
# MARTE and Analysis – What is One without the Other?

## Architecture Assurance Through Analysis

WWW.TRIPAC.COM

June 26, 2008

# First – this is NOT an RMA Tutorial

- Don’t everyone clap at once!
- This is also not a Wonder Cure for MARTE – No Kidding!
- So why are we here?
  - To ask for Enlightenment? Maybe…
  - To beg for Users to want to do MARTE? Maybe…
  - To tell users that Analysis for UML exists for the Umpteenth time? Absolutely!

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# More Questions Without Answers - Maybe

- Wasn't SPT "suppose to" provide MARTE-like capabilities?
- Why Did SPT Fail miserably?
- Why should MARTE succeed where SPT failed?
- Do Users and Tool Vendors want MARTE?
- All Good questions with potentially differing answers

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# IMHO

- SPT was a great first crack at adding RT fields to UML
- Since OMG cannot dictate or “require” implementation specifics it was poorly adopted.
- Have things changed to make MARTE different?
- We hope so… Why? Because more users and vendors involved in the process.

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# What Are These Things Without the Other?

- In without Out
- Up without Down
- No without Yes
- Laugh without Cry
- Abbott without Costello
- …… AND …….

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Finally …What Is …?

- MARTE without Analysis
  - Documentation
  - No validation
  - Similar to the SPT profile – Who used it?
  - What is the impetus to use MARTE?

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# So Why Use MARTE?

- Adds many new fields describing Real-Time and Embedded architectures
- Provide Designers with templates for collecting necessary Data for RT Systems
- But Most Important -- include these data fields as First Class Objects in UML Modeling
- Do NOT Leave Defining these Fields until System Implementation

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# MARTE and Analysis – So What?

- IF …
  - Architects collect Timing and Resource data
  - Put it into the applicable MARTE fields
  - Run any type of “timing” or “schedulability” analysis
- Then ……

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# MARTE and Analysis – So What?

- Then…
  - Understand early on if there are timing or resource issues
  - Better understand the “real-time-ness” requirements prior to System Integration and Testing
  - Not leave any potential Architecture changes to the “testing” Phase
  - ☺ Maybe – Be on time and on budget for a change

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# What Type of Analysis Are We Talking About?

- Many Types of Analysis Can be Done
- Rate Monotonic Analysis (RMA) is the type TPSI has worked with and developed since 1989.
- Why RMA?
- Doesn't it only work for Single Processor architectures? - NO
- It doesn't address resource contention does it? – YES
- And NO this is STILL not an RMA Tutorial ☺

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# Understanding Timing Performance Through Method-Relevant Analysis

- **Technology base:**
  - Rate Monotonic Analyses for Predictable Worst Case Response Times rather than traditional simulation based- average response statistics
  - Traditional discrete event simulation can only generate probabilistic timing characterization (not worst case).

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# Understanding Timing Performance Through Method-Relevant Analysis

- **Benefits:**
  - "Margin Analysis" - how much more computation can be added until the performance goals cannot be met
  - Understanding of effective CPU utilization INCLUDING blocking and contention overhead timing
  - Early detection of possible architecture flaws

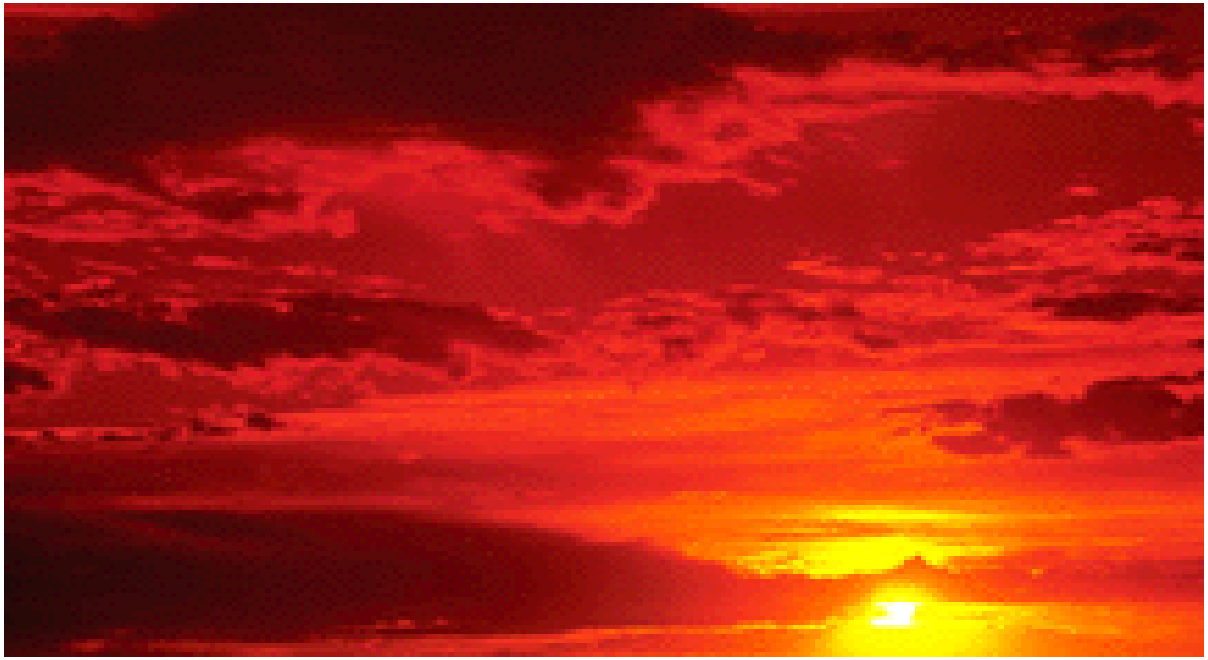
TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# RapidRMA

- Scheduling Analysis Workbench
  - Create application model
  - Analyze worst-case timing behavior
  - Represent hardware, network, and software configurations
  - Examine “what if” scenarios

