A Stone-Type Duality Theorem for Separation Logic via its Underlying Bunched Logics

MFPS 2017

Simon Docherty

University College London

Monday 12th June 2017

Joint work with David Pym
What is Duality?
A Really Informal Description of Duality
A formal definition of duality

A **dual equivalence of categories** is

- a pair of functors $F : C \to D^\text{op}$ and $G : D^\text{op} \to C$
A formal definition of duality

A dual equivalence of categories is

- a pair of functors \( F : C \to \mathcal{D}^{\text{op}} \) and \( G : \mathcal{D}^{\text{op}} \to C \)
- together with natural isomorphisms \( \epsilon : \text{Id}_{\mathcal{D}^{\text{op}}} \to FG \) and \( \eta : \text{Id}_C \to GF \)
Stone Duality

Theorem (Representation Theorem for Boolean Algebras)

Every Boolean algebra embeds into the power set algebra of its ultrafilters by \( h(a) = \{ F \mid a \in F \} \).

- Add topology to the set of ultrafilters and this strengthens to a duality.

---

Stone Duality

Theorem (Representation Theorem for Boolean Algebras)

1. Every Boolean algebra embeds into the power set algebra of its ultrafilters by \( h(a) = \{ F \mid a \in F \} \).

- Add topology to the set of ultrafilters and this strengthens to a duality.
- Application to logic 1:
  1. Algebra generalizes syntax
  2. Topology generalizes semantics
  3. Stone Duality generalizes soundness and completeness.

---

Stone Duality

**Theorem (Representation Theorem for Boolean Algebras)**

1. Every Boolean algebra embeds into the power set algebra of its ultrafilters by \( h(a) = \{ F \mid a \in F \} \).

- Add topology to the set of ultrafilters and this strengthens to a duality.
- Application to logic 1:
  1. Algebra generalizes syntax
  2. Topology generalizes semantics
  3. Stone Duality generalizes soundness and completeness.
- Application to logic 2:
  Prove metatheory algebraically/topologically.

---

What Is Separation Logic?
Separation Logic

- Tool used in static analysis of programs that access and mutate data\textsuperscript{23}
- Hoare triples $\{\varphi\} C \{\psi\}$ where $\varphi, \psi$ formulae of an assertion language extending first-order logic with separating connectives $\ast, I$ and $\neg\ast$ and the points-to predicate $\mapsto$.
- Interpreted on memory states comprised of a store $s$ (stack-allocated memory) and a heap $h$ (dynamically-allocated memory).

\textsuperscript{2}J. Reynolds. Separation Logic: A Logic for Shared Mutable Data Structures, LICS 2002
\textsuperscript{3}S. Ishtiaq and P. O’Hearn. BI as an Assertion Language for Mutable Data Structures, POPL 2001
Separation Logic

- Tool used in static analysis of programs that access and mutate data\(^2\).
- Hoare triples \(\{\varphi\}C\{\psi\}\) where \(\varphi, \psi\) formulae of an assertion language extending first-order logic with separating connectives \(\ast, \bot\) and the points-to predicate \(\equiv\).
- Interpreted on memory states comprised of a store \(s\) (stack-allocated memory) and a heap \(h\) (dynamically-allocated memory).

---

\(^2\) J. Reynolds. Separation Logic: A Logic for Shared Mutable Data Structures, LICS 2002

\(^3\) S. Ishtiaq and P. O’Hearn. BI as an Assertion Language for Mutable Data Structures, POPL 2001
Store-Heap Semantics of Separation Logic

- Store $s$ gives evaluation of terms $[e]s$ built out of variables, integers, arithmetic functions.
- $s, h \models e \iff e'$ iff $\text{dom}(h) = \{[e]s\}$ and $h([e]s) = [e']s$. 
What is Duality?  What Is Separation Logic?  Duality for Propositional Separation Logic  Duality for Separation Logic  Conclusions

