

# Deadline verification and freedom from interference in safety-critical systems

Torino  
June 09, 2011

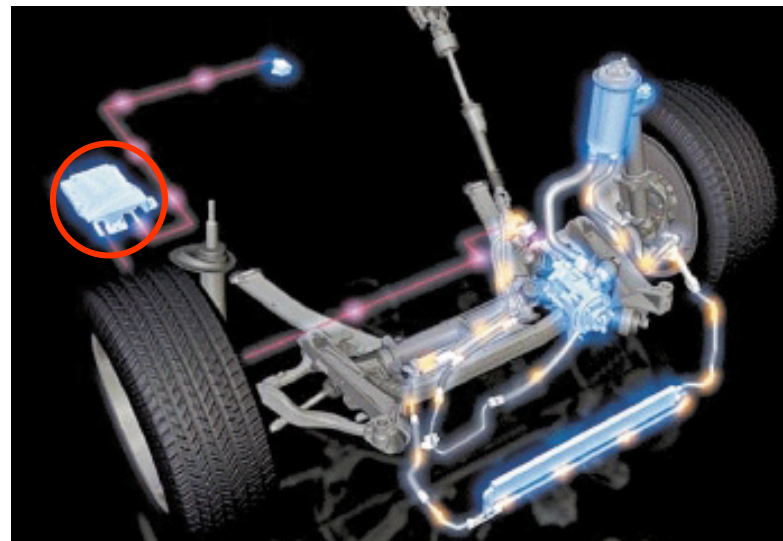
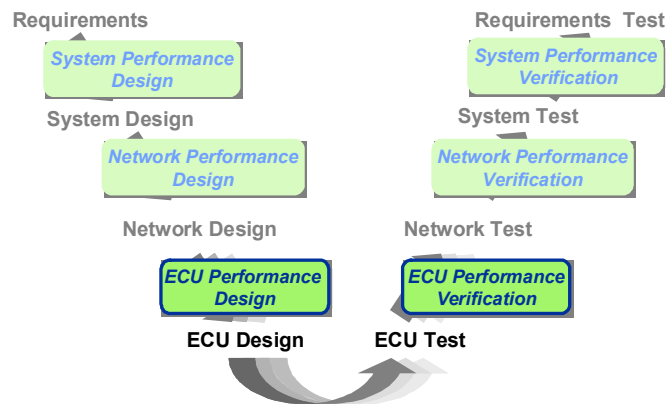


Solutions for Complex  
Real-Time Systems

# BMW Example: Safety-Critical ECU

## Chassis domain: Active Front Steering

- ❑ Verifying Performance and Timing for all critical cases
- ❑ Safeguarding against liability claims
- ❑ Optimizing ECU performance and cost (use of cheaper CPU)



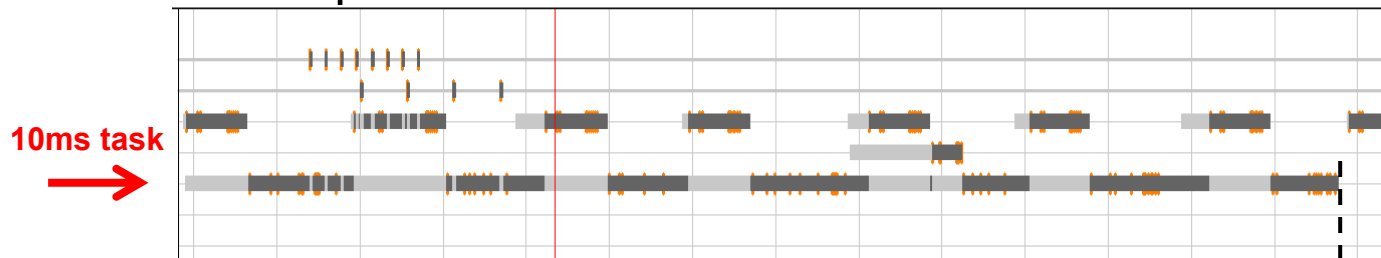
Source: BMW

*Hans Sarnowski, responsible BMW Engineer: „You really get to know your system and can detect real-time errors in a fraction of time“*