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Pretty Photo



TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# RapidRMA benefits

- Allows cost effective testing and modeling of systems
- Guarantees schedulability under “worst-case” conditions
- Isolates and identifies timing problems
- Reduces development time and cost
- Provides “what if” support to identify bottlenecks and performance issues
- Provides “worst-case” schedulability analysis

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Task / Resource Model

- The Task / Resource model is an expression of the key characteristics of a real-time system for performing schedulability analysis.
  - Timing values
  - Overhead values
  - Relationships
  - Precedence

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

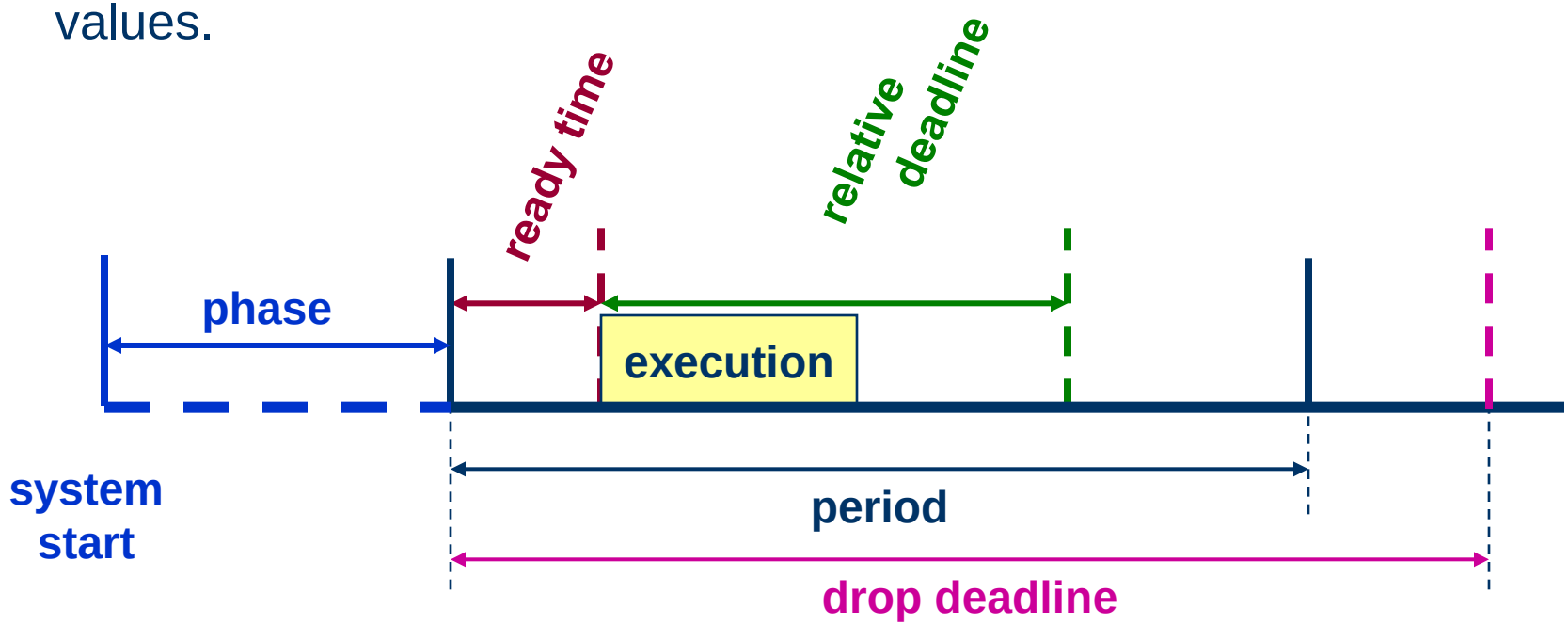
# What Are Tasks?

- A task is the basic unit of execution.
  - A fragment of code on a processor.
  - A message on a communication link.
  - A message across a backplane.
- A task is allocated to an active resource.
  - Processor
  - Communication link
  - Bus
- Attributes
  - Priority
  - Deadline type (hard or soft)
  - Instances
- Refined timing values

