Experiences with a Tablet PC Based Lecture Presentation System in Computer Science Courses

Richard Anderson  
University of Washington

Ruth Anderson  
University of Virginia

Beth Simon  
University of San Diego

Steven A. Wolfman  
University of Washington

Tammy VanDeGrift  
University of Washington

Ken Yasuhara  
University of Washington

Slides Projected from Computer

Pros:
- High quality materials
- Ease of sharing and re-use
- Ability to switch to other computer applications

Cons:
- Limits Flexibility of Presentation
  e.g. respond to student questions, work out examples in real time, integrate student input into presentation

Classroom Presenter

• Tablet-PC
• Slide-Based
• Distributed Application

• First deployed in Summer 2002
• Deployed in > 40 Computer Science courses at UW, UVA, USD, and elsewhere.

• Available free for educational and research use:  
  www.cs.washington.edu/education/dl/presenter/

Related Work

• Lecturer's Assistant [Buckalew]
• eClass [Abowd, Brotherton]
• TMS [Golub]
• DEBBIE/DyKnow [Berque]

• PowerPoint with Tablet extensions
• Tablet PC Journal

Presenter Configurations

• Single Machine (1 Tablet PC)
  - Direct projection from instructor view
    - Single view, tethered
  - Projection of second monitor from tablet
    - Multiple views, but tethered

• Multiple Machine (1 Tablet PC + Other devices)
  - Wireless connection to display view
    - Multiple views, untethered
    - Distance learning and integration with student devices

Outline

• Problem Addressed
• Related Work
• System Description
• Deployments in CS Courses
• Examples of Use
Instructor View

Projector View

Presenter Deployments

> 40 different classes at UW, UVA, USD, and elsewhere
> 20 different Instructors
> 2000 students (Class sizes: 7 - 211 students)


Presenter in Distance Course

Archived Lecture of Instructor Using Presenter

Student Survey Results

- **Less**
- **No Change**
- **More**

<table>
<thead>
<tr>
<th></th>
<th>attention to lecture</th>
<th>understanding of material</th>
<th>encourage instrs to use</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Survey Results</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Examples of Use

- Instructor Notes
- Interactive Writing
- Diagrams
- Attention Mechanisms

Instructor Notes

- Private notes viewable only by instructor
  - Reminder of important points to emphasize
  - Contents of blank spaces to fill in
  - Answers to math calculations

Interactive Writing

\[ ET = IC \times CPI \times 1/CR \]
- CT doubling the GHz doesn’t double the SPEC number
- IC: Bigger improvement on P4 on FP
- SSE/2 instruction set - stack registers to regular FP register set had to recompile to use these instruction sets

SPEC on Pentium III and Pentium 4

- What do you notice?

SPEC on Pentium III and Pentium 4

- What do you notice?

Stack FP registers = “regular” registers
**Whiteboard Feature**

```
A=62, B=3, C=2
A=B=B=0, C=2
A=1, B=0, C=3
A=1, B=0, C=2
A=3, B=3, C=55
A=2, B=4, C=3
```

**Compound Boolean Expressions**
- Use `&` to AND two or more conditions
- Use `||` to OR two or more conditions
- For example, write a test to see if
  - B is either 0 or between the values of B and C: `(B == 0) || (A <= B && B < C)`

<table>
<thead>
<tr>
<th>Check values</th>
<th>Expression</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>A=1, B=2, C=3</td>
<td>`(B == 0)</td>
<td></td>
</tr>
<tr>
<td>A=1, B=0, C=3</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>A=1, B=0, C=2</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>A=3, B=3, C=55</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>A=2, B=4, C=3</td>
<td>F</td>
<td></td>
</tr>
</tbody>
</table>

**Sparse Slides**

```
if B
    S1;
else
    S2;
```

**Walking Thru Examples**

```
Memory

<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0000</td>
</tr>
<tr>
<td>01</td>
<td>000F</td>
</tr>
<tr>
<td>02</td>
<td>0005</td>
</tr>
<tr>
<td>03</td>
<td>3001</td>
</tr>
<tr>
<td>04</td>
<td>5002</td>
</tr>
<tr>
<td>05</td>
<td>-0000</td>
</tr>
</tbody>
</table>

IR: 0000 0014 F 10011 3001 5002 0000
Acc: 0000 0000 E +0000 5 0 0 0 1 4
0000 Store 000
```

**Recursive Factorial**

```
int factorial(int n)
{
    if (n == 0)
        return 1;
    else
        return n * factorial(n-1);
}
```
Preparation for Lab: Swapping a value
Write code to swap the values stored in red and yellow.

```java
int yellow = 5;
int blue = 10;
int friend = yellow;
blue = 10;
yellow = blue;
friend = yellow;
```

Dynamic Corrections

```
ree \rightarrow \delta
\{fre, free\}
five
\rightarrow \delta
\{f\}
```

Diagramming

- Annotate interactively with ink
  - To draw attention to details
  - To trace a process

Marginal Diagrams

- Signal Processing
- Language Model (LM)
- Viterbi Algorithm

Rich Diagrams

```
Markov Blanket Sampling
```

```
Iw without the displacement
```

```
Jump Branch MemRead MemWrite
ALUsrc ALUop MemToReg RegWrite RegDest
```
Attention Mechanisms

- Draw attention to specific content on the slides in a persistent manner.
  - Checks
  - Underlines
  - Circles
  - Arrows
  - Grouping related points
- Focus attention

Summary

- System Implemented
- Substantial deployment in CS courses
- Examples of Usage

Classroom Presenter

Available free for educational and research use:
www.cs.washington.edu/education/dl/presenter

Ruth Anderson  ruth@cs.virginia.edu