

# Real-time Analysis and Design Working Group

## - Requirements vs. RFP topics -

Johannes Ernst, Aviatis Corp.  
Nov. 10th, 1998

	RF	1 - Time Modeling Issues	2 - Schedulability Issues	3 - Modeling End-to-End Behavior	4 - Modeling of Resources	5 - Non-time-related aspects of QoS Modeling	6 - Non-time-related aspects of RT Messaging Models	7 - "OS" Services	9 - Complex systems architectures	10 - Model Semantics - RTF?
Requirements	P									
<b>Real-time messaging</b>										
specify basic message arrival patterns (incl. periodic, aperiodic, episodic)	B						X			
model common IPC paradigms (e.g. synchronous (balking, waiting, simple), asynchronous, timed, distributed/local)	B						X			
<b>Time semantics and related</b>										
models of time	A	X								
time resolution	A	X								
deadlines	A	X	X	X	(X)			X		
specify execution times	A	X	X	X	X					
step-to-step in sequences (i.e. causal sequences in state machines)	A									
end-to-end times in sequences	A	X	X	X						
message response (causal rel's)	B						X			
response times	A	X	X	X						
time, concurrency, deadlines	A	X	X	X			X			
period and jitter	A	X	X	X						
interarrival time, its statistical properties (distribution type, min, max, ...)	A	X	X	X			X			
explicit timer objects	A, B							X		
timing estimates, budgets, actuals incl. exec.	A	X	X	X	X					
memory estimates, budgets, actuals	A		X		X					

RTF?

bandwidth estimates, budgets, actuals	A		X		X					
information permitting safety, liveness, etc. analysis	B			X	X	X	x			
<b>QoS</b>										
performance/throughput	A	X	X	X	X					
availability	B				X	X				
reliability & failure models	B				X	X	X			
priority/precedence (incl. dynamic/fixed)	A		X	X						
cost	B					X				
safety	B					X				
utilization	A		X	X	X	X				
<b>Physical resources</b>										
computing hardware (e.g. memory, processors, busses, comm. HW)	B				X			X		
other hardware elements (e.g. sensor, actuator, motor, analog devices ...)	B				X					
physical components with physical interfaces and physical attributes	B				X					
detailed HW/SW interface incl. memory maps, I/O, devices, IRQs ...	B				X			X		
relationship between software elements and resources	A		X		X			X		
<b>Software Infrastructure</b>										
concurrency units (e.g. processes, threads)	A		X		X			X		
run-time services (comms, memory allocation, software loading, restart/reset, logging, databases, ...)	B				X			X		
messages queues	A		X		X		X	X		
RT(OS) primitives	B							X		
RT(OS) kernels incl. startup models, failure detection/handling/recovery models	B			X		X		X		
software loading mechanisms	B							X		
synchronization mechanisms (e.g. locks)	A		X					X		
interrupts and IR-handlers and their characteristics (e.g. priority, masking ...)	A		X		X			X		
<b>schedulers and scheduling policies</b>										
resource-consuming entities (e.g. thread, message), schedulable unit	A		X		X					

"binding" of schedulable units to resources	A		X		X					
schedulers (as controllable entities)	A		X		X					
scheduling policies (RMS and other scheduling strategies)	A		X		X					
<b>complex systems architectures</b>										
large-granularity objects (e.g. subsystems)	C								X	
modeling software layers	C								X	
dynamic structures incl. mobile architectures	C								X	
interaction protocols	B			X			X		X	
end-to-end flows for distributed real-time systems (incl. visual representation, paths through system elements)	A, C				X					X
<b>model semantics</b>										
display object states over physical time	A				X					X
event priority on statecharts	D									X
fix control flow section in UML	D									X
precise definition of execution semantics	D									X
precise definition of processing semantics	D									X
execution sequence triggered from an event	D									X
clear definition of state machine inheritance	D									X