Critical embedded systems : trends in the design methods

Nicolas NAVET

INRIA / RealTime-at-Work

Real-time and interoperability team (TRIO)

http://www.loria.fr/~nnavet

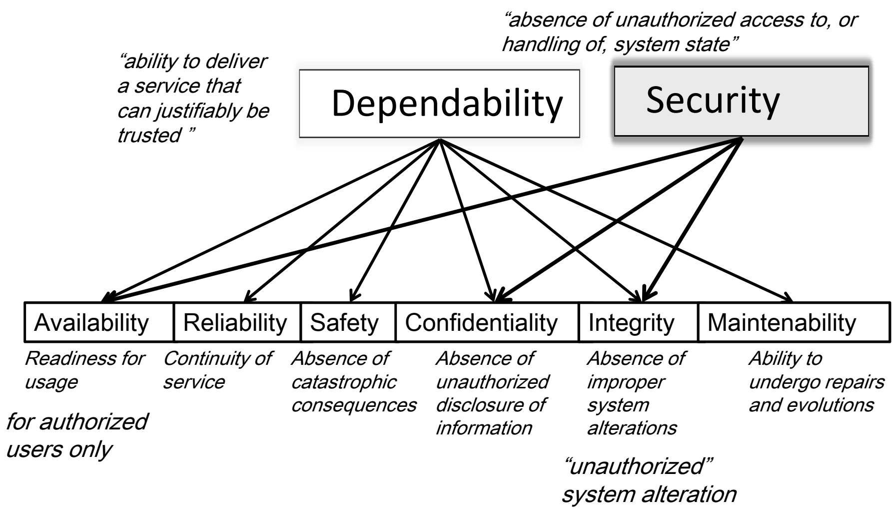
Industrial Trend Forecasting day of the CNRS GDR ASR, Paris - 03/11/2011

Outline

1. **Illustration : automotive embedded systems**
   - **Threats to their dependability ?**
   - **Focus on the timing constraints**
2. **Evolution technologies and practices in the design of critical embedded systems**
3. **Open and emerging problems**

Embedded systems in our day-to-day life : some of them are critical in the sense they are subject to dependability constraints

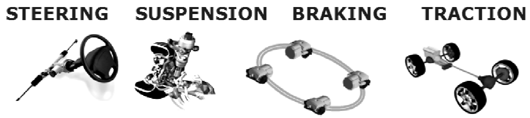
Dependability vs Security [from Laprie et al, 3]



Automotive Embedded Systems: threats to their dependability

Electronics is the driving force of innovation in automotive

Many new functions are safety critical: brake assist, cruise control, lane keeping, dynamic lights, etc

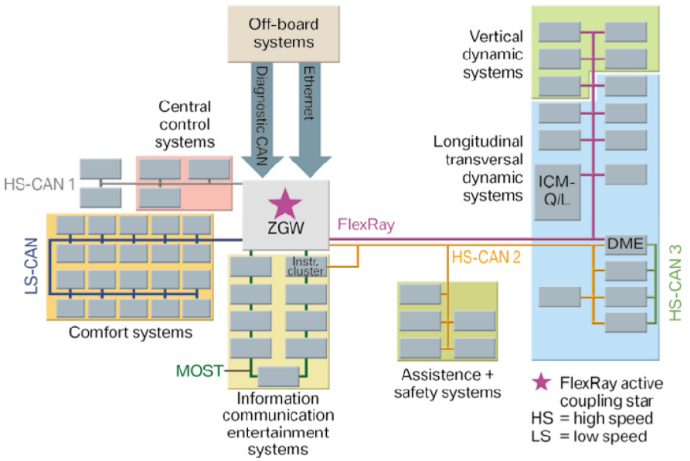


Picture from [10]

- 90% of new functions use software
- Electronics: 40% of total costs
- Huge complexity: 70 ECUs, 2500 signals, >6 comm. protocols, multi-layered run-time environment (AUTOSAR), multi-source software, multi-core CPUs, number of variants, etc

Strong costs and time-to-market constraints !

