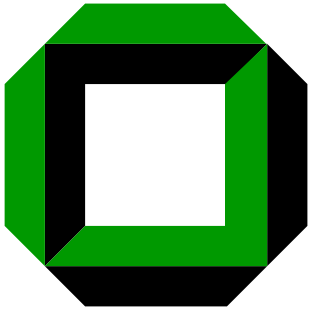
Explicit Representation of the Heap

P. H. Schmitt, Benjamin Weiß, Mattias Ulbrich

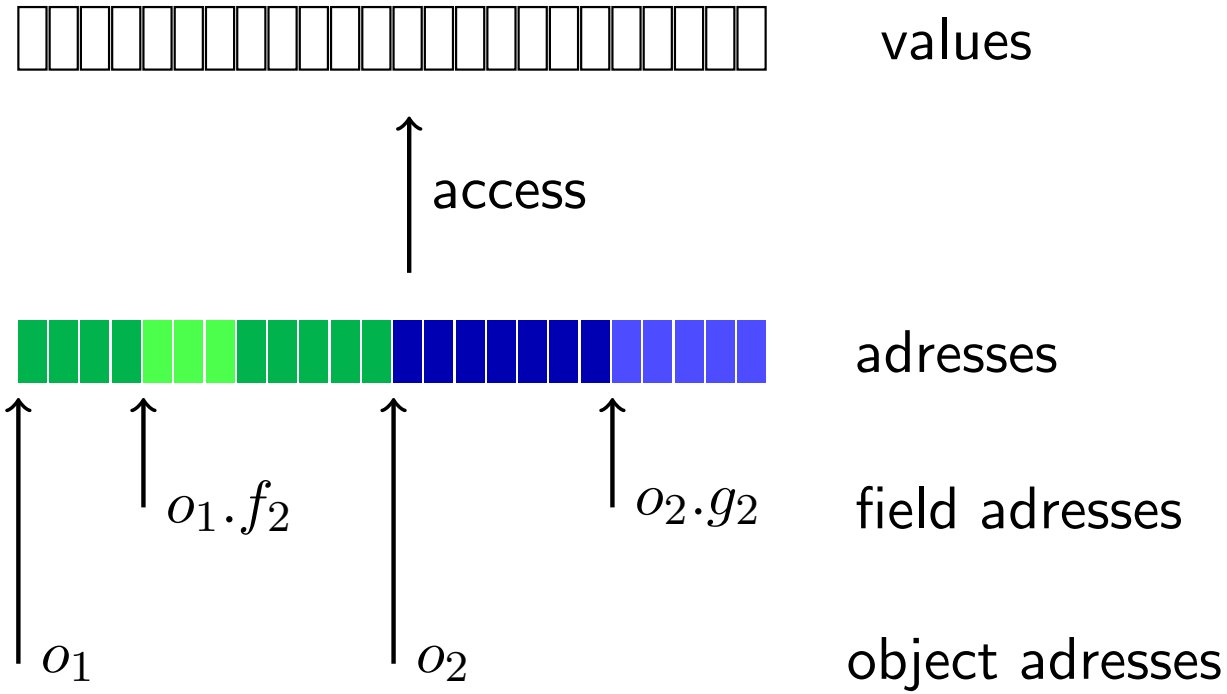
Institut für Theoretische Informatik
Fakultät für Informatik
Universität Karlsruhe (TH)



8. KeY-Symposium, May 19, 2009

Views of the Heap

Low Level View



Views of the Heap

Abstract View

v_1 v_2 v_3 v_4 v_5 values

$\uparrow$ access

$o_1.f_1$ $o_1.f_2$ $o_1.f_3$ $o_2.g_1$ $o_2.g_2$ locations

Representations of the Heap

Options

D is the universe of all objects and values.

1. For every field f there is a function $I(f) : D \to D$.
   A heap state is the collection of all functions $I(f)$.
   Used by ASM,JACK, present KeY.

Representations of the Heap

Options

D is the universe of all objects and values.

