UNIVERSITY OF Southampton
School of Electronics and Computer Science

Theory Plug-in for Rodin 3.0

T.S. Hoang[1] A. Salehi[1] M. Butler[1] L. Voisin[2]

[1]ECS, University of Southampton, U.K.

[2]Systerel, France

RODIN Workshop 2016

Linz, Austria

23rd May 2015

From Rodin 2.8 to Rodin 3.0 (1/2)

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Major (necessary) changes to the Rodin Core.

Stronger AST Library

- Mitigate risks of unsoundness: mixing several formula factories.
- Every AST node carries its building formula factory.
- Operation combining formulas check for factories compatibility.

From Rodin 2.8 to Rodin 3.0 (2/2)

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Major (necessary) changes to the Rodin Core.

Stronger sequent prover

- Introduction of context-dependent reasoner.
- Context-dependent reasoner cannot be reused.
- Context-dependent reasoner has to be replayed
- The rule-based provers' reasoners are context-dependent.

Problems for the Theory Plug-in

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Exceptions when openning proof obligation.
- Exceptions when applying rule-based provers' reasoners
- Changing the model has no effects on existing proofs.
- Losing proofs when saving
  (the exact problem is in loading previously saved proof).

Upgrading the Theory Plug-in

Pattern Matching Facility

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Use *ISpecialization* insteads of *ISubstitution*.
- Allows to specialize types consistently.

Patterns		*Formulae*
S	$\longrightarrow$	$\mathbb{P}(S)$
S	$\longrightarrow$	$S \times T$

Upgrading the Theory Plug-in

Matching for Associative Operators

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Proper implementation for matching associative operators.

Patterns	Formulae	Result
$f; \{x \mapsto c\}$	$g; h; \{y \mapsto c\}$	$f \leftarrow g; h$
		$x \leftarrow y$
		$c \leftarrow c$

Upgrading the Theory Plug-in

Matching for Associative Operators

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Proper implementation for matching associative operators.

Patterns	Formulae	Result
$f; \{x \mapsto c\}$	$g; h; \{y \mapsto c\}$	$f \leftarrow g; h$ $x \leftarrow y$ $c \leftarrow c$
$e; f$	$g; h; \{y \mapsto c\}$	$e \leftarrow g$ $f \leftarrow h; \{y \mapsto c\}$

Datatype and Operator Extensions

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Correctly implement equality for datatype/operator extensions.
- $\Longrightarrow$ Datatypes/Operators with the same definition will be assigned identical IDs.
- $\Longrightarrow$ Formula factories can be correctly compared and upgraded.
- $\Longrightarrow$ saved proofs are loaded with the correct formula factories.

Theory Plug-in Version 4.0

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Major upgrade of the Theory Plug-in
- Previously saved proofs will be lost.
- The upgrade requires fixed in the Rodin Core
- Will be available after the next release of the Rodin Platform (Rodin 3.3)

Further Development

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Support for infix predicate operators.
- Support for predicate variables in theories.
- Usability improvement
- Improve matching facility for associative commutative operators
- Tatics for theory.
- Theory instantiation

Infix Predicate Operators

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Cosmetic changes to improve readability.
- For example, for real numbers x_1, x_2, insteads of

$$smr(x_1, x_2),$$

  we can write

$$x_1 \lessdot x_2$$

- (No overloading of arithmetic operators).

Predicate Variables in Theories

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Currently cannot be statically checked
- Despite the rule-based provers already have some support.
- Need some additional supports from the Rodin Core.

Usability Improvement

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Interactive proofs slow in computing “applicable positions”
- $\Longrightarrow$ Compute applicable positions on demand.
- Rodin Interactive proofs support needs to be changed.

Matching for AC operators

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Matching for Associative and Commutative operators use the same algorithm for Associative operators.
- More matching can be found if take into account commutivity.

Example

- Pattern: $x + f(y)$
- Formula: $a + f(b) + c$
- Match: $x \leftarrow a + c$, $y \leftarrow b$.

Tactics for Theory

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Proof rules and definitions are applied in some predefined order.
- Often, users want dedicate tactics
- Simple tactic language: Sequential composition, loops (similar to the current Rodin's preferences)
- Tactics associated with theories or with the developments?

Theory Instantiation

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Enhance reuse of theories.
- Suited for defining Abstract Data Types and their concrete representation.
- Supporting model variations through theories.

A Wish List

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Support for infix predicate operators.
- Support for predicate variables in theories.
- Usability improvement
- Improve matching facility for associative commutative operators
- Tatics for theory.
- Theory instantiation
- ...