

# ***HOOD, STOOD and AADL***

***Ada Europe, Valencia  
June 2010***

***Tony Elliston***

# The ELLIDISS Group

*Model Driven Engineering with  
Component Based Approaches for  
Critical Systems and Software*



**UK based company**  
*TNI Europe Ltd*  
**Tools commercial office**



**FR based company**  
**New tools development**  
**R&D center**



# What do we do?

- **Develop and Market Software Tools**
  - Development, Distribution and Support of COTS toolsets
  - Consultancy:
    - Training Course
    - Tool Customization (code generators, ... )
    - Project Assistance
- **Research and Development**
  - Model Driven Engineering & Component Based Architectures
  - Bridging System & Software modeling activities
  - Contribution to R&D programs:
    - ASSERT (ESA, Astrium, Alcatel, ...)
    - TOPCASED (Airbus, ...)
    - SMP2 (CNES, ...)
    - SPICES (Thales, Airbus, ...)
    - AADL committee



# SOFTWARE TOOLS

- **CP HOOD**

- Current release 6.0
- Beta release 6.1

- **STOOD**

- Current release 5.2.2
- Special release 5.3

- **HRT UML**

- INTECS Toolset
  - Pure UML2 tool
  - Relevant to Space Industry
  - Generates Applications for Microsek

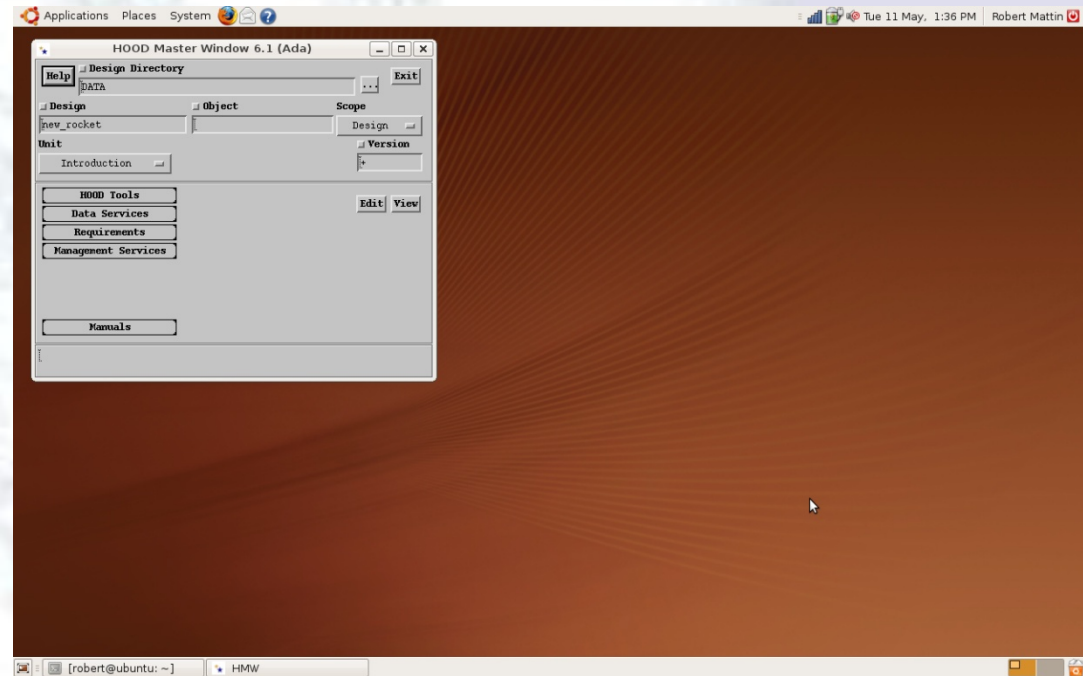
- **AADL INSPECTOR**

# **CP-HOOD 6.1**

**New Features**

# Linux Version

**CP HOOD is now available  
on Linux as well as Solaris**



# ADA 95

- Can use ADA 95 features such as tagged records, with clauses and protected types
- Can reverse engineer ADA 95 programs

HOOD Structured Text Editor 6.1 (Ada) [liss.com](http://liss.com)

Help Object End

Scope Layout File

Object ODS

Text Ops Other Editors

```
PROVIDED_INTERFACE
CONSTANTS
  NONE
TYPES
  Point IS
    tagged record
      X, Y: Real := 0.0;
    end record;
  Expression IS tagged null record;
  Painted_Point IS
    new Point with record
      Paint: Colour := White;
    end record;
  Literal IS
    new Expression with record
      Value: Real;
    end record;
  Expr_Ptr IS access all Expression;
  Binary_Operation IS
    new Expression with record
      Left, Right: Expr_Ptr;
    end record;
  Addition IS new Binary_Operation with null record;
  Subtraction IS new Binary_Operation with null record;
DATA
  Control : Resource;
  Flags : array(1..100) of Resource;
  Tree : Expr_Ptr;
DECLARATIONS

  protected type Resource is
    entry Seize;
    procedure Release;
  private
    Busy : Boolean := False;
  end Resource;

OPERATIONS
  Component (
    N : IN Index ) RETURN item;
  Set_Component (
    N : IN index;
    E : IN Item );
  Seize;
  Release;
OPERATION_SETS
  NONE
```

# Requirements Maintenance

- Program can be used without external requirements system
- Record text details of requirement
- Add new requirements
- Delete unfulfilled requirements

The screenshot shows the 'HOOD Master Window 6.1 (Ada)' interface. It features a 'Design Directory' section with fields for 'Design' (containing 'new\_rocket'), 'Object' (empty), and 'Scope' (containing 'Design'). Below this is a 'Unit' field (containing 'Introduction') and a 'Version' field (containing '+'). A 'Requirement ID' section contains a field with 'RR01' and a 'Name' field (containing 'rocket\_reqla'). There is also a 'Description' section with a text area containing 'First requirement for rocket'. Buttons for 'Help', 'Exit', and 'Cancel' are visible.

