Extending AADL graphics capabilities, missing bits
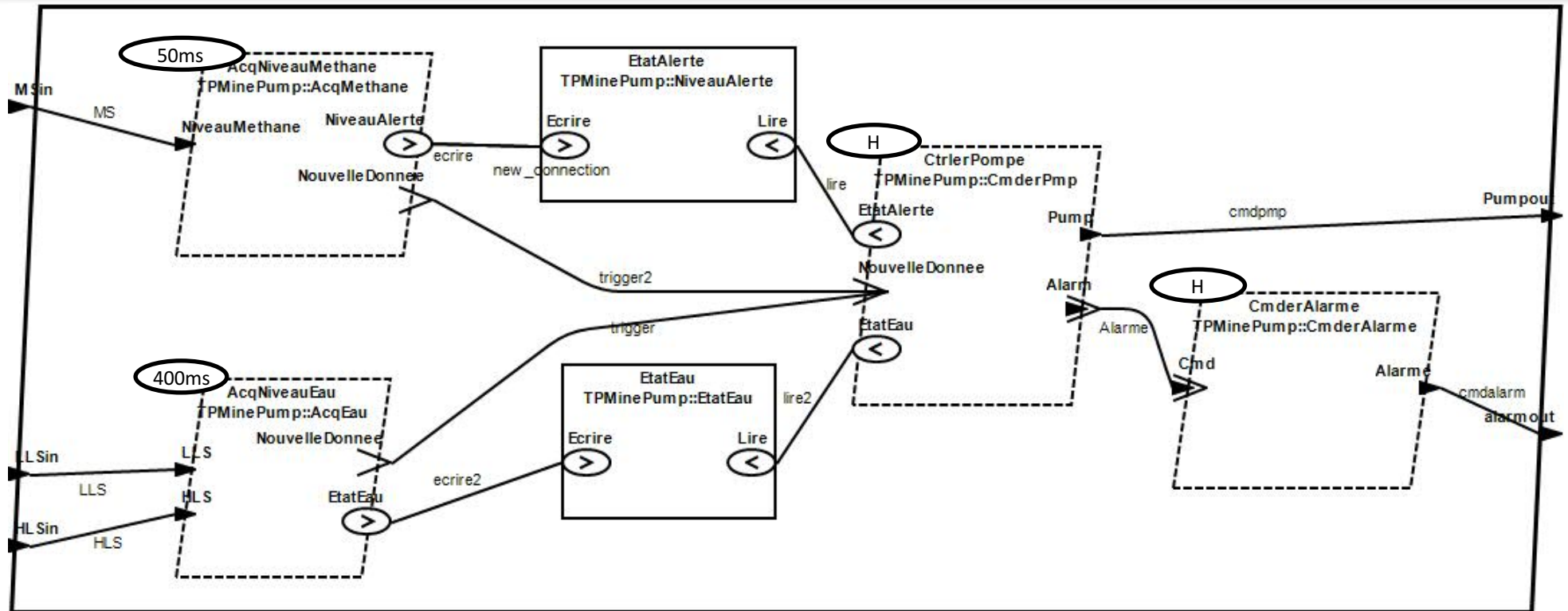
Jérôme Hugues, Emmanuel Grolleau

Rationale for some extensions on the graphical editor

> **We use AADL for teaching real-time systems**

> **Entry point is an AADL model with tasks, serve as graphical specification for classes**

   » Teaching-focused, for scheduling and real-time classes

> **We updated a textbook written by ENSMA to use SysML and AADL instead of DARTS**

