Partitioned Model for Space Applications (IMA 4 space)

Ismael Ripoll , Vicente Brocal, Miguel Masmano

Santander, Feb, 2011

UNIVERSIDAD POLITECNICA DE VALENCIA

fentISS
FENT innovative Software Solutions

Contents

Introduction (i)

- Work done in the frame of the IMA for space initiative (CNES & ESA).
  - Apply the IMA development model to the space domain.
  - Use ARINC-653 (avionics) standard as starting point.
  - Include the space requirements in the IMA model.
- The goal is to assist the *integrator* to build a scheduling plan for a single CPU board.

Introduction (ii)

- The Xoncrete tool has been developed.
- We use the “agile” development paradigm:
  - Start from the user requirements.
  - Build a prototype.
    - Work on the model which better captures the user reqs.
    - If the model is not enough then change it.
  - The model must be:
    - Simple.
    - Capture the user needs.
    - Be able to generate a plan in bounded time.

Index

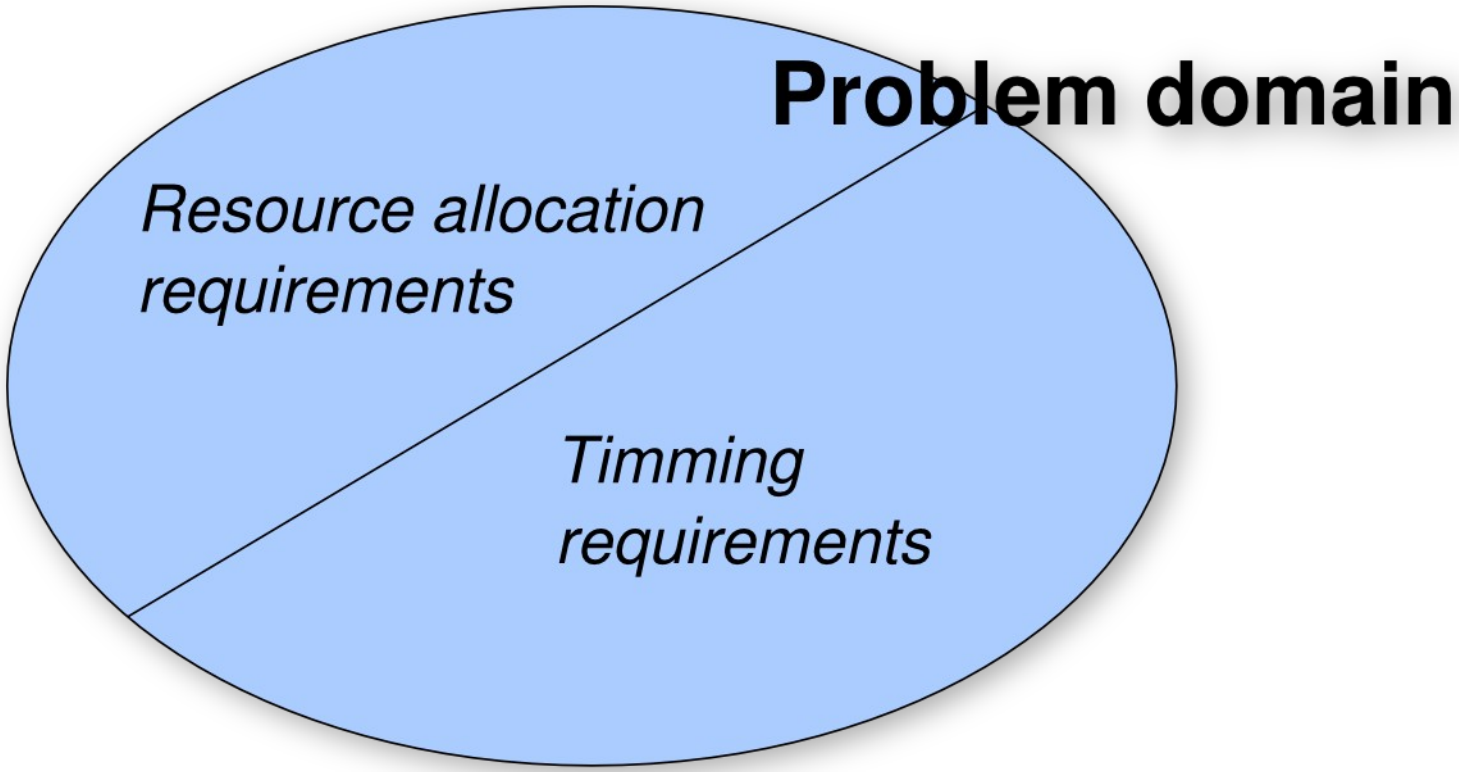
1. Introduction
2. **Reference platform**
3. Problem domain
4. Model
5. Open issues

