Concepts Come Alive through Visualization and Interaction
University of Wisconsin Oshkosh
July 16-17, 2009

Susan Rodger
Professor of the Practice
Duke University
www.cs.duke.edu/~rodger
Outline

• How I got to where I am today
• My Research Area and Projects
• Visualization Tools
  – JAWAA
  – JFLAP
• Integrating tools into the classroom
• Current Mission – Making CS accessible to K-12 students
Grew up in Raleigh, NC

- Rode motorcycles for ten years
- Played clarinet in H.S. marching band
- Loved math, puzzles
Undergraduate at NC State

• First Computer Course – first semester
  – PL/I, punch cards
  – TUCC – Triangle University Computing Center

• Other languages : Fortran, Spitball
• Last year – we got monitors!
Graduate School to the Real World

PhD, 1989
Computer Science

Assistant Prof.
1989-1994

Assistant Prof. Of Practice
to
Professor of the Practice
1994-present
Along the way, have had 2 boys

- Erich in 1997 and Markus in 1999
We have 3 cats...
Underwater Family Portrait
Hobby – Baking Shape cakes, cookies
How do you make those cakes?
What is Professor “of the Practice”?

• Position to Focus on “education in the discipline”
• Position exists in many departments at Duke
  – About 20% of Arts and Sciences Faculty
• PhD preferred, or appropriate professional experience
• Non-tenure track, permanent position, promotable
• Renewable contracts (4 –8 yrs)
• Main tasks
  – Teaching (2 courses per semester)
  – Research (related to education)
  – Service, advising
My Research Area

• Computer Science Education
• Visualization and Interaction
  – Instructional Tools for Theoretical concepts
    • Automata theory and formal languages
• Algorithm Animation
Three Projects I’m involved in

• JAWAA
  – Algorithm animation

• JFLAP
  – Software for automata theory
  – Study with 14 universities

• The Alice project
  – Create 3D virtual worlds
  – Teaching programming non-majors college
  – Teaching to K-12 (6 regional sites)
JAWAA
Java and Web-based Algorithm Animation

• Scripting Language for Animation
• Easily create, modify and move objects
• Runs over the web, no need to install
• More Advanced Students
  • Output JAWAA Command from Program
  • Animate Data Structures Easily
• SIGCSE 2003 and SIGCSE 1998
• www.cs.duke.edu/~rodger/tools/
• Students: Pierson, Patel, Finley, Akingbade, Jackson, Gibson, Gartland
Related Work

• Samba, Jsamba - Stasko (Georgia Tech)
• AnimalScript – Roessling (Darmstadt Univ of Tech, SIGCSE 2001)
• JHAVE – Naps (U. Wisc. Oshkosh, SIGCSE 2000)
• Lots of animations and systems on the web!
**JAWAA Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>circle cl 30 20 60 blue red</code></td>
<td></td>
</tr>
<tr>
<td><code>moveRelative c1 60 0</code></td>
<td>move right</td>
</tr>
<tr>
<td><code>moveRelative c1 0 50</code></td>
<td>move down</td>
</tr>
<tr>
<td><code>changeParam c1 bkgrd blue</code></td>
<td></td>
</tr>
<tr>
<td><strong>JAWAA Primitives</strong></td>
<td></td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>circle</td>
<td><img src="image" alt="circle" /></td>
</tr>
<tr>
<td>rectangle</td>
<td><img src="image" alt="rectangle" /></td>
</tr>
<tr>
<td>line</td>
<td><img src="image" alt="line" /></td>
</tr>
<tr>
<td>oval</td>
<td><img src="image" alt="oval" /></td>
</tr>
<tr>
<td>polygon</td>
<td><img src="image" alt="polygon" /></td>
</tr>
<tr>
<td>text</td>
<td>jawaa</td>
</tr>
</tbody>
</table>
JAWAA Data Structures

Array

Array people 25 25 4.2 Owen running Gail boating
Robert toys Susan cakes vert red yellow black
changeParam people index on
changeParam people[1] bkgrd white
changeParam people[0].1 text bubblesort
moveRelative people people[2] 30 0
changeParam people[2] swap people[0]
JAWAA Data Structures

• Stack

```
stack s1 200 200 4 Pop The Top Off black red
pop s1
pop s1
```

• Queue

```
queue q1 200 200 6 A 1 B 2 C 3 red blue
dequeue q1
dequeue q2
```

```
3C2B1A  3C2B1  3C2B
```
JAWAA Data Structures

