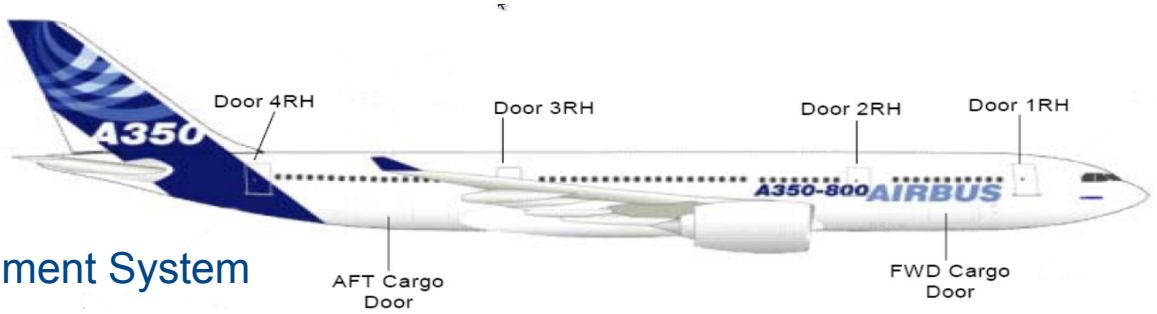
CESAR - Cost-**e**fficient methods and processes for **sa**fety **r**elevant embedded systems

System-level Co-simulation of Integrated Avionics Using Polychrony

Huafeng Yu

Espresso, INRIA Rennes - Bretagne Atlantique / IRISA



Tools demonstration
A350 Doors Management System

SYNCHRON
Dec. 1st, 2010

The CESAR project

Innovations in

- Requirements engineering
  - Formalization of multi viewpoint, multi criteria and multi level requirements
- Component-based engineering
  - Application on design space exploration

Reference Technology Platform (RTP)

- Integration or interoperation of existing or emerging technologies

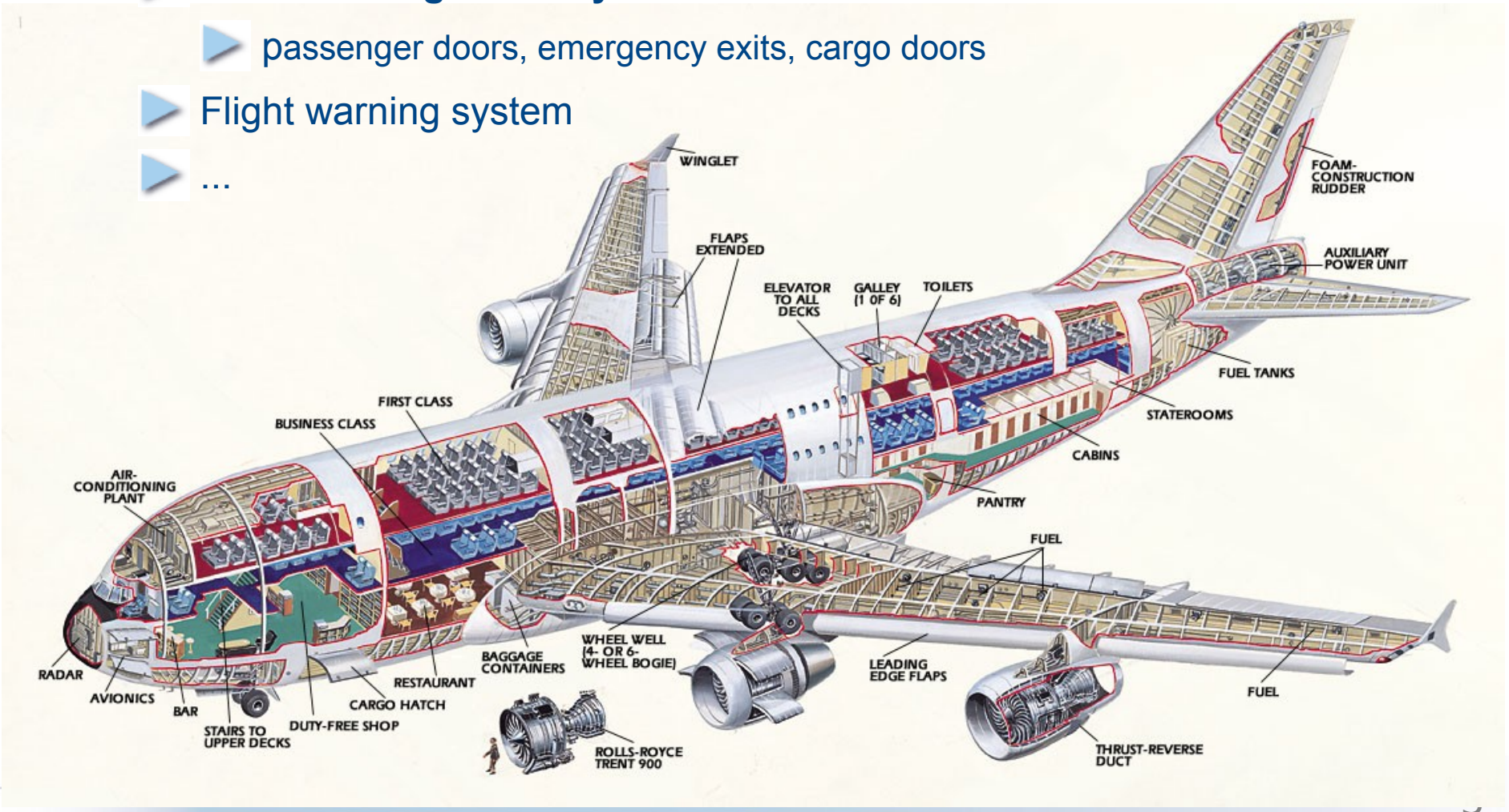
Tools demonstration

- Pilot applications: avionics, automotive, rail, ...
- Tools and tool chain demonstration
- Tools (technologies) integration via Eclipse-based ModelBus

A350 Doors Management System

Flight control systems

- Landing gear system
- **Doors management system**
  - passenger doors, emergency exits, cargo doors
- Flight warning system
- ...



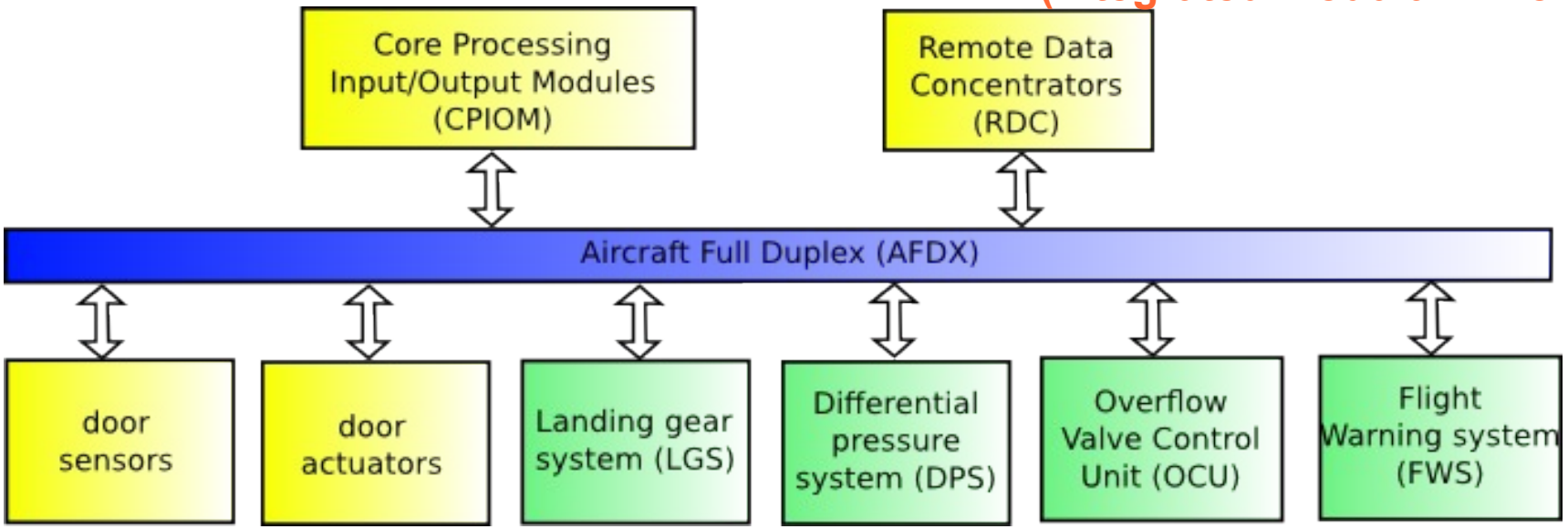
A350 Doors Management System

Simplified Doors and Slides Control System (SDSCS)

- Monitor doors status via door sensors
- Control flight lock actuators
- Manage the residual pressure
- Inhibit incorrect cabin pressurization