TSP reference platform

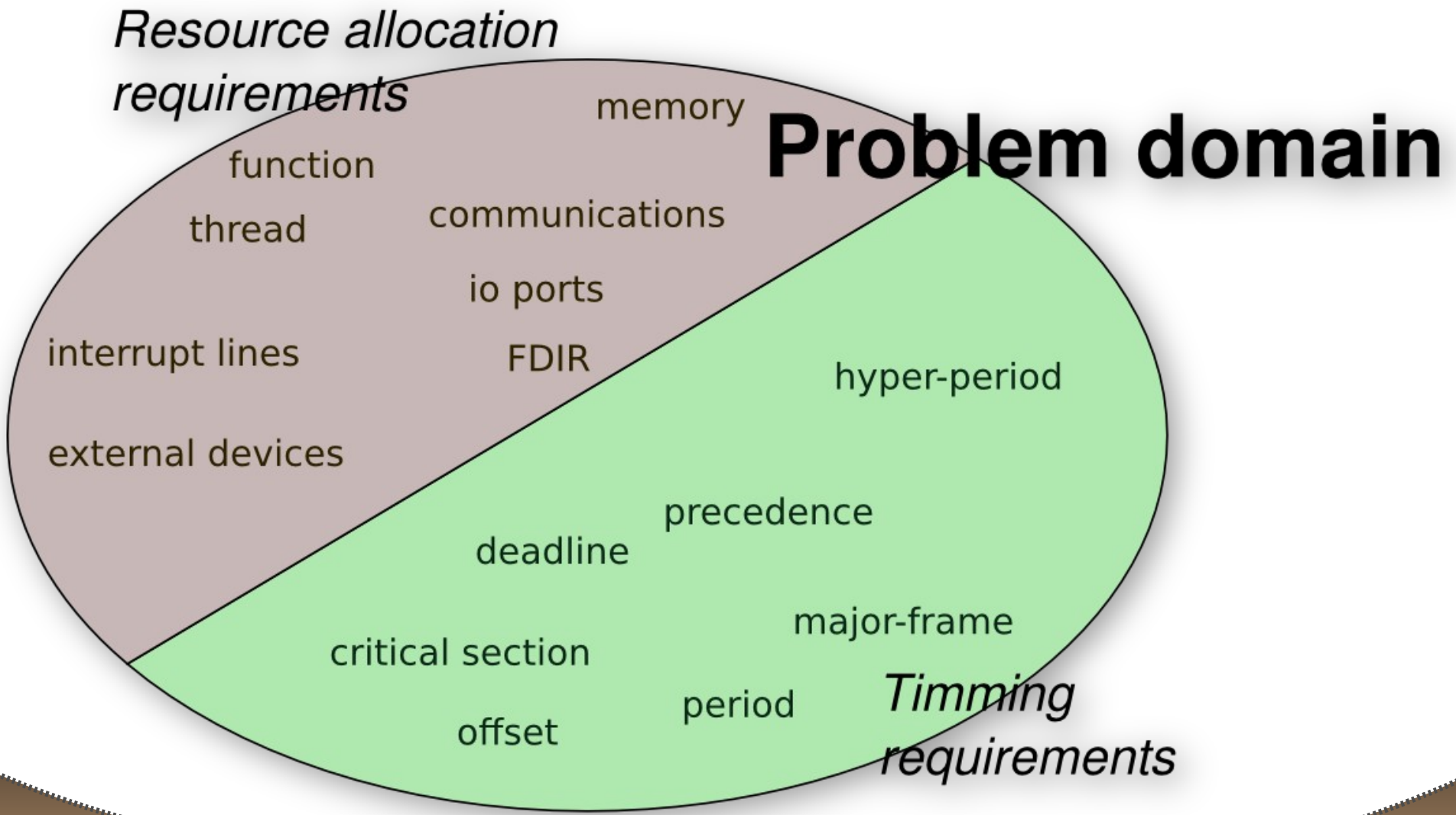
Contents

1. Introduction
2. Reference platform
3. **Problem domain**
4. Model
5. Open issues

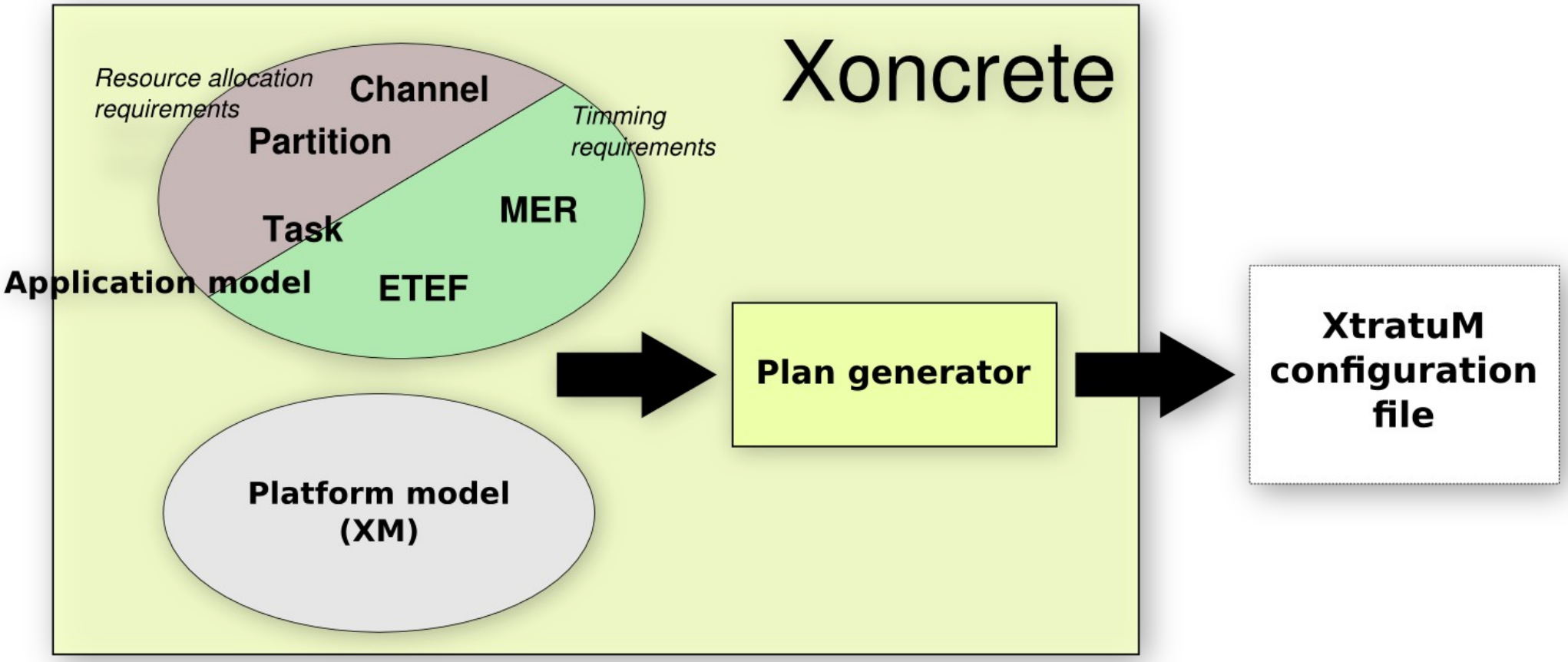
Problem domain(i)



Problem domain (ii)



Problem domain (iii)



Contents

Partitions

- Container of code and physical resources:
  - Tasks (threads, functions,...).
  - Irq lines.
  - Input/Output ports.
  - Communication ports (sampling/queuing).
  - Health monitoring configuration.
