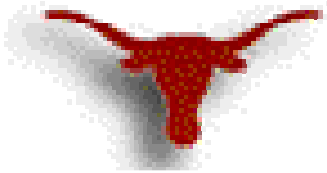
Getting Started in Programming Language Design Research

Object-Oriented and Imperative Languages



William Cook
Univertisy of Texas at Austin

More about Me

- **I've lived in nine sates**
  - MI, PA, IL, MA, FL, LA, RI, CA, TX
- **Worked many jobs**
  - HP, Apple, BAM!, Net-It, Allegis, UT
  - (ask about the $60M in venture capital)
- **Married w/8-year old son Miles**
  - My wife Robin is a user-interface designer
- **Research interests**
  - Programming languages & databases
  - Structured concurrency/workflow
  - Model-driven & feature-oriented programming

Picking Topics

- **Train your "spider senses"**
  - As undergraduates, we work around problems
    - Also true of most programming tasks
  - As grad students, focus on the problems
- **Do *something***
  … even if it doesn't seem big enough
  - In the doing, you may bump into a bigger problem
- **Solve someone else's problem**
  - Someone else = person outside PL
  - Learn about other areas in CS, and outside CS

Picking Topics

- **Best: direct contact with problem area**
  - Get your hands dirty
- **Theory**
  - Theory does not tell you what to do
  - It helps guide/constraint/analyze
- **Learn 2 or 3 things deeply**
  - Opportunities are found at the interfaces
- **Find problem and solution together**

Criteria Different by Topic

- **Types**
  - Proof of soundness (mechanically checked!)
- **Language runtime (systems)**
  - Implementation, performance
- **Language design papers**
  - Good motivation, examples
- **Analysis papers**
  - Proof, implementation, complexity, performance
- **Garbage Collection**
  - careful experiment design and measurement

Use the Tools

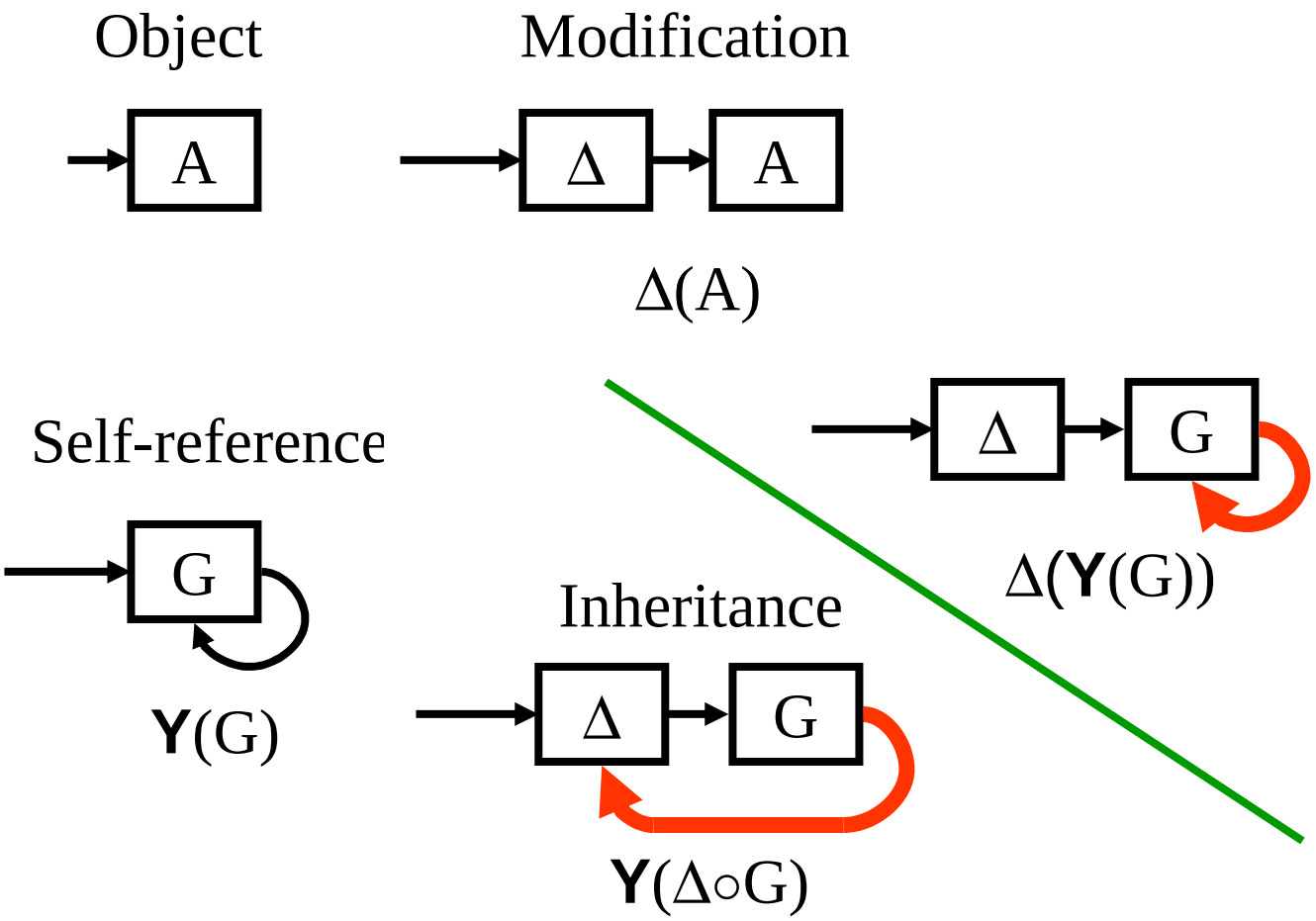
- Semantics and Analysis
- Language Theory Substrates
- Implementation Substrates
- Verification and mechanization
- (Runtimes… in another talk)
- Practical Tools
- Evaluation + Packaging

