Modelling real-time applications based on resource reservation

Laura Barros, César Cuevas, Patricia López Martínez, José M. Drake and Michael González Harbour

Grupo de Computadores y Tiempo Real
Universidad de Cantabria, Spain
{barrosl, cuevasce, lopezpa, drakej, mgh @unican.es}

2nd International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems
Porto, July 5th, 2011

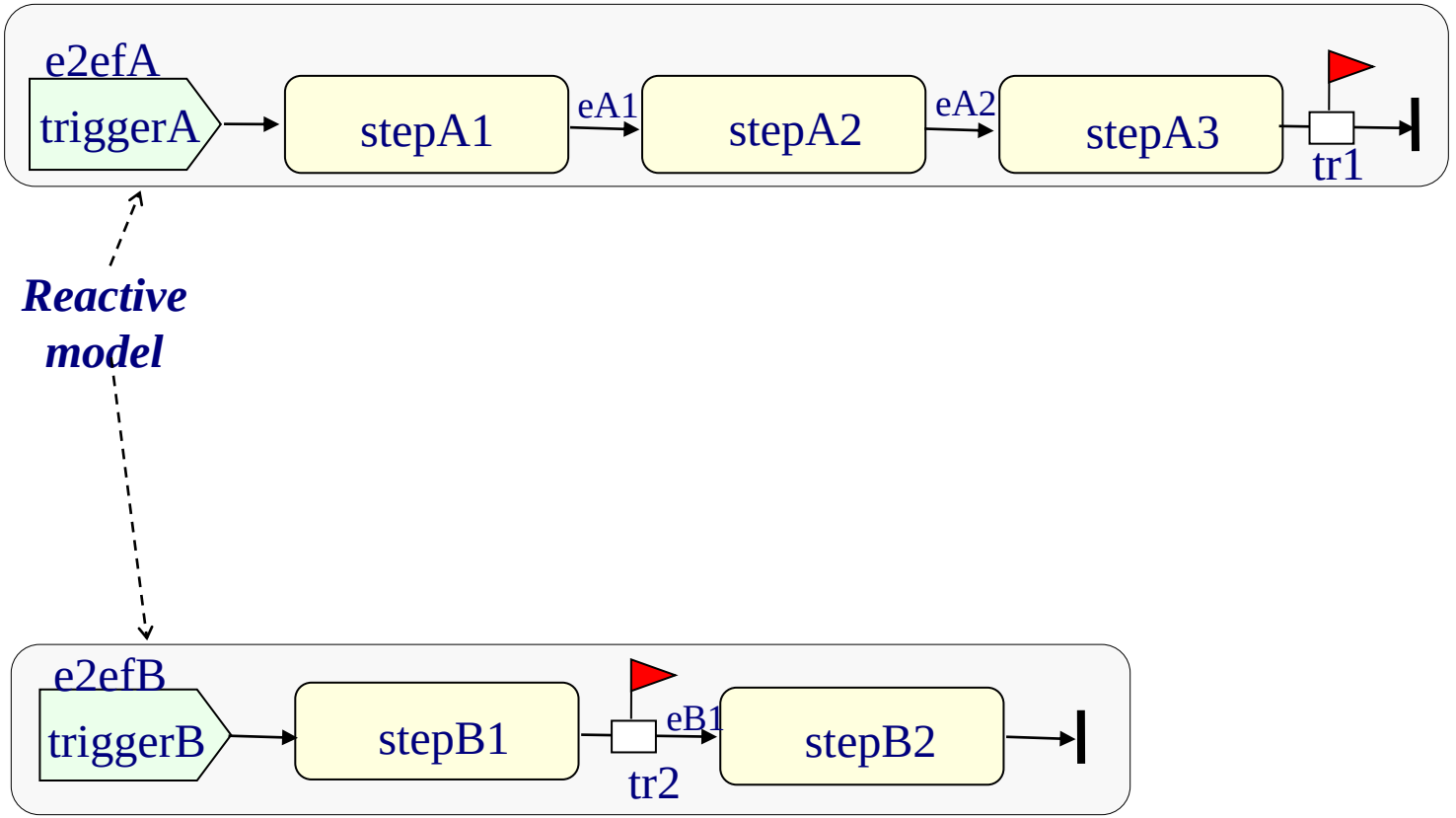
Funded by the EU under contracts FP7/NoE/214373, and by the Spanish Government under grant TIN2008-06766-C03-03

