Minimizing Potential “Immune Response” to Functional Programming

David Balaban, PhD
Vice President, Research & Development Informatics
Presentation Developed with the Amgen Systems Informatics Group

Commercial Users of Functional Programming
September 2008
We Have Had Varying Success in Using Functional Programming

- **Successful**
  - Using FP for scientific applications

- **Barely Started**
  - Using FP to help validate code

- **Bi-modal Success**
  - Boosting programmer morale
Some Mathematical Simulations That Used FP

- Economic process modeling and production planning
- Drug discovery process optimization via Petri-nets
- Pharmacokinetic/Pharmacodynamic (PKPD) modeling
- Software project portfolio management
Creation of Complex Models Requires Multi-disciplinary Cooperation

Scientist / Physician

Applied Mathematician

Computer Scientist

Projectile fired from cannon

Coord. system, init. Conditions, Constants

Predicted Trajectory

Newtonian Differential Equations

Continuous Dynamics

Discretization Sample time

Interpolation

Euler Difference Equations

Discrete Dynamics

Coding

Output

Computer Program

Iteration
Planning Functions are Mutually Recursive and Range Over both Time and Manufacturing Sequence.

- **Current Balance**
  - Net Demand
  - Target Balance
  - Planned Supply

- **Balance Forward**
  - Net Demand
  - Target Balance
  - Planned Supply

- **Finished Goods**
  - Successor
  - Precursor

- **Manufacturing Sequence**

- **Time**
  - Now
  - Next Period

- **Indefinite Future**
Simulating PK PD and Clinical Trials

Human

PK Differential Equations

PK+PD Differential Equations

Group of Humans

PK+PD ODE Monte Carlo Simulation

Trial Designers (Physicians)

Trial Simulator Program

NONMEM Program

Dose

Predicted Clearance

Refinement

Code

Predicted PKPD Observables

Embed parameter in a probability distribution

Model and Parameter Probability Distributions

Safety constraints, target outcome

Dosing protocol

To full trial design

Median, 5th-95th quantile Hgb (g/dL)
Software Validation Needs

- **Scientific validation**
  - Mathematical models
  - Mathematical algorithms

- **Regulatory validation**
  - Requirements and testing traceability
Rigorous Requirements Traceability Can Guarantee that “Good Behavior” is Preserved

- Requirements are Preserved
- Accuracy is Preserved
- Semantics Preserved

Diagram:

1. Projectile fired from cannon
2. Newtonian Differential Equations
3. Euler Difference Equations
4. Computer Program

Relationships:
- Coord, system, init. Conditions, Constants
- Trajectory
- Continuous Dynamics
- Discretization Sample time
- Interpolation
- Discrete Dynamics
- Coding
- Output
- Iteration
“Inoculation” -- Haskell Education

- Oxford Haskell course
- Haskell presentations
  - Jeremy Gibbons
  - Simon Peyton-Jones
- Scientific exchange
  - Galois
  - Various researchers
“Immune Response” -- Adoption of FP (Or Not)

- High curiosity about FP
- Steep learning curve created attrition from courses
- Level of abstraction was a barrier
- Fear that “non-mainstream” languages may not be supportable
- Where FP has been used, it has worked well
Next Steps for Systems Informatics

- Continue to use FP
- If possible, hire more functional programmers with science or math backgrounds
- Improve testing
  - Start using QuickCheck
- Focus more on using parallelism
END