  - Memory areas.
  - Etc.

Tasks

- Schedulable unit (task, thread, process, function,...)
- By default, tasks are preemptable.
  - Optionally a task can be declared as “atomic”: non-preemptable.
- Tasks has only the following attributes:
  - WCET (Worst case execution time).
  - The set of mutual exclusion resources (MER)
    - *Cooldown* time to model devices that execute in parallel with the CPU.

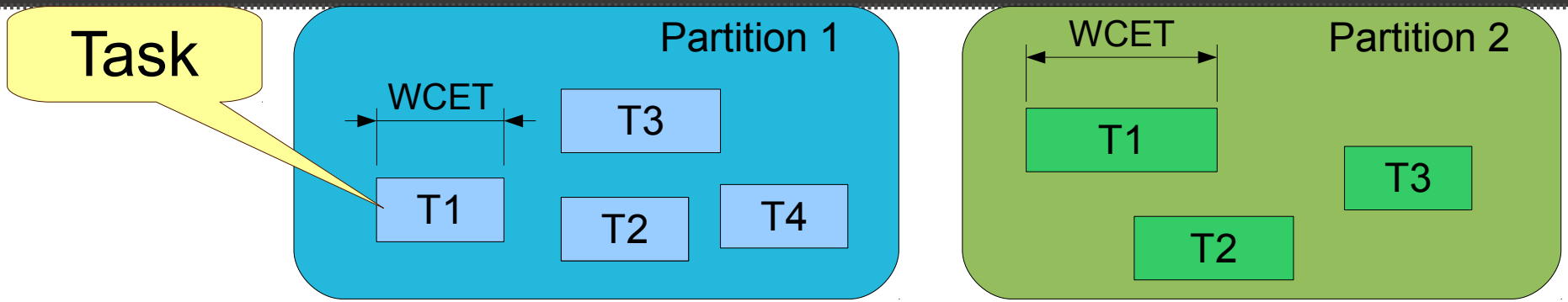
Mutual exclusion resources (MER)