Design	Object	Scope
new_rocket		Design

Unit	Version
Introduction	+

Requirement ID	Name
RR01	rocket_reqla

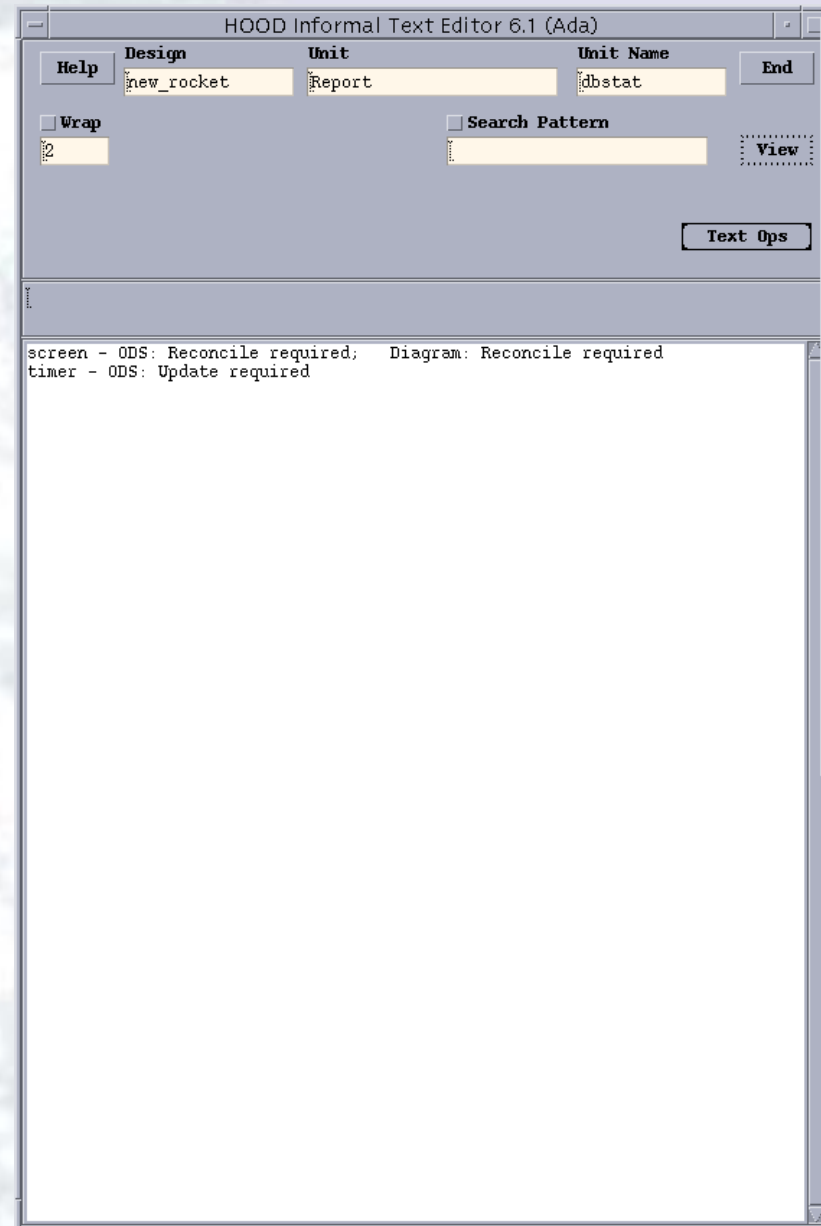
  

☐ Description

First requirement for rocket

# Database Status report

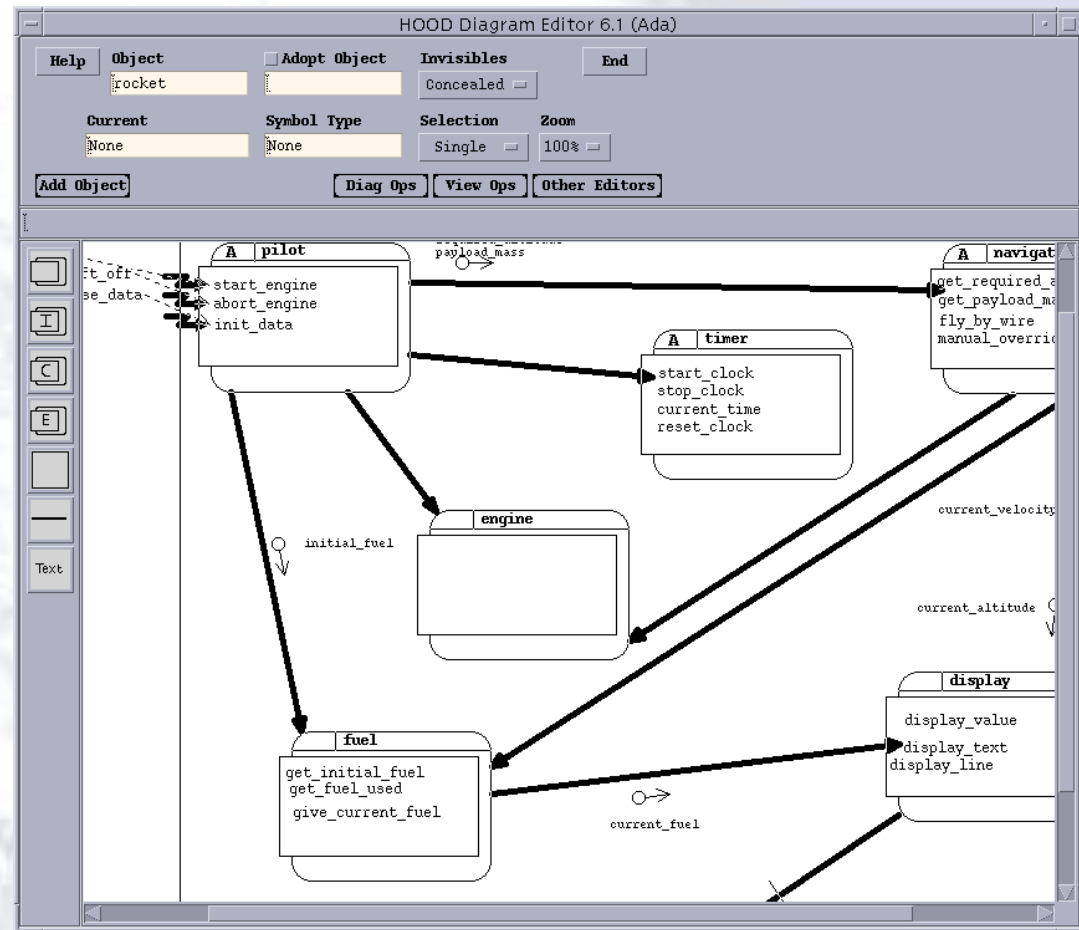
- Lists objects requiring updates or reconciles
- View on screen using text editor
- Alternately, output to printer or file