# Why Scheduling Analysis? *Reliability & Safety!*

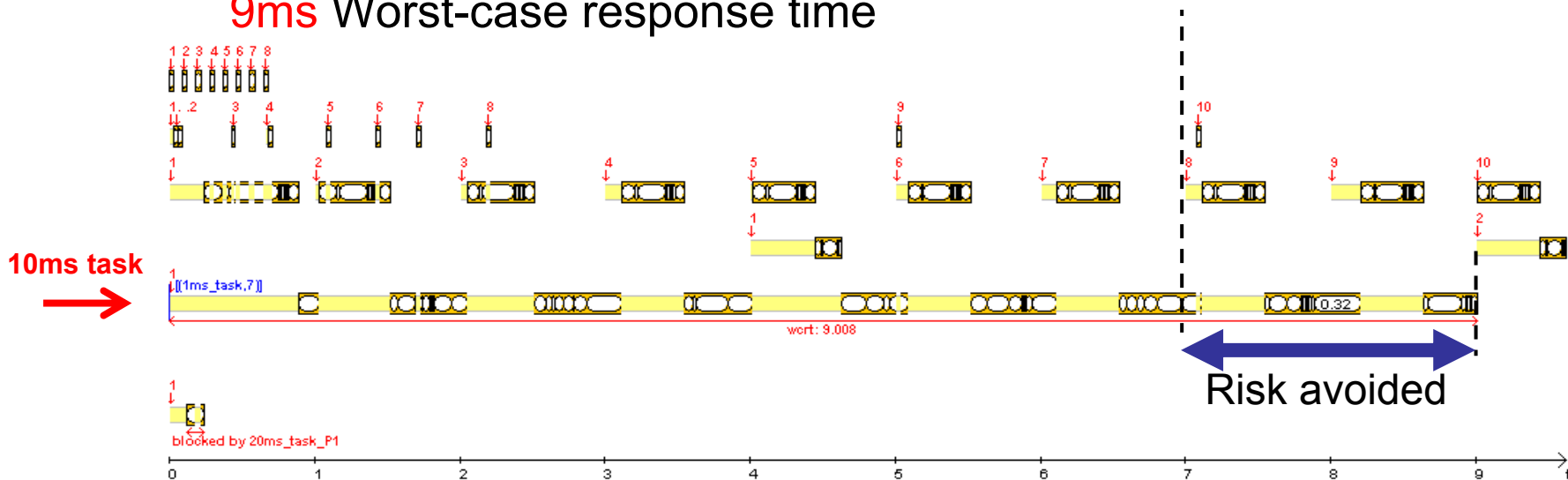
- ❑ **Tracing / Simulation does not find corner cases**

- ❑ 4 CAN, 8 SPI interrupts, 7 preemptions by 1ms task → 6.9 ms response time



- ❑ **SymTA/S guarantees worst-case coverage**

- ❑ 10 CAN, 8 SPI interrupts, 9 preemptions by 1ms task, **blocking** → 9ms Worst-case response time

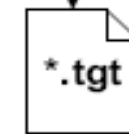
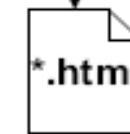
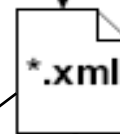
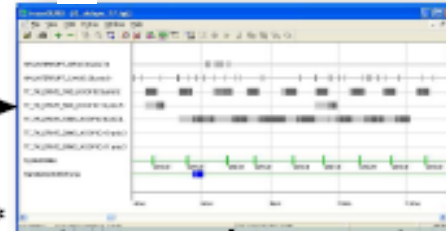