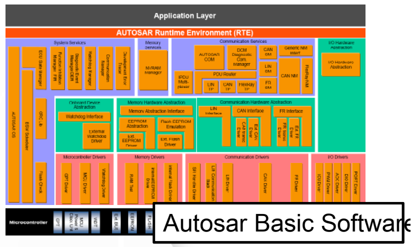
BMW 7 Series networking architecture [11]



Picture from [11]

- ZGW = central gateway
- 4 CAN buses
- 1 FlexRay Bus
- 1 MOST bus
- Several LIN Buses (not shown here)
- 1 Ethernet bus
- Wireless

Main impediments to safety imho: complexity!



Technologies: numerous, complex and not explicit. conceived for critical systems

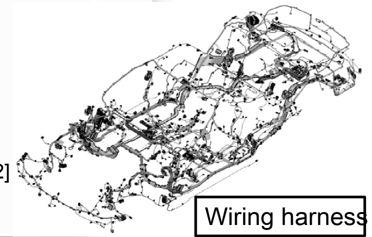
- e.g.: more than 150 specification documents (textual) for Autosar, no two implementations can behave identically!

Size of the system!

- Number of functional domains, buses, gateways, ECUs, size of code, tasks, wiring, number of variants, etc

Design process

- Most developments are not done in-house : less control on externalized developments
- Carry-over / Vehicle Family Management : need to share/re-use architecture and sub-systems between several brands/models with different requirements [2]
- Optimized solutions for each component / function does not lead to a global optimal [2]



Picture from [11]

Threats to dependability : the big picture

When faults are introduced in the development phase ?

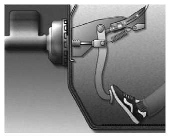
- Requirements capture (20%) + Specification (50%) + SW development: (30%) (infineon [10])
- HW development : ε

Risk factors beside complexity:

- Technologies: not all conceived with dependability as a priority
- Little hardware redundancy
- Developments are mainly externalized: incomplete knowledge for the OEM technical parameters are regarded as less important than cost for supplier / components selection [2]
- Strong cost / time-to-market pressure
- Limited regulatory constraints even with upcoming ISO26262
- Verification / validation does not ensure 100% coverage, limited used of formal methods
- Human factors
- etc

Focus on the timing constraints

Several hundreds of timing constraints – example of an end-to –end constraints



Constraint :
brake light on < 50ms

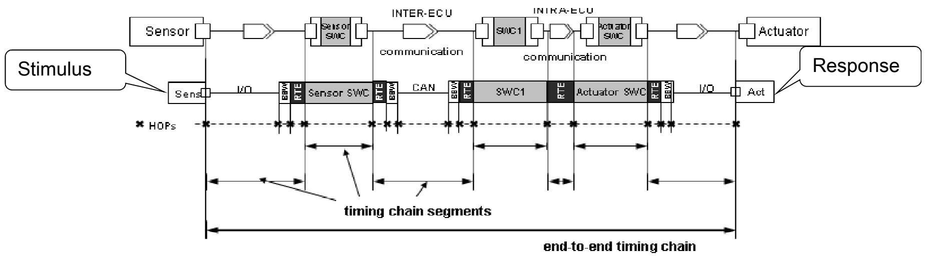
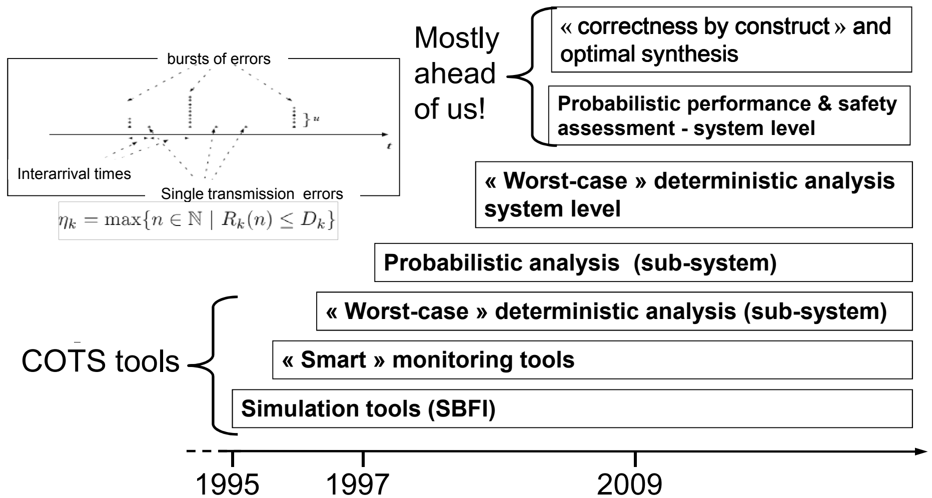