Simplified Doors and Slides Control System

IMA (Integrated Modular Avionics)



A safety-critical system

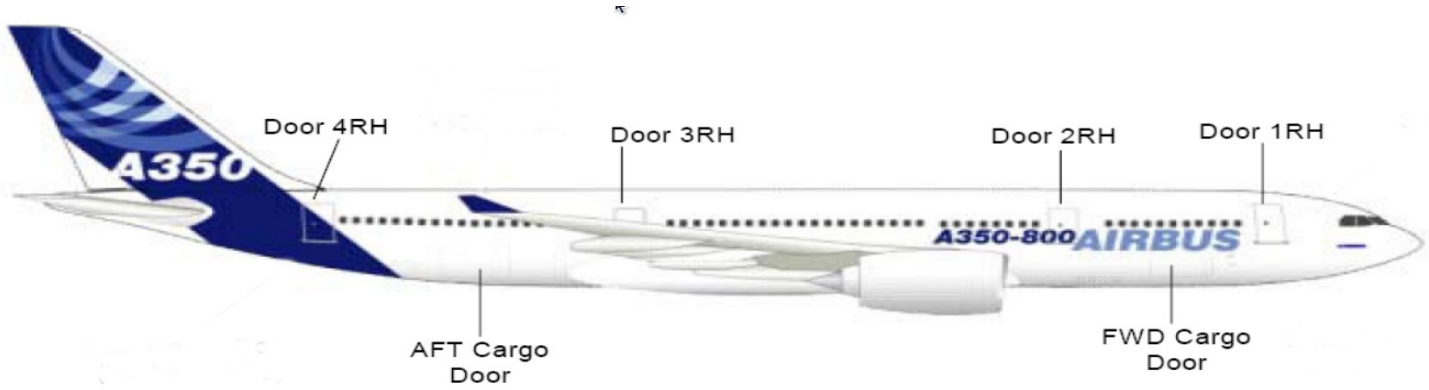
- High-level modeling
- Early phase validation & verification
- Architecture exploration

Outline

- The Polychrony approach for CESAR
- Architectural modeling based on AADL
- Functional modeling based on Simulink/Gene-Auto
- Additional models and system integration
- VCD-based simulation and profiling
- Distribution and scheduling via Syndex

AADL: Architecture Analysis & Design Language
VCD: Value Change Dump

The Polychrony approach for CESAR

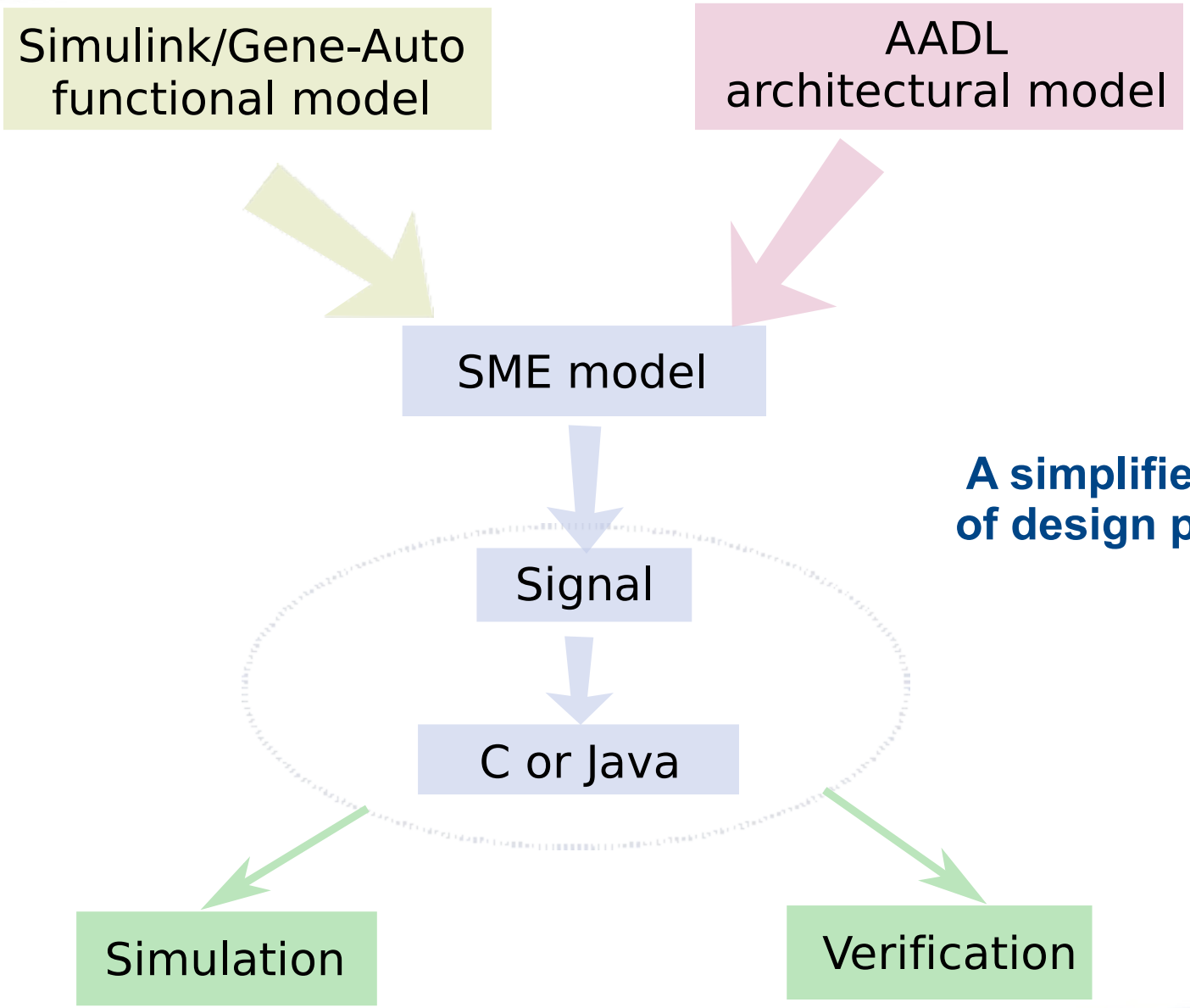


Polychrony for CESAR

- Timing analysis
- Formal verification, simulation, synthesis, etc.
- (Partial) specifications
  - Incomplete system description
  - Parallel development
- GALS design
- Eclipse Integration in the framework of MDE
  - SME/Polychrony
  - Tools connectivity

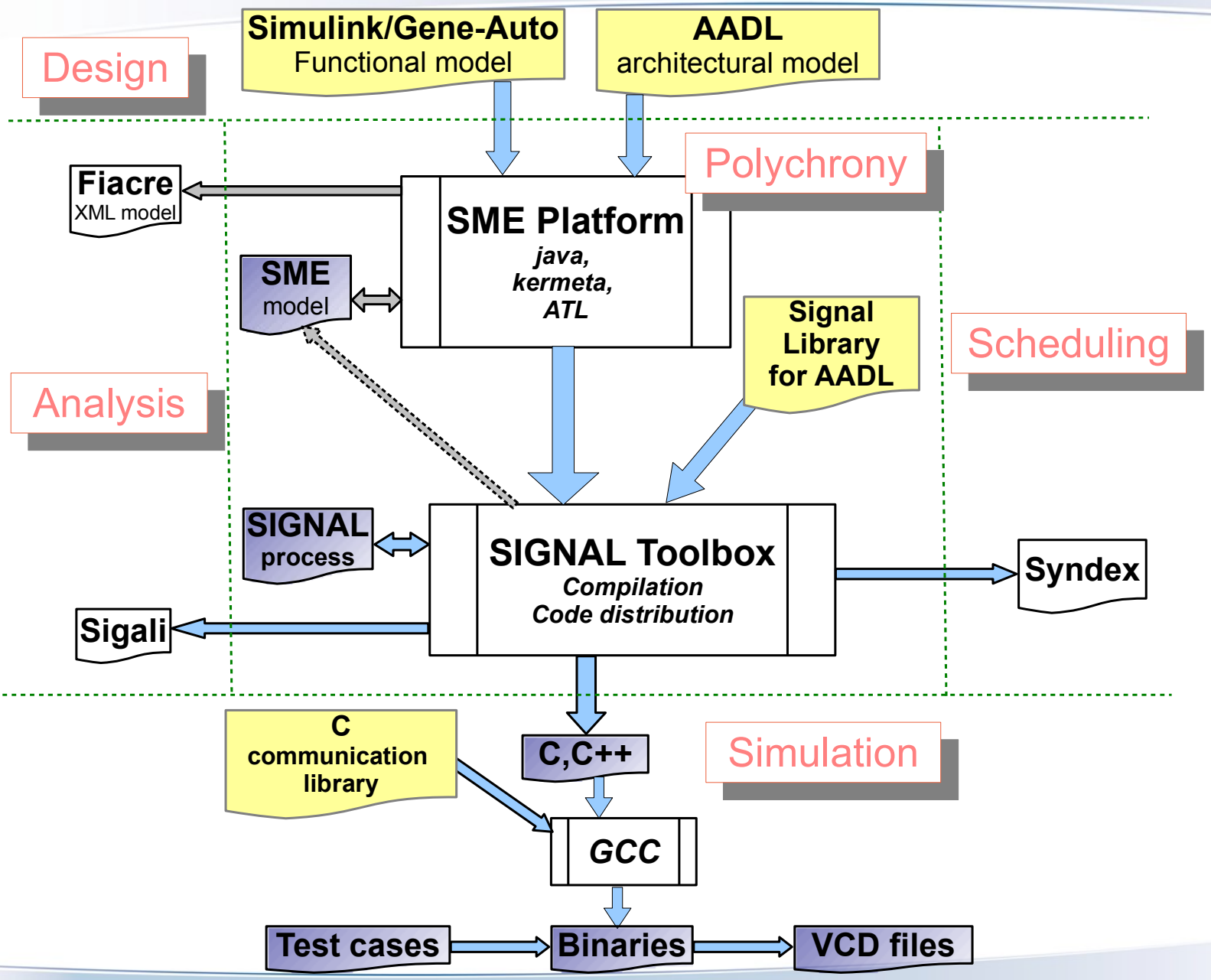
GALS: Globally Asynchronous Locally Synchronous
MDE: Model-Driven Engineering
SME: Signal Meta under Eclipse

