Separation Logic

Introduction

Arthur Charguéraud

February 2014

Motivation for Separation Logic

Separation Logic: a technique for modular specification and verification of arbitrarily-complex imperative programs.

Covered in this course:

- theory of Separation Logic,
- writing of specifications in practice,
- verification proofs in Separation Logic.

Not covered:

- Automated analyses based on Separation Logic.

Essence of Separation Logic

Linearity:

- from theorists: linear logic, linear types
- from practicioners: region, ownership, capabilities

Local reasoning:

- to isolate the relevant part of the heap,
- to allow for modular verification.

Origins of Separation Logic

- John Reynolds (2000)
  - Intuitionistic Reasoning about Shared Mutable Data Structure
  - —building on ideas from Burstall (1972).
- John Reynolds, Peter O'Hearn, Hongseok Yang (2001)
  - Local reasoning about programs that alter data structures
- John Reynolds (2002)
  - Separation Logic: A logic for shared mutable data structure.

Applications of Separation Logic

Micro-controller	Klein et al	NICTA
Assembly language	Chlipala et al	MIT
Operating system	Shao et al	Yale
C (drivers)	Yang et al	Oxford
C-light	Appel et al	Princeton
C11 (concurrent)	Vafeiadis, Parkinson et al	MPI and MSR
ML	Morisset et al	Harvard
Java	Parkinson et al	MSR and Cambridge
Java	Jacobs et al	Leuven
Javascript	Gardner et al	Imperial College
Caml	Charguéraud	Inria

→ For more projects, see Peter O'Hearn's webpage on Separation Logic:

`http://www0.cs.ucl.ac.uk/staff/p.ohearn/SeparationLogic/Separation_Logic/SL_Home.html`

This course on Separation Logic

Compared with other Separation Logic courses:

- targets a clean ML language (not C)
- presents definitions used in practice (not $-\!\!*$, not $\wedge$)
- targets higher-order logic (not first-order)
- supports higher-order functions

Main difference with the Why approach

Compared with the previous course based on Why:

- applies to un-annotated programs
- proof automation is not a concern
- many similar rules, but different interpretations
- addresses the aliasing problem
- allows for local reasoning.