Introduction

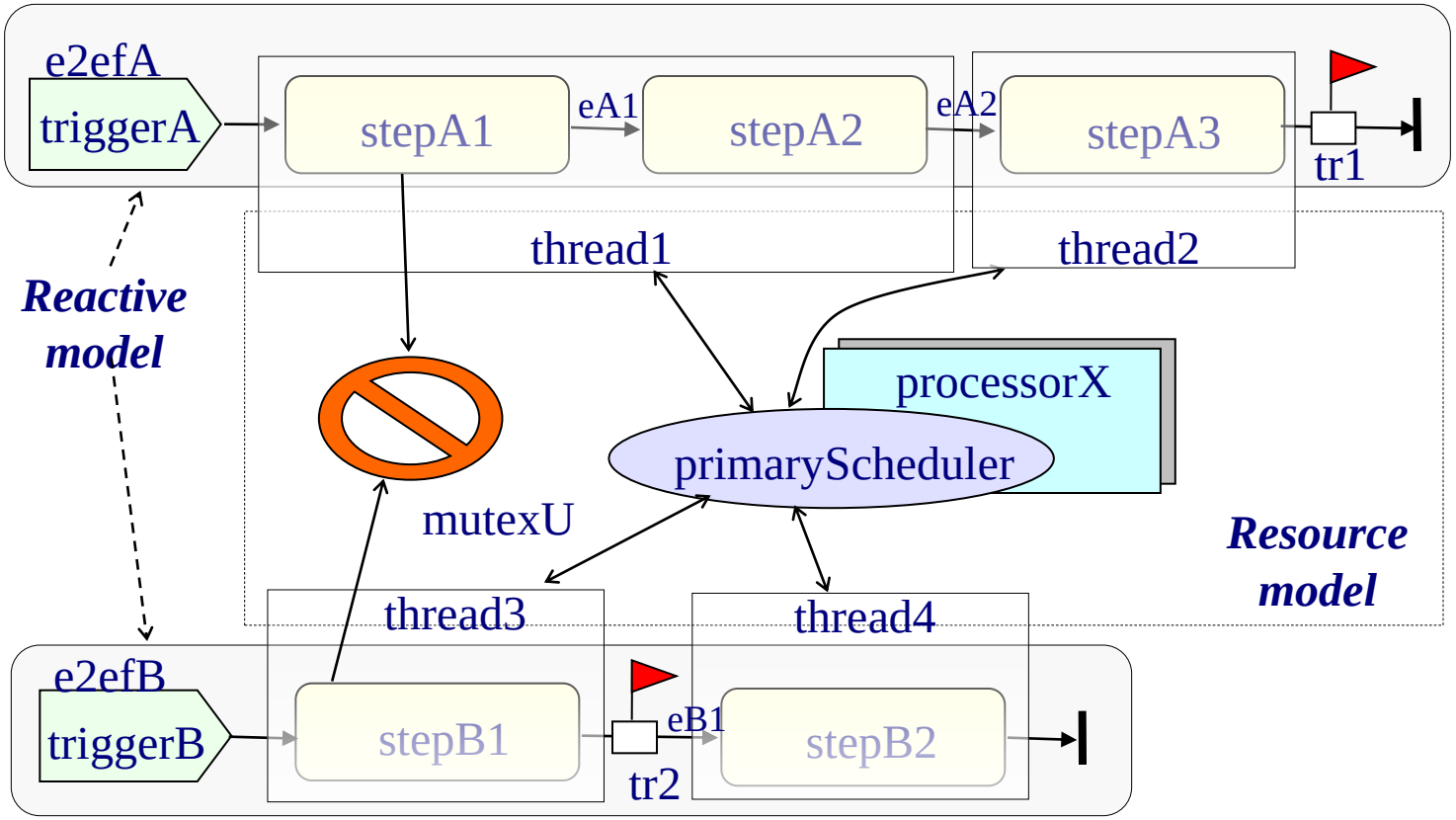
- Resource Reservation (RR)
  - Executing each system thread or communication session in a server
  - Server: it has assigned a fraction of the processor capacity or the communication network.
  - Advantages:
    - System robustness
    - Design simplicity
    - Reusability of software components

- MAST
  - Open source set of tools to design and analysis of RT applications
  - MAST *model* extended =>MAST 2
  - MAST 2 *tools* under development:
    - Possibility of usage of MAST 1 tools transforming models by MDA