The Polychrony approach



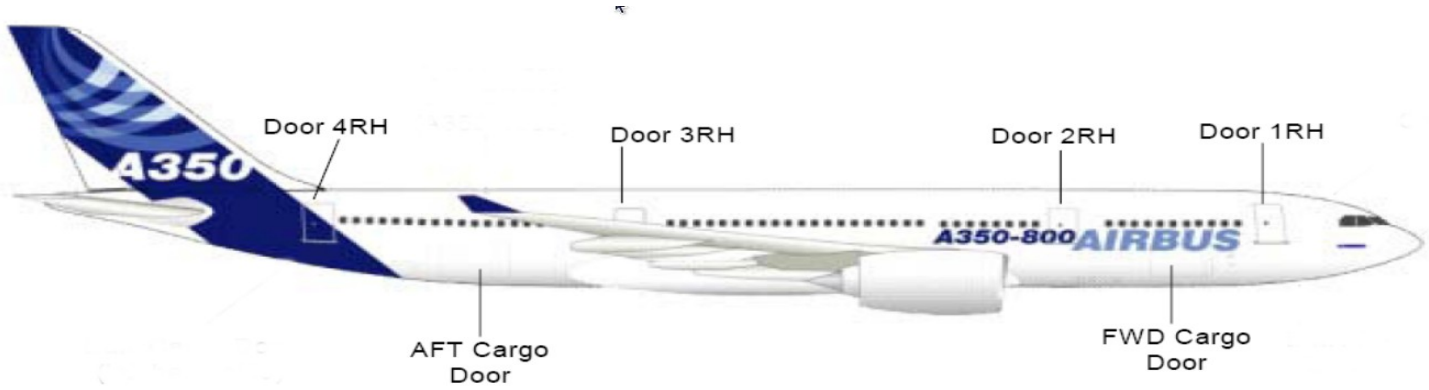
A simplified view of design process

The Polychrony approach



CESAR

Architectural modeling in AADL



CESAR

AADL (Architecture Analysis and Design Language)

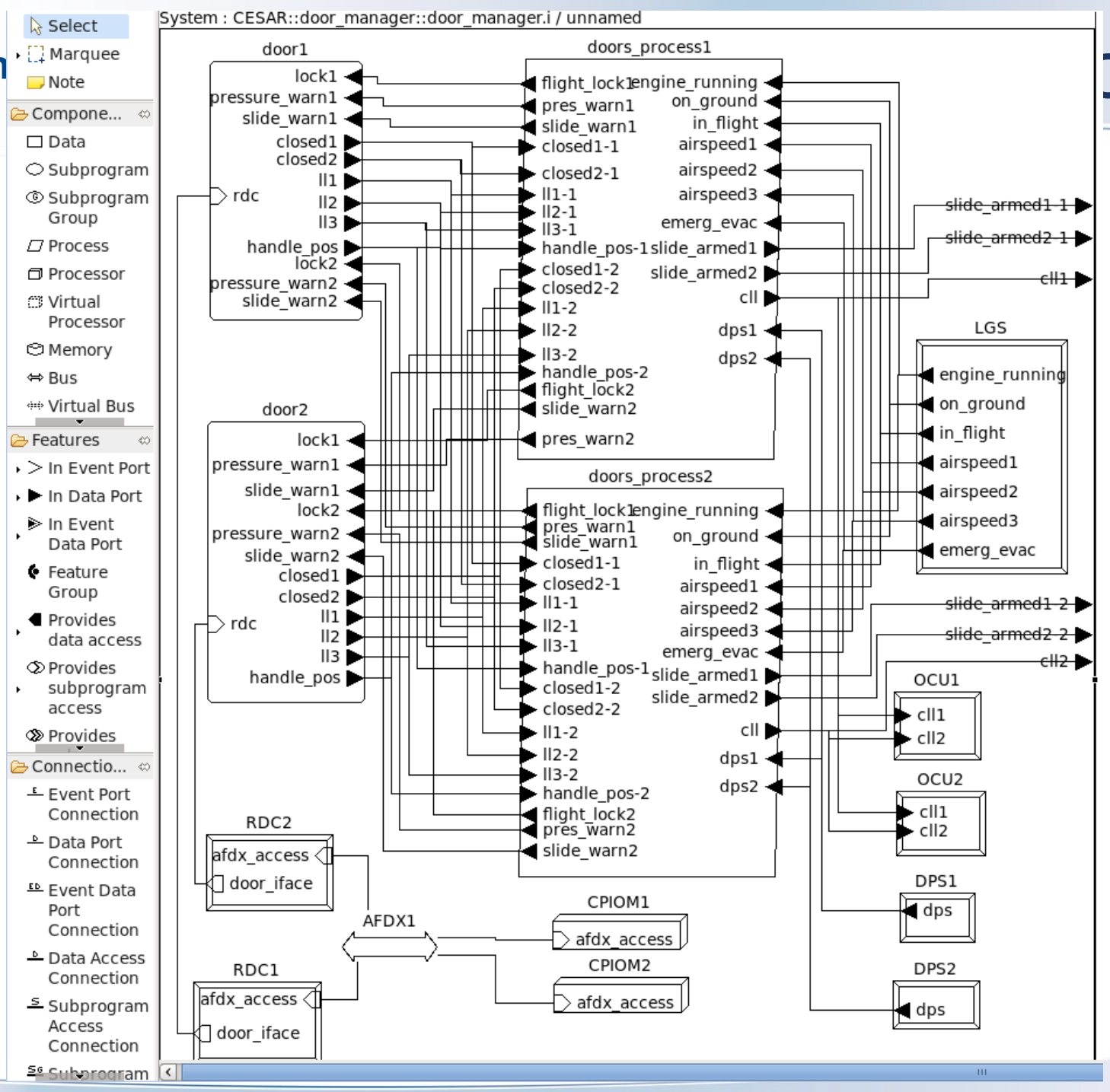
- SAE (Society of Automotive Engineers) standard
- High-level architecture design and evaluation for embedded systems
- Component-based paradigm
- AADL components
  - application software (*process, thread, thread group, subprogram, and data*)
  - execution platform (*processor, memory, device, and bus*)
  - composite (*system*, etc.)

ARINC 653 (Avionics Application Standard Software Interface)

- An API for software of avionics, following the IMA architecture
- APEX (APplication EXecutive) for space and time partitioning
- An ARINC partition is a logical allocation unit

