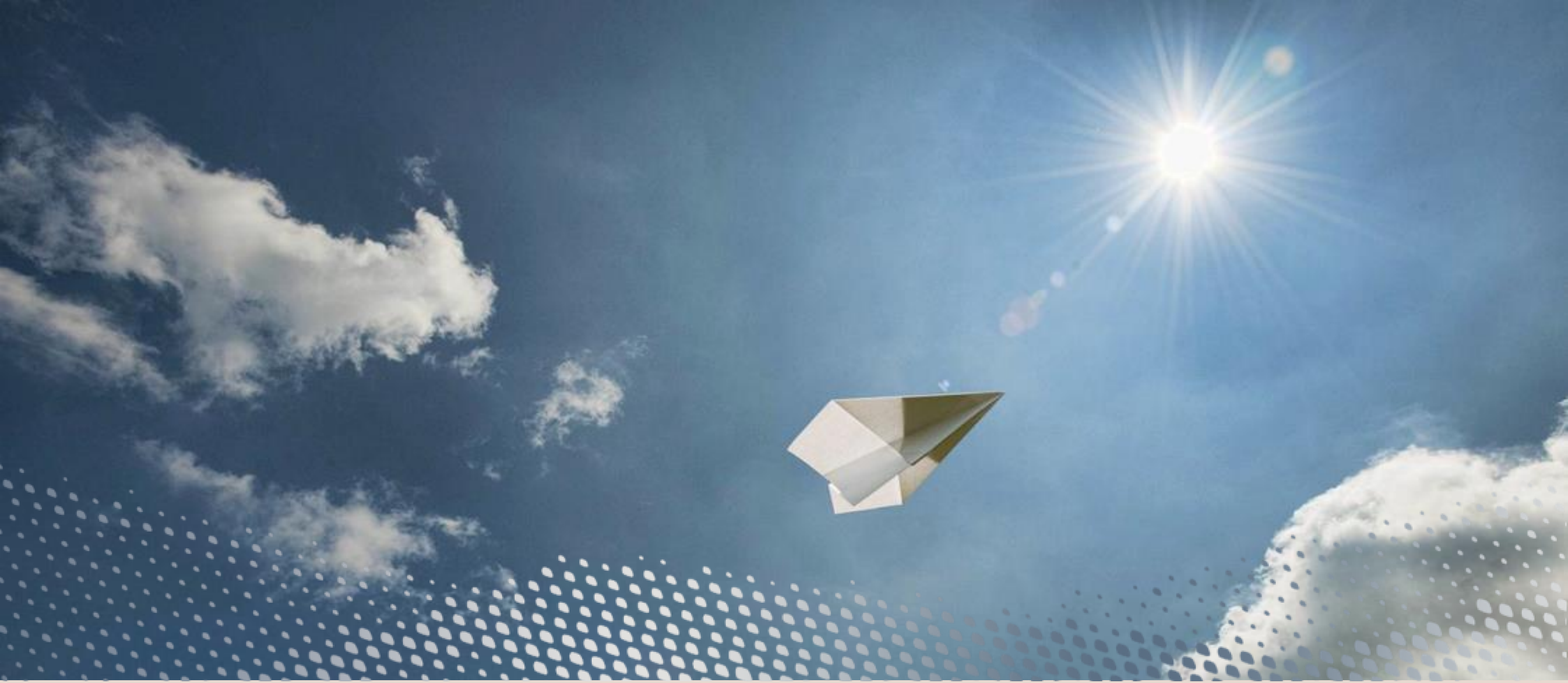


Research challenges and opportunities in the scope of complex systems development

Keynote at SoftNet 2014, Nice, France

Philipp Helle, Airbus Group Innovations

Content

1. Introduction

2. Motivation

3. Typical systems development process

4. Research challenges and opportunities

   - Formal requirements
   - Model-based specification
   - Architecture optimization
   - Formal verification
   - Virtual verification
   - Model-based testing

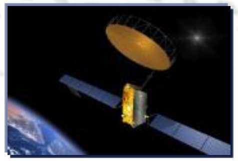
5. Conclusion

AIRBUS GROUP …at a glance

AIRBUS

AIRBUS HELICOPTERS

AIRBUS DEFENCE & SPACE



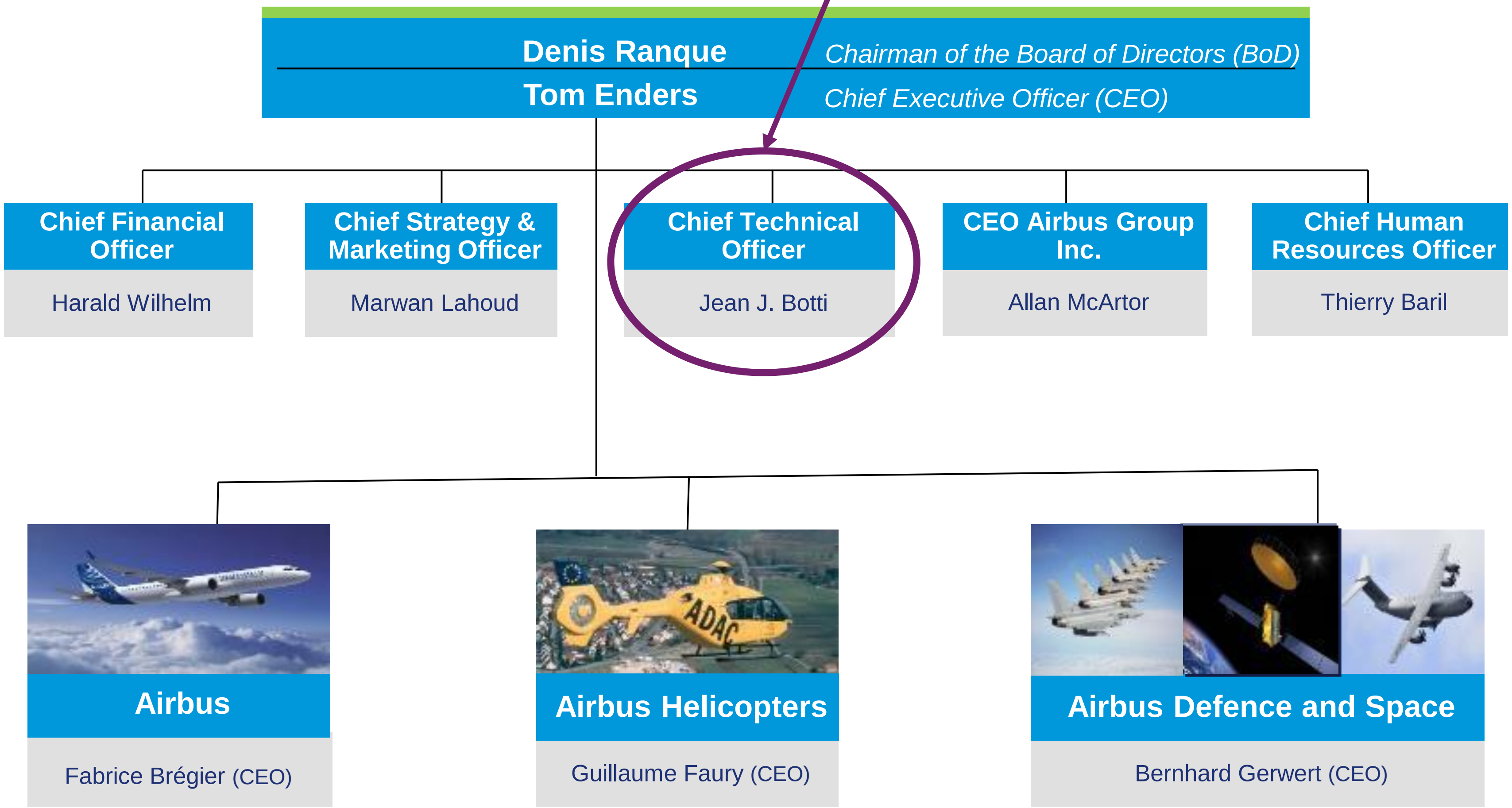
Key figures

- Over 800 Researchers, Scientists, Engineers worldwide
- 20 sites around the world
- Located in 12 countries
- More than 100 new patent applications every year