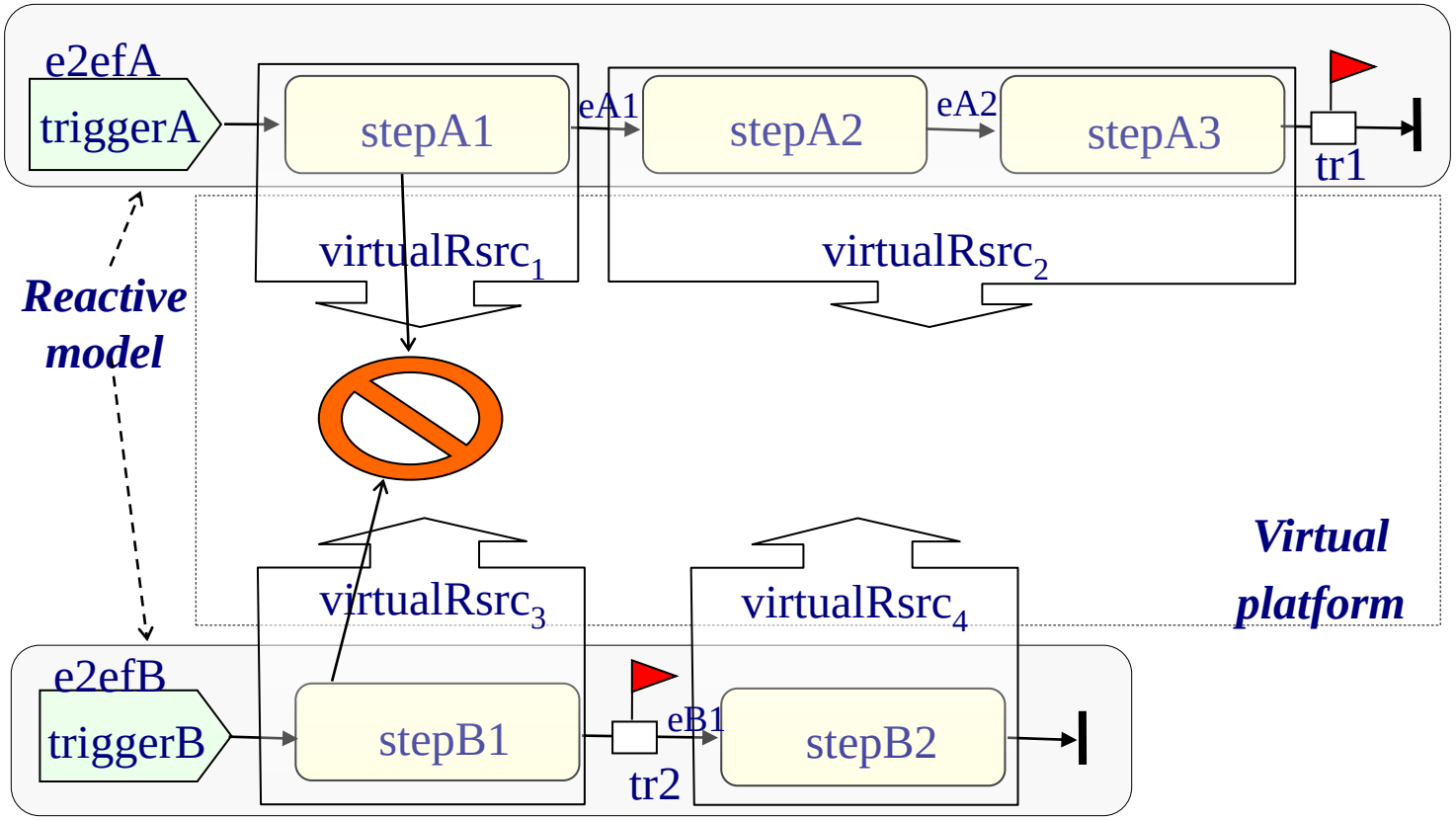
Modelling elements for the resource reservation paradigm I



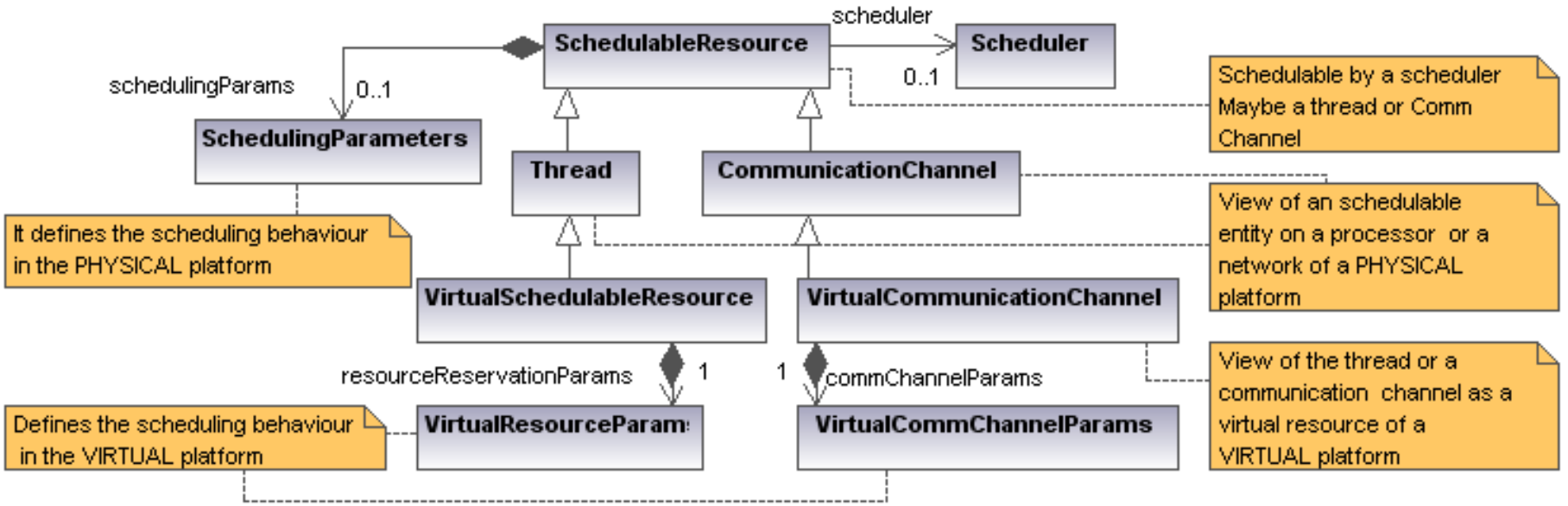
Modelling elements for the resource reservation paradigm I