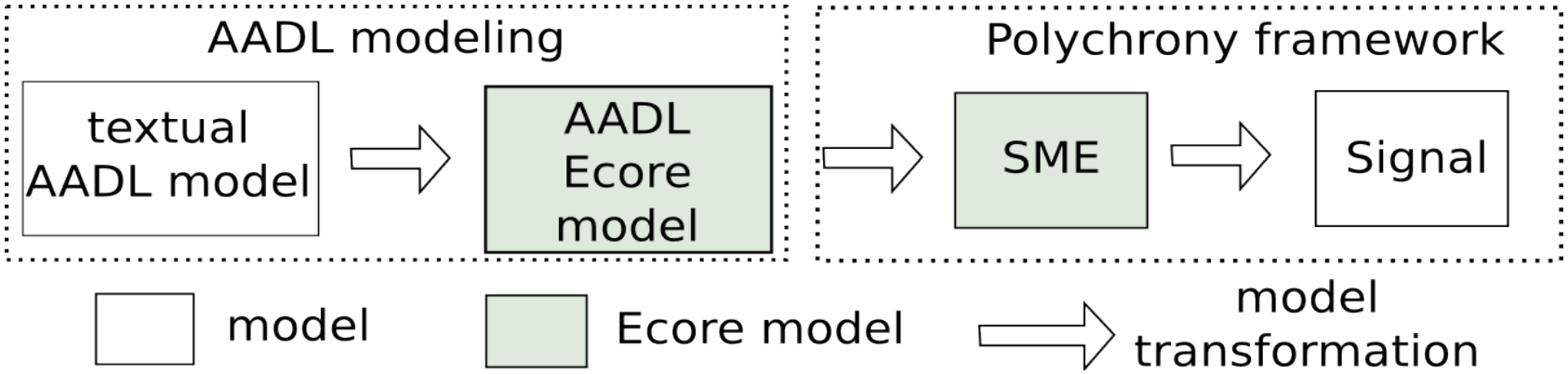
Arch

CESAR

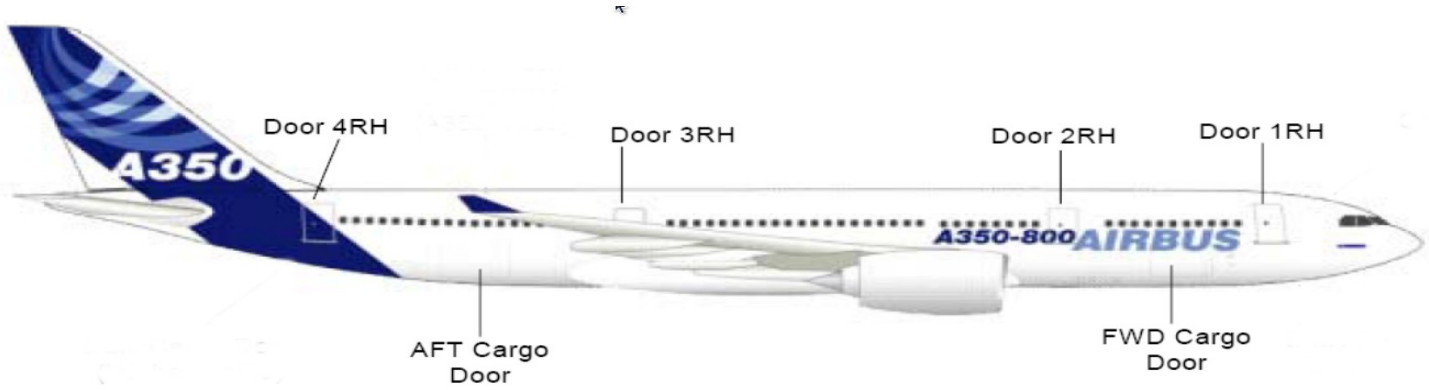

Select
Marquee
Note
Compone...
Data
Subprogram
Subprogram Group
Process
Processor
Virtual Processor
Memory
Bus
Virtual Bus
Features
In Event Port
In Data Port
In Event Data Port
Feature Group
Provides data access
Provides subprogram access
Provides
Connectio...
Event Port Connection
Data Port Connection
Event Data Port Connection
Data Access Connection
Subprogram Access Connection
Subprogram
System : CESAR::door_manager::door_manager.i / unnamed
door1
lock1
pressure_warn1
slide_warn1
closed1
closed2
ll1
ll2
ll3
rdc
handle_pos
lock2
pressure_warn2
slide_warn2
door2
lock1
pressure_warn1
slide_warn1
lock2
pressure_warn2
slide_warn2
closed1
closed2
ll1
ll2
ll3
rdc
handle_pos
doors_process1
flight_lock1
pres_warn1
slide_warn1
closed1-1
closed2-1
ll1-1
ll2-1
ll3-1
handle_pos-1
closed1-2
closed2-2
ll1-2
ll2-2
ll3-2
handle_pos-2
flight_lock2
slide_warn2
pres_warn2
engine_running
on_ground
in_flight
airspeed1
airspeed2
airspeed3
emerg_evac
slide_armed1
slide_armed2
cll
dps1
dps2
doors_process2
flight_lock1
pres_warn1
slide_warn1
closed1-1
closed2-1
ll1-1
ll2-1
ll3-1
handle_pos-1
closed1-2
closed2-2
ll1-2
ll2-2
ll3-2
handle_pos-2
flight_lock2
pres_warn2
slide_warn2
engine_running
on_ground
in_flight
airspeed1
airspeed2
airspeed3
emerg_evac
slide_armed1
slide_armed2
cll
dps1
dps2
LGS
engine_running
on_ground
in_flight
airspeed1
airspeed2
airspeed3
emerg_evac
slide_armed1_1
slide_armed2_1
cll1
slide_armed1_2
slide_armed2_2
cll2
OCU1
cll1
cll2
OCU2
cll1
cll2
DPS1
dps
DPS2
dps
RDC2
afdx_access
door_iface
RDC1
afdx_access
door_iface
AFDX1
CPIOM1
afdx_access
CPIOM2
afdx_access

A complete AADL transformation chain

- AADL textual model
- AADL Ecore model
- SME
- Signal
- C / Java



Functional modeling in Simulink/Gene-Auto



Functional modeling in Simulink/Gene-Auto

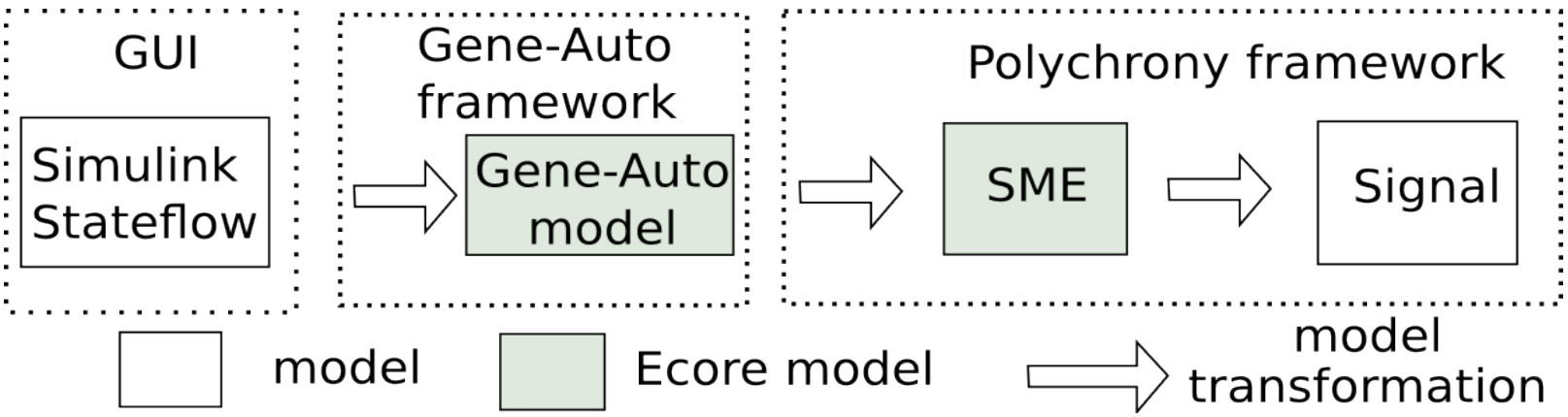
Simulink and Gene-Auto

- Matlab Simulink and Stateflow: wide-spread high-level modeling languages
- Gene-Auto: a safe subset of Simulink/Stateflow for ES design