Airbus Group Innovations

Airbus Group Management structure

Airbus Group Innovations is part of Chief Technical Officer Organization



Denis Ranque *Chairman of the Board of Directors (BoD)*

Tom Enders *Chief Executive Officer (CEO)*

Chief Financial Officer
Harald Wilhelm

Chief Strategy & Marketing Officer
Marwan Lahoud

Chief Technical Officer
Jean J. Botti

CEO Airbus Group Inc.
Allan McArtor

Chief Human Resources Officer
Thierry Baril

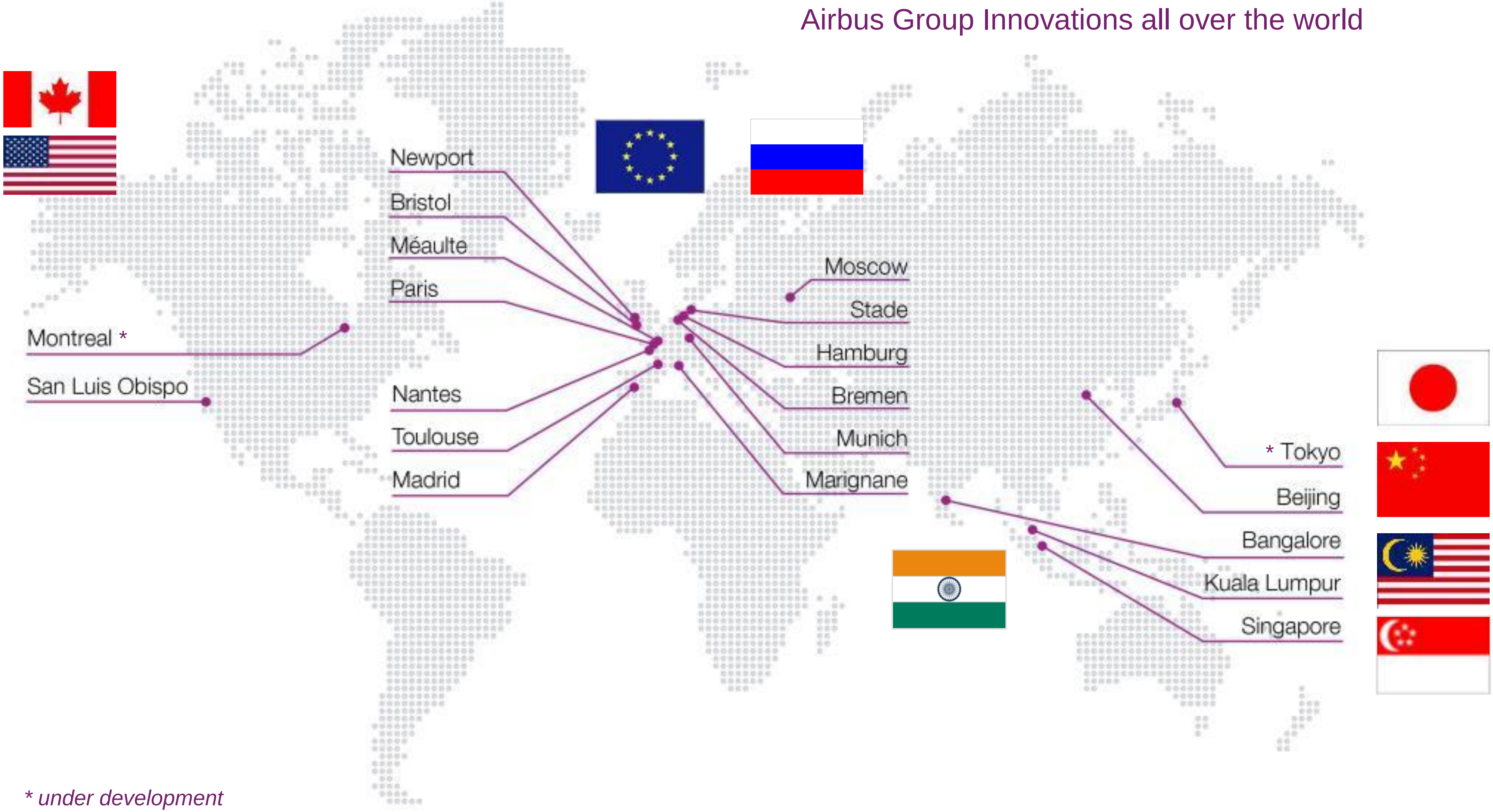
Airbus
Fabrice Brégier (CEO)

Airbus Helicopters
Guillaume Faury (CEO)

Airbus Defence and Space
Bernhard Gerwert (CEO)

Where we operate?

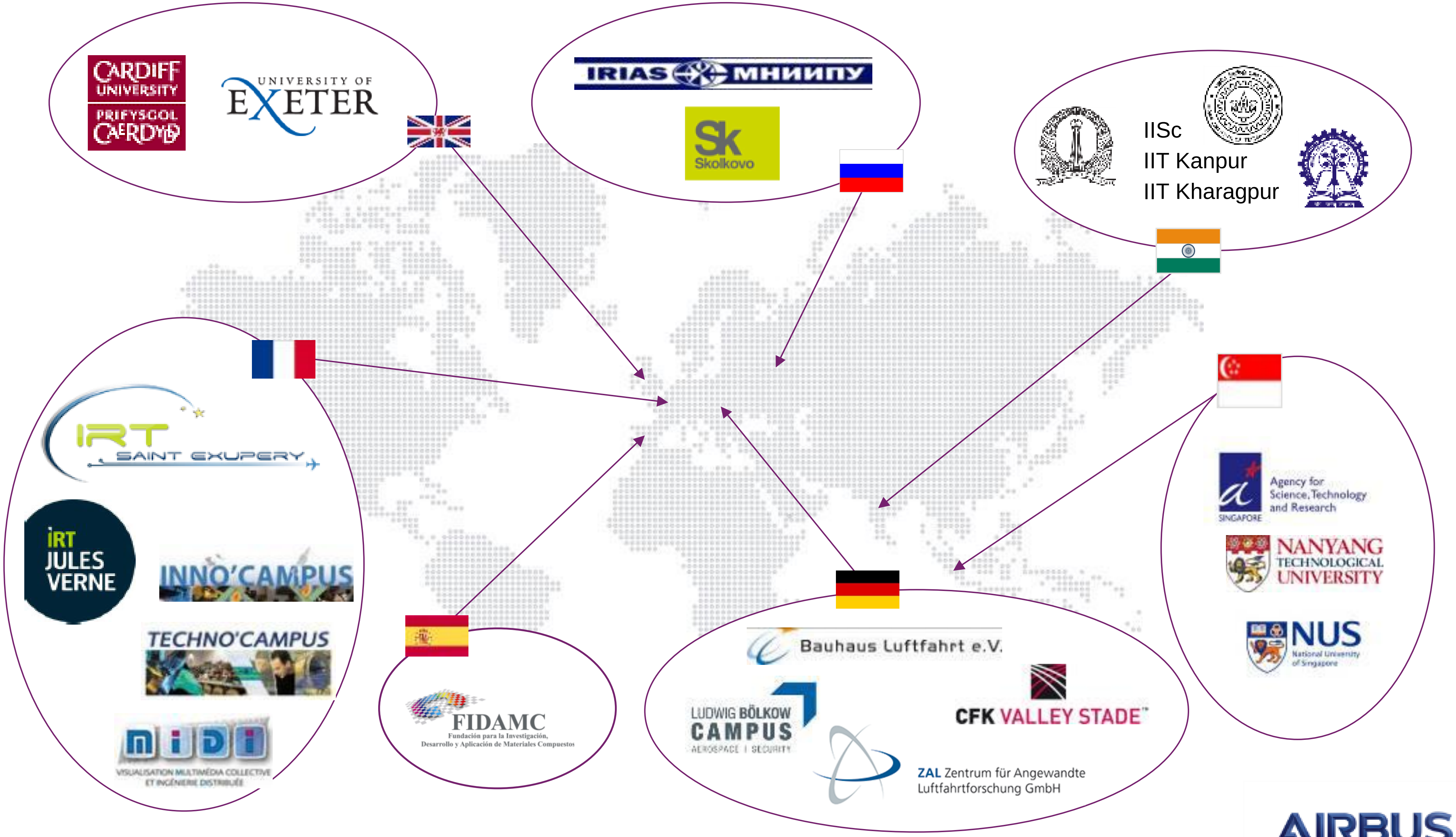
Airbus Group Innovations all over the world