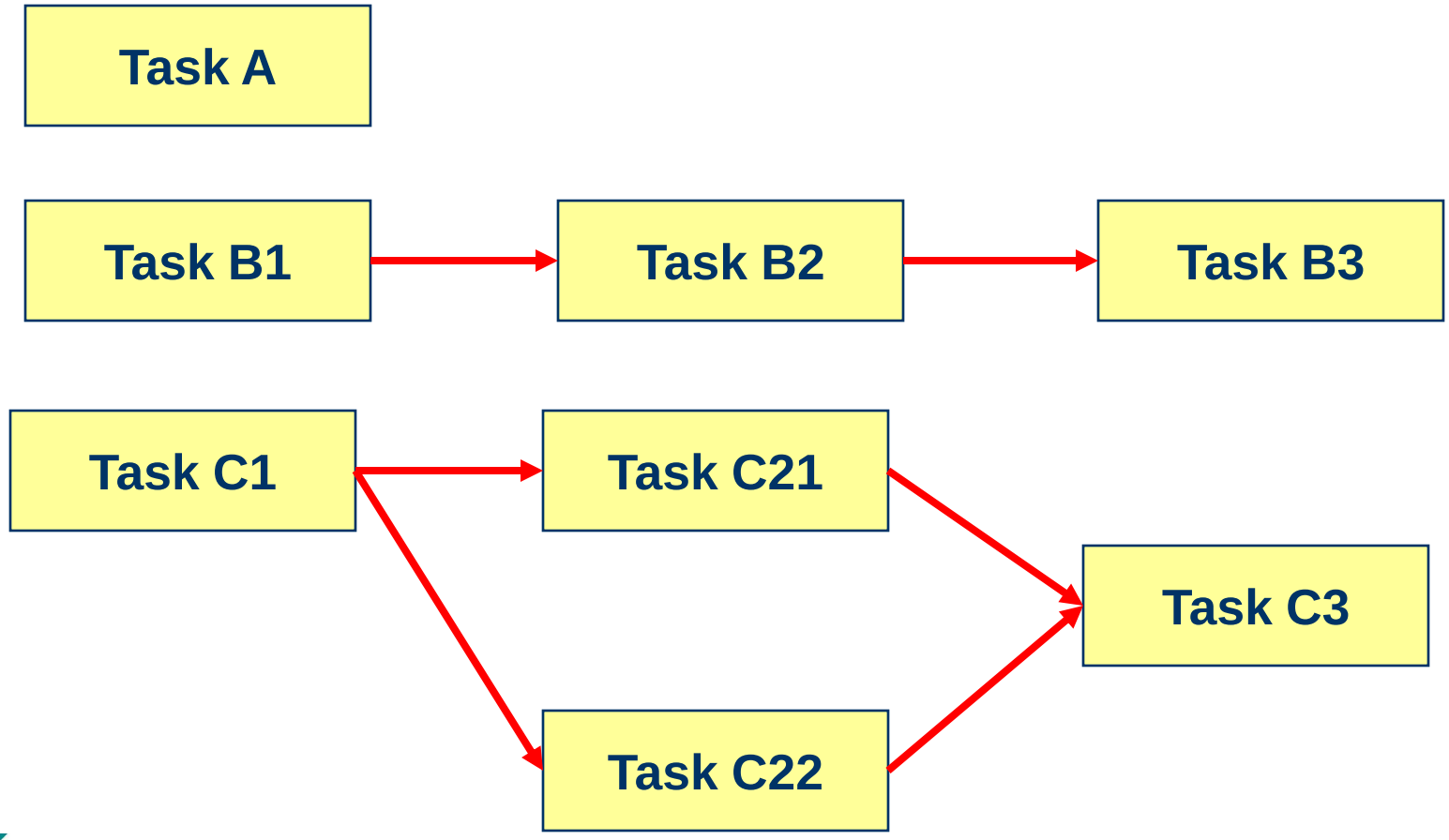
TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Task Timing Values

- Tasks have a more refined timing value.
- An individual value can be either *deterministic* or a *distribution* of values.



# Task Precedence Relationships

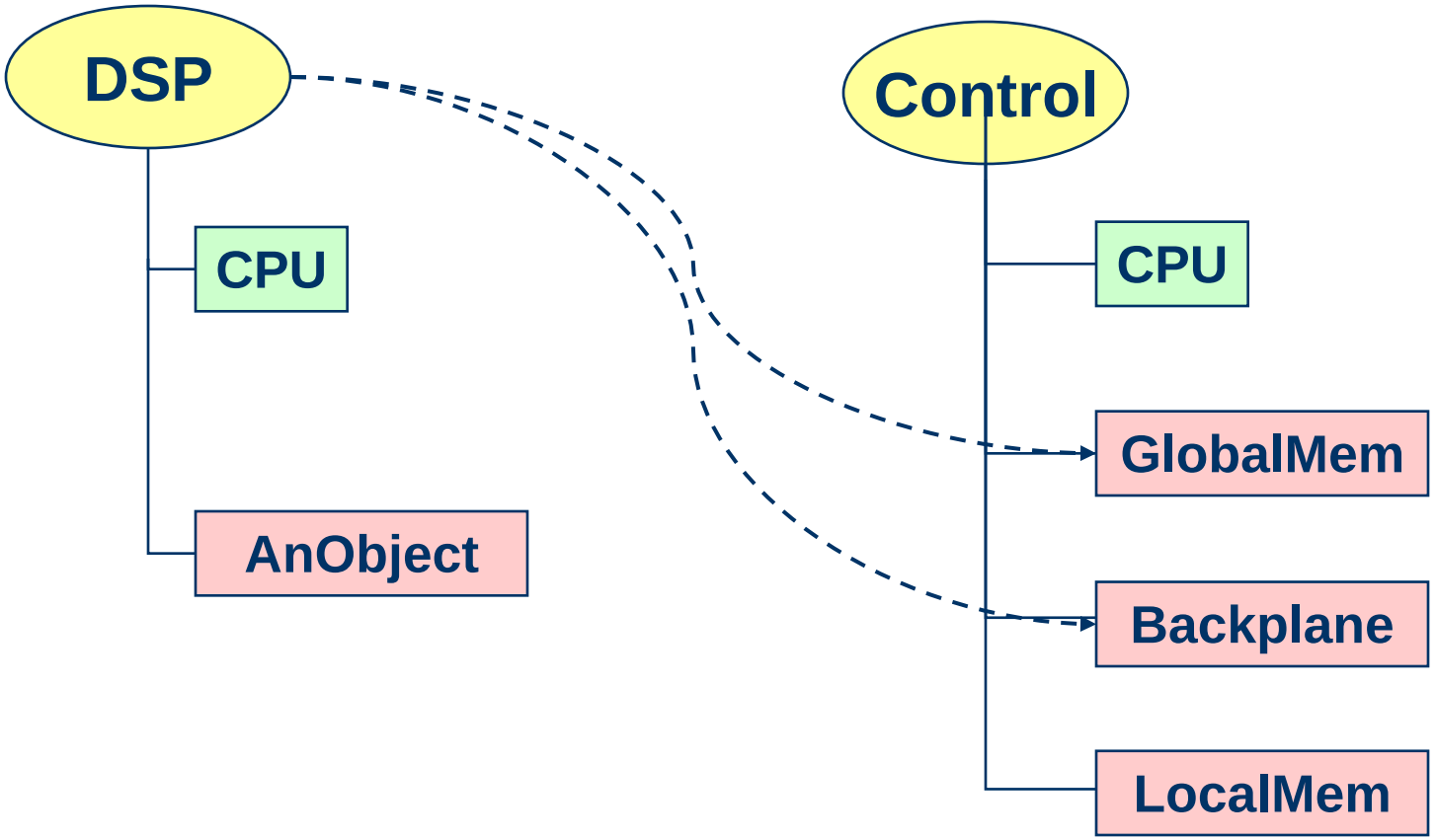


# Resources

- A resource, or a shared resource.
  - Implies exclusive use by one or more tasks
  - Can be either preemptable or non-preemptable
  - Can be either active or passive
- Timing values for active resources
  - Processing rate
  - Context switch time
  - Preemption
- Timing values for passive resources
  - Acquisition and deacquisition time
  - Preemption
  - Capacity
- Structure
  - Nodes that own resources

