### MBSE 2020 TASTE: a toolchain for multicore TSP applications

Authors:

Laura Gouveia GMV, lasequeiragouveia@gmv.com

Maxime Perrotin ESA, <u>maxime.perrotin@esa.int</u>

Thanassis Tsiodras ESA, <u>thanassis.tsiodras@esa.int</u>

Jérôme Hugues CMU/SEI, jhugues@andrew.cmu.edu

**Daniel Silveira** *GMV, daniel.silveira@gmv.com*  Presenter:

Laura Gouveia GMV, <u>lasequeiragouveia@gmv.com</u>



- Overview
  - TASTE
  - AIR
- Deploying TSP application on TASTE
- TASTE improvements:
  - AADL
  - Code generation
  - Build system
- EagleEye use case
- Future work



# OVErview



### **What is TASTE**



### The ASSERT Set of Tools for Engineering

- TASTE is a development environment dedicated to embedded, real-time systems.
- Based on formal languages and the concept of building "correct by construction" software.



MBSE2020 – TASTE: a toolchain for multicore TSP applications

9/28/2020





- AIR is a type-1 hypervisor that guarantees **TSP** in the target hardware
- Allows multiple avionics applications with **different** criticalities to run independently in the same hardware resource

Prototyping

**Development** 





### SPĄRC

OS	ΙΟ
Bare	Ethernet
RTEMS5	MIL1553
RTEMS48i	SpaceWire
	CAN

ARM

OS	ΙΟ
Bare	TSN
RTEMS5	UART
	CAN



MBSE2020 – TASTE: a toolchain for multicore TSP applications

9/28/2020

LAS Normalized States of the second states of the s



- TSP applications are deployed as standard TASTE applications
- TSP properties added as extensions to TASTE process.







### **Interface View**

- Defines logical functions and interactions.
- Code skeletons are generated from the model.
- User specifies function behaviour:
  - Programming language C, C++, Ada or Micropython
  - Modelling language SDL or Simulink



### Interface View unchanged for TSP applications



### **TSP on TASTE Deploying TSP applications on** TASTE

### **Deployment View**

- Logical functions deployed ٠ on hardware.
- AADL library allows reusing • of hardware components.
- Configuration parameters • configure the system on target platform.





### **Deployment View features TSP specific entities and** attributes



### **Deployment View**

- Time partitioning:
  - Partition scheduling attributes within processor





### **Deployment View**

- Space partitioning:
  - Memory segments specified per partition.

Nor	del p/ocl part1	P	art2					
	nello	C				Edit Data		= 0 ×
			<u>N</u> ode Attributes <u>M</u>	emory	<u>D</u> escri	ption		
			В	ase Ad	dress		S	ze
		L	part1 1				1	Bytes
	part3	-	part2 1				1	Bytes
	puico		part3 1				1	Bytes
	Function1		part4 1				1	Bytes
			<b>a</b>					
Ľ		-4		_				· · · · · · · · · · · · · · · · · · ·
					0k	Apply	Cancel	



### **Deployment View**

• Criticality level per partition.





### **TSP on TASTE Deploying TSP applications on** TASTE

### **Concurrency View**

- Automatic model transformation from Interface View and • Deployment View.
- Provides scheduling analysis. •



### **Concurrency View**

- SMP properties can de defined:
  - Task allocation to core.
  - Task priority.
  - Stack size.