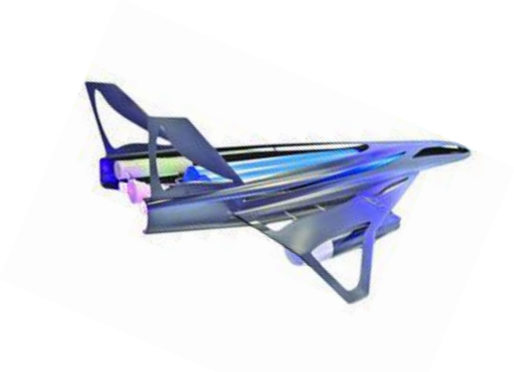
** under development*

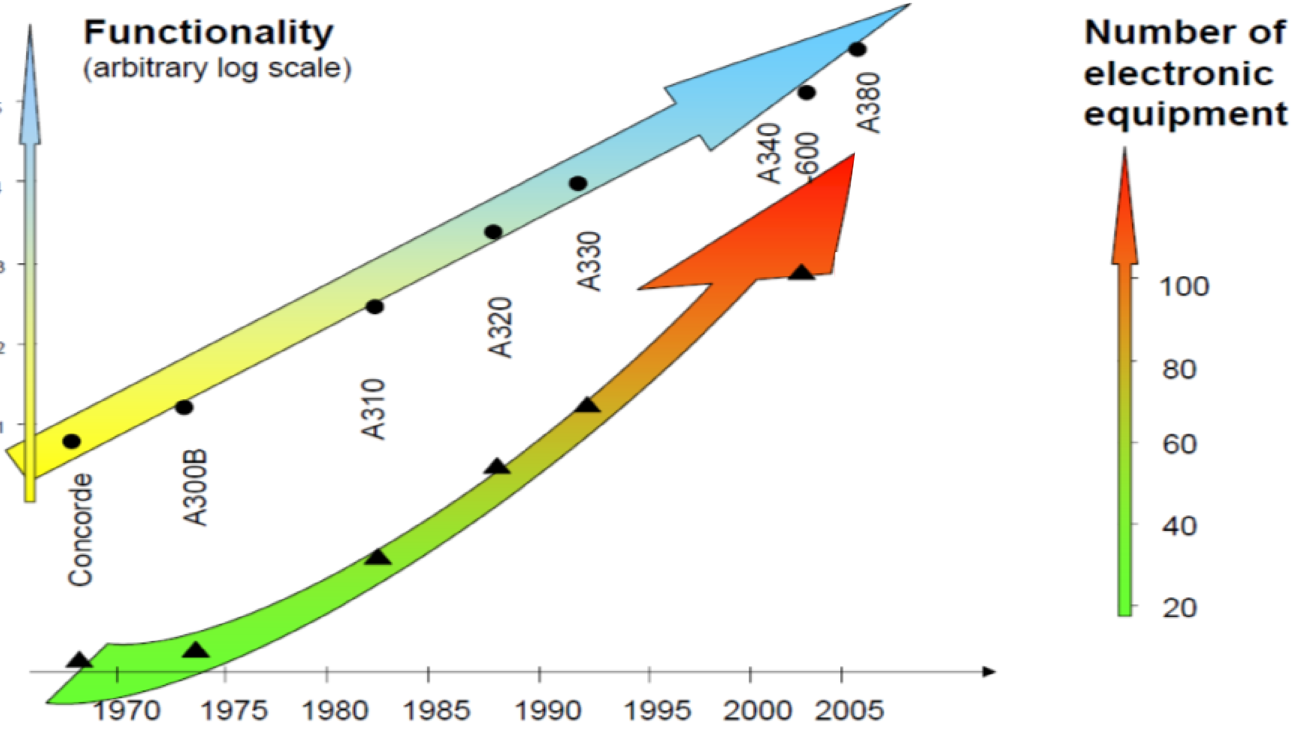
We optimize networking synergy to get innovations

○ Main partnerships


CARDIFF UNIVERSITY
PRIFYSGOL CAERDYDD
UNIVERSITY OF EXETER
IRIAS МНИИПУ
Sk Skolkovo
IISc
IIT Kanpur
IIT Kharagpur
IRT SAINT EXUPERY
IRT JULES VERNE
INNO'CAMPUS
TECHNO'CAMPUS
midi
VISUALISATION MULTIMÉDIA COLLECTIVE ET INGÉNIERIE DISTRIBUÉE
FIDAMC
Fundación para la Investigación, Desarrollo y Aplicación de Materiales Compuestos
Bauhaus Luftfahrt e.V.
LUDWIG BÖLKOW CAMPUS
AEROSPACE I SECURITY
CFK VALLEY STADE
ZAL Zentrum für Angewandte Luftfahrtforschung GmbH
Agency for Science, Technology and Research
SINGAPORE
NANYANG TECHNOLOGICAL UNIVERSITY
NUS National University of Singapore

Motivation Airbus Group





Trend: Growing Complexity of Systems

Motivation Airbus Group

How can we develop systems that have to fulfill a large number of often opposing requirements?

Required Functions addressed?

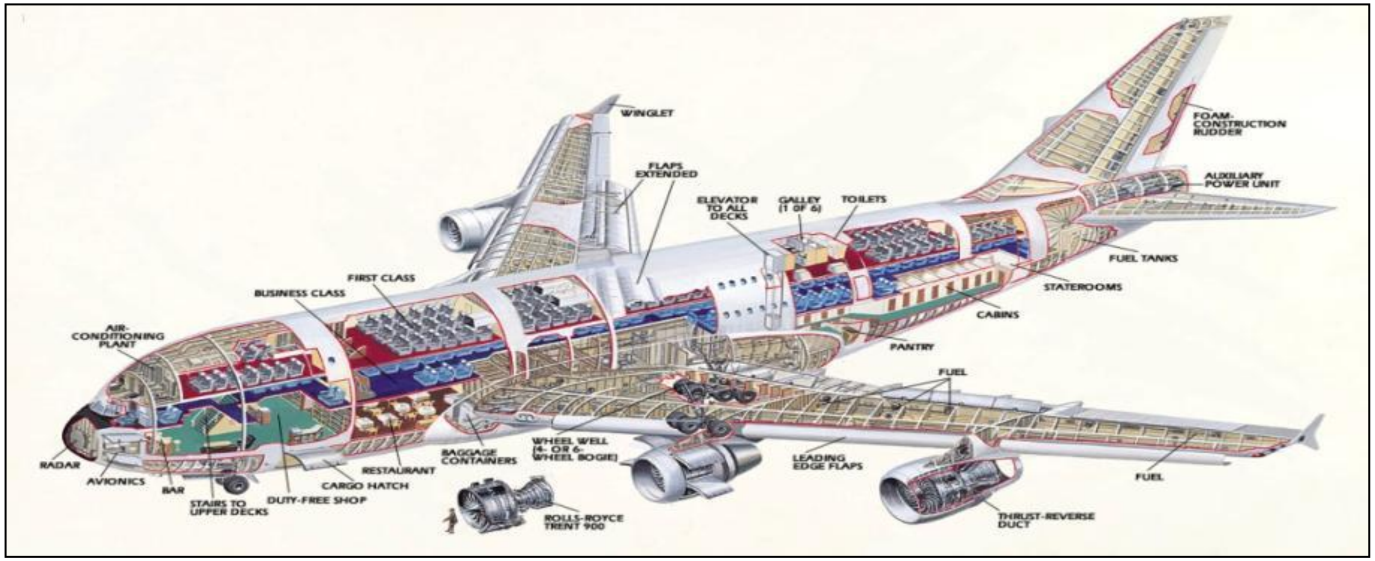
Weight?

