RODIN - the next generation refinement tools

Michael Butler

Rodin

2004-2007

- **Goal**: methodology and supporting open tool platform for rigorous development of dependable complex software systems and services.
  - Formal methods + fault tolerance

Partners

ClearSy	Newcastle
Nokia	Åbo Akademi
Praxis-CS	Southampton
ATEC	ETH Zurich

rodin.cs.ncl.ac.uk

RODIN Philosophy

- System level modelling is essential for *reasoning* about complex systems
- *Development* requires formal modelling at multiple levels of abstraction
- Models form refinement chains
- Construction of refinement chains requires strong tool support
- Event B

Expected Results of RODIN

- A collection of developments (models, architectures, proofs, components, etc.) produced by the case studies.
- A set of guidelines on rigorous development of complex systems
- An open tool platform supporting extensibility
- A collection of plug-in tools

rodin.cs.ncl.ac.uk

Case Studies

1. Position calculation for 3G phones
2. Engine failure management
3. Mobile internet application
4. Air traffic display (CDIS)
5. Ambient campus

Key Tool Decisions (I)

- Support incremental development
  - *Reactive*: analysis tools are automatically invoked in the background whenever a change is made
  - *Differential*: analytical impact of changes is minimised as much as possible
  - *Feedback*: traceability from errors to model elements

Key Tool Decisions (II)

- No concrete language
- Instead the platform provides a repository of structured modelling elements
  - the only concrete language is set theory and logic
- Extensibility:
  - extend modelling elements
  - extend functionality through plugins

RODIN platform development team

- ETH:
  - Jean-Raymond Abrial
  - Laurent Voisin
  - Stefan Hallerstede
  - Farhad Mehta
  - Thai Son Hoang
- Clearsy
  - Francois Terrier

RODIN Open Tool Platform

- Extension of Eclipse IDE (Java based)
- Repository of structured modelling elements (Java objects and XML files)
- RODIN Eclipse Builder manages:
  - Well-formedness + type checker
  - Consistency/refinement PO generator
  - Prover
  - Propagation of changes

DEMO

eprints.ecs.soton.ac.uk/12711/

http://sourceforge.net/projects/rodin-b-sharp/

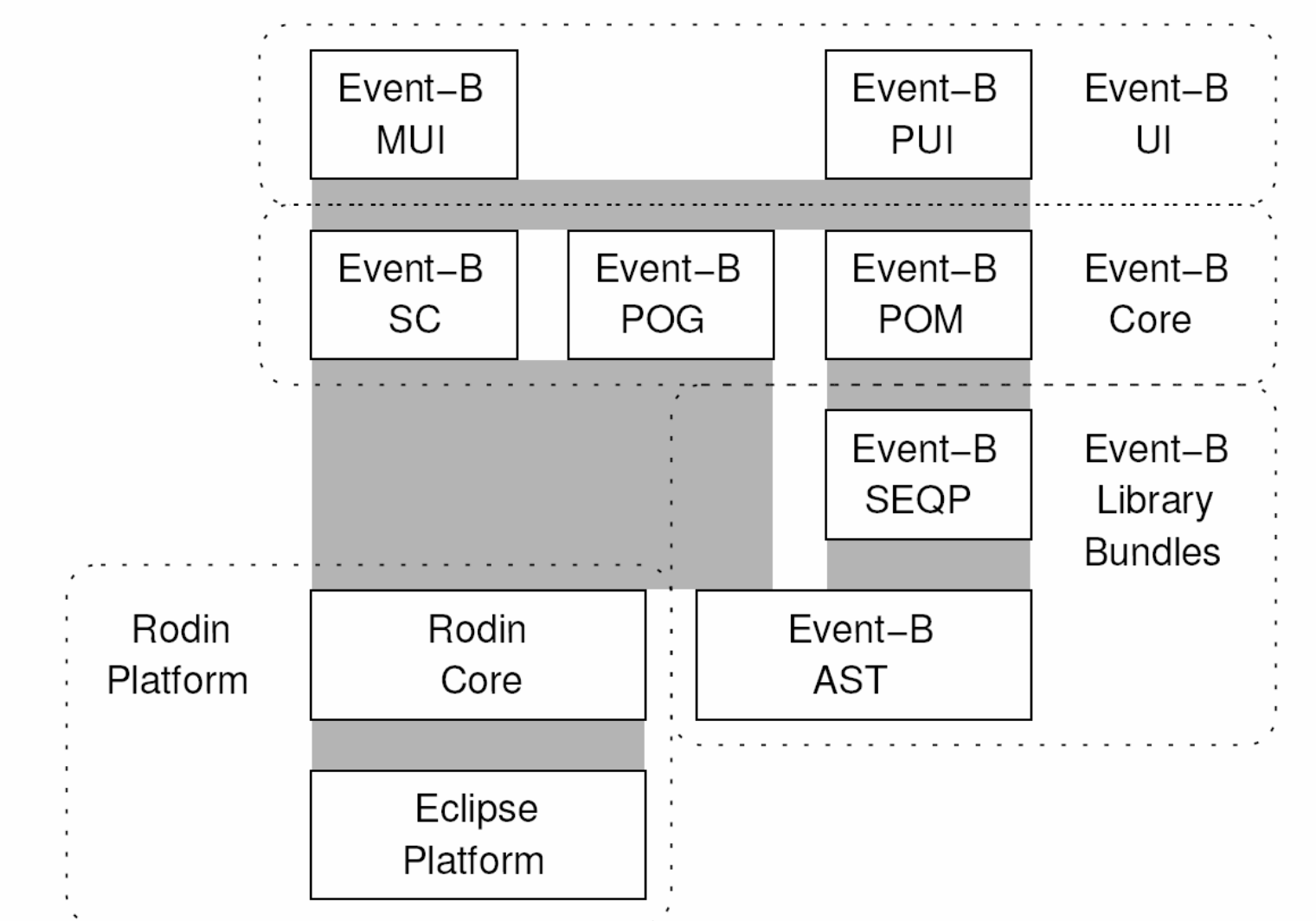
RODIN Plug-ins

- Linking UML and B
- Model checking
  - ProB: consistency and refinement checking
  - Mobility checker (Petri-net based)
- Graphical model animation
  - ProB
  - Brama
- Documentation generation
- Code Generation
- Model-based testing
- …

Integration

- Models managed in Platform repository
- Plug-ins generate, access and manipulate repository elements
- Sharable repository elements:
  - Models
  - Proof obligations
  - Proofs
  - Counterexamples
  - Documents
  - Graphical animators
  - Implementations
  - …

Platform architecture



Rodin

Extension points for plug-ins

- Extend repository elements
- Extend proof obligation generation
- Extend interface with menus, buttons, views, etc
- Extend project builder to support the reactive development process
  - Extend differential analysis
  - Extend feedback to user