Modelling elements for the resource reservation paradigm II

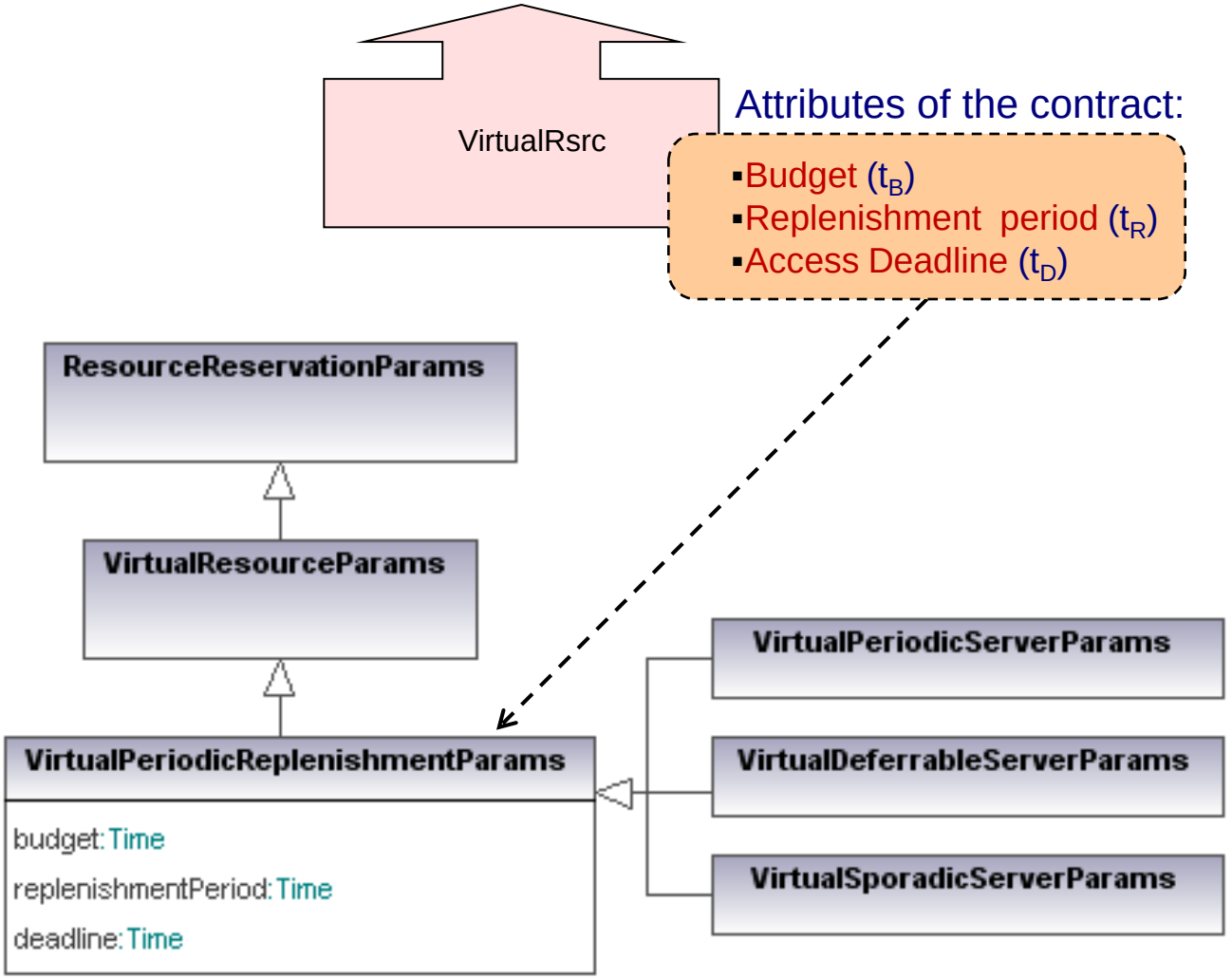


New classes in resource reservation MAST models I

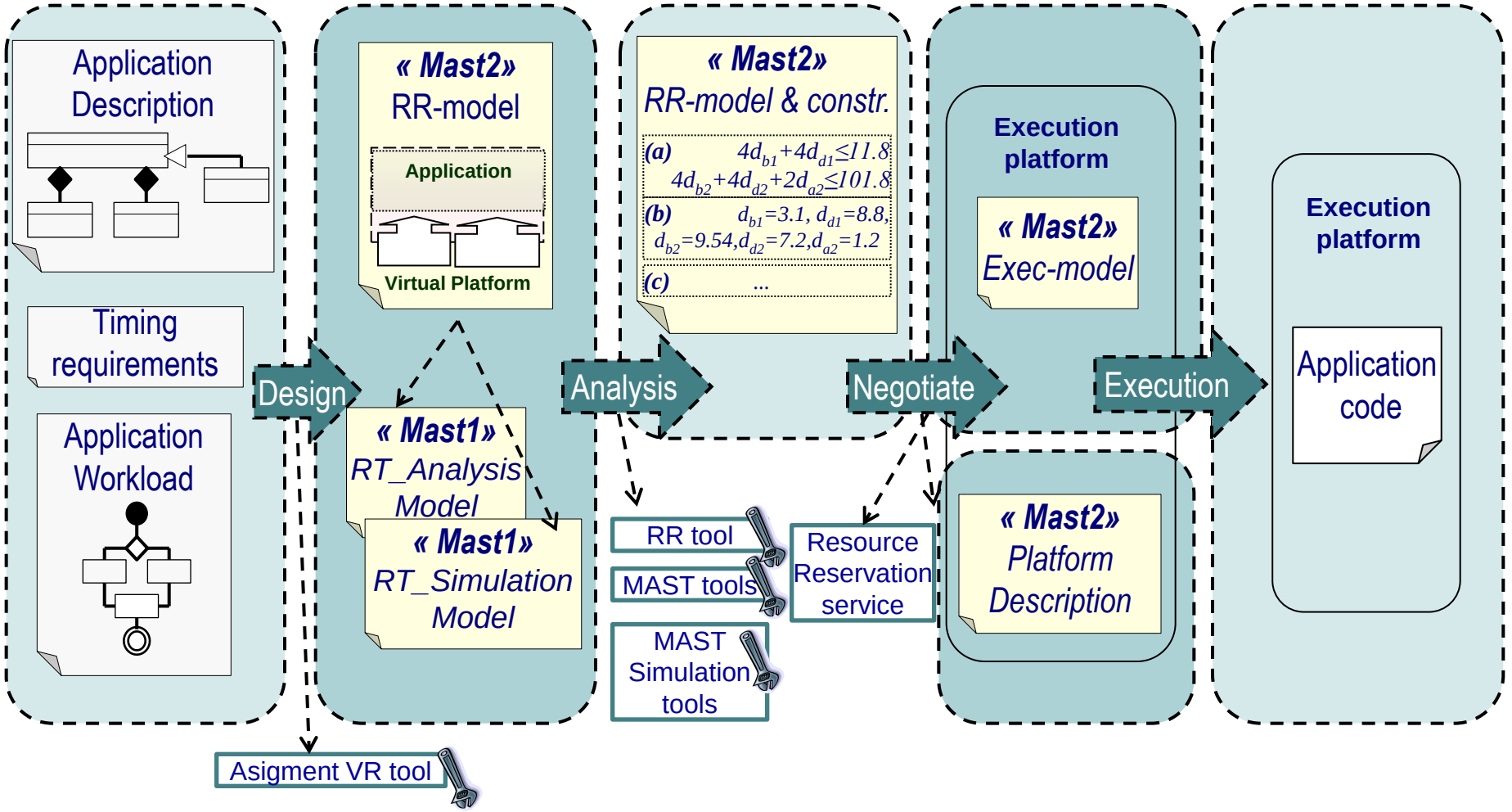