1. For every field f there is a function $I(f) : D \to D$.
   A heap state is the collection of all functions $I(f)$.
   Used by ASM,JACK, present KeY.
2. A heap state is represented by one partial function

$$\begin{array}{lcll} H & : & D \times Fields & \to & D \\ H & : & Locations & \to & D \end{array}$$

   used by Boogie,ESC2/Java,Krakatoa,KIV.

Representations of the Heap

Options

D is the universe of all objects and values.

1. For every field f there is a function $I(f) : D \to D$.
   A heap state is the collection of all functions $I(f)$.
   Used by ASM,JACK, present KeY.
2. A heap state is represented by one partial function

$$\begin{array}{lcll} H & : & D \times Fields & \to & D \\ H & : & Locations & \to & D \end{array}$$

   used by Boogie,ESC2/Java,Krakatoa,KIV.
3. Theoretically there are other options.

New Proposal

Changes to the Syntax

1. The only non-rigid symbols are constants corresponding to program variables.

New Proposal

Changes to the Syntax

1. The only non-rigid symbols are constants corresponding to program variables.
2. We consider only elementary updates with sequential and parallel composition.
   Maybe quantified and guarded updates are no longer necessary.

New Proposal

Changes to the Syntax

1. The only non-rigid symbols are constants corresponding to program variables.
2. We consider only elementary updates with sequential and parallel composition.
   Maybe quantified and guarded updates are no longer necessary.
3. There are two new types $Heap$ and $Field$ and an implicit program variable `H`.

New Proposal

Changes to the Syntax

1. The only non-rigid symbols are constants corresponding to program variables.
2. We consider only elementary updates with sequential and parallel composition.
   Maybe quantified and guarded updates are no longer necessary.
3. There are two new types $Heap$ and $Field$ and an implicit program variable `H`.
4. There are function symbols

$$\begin{array}{lcl} select & : & Heap \times Object \times Field \to Any \\ store & : & Heap \times (Object \times Field) \times Any \to Heap \end{array}$$

New Proposal

Changes to the Syntax

1. The only non-rigid symbols are constants corresponding to program variables.
2. We consider only elementary updates with sequential and parallel composition.
   Maybe quantified and guarded updates are no longer necessary.
3. There are two new types $Heap$ and $Field$ and an implicit program variable `H`.
4. There are function symbols

$$\begin{array}{lcl} select & : & Heap \times Object \times Field \to Any \\ store & : & Heap \times (Object \times Field) \times Any \to Heap \end{array}$$

New Proposal

Changes to the Syntax

1. The only non-rigid symbols are constants corresponding to program variables.
2. We consider only elementary updates with sequential and parallel composition.
   Maybe quantified and guarded updates are no longer necessary.
3. There are two new types $Heap$ and $Field$ and an implicit program variable `H`.
4. There are function symbols

$$\begin{array}{lcl} select & : & Heap \times Object \times Field \rightarrow Any \\ store & : & Heap \times (Object \times Field) \times Any \rightarrow Heap \end{array}$$

   We ignore the problem of typing $select$ for the moment.
   Maybe we can use Philip's idea for Boogie.

New Proposal

Semantics

- The states of a DL Kripke structure are given by the state vectors of the non-rigid constants (including `H`).

New Proposal

Semantics

- The states of a DL Kripke structure are given by the state vectors of the non-rigid constants (including $\mathtt{H}$).
- Non-rigid constants take values in the *computational domain* $\mathcal{D}$.

New Proposal

Semantics

- The states of a DL Kripke structure are given by the state vectors of the non-rigid constants (including $\mathtt{H}$).
- Non-rigid constants take values in the *computational domain* $\mathcal{D}$.
- $\mathcal{D}$ obviously satisfies the constant domain assumption.

New Proposal

Semantics

- The states of a DL Kripke structure are given by the state vectors of the non-rigid constants (including $\mathtt{H}$).
- Non-rigid constants take values in the *computational domain* $\mathcal{D}$.
- $\mathcal{D}$ obviously satisfies the constant domain assumption.
- If we use underspecification of partial functions this amounts to considering different computational domains.