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

# Scrollbars and Zoom Menu

- Scrollbars for viewing part of large image
- Zoom menu for zooming out
- Available on all diagram types



# Improved Rename

The rename facility can now be used to rename units and whole designs as well as objects

The screenshot shows the 'HOOD Master Window 6.1 (Ada)' interface. At the top, there's a 'Design Directory' section with a 'Help' button on the left and an 'Exit' button on the right. Below this is a text field containing 'DATA8'. The main area is divided into three columns: 'Design', 'Object', and 'Scope'. The 'Design' column contains 'new\_rocket', the 'Object' column contains 'rocket', and the 'Scope' column contains 'Object'. Below these columns are two more rows: 'Unit' and 'Version'. The 'Unit' row contains 'HOOD DFD' and 'rocket', and the 'Version' row contains a '+' sign. At the bottom, there's a 'Scope' section with a 'Unit' dropdown menu set to 'Unit' and a 'New Name' checkbox. The 'New Name' checkbox is checked, and the text field next to it contains 'rocket'. A 'Cancel' button is located to the right of the 'New Name' text field.

Design	Object	Scope
new_rocket	rocket	Object
Unit	Unit Name	Version
HOOD DFD	rocket	+

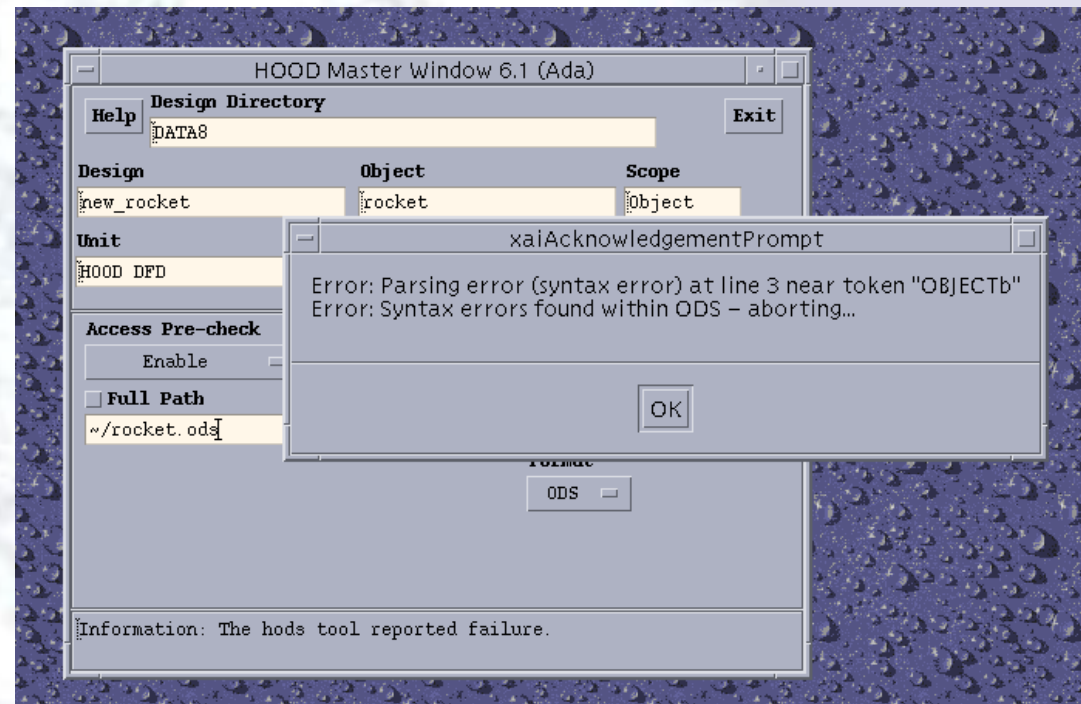
Scope ☐ New Name

Unit ☐ rocket

Cancel

# Import Error Messages

- Import ODS displays error messages immediately
- Messages still recorded in import.report as well
- With command line use, messages sent to stderr.