New classes in resource reservation MAST models II

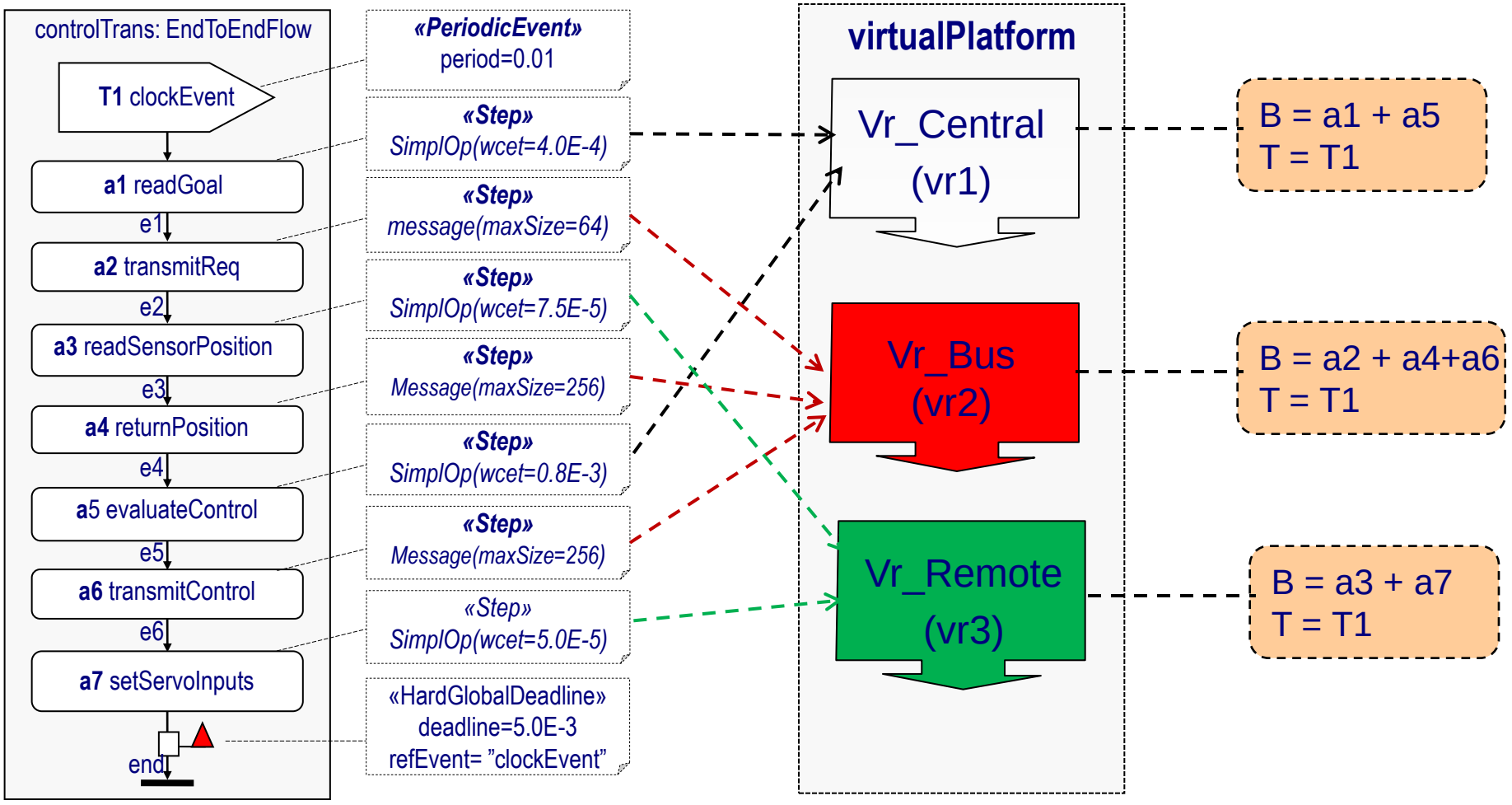
- **Contract:** it represents the capacity required by the application to be executed.



