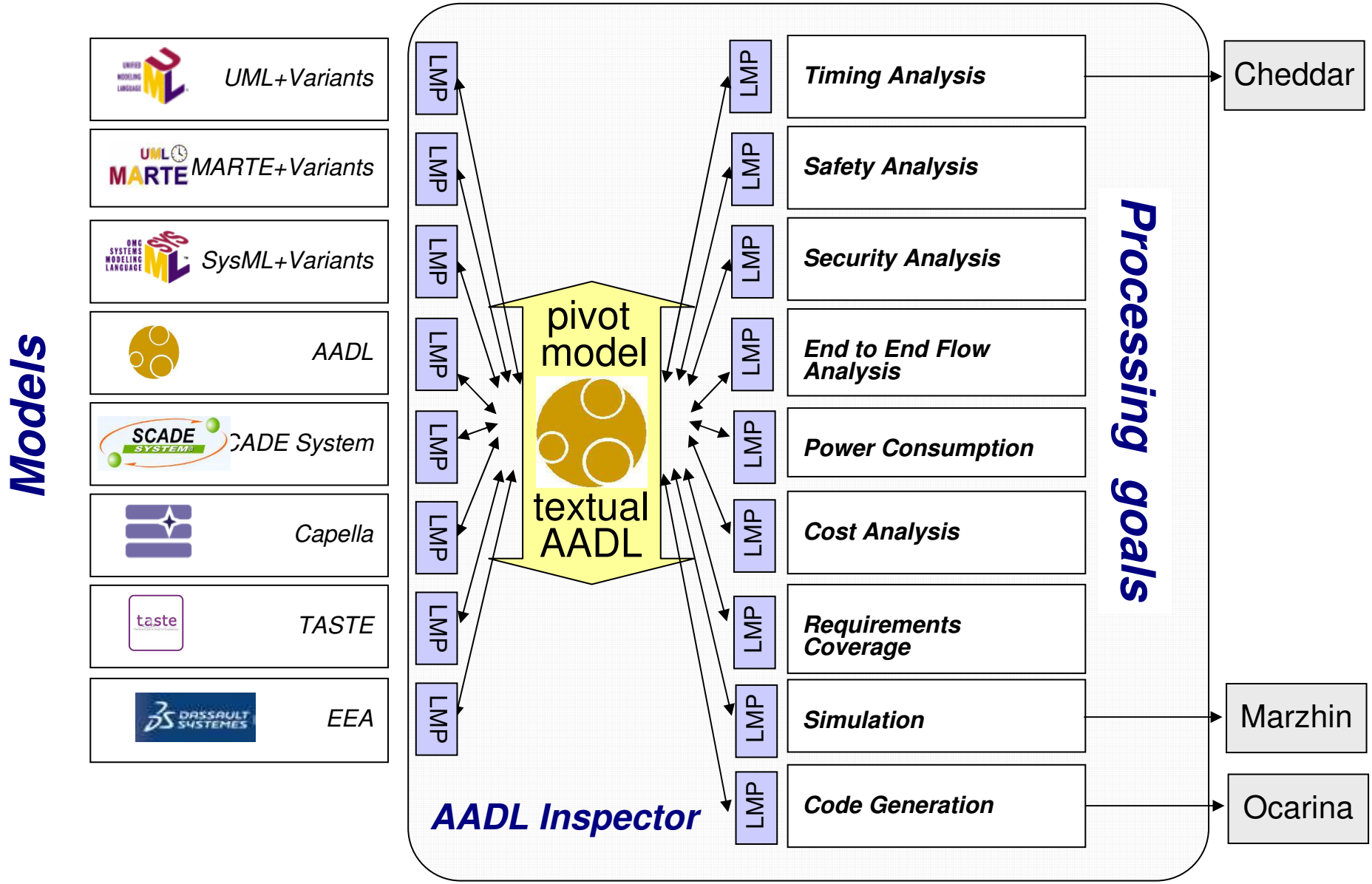


Ellidiss Technologies
www.ellidiss.fr

Top-Down modeling process for AADL

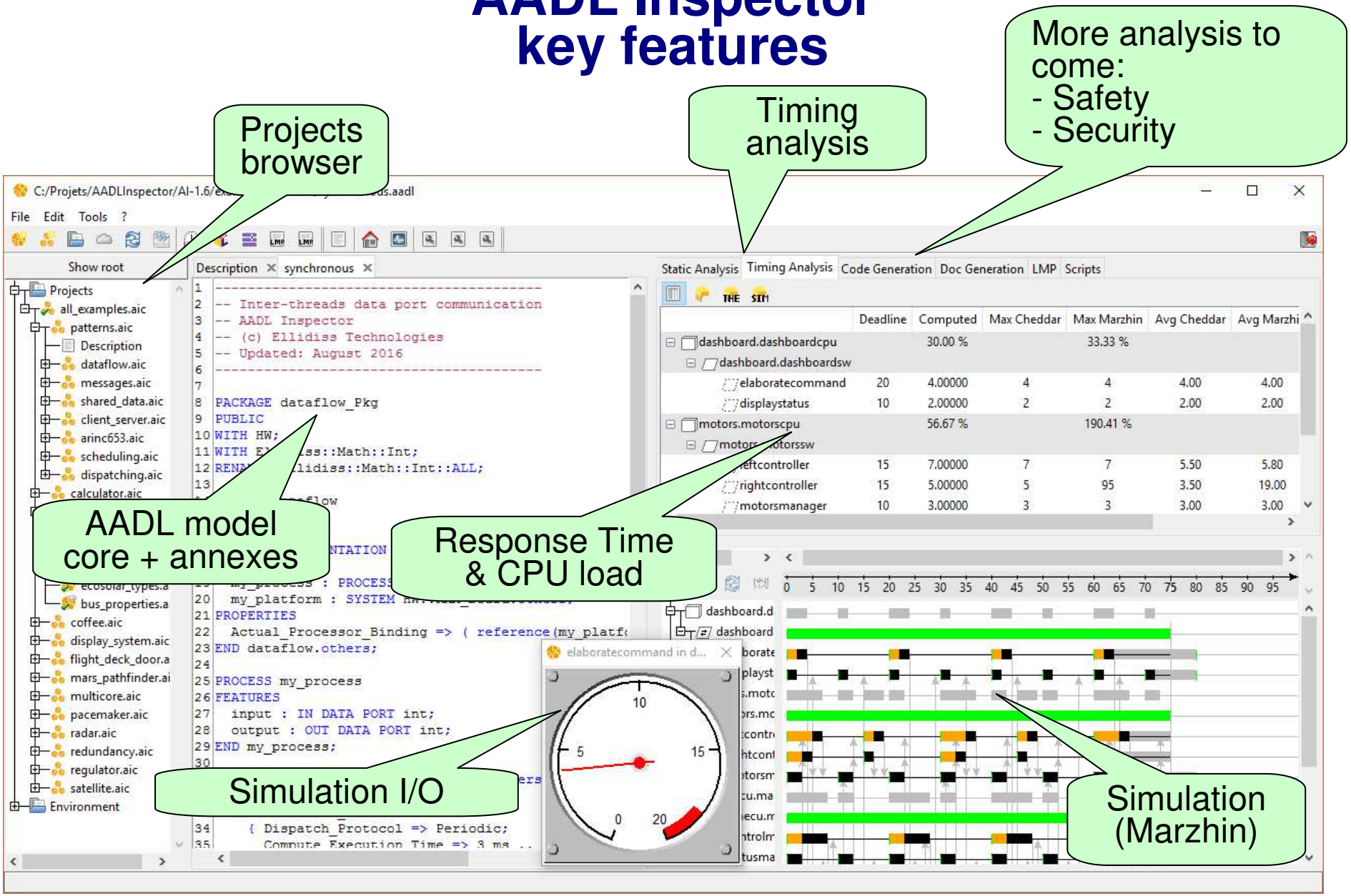
- Offers an industry proven practical modeling process to AADL designers
- Hierarchical Object Oriented Design (HOOD)
  - Inherits 20 years usage for the biggest European avionics projects (Airbus, Eurofighter)
  - Architectural Design (diagrams):
    - promotes SW engineering good practices
    - hierarchy of components with rigorous visibility rules (low coupling):
    - enable safe subcontracting (sub-trees)
    - ease testing, integration and maintenance
    - prevent from producing "spaghettiware"
  - Detailed Design (structured text):
    - keep track of design decisions
    - requirements coverage
    - supporting framework for design documentation, coding and testing
- Benefits for the AADL user (Stood for AADL)
  - Graphical editor of the AADL Instance Model (what you design is what you get)
  - Data Hiding enforcement (visibility rules, no provides data access)
  - AADL Declarative Model generator (textual AADL) for early verification activities
  - Complement AADL design activities with detailed design (documentation and coding)

