THALES

Rodin in the field of railway system engineering

Tomas Fischer, Thales Austria GmbH

Contents

THALES

Motivation

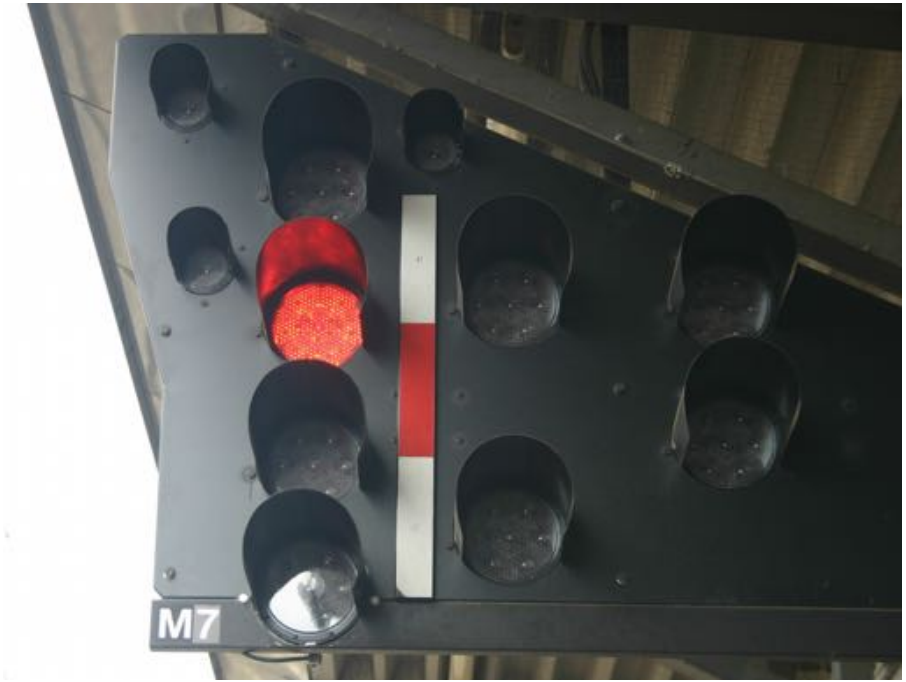
- **Formally written is better than informally told**
  - Provided it can be read and understood
- **Mathematically proven is better than just tested, but**
  - Incomplete proof is not worth anything
  - Sometimes (often) strongly confident is good enough
- **Business Driver: Reduce development costs and time to market**
  - Still maintaining the product quality
  - Time and cost can be decreased through productivity improvements

3 Processes – 3 Teams – 3 Skill Levels

- **Core development**
  - Development of core assets as generic definitions with extension points
  - Full Event-B expertise
- **Application engineering**
  - Instantiation of core assets for a given customer with customer-specific definitions
  - Partial Event-B expertise – can rely on support from core team
- **Field installation**
  - Deployment and validation of a customer system on a specific station
  - No Event-B expertise – Loops back to application development is expensive

Development process – Feature Driven Development

- **Design and realization by features**
  - Impact on already realized features
  - Parallel development
- **Example features**
  - Main signal, distant signal
  - Main / distant signal on same mast
  - Combined main and distant signal
  - Additional signal types
- **Different customers – different rules**



Von Abutoum - Eigenes Werk, Gemeinfrei, https://commons.wikimedia.org/w/index.php?curid=21556280

THALES

Development process – Domain Driven Design

- **Split the problem (and the solution) into subdomains**
  - Separation of concerns
  - Communication and collaboration – technical and domain experts
- **Elements domain**
  - Track, Point
  - Crossing with/without movable frog
  - Single / double slip switch
- **Route domain**
  - Points locked in the proper position