Store-Heap Semantics of Separation Logic

- Store $s$ gives evaluation of terms $[e]s$ built out of variables, integers, arithmetic functions.
- $s, h ⊨ e \mapsto e'$ iff $\text{dom}(h) = \{[e]s\}$ and $h([e]s) = [e']s$.
- $s, h ⊨ \varphi * \psi$ iff $h$ can be separated into disjoint $h', h''$ s.t. $s, h' ⊨ \varphi$ and $s, h'' ⊨ \psi$. 
Store-Heap Semantics of Separation Logic

- Store $s$ gives evaluation of terms $[e]s$ built out of variables, integers, arithemetic functions.
- $s, h \models e \leftrightarrow e'$ iff $\text{dom}(h) = \{[e]s\}$ and $h([e]s) = [e']s$.
- $s, h \models \varphi \ast \psi$ iff $h$ can be separated into disjoint $h'$, $h''$ s.t. $s, h' \models \varphi$ and $s, h'' \models \psi$.
- $s, h \models \exists x \varphi$ iff there exists $a$ such that $s[x \leftarrow a], h \models \varphi$. 
Duality for Propositional Separation Logic
Boolean Bunched Logic

BBI is a **bunched logic**\(^4\): the free combination of CPL and MILL.

**Syntax**

A resource algebra \(A\) is an algebra \((A, \land, \lor, \bot, \top, *, \#)\) such that

- \((A, \land, \lor, \bot, \top)\) is a Boolean algebra,
- \((A, *, I)\) is a commutative monoid,
- For all \(a, b, c \in A: a * b \leq c\) iff \(a \leq b \rightarrow^* c\).

Boolean Bunched Logic

BBI is a bunched logic⁴: the free combination of CPL and MILL.

Syntax

A resource algebra $A$ is an algebra $(A, \land, \lor, \bot, \top, \ast, \ast, I)$ such that

- $(A, \land, \lor, \bot, \top)$ is a Boolean algebra,
- $(A, \ast, I)$ is a commutative monoid,
- For all $a, b, c \in A$: $a \ast b \leq c$ iff $a \leq b \ast c$.

Semantics

A resource frame is a structure $(X, \circ, E)$ such that

- $\circ : X^2 \to \mathcal{P}(X)$ is an associative & commutative operation,
- $E \subseteq X$ satisfies $\{r\} \circ E = \{r\}$ for all $r \in X$ (where $\circ$ extended to an operation on sets).

Theorem (Representation Theorem for Resource Algebras)

Every resource algebra $\mathbf{A}$ can be embedded into a resource algebra of sets generated by a resource frame.
Theorem (Representation Theorem for Resource Algebras)

Every resource algebra \( \mathbb{A} \) can be embedded into a resource algebra of sets generated by a resource frame.

Proof Sketch.

\[ \circ_{Uf} : Uf(\mathbb{A})^2 \rightarrow \mathcal{P}(Uf(\mathbb{A})) \text{ given by} \]

\[ F \circ_{Uf} F' = \{ F'' | \forall a \in F, \forall b \in F' : a \ast b \in F'' \} \]
Representation and Duality

Theorem (Representation Theorem for Resource Algebras)

Every resource algebra $\mathbb{A}$ can be embedded into a resource algebra of sets generated by a resource frame.

Proof Sketch.

1. $\circ_{Uf} : Uf(\mathbb{A})^2 \rightarrow \mathcal{P}(Uf(\mathbb{A}))$ given by
   
   $$F \circ_{Uf} F' = \{F'' \mid \forall a \in F, \forall b \in F' : a * b \in F''\}$$

2. $E_{Uf} = \{F \in Uf(\mathbb{A}) \mid I \in F\}$. 

Add topology and coherence conditions to resource frames to get resource spaces and a dual equivalence of categories.
Theorem (Representation Theorem for Resource Algebras)

Every resource algebra $\mathbb{A}$ can be embedded into a resource algebra of sets generated by a resource frame.

Proof Sketch.