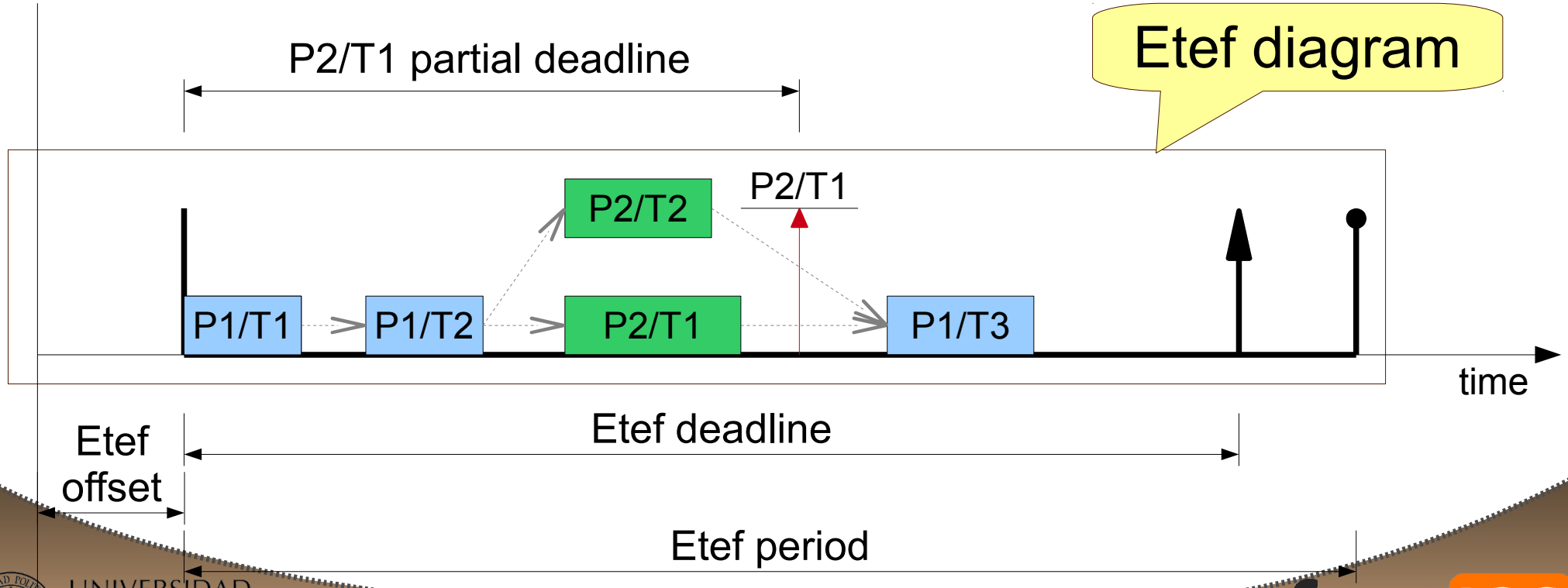
- Defines a critical section.
- The MERs are used along the task execution:
  - Are requested at the start, and released at the end of the task.
- The only attribute is the “name”.

End to End Flow (ETEF)