Foundation

- **Denotational Semantics**
  - good for intuition
  - λ-calculus
- **Operational Semantics**
  - Small-step
    - good for proofs
  - Large-step
    - natural, easy interpreters
    - recent proof techniques

Combination of Two Areas

- **Denotational Semantics + Objects**



The Tools

- **Abstract Interpretation**
  - compute over abstract values – types, properties, states
- **Static analysis**
  - lightweight formal methods
  - typing
  - shape analysis
  - ownership
- **Model checking**
- **Partial evaluation**

Language Substrates

- **Featherweight Java (Pierce's book)**
  - functional subset of java
- **Lightweight Java**
  - http://www.cl.cam.ac.uk/~rs456/lj/
  - imperative, true subset of java
  - comes with formalization in Isabelle/HOL
- **ClassicJava**
  - imperative
  - also support for mixins

Implementing Java Extensions

- **Polyglot**
  - Widely used
  - Complex plug-in model
- **JastAdd**
  - Newer
  - Based on declarative attribute grammars

Thought Tools

- **Galois connections**
  - loose isomorphism between ordered sets
- **Fixed points**
  - recursion and induction
- **Linear Types**
  - control over resources
- **Abstract data types and algebra**
  - Contrast with objects
- **Bisimulation**
  - equivalence of processes
- **Datalog**
  - data query/transformation
- **Attribute grammars**
  - declarative static analysis
- **Category theory**
  - theory of structure (some say: "content-free")

Language Environments

- **Implementations are convincing**
- **Ott tool**
  - Generates Isabelle/HOL specs (an Latex)
  - Includes formalization of Lightweight Java
- **PLT Redex**
  - Based on PLT Scheme
  - domain-specific language (DLS)
  - specifying and debugging operational semantics
- **Eclipse**
  - For refactoring/Development tools

$$\frac{\Gamma \vdash_{\mathtt{k}} M : \Pi^{\mathsf{par}} s{:}\sigma.\, \rho \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \forall \vec{s}{::}\hat{\bar{\sigma}}.\, \bar{\sigma}\vec{s} \to \exists \bar{\rho} \qquad \Gamma \vdash_{\mathtt{P}} N : \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \bar{N} : \Pi\bar{\Gamma}.\, \bar{\sigma}\vec{\tau}}{\Gamma \vdash_{\mathtt{k}\sqcup\mathtt{S}} MN : \rho[N/s] \;\rightsquigarrow\; \Gamma_0 \vdash \mathsf{open}\, \bar{M}\, \mathsf{as}\, \langle \vec{t_0}, x\rangle.\, \Lambda\bar{\Gamma}.\, \mathsf{open}\, x\bar{\Gamma}\, \mathsf{as}\, \langle \vec{t_1}, y\rangle.\, y\vec{\tau}(\bar{N}\bar{\Gamma}) : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \exists \vec{t}{::}\hat{\bar{\rho}}.\, \bar{\rho}[\vec{\tau}'/\vec{s}]\vec{t}}\ 46$$