		Edit real tir	ne properties		
Thread properties Thr	ead Placement				
Name	Dispatch Pro	tocol Per	iod Priori	ty Dispatch Off	set Stack Size
Parti bello s	sporadic	100		Ams	5000 bytes
Part1 Poll Port	periodic	100	ns A	Ams	Abytes
Part2 r2 h2	periodic	100	Ams 3	Ams	5000 bytes
Part2 read h	periodic	100	Ams 1	Ams	5000 bytes
Part2 read new	sporadic	100 10m	5 4	Ams	50000 bytes
Part2.Poll Port	periodic	100	ns 0	Oms	Obvtes
Part3.function1_cvcl	periodic	100	Oms 4	Oms	5000 bytes
Part4.function2 t	sporadic	10m	s 1	Oms	50000 bytes
Part4.Poll Port	periodic	100	ns 0	Oms	Obvtes
					,
					I
		0k Ap	ply Can	cel	
L					L
		Folia and Ale			
		Edit real ti	ne properties		
」 hread properties Thr	ead Placement	Ealt real ti	ne properties		
」 hread properties Thr Name	ead Placement	r(s) Allow	d Processor	(c)	
hread properties Thr Name Part1 bello s	ead Placement	r(s) Allowe	d Processor	s)	art2 vp. podel procl. pa
J hread properties Thr Name Part1.hello_s Part1_Poll_Port	ead Placement	r(s) Allowe	d Processor procl.part1	s) vp,nodel_procl.pa	art2 vp,node1_proc1.pa
hread properties Thr Name Partl.hello_s Partl.Poll_Port Part2_r2_b2	ead Placement Actual Processo nodel_procl.par nodel_procl.par	r(s) Allowe t1_vp_node1_ t1_vp_node1_	d Processor( procl.part1 procl.part1	s) vp,nodel_procl.pa vp,nodel_procl.pa	art2_vp,node1_proc1.pa art2_vp,node1_proc1.pa
hread properties Thr Name Part1.hello_s Part1.Poll_Port Part2.r2_h2 Part2.rad_b	ead Placement Actual Processo nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowe tl_vp_nodel tl_vp_nodel t2_vp_nodel	d Processor( procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2_vp,node1_proc1.pa art2_vp,node1_proc1.pa art2_vp,node1_proc1.pa art2_vp,node1_proc1.pa
hread properties Thr Name Part1.hello_s Part1.Poll_Port Part2.r2_h2 Part2.read_h Part2.read_nov	ead Placement Actual Processo nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowe t1 vp node1 t1_vp node1 t2_vp node1 t2_vp node1	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2_vp,node1_proc1.pa art2_vp,node1_proc1.pa art2_vp,node1_proc1.pa art2_vp,node1_proc1.pa art2_vp,node1_proc1.pa
hread properties Thr Name Part1.hello_s Part1.Poll Port Part2.r2_h2 Part2.read_h Part2.read_new Part2.nel_Port	Actual Processo nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowe t1 vp node1 t1_vp node1 t2_vp node1 t2_vp node1 t2_vp node1	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp.nodel_procl.p;	art2 vp,nodel procl.pa art2 vp,nodel procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa
Name Part1.hello s Part1.hello s Part1.Poll_Port Part2.r2_h2 Part2.read_new Part2.read_new Part2.Poll_Port Part3_furction1_crut	ead Placement Actual Processo nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowe t1 vp node1 t1_vp node1 t2_vp node1 t2_vp node1 t2_vp node1 t2_vp node1 t2_vp node1	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2 vp, nodel procl.pa art2_vp, nodel procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa
Anread properties Thr Name Part1.hello_s Part1.Poll_Port Part2.r2_h2 Part2.read_h Part2.read_new Part2.read_new Part2.Poll_Port Part3.function1_cycl Part4_function2_t	ead Placement Actual Processo nodel procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowe t1 vp node1 t2 vp node1 t2 vp node1 t2 vp node1 t2 vp node1 t2 vp node1 t3 vp node1	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa
Name Part1.hello s Part1.hello s Part1.Poll Port Part2.read h Part2.read new Part2.read new Part2.Poll_Port Part3.function1_cycl Part4.function2_t Part4.function2_t	ead Placement Actual Processo nodel procl.par nodel procl.par nodel procl.par nodel procl.par nodel procl.par nodel procl.par nodel procl.par	r(s) Allower tl vp nodel tl vp nodel t2_vp nodel t2_vp nodel t2_vp nodel t2_vp nodel t3_vp nodel t3_vp nodel	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2 vp,nodel procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa
Ame Part1.hello s Part1.Poll Port Part2.r2.h2 Part2.read_new Part2.read_new Part2.read_new Part2.read_rew Part2.Poll_Port Part3.function1_cycl Part4.Poll_Port	ead Placement Actual Processo nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allower t1_vp_node1 t1_vp_node1 t2_vp_node1 t2_vp_node1 t2_vp_node1 t2_vp_node1 t2_vp_node1 t4_vp_node1 t4_vp_node1	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2 vp,nodel procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa
Name Part1.hello s Part1.hello s Part2.r2_h2 Part2.read_h Part2.read_new Part2.read_new Part2.Poll_Port Part4.function1_cycl Part4.function2_t Part4.Poll_Port	ead Placement Actual Processo nodel procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allower r(s) Allower t1 vp nodel t2 vp nodel t2 vp nodel t2 vp nodel t2 vp nodel t2 vp nodel t4 vp nodel	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2 vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa
Name Part1.hello s Part1.Poll Port Part2.r2_h2 Part2.read_h Part2.read_new Part2.read_new Part2.Poll_Port Part3.function1_cycl Part4.function2_t Part4.Poll_Port	ead Placement Actual Processo nodel procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowed t1 vp node1 t2 vp node1 t2 vp node1 t2 vp node1 t2 vp node1 t2 vp node1 t4 vp node1 t4 vp node1	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1_	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2 vp,nodel procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa
hread properties Thr Name Part1.hello s Part1.Poll Port Part2.rc2 h2 Part2.read h Part2.read new Part2.read_new Part2.read_new Part2.reat_new Part3.function1_cycl Part4.function2_t Part4.Poll_Port	ead Placement Actual Processo nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowed t1 vp nodel t1_vp nodel t2_vp nodel t2_vp nodel t2_vp nodel t3_vp nodel t3_vp nodel t4_vp nodel t4_vp nodel	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2 vp,nodel procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa
hread properties Thr Name Part1.hello s Part1.Poll Port Part2.r2_h2 Part2.read h Part2.read new Part2.Poll_Port Part3.function1_cycl Part4.function2_t Part4.Poll_Port	ead Placement Actual Processo nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allower r(s) Allower t1 vp nodel t2 vp nodel t2 vp nodel t2 vp nodel t2 vp nodel t4 vp nodel t4 vp nodel	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;	art2 vp, nodel procl.pa art2_vp, nodel procl.pa art2_vp, nodel procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa art2_vp, nodel_procl.pa
hread properties Thr Name Part1.hello s Part1.Poll Port Part2.read h Part2.read new Part2.read new Part2.Poll_Port Part3.function1_cycl Part4.function2_t Part4.Poll_Port	ead Placement Actual Processo nodel procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowe ti vp nodel ti vp nodel t2_vp nodel t2_vp nodel t2_vp nodel t2_vp nodel t3_vp nodel t4_vp nodel t4_vp nodel	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	<pre>is) vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p; vp,nodel_procl.p;</pre>	art2 vp,nodel procl.pa art2_vp,nodel procl.pa art2_vp,nodel procl.pa art2_vp,nodel procl.pa art2_vp,nodel procl.pa art2_vp,nodel procl.pa art2_vp,nodel procl.pa art2_vp,nodel procl.pa art2_vp,nodel procl.pa art2_vp,nodel procl.pa
hread properties Thr Name Part1.hello s Part1.Poll Port Part2.rc_h2 Part2.read h Part2.read new Part2.read_new Part2.Poll_Port Part3.function1_cycl Part4.function2_t Part4.Poll_Port	ead Placement Actual Processo nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par nodel_procl.par	r(s) Allowe ti vp nodel ti_vp nodel t2_vp nodel t2_vp nodel t2_vp nodel t2_vp nodel t4_vp nodel t4_vp nodel	d Processori procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1 procl.part1	s) yp,nodel_procl.p; yp,nodel_procl.p; yp,nodel_procl.p; yp,nodel_procl.p; yp,nodel_procl.p; yp,nodel_procl.p; yp,nodel_procl.p; yp,nodel_procl.p;	art2 vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa art2_vp,nodel_procl.pa