Synchronous semantics of Gene-Auto

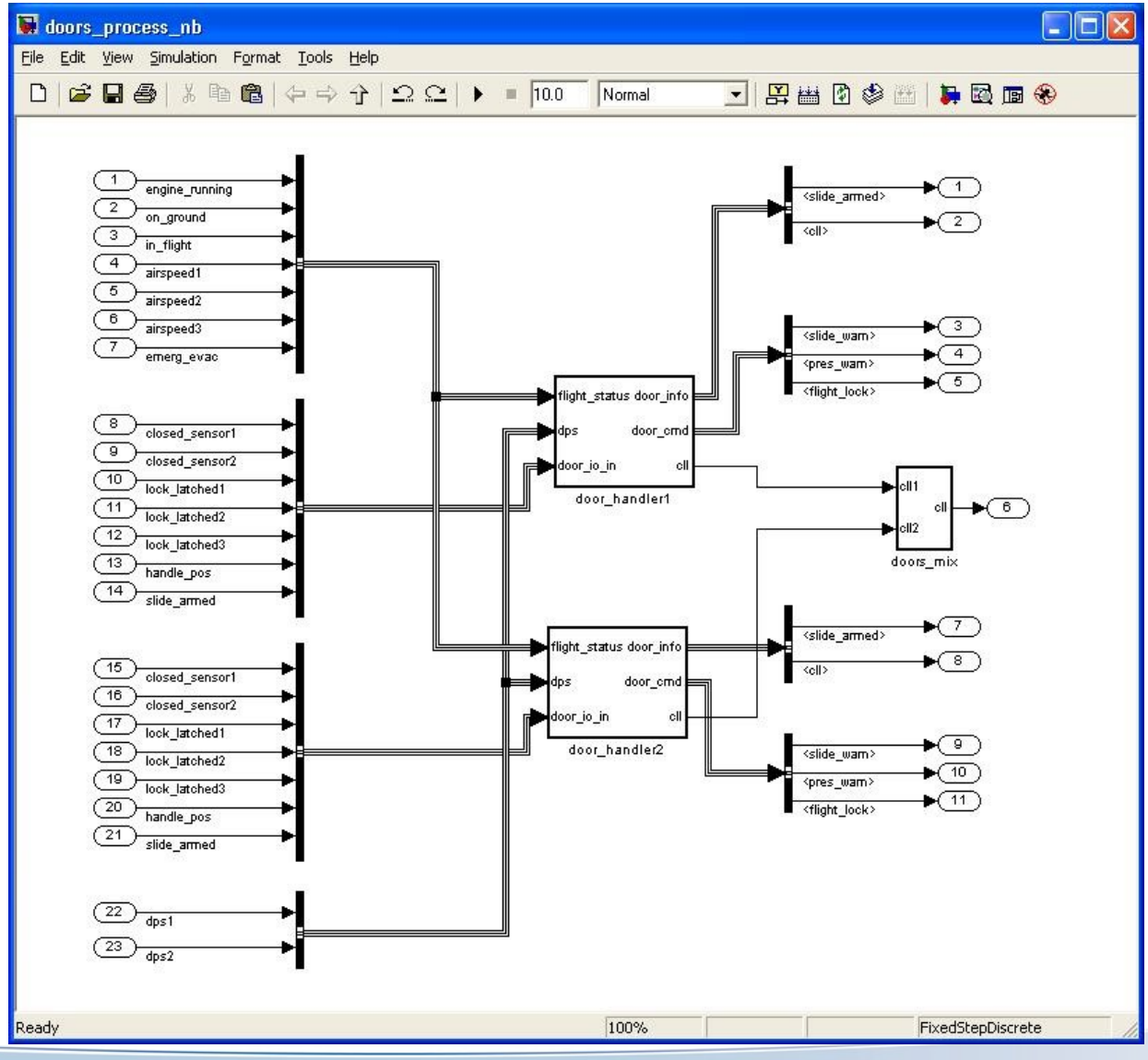
- Logical time
- Synchronized data-flow

A complete transformation chain

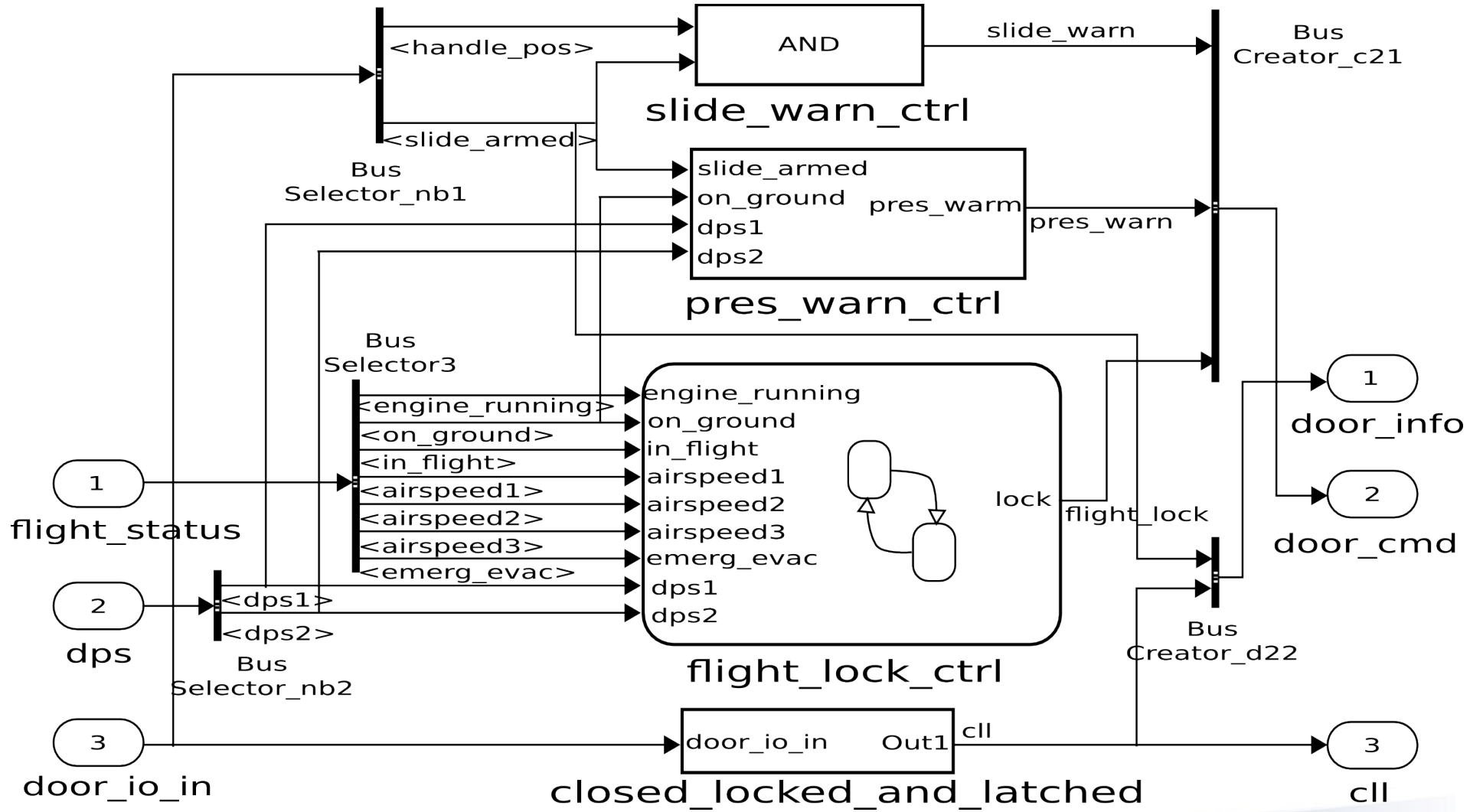


Functional modeling in Simulink/Gene-Auto

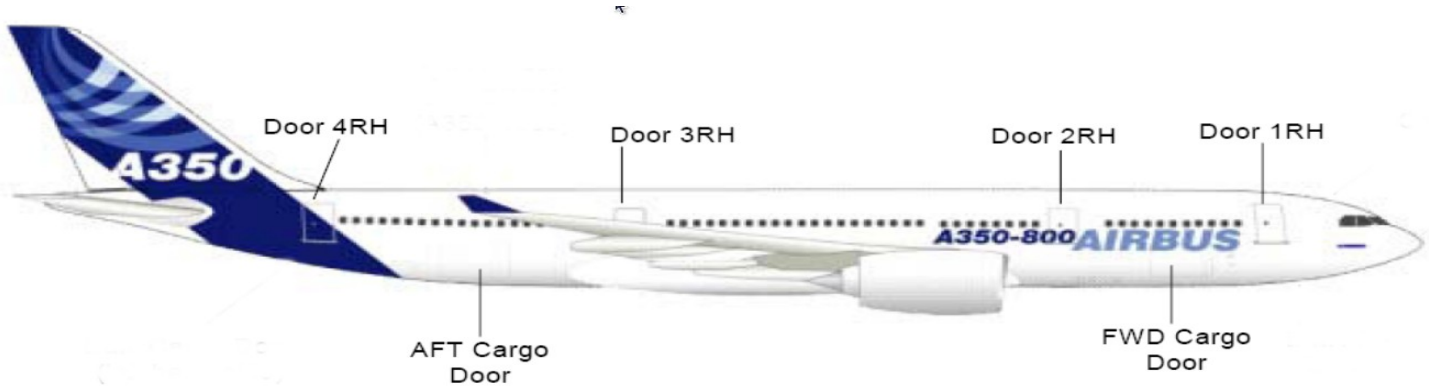
Simulink point of view of SDSCS



The door handler block



Additional models and system integration



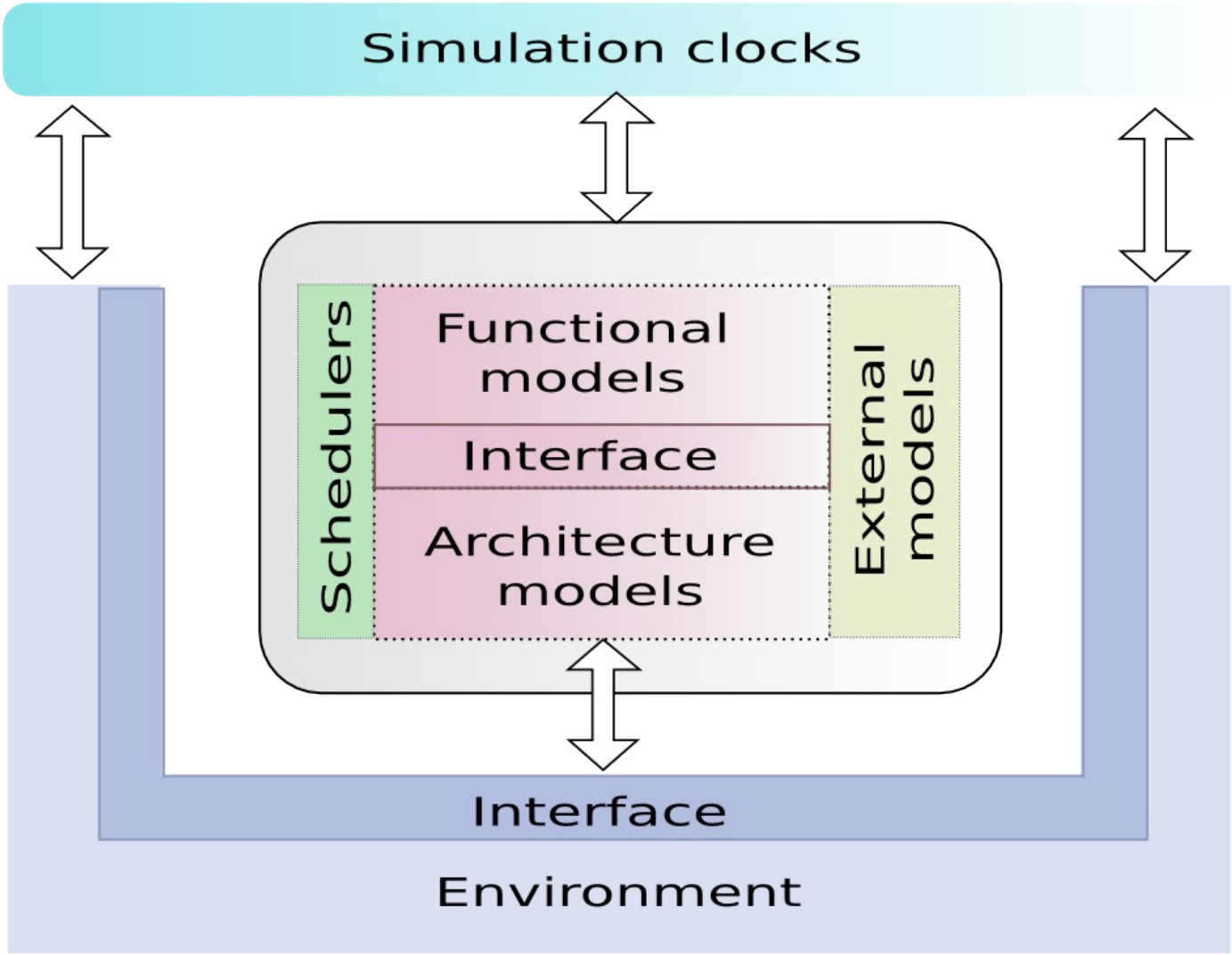
Additional models and system integration

Additional models for open system simulation

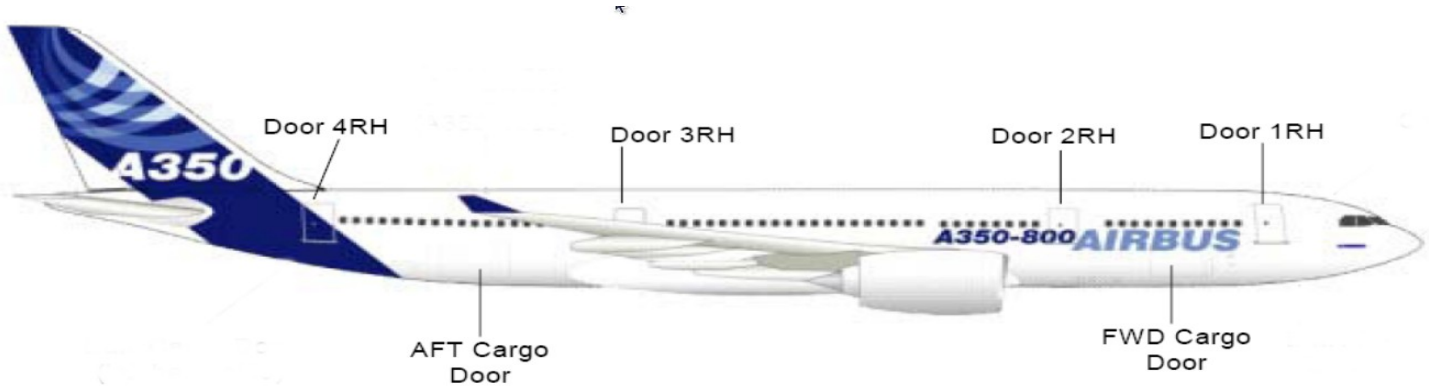
- Scheduler
  - A simple and static scheduler without preemption