# Integration: Tracing + SymTA/S



CAN, Nexus,  
Debugger,  
oder FlexRay\*

traceGURU

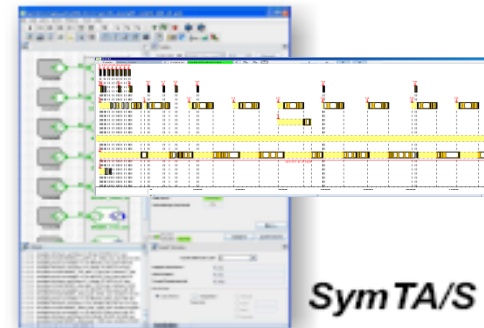
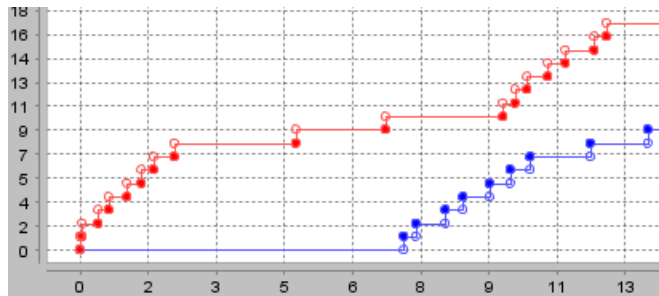


Report Trace

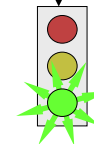
– Single function execution times

	A	B	C	D	E
1	Process	Task	BCET	WCET	Freq.
2	Proc_001	200ms_Task (3)	1,62	2,16	200
3	Proc_002	1000ms_Task (2)	3,08	3,16	1000
4	Proc_003	1000ms_Task (2)	2,36	2,36	1000
5	Proc_004	TSK_Dynamic (12)	2,76	2,76	500
6	Proc_005	TSK_Dyn	3,44	4,28	500
7	Proc_006	TSK	143,7	623,46	500

