Systerel
Safe real-time solutions

More Abstraction

Laurent Voisin, Systerel

About Systerel

- Software development
  - Embedded
  - Safety-critical
  - Real-Time
- System design, safety evaluation, tools

- Formal Methods: B, Event-B, Tecla, Scade,…
- Data validation

Systerel
Safe real-time solutions

Stepwise Refinement

- Concepts are introduced gradually
- Complexity is sliced down in small pieces
- Makes a model easier to grasp
- And easier to prove

Instead of a gigantic proof

several small proofs

that can be automated

Systerel
Safe real-time solutions

But refinement is not enough

When you want to work with large models,
you need more than refinement

Systerel
Safe real-time solutions

A modelling example

- Interlocking system
  - First iteration in 2007
  - Second iteration in 2010
  - Third iteration in 2012
- What we learned…
  - Extract complicated data-structures
  - together with rules for reasoning on them

Systerel
Safe real-time solutions

The language is important

- Level of discourse
- Having the right tool (i.e., language)
- Mathematicians have known this for centuries

- Hence an extensible mathematical language

- AI could help detecting that a model is not at the right level (esp. for beginners)

Systerel
Safe real-time solutions

Don’t Repeat Yourself

- Already detected in classical B
- Modelling patterns
- Solution: Generate models + proof tactics

- In event-B, generic instantiation of patterns
- Proved once, used several times

Systerel
Safe real-time solutions

Link with AI

- Refinement plan:
- From a failed invariant proof,
- Based on pattern recognition
- Suggest a correction to the model

- Suggest a pattern instantiation instead

Systerel
Safe real-time solutions

Conclusion

- Theories + Patterns
- higher-level building blocks
- like a programming language library

- AI could help finding when a library should be used instead of inlined in a model

Systerel
Safe real-time solutions

Questions ?