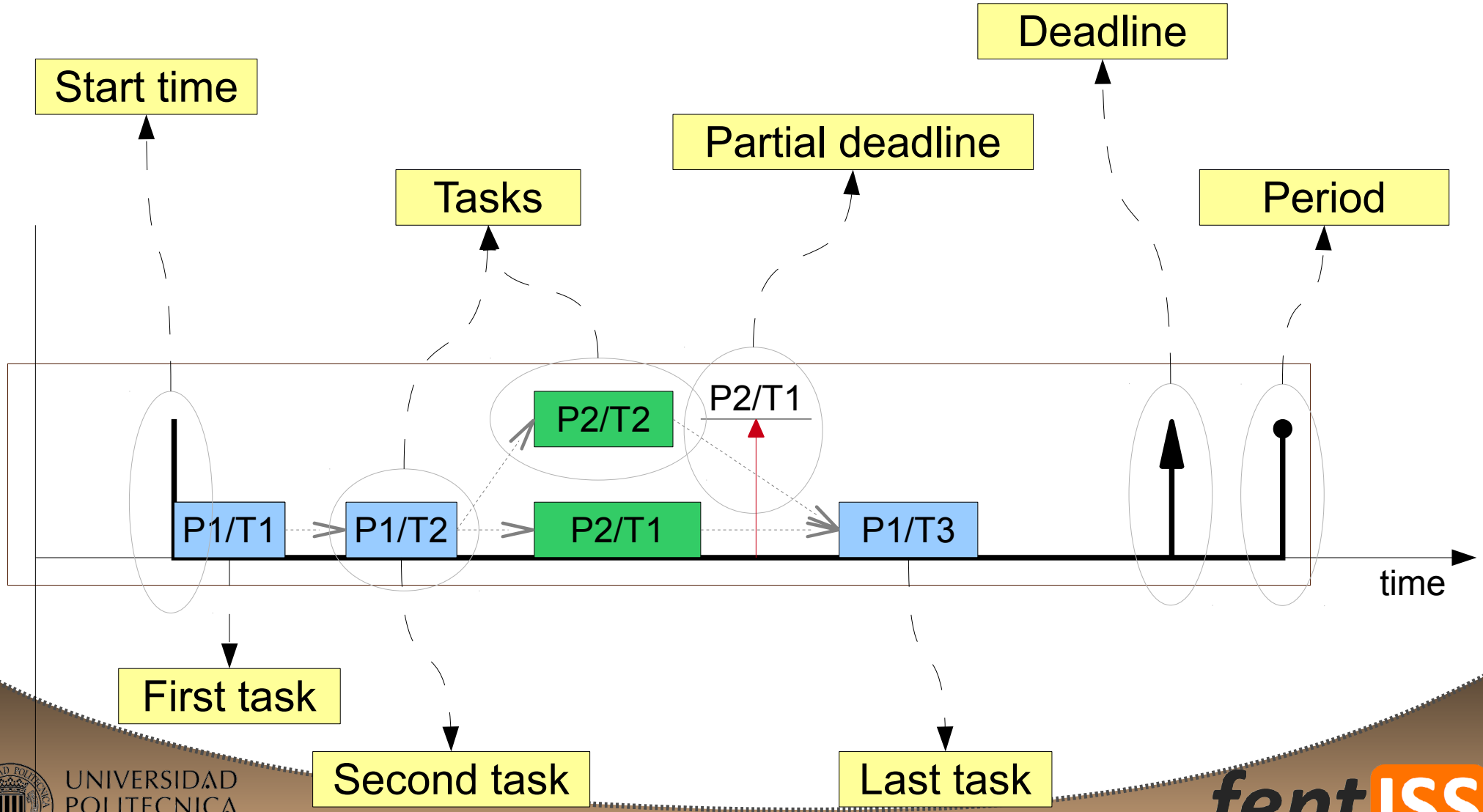
- An End to End Flow (ETEF) is a description of the temporal behaviour of the workload.
- An etef contains:
  - An directed acyclic graph (DAG) of tasks.
  - An activation pattern (periodic behaviour)
  - A deadline.
  - An offset.
  - Each task may have a partial deadline.