– Interrupt Frequency



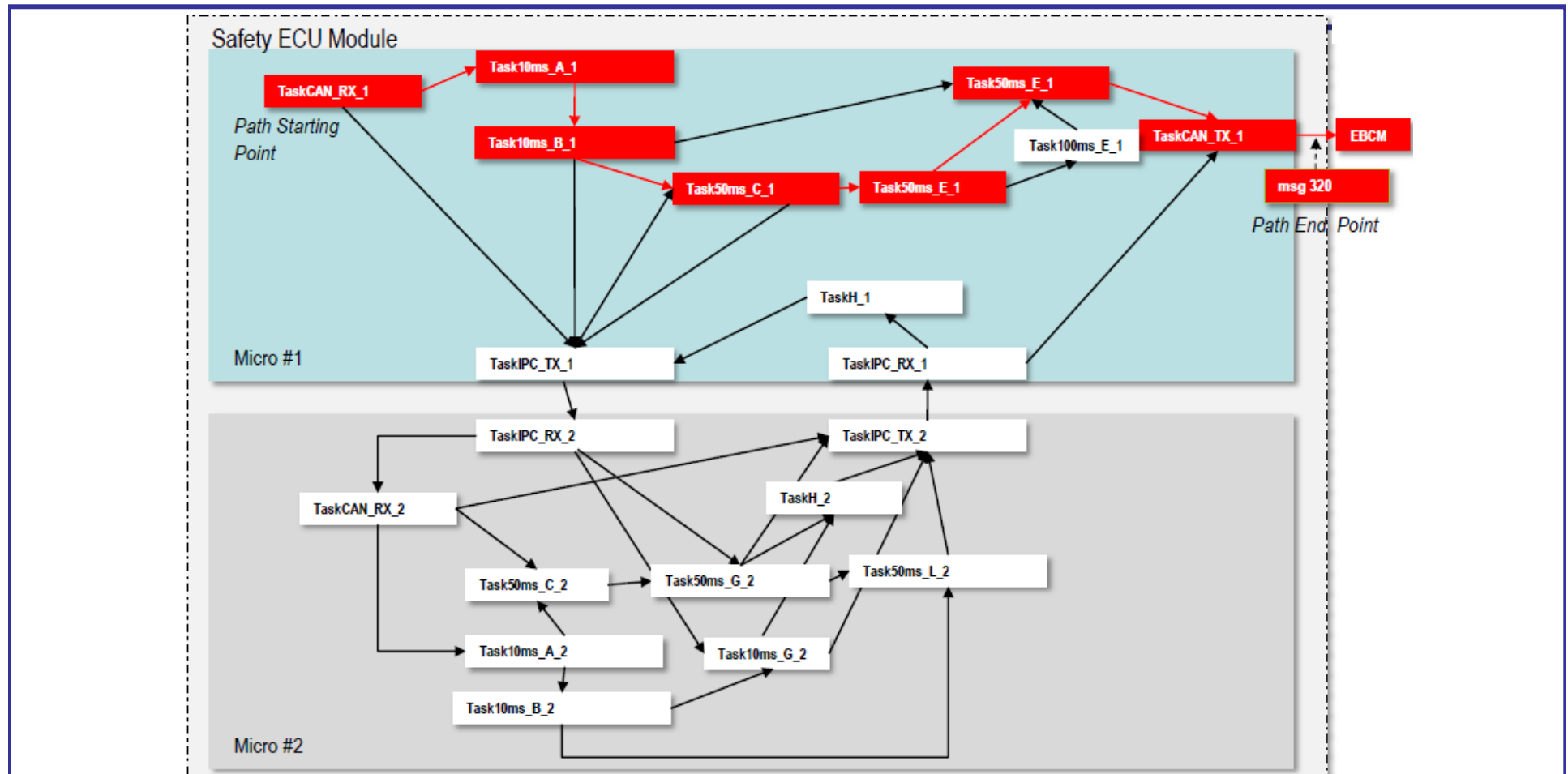
SymTA/S



Report

# Example Use Case – Active Safety Module (FEOCM)

- ❑ Safety req 1: *minimize two critical-path latencies (<100ms)*
- ❑ Safety req 2: *minimize difference between the two latencies (<10%)*



# Example Use Case – Active Safety Module (FEOCM)

- ❑ Safety req 1: *minimize two critical-path latencies (<100ms)*
