

Advances and challenges in modular semantics

Peter D Mosses
Swansea University

Seminar, SBLP 2008, Fortaleza, Brazil, 29 August 2008

Topics

- ▶ Structure of semantic descriptions
- ▶ Semantic description frameworks
- ▶ Challenges

Topics

- ▶ Structure of semantic descriptions
- ▶ Semantic description frameworks
- ▶ Challenges

Conventional descriptions

Conventional descriptions

Language semantics

Conventional descriptions

Language semantics

Syntax

**Auxiliary
entities**

Semantics

Conventional descriptions

Language semantics

Syntax

Expr

Decl

Cmd

...

Auxiliary entities

Environ

Stores

...

Semantics

Expressions

Declarations

Commands

...

Incremental descriptions

Incremental descriptions

Language semantics

Syntax

**Auxiliary
entities**

Semantics

Incremental descriptions

Language semantics

Syntax

Auxiliary
entities

Semantics

Extended language semantics

Syntax

Auxiliary
entities

Semantics

Incremental descriptions

Language semantics

Syntax

Auxiliary
entities

Semantics

Extended language semantics

Syntax

Auxiliary
entities

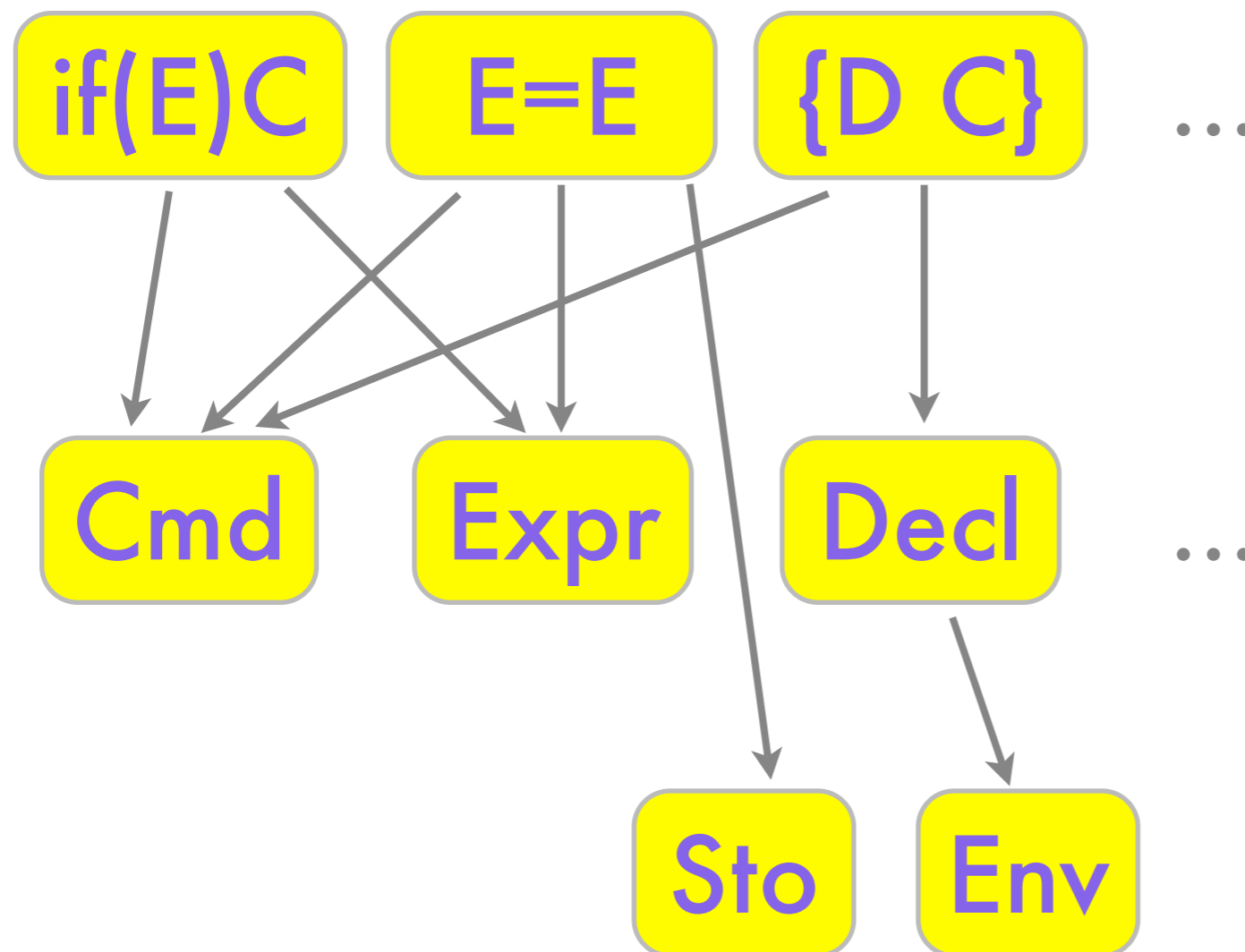
Semantics

...