- Linked List

- Trees
JAWAA Editor

- Easily create animations
- Graphically layout primitives
- Modify across time
- No knowledge of JAWAA
- Export to JAWAA file
- Start with JAWAA editor, finish with JAWAA output from program
JAWAA w/o Editor vs Editor
Nonmajors course

Spring 2001
No JAWAA Editor

Fall 2002
Using JAWAA Editor
Instructor Use of JAWAA in CS 1/2

• Use JAWAA Editor to make quick animations for lecture
  • Fast - 4-8 minutes each animations, Fall 2002 CS 2 Course

• Create quick animation of data structure in an existing program, add JAWAA commands as output

• Show web pages with JAWAA animations in lecture

• Students replay animations later
Instructor Animations for CS 2 Lecture

- How Pointers Work in Memory
- Recursion
- Shellsort
- Linked List - Insert at the Front
- Quadratic Collision Resolution
- Build Heap and Heapsort
JFLAP

- **Java Formal Languages and Automata Package**
- Instructional tool to learn concepts of Formal Languages and Automata Theory
- SIGCSE 2006 and SIGCSE 2009
- www.jflap.org

Supported by NSF Grant DUE 0442513
Formal Languages and Automata Theory

• Traditionally taught
  – Pencil and paper exercises
  – No immediate feedback

• Different
  – More mathematical than most CS courses
  – Less hands-on than most CS courses
  – Programming is in most of their CS courses, not here
Why Develop Tools for Automata?

Textual

\[\{(q_0, q_1, q_2), \{a, b\}, \delta, q_0, \{q_2\}\}\]
\[\delta = \{(q_0, b, q_0), (q_0, a, q_1), (q_1, a, q_0), (q_1, b, q_2), (q_2, a, q_1)\}\]

Tabular

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>q0</td>
<td>q1</td>
</tr>
<tr>
<td>q0</td>
<td>q0</td>
</tr>
<tr>
<td>q1</td>
<td>q2</td>
</tr>
<tr>
<td>q2</td>
<td></td>
</tr>
</tbody>
</table>

Visual

![Diagram of an automaton]

Interactive

![Interactive version of the automaton diagram]
Students Ready to learn Automata Theory!
Things start well enough ...
But soon, instead of pictures, there are **WORDS**.
Big words! The type with more than one syllable!
VIOLENCE AMONG STUDENTS AS NERVES FRAY!
We only wanted to learn automata theory! Isn’t there a better way?
Try JFLAP ...
Students Learning Automata
with JFLAP
Overview of JFLAP

• **Java Formal Languages and Automata Package**

• Instructional tool to learn concepts of Formal Languages and Automata Theory

• Topics:
  – Regular Languages
  – Context-Free Languages
  – Recursively Enumerable Languages
  – Lsystems

• **With JFLAP your creations come to life!**
Thanks to Students - Worked on JFLAP and Automata Theory Tools

- NPDA - 1990, C++, Dan Caugherty  
  Over 19 years!
- JFLAP - 1996-1999, Java version
  Eric Gramond, Ted Hung, Magda and Octavii Procopiuc
- Pâté, JeLLRap, Lsys
  Anna Bilska, Jason Salemme, Lenore Ramm, Alex Karweit, Robyn Geer
- JFLAP 4.0 – 2003, Thomas Finley, Ryan Cavalcante
- JFLAP 6.0 – 2005-2009 Stephen Reading, Bart Bressler, Jinghui Lim, Chris Morgan, Jason Lee, Jonathan Su, Henry Qin
JFLAP – Regular Languages

• Create
  – DFA and NFA
  – Moore and Mealy
  – regular grammar
  – regular expression

• Conversions
  – NFA to DFA to minimal DFA
  – NFA $\leftrightarrow$ regular expression
  – NFA $\leftrightarrow$ regular grammar
JFLAP – Regular languages (more)

• Simulate DFA and NFA
  – Step with Closure or Step by State
  – Fast Run
  – Multiple Run
• Combine two DFA
• Compare Equivalence
• Brute Force Parser
• Pumping Lemma
Why study finite automata?