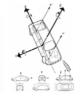
Figure from [12]

Verification of the timing constraints
Personal view / experiences



$$\eta_k = \max\{n \in \mathbb{N} \mid R_k(n) \le D_k\}$$

Why timing constraints may not be respected occasionally?



Lack of precise specification : hard to identify the right timing requirements for each function

Lack of traceability : from the architects to the suppliers

Flaws in the verification:

- Knowledge of the system and its environment is incomplete:
  - What is done by the suppliers?
  - Implementation choices really matter and standards do not say everything
  - Environmental issues: EMI, α-particles, heat, etc
  - Traffic is not always well characterized and/or well modeled e.g. aperiodic traffic ?! see [5]
- Testing /simulation alone is not enough
- Analysis is not enough too:
  - Analytic models, especially complex ones, can be wrong (remember " CAN analysis refuted, revisited, etc" [6] ?!)
  - They are often much simplified abstraction of reality and might become optimistic: neglect FIFOs, hardware limitations

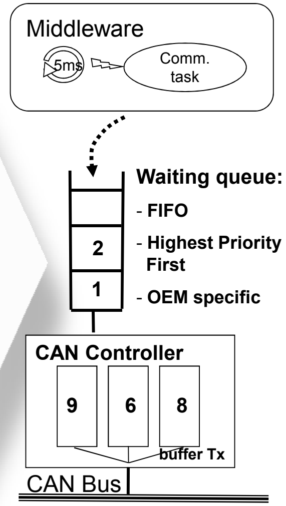
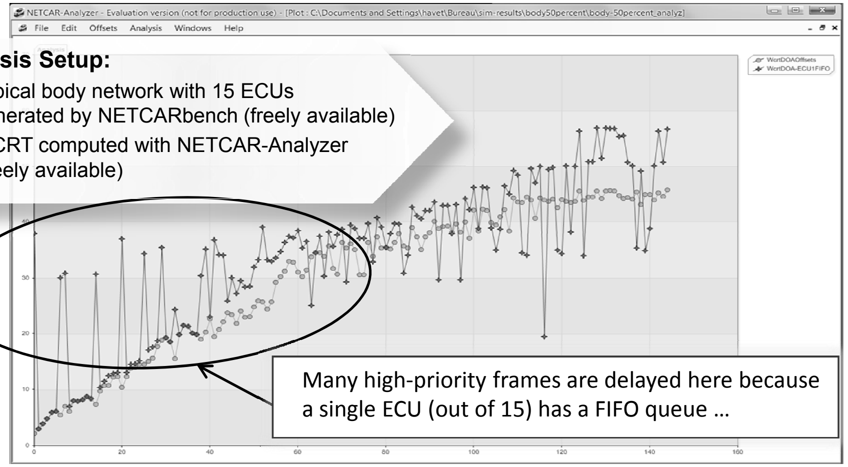


Illustration: Worst-Case Response Times on a CAN bus

Frame waiting queues are HPF, except ECU1 where queue is FIFO
the OEM does not know or verification software cannot handle it …



Analysis Setup:

- Typical body network with 15 ECUs generated by NETCARbench (freely available)
- WCRT computed with NETCAR-Analyzer (freely available)

Many high-priority frames are delayed here because a single ECU (out of 15) has a FIFO queue …

Evolution in the development of safety critical software – personal views

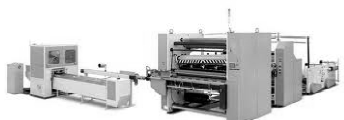
- Safety standards
- Design process
- Technologies, computing platforms

