MSc Project Notes

Opale

April 6, 2024

1 Memory model constructor cheatsheet

In italic, "primitives", ie. types that aren't memory models.
Note $X^? \stackrel{\text{def}}{=} X \uplus \bot$, $X^\emptyset \stackrel{\text{def}}{=} X \uplus \emptyset$

Language	Memory Model
WISL	PMap(*Loc*, OneShot(List(Exc(*Val*))))
JSIL	PMap(*Loc*, PMap(*Str*,Exc(*Val*$^\emptyset$)) $\times$ PMap(*Loc*,Ag(*Val*))

Name	Purpose	Type	Actions	Predicates
Exc	Exclusive ownership of a specific resources	$\tau^?$	`load`, `store` *[1]	`PointsTo`
Ag	Multiple parties agree on the same value for a resource	τ		`Agree`
Frac	Allow partial (readonly) ownership of an object	$\tau \times (0,1]$		`Frac`
List	Ensure continuous memory allocation	$(\mathbb{N} \xrightarrow{fin} \tau) \times \mathbb{N}^?$ *[2]		
OneShot	The program only has one go at something (eg. freeing memory)	$\text{Exc}(\tau) + \text{Ag}(\{\emptyset\})$	`free`	
PMap	Define memory as a map of address (a sort I) to value	$(\text{I} \xrightarrow{fin} \tau) \times \mathcal{P}(I)^?$ *[3]		lift with index in-param
Product ($\times$)	Two simultaneous states, each being updated separately (eg. List)	$\tau_1 \times \tau_2$	lift with `A1`, `A2`	
Sum (+)	Either of two states existing (eg. OneShot)	$\tau_1 \uplus \tau_2$	lift with `A1`, `A2`	

*[1] Would we define `load` and `store` at this level, or at a more primitive "Value" memory model level?
*[2] Full definition: $\left\{(b, n^?) \in (\mathbb{N} \xrightarrow{fin} \tau) \times \mathbb{N}^? \mid \text{dom}(b) \subseteq [0, n^?)\right\}$
*[3] Full definition: $\left\{(h, d) \mid h \in (\text{I} \xrightarrow{fin} \tau) \wedge d \in \mathcal{P}(I)^? \wedge \text{dom}(h)^? \subseteq d\right\}$