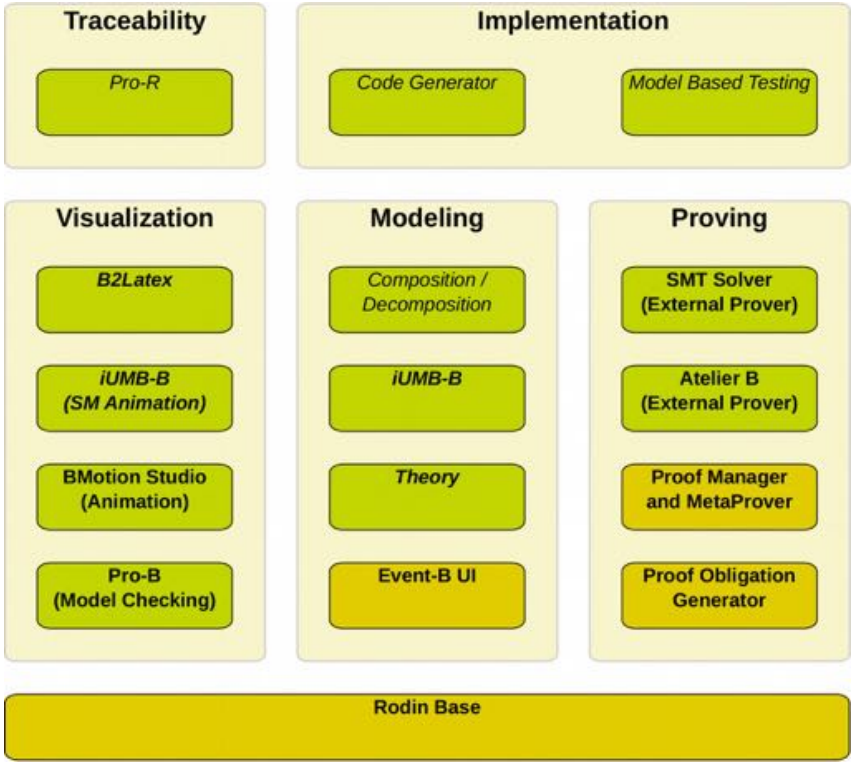
THALES

Additional Concepts

- **Object oriented analysis and design**
- **Variability management (product line)**
- **Model based engineering**
  - E.g. system model in Capella: https://www.polarsys.org/solutions/capella
- **Continuous integration**
  - Delta verification and validation
- **Quality assurance**
  - Rigorous reviews
- **Sustainability (25+ years)**

Rodin Toolset @ Thales Austria

- **Rodin utilization**
  - Specification and station data
  - Core domain
  - Full integration possible?
- **Assessment**
  - Event-B environment: TRL 5+
  - Provers: TRL 5+
  - Pro-B : TRL 5
  - iUML-B: TRL 4 – 5
  - Code generator, MBT: TRL 2



Plugins: **Accepted** – ***Tentative*** – *Declined*

THALES

Experiences and demand – Base

- **General Issues**
  - Tool stability
  - Documentation and tutorials
  - Maintenance (tool lock)
- **Plain text external representation**
  - Source control, teaming
  - Diff & merge, review support
  - Integration in the overall development process and toolchain
- **Syntactic sugar**
  - Named functions and predicates

Experiences and demand – Modeling and proving

- **High level langages**
  - Objects (records), control flow
  - High level feedback (provers, model checking, animation)
- **Not only refinement**
  - Relaxed refinement rules (refine more machines, combine unrelated events)
  - Top-down (refinement) as well as bottom-up (composition) approach
- **Structures (with namespaces)**
  - Packages, Modules, Components
- **Impact analysis**
- **Temporal logic (LTL) formulae**

Key Messages

- **Set theory and 1st order logic are reasonable**
  - High level langages (OO) are needed for the non Event-B experts
- **Practical applicability for real-word applications matters**
  - Hard facts : Problem space gets intractable
  - Soft facts : Usability gets unsatisfactory
- **Improvements and enhancements**
  - Many are possible in the Rodin toolset
  - More can be done by augmenting ‘pure’ Event-B
- **Cost – Benefit analysis (business case) will decide**

Thank you for your attention

Open source demonstration model:
https://github.com/klar42/railground