Safety standards and certification processes cannot be ignored

DO 178 / DO 254

EN 50126/28/29

IEC61226
IEC60880
…



IEC 61508

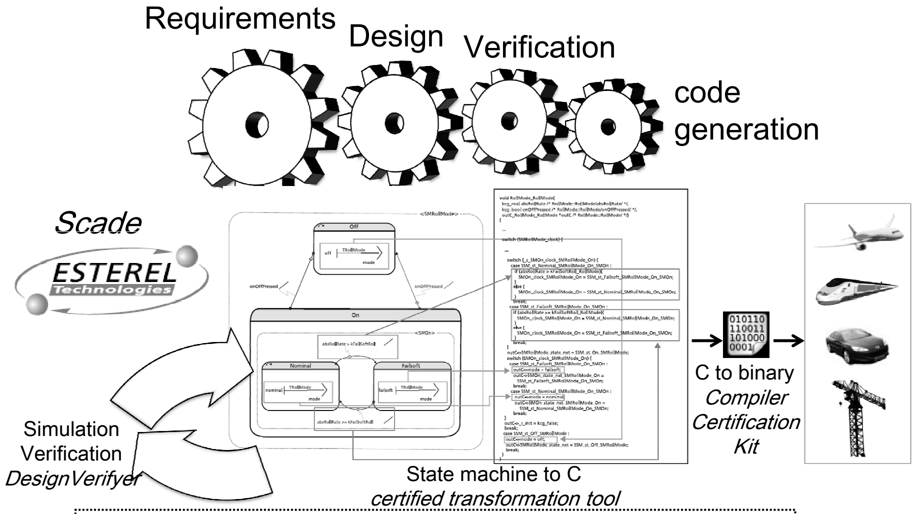
ECSS / CNES

ISO 26262

Airbus: 1/3 of the design costs of an airplane due to certification !

[Multi-domain comparison of safety standards, ERTSS-2010]

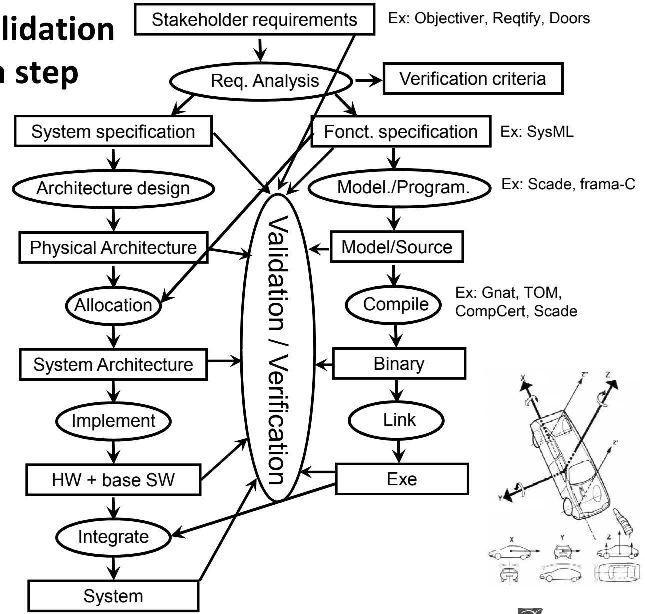
Model Based Design for dependable system development no more hand-coded programs



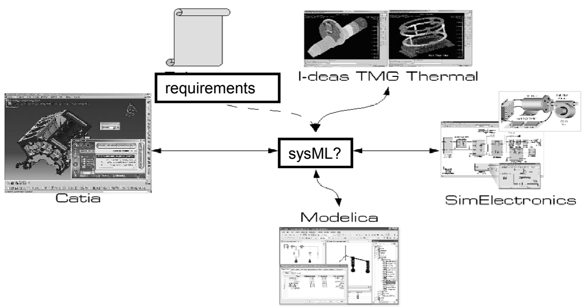
End-to-end design flows with proven outcomes at each step