$$\frac{: \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \bar{\sigma}\vec{\tau} \qquad \Gamma,\ s : \sigma \vdash_{\mathtt{k}} N : \rho \;\rightsquigarrow\; \Gamma_0 \vdash \bar{N} : \exists \vec{t'_0}{::}\hat{\kappa}'_0.\, \Pi\bar{\Gamma}.\, \forall \vec{s}{::}\hat{\bar{\sigma}}.\, \bar{\sigma}\vec{s} \to \exists \vec{t'_1}{::}\hat{\kappa}}{M, N\rangle : \Sigma s{:}\sigma.\, \rho \;\rightsquigarrow\; \Gamma_0 \vdash \mathsf{open}\, \bar{M}\, \mathsf{as}\, \langle \vec{t_0}, x\rangle.\, \mathsf{open}\, \bar{N}\, \mathsf{as}\, \langle \vec{t'_0}, y\rangle.\, \Lambda\bar{\Gamma}.\, \mathsf{open}\, x\bar{\Gamma}\, \mathsf{as}\, \langle \vec{t_1}, z\rangle.\, \mathsf{open}\, y\bar{\Gamma}\vec{\tau}z\, \mathsf{as}\, \langle \vec{t'_1}, w\rangle.\, \langle z : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \exists \vec{t'_0}{::}\hat{\kappa}'_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \exists \vec{t'_1}{::}\hat{\kappa}'_1.\, (\lambda \vec{s}{::}\hat{\bar{\sigma}}.\, \lambda \vec{t}{::}\hat{\bar{\rho}}.\, \bar{\sigma}\vec{s} \times \bar{\rho}t)\vec{\tau}$$

$$\frac{\Gamma \vdash_{\mathtt{k}} M : \Sigma s{:}\sigma.\, \rho \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, (\lambda \vec{s}{::}\hat{\bar{\sigma}}.\, \lambda \vec{t}{::}\hat{\bar{\rho}}.\, \bar{\sigma}\vec{s} \times \bar{\rho}\vec{t})\vec{\tau}\vec{\tau}'}{\Gamma \vdash_{\mathtt{k}} \pi_1 M : \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \mathsf{open}\, \bar{M}\, \mathsf{as}\, \langle \vec{t_0}, x\rangle.\, \Lambda\bar{\Gamma}.\, \mathsf{open}\, x\bar{\Gamma}\, \mathsf{as}\, \langle \vec{t_1}, y\rangle.\, \pi_1 y : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \bar{\sigma}\vec{\tau}}\ 48$$

$$\frac{\Gamma \vdash_{\mathtt{k}} M : \Sigma s{:}\sigma.\, \rho \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, (\lambda \vec{s}{::}\hat{\bar{\sigma}}.\, \lambda \vec{t}{::}\hat{\bar{\rho}}.\, \bar{\sigma}\vec{s} \times \bar{\rho}\vec{t})\vec{\tau}\vec{\tau}'}{\Gamma \vdash_{\mathtt{k}} \pi_2 M : \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \mathsf{open}\, \bar{M}\, \mathsf{as}\, \langle \vec{t_0}, x\rangle.\, \Lambda\bar{\Gamma}.\, \mathsf{open}\, x\bar{\Gamma}\, \mathsf{as}\, \langle \vec{t_1}, y\rangle.\, \pi_2 y : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \bar{\rho}[\vec{\tau}/\vec{s}]\vec{\tau}'}\ 49$$

$$\frac{\Gamma \vdash e : (\!|\sigma|\!) \;\rightsquigarrow\; \Gamma_0,\ \bar{\Gamma} \vdash \bar{e} : \exists \bar{\sigma}}{\Gamma \vdash_{\mathtt{S}} \mathsf{unpack}\, e\, \mathsf{as}\, \sigma : \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \Lambda\bar{\Gamma}.\, \bar{e} : \Pi\bar{\Gamma}.\, \exists \bar{\sigma}}\ 50$$

$$\frac{\Gamma \vdash_{\mathtt{k}} M : \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \bar{\sigma}\vec{\tau}}{\Gamma \vdash_{\mathtt{k}\sqcup\mathtt{D}} (M :: \sigma) : \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \mathsf{open}\, \bar{M}\, \mathsf{as}\, \langle \vec{t_0}, x\rangle.\, \langle \vec{t} = \lambda\bar{\Gamma}.\, \lambda \vec{t_1}{::}\hat{\kappa}_1.\, \vec{\tau}, x : \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \bar{\sigma}(t\bar{\Gamma}\vec{t_1})\rangle : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \exists \vec{t}{::}\bar{\Gamma} \Rightarrow \hat{\kappa}_1 \Rightarrow \hat{\bar{\sigma}}.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \bar{\sigma}(t\bar{\Gamma}\vec{t_1})}\ 51$$

$$\frac{\Gamma \vdash_{\mathtt{k}} M : \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \bar{\sigma}\vec{\tau}}{\Gamma \vdash_{\mathtt{W}} (M :> \sigma) : \sigma \;\rightsquigarrow\; \Gamma_0 \vdash \mathsf{open}\, \bar{M}\, \mathsf{as}\, \langle \vec{t_0}, x\rangle.\, \Lambda\bar{\Gamma}.\, \mathsf{open}\, x\bar{\Gamma}\, \mathsf{as}\, \langle \vec{t_1}, y\rangle.\, \langle \vec{t} = \vec{\tau}, y : \bar{\sigma}\vec{t}\rangle : \exists \vec{t_0}{::}\hat{\kappa}_0.\, \Pi\bar{\Gamma}.\, \exists \vec{t_1}{::}\hat{\kappa}_1.\, \exists \bar{\sigma}}\ 52$$