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Resource Graph




TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Analysis Tools

- Schedule simulation
  - Execution patterns
  - Processor usage
  - Resource usage
  - Scheduling events
- Suggestions for:
  - Adjusting execution
  - Adjusting period
  - Adjusting processor speed
  - Adjusting context switch
- Time demand graph
- Task allocation to processors
  - Best fit
  - Worst fit
  - Next fit
  - First fit
- Sensitivity analysis
  - Reduce blocking
  - Identify critical time values
- Periodic server simulation
  - Queueing simulation
  - Simple to complex context

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# Analysis Results

- Schedulable (y/n)
- Worst-case response time
- End-to-end worst-case response time
- Priority
- Spare capacity
- Blocking time
- Sources of blocking
- Preemption time
- Utilization
  - Task
  - Processor
  - Resource

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# Example of Bad Design Allocation of Tasks to Resources

- Example “stolen” from very wise and very old colleague (and no it is not Ben Watson ☺ )
- Family of four designing a house
  - Mother, Father, teenage son and daughter
  - Mother and father work Monday through Friday out of the house
  - Teenage son and daughter going through high school and all those issues ☹

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# Example of Bad Design Allocation of Tasks to Resources

- Build a 3 bedroom and one bathroom house after assessing their needs
  - Mother and Father will share one bedroom and son and daughter will have their own rooms.
  - Decide One bathroom is enough as they add up each of their usages of the “resource” and it is less then 2 hours per day.
- They build the house but are very unhappy after one week- Why?

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# Example of Bad Design Allocation of Tasks to Resources

- Poor analysis of the “resource” called “bathroom”
- It is a “shared” resource and “sharing” was not taken into consideration during design.
- Mother and Father get up 7 AM each day and need to:
  - Take a shower
  - Brush their teeth
  - Dry and comb hair
  - Put on appropriate personal hygiene products
  - Approximate time needed by each 25 minutes (50 minutes total)

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# Example of Bad Design Allocation of Tasks to Resources

- The daughter gets up 15 minutes later at 7:15 AM and needs to do all of these things but take an additional 10 minutes of “teenage” time to complete her morning. Total Time 35 minutes
- The son gets up even later at 7:30 AM and is a little faster then his sister but still take 30 minutes to get ready.
- If all 4 family members add up their “resource” usage it comes to less then 2 hours ( 115 minutes out of the possible 120 allowed).
- So this is within their “design” parameters when they made their initial needs assessment.

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Example of Bad Design Allocation of Tasks to Resources

- BUT ---- The deadline for all of them to be out of the house and on to work or school is 8:15 AM at the latest.
- So if their patterns hold true to form “someone” will always be late for leaving the house.
- When doing scheduling of “any” tasks that use “resources” more then just “total time” needs to be addressed to insure “schedulability”.

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Is This Hard?

- No this is HARD
- Climbing the Alps is Hard.

- Collecting Timing data and modeling it is Not Hard.
- Maybe not a picnic but not hard.

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

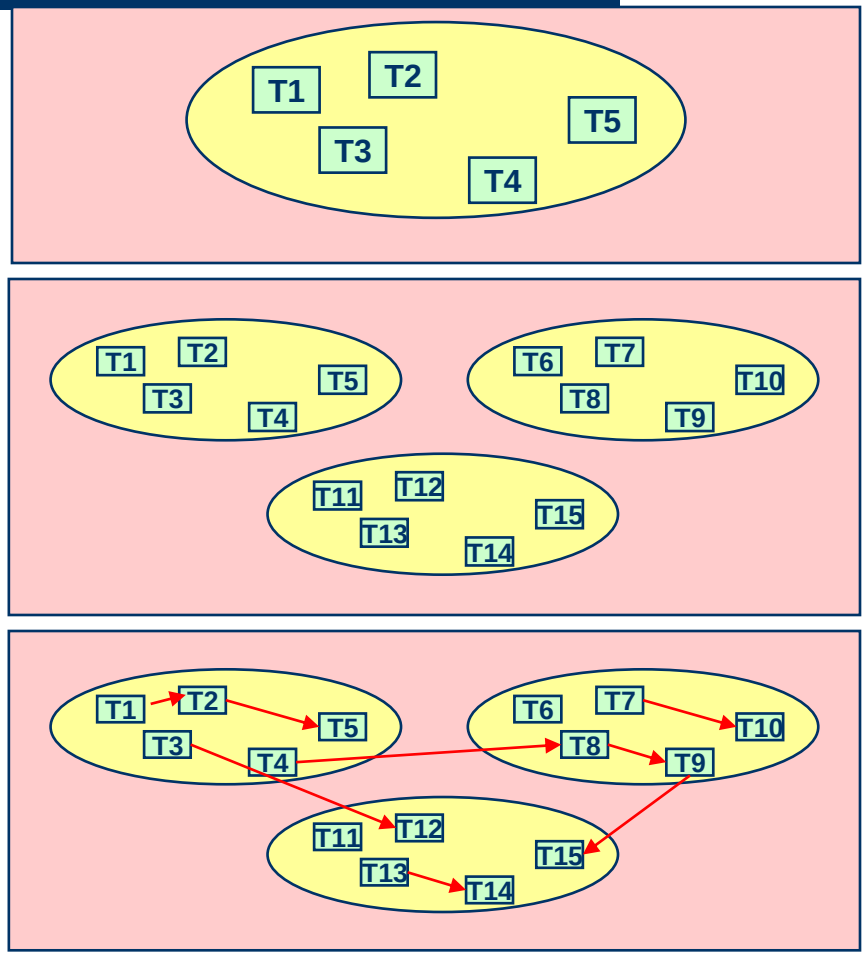
# Distributed Systems

- A distributed system is one where one or more processors communicate across a physical communications channel
- Schedulability is not simply a case of scheduling each processor separately
- There are some unique issues regarding schedulability analysis for distributed systems
  - Sharing the communication channel
  - Predictability for the communication channel
  - Sharing resources (physical or logical) across processor boundaries