Verification & Validation is needed at each step

Simplified INCOSE approach



MBD: domain-specific models and tools must be dealt with

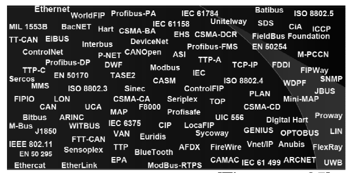


Some open issues: semantic interoperability, pivotal language? local versus global verification

Technology : from domain specific to cross-industry solutions

Today :

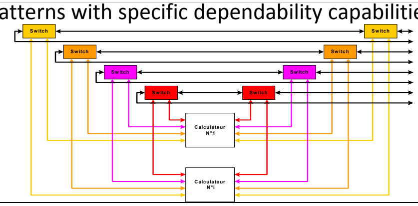
- Avionics: IEEE1553, AFDX, TTP, ARINC 653, ..
- Automotive: CAN, FlexRay, Autosar, Lin, ..
- Power plants: Alstom Alspa, Siemens Teleperm, ..

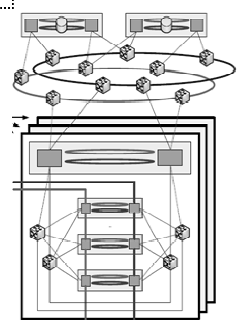


[Thomesse05]

Tomorrow : **Objective of the DDASCA consortium**

- Convergence of safety standards
- Computing platforms: cross-industry solutionS with profile per application domain and scalable dependability : e.g., switched Ethernet, virtualization, etc
- Architecture patterns with specific dependability capabilities





What is needed now: achieving affordable dependability

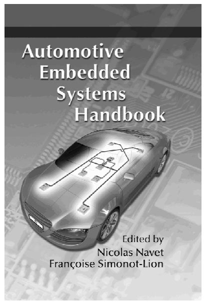
1. A large body of techniques, development processes, tools, know-how is increasingly available – they have to become more accessible
2. Simpler systems are more amenable to verification!
3. Formal methods are now sufficiently mature to handle real-world industrial problems.

Public research : provide support to both companies and public authorities so that there is no compromise in safety

Thank you for your attention

contact: nicolas.navet@inria.fr

References



[1] N. Navet, F. Simonot-Lion, editors, The Automotive Embedded Systems Handbook, Industrial Information Technology series, CRC Press / Taylor and Francis, ISBN 978-0849380266, December 2008.

[2] P. Wallin, Axelsson, A Case Study of Issues Related to Automotive E/E System Architecture Development, IEEE International Conference and Workshop on the Engineering of Computer Based Systems, 2008.

[3] A. Avizienis, J.C. Laprie, B. Randell, "Dependability and its threat: a taxonomy", IFIP Congress Topical Sessions 2004.

[4] D. Khan, R. Bril, N. Navet, "Integrating Hardware Limitations in CAN Schedulability Analysis", WiP at the 8th IEEE International Workshop on Factory Communication Systems (WFCS 2010), Nancy, France, May 2010.

[5] D. Khan, N. Navet, B. Bavoux, J. Migge, "Aperiodic Traffic in Response Time Analyses with Adjustable Safety Level", IEEE ETFA2009, Mallorca, Spain, September 22-26, 2009.

[6] R. Davis, A. Burn, R. Bril, and J. Lukkien, "Controller Area Network (CAN) schedulability analysis: Refuted, revisited and revised", Real-Time Systems, vol. 35, pp. 239–272, 2007.

[7] M. D. Natale, "Evaluating message transmission times in Controller Area Networks without buffer preemption", in 8th Brazilian Workshop on Real-Time Systems, 2006.

[8] C. Braun, L. Havet, N. Navet, "NETCARBENCH: a benchmark for techniques and tools used in the design of automotive communication systems", Proc IFAC FeT 2007, Toulouse, France, November 7-9, 2007.

[10] P. Leteinturier, "Next Generation Powertrain Microcontrollers", International Automotive Electronics Congress, November 2007.

[11] H. Kellerman, G. Nemeth, J. Kostelezky, K. Barbehön, F. El-Dwaik, L. Hochmuth, "BMW 7 Series architecture", ATZextra, November 2008.

[12] AUTOSAR, "Specification of Timing Extensions", Release 4.0 Rev 2, 2010.