  - Time interval is abstracted
- Simulation clocks
  - Reference clocks
  - Period clocks (for periodical threads)

Additional models for “almost” closed system simulation

- A simple environment model
  - Representation of the system outside SDSCS and pilot commands
- System inputs (pilot commands: *take_off, open_door, close_door, land*)

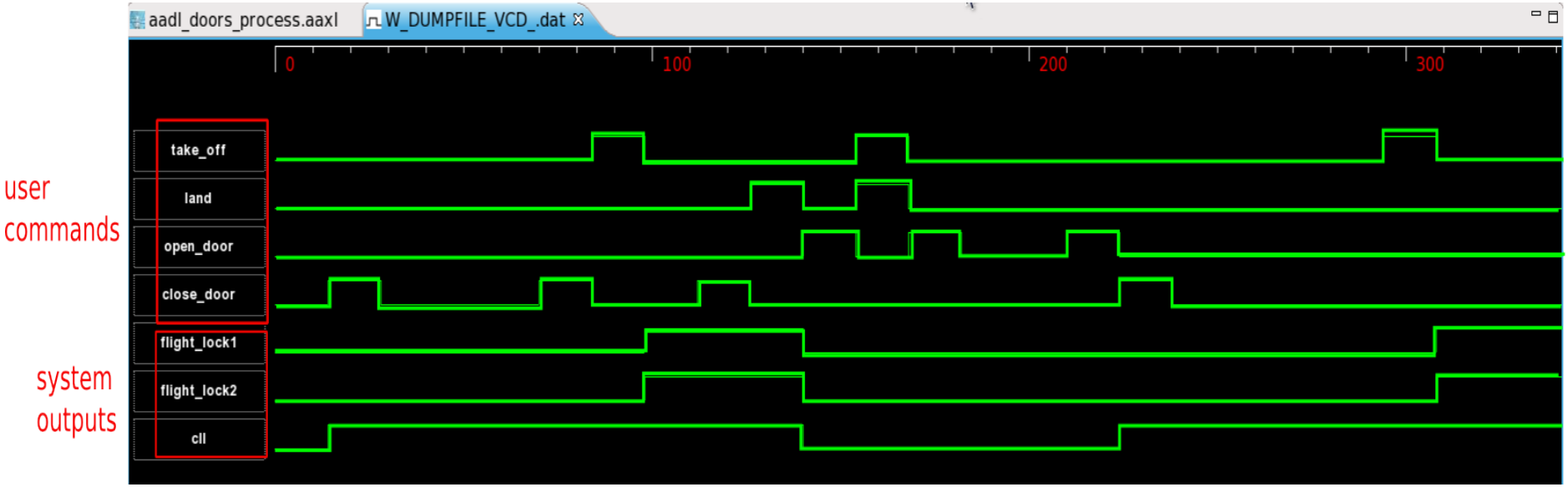

Simulation clocks
Schedulers
Functional models
Interface
Architecture models
External models
Interface
Environment

Simulation


Door 4RH
Door 3RH
Door 2RH
Door 1RH
A350
A350-800
AIRBUS
AFT Cargo Door
FWD Cargo Door

