The Problem

- Solvency II imposes new and complicated computational demands on insurers
- Software systems with forms-based interaction are not sufficiently flexible
- Current techniques do not support the full generality of actuarial models

Current Practice

1. Actuarial Models
Actuaries use continuous-time Markov models. Payment streams $b_j(t)$ are attached to states, single payments $b_k(t)$ are attached to transitions. Transition intensities $\mu_k(t)$ represent chance of state change.

$$b_0(t) = 0 \quad b_1(t) = 1_{t < \tau}$$

This product pays $1 per year in case of disability until some expiration $\tau$ and a $1 lump-sum benefit upon death prior to $g$.

2. Calculating Reserves
Using payment and risk information from the model, derive Thiele’s differential equations:

$$\frac{d}{dt} V_j(t) = \left[ r(t) + \sum_{k \neq j} \mu_k(t) \right] V_j(t) - \sum_{k \neq j} \mu_k(t) V_k(t) - b_j(t) - \sum_{k \neq j} b_k(t) \mu_k(t)$$

(In practice, most companies use a standard reference work of products)

3. Programmers Write Code
Professional software developers convert analytic solutions into software. Consequences:
- Long turnaround time
- Repeated effort for similar products
- Models must be simplified

Our Vision: the Actulus Modeling Language

The Actulus Modeling Language (AML) seeks to empower actuaries to write their own analyses and run them efficiently. The language supports actuarial concepts directly.

Actuaries Write the Code

- Static Types in AML
  Static types catch errors early:
  - Wrong state model
  - Missing transition intensities
  - Adding currency to time

- Dependent Types
  Dependent types allow library writers to give very strong guarantees:
  - A product for Cathy is used with a statistical model for Cathy, not one for Joe
  - Tracking units: dollars per year vs. dollars

Goals

- Fast turnaround — no professional programmers needed
- Use one technical artifact for different calculations and for administration, across technical platforms
- Catch errors early
- Readable notation

Ongoing Work

- Specification and formalization of the type system
- Implement AML and put it in front of real users.