- ❑ Safety req 2: *minimize difference between the two latencies (<10%)*

## ■ Original Design

- W/B Case Latencies
- Statistics

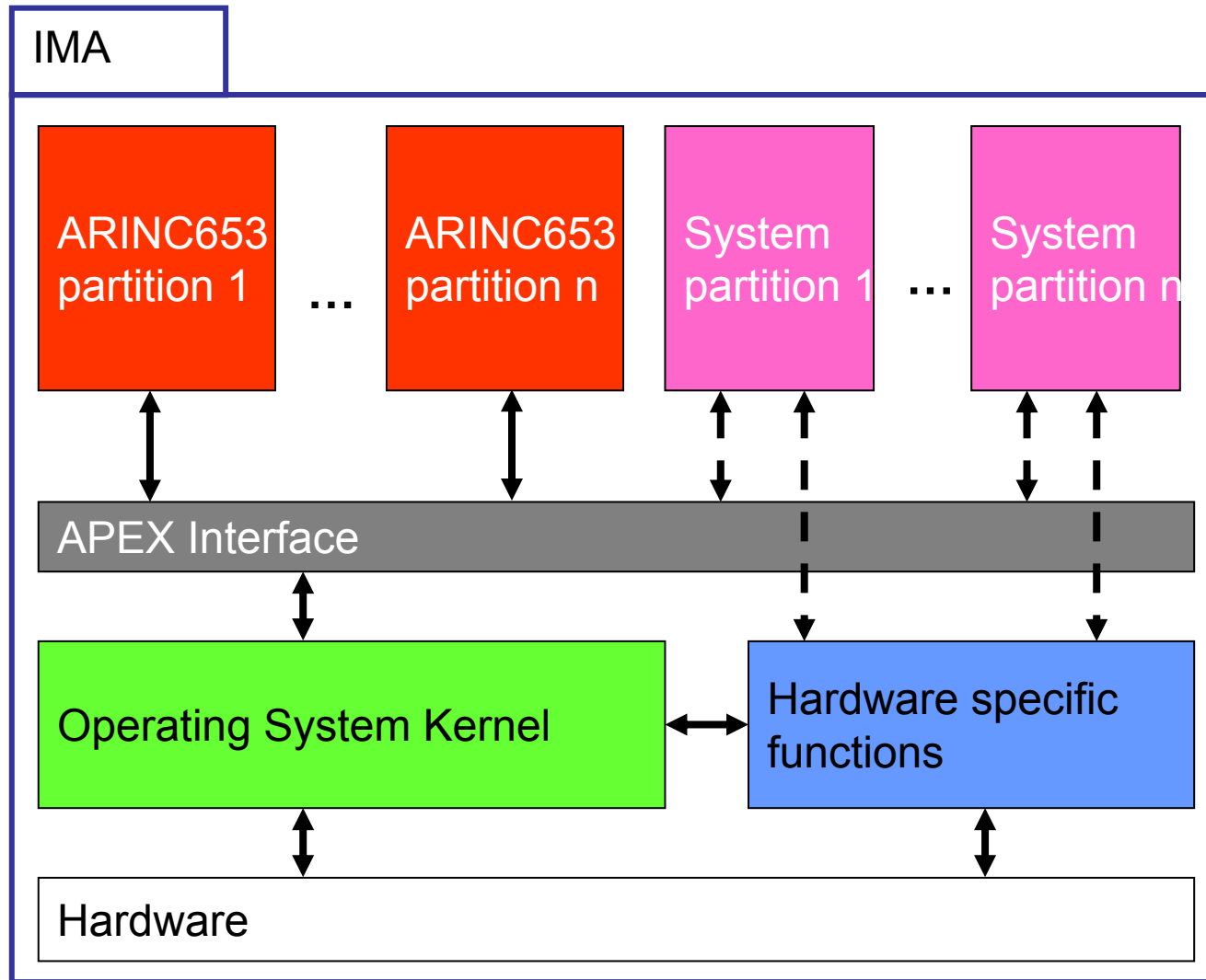
Secondary		Primary	
Best Case	Worst Case	Best Case	Worst Case
80.388	132.884	110.398	262.884
	Secondary	Primary	Secondary
Mean Latency	94.33	142.39	48

