Integrating Dyninst into the Computer Organization Course

Karen L. Karavanic

Portland State University

Portland State
UNIVERSITY

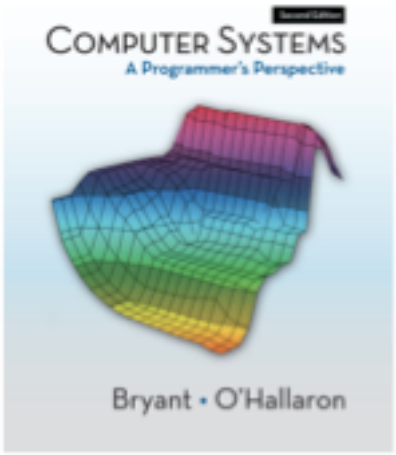
The Key Idea

- Integrate tools into the undergraduate curriculum early on
  - Students develop early habits of tool use
  - Students develop a strong mental model of the program as it changes from source code to/during runtime
  - Students exposed to research early in the degree
  - Initial focus on one course: Computer Organization (year 2)

Our Starting Point: CS 201

- CS 201: Introduction to Computer Systems from a programming perspective
- Topics: machine organization, x86 assembly and data representation, C, compile-link-load, tools
- Prerequisites: Introduction to programming, data structures
- Students' first introduction to a program as something other than source code
- Notes from the trenches: students lack motivation, conceptual models, big picture, hands on skills

Course Textbook



- Goals *seem* aligned with ours: "
  - Students should be introduced to computer systems from the perspective of a programmer, rather from the more traditional perspective of a system implementer.
  - Students should get a view of the complete system, comprising the hardware, operating system, compiler, and network.
  - Students learn best by developing and evaluating real programs that run on real machines."
- BUT
  - Tools come later – as a new skill, *not* as a way to learn the material

Desired Outcomes

- Every student will be able to use a variety of tools to investigate runtime behaviors, libraries, etc.
- A solid understanding of the contents and organization of the object file
- A solid understanding of the runtime memory organization – stack, heap, etc.
- Enough familiarity with x86 instruction set to write small routines and read and understand generated code
- Introduction of [large scale] parallelism

Some Challenges

- Translating documentation to a much lighter technical background
- Visualizations targeting the student level – stack, code changes
- Everything must be open source
- Tools must be robust enough for student lab installation and beginner student use
- Academic quarters: 10 weeks !!

Our Approach

- Add Individual Modules to standard curriculum
  - Motivational demonstration: retee first class
  - Visual and textual representations of runtime
  - Looking at portions of the dyninst code
  - StackWalker, DynC
  - Separable modules to allow more flexible adoption
- Lab exercises
  - “black box” binaries, adding instrumentation
- Assessment
  - Surveys
  - Programming skills checkpoint

Tie ins to Larger Efforts

- ACMIEEE-CS Joint Task Force Curricula 2013
  - http://ai.stanford.edu/users/sahami/CS2013/
  - Knowledge area = AR (Architecture and Organization)
  - Knowledge area = SF (Systems Fundamentals)
- NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing
  - http://www.cs.gsu.edu/~tcpp/curriculum/?q=home
  - Goal: to push knowledge of concurrency and parallel/ distributed computing into the core curriculum and start as early as possible

Tie ins to Larger Efforts

- The Flipped Classroom: Lectures can be disseminated online; interactive classroom activities can replace traditional "homework"
  - Maureen Lage, Glenn Platt, Michael Treglia, Inverting the Classroom: A Gateway to Creating an Inclusive Learning Environment. Journal of Economic Education, Winter 2000
  - Julie Foertsch, Gregory Moses, John Strikwerda, Mike Litzkow, Reversing the Lecture/Homework Paradigm Using eTEACH Web-based Streaming Video Software, Journal of Engineering Education, July 2002.
- Closed Lab for Introduction to Operating Systems
  - Enrollment approx. 130 / year
  - Class held in computer lab
  - Internet access shut down (???)
  - Cell phones and laptops BANNED
  - Instructors, tutors, students help each other

Preliminary Timeline

- First offering of CS 201 Fall 2013
  - "honors" section: average of B or better first year
  - "open lab" format
- Repeat planned Winter 2013
  - "honors" ?
  - "closed lab" format
- Dissemination (first year)
  - Publish our experiences
  - Release all materials

Thank You

- This is very early work
- Inputs and Ideas welcome
- Find me here or
  - karavan@cs.pdx.edu
  - www.cs.pdx.edu/~karavan