Real-time application development



Reactive model of ServoControl example



Virtual Platform assignment tool

Virtual Platform

	Period	Budget	Deadline
Vr_Central(vr1)	T1	a1+ a5	?
Vr_Bus(vr2)	T1	a2+a4+a6	?
Vr_Remote(vr3)	T1	a3+a7	?

$2\ \mathbf{vr1}.t_{D+}\ 3\ \mathbf{vr2}.t_{D+}\ 2\ \mathbf{vr3}.t_{D} \leq 9.182$ ms

It is possible to calculate the *wrt* (t_X) of an activity (t_a) in a VirtualRsrc.

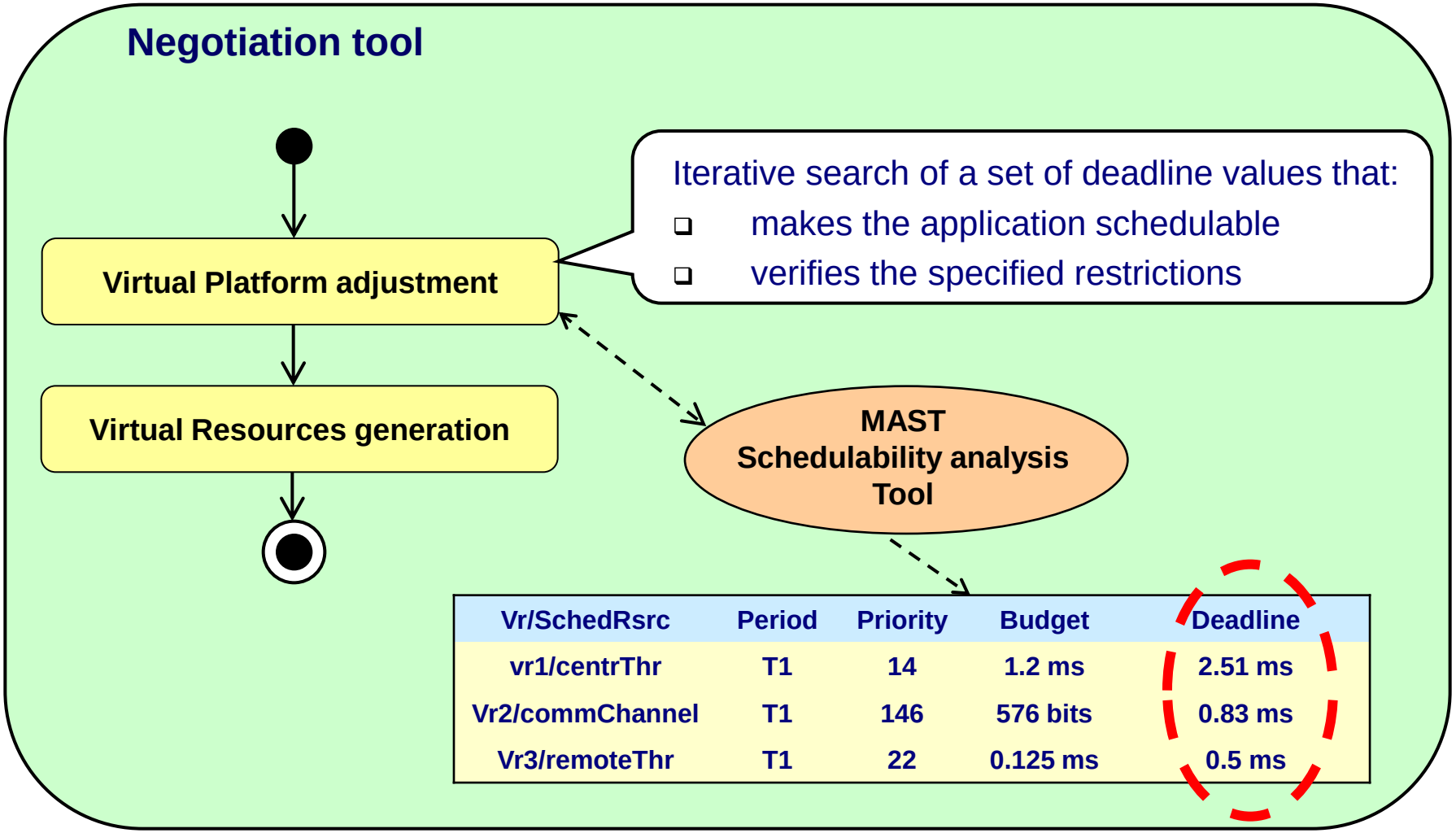
$wrt\ (a_x) < \mathbf{vr}.t_D - (\mathbf{vr}.t_B - a_x)$