## ■ Optimized Original Design

- W/B Case Latencies
- Constraint on Latency (<100ms)

Secondary		Primary	
Best Case	Worst Case	Best Case	Worst Case
....	81.544	....	79.334

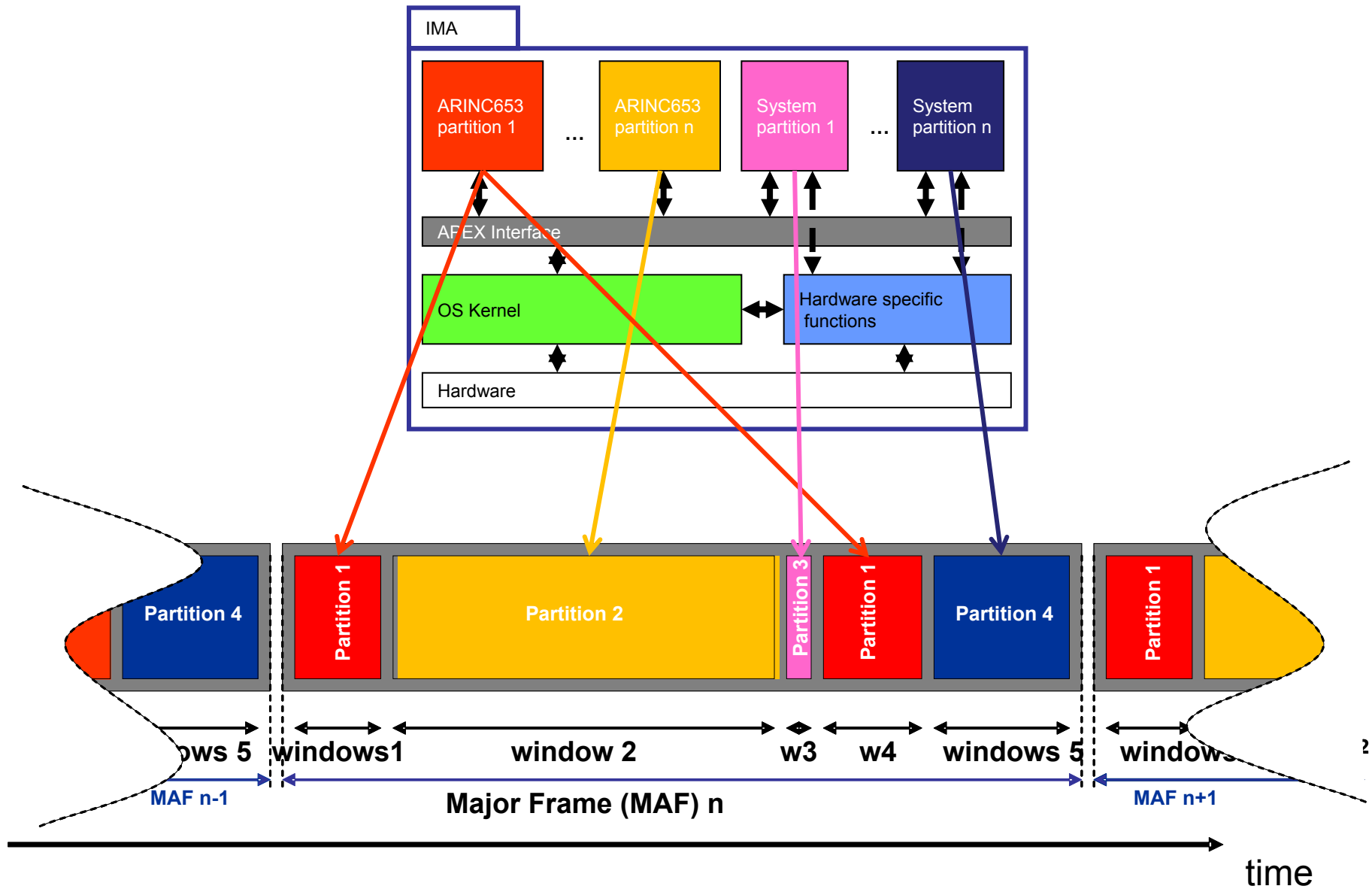
# Partitioned Operating Systems – Virtualization