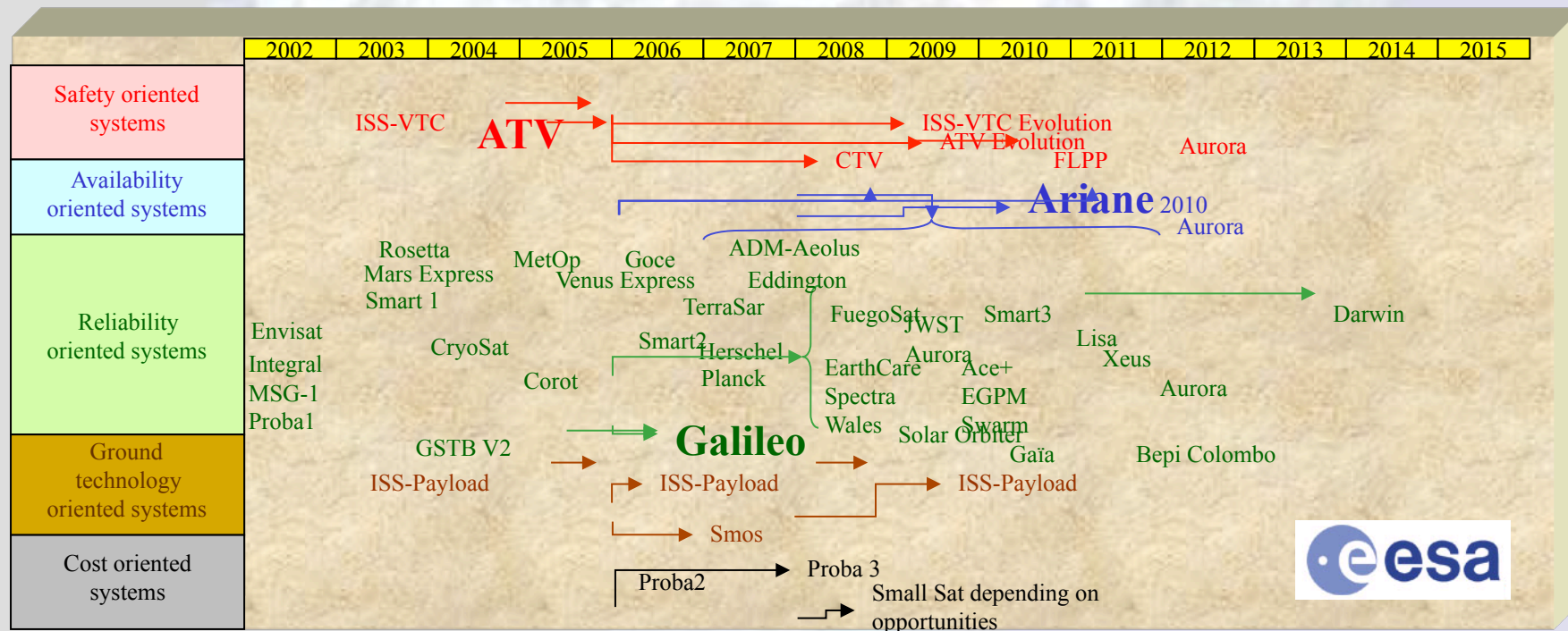
# **Other CP HOOD Enhancements**

- **Choice of page sizes when printing diagrams**
- **Option to output formatted documents as PDF**
- **Manual display does not block program**

## STOOD 5.2.2

- **Multi-standard software modeling tool:**
  - UML2.0, HOOD 4.0, HRT-HOOD and AADL 1.0
- **Well defined step-by-step modeling process:**
  - import of functional and non-functional requirements
  - multi-notation graphical design of the architecture
  - multi-language detailed design and coding (Ada, C, C++, ...)
  - static design verification tools
  - multi-format documentation generators (html, pdf, word, ps, ...)
  - code generators and reverse engineering (i.e. Ada <-> AADL)
- **Matured on important current mission-critical projects:**
  - multi-user, configuration management, requirements traceability
  - Unix-Windows interoperability
  - Complies with process standards:  
DO-178B, EN-50128, ECSS-E40, MIL-STD-498, ...

# Mission critical systems



# AADL Toolbox overview



# AADL Toolbox

- **STOOD**

- graphical support of AADL
- import/export of textual AADL
- connection with production/analysis tool (OSATE, OCARINA, CHEDDAR)
- download: <http://www.ellidiss.com>

- **ADELE**

- included into TOPCASED experimental package
- download: <http://www.topcased.org>

- **CHEDDAR**

- download: <http://beru.univ-brest.fr/~singhoff/cheddar>
- support available from Ellidiss Technologies

- **Developments :**

- TASTE: domain specific graphical editor (ASSERT) and AADL generator
- UML/MARTE to AADL model transformation
- Stood 5.3: now (special for the A350 program)
- Stood 5.4: planned Q4 2010 (support of AADL v2)
- AADL Inspector – Newly available - further test capability

# AADL Inspector

- **IMPORT of AADL textual specifications**
  - AADL V2
  - Behavioural Annex
  - AADL V1 compatibility
- **STATIC RULES ANALYSIS**
  - AADL legality, consistency and naming rules
  - Architectural metrics
  - Project specific rules
- **SCHEDULABILITY ANALYSIS**
  - Integration of CHEDDAR
- **DYNAMIC SIMULATION**
  - Multi-Agents simulation engine
  - Complies with the AADL run-time