MBSE2020 – TASTE: a toolchain for multicore TSP applications

## 



Code generation and build system

### Code generation improvements

### **PolyORB-HI**

- Main execution platform in TASTE •
- Provides: ٠
  - Task handling
  - Data types •
  - Communication interfaces
  - Device drivers
- Upgraded to support: •
  - Latest RTEMS5
  - AIR hypervisor

### Kazoo

- Novel build system of TASTE.
- Flexible design based on templating engine.
- Generates Concurrency View and ensures orchestration with PolyORB-HI.
- Extended to deploy threads on TSP Partitions.



D S D  $\mathbf{O}$ S



### EAGLEEYE USE CASE MODEL







MBSE2020 – TASTE: a toolchain for multicore TSP applications







### Several EagleEye scheduling scenario have been drawn including partitions running in multiple cores with RTEMS 5 SMP.





MBSE2020 - TASTE: a toolchain for multicore TSP applications

9/28/2020

### 





- TASTE allows for multi-partition communicating systems, supported by AIR.
- Full TSP system prototyped by TASTE. •
- TASTE tools extended to support AIR as an execution ٠ platform.
- One step further from manual, error-prone development ٠ lifecycle.



### **Future work** Much to be done...

- Support for IO Partition. •
- Scheduling analysis for TSP systems. ٠
- Specification of processor core usage in multi-partition ٠ systems.





• Support for IO Partition.





### Auris De

THANK YOU

laura.gouveia@gmv.com