APEX: Application/Executive

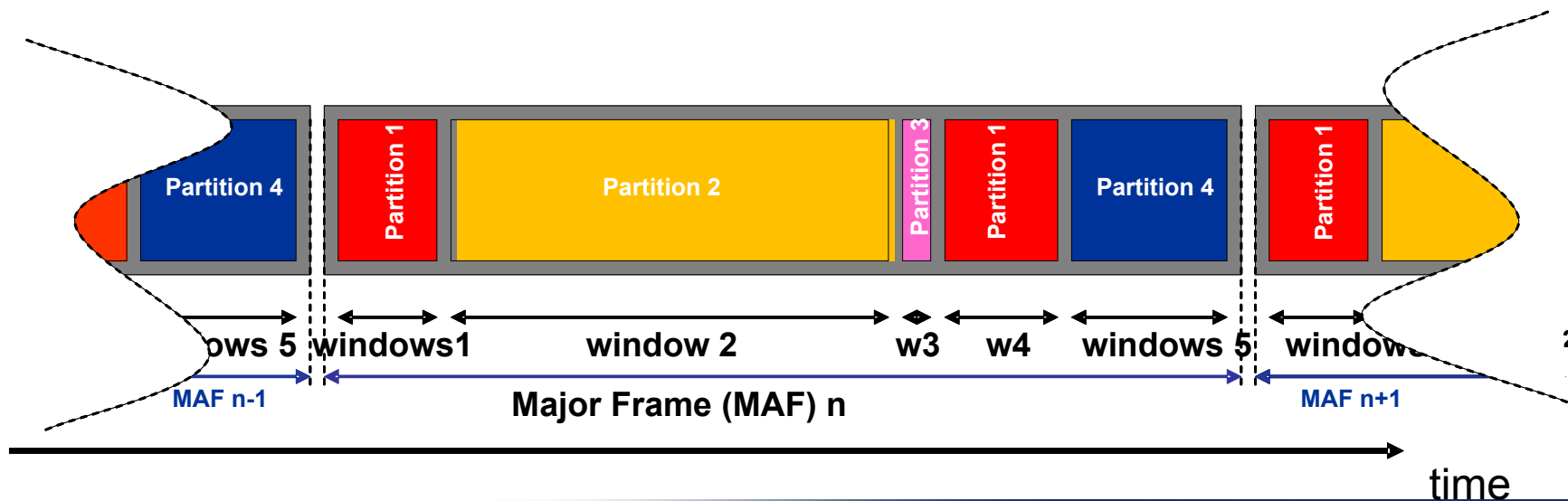
IMA: Integrated Modular Avionics

# Partitioned Scheduling of ARINC 653

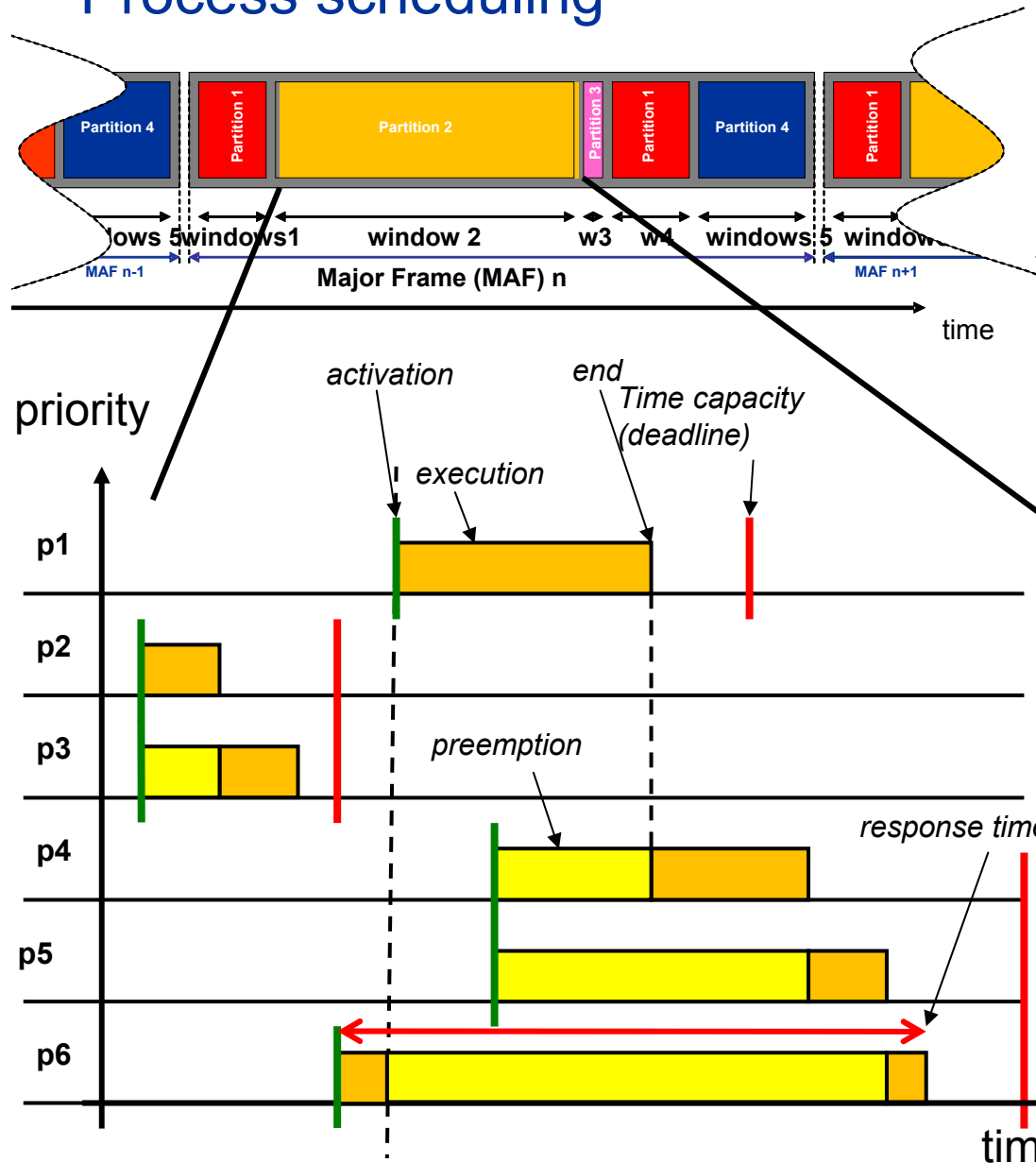


# Partitioned Scheduling – Timing Protection

- Static scheduling, Time-Division Multiple Access (TDMA)  
→ **guaranteed time-slots** for each partition, no disturbance possible
- Properties of ARINC 653 Scheduling:
  - repetitive execution of **major frames**
  - **m partitions** → **n sequential windows**



# Process scheduling



□ Partition software consists of **processes**