# STOOD

## The AADL Coordinating 'Engine'

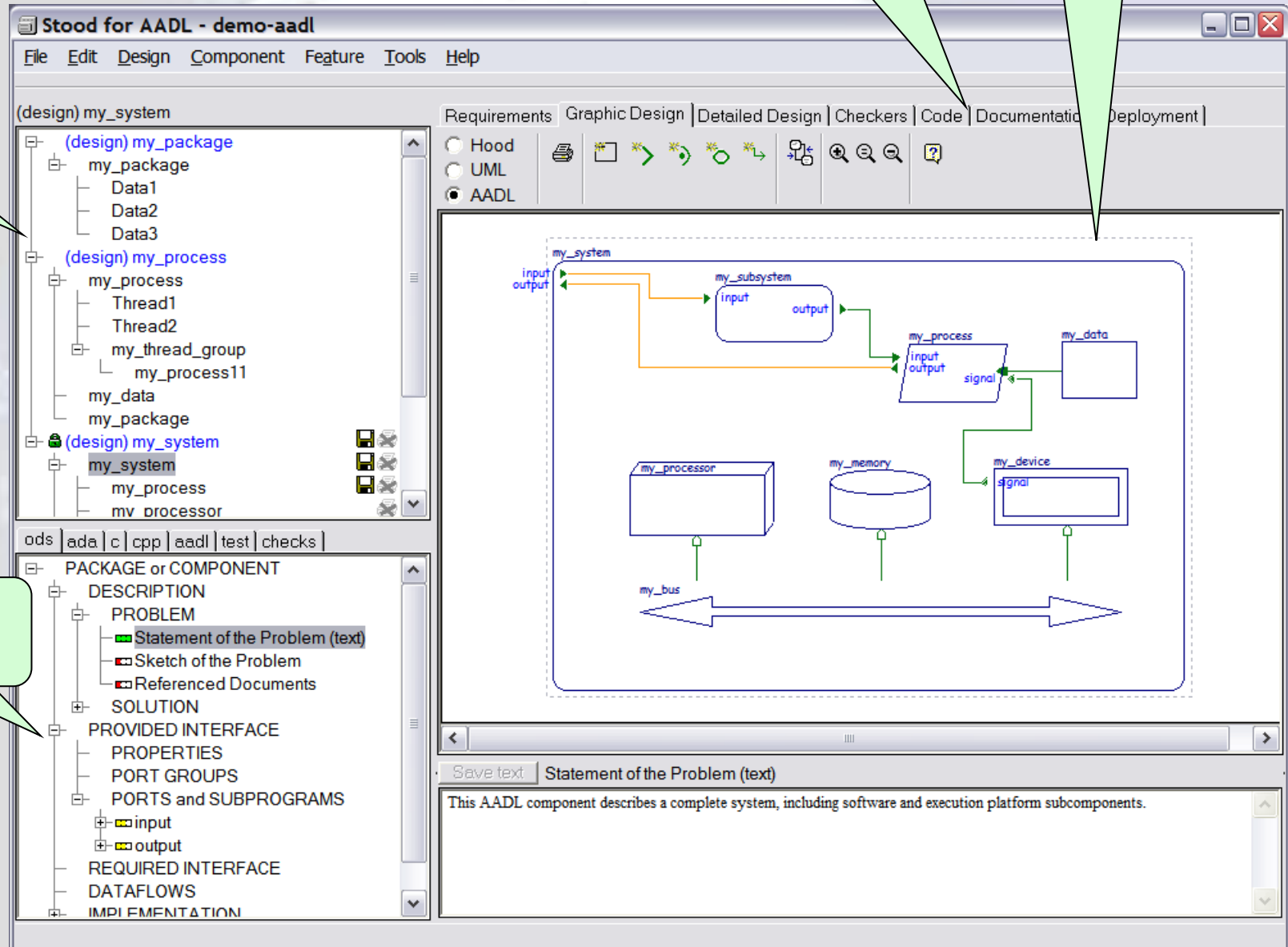
# Stood GUI

code and  
document  
generators

AADL  
graphical  
editor

project  
structure

component  
structure



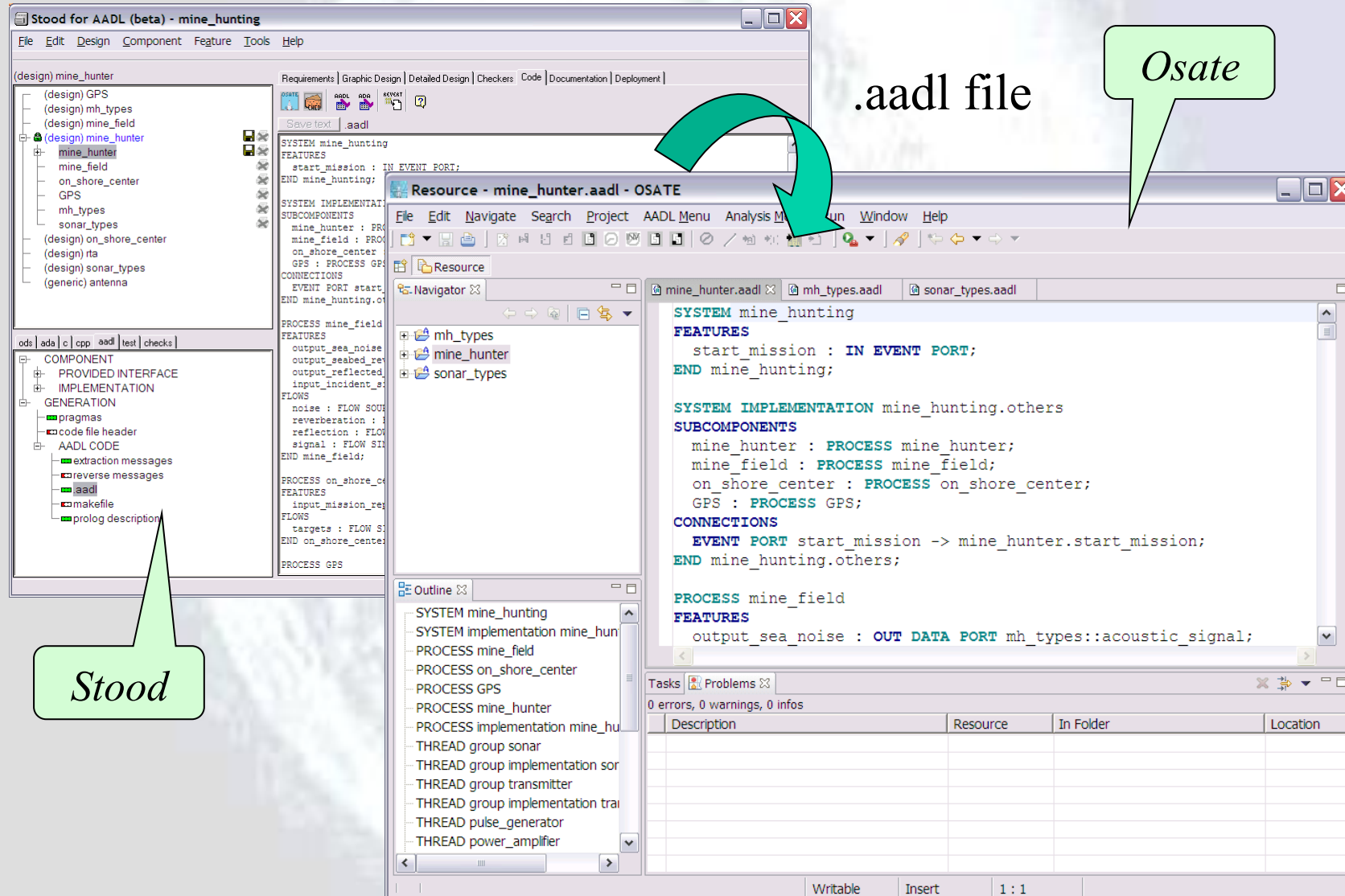
# Stood <sup>LMP</sup> plugins

The screenshot displays the Stood for AADL (beta) - cotre application interface. The main window is divided into several panes:

- Left Pane (Design Tree):** Shows a hierarchical view of the design. The tree includes nodes for `(design) call_subprogram`, `call_subprogram`, `print_service`, `printer`, `COTRE`, `(design) COTRE`, `(design) COTRE_patterns`, and `(design) testModes`. A green callout labeled **model** points to this pane.
- Top Right Pane (Text Editor):** Displays a rule definition for L087. The rule text is: `/* L087 When a server subprogram declaration appears in a thread component type, the scheduling protocol property value for all thread implementations, subcomponents, and instances having that component type must be Aperiodic or Sporadic. */`. Below the comment, the rule is defined as `errL087(X) :- hrtPeriod(X, _), isConstrained(_, 'OPERATION', X, 'HSER', 'NIL').`. A green callout labeled **rules** points to this pane.
- Bottom Left Pane (Design Verification):** Shows a list of verification checks. The `Legality [rule]` check is highlighted. A green callout labeled **result** points to the bottom right pane.
- Bottom Right Pane (Results):** Displays the output of the verification process. It shows an error message: `ERROR : The thread print_service should not be periodic (L087)`.

A large yellow callout box in the center of the interface contains the text: *Stood design checker has been qualified as a verification tool for the A380 DO-178B certification*.

# Static analysis



The screenshot displays the Stood for AADL (beta) - mine\_hunting application. The interface includes a menu bar (File, Edit, Design, Component, Feature, Tools, Help), a toolbar, and several panes:

- Left Pane (Project Structure):** Shows a tree view of the project components. A green callout bubble labeled "Stood" points to the "AADL CODE" section, which includes "extraction messages", "reverse messages", "aadl", "makefile", and "prolog description".
- Top Pane (Design View):** Displays the "SYSTEM mine\_hunting" with its features and implementation details. A green callout bubble labeled ".aadl file" points to the "Save text" button in the toolbar.
- Right Pane (Code Editor):** Shows the source code for "mine\_hunter.aadl - OSATE". The code includes system features, implementation, and connections. A green callout bubble labeled "Osate" points to the "Resource" menu item in the editor's menu bar.
- Bottom Pane (Task List):** Displays a table with columns: Description, Resource, In Folder, and Location. The table is currently empty.

The code in the editor is as follows:

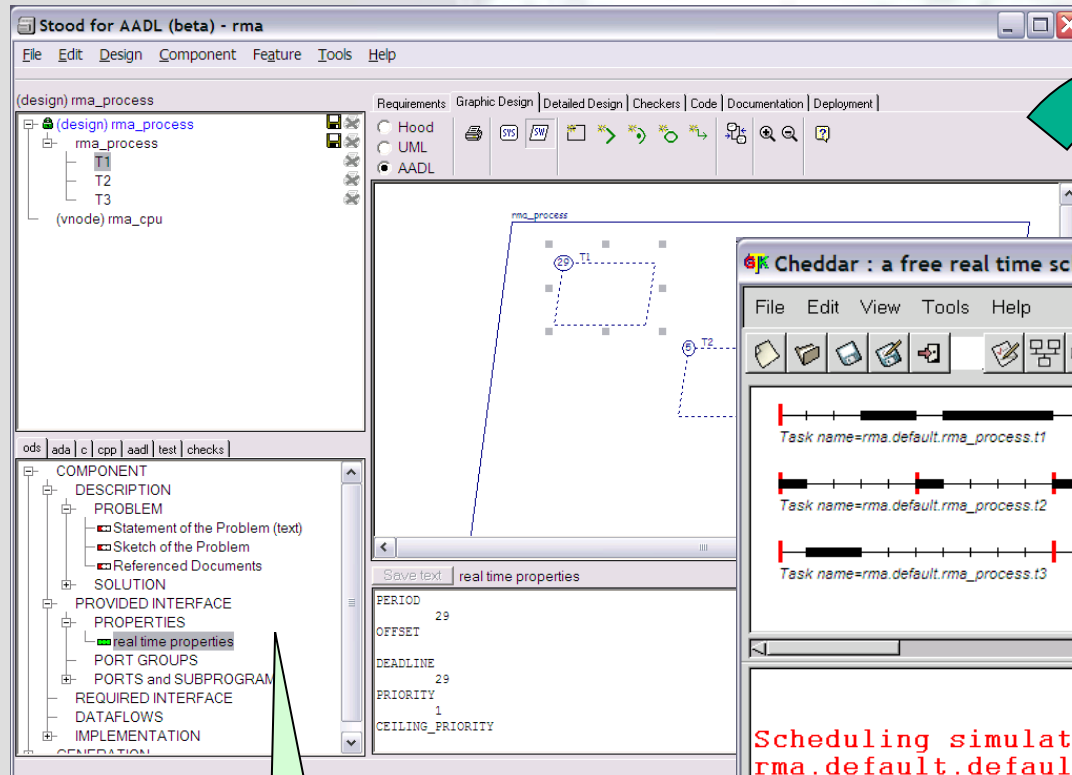
```

SYSTEM mine_hunting
FEATURES
  start_mission : IN EVENT PORT;
END mine_hunting;

SYSTEM IMPLEMENTATION mine_hunting.others
SUBCOMPONENTS
  mine_hunter : PROCESS mine_hunter;
  mine_field : PROCESS mine_field;
  on_shore_center : PROCESS on_shore_center;
  GPS : PROCESS GPS;
CONNECTIONS
  EVENT PORT start_mission -> mine_hunter.start_mission;
END mine_hunting.others;

PROCESS mine_field
FEATURES
  output_sea_noise : OUT DATA PORT mh_types::acoustic_signal;
  