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

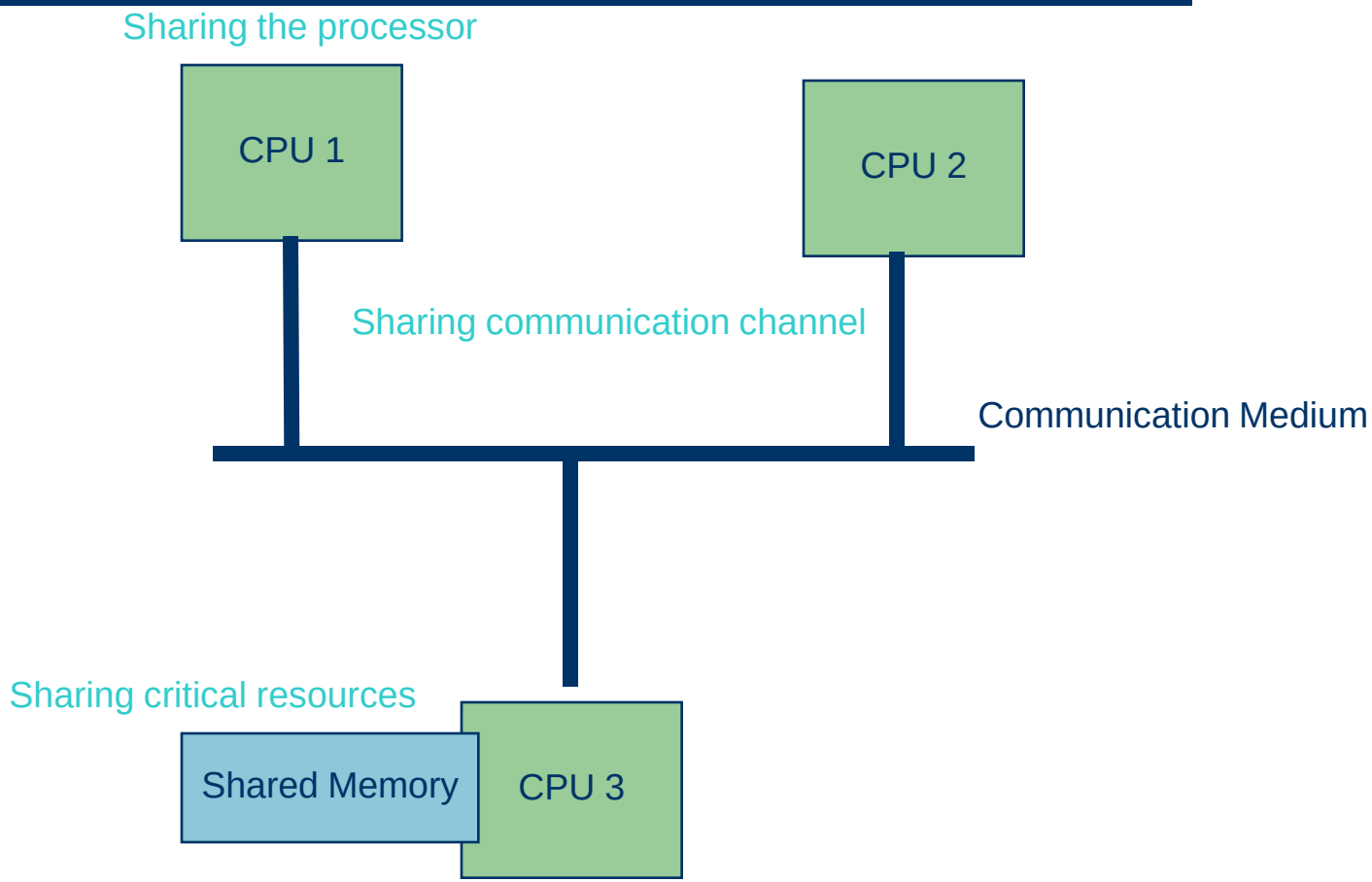
# Multi-node Analysis Algorithms

- Single Node:
  - Single processor with shared resources
- Multiple Node:
  - Multiple instances of single node analysis
  - Shared global resources
- End-to-end:
  - Single or multiple node
  - Precedence constraints
  - Successor task delayed by worst-case completion of predecessor