Component-based descriptions

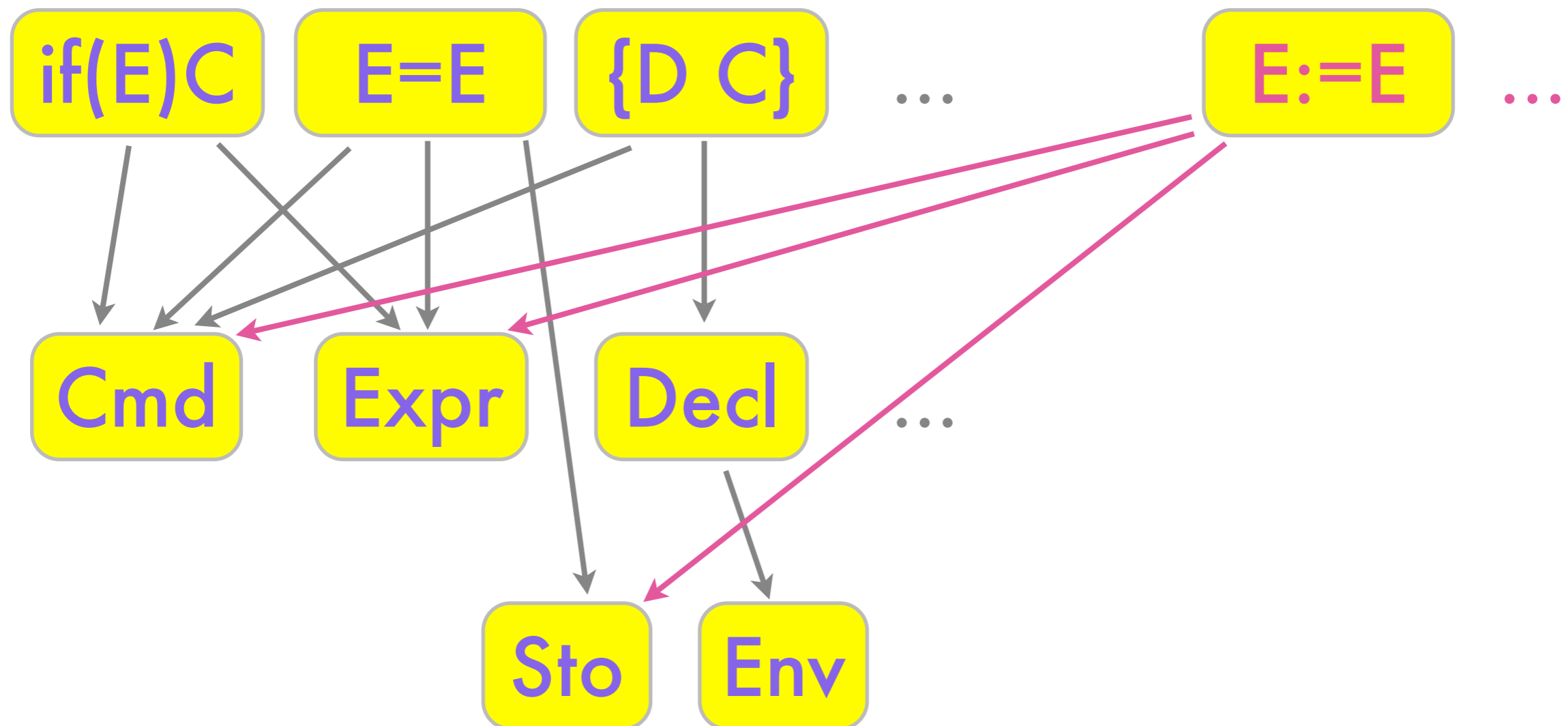
Component-based descriptions

Language semantics



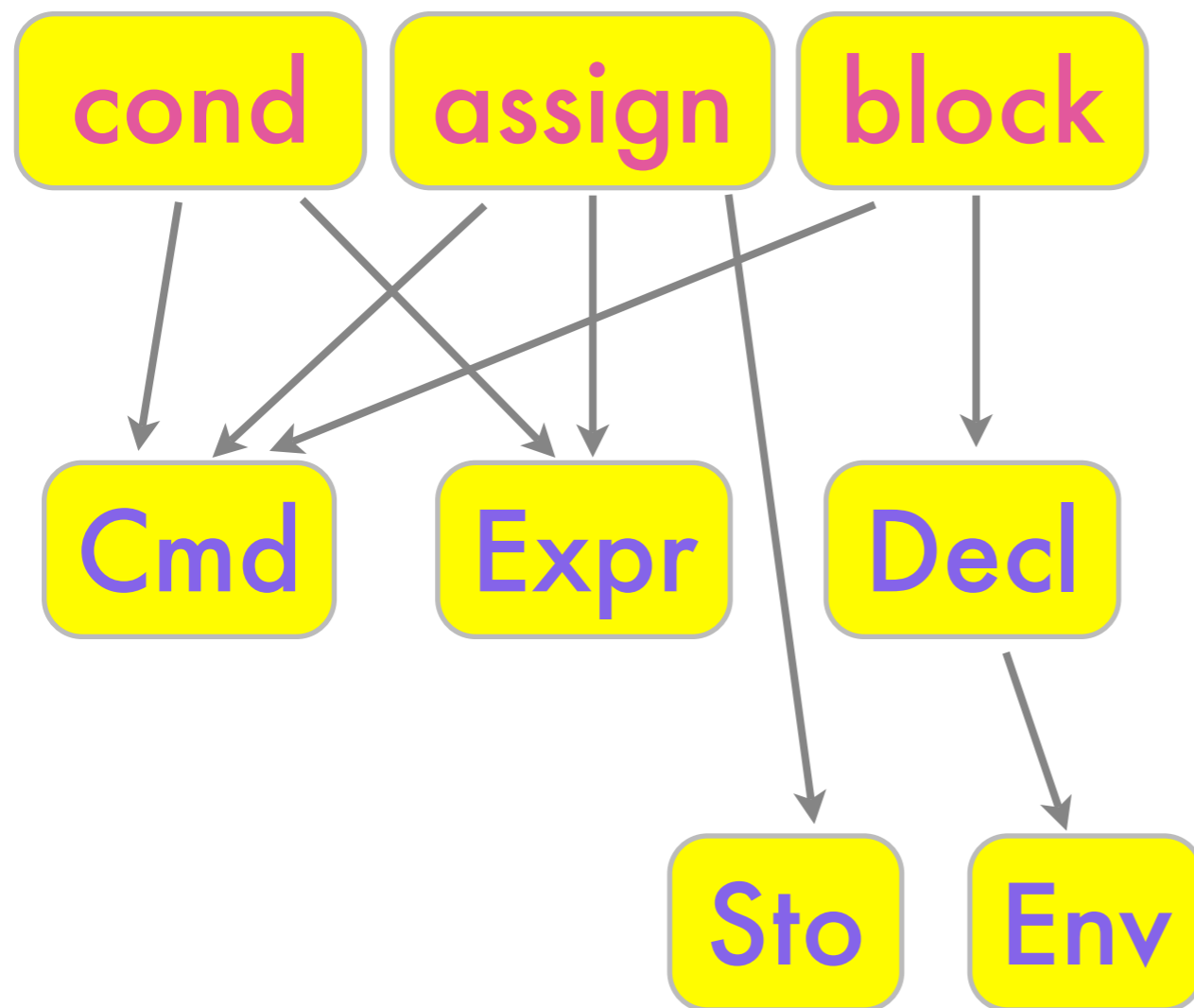
Component-based descriptions

Language semantics



Constructive descriptions

Constructive descriptions



... Abstract construct semantics

...

Constructive descriptions

if(E)C E=E {D C} ...

cond assign block ...

Cmd Expr Decl ...