```

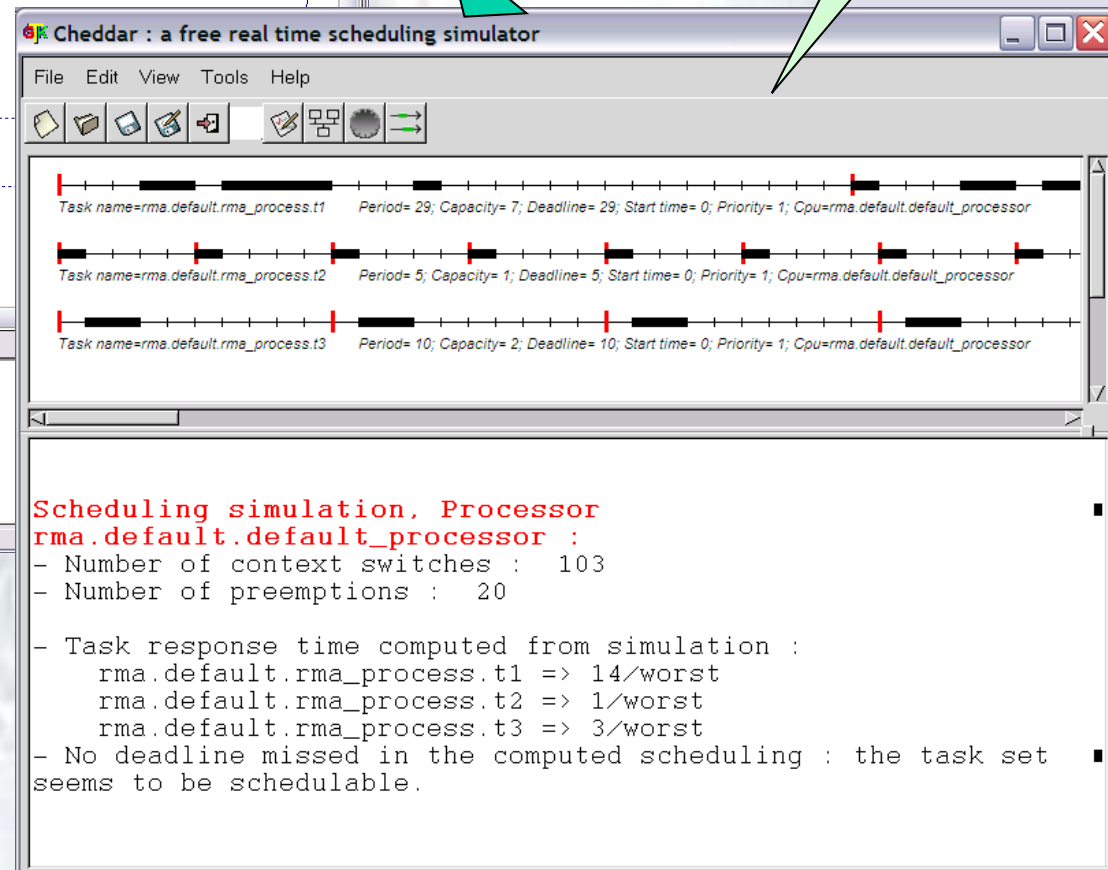
# Schedulability analysis



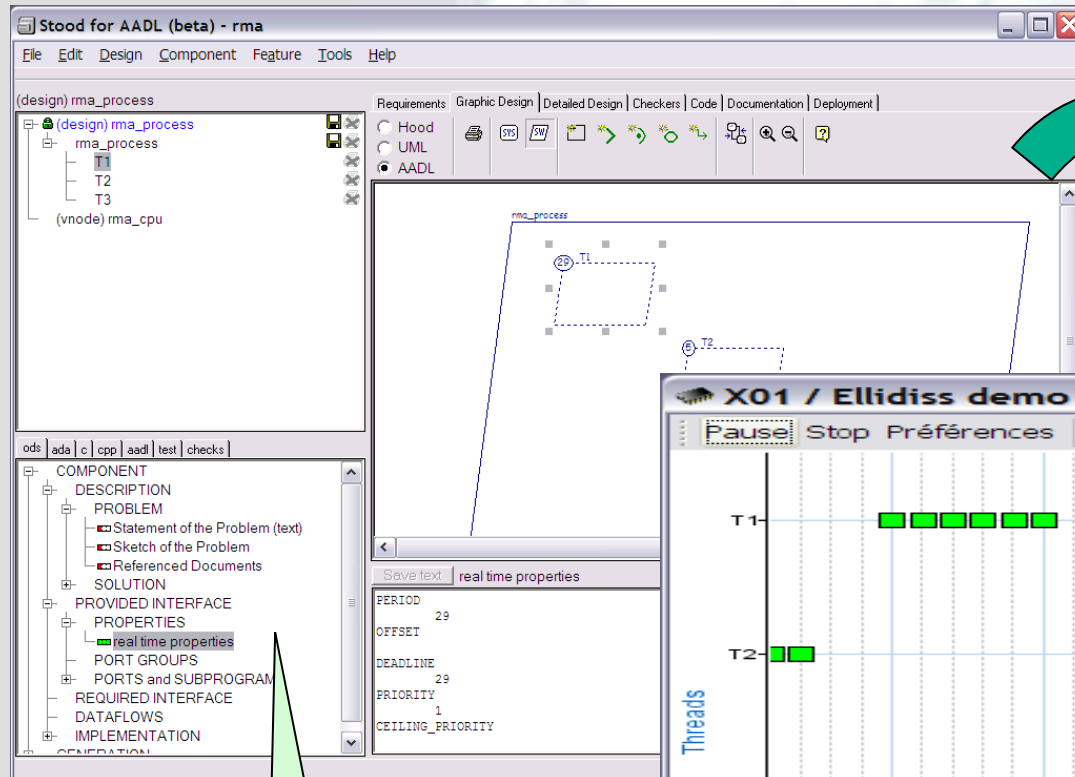
*Stood*

.aadl file

*Cheddar*



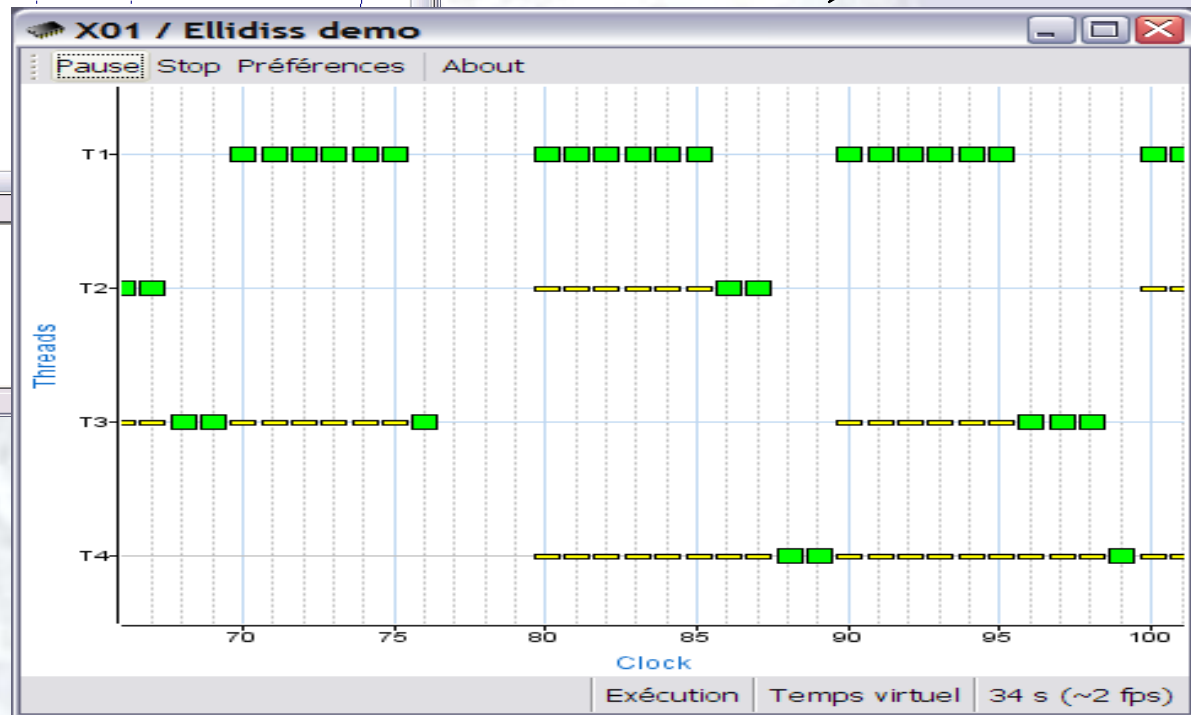
# Model Animation



.aadl file

*Multi-  
Agents  
Simulator*

*Stood*



# STOOD summary 1/2

## Support of the Software Design activities



### Architectural Design

- components based approach with black-box and white-box views
- UML 2.0 graphical notation
- AADL import/export
- support of HOOD and HRT-HOOD methodology
- built-in real-time model

### Verifications

- cross references table
- automatic calculation of the required interfaces
- automatic generation of call trees and dataflow graphs
- real-time schedulability analysis
- requirements traceability matrix
- design rules checker
- design metrics

### Detailed Design & Coding

- customizable structured detailed design framework
- incremental documentation
- incremental coding and round-trip engineering
- incremental requirements coverage
- legacy Ada and C code reverse engineering

# STOOD summary 2/2

## Workflow Integration

### Project management

- full Windows-Unix interoperability
- network distributed project bases
- integrated interface to remote Configuration Management Systems
- multi user management at system and subsystem level
- SIF and XML design model interchange

### Compliance to Standards

- DO-178B for embedded avionics