   » We hit a few issues with the graphical editor and notation

     • Lack of some visual information compared to DARTS

     • Some elements missing from the standard

     • Suggestions for improved graphical look-and-feel

> **Following is a list of propositions to improve this area**

   » One usage among many users, must be configurable

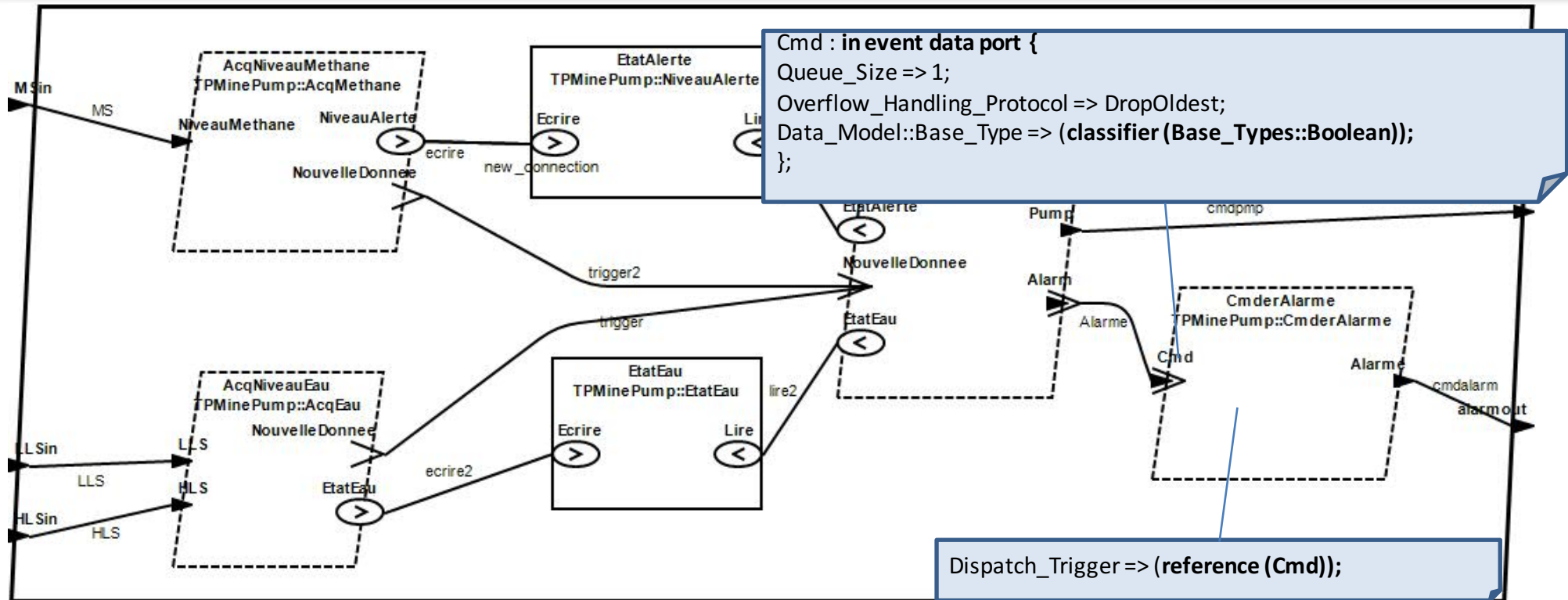   » System-level vs task-level to be separated and configurable !
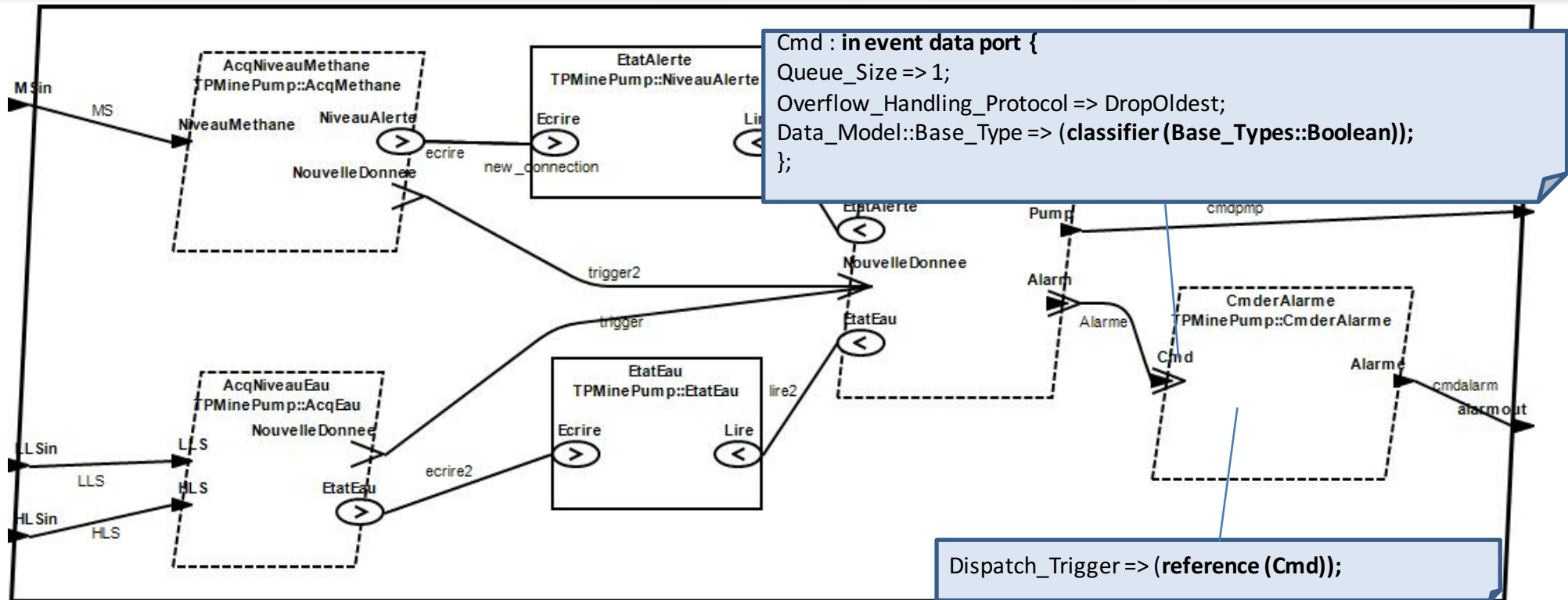
Threads: missing elements