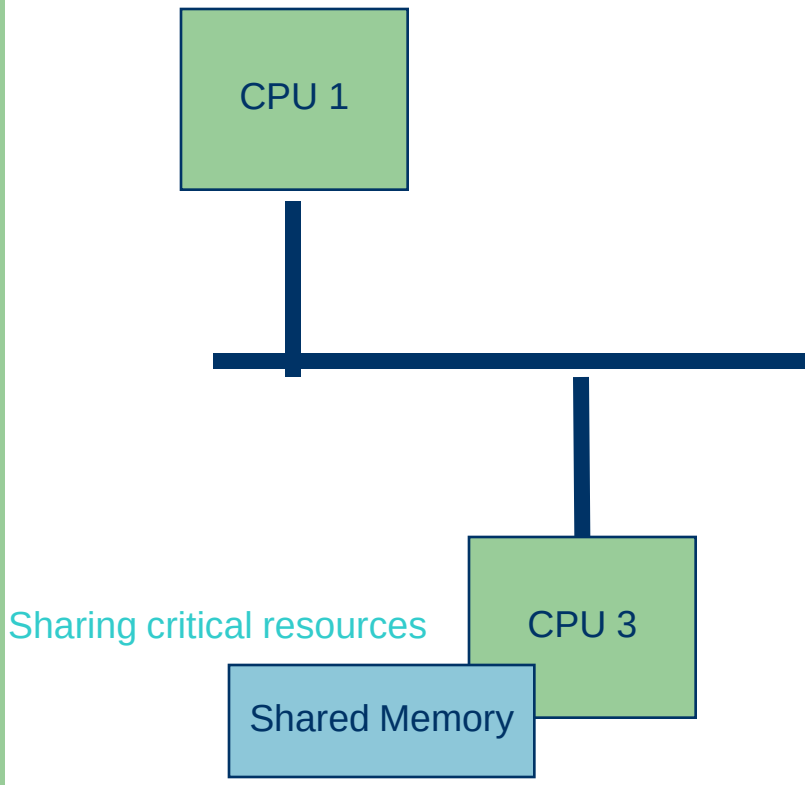



TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Distributed System Example



# Sharing Critical Resources

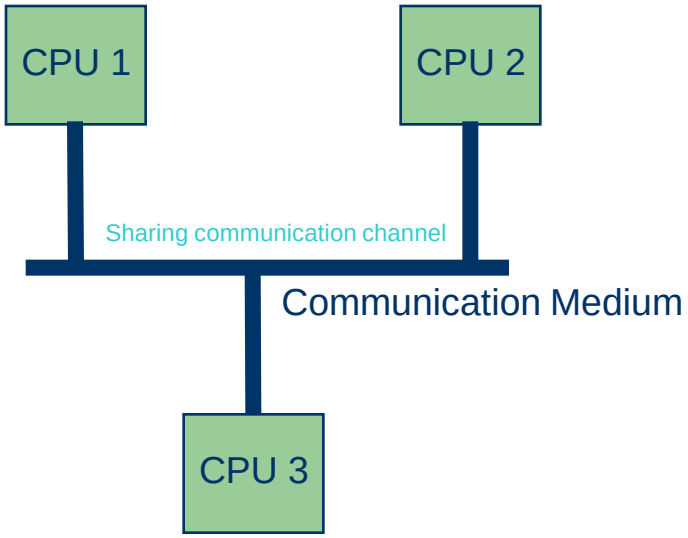



- Scenario: Task on CPU 1 requires exclusive use of shared memory on CPU 3
- Problem: Unbounded blocking from other tasks using shared memory
- Solution: Extend the resource sharing protocol to global resources

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# Sharing the Communication Channel

- Scenario: Tasks running on CPU 1 and CPU 2 need to access the communications medium
- Problem: The channel does not behave well
  - Serial access
    - Predictable access
    - Unpredictable access
  - Unpredictable delays
- Solution(s):
  - Simplifying assumptions
  - End-to-end models
  - Explore multiple scenarios




TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

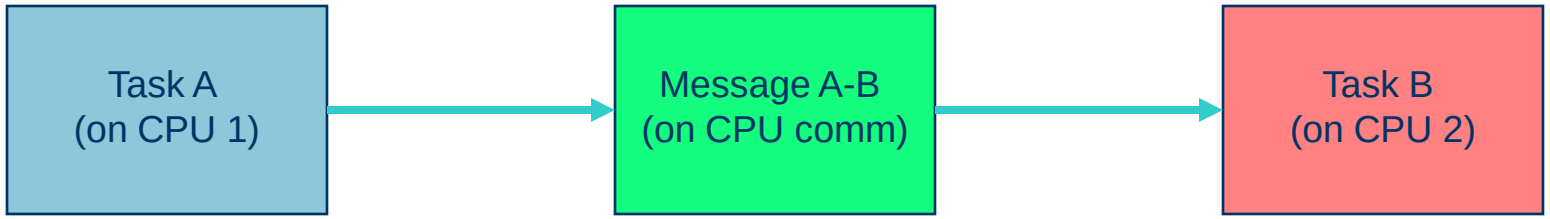
# Simplifying Assumptions

- It may be adequate to postulate a performance value for the channel
  - Dedicated path
  - More predictable implementations
- Treat the channel delays as part of the system overhead
- Permits you to examine the overall system correctness over a range of plausible channel values
- Scheduling Distributed Systems can be Hard but we sometimes treat it as easy.