Modular Analysis Framework



Ellidiss Technologies
www.ellidiss.fr

AADL Inspector key features



Ellidiss Technologies
www.ellidiss.fr

Products & Services Summary

- **Critical software development tools**
  - An industry proven SW design process: HOOD
  - Stood for AADL:
    - architectural design and AADL generator:
    - v 5.5 to be released soon
  - AADL Inspector 1.6:
    - early verification of AADL models
    - v 1.7 in development (more analysis plugins)
- **Tool development/integration technologies**
  - LMP Dev Kit: model processing toolbox (prolog)
  - GMP: DSL graphical editor framework
- **Services**
  - COTS tools sales support
  - Tools development, integration and support
  - AADL consulting and training
  - R&D partnerships

Ellidiss Technologies
www.ellidiss.fr

Demo:

Response Time based End to End Flow Latency analysis

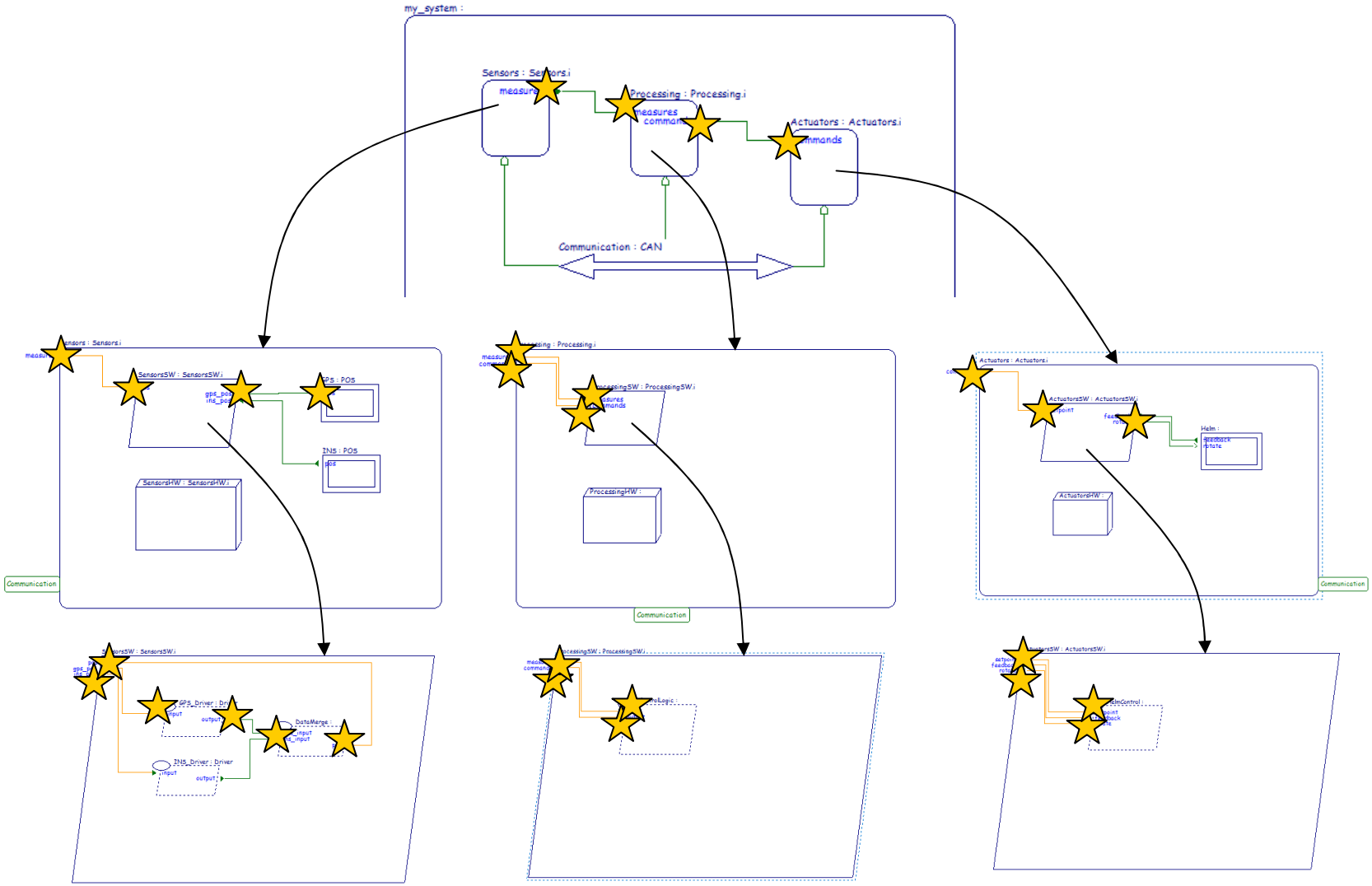
- **Graphical design with Stood for AADL**
  - Building the Instance Model with Top-Down approach
  - Specifying End to End Flows with port annotations
  - Generating the AADL Declarative model from the Stood model
  - End to End Flow generation will be available with Stood 5.5