- period
- priority (static priority scheduling, SP)
- execution time
- **time capacity (deadline !)**
- ...

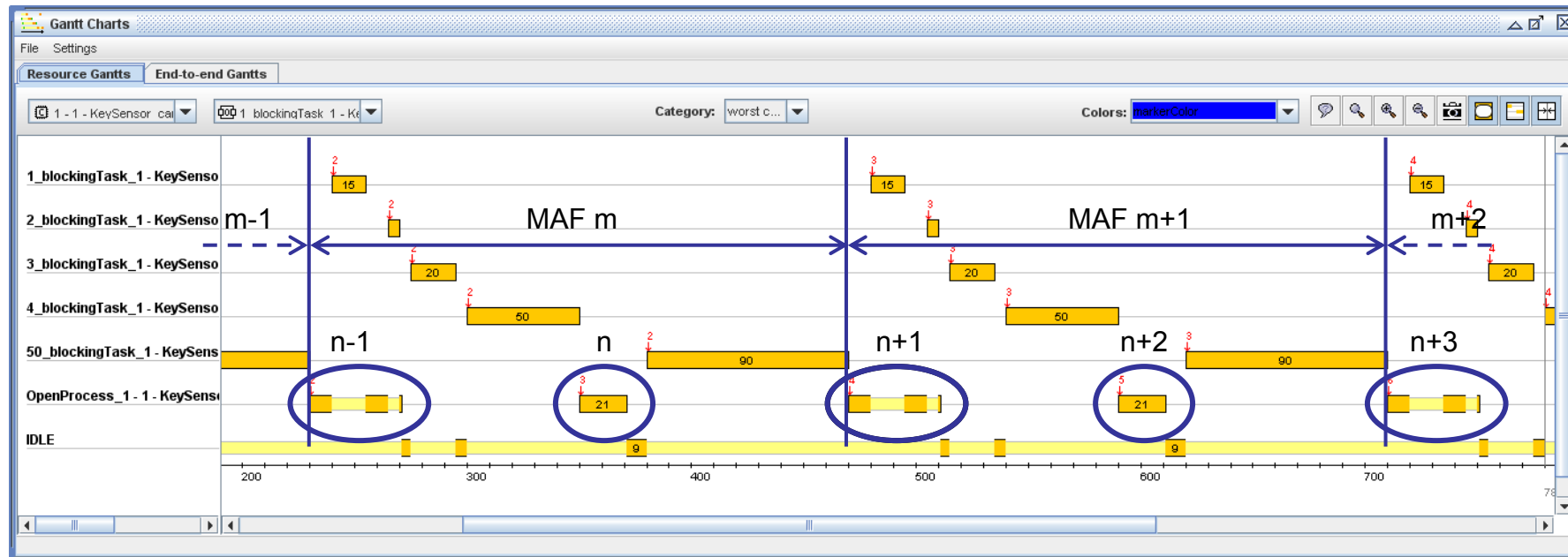
➔ **hierarchical scheduling: SP over TDMA**

➔ **capacity verification requires scheduling analysis**

➔ **must also fit to Major Frame Layout → Optimization**

# Preemptions of Process in Time Partition

- 2 partition executions in 1 MAF, not “symmetric”
- → two different execution timings



- can only be answered with knowledge about MAF and processes

Thank You!



**SYMITA VISION**