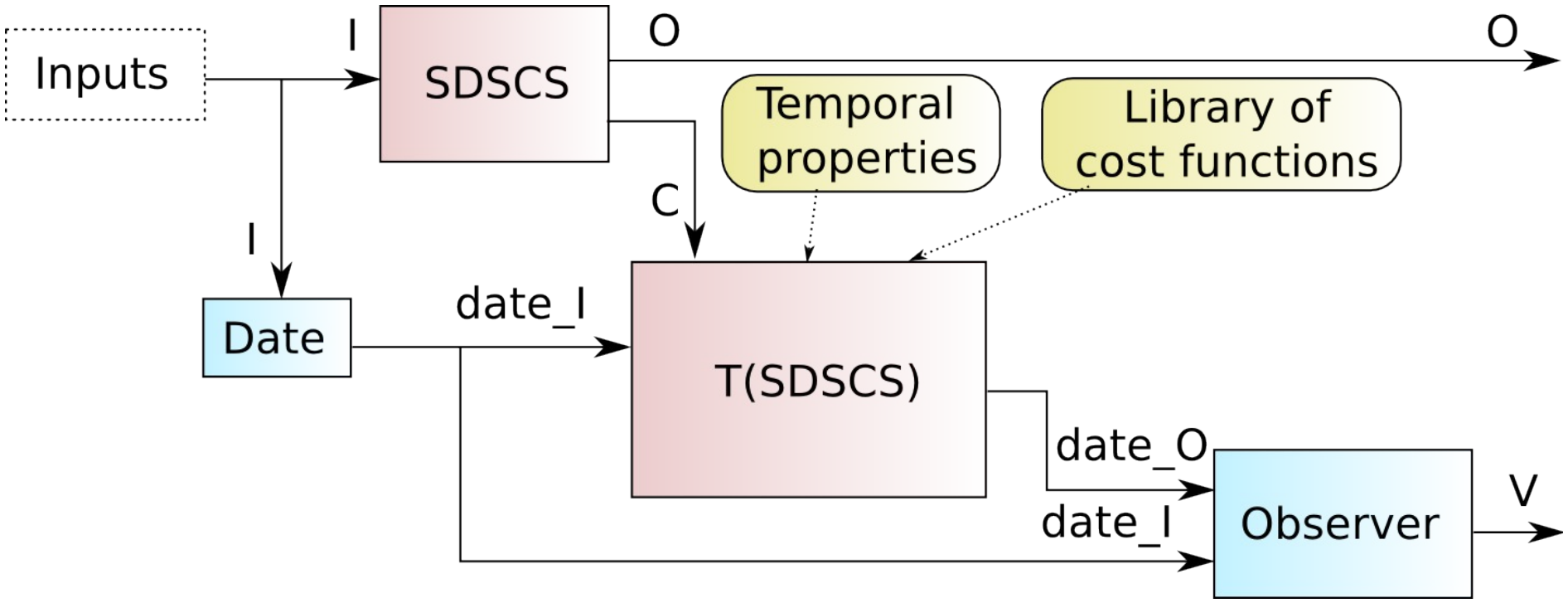
VCD visualization

- Traces (changed values) recorded in VCD format
- Global synchronization clock
- Interactive or non-interactive mode

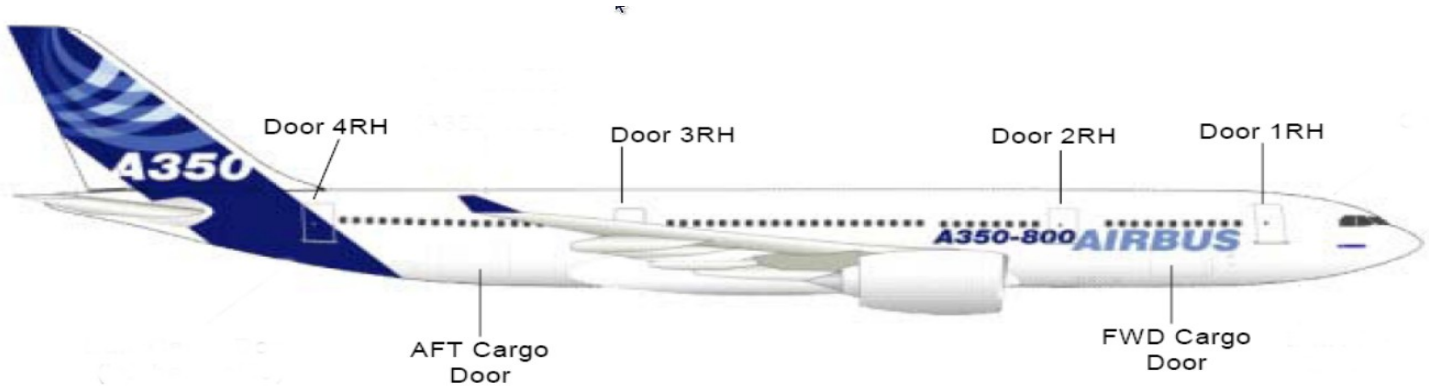


Profiling

- Temporal properties
- Temporal homomophism
- Co-simulation



Distribution and scheduling via Syndex



Distribution and scheduling via Syndex

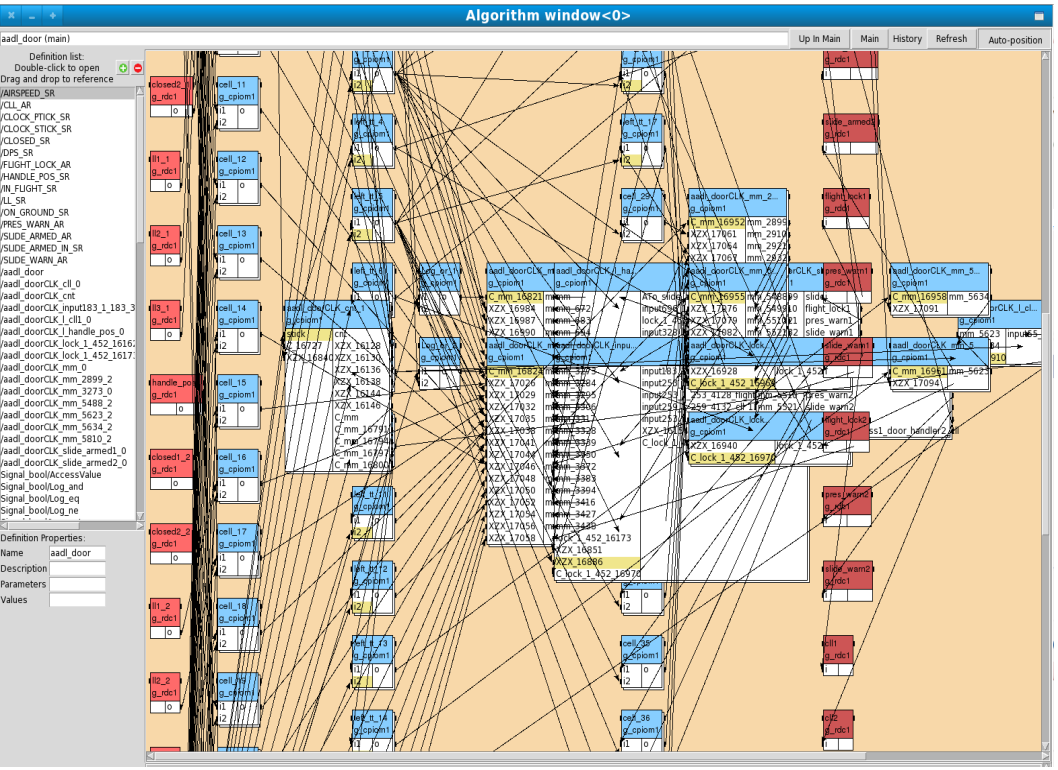
Syndex

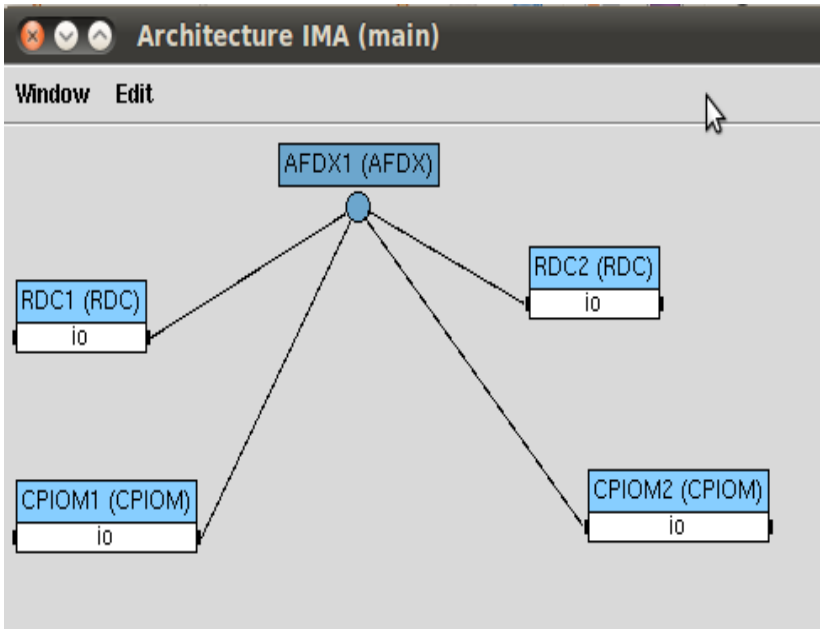
- Algorithm, architecture, and adequation
- Heuristic algorithm for adequation
  - Automatic code distribution
  - Processor-level scheduling and communication
  - Synchronization, ...