- ECSS-E40 for space systems
- EN-50128 for railways
- MIL-STD-498 for military



### Requirements traceability

- import of high level requirements
- incremental requirements coverage
- management of the derived requirements
- bidirectional interface with Reqtify™

### Code & Doc generators

- Ada95
- C/C++
- HTML
- PostScript/PDF
- RTF (Word™)
- MIF (FrameMaker™)

# Stood at Airbus

- **Major industrial return of experience:**
  - A340, A380, A400M and now A350
  - More than 40 licences
  - Fully integrated inside the Airbus Software development environment
  - Customized features:
    - code generators: C + Assembly
    - design checkers: DO-178B qualified for the A380
    - documentation generator



# Stood at Eurocopter

## - **Tiger Helicopter:** mission calculators

- French/German program
- Australian program
- Spanish program



# Stood at CNES (French Space Agency)

## - Jason satellite

3 Stood Projects:

Proteus: platform

Poseidon: altimetry

Doris: positioning

## - Helios 2 satellite

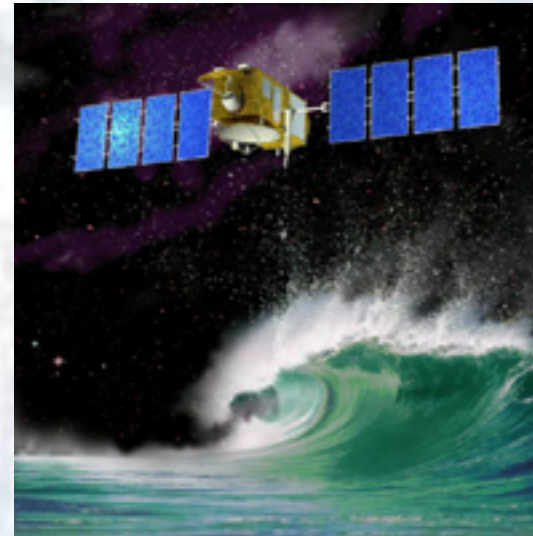
40 Stood Designs

900 000 Lines of Ada code

## - Spot 5 satellite

- Ariane 5 launcher telemetry

- Demeter micro-satellite



# Stood at ESA (European Space Agency)

- **Envisat satellite**
- **Metop satellite**
- **Galileo positioning system**



- **European Robotic Arm for the International Space Station**