Etef diagram relations

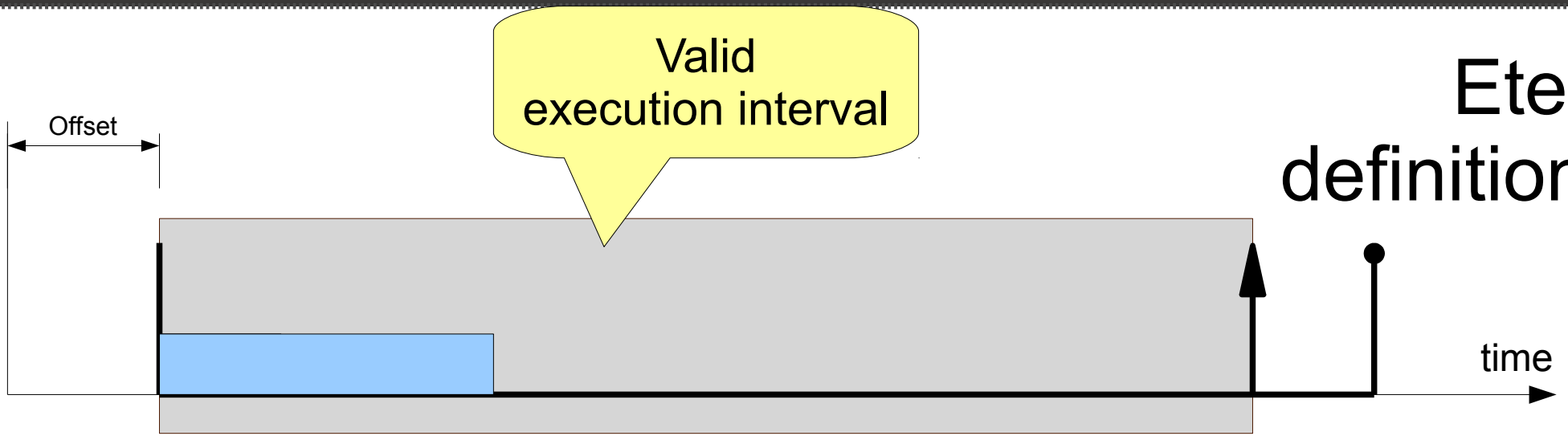




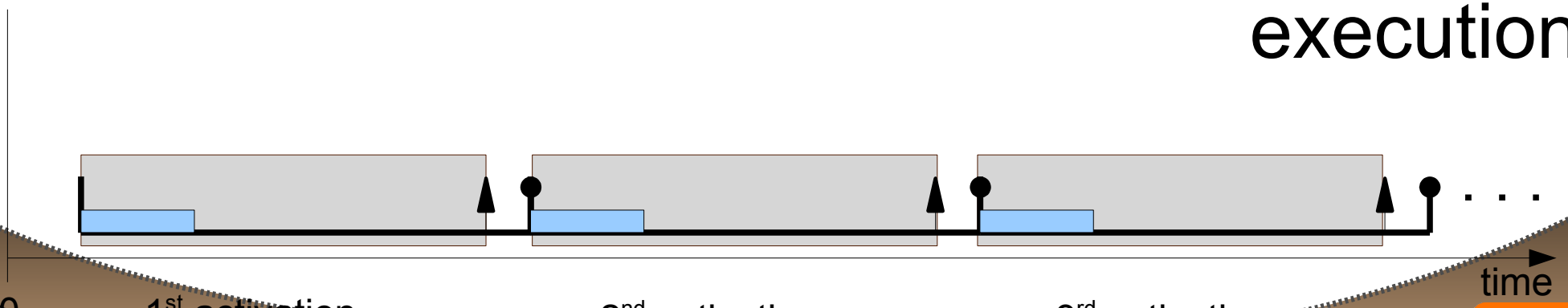
Etef diagram elements



Activations



Etef definition



Etef execution

Contents

Open issues

1. Periodic behaviour: what is a period?
2. Offsets: What is an offset?
3. Complex MERs: Cooldown time
4. Multi-plan (New in the ARINC standard)
5. What if the playground is the whole hyperperiod?
   - Task → step → job → slice
   - A DOM to refer not only to tasks but also to steps and jobs.

Periodic behaviour = period?

- Only those activities whose execution patterns are known can be analysed/planned off-line.
  - Priority scheduling analysis works with periodic tasks. Non-periodic are scheduling on the best-effort basis, and so, non-guaranteed.
- How the periodic behaviour is defined?
  - Inter-arrival time.
  - Activated frequency.
  - Other?

Periodic behaviour: ranges

- Are the periods absolute immutable values?
  - YES: Mathematical strict interpretation.
  - No: Engineering approach.
- Benefits of period selection/adjustment:
  - Suppose to tasks whose periods are: 12 and 25.
    - The LCM(12,25)=300
  - If the second task period is reduced to 24:
    - The LCM(12,24)=24
- The hyperperiod is much shorter.

Periodic behaviour: ranges

- In Xoncrete, the periods are specified as "ranges" of acceptable values.
  - Xoncrete computes que set of periods that produce the minimum hyperperiod.
- AADL allows to restrict the task periods to be selected among a pre-defined set of periods (`Allowed_period`).

Periodic behaviour: #Activations

- Shall the periods be "**Natural**" numbers?
  - On most scheduling works: "*without loss of generality we will assume that time is an integer*".
- Suppose that the Hyperperiod of all etefs **but the last one** is 8000, and the defined valid periods of the last etef is [130,150].
  - No number in the range [130,150] is divisor of 8000

Periodic behaviour: #Activations

(0)	(1)	(2)	(3)	(4)	(5)
MAF	Period	#Activations	Rounded # of Activations	(3) X (1)	Error (4) - (0)
8000	130	61,54	62	8060	60
	131	61,07	61	7991	9
	132	60,61	61	8052	52
	133	60,15	60	7980	20
	134	59,7	60	8040	40
	135	59,26	59	7965	35
	136	58,82	59	8024	24
	137	58,39	58	7946	54
	138	**57,97**	**58**	**8004**	**4**
	139	57,55	58	8062	62
	140	57,14	57	7980	20
	141	56,74	57	8037	37
	142	56,34	56	7952	48
	143	55,94	56	8008	8
	144	55,56	56	8064	64
	145	55,17	55	7975	25
	146	54,79	55	8030	30
	147	54,42	54	7938	62
	148	54,05	54	7992	8
	149	53,69	54	8046	46
	150	53,33	53	7950	50

Choosing the period:
138 t.u.

