



# Virtual Laboratories in Education

Ivan Marsic

*Department of Electrical and Computer Engineering  
and Center for Advanced Information Processing (CAIP)*

*Rutgers — The State University of New Jersey*

*Piscataway, NJ 08854-8058 USA*

<http://www.cai.p.rutgers.edu/~marsic>

# Goals

---

- **Web-based Virtual Laboratories**
  - Distance education
  - Available any-time, any-place, any number of repetitions
  - Learn and practice in spite of making errors
- **Development & Deployment Dynamics**
  - Short-term goal: supplement for actual labs  
(preparation & rehearsal)
  - Long-term goal: virtual labs substitute actual labs
- **Development Emphasis**
  - Look-and-feel Fidelity vs. Learning Concepts

# Technical Challenges

---

- **Software Architecture**
  - Web-based (Java Applets)
  - Extensible; Multipurpose; Rapid development
- **Cost vs. Quality of Service**
  - Visualization fidelity; Interaction agents
  - Responsiveness
- **Framework for Application Development**
  - Application specification language; End-user programming
  - Easy modifications
- **Human Factors**
  - Usability
  - Educational impact

# Related Work

---

## Existing Interactive Virtual Labs:

- **Non-Web-based or require plug-in**
  - California State University's Center for Distributed Learning
  - The University of Melbourne's Science Media Teaching Unit
  - Edmark, Inc.
  - Olympus America Inc. and The Florida State University
- **Java Applets**
  - Hughes Medical Institute: Bio-Interactive
  - University of Colorado (Boulder): Physics 2000
  - TeleLearning Network of Centers of Excellence

# Related Work (Cont'd)

---

## Summary:

- Not Web-based or require plug-in
  - Not platform-independent Java Applets
- No generic software architecture
- Interaction by “clicking” rather than direct manipulation
- “Linear” interaction
  - All users have the “exact” same lab experience
  - Our goal: Users have different experience, but learn the same concepts