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

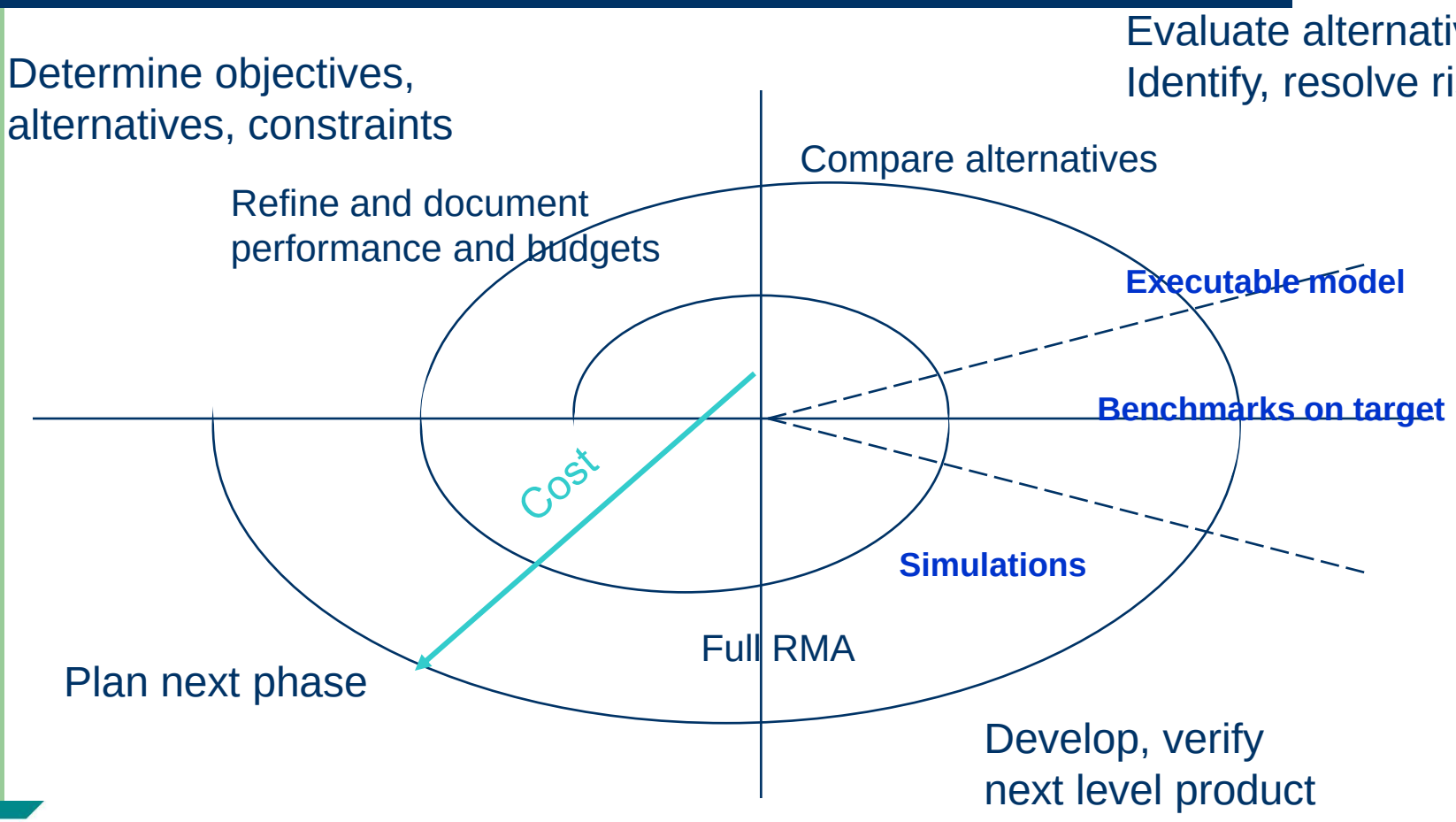
# End-to-End Models

- Model the channel as a “CPU” (serial resource)
  - Each message is a task
  - The execution time is proportional to the message length
  - Periodicity and deadline are inherited from the initiating task




TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Where Does RMA Fit?



# Defect costs

| Relative defect cost (src TRW) | |
|---|---|
| Requirements | 1 to 3 |
| Design | 3 to 7 |
| Coding | 10 |
| Test Development | 13 to 20 |
| Acceptance Test | 30 to 70 |
| Operation | 40 to 400 |

Sun Microsystems estimates that, during operations, the cost of a defect doubles every 3 months until it is fixed.

TRI-PACIFIC SOFTWARE, INC. Engineering Solutions

# So What Has Been Done For MARTE?

- Thales, CEA and INRIA created a “bridge” to RapidRMA from IBM/Rational RSA tool.
- The “bridge” is done through eclipse and with some “magic” it allows a user to:
  - Create a MARTE- aware model
  - Pass the data to RapidRMA
  - Run analysis on the MARTE model

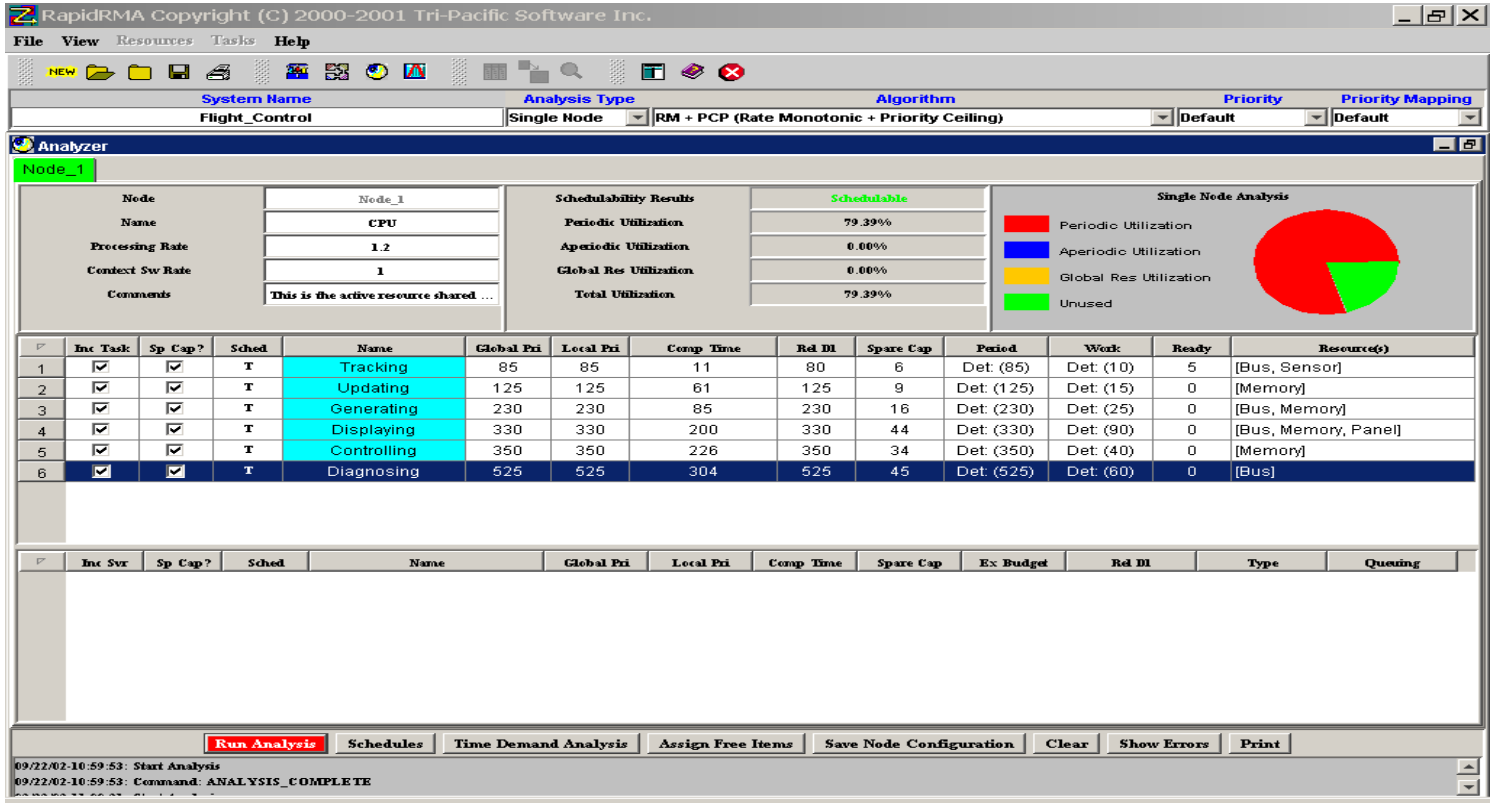
TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# So What Has Been Done For MARTE?