$$\frac{\Gamma \vdash_{\mathtt{P}} M : [\![T]\!] \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \Pi\bar{\Gamma}.\, \mathsf{Ty}\, \bar{\tau}}{\Gamma \vdash_{\mathtt{P}} M : \mathfrak{S}(M) \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \Pi\bar{\Gamma}.\, \mathsf{Ty}\, \bar{\tau}}\ 53$$

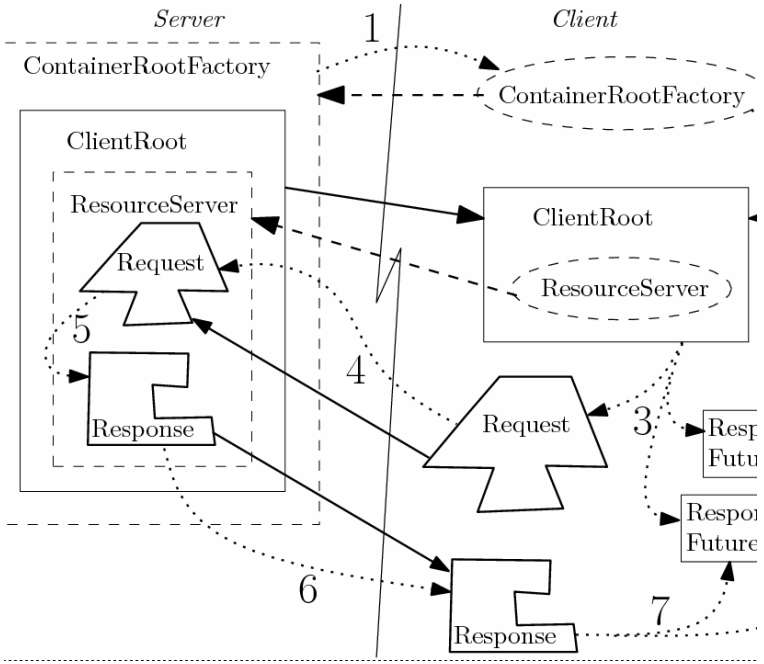
$$\frac{\Gamma \vdash_{\mathtt{P}} \lambda s{:}\sigma.\, Ms : \Pi^{\mathsf{tot}} s{:}\sigma.\, \rho \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \bar{\tau}}{\Gamma \vdash_{\mathtt{P}} M : \Pi^{\mathsf{tot}} s{:}\sigma.\, \rho \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \bar{\tau}}\ 54 \qquad \frac{\Gamma \vdash_{\mathtt{P}} \langle s = \pi_1 M, \pi_2 M\rangle : \Sigma s{:}\sigma.\, \rho \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \bar{\tau}}{\Gamma \vdash_{\mathtt{P}} M : \Sigma s{:}\sigma.\, \rho \;\rightsquigarrow\; \Gamma_0 \vdash \bar{M} : \bar{\tau}}\ 55$$

Mechanical Proof Checking

- **Theorem provers**
  - Isabelle/HOL
  - Twelf
  - ACL2
  - PVS
- **Specification languages**
  - Alloy, Maude, Z
- **POPLMark Challenge**
  - Challenge problems for "mechanizing meta-theory"

Practical Tools

- **Latex**
  - Inference Rules: Pierce's "bcprules.sty"
  - Presentations: PP or Latex??
- **Graphics**
  - OpenOffice → eps, IPE → esp
- **Use CVS for collaboration**
  - Latex too --- 5 author papers
  - Eclipse interface
- **Unison file sync**
- **Shell script!**
  - running tests
  - gathering results



Learn new skills

You are here because
you are good at something

To be successful,
need to be good
at a range of skills

Tasks

- Managing a small business
- Presenting your work
- Get funding
- Starting projects
- Accounting
- How to really dig into unfamiliar territory
- Finding topics to work on
- How to skim
- Managing a team

Understand your work style

~~

Compensate for deficiencies

my profile

- Quick thinking, sometimes too quick
- A programmer at heart
- Use theory as a tool
- Relentless, creative, intuitive
- Know what can be proven
- Proofs themselves are not easy
- Struggle with email (time management)
- Not a "born" writer, bad spelling
- How do you think I compensate?

What do you have to produce to get a PhD?

No,
it's not your thesis

It is you