Costs?

Safety?

Timing?

...



Security?

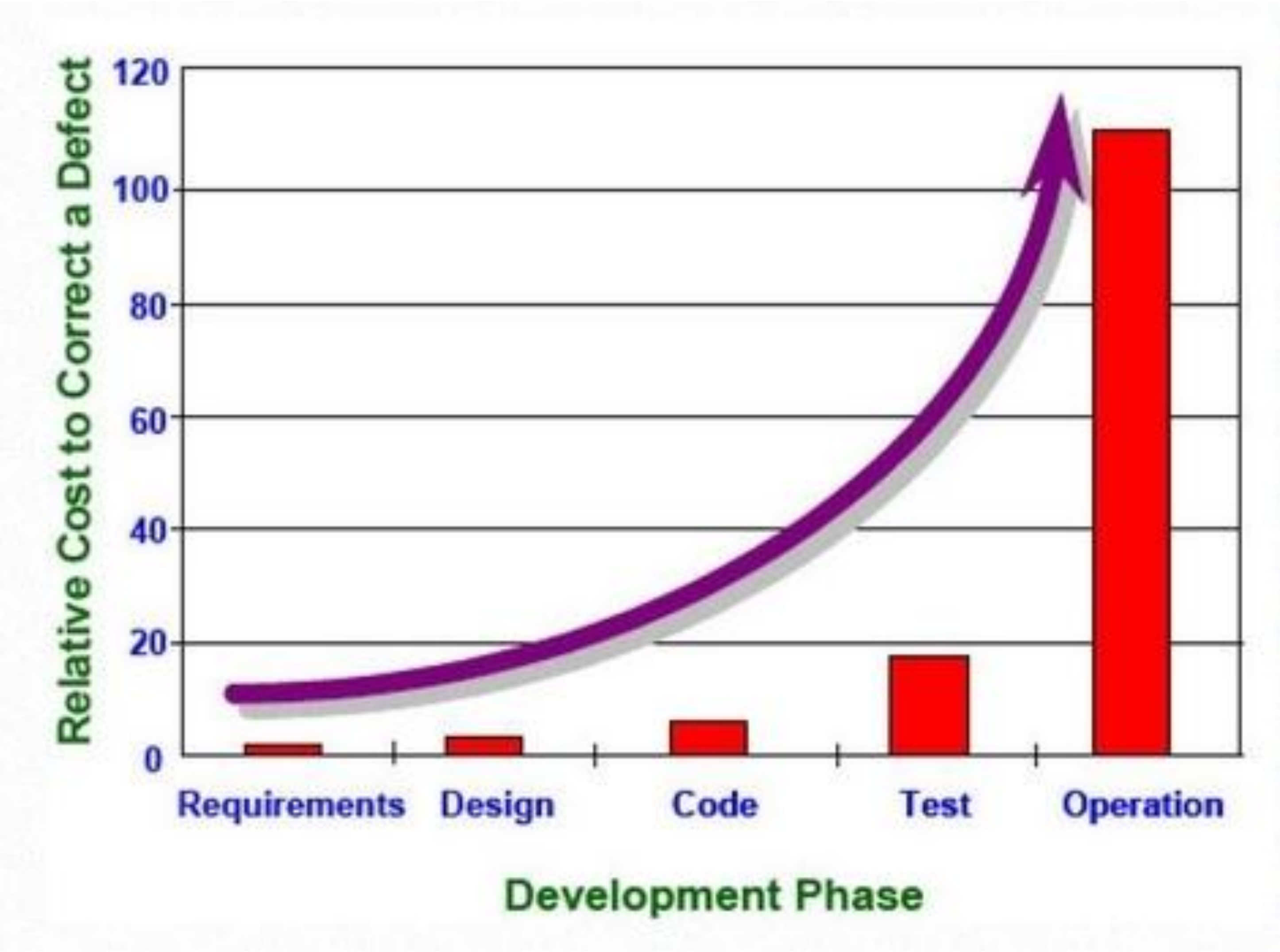
Flexibility?

Maintenance?

Production?

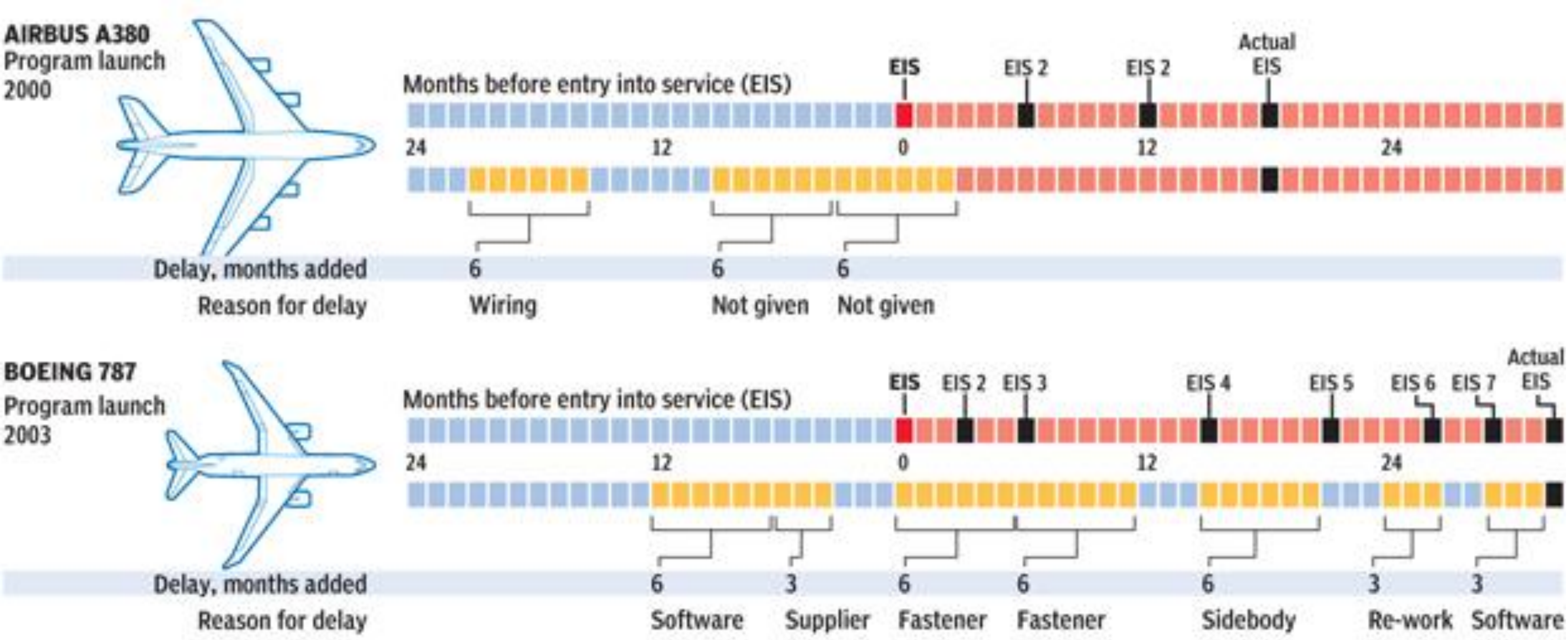
Power Consumption?

Motivation Airbus Group



Source:
http://newsbytes.ph/2014/08/11/blog-jollibee-chickensad-an-it-management-case-study/

Airbus Group Motivation



EIS: 2007
11B€

EIS: 2011
15B$

Source:
http://business.financialpost.com/2011/11/15/bombardiers-bid-to-avoid-boeings-mistakes/

Airbus Group Motivation

First Flight: 1963

Expected life: 2020

First Flight: 1987

Expected life: ???