- $\circ_{Uf} : Uf(\mathbb{A})^2 \rightarrow \mathcal{P}(Uf(\mathbb{A}))$ given by
  $F \circ_{Uf} F' = \{ F'' | \forall a \in F, \forall b \in F' : a \ast b \in F'' \}$
- $E_{Uf} = \{ F \in Uf(\mathbb{A}) | I \in F \}$.
- $(Uf(\mathbb{A}), \circ_{Uf}, E_{Uf})$ generates power set resource algebra that $\mathbb{A}$ embeds into with $h(a) = \{ F \in Uf | a \in F \}$.

Add topology and coherence conditions to resource frames to get resource spaces and a dual equivalence of categories.
Duality for Separation Logic
Separation Logic Algebraized

A **resource hyperdoctrine**\(^5\) is a functor \(P : C^{\text{op}} \to \text{ResAlg}\) such that

1. \(C\) is a category with **finite products**: for every \(C_1, \ldots, C_n\) in \(C\), \(C_1 \times \cdots \times C_n\) exists.

---

\(^5\)B. Biering, L. Birkedal, and N. Torp-Smith. BI Hyperdoctrines and Higher-order Separation Logic. ESOP 2005
Separation Logic Algebraized

A **resource hyperdoctrine** is a functor $\mathbb{P} : C^{\text{op}} \to \text{ResAlg}$ such that

1. $C$ is a category with **finite products**: for every $C_1, \ldots, C_n$ in $C$, $C_1 \times \cdots \times C_n$ exists.
2. For each $X, \Gamma$ in $C$ there exist monotone maps
   $\exists X_\Gamma, \forall X_\Gamma : \mathbb{P}(\Gamma \times X) \rightarrow \mathbb{P}(\Gamma)$ satisfying **adjointness**
   $$\exists X_\Gamma(a) \leq b \iff a \leq \mathbb{P}(\pi_{\Gamma,X})(b)$$
   and **naturality** properties.

---

5B. Biering, L. Birkedal, and N. Torp-Smith. BI Hyperdoctrines and Higher-order Separation Logic. ESOP 2005
Separation Logic Algebraized

A resource hyperdoctrine is a functor \( \mathbb{P} : C^{\text{op}} \to \text{ResAlg} \) such that

1. \( C \) is a category with finite products: for every \( C_1, \ldots, C_n \) in \( C \), \( C_1 \times \cdots \times C_n \) exists.
2. For each \( X, \Gamma \) in \( C \) there exist monotone maps
   \( \exists X_\Gamma, \forall X_\Gamma : \mathbb{P}(\Gamma \times X) \to \mathbb{P}(\Gamma) \) satisfying adjointness
   \[ \exists X_\Gamma(a) \leq b \text{ iff } a \leq \mathbb{P}(\pi_{\Gamma,X})(b) \]
   and naturality properties.
3. For each \( X \) in \( C \) there exists an element \( =_X \in \mathbb{P}(X \times X) \) satisfying an adjointness property. Given diagonal map
   \( \Delta_X : X \to X \times X : \)
   \[ \top \leq \mathbb{P}(\Delta_X)(a) \text{ iff } =_X \leq a \]

---

5 B. Biering, L. Birkedal, and N. Torp-Smith. BI Hyperdoctrines and Higher-order Separation Logic. ESOP 2005
Separation Logic Topologized

An **indexed resource frame** is a functor $\mathcal{R} : C \rightarrow \text{ResFr}$ such that:

- $C$ is a category with finite products.
- For all objects $\Gamma, \Gamma', X$ and morphisms $s : \Gamma \rightarrow \Gamma'$ the following commutative square satisfies the **quasi-pullback**\(^6\) property.

\[
\begin{array}{ccc}
\mathcal{R}(\Gamma \times X) & \xrightarrow{\mathcal{R}(\pi_{\Gamma,X})} & \mathcal{R}(\Gamma) \\
\mathcal{R}(s \times \text{id}_X) \downarrow & & \downarrow \mathcal{R}(s) \\
\mathcal{R}(\Gamma' \times X) & \xrightarrow{\mathcal{R}(\pi_{\Gamma',X})} & \mathcal{R}(\Gamma')
\end{array}
\]

Replace $\text{ResFr}$ with $\text{ResSp}$ and add coherence conditions = **indexed resource spaces**.

---

The Store-Heap Model is an Indexed Resource Frame

- The set of heaps forms a resource frame: \((\text{Heaps}, \cdot, \{[]\})\).
- Define \(\text{Store} : \text{Set} \rightarrow \text{ResFr}\) by

\[
\text{Store}(X) = (X \times \text{Heaps}, \cdot_X, X \times \{[]\})
\]

\[(x, h) \cdot_X (y, h') = \begin{cases} 
\emptyset & \text{if } x \neq y \text{ or } \neg h \# h' \\
\{(x, h \cdot h')\} & \text{otherwise}.
\end{cases}
\]

- Each \(\text{Store}(X)\) is a resource frame.
The Store-Heap Model is an Indexed Resource Frame

- The set of heaps forms a resource frame: \((\text{Heaps}, \cdot, \{[]\})\).
- Define \(\text{Store} : \text{Set} \rightarrow \text{ResFr}\) by

\[
\text{Store}(X) = (X \times \text{Heaps}, \cdot_X, X \times \{[]\})
\]

\[
(x, h) \cdot_X (y, h') = \begin{cases} 
\emptyset & \text{if } x \neq y \text{ or } \neg h \# h' \\
\{(x, h \cdot h')\} & \text{otherwise}.
\end{cases}
\]

- Each \(\text{Store}(X)\) is a resource frame.
- An \(n\)-ary store \(s = [x_1 \rightarrow a_1, \ldots, x_n \rightarrow a_n]\) with heap \(h\) is encoded as \(((a_1, \ldots, a_n), h) \in \text{Store}(\text{Val}^n)\).

**Theorem**

The Kripke semantics of the indexed resource frame \(\text{Store}\) coincides with the memory model semantics of Separation Logic.
Duality for BBI gives us

- a pair of functors $F : \text{ResAlg} \to \text{ResSp}^{\text{op}}$ and $G : \text{ResSp}^{\text{op}} \to \text{ResAlg}$

- together with natural isomorphisms $\epsilon : \text{Id}_{\text{ResSp}^{\text{op}}} \to FG$ and $\eta : \text{Id}_{\text{ResAlg}} \to GF$. 
Putting it all together

- Take a resource hyperdoctrine $P : C^{op} \rightarrow \text{ResAlg}$. 
Putting it all together

- Take a resource hyperdoctrine $\mathcal{P} : C^{op} \to \text{ResAlg}$.
- Composing with $F : \text{ResAlg} \to \text{ResSp}^{op}$ gives an indexed resource space $F \circ \mathcal{P} : C \to \text{ResSp}$. 
Putting it all together

- Take a resource hyperdoctrine $\mathcal{P} : C^{\text{op}} \to \text{ResAlg}$.
- Composing with $F : \text{ResAlg} \to \text{ResSp}^{\text{op}}$ gives an indexed resource space $F \circ \mathcal{P} : C \to \text{ResSp}$.
- Take an indexed resource space $\mathcal{R} : C \to \text{ResSp}$. 
Putting it all together

- Take a resource hyperdoctrine $\mathcal{P} : C^{op} \rightarrow \text{ResAlg}$.
- Composing with $F : \text{ResAlg} \rightarrow \text{ResSp}^{op}$ gives an indexed resource space $F \circ \mathcal{P} : C \rightarrow \text{ResSp}$.
- Take an indexed resource space $\mathcal{R} : C \rightarrow \text{ResSp}$.
- Composing with $G : \text{ResSp}^{op} \rightarrow \text{ResAlg}$ gives a resource hyperdoctrine $G \circ \mathcal{R} : C^{op} \rightarrow \text{ResAlg}$.
Duality for **BBI** gives us

- a pair of functors $F : \text{ResAlg} \rightarrow \text{ResSp}^{\text{op}}$ and $G : \text{ResSp}^{\text{op}} \rightarrow \text{ResAlg}$
- together with **natural isomorphisms** $\epsilon : \text{Id}_{\text{ResSp}^{\text{op}}} \rightarrow FG$ and $\eta : \text{Id}_{\text{ResAlg}} \rightarrow GF$. 
Putting it all together

- Take a resource hyperdoctrine $\mathcal{P} : C^{op} \rightarrow \text{ResAlg}$.
- Composing with $F : \text{ResAlg} \rightarrow \text{ResSp}^{op}$ gives an indexed resource space $F \circ \mathcal{P} : C \rightarrow \text{ResSp}$.
- Take an indexed resource space $\mathcal{R} : C \rightarrow \text{ResSp}$.
- Composing with $G : \text{ResSp}^{op} \rightarrow \text{ResAlg}$ gives a resource hyperdoctrine $G \circ \mathcal{R} : C^{op} \rightarrow \text{ResAlg}$.
- Proof by "abstract nonsense": $\mathcal{P}$ isomorphic to $GF \circ \mathcal{P}$, $\mathcal{R}$ isomorphic to $FG \circ \mathcal{R}$. 
Putting it all together

- Take a resource hyperdoctrine $\mathcal{P} : C^{\text{op}} \to \text{ResAlg}$.
- Composing with $F : \text{ResAlg} \to \text{ResSp}^{\text{op}}$ gives an indexed resource space $F \circ \mathcal{P} : C \to \text{ResSp}$.
- Take an indexed resource space $\mathcal{R} : C \to \text{ResSp}$.
- Composing with $G : \text{ResSp}^{\text{op}} \to \text{ResAlg}$ gives a resource hyperdoctrine $G \circ \mathcal{R} : C^{\text{op}} \to \text{ResAlg}$.
- Proof by "abstract nonsense":
  $\mathcal{P}$ isomorphic to $GF \circ \mathcal{P}$, $\mathcal{R}$ isomorphic to $FG \circ \mathcal{R}$.
- Morphisms: slightly more complicated - see paper!
The Duality Theorem

**Theorem**

*The categories of resource hyperdoctrines and indexed resource spaces are dually equivalent.*
Conclusions
Conclusions

- Resource hyperdoctrines generalize the syntax of Separation Logic.
- Indexed resource spaces generalize the semantics of Separation Logic.
- This work gives a complete algebraic and topological foundation for the **assertion language** of Separation Logic.
- **Duality** strengthens soundness and completeness and allows transfer of results between the two perspectives.
- This analysis is fully general: the analogous results can be given for every other **bunched logic** in the literature, both propositional and predicate (to be presented elsewhere).
Further Work

- In process: algebraic and topological metatheory proofs (first result: interpolation fails for BBI).
Further Work

- In process: algebraic and topological metatheory proofs (first result: interpolation fails for BBI).
- Next: extension with structure to interpret the Hoare triples and proof rules of Separation Logic.
Further Work

- In process: algebraic and topological metatheory proofs (first result: interpolation fails for BBI).
- Next: extension with structure to interpret the Hoare triples and proof rules of Separation Logic.
- And then: interpretation of computationally important properties like the frame rule and bi-abduction in this framework.
Further Work

- In process: algebraic and topological metatheory proofs (first result: interpolation fails for BBI).
- Next: extension with structure to interpret the Hoare triples and proof rules of Separation Logic.
- And then: interpretation of computationally important properties like the frame rule and bi-abduction in this framework.
- And possibly: **Concurrent** Separation Logic?
Further Work

▶ In process: algebraic and topological metatheory proofs (first result: interpolation fails for BBI).
▶ Next: extension with structure to interpret the Hoare triples and proof rules of Separation Logic.
▶ And then: interpretation of computationally important properties like the frame rule and bi-abduction in this framework.
▶ And possibly: Concurrent Separation Logic?
▶ Thanks for listening!