- There will be 58 activations in the MAF.
- There will be a total accumulated error of 4 t.u.

Periodic behaviour: #Activations

- In the worst case, the error is half the selected period.
- The error can be proportionally distributed along the activations.
- The time distance between two consecutive activations is not constant.
- In this case, there will be 4 activations shorter.
- The period is defined as:

“58 activations every 8000 t.u.”

Periodic behaviour: #Activations

- What is the error caused by timer granularity?
- What is the error caused by high priority etefs?
- **Is this source of "errors" acceptable?**

This interpretation makes the MAF and the periods of the etefs quite independent one from each other.

What is an *offset*? (i)

- We think it is an ambiguous word:
- 1.- A delay on the initial repetition of a periodic task
  - Activation time: Offset+Period*k
  - Relative deadline: **Offset+Period*k + Abs_deadline**
- 2.- A delay on the dispatch time
  - Activation time: Offset+Period*k
  - Relative deadline: **Period*k + Abs_deadline**
- 3.- May the Offset be larger than Period.

What is an *offset*? (ii)

- What is the relation between user defined “offsets”, “release times” and actual “start time” of the jobs.
- At least offset shall be renamed as:
  - “minimum offset”
  - “minimum start time”
  - “minimum activation time”

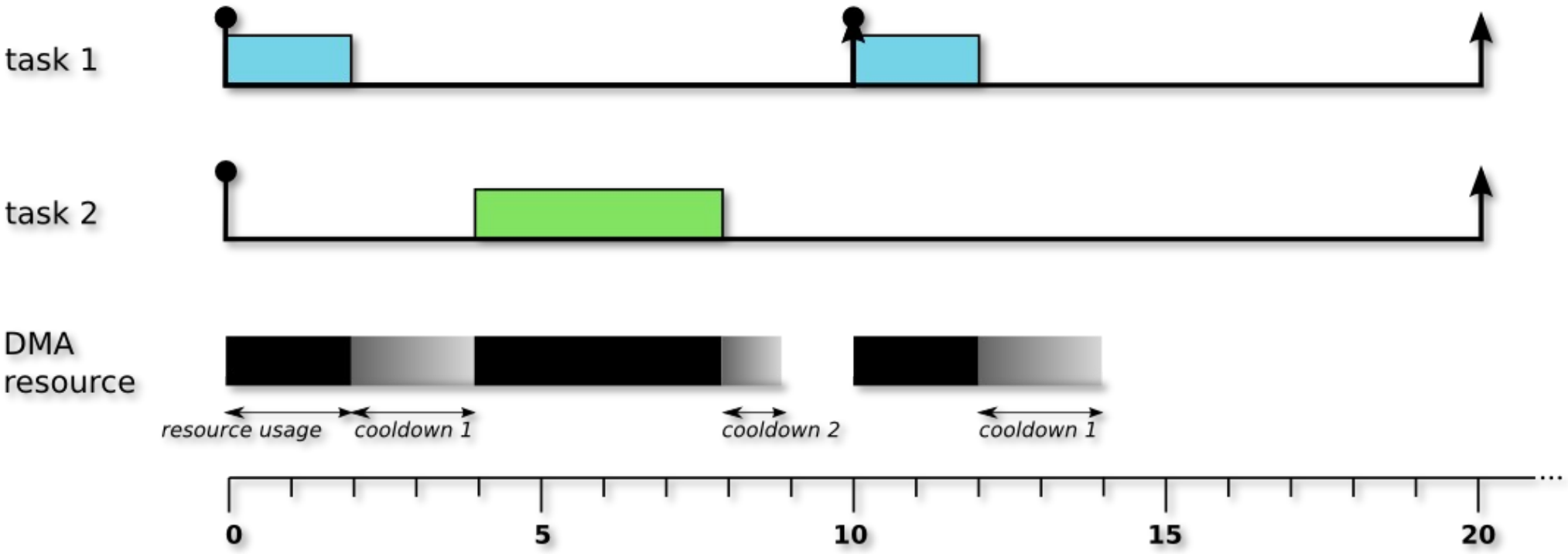
Multi-plan

- The etefs are instantiated in each “plan workload” and used to build the plan.
- How is defined the transition from one plan to the next one:
  - Synchronous?
  - A-synchronous?

Modeling parallel devices (cooldown)

- A peripheral may be programmed by a task and be active some time after the task finishes.
  - A network card.
  - A DMA device.
  - A disk
- The device will be ready after a “cooldown time”.
- The “cooldown time” of a resource can be different for each task.
  - The cooldown time is NOT an attribute of the MER.

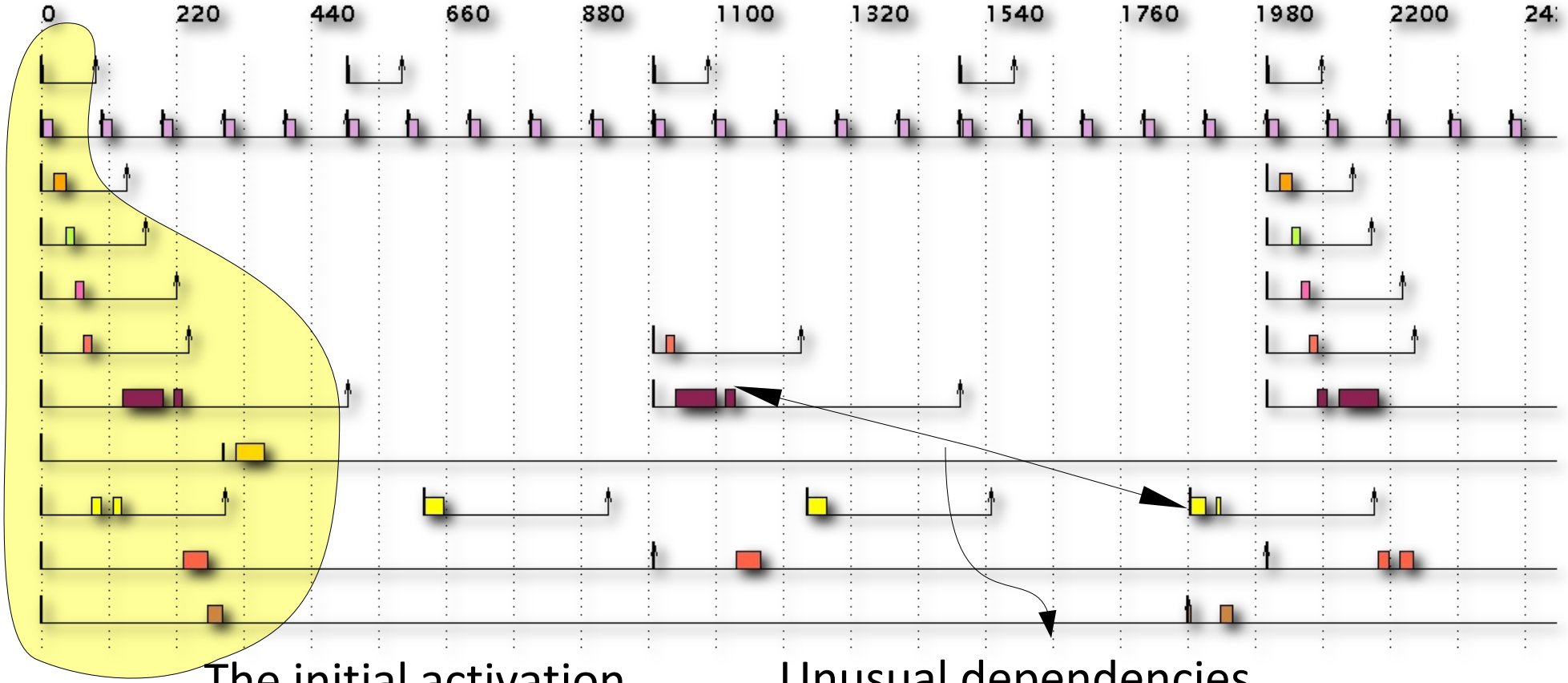
Cooldown time



RTDOM (RT Domain Object Model)

- The model is a simplification of the world.
- The world tend to be more complex than the model.
- If we add elements to the model the model becomes baroque and difficult.
  - The most representative elements shall be present in the model.
  - **But what if I need to define a tricky/complex/rare requirement?**

What do we model?



The initial activation,
plus the rep. pattern

Unusual dependencies.
Unusual for us, …
but may be not for others

RTDOM (ii)

- **But what if I need to define a tricky/complex/rare requirement?**
  - Do not add complexity to the model.
  - **Add a generic mechanism to do it**.
- We can reuse the W3C solutions:
  - Define an easy understand **structure** of the temporal elements.
  - Define a **syntax** to refer to those temporal elements.
  - Define a simple programming **language** to operate with them.

RTDOM (iii)

- The result would be pretty similar to the XML+DOM+javascript, but with the expansion in the time.
- Examples:
  - `etef[“first”].step[1].job[12].deadline`
  - `etef[“first”].step[1].job[12].dispatch_time`
- AADL suggest to express those “out-of-the-mode” requirements using “textual description”