First Flight: 1974

Expected life: 2019

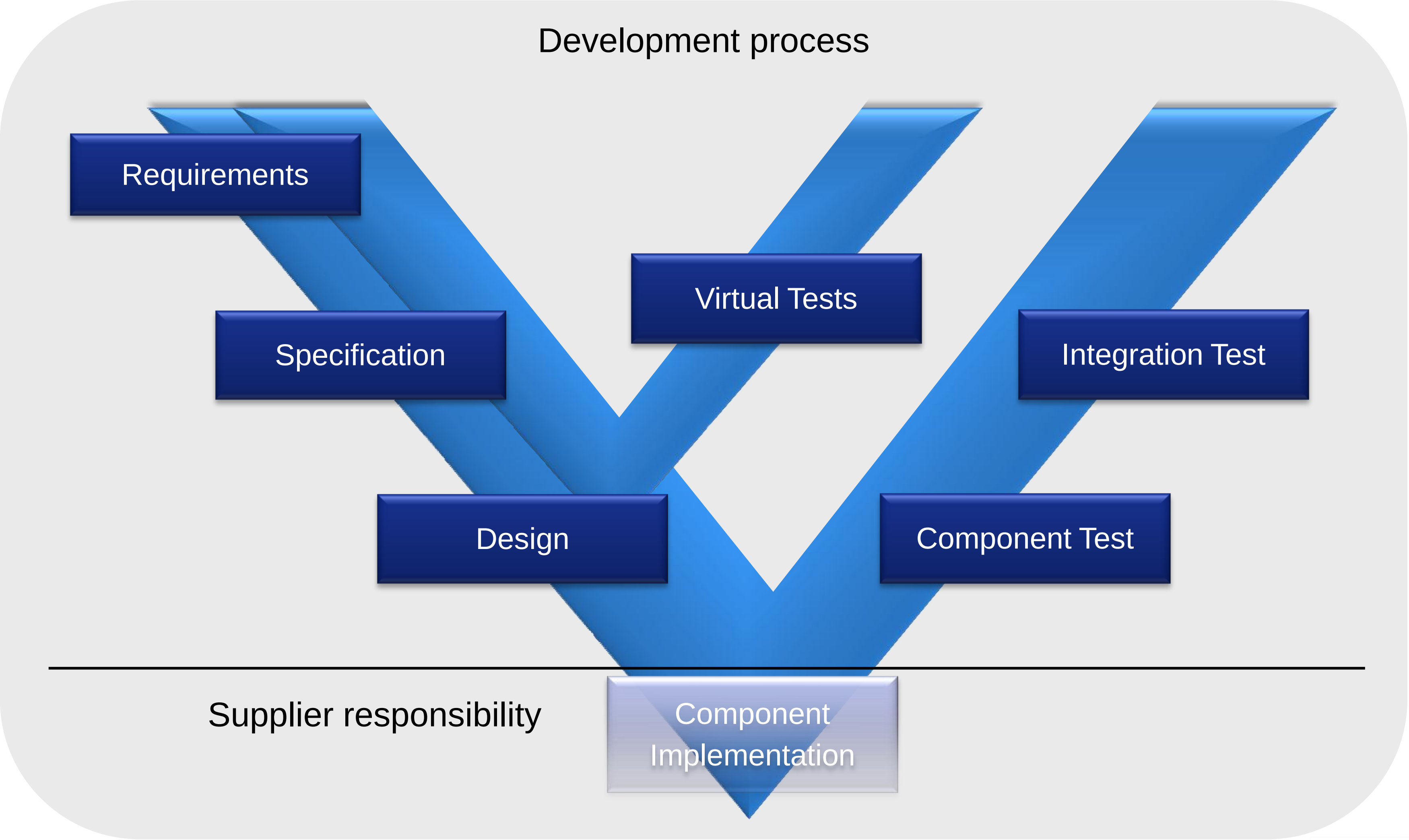
Source:

http://en.wikipedia.org/wiki/Transall_C-160

http://en.wikipedia.org/wiki/Panavia_Tornado

http://en.wikipedia.org/wiki/Airbus_A320_family

Typical system development process


Development process

Requirements

Specification

Design

Virtual Tests

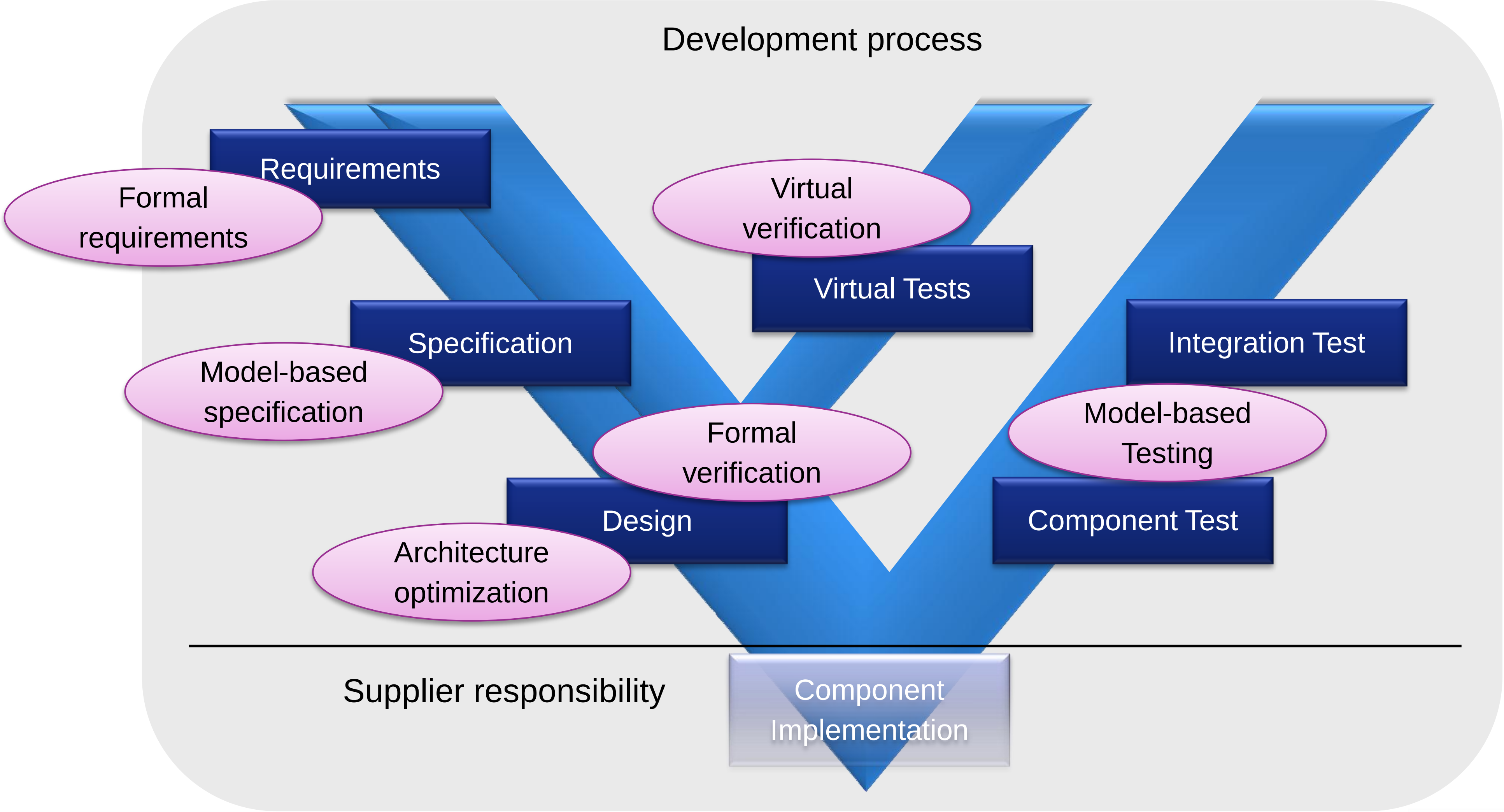
Integration Test

Component Test

Supplier responsibility

Component Implementation

Typical system development process



Formal requirements

What we see today:

- Natural language requirement still dominate the system requirements definition
- Working formal specification approaches exist

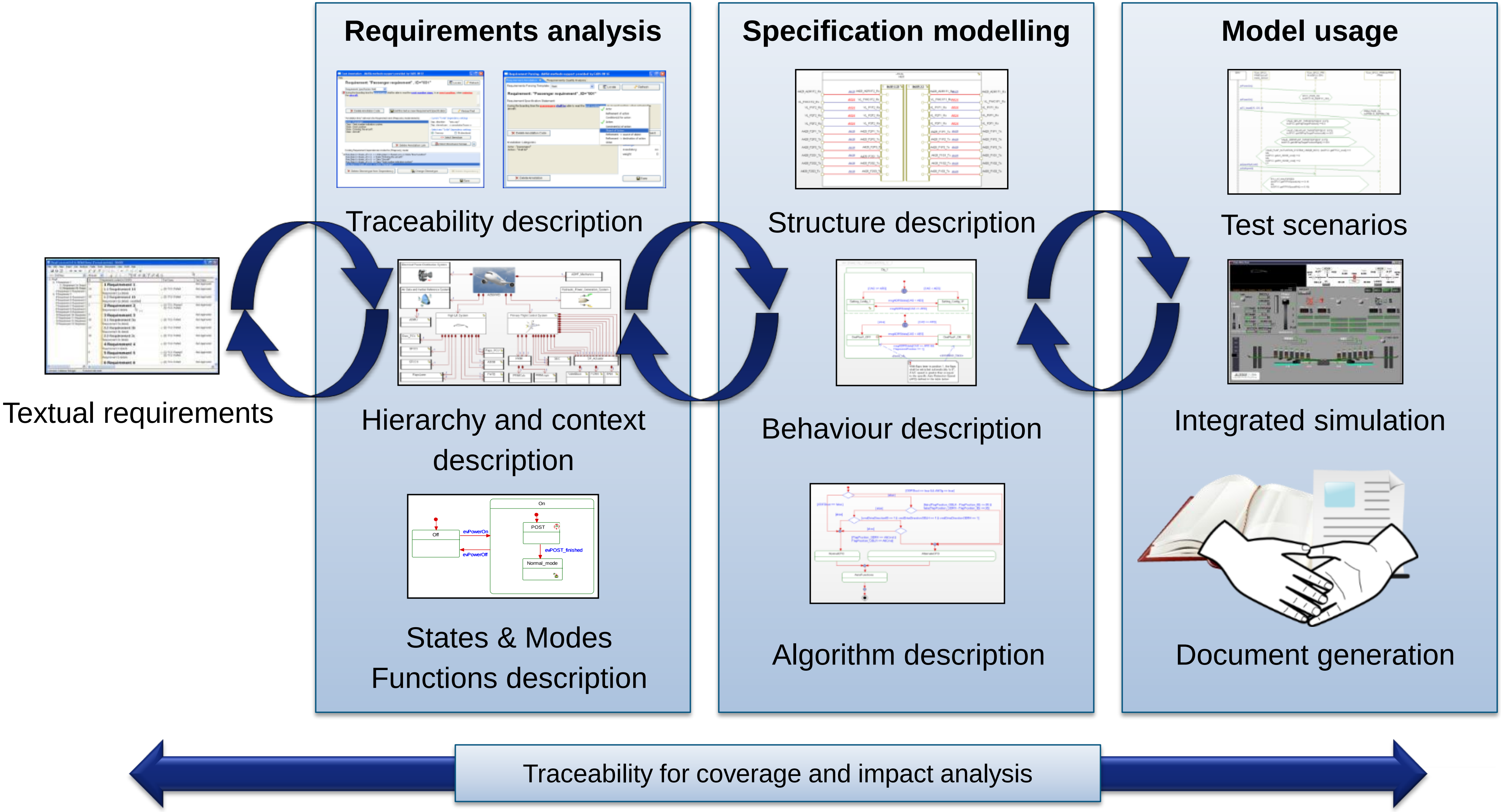
But:

- Increasing systems complexity make it increasingly impossible to communicate the system definition using text alone (inconsistencies, gaps, contradictions,…)
- Formal languages not alluring to system engineers

Open challenges:

- Make model-based approaches and formal specification more usable for system engineers (structured text, boilerplates, domain-specific formal languages for writing requirements, tool supported formalization)

Model-based specification



Requirements analysis

Specification modelling

Model usage

Textual requirements

Traceability description

Hierarchy and context description

States & Modes Functions description

Structure description

Behaviour description

Algorithm description

Test scenarios

Integrated simulation

Document generation

Traceability for coverage and impact analysis

Model-based specification

What we see today:

- Model-based approaches for modelling functional requirements,
- Document generation and model execution for specification validation is possible

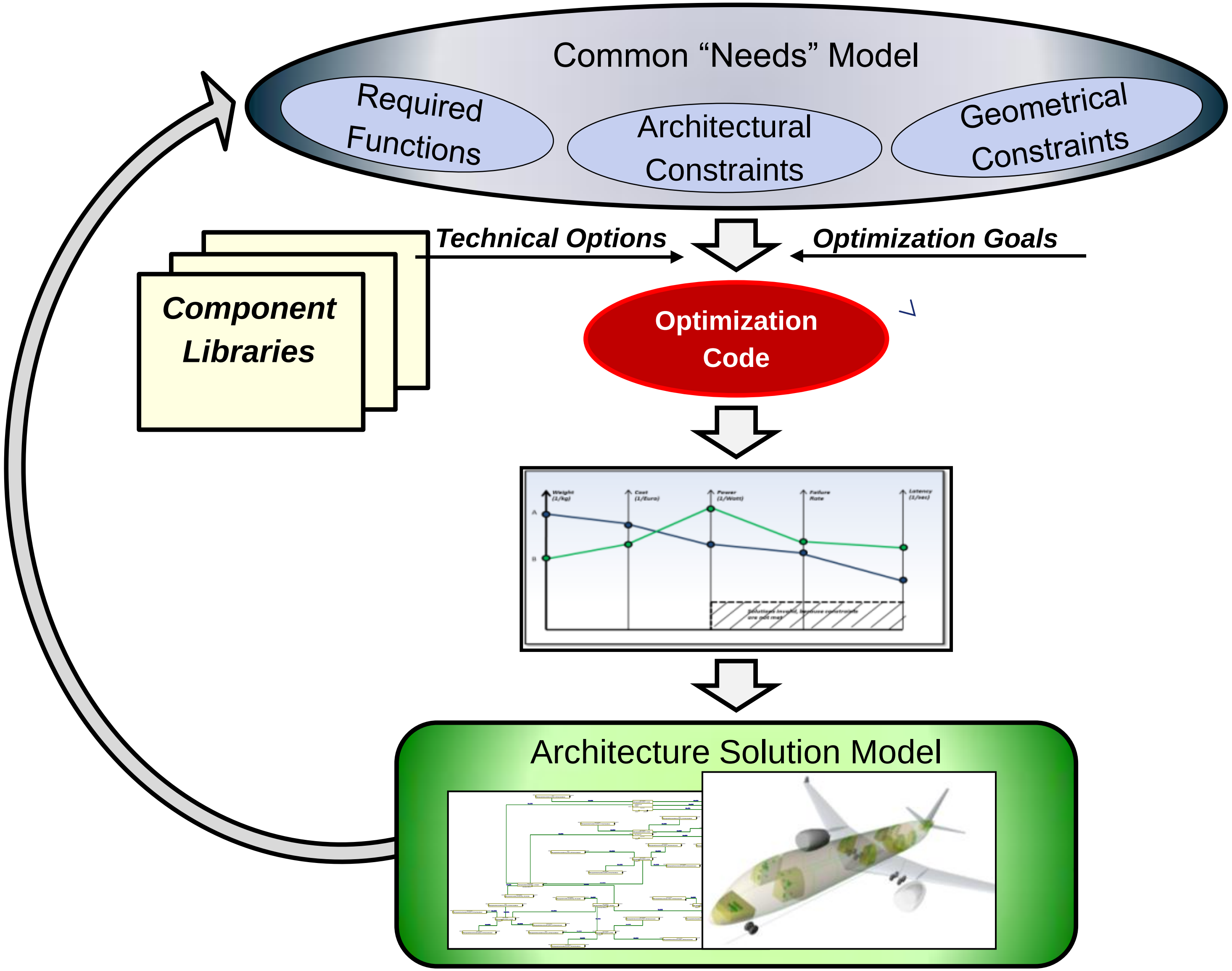
But:

- Non-functional requirements (and other process requirements, e.g. installation) are often still in natural language
- Very slow uptake in industrial practice
- Tools are not adapted to systems engineer needs (many tools started in the SW domain)

Open challenges:

- Increase usability for system engineers, focus on the way they used to express the specification

Architecture optimization



Common "Needs" Model

Required Functions

Architectural Constraints

Geometrical Constraints

Technical Options

Optimization Goals

Component Libraries

Optimization Code

Architecture Solution Model

Architecture optimization

What we see today:

- Proof-of-concept demonstrator for system architecture optimization approach
- Isolated optimisation (cable routing, task allocation,…)

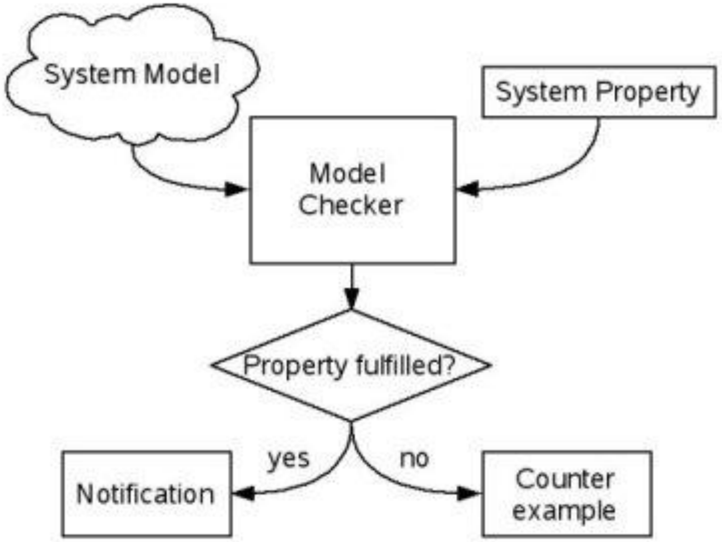
But:

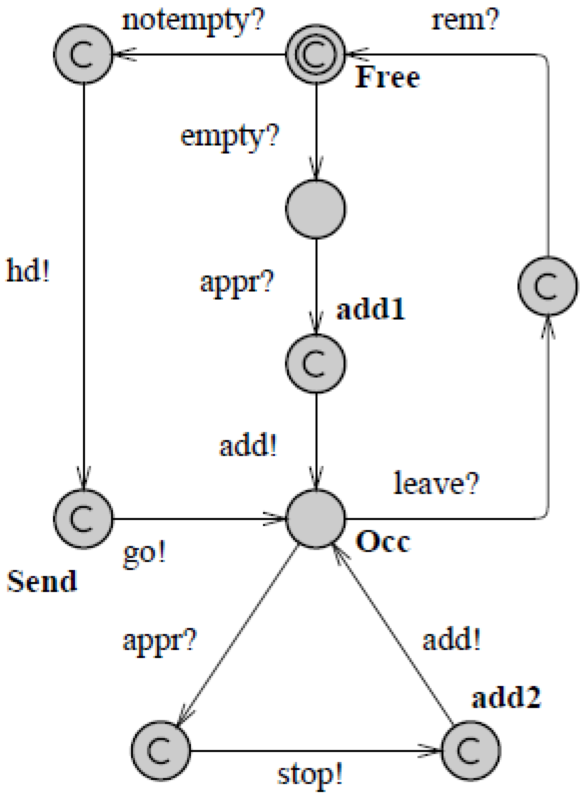
- Focus on standard metrics (weight, cost, …)
- Does not consider dynamic behaviour
- Limited to linear problems

Open challenges:

- Consider more metrics during optimization
- Measurement of hard-to-measure metrics (security, maintainability, flexibility,…)
- Efficient algorithms for non-linear optimization problems

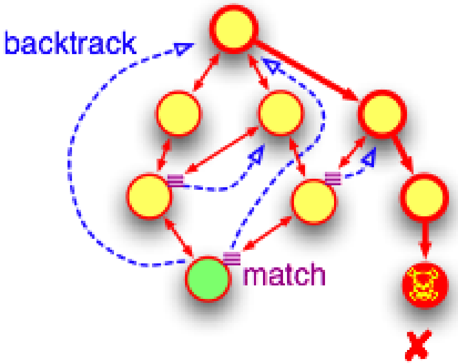
Formal verification





model checking:

all program state are explored
until none left or defect found



http://embsys.technikum-wien.at/projects/decs/verification/formalmethods.php
http://babelfish.arc.nasa.gov/trac/jpf/iki/intro/testing_vs_model_checking
http://www.cs.aau.dk/~adavid/publications/21-tutorial.pdf

Formal verification

What we see today:

- Usable industrial strength model checkers are available (UPPAAL, CBMC, SPIN)

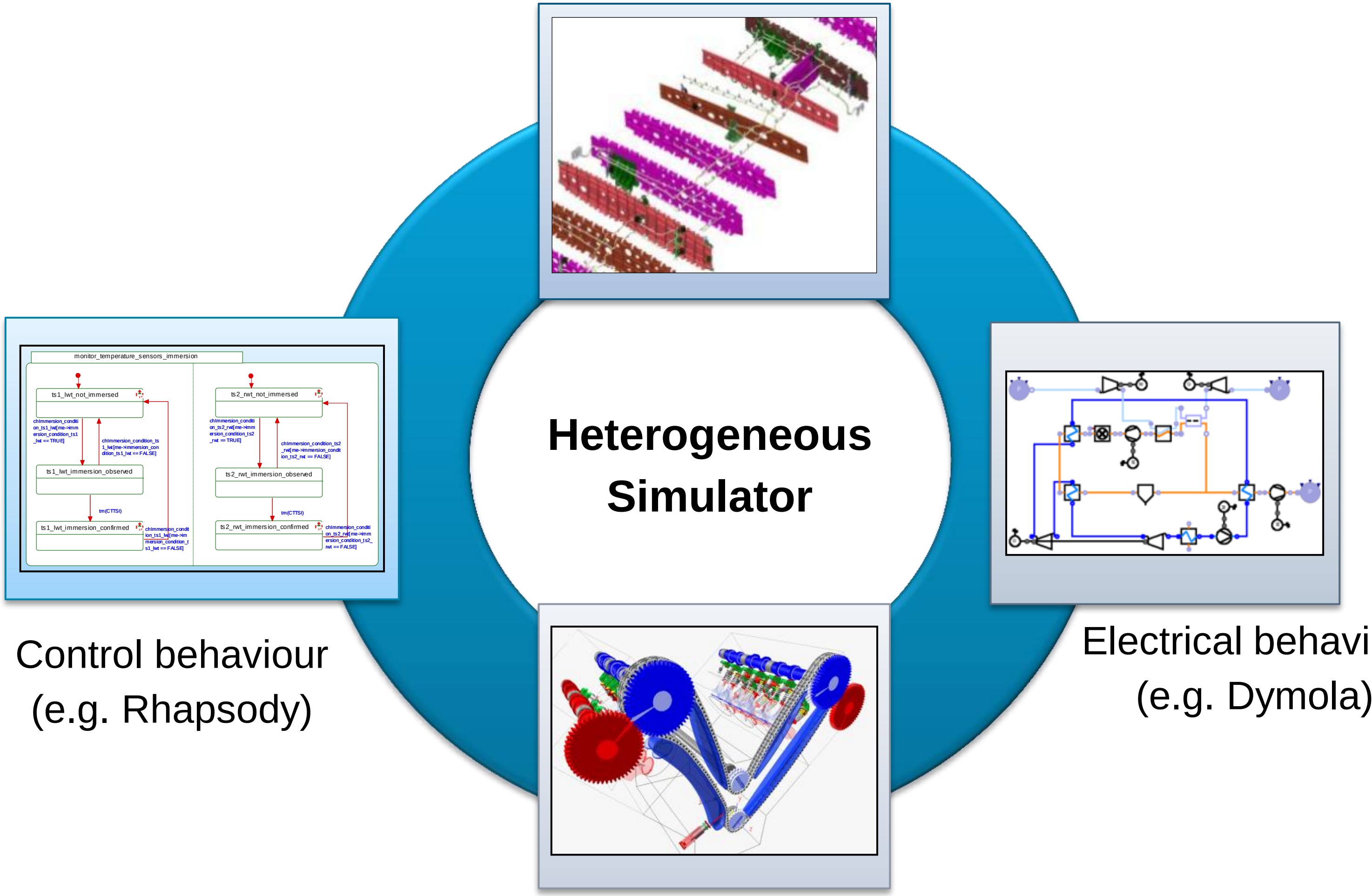
But:

- Available model checkers usually work on a proprietary language (UPPAAL) or heavily restrict the usage of standard modelling languages (STSTest)
- Model checkers either work on code (CBMC) or on a model not on models that include code
- Scalability is still an issue
- Formal verification hardly used in industrial systems engineering

Open challenges:

- Easy to use model checkers that work on large-scale system models which include code for model execution (see MBS) integrated into existing tool chains
- Making sure that an implemented component behaves according to the checked model

Virtual verification



Heterogeneous
Simulator
monitor_temperature_sensors_immersion
ts1_lwt_not_immersed
chImmersion_condition_ts1_lwt[me->immersion_condition_ts1_lwt == TRUE]
chImmersion_condition_ts1_lwt[me->immersion_condition_ts1_lwt == FALSE]
ts1_lwt_immersion_observed
tm(CTTSI)
ts1_lwt_immersion_confirmed
chImmersion_condition_ts1_lwt[me->immersion_condition_ts1_lwt == FALSE]
ts2_rwt_not_immersed
chImmersion_condition_ts2_rwt[me->immersion_condition_ts2_rwt == TRUE]
chImmersion_condition_ts2_rwt[me->immersion_condition_ts2_rwt == FALSE]
ts2_rwt_immersion_observed
tm(CTTSI)
ts2_rwt_immersion_confirmed
chImmersion_condition_ts2_rwt[me->immersion_condition_ts2_rwt == FALSE]

Visualisation in DMU
(e.g. Catia)

Control behaviour
(e.g. Rhapsody)

Electrical behaviour
(e.g. Dymola)

Mechanical behaviour
(e.g. SimPack)

Virtual verification

What we see today:

- Simulation is a standard mean for virtual verification
- FMI as emerging standard for simulation exchange

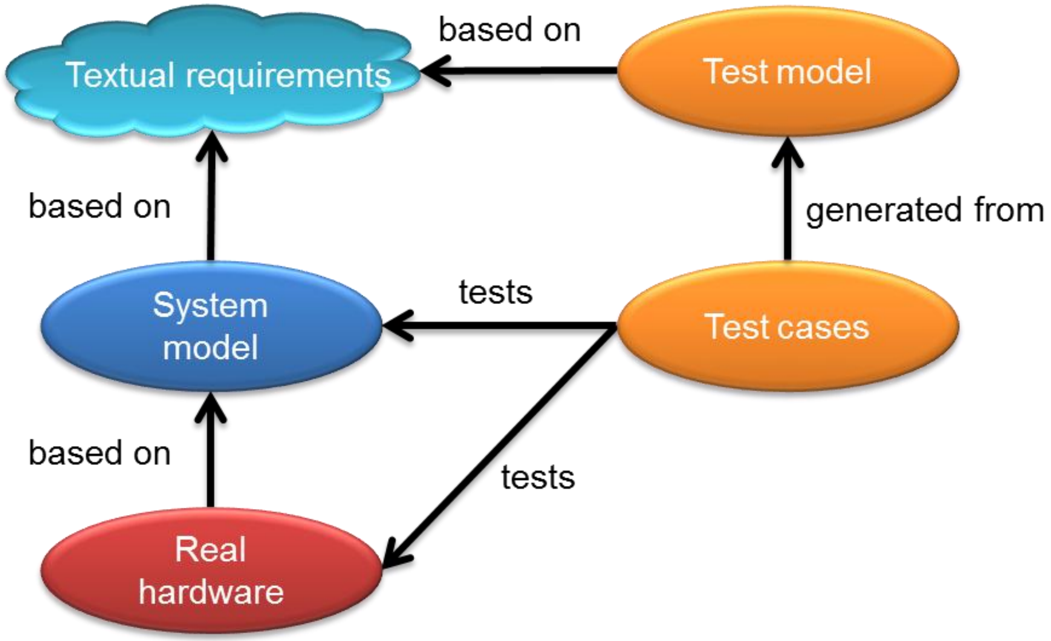
But:

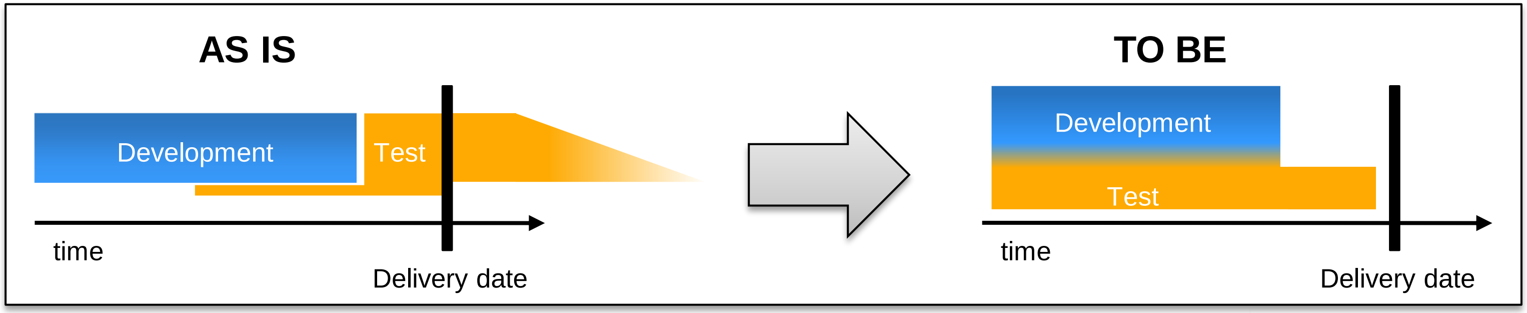
- Multi-domain simulation are often pressed into a single tool
- Co-simulation often not easy to use

Open challenges:

- Simulation environment that allow choosing the best tool for the job
- Seamless integration of simulations coming from different tools

Model-based testing





Model-based testing

Where we are today:

- Model-based testing has been proven as an effective technique for testing (see e.g. ARTEMIS MBAT project) and commercial tools are available

But:

- MBT typically generates a large number of test cases based on exhaustive search
- Expert knowledge is not taken into account in order to optimize the generated test suite include all relevant and realistic test cases

Open challenges:

- Integration of testing and development activities
- Reuse of artefacts from the development phase in the testing phase
- Optimisation of test suites from different sources
- Identification of the most critical test cases in a test suite

Conclusion

- Many open challenges for complex systems development
- Main obstacle for adaptation of new methods and tools is usability for engineers
- Integration into existing workflows and tool environments is important

- However, it is crucial for ensure research questions are solved in realistic context
  - Scalability and usability considerations should be part of the problem definition from the beginning
  - Make sure that end users are involved from the beginning
  - Any new solution should be able to communicate its practical relevance, expected acceptance of end users, and perspectives for mid- and long-term applications

Questions?

Airbus Group Innovations

Philipp Helle
Philipp.Helle@airbus.com