• Application: Compiler
• Compiler identifies your syntax errors
• Can write a big DFA to identify all words in a Java program
  – integers, doubles, boolean
  – keywords, variable names
  – arithmetic operators, punctuation symbols
Finite Automata Editing and Simulation

- The most basic feature of JFLAP has always been the creation of automata, and simulation of input on automata.
- Here we demonstrate the creation and simulation on a simple NFA.
FA Edit & Simulation
Start up JFLAP

• When we start up JFLAP we have a choice of structures.
• The first of these is the Finite Automata!
FA Edit & Simulation
Start Editing!

- We start with an empty automaton editor window.
We create some states ...
FA Edit & Simulation
Create Transitions

- We create some transitions ...
FA Edit & Simulation
Initial and Final State

- We set an initial and final state.
- Now we can simulate input on this automaton!
When we say we want to simulate input on this automaton, a dialog asks us for the input.
FA Edit & Simulation
Start Simulation!

- When simulation starts, we have a configuration on the initial state with all input remaining to be processed.
FA Edit & Simulation
After One Step

• This is a nondeterministic FA, and on this input we have multiple configurations after we “Step.”
FA Edit & Simulation
After Two Steps

- The previous configurations on $q_1$ and $q_2$ are rejected, and are shown in red.
- The remaining uncolored configurations paths are not rejected, and are still open.
FA Edit & Simulation
After Three Steps

• Yet another step.
FA Edit & Simulation
After Four Steps

• One of the final configurations has been accepted!
FA Edit & Simulation
Traceback

• One can then see a traceback to see the succession of configurations that led to the accepting configuration.
FA Multiple Run

- Select Multiple Run
- One can then enter many strings and receive acceptance info.
JFLAP – Context-free Languages

• Create
  – Nondeterministic PDA
  – Context-free grammar
  – Pumping Lemma

• Transform
  – PDA $\rightarrow$ CFG
  – CFG $\rightarrow$ PDA (LL & SLR parser)
  – CFG $\rightarrow$ CNF
  – CFG $\rightarrow$ Parse table (LL and SLR)
  – CFG $\rightarrow$ Brute Force Parser
JFLAP – Recursively Enumerable Languages

• Create
  – Turing Machine (1-Tape)
  – Turing Machine (multi-tape)
  – Building Blocks
  – Unrestricted grammar

• Parsing
  – Unrestricted grammar with brute force parser
JFLAP - L-Systems

• This L-System renders as a tree that grows larger with each successive derivation step.
Students love L-Systems
Other Tools for Automata

• Turing’s World (Barwise and Etchemendy)
• Deus Ex Machina (Taylor and Savoiu)
• Theory of Computing Hypertextbook (Ross)
• Many others
  – L-System tools
  – Compiler tools
  – Finite State machine tools
Using JFLAP during Lecture

- Use JFLAP to build examples of automata or grammars
- Use JFLAP to demo proofs
- Load a JFLAP example and students work in pairs to determine what it does, or fix it if it is not correct.
Use of JFLAP by Instructor

Showing how to layout items

Poor:

Better:
Use of JFLAP by Instructor

Is this correct for $a^n b^n c^n$?

How do we fix it?
Use of JFLAP by Instructor

Experimenting with Difficult Concepts

Nondeterminism: \(ww^R\)

- Students attempt at desk - difficult: want to find the “middle”
- Instructor builds with class using JFLAP
Use of JFLAP by Instructor

Testing Student Programs

![Diagram of a finite automaton with transitions labeled by input symbols and states labeled as q0, q1, q2, q3. The transitions include moves to the left (L) or right (R), and the alphabet includes a, b, and c. A table to the right shows input sequences and their corresponding results: abc (Accept), aabbcc (Accept), abbc (Reject), aabbcc (Reject), aabbc (Reject), abc (Accept).]
JFLAP’s use Outside of Class

• Use with Homework problems
  – Turn in JFLAP files
  – OR turn in on paper, check answers in JFLAP

• Recreate examples from class

• Work additional problems
  – Receive immediate feedback
JFLAP Study

• Study of JFLAP’s effectiveness in learning
  – Two year study
  – Fourteen Faculty Adopters
  – Two 2-day faculty Adopter Workshops – June 2005, June 2006
  – Collect data 2005-06 and 2006-07 Academic years
  – Pretest/Posttest
  – Interviews
  – Team of three evaluators
    • Eric Weibe – Education
    • Rocky Ross – Computer Science Theory
    • Joe Bergin – Computer Science Tools
Fourteen Faculty Adopter Participants

- small, large
- public, private
- includes minority institutions

• Duke
• UNC-Chapel Hill
• Emory
• Winston-Salem State University
• United States Naval Academy
• Rensselaer Polytechnic Institute
• UC Davis
• Virginia State University
• Norfolk State University
• University of Houston
• Fayetteville State University
• University of Richmond
• San Jose State University
• Rochester Institute of Technology
We hoped to show with this learning approach...