Signal to Syndex

- Endochronous programs transformation
- Algorithm is translated from Signal programs automatically
- Architecture is translated from AADL manually
- Constraints are added for specific binding between software and hardware

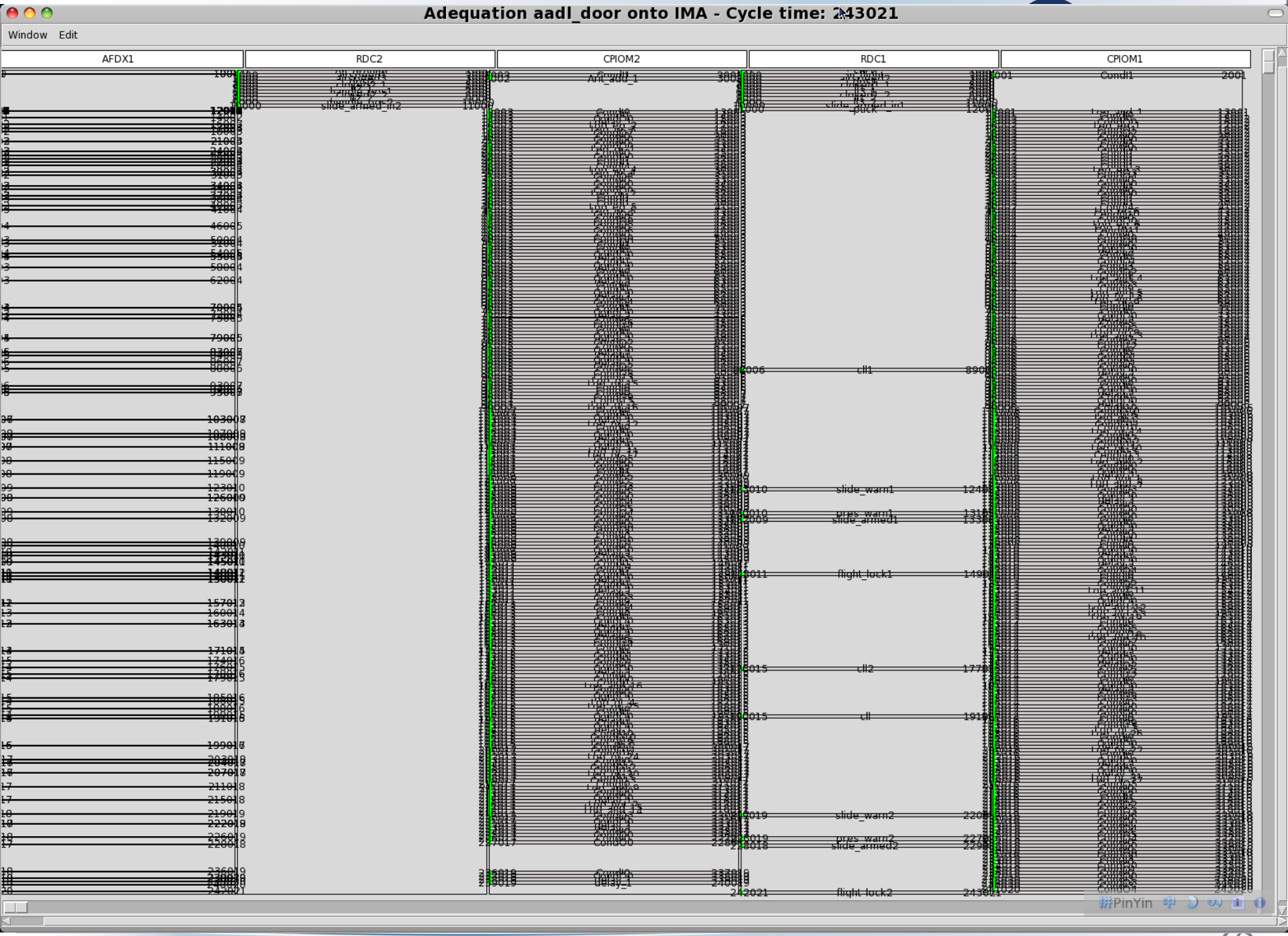
Distribution and scheduling via Syndex



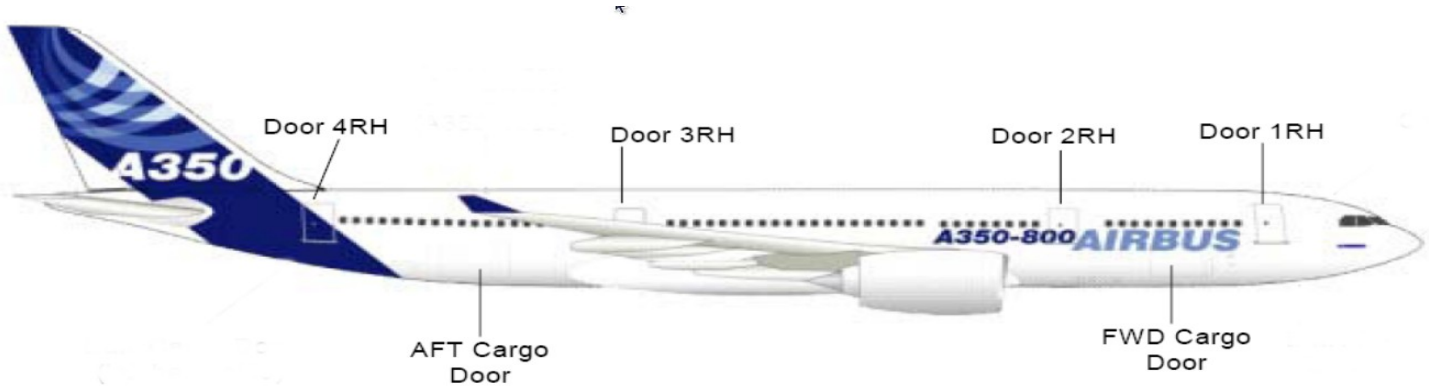


Algorithm

Architecture

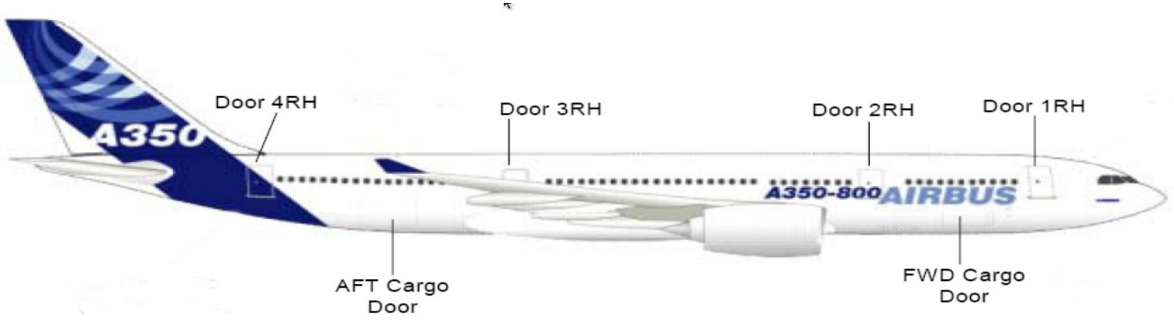

Adequation aadl_door onto IMA - Cycle time: 243021
Window
Edit
AFDX1
RDC2
CPIOM2
RDC1
CPIOM1
slide_armed_in2
Cond1
2001
cll1
8900
slide_warn1
pres_warn1
slide_armed1
flight_lock1
cll2
cll
slide_warn2
pres_warn2
slide_armed2
flight_lock2
243021

Conclusion and perspective



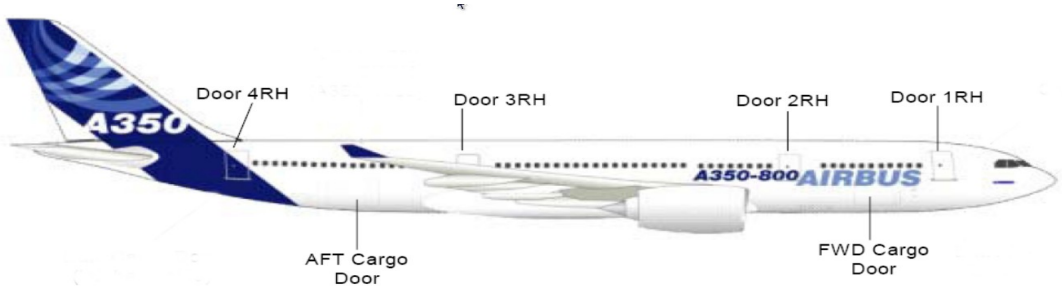
Conclusion

- High-level functional and architectural design
  - High-level modeling with AADL and Simulink/Gene-Auto
- Polychrony as a common development platform
  - Formal polychronous model
  - Automatic model transformations
  - Good interoperability between tools
- Early phase co-simulation
  - Demonstration by VCD viewers
  - Profiling
  - Syndex



Perspective

- More simulation with timing analysis
  - Sophisticated schedulers, such as Syndex and OS-level scheduler
  - Clock constraints in MARTE/CCSL
  - RT-Builder
- Architecture exploration
  - Performance, energy, flexibility, etc.
- Formal verification, synthesis, fault modeling and analysis
  - Sigali, Fiacre, Altarica, etc.
- Automatic test case generation
  - GATeL, TGV, etc.



Thank you for your attention