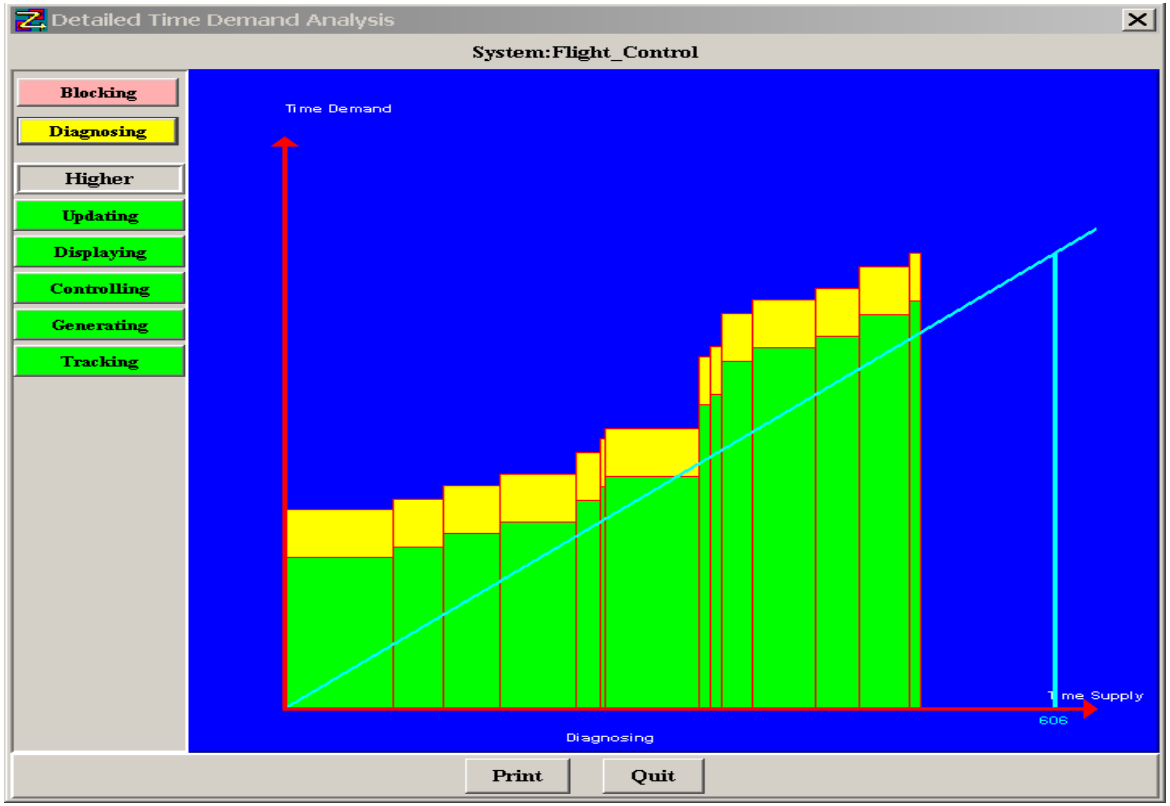
- Thales, CEA and INRIA will be putting this "bridge" into Open Source.
- Tri-Pacific will work with potential users to push this technology forward to use with RapidRMA.. But potential users need to step forward sooner then later ☺
- The only "bridge" currently is the one for IBM/Rational RSA.
- Demonstrations of this will be coordinated with interested parties individually by contacting sales@tripac.com.
- Any additional "bridges" will be looked at and evaluated for potential new work.

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# The Ubiquitous Sample RapidRMA Analysis Screen



# Sample RapidRMA Task Utilization Screen




TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Summary

- **Applications:**
  - Performance Critical, Mission Critical, Safety Critical deployed systems where timing failure results in unacceptable harm.
  - Any Resource constrained architecture. These include but are not limited to Cell phone applications, Medical Devices, Automotive and most other embedded application.

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Summary

- Create “bridges” to the leading design tools from RapidRMA through Eclipse
- Annotate UML diagrams to include both REQUIRED performance (deadlines) and SUPPLIED performance (Computation and resource usage times) in the MARTE fields.
- Then Perform Rate Monotonic Analysis (RMA) and evaluate worst case responses and resource margins.
- New Users Swarm to buy new MARTE solutions – Maybe ☺

TRI-PACIFIC SOFTWARE, INC.
Engineering Solutions

# Questions?


Peter Kortmann
peter@tripac.com
www.tripac.com