- Students gain a better and deeper understanding of FLA
- Students are happier and more confident in learning FLA
- Students are more interested in using the tools on their own
- Instructors can easily use the tools in class
- Instructors can easily grade electronic submissions
Goals of the JFLAP Study - Formal Languages and Automata (FLA)

• Present FLA in a visual and interactive manner in addition to the more traditional approach
  – Integrated

• Present Applications of FLA

• Provide a tool for allowing students to explore FLA in a computational manner

• Provide Materials for instructors to integrate this approach in their courses
Running a Study is hard!

- Hit by the drop in enrollments in after dot-com burst
- IRBs are different process at every institution
  - One page writeup ok’d (simplest)
  - Full medical IRB (many pages)
- One institution shut down all IRB research projects – we could not use data already collected.
- One University - Control Group – different times means different types of students, different professors.
- Some faculty came to workshop and did not follow through
- There were also some fantastic faculty!
Year One Instructor Interviews

• Used JFLAP in their courses
  – Primary use in class – demonstrations
  – Some used it to generate the graphics for their lecture
  – Extensive use – homeworks – includes electronic submission
  – One used it in office hours
### Year One – Software Implementation

<table>
<thead>
<tr>
<th>Question</th>
<th>YES/NO</th>
<th>No.</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Did you use JFLAP software to study for inclass exams?</td>
<td>YES</td>
<td>20</td>
<td>55%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>16</td>
<td>45%</td>
</tr>
<tr>
<td>Did you feel you had time to learn how to use the JFLAP software?</td>
<td>YES</td>
<td>33</td>
<td>94%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>2</td>
<td>6%</td>
</tr>
<tr>
<td>Did you feel that using the software took time away from other study activities?</td>
<td>YES</td>
<td>3</td>
<td>8%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>33</td>
<td>92%</td>
</tr>
<tr>
<td>Did the time and effort it took to use JFLAP help you get a better grade in the course?</td>
<td>YES</td>
<td>23</td>
<td>64%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>13</td>
<td>36%</td>
</tr>
<tr>
<td>Was it easier to use JFLAP software or was it easier to draw it out by hand?</td>
<td>software</td>
<td>30</td>
<td>83%</td>
</tr>
<tr>
<td></td>
<td>by hand</td>
<td>6</td>
<td>17%</td>
</tr>
<tr>
<td>Did you feel you would have done as well in the course if you had not used JFLAP?</td>
<td>YES</td>
<td>18</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>13</td>
<td>36%</td>
</tr>
<tr>
<td></td>
<td>NA</td>
<td>5</td>
<td>14%</td>
</tr>
</tbody>
</table>
### Years 1 and 2: Usability Survey

<table>
<thead>
<tr>
<th>Question</th>
<th>Very Easy</th>
<th>31%</th>
</tr>
</thead>
<tbody>
<tr>
<td>How easy was it to use the drawing tool of JFLAP?</td>
<td>Easy</td>
<td>48%</td>
</tr>
<tr>
<td>(134 respondents)</td>
<td>Neither</td>
<td>15%</td>
</tr>
<tr>
<td></td>
<td>Difficult</td>
<td>5%</td>
</tr>
<tr>
<td></td>
<td>Very Difficult</td>
<td>0%</td>
</tr>
<tr>
<td>How easy was it to run the automata you designed in JFLAP?</td>
<td>Very easy</td>
<td>33%</td>
</tr>
<tr>
<td>(134 respondents)</td>
<td>Easy</td>
<td>47%</td>
</tr>
<tr>
<td></td>
<td>Neither</td>
<td>12%</td>
</tr>
<tr>
<td></td>
<td>Difficult</td>
<td>6%</td>
</tr>
<tr>
<td></td>
<td>Very Difficult</td>
<td>2%</td>
</tr>
<tr>
<td>How easy was it to interpret results from the test run in JFLAP?</td>
<td>Very Easy</td>
<td>23%</td>
</tr>
<tr>
<td>(134 respondents)</td>
<td>Easy</td>
<td>45%</td>
</tr>
<tr>
<td></td>
<td>Neither</td>
<td>19%</td>
</tr>
<tr>
<td></td>
<td>Difficult</td>
<td>10%</td>
</tr>
<tr>
<td></td>
<td>Very Difficulty</td>
<td>3%</td>
</tr>
<tr>
<td>What is your overall assessment of the JFLAP software?</td>
<td>Very Poor</td>
<td>2%</td>
</tr>
<tr>
<td>(133 respondents)</td>
<td>Poor</td>
<td>4%</td>
</tr>
<tr>
<td></td>
<td>Neither</td>
<td>11%</td>
</tr>
<tr>
<td></td>
<td>Good</td>
<td>63%</td>
</tr>
<tr>
<td></td>
<td>Very Good</td>
<td>20%</td>
</tr>
</tbody>
</table>
## Year 2 – Implementation Survey