$\mathbf{vr1}.t_D\text{-}(\mathbf{vr1}.t_B\text{-}a1) + \mathbf{vr1}.t_D\text{-}(\mathbf{vr1}.t_B\text{-}a5) +$

$+ \mathbf{vr2}.t_D\text{-}(\mathbf{vr2}.t_B\text{-}a2) + \mathbf{vr2}.t_D\text{-}(\mathbf{vr2}.t_B\text{-}a4) + \mathbf{vr2}.t_D\text{-}(\mathbf{vr2}.t_B\text{-}a6) +$

$+ \mathbf{vr3}.t_D\text{-}(\mathbf{vr3}.t_B\text{-}a3) + \mathbf{vr3}.t_D\text{-}(\mathbf{vr3}.t_B\text{-}a7) < t_{GD}$

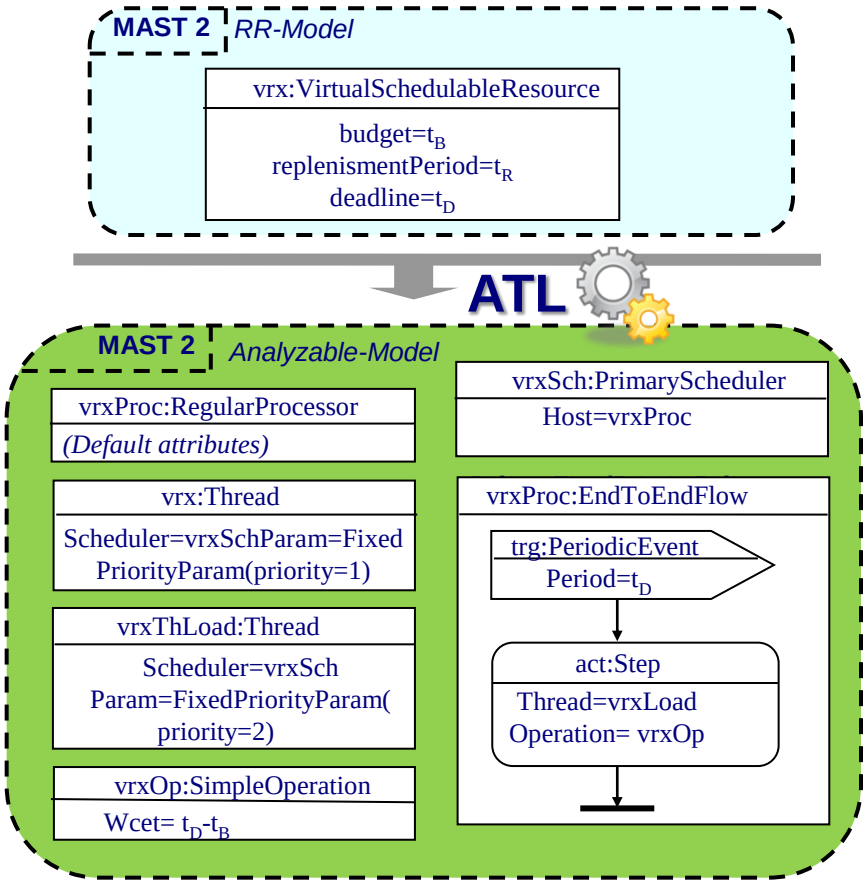
Negotiation tool



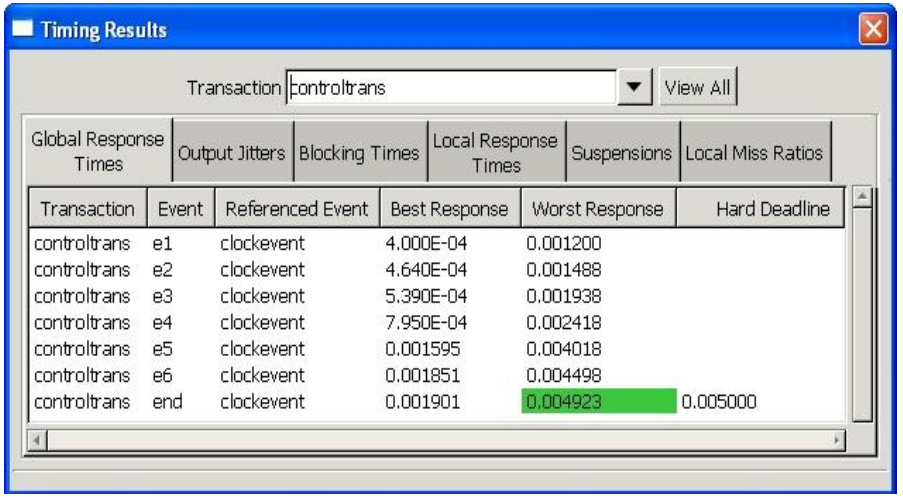
Vr/SchedRsrc	Period	Priority	Budget	Deadline
vr1/centrThr	T1	14	1.2 ms	2.51 ms
Vr2/commChannel	T1	146	576 bits	0.83 ms
Vr3/remoteThr	T1	22	0.125 ms	0.5 ms

