

1 Virtual Machine Abstract Syntax

e	$::=$	x	<i>expressions</i>		
		new $\tau \{x \Rightarrow \bar{d}\}$		T	$::=$ c <i>type</i>
		$e.m(e)$			extag c
		$e.f$			datatag $\overline{p.L} \ c$
		$e.f = e$			
		let $x = e$ in e		c	$::=$ τ <i>case type</i>
		letrec \bar{d} in e			case of $p.L \ \tau$
		$e : \tau$			
		\mathcal{L}		τ	$::=$ $\{x \Rightarrow \bar{\sigma}\}$ <i>structural type</i>
		$e.\text{match } \overline{x : p.L \Rightarrow e} \text{ else } e$			$p.L$
					$\tau[L = T]$
\mathcal{L}	$::=$	<i>string</i>	<i>literals</i>		
		<i>integer</i>			
		<i>rational</i>		p	$::=$ x <i>paths</i>
					$p.f$
v	$::=$	x	<i>values</i>		
				σ	$::=$ val $f : \tau$ <i>decl type</i>
d	$::=$	val $f : \tau = v$	<i>declarations</i>		var $f : \tau$
		var $f : \tau = v$			def $m : \prod \bar{x} : \tau. \tau$
		def $m(\bar{x} : \tau) : \tau = e$			type $L = T$
		type $L = \tau$			type L
		tagtype L is T			
		delegate τ to f			

Notes on semantics:

- **letrec** can only contain **def**, **type**, and **tagtype** declarations.
- Errors (e.g. nothing to do in the default case of a match) can be handled by calling a library function taking a string argument that reports an error (with the given string) and halts (optionally, in the debugger)
- We have a separate **tagtype** L **is** T declaration for creating new tagged types. This makes creation of tags more explicit, making the semantics more obvious and the translation to lower levels simpler.
- As the type syntax for methods suggests, method types are dependent, so that the result type can depend on the argument type, and the types of later arguments can depend on earlier arguments. We need this to encode type parameters. Interestingly, first-class traits could encode this (treat a trait as a method, merge it with something that specifies the type) as could the ability to somehow create an object leaving the type abstract and then set the type (only once!)

Notes on encodings:

- encoding type parameters - see whiteboard picture from 6/2/2015
- FFI interface spec - see whiteboard picture from 5/27/2015
- encoding of tags to lower-level - see whiteboard picture from 6/2/2015