<table>
<thead>
<tr>
<th>QUESTION</th>
<th>Time</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>When preparing for exams what percentage of study time involved the use of JFLAP software? (100 responses)</td>
<td>0-20%</td>
<td>68%</td>
</tr>
<tr>
<td></td>
<td>21-40%</td>
<td>16%</td>
</tr>
<tr>
<td></td>
<td>41-60%</td>
<td>11%</td>
</tr>
<tr>
<td></td>
<td>61-80%</td>
<td>3%</td>
</tr>
<tr>
<td></td>
<td>81-100%</td>
<td>2%</td>
</tr>
<tr>
<td>How often did you use JFLAP to do additional practice problems? (99 responses)</td>
<td>Never</td>
<td>46%</td>
</tr>
<tr>
<td></td>
<td>Rarely</td>
<td>25%</td>
</tr>
<tr>
<td></td>
<td>Occasionally</td>
<td>21%</td>
</tr>
<tr>
<td></td>
<td>Often</td>
<td>4%</td>
</tr>
<tr>
<td></td>
<td>Very Often</td>
<td>4%</td>
</tr>
</tbody>
</table>
## Year 2 – Usability Survey

<table>
<thead>
<tr>
<th>Question</th>
<th>Strongly Agree</th>
<th>Agree</th>
<th>Neither</th>
<th>Disagree</th>
<th>Strongly Disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using JFLAP made the course more enjoyable for me. (98 responses)</td>
<td>12%</td>
<td>51%</td>
<td>25%</td>
<td>6%</td>
<td>4%</td>
</tr>
<tr>
<td>Using JFLAP made me feel more engaged in the course. (98 responses)</td>
<td>13%</td>
<td>59%</td>
<td>15%</td>
<td>9%</td>
<td>3%</td>
</tr>
<tr>
<td>Having access to JFLAP made learning course concepts ... (97 responses)</td>
<td>Much harder</td>
<td>Harder</td>
<td>Neither</td>
<td>Somewhat easier</td>
<td>Much easier</td>
</tr>
<tr>
<td></td>
<td>1%</td>
<td>5%</td>
<td>26%</td>
<td>54%</td>
<td>14%</td>
</tr>
</tbody>
</table>
Getting interaction into the automata theory course......

There are lots of ways to get interaction in this course...
Students Work in Groups to Solve Problems – With or W/out JFLAP

• Lecture some, then stop
• Students work on problem with JFLAP
• Bring students back together
Interaction in Class – Props
Edible Turing Machine

- TM for $f(x)=2x$ where $x$ is unary

- TM is not correct, can you fix it? Then eat it!

- States are blueberry muffins
Students building DFA with cookies and icing
JFLAP’s Use Around the World

- JFLAP web page has over 220,000 hits since 1996
- Google Search
  - JFLAP appears on over 38,000 web pages
  - Note: search only public web pages
- JFLAP been downloaded in over 160 countries
Conclusions From Study

• Results of Study showed
  – All the faculty used JFLAP in their courses, mostly for homework, some in lecture
  – Students had a high opinion of JFLAP
  – Majority of students felt access to JFLAP
    • Made learning course concepts easier
    • Made them feel more engaged
    • Made the course more enjoyable
  – Over half the students used JFLAP to study for exams
  – Over half the student thought time and effort using JFLAP helped them get a better grade.
Resources/Questions?

JFLAP book

www.jflap.org

JFLAP tutorial