Sto Env

Abstract construct semantics

Constructive descriptions

if(E)C E=E {D C} ... E:=E

cond assign block

Cmd Expr Decl

Sto Env

Abstract construct semantics

Topics

- ▶ Structure of semantic descriptions
- ▶ Semantic description frameworks
- ▶ Challenges

Denotational semantics

Original Scott-Strachey style (1971)

- ▶ conventional description structure
- ▶ **no** support for incremental semantics

Problem:

- ▶ **fixed interpretation of λ -notation**

$$\mathcal{C}[\varepsilon_0 := \varepsilon_1] = \lambda\rho.(\lambda\beta_0.(\lambda\beta_1.Assign(\beta_0|L, \beta_1)) * \mathcal{E}[\varepsilon_1](\rho)) * \mathcal{E}[\varepsilon_0](\rho)$$

Denotational semantics

VDM style (Bekič et al. 1974)

- ▶ conventional description structure
- ▶ **some** support for incremental semantics

Advance:

- ▶ changing interpretation of auxiliary notation

$$\begin{aligned} M[\text{mk-Assign}(lrs, rhs)](env) = \\ \quad \underline{\text{def}} \ l: M[lhs](env); \\ \quad \underline{\text{def}} \ v: M[rhs](env); \\ \text{STR} := \underline{c} \text{ STR} + [l \mapsto v] \end{aligned}$$

Structural operational semantics

Plotkin-style (1981), “natural” style (Kahn 1987)

- ▶ conventional description structure
- ▶ **no** support for incremental semantics

Problem:

- ▶ **explicit auxiliary information in states**

$$\frac{\rho[\rho_1] \vdash \langle C, \sigma \rangle \rightarrow \langle C', \sigma' \rangle}{\rho \vdash \langle \{ \rho_1 C \}, \sigma \rangle \rightarrow \langle \{ \rho_1 C' \}, \sigma' \rangle}$$

Abstract state machines

Gurevich et al. (mid 1980s)

- ▶ conventional description structure
- ▶ **some** support for incremental semantics

Advance:

- ▶ suppression of auxiliary entities in transition rules using imperative assignments

$\text{cond}(\blacktriangleright V_1, \beta E_2, \gamma E_3) \quad \rightarrow \quad \mathbf{if} \ V_1 \ \mathbf{then} \ pos := \beta \ \mathbf{else} \ pos := \gamma$

Evaluation contexts

Felleisen et al. (1987)

- ▶ conventional description structure
- ▶ **some** support for incremental semantics

Advance:

- ▶ suppression of auxiliary entities in reduction rules using program contexts, substitution

$$\{ \rho \ C \} \rightarrow [\rho]C$$

Action semantics

Denotational / operational (with Watt, 1985–2000)

- ▶ conventional description structure
- ▶ **good** support for incremental semantics

Advance:

- ▶ operational interpretation of auxiliary notation

execute $\llbracket \{ D C \} \rrbracket =$ furthermore *elaborate* D hence *execute* C

Monadic semantics

Moggi (1989), Plotkin & Power (2004)

- ▶ conventional description structure
- ▶ **good** support for incremental semantics

Advance:

- ▶ flexible interpretation of auxiliary notation

$$\mathcal{C}[\{ D C \}] = \mathcal{D}[D] \gg = \lambda \rho_1. \mathbf{local}(\lambda \rho. \rho[\rho_1])(\mathcal{C}[C])$$

Modular SOS

(MFCS 1999, JLAP 2004)

- ▶ conventional description structure
- ▶ **good** support for incremental semantics

Advance:

- ▶ propagation of auxiliary entities in labels

$$\frac{C \xrightarrow{\{\rho=\rho_0[\rho_1], \dots\}} C'}{\{\rho_1 C\} \xrightarrow{\{\rho=\rho_0, \dots\}} \{\rho_1 C'\}}$$

Component-based semantics

Component-based semantics

With Doh (LDTA 2001) and Iversen (IEE 2005)

- ▶ component-based description structure
- ▶ initially for use in action semantics, subsequently also in Modular SOS

Component-based semantics

With Doh (LDTA 2001) and Iversen (IEE 2005)

- ▶ component-based description structure
- ▶ initially for use in action semantics, subsequently also in Modular SOS

Menezes et al. (AAMAS 2002)

- ▶ agent-oriented action semantics

Component-based semantics

With Doh (LDTA 2001) and Iversen (IEE 2005)

- ▶ component-based description structure
- ▶ initially for use in action semantics, subsequently also in Modular SOS

Menezes et al. (AAMAS 2002)

- ▶ agent-oriented action semantics

Araújo and Musicante (SCCC 2004),

- ▶ object-oriented action semantics

Constructive semantics

(SBLP 2005)

- ▶ constructive description structure
- ▶ initially for use in action semantics, subsequently also in Modular SOS

Advance:

- ▶ mapping concrete language constructs to combinations of abstract constructs

Aspect-oriented semantics

Tirelo et al. (SBLP 2008)

- ▶ aspect-oriented description structure
- ▶ **good** support for **incremental semantics**

Advance:

- ▶ transformations of denotations

Implicitly-Modular SOS

With New (SOS 2008)

- ▶ good support for component-based and constructive semantics

Advance:

- ▶ implicit propagation of auxiliary entities

$$\frac{\rho[\rho_1] \vdash C \rightarrow C'}{\rho \vdash \{ \rho_1 C \} \rightarrow \{ \rho_1 C' \}}$$

Topics

- ▶ Structure of semantic descriptions
- ▶ Semantic description frameworks
- ▶ Challenges

The POPLmark Challenge

Pierce et al. (2005)

- ▶ Mechanized Metatheory for the Masses

Main aim:

- ▶ machine-checked definitions and proofs about programming languages
- ▶ challenge problems involve “aspects known to be difficult to formalise” – F_{\leq} with records

K-Challenge

Rosu (UIUC report 2007):

- ▶ A rewriting-based framework for computation

Main aim:

- ▶ demonstration of support for **incremental semantics** – extending a simple imperative language with output, procedural abstraction, recursion, references, call/cc, nondeterminism, aspects, concurrency, synchronization, quote

Semantics-online challenge

(2008, in preparation)

Main aims:

- ▶ demonstration of support for **independent** semantic descriptions of a collection of basic **abstract constructs**
- ▶ provide an online repository of **reusable components** in different frameworks

Discussion?

Discussion?

Modular structure

- ▶ any alternatives?

Discussion?

Modular structure

- ▶ any alternatives?

Assessment of advances in various frameworks

- ▶ any omissions or injustices?

Discussion?

Modular structure

- ▶ any alternatives?

Assessment of advances in various frameworks

- ▶ any omissions or injustices?

Challenges

- ▶ any better proposals?

References

Scott and Strachey (1971). Towards a mathematical semantics for computer languages. In *Proc. Symp. on Computers and Automata*, Microwave Research Inst. Symposia 21:19–46. Polytechnic Institute of Brooklyn.

Bekič et al. (1974). On the formal definition of a PL/I subset (selected parts). In *Programming Languages and Their Definition - Hans Bekič (1936–1982)*, LNCS 177:107–155, 1984. Full version published as Tech. Rep. 25.139, IBM Lab. Vienna, 1974. <http://homepages.cs.ncl.ac.uk/cliff.jones/ftp-stuff/>

Plotkin (1981, 2004). A structural approach to operational semantics. *J. Log. Alg. Program.*, 60-61:17–139, 2004.

Kahn (1987). Natural semantics. In *STACS'87*, LNCS 247:22–39.

Gurevich (1987). Algebraic operational semantics. In *FSTTCS'87*, LNCS 287:1-2.

Felleisen et al. (1987). A syntactic theory of sequential control. *Theoret. Comput. Sci.* 52:205-237.

Mosses and Watt (1987). The Use of Action Semantics. In *Formal Description of Programming Concepts III*, 135–166. North-Holland.

Moggi (1989). An abstract view of programming languages. Tech. Report ECS-LFCS-90-113, Computer Science Dept., University of Edinburgh.

Plotkin and Power (2004). Computational effects and operations: An overview. In *Domains VI*, ENTCS 73:149–163.

References, ctd.

Mosses (1999, 2004). Modular structural operational semantics. *J. Log. Algebr. Program.*, 60-61:195–228.

Doh and Mosses (2001, 2003). Composing programming languages by combining action-semantics modules. *Sci. Comput. Program.*, 47(1):3–36

Iversen and Mosses (2005). Constructive action semantics for Core ML. *Software, IEE Proceedings*, 152:79–98.

Menezes et al. (2002). Modular definition of agent-oriented languages using action semantics. In *AAMAS 2002*: 948-949, ACM.

Araújo and Musicante (2004). LFL: A Library of Generic Classes for Object-Oriented Action Semantics. In *SCCC'04*: 39-47, IEEE.

Mosses (2005). A constructive approach to language definition. *J. UCS* 11(7):1117-1134.

Tirelo, Bigonha and Saraiva (2008). Disentangling denotational semantics definitions. In *SBLP 2008*: 143–156.

Mosses and New (2008). Implicit propagation in structural operational semantics. In *SOS 2008, ENTCS*, to appear.

Pierce et al. (2005). The PoplMark Challenge. alliance.seas.upenn.edu/~plclub/cgi-bin/poplmark/

Roşu (2007). K: A Rewriting-Based Framework for Computations -- Preliminary version. Technical report UIUCDCS-R-2007-2926, fsl.cs.uiuc.edu