> Dispatch protocol decorators *may* be attached, how to enable them in the editor ?
> Rationale: for a given level of details, user may want to have them
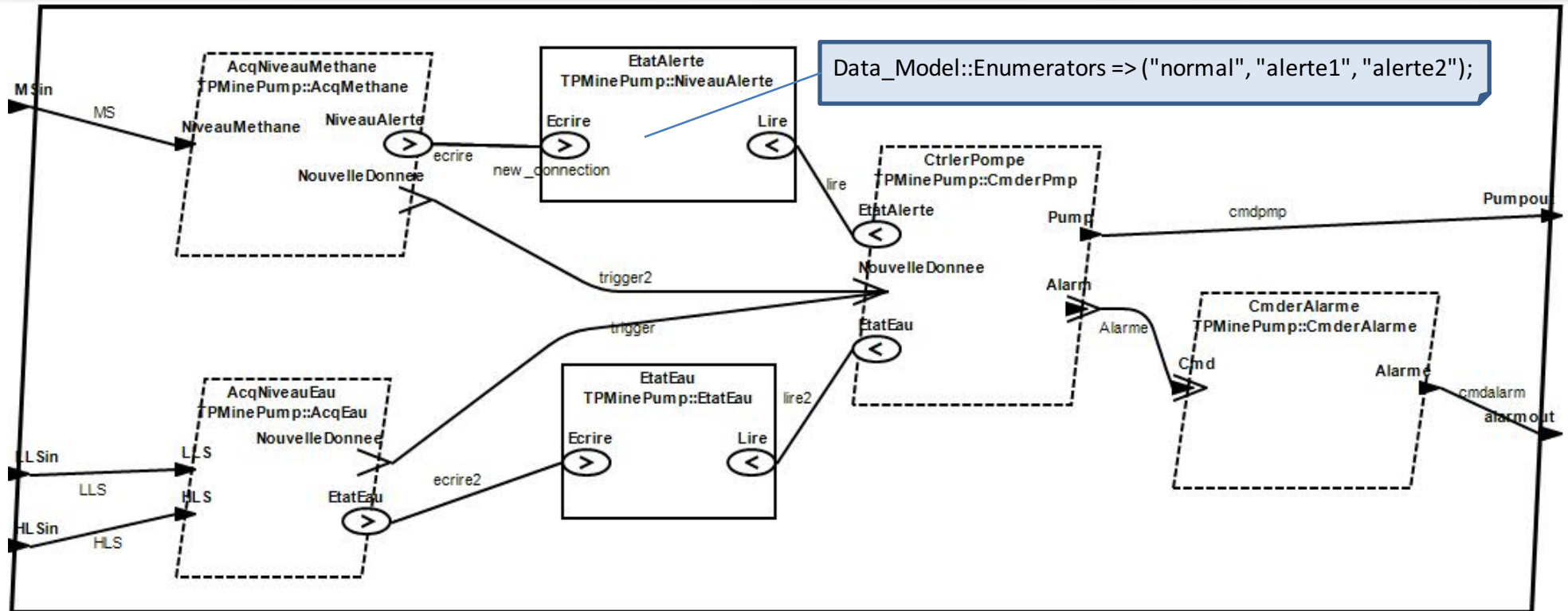
Information on dispatch condition

> Some elements are critical for a given perspective, those could appear in particular tooltips, for a set of viewpoints (filter) to be defined, e.g. scheduling, memory, etc.

> Tooltip can appear when cursor hover on components, or displayed when exporting

```
Cmd : in event data port {
Queue_Size => 1;
Overflow_Handling_Protocol => DropOldest;
Data_Model::Base_Type => (classifier (Base_Types::Boolean));
};
```

```
Dispatch_Trigger => (reference (Cmd));
```

> Queue_Size and actual semantics of communication should appear clearly. 9.2 (L5) allows event -> data port, this could appear clearly in graphics

Data types

> **Filters also to see details, e.g. data type associated to ports. User defined**
> **Missing: decorators on data access for read/write capability, concurrency control protocol**

Other elements on wishlist

> **Simulink, AltaRica and others allow user to attach additional info on the graphics**

  » E.g. colored background: component in one colored area == one concern

  » Attach pictures, e.g. to system blocks to represent the system, e.g. a plane, a wheel, etc.

> **Export diagram to SVG, not just bitmap format**

> **Provide an outline: clicking on one component may select/highlight it.**

  » Useful when you're lost on big models (keep in mind some are built by students ;)

> **"in" features could be on left side, out "features" on the right side of subcomponents**

  • I read left to right

> **I ran into issues with synchronization model, need a "force refresh" button somewhere, if that makes sense**