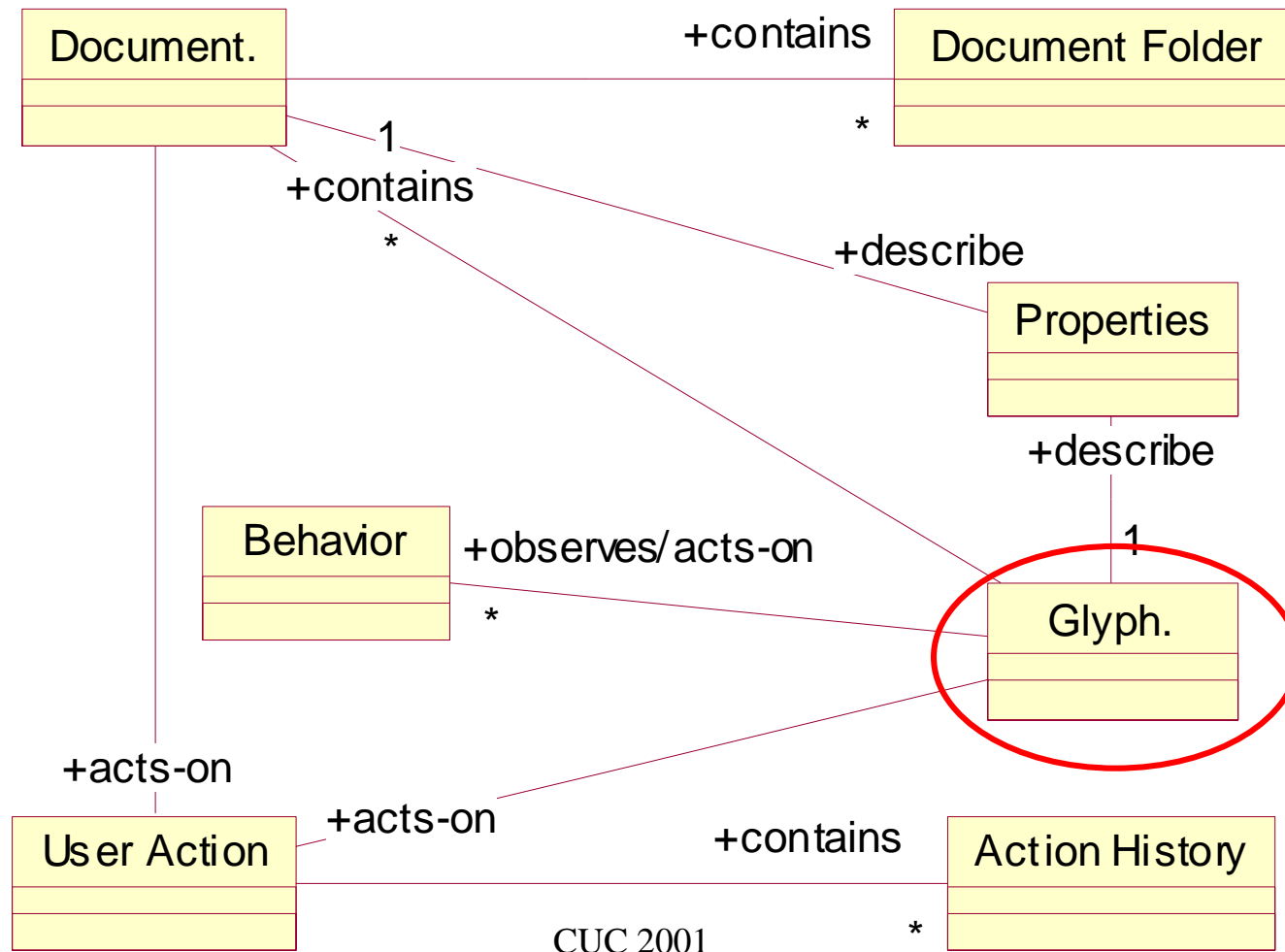
# Assumptions

---

- Any interactive application can be seen as a document editor
  - Editing values of parameters of application objects
  - Directly “grasping” screen visualizations of application objects and rotating, translating, or otherwise manipulating them

# Generalized Editor

## Conceptual design: Editing structured documents

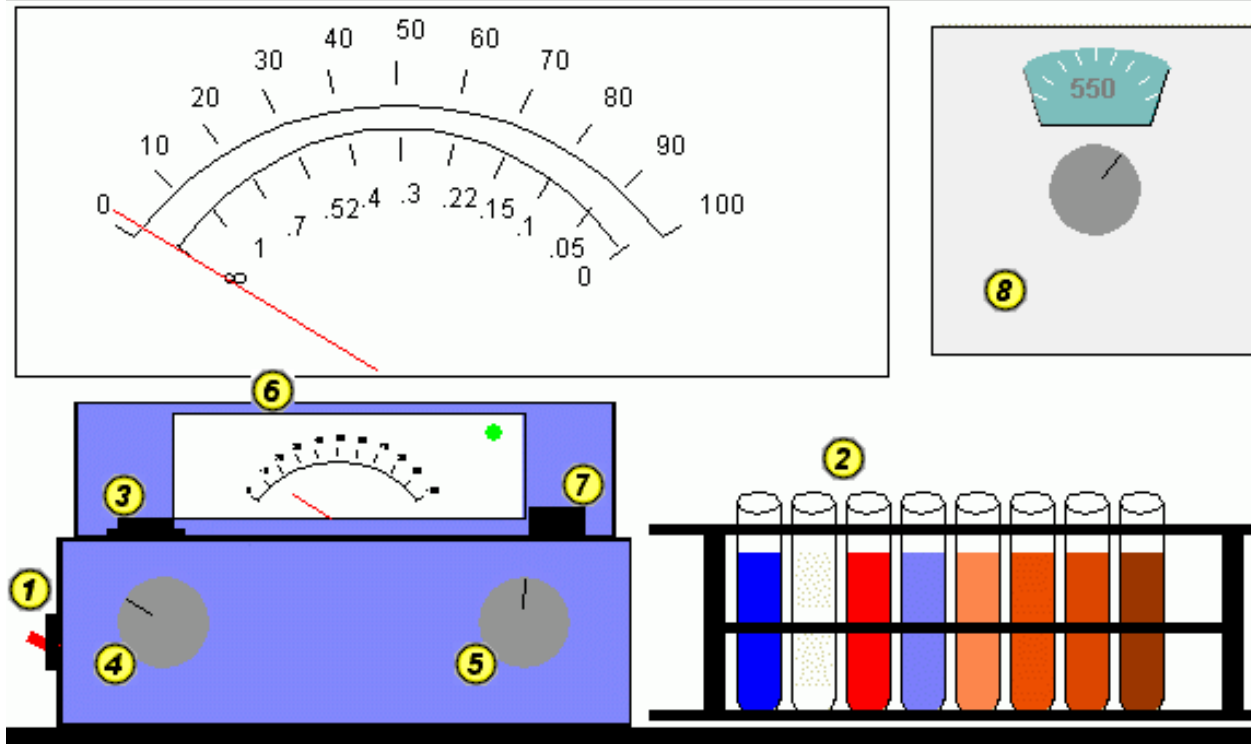


# Spectrophotometer Lab

What ho! Welcome to Virtual Spectrophotometry. Begin by setting the Wavelength. For this, click on the wave dial to obtain a top view. Rotate the wavelength control until the desired wavelength (in nanometers) is indicated by the wavelength scale. Now click on the switch to turn it on. Bring the meter needle to infinity on the absorbance scale by adjusting the dark control.

Back

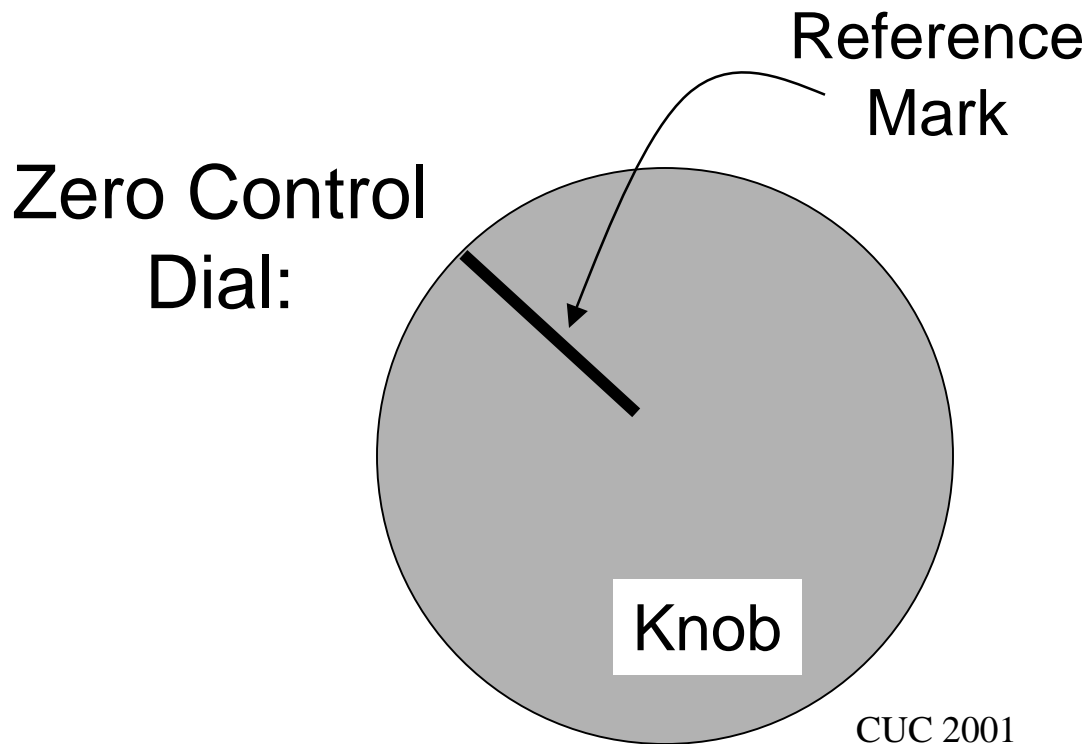
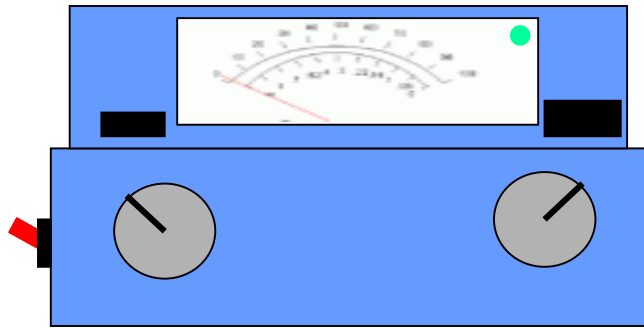
Next



Measures the concentration of a substance in a solution and displays the % transmittance of light received by the photocell

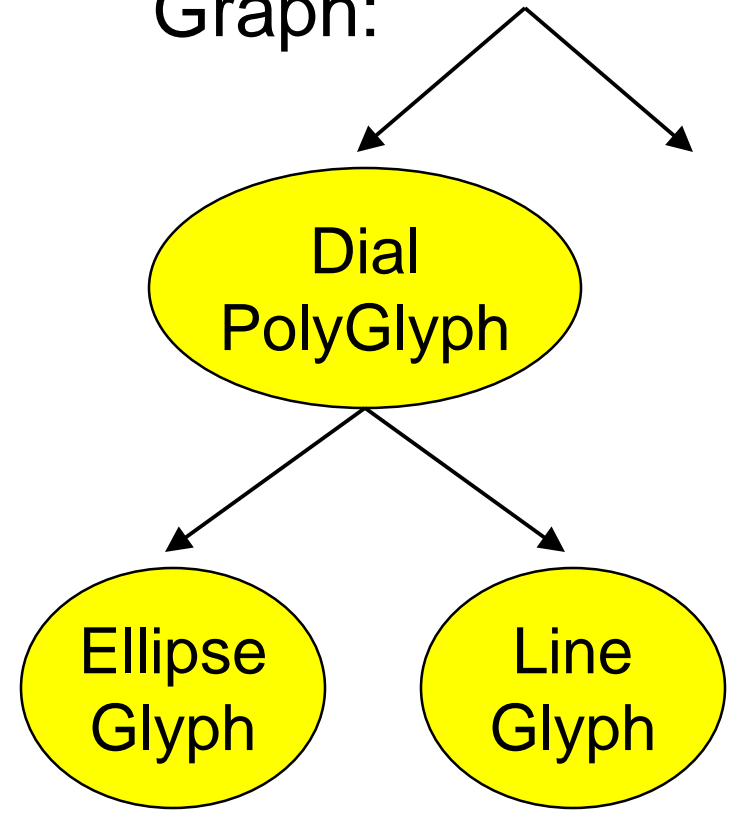