- **Timing analysis with AADL Inspector**
  - Loading the Declarative AADL model
  - Running Response Time Analysis with Cheddar and Marzhin

- **Response Time based Flow Latency analysis**
  - More accurate than usual WCET/Deadline based analysis
  - Adding the Response Time of each Thread involved in the Flow
  - Use the Bus Messages/Thread analogy to compute bus communication latency
  - Response Time based Flow latency is not integrated yet (planned for AI 1.7)

Ellidiss Technologies
www.ellidiss.fr

AADL Instance Graphical Design



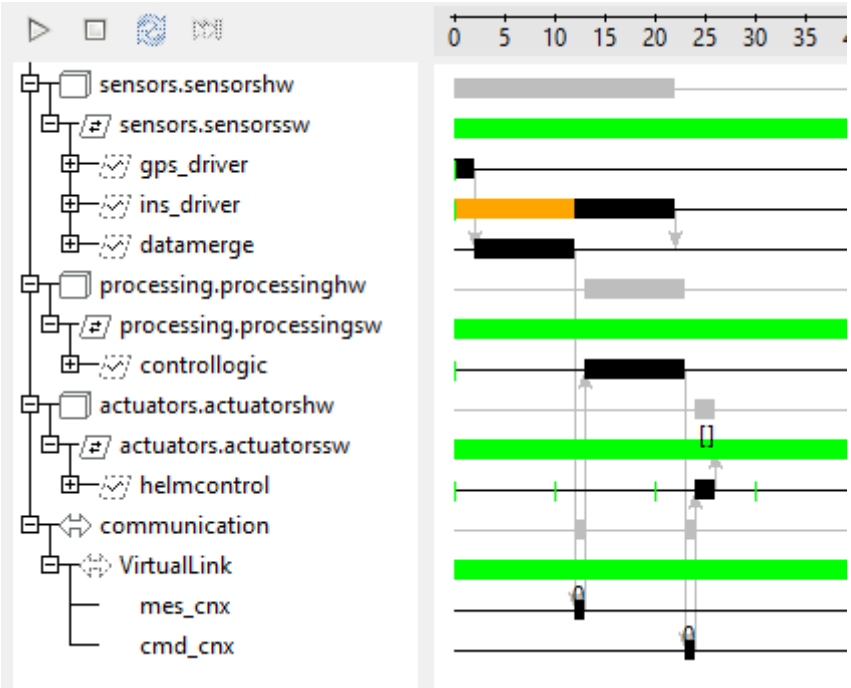
Flow annotation

Ellidiss Technologies
www.ellidiss.fr

AADL Declarative Text generation

```
SYSTEM IMPLEMENTATION my_system.others
SUBCOMPONENTS
  Sensors : SYSTEM Sensors.i;
  Processing : SYSTEM Processing.i;
  Actuators : SYSTEM Actuators.i;
  Communication : BUS CAN;
CONNECTIONS
  mes_cnx : PORT Sensors.measures -> Processing.measures;
  cmd_cnx : PORT Processing.commands -> Actuators.commands;
FLOWS
  fl : END TO END FLOW Sensors.fl -> mes_cnx -> Processing.fl -> cmd_cnx -> Actuators.fl;
PROPERTIES
  Actual_Connection_Binding => ( reference(Communication) ) applies to mes_cnx, cmd_cnx;
END my_system.others;
```

Scheduling aware
End to End Flow latency analysis



	Deadline	Computed	Max Cheddar
sensors.sensorshw		24.00 %	
sensors.sensorssw			
gps_driver	100	14.00000	4
ins_driver	100	24.00000	14
datamerge	100	10.00000	24
processing.processinghw		20.00 %	
processing.processingsw			
controllogic	50	10.00000	0
actuators.actuatorshw		20.00 %	
actuators.actuatorssw			
helmcontrol	10	2.00000	0

Response Time based flow latency computation:

Rt(gps_driver) + Rt(mes_cnx) + Rt(controllogic) + Rt(cmd_cnx) + Rt(helmcontrol)