Complex application: No restrictions. Virtual schedulability analysis

Model transformation



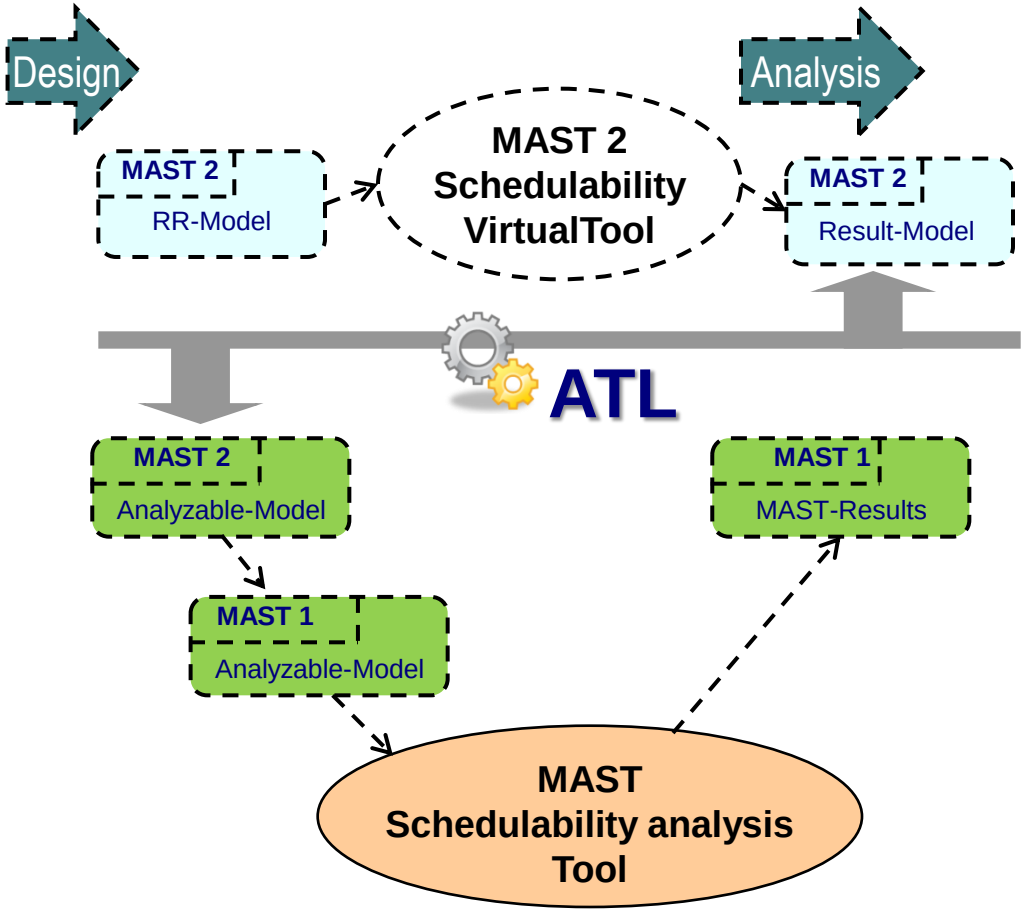
Results of analysis using MAST tools



Transaction	Event	Referenced Event	Best Response	Worst Response	Hard Deadline
controltrans	e1	clockevent	4.000E-04	0.001200	
controltrans	e2	clockevent	4.640E-04	0.001488	
controltrans	e3	clockevent	5.390E-04	0.001938	
controltrans	e4	clockevent	7.950E-04	0.002418	
controltrans	e5	clockevent	0.001595	0.004018	
controltrans	e6	clockevent	0.001851	0.004498	
controltrans	end	clockevent	0.001901	0.004923	0.005000

Laura Barros, César Cuevas, Patricia López, J.M. Drake, M.G.Harbour

Model Transformation for the MAST 1 compatibility



Conclusions

- Modelling elements based on RR paradigm with MAST:
  - Scenario:
    - The application can be analysed independently of the current workload
    - The programmers do not require any knowledge about the underlying platform
  - Solution:
    - MAST tools cover the different phases in the development and execution of applications based on resource reservation
    - Relying in the availability of a resource reservation middleware installed in the platform
  - Currently working on:
    - Updating the MAST tools in order to support the new advanced paradigms for real time systems covered by MAST2
    - Other :
      - Implementation of the virtual platform assignment tool
      - Implementation of the resource reservation service based on Rt-linux