New Proposal

Axioms

- Different constants f, g of type $Field$ have to be interpreted as different objects. Thus $f \neq g$ is an axiom.

New Proposal

Axioms

- Different constants f, g of type $Field$ have to be interpreted as different objects. Thus $f \neq g$ is an axiom.
- $\forall Heap\ h; \forall Object\ o; \forall Field\ f; \forall Any\ x;$
  $select(store(h, o, f, x), o, f) = x$

New Proposal

Axioms

- Different constants f, g of type $Field$ have to be interpreted as different objects. Thus $f \neq g$ is an axiom.
- $\forall Heap\ h; \forall Object\ o; \forall Field\ f; \forall Any\ x;$
  $select(store(h, o, f, x), o, f) = x$

New Proposal

Axioms

- Different constants f, g of type $Field$ have to be interpreted as different objects. Thus $f \neq g$ is an axiom.
- $\forall Heap\ h; \forall Object\ o; \forall Field\ f; \forall Any\ x;$
  $select(store(h, o, f, x), o, f) = x$

  This is an instance of the theory of arrays, supported by many SMT solvers.

New Proposal

Axioms

- Different constants f, g of type $Field$ have to be interpreted as different objects. Thus $f \neq g$ is an axiom.
- $\forall Heap\ h; \forall Object\ o; \forall Field\ f; \forall Any\ x;$
  $select(store(h, o, f, x), o, f) = x$

  This is an instance of the theory of arrays, supported by many SMT solvers.
  It consists of the sorts $Array$, $Index$ and $Element$, of the two function symbols $select : Array \times Index \rightarrow Element$ and $store : Array \times Index \times Element \rightarrow Array$, and the axiom:

  $\forall Array\ a; \forall Index\ i; \forall Element\ e; \forall Index\ j;$
  $select(store(a, i, e), j) \doteq if(i \doteq j) then(e) else(select(a, j))$

New Proposal

Pretty Printing

- We can render $select(\mathtt{H}, \mathtt{o}, \mathtt{f})$ as `o.f`.

New Proposal

Pretty Printing

- We can render $select(\mathtt{H}, \mathtt{o}, \mathtt{f})$ as `o.f`.
- An update $\mathtt{H} := store(\mathtt{H}, \mathtt{o}, \mathtt{f}, x)$ can be rendered as $\mathtt{o.f} := x$.

New Proposal

Pretty Printing

- We can render $\mathit{select}(\mathtt{H}, \mathtt{o}, \mathtt{f})$ as $\mathtt{o.f}$.
- An update $\mathtt{H} := \mathit{store}(\mathtt{H}, \mathtt{o}, \mathtt{f}, x)$ can be rendered as $\mathtt{o.f} := x$.
- As a generalisation of the above, an update

$$\mathtt{H} := \mathit{store}(\ldots(\mathit{store}(\mathtt{H}, \mathtt{o}_1, \mathtt{f}_1, x_1), \ldots), \mathtt{o}_n, \mathtt{f}_n, x_n)$$

can be rendered as a fake "parallel update"

$$\mathtt{o}_1\mathtt{.f}_1 := x_1 \parallel \ldots \parallel \mathtt{o}_n\mathtt{.f}_n := x_n$$

A Frist Example

Present Version

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$

A Frist Example

Present Version

$$
\begin{array}{l}
[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1) \\
\qquad\qquad\qquad\qquad \Big\downarrow \text{assignment} \\
\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)
\end{array}
\tag{1}
$$

A Frist Example

Present Version

$$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{assignment} \tag{1}$$

$$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{assignment} \tag{2}$$

$$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$$

A Frist Example

Present Version

$$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{assignment} \tag{1}$$

$$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{assignment} \tag{2}$$

$$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{update simplification} \tag{3}$$

$$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$$

A Frist Example

Present Version

$$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{assignment} \tag{1}$$

$$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{assignment} \tag{2}$$

$$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{update simplification} \tag{3}$$

$$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{empty modality, update simplification} \tag{4}$$

$$1 \doteq 1$$

A Frist Example

Present Version

$$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{assignment} \tag{1}$$

$$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{assignment} \tag{2}$$

$$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{update simplification} \tag{3}$$

$$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$$

$$\downarrow \text{empty modality, update simplification} \tag{4}$$

$$1 \doteq 1$$

$$\downarrow \text{close} \tag{5}$$

$$*$$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\quad$ assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\quad \text{assignment} \downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

$\quad \text{assignment} \downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

empty mod,
update simpl. $\downarrow$

$1 \doteq 1$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\mathit{select}(\mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2), \texttt{o}, \texttt{f}) \doteq 1$

empty mod,
update simpl. $\downarrow$

$1 \doteq 1$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\mathit{select}(\mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{g})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

empty mod,
update simpl. $\downarrow$

$1 \doteq 1$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\mathit{select}(\mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{g})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

empty mod, update simpl. $\downarrow$ $\quad$ $\mathit{if}(\mathit{false})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

$1 \doteq 1$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\mathit{select}(\mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{g})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

empty mod, update simpl. $\downarrow$

$\mathit{if}(\mathit{false})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

$\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f}) \doteq 1$

$1 \doteq 1$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\mathit{select}(\mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{g})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

empty mod,
update simpl. $\downarrow$

$\mathit{if}(\mathit{false})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

$\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{f})\mathit{then}(1)\mathit{else}(\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f})) \doteq 1$

$1 \doteq 1$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\mathit{select}(\mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{g})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

empty mod, update simpl. $\downarrow$

$\mathit{if}(\mathit{false})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

$\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{f})\mathit{then}(1)\mathit{else}(\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f})) \doteq 1$

$\mathit{if}(\mathit{true})\mathit{then}(1)\mathit{else}(\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f})) \doteq 1$

$1 \doteq 1$

A Frist Example (New Version)

$[\texttt{o.f = 1; o.g = 2;}](\texttt{o.f} \doteq 1)$ $\quad$ $[\texttt{o.f = 1; o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

assignment $\downarrow$

$\{\texttt{o.f} := 1\}[\texttt{o.g = 2;}](\texttt{o.f} \doteq 1)$

assignment $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}[\texttt{o.g = 2;}](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\{\texttt{o.f} := 1\}\{\texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

upd simpl. $\downarrow$ $\quad$ $\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1)\}\{\texttt{H} := \mathit{store}(\texttt{H}, \texttt{o}, \texttt{g}, 2)\}[](\ldots)$

$\{\texttt{o.f} := 1 \| \texttt{o.g} := 2\}[](\texttt{o.f} \doteq 1)$

$\{\texttt{H} := \mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2)\}[](\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f}) \doteq 1)$

$\mathit{select}(\mathit{store}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{g}, 2), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{g})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

empty mod, update simpl. $\downarrow$

$\mathit{if}(\mathit{false})\mathit{then}(2)\mathit{else}(\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f})) \doteq 1$

$\mathit{select}(\mathit{store}(\texttt{H}, \texttt{o}, \texttt{f}, 1), \texttt{o}, \texttt{f}) \doteq 1$

$\mathit{if}(\texttt{o} \doteq \texttt{o} \wedge \texttt{f} \doteq \texttt{f})\mathit{then}(1)\mathit{else}(\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f})) \doteq 1$

$\mathit{if}(\mathit{true})\mathit{then}(1)\mathit{else}(\mathit{select}(\texttt{H}, \texttt{o}, \texttt{f})) \doteq 1$

$1 \doteq 1$ $\quad$ $1 \doteq 1$

Benefits

- Modeling location dependent functions with the present heap representation grew into a monstrous task.

Benefits

- Modeling location dependent functions with the present heap representation grew into a monstrous task.
- In particular modeling advanced concepts like *data groups* or *dynamic frames* poses serious problems.

Benefits

- Modeling location dependent functions with the present heap representation grew into a monstrous task.
- In particular modeling advanced concepts like *data groups* or *dynamic frames* poses serious problems.
- Here is an example of a postcondition that formalizes purity of a method:

$$\forall Object\ o; \forall Field\ f; select(\mathtt{H}, o, f) \doteq select(H^{@pre}, o, f)$$