

More and more needs for RT modeling...

1

☞ Needs of “Universal” modeling framework

❖ Specialized solutions must reach common users

☞ Needs for component based solutions

➤ UML not so bad... not only a fashion !

✓ Includes main common notations for modeling

→ Structure diagrams, behavior diagrams...

→ Widely learn... soon widely known !

✓ OO modeling favors component based development

What about existing practices ?

2

- ➡ **“Statechart” based approaches are already integrated**
(if we “forget” the semantic differences... !)
- ➡ **“SDL” based approaches are integrated...**
 - ❖ But they require both competencies and limit use of UML vs SDL on two separated modeling stages
- ➡ **Classical RT development should be supported soon**
 - ❖ Definition of an OO view of low level RT-OS paradigm

➔ Convergence requires efforts from all parts . . .

Which solution can satisfy everybody ?

☞ **Open developing environments must be proposed**

- ❖ Supporting all the main notations,
or allowing to introduce them (with their semantics)...

→ **Meta-modeling facilities !**

☞ **Standard mechanisms for model exchange (→ XMI)**

- ❖ With standard “profile” importation facilities

☞ **Translation “profiles”...**

- ❖ Describing how to translate a model based on a given set of paradigms into a model based on another

→ reuse of existing tools and practice

→ exchange of models between people with different cultures

☞ **and why not also “optimization” profiles**

- ❖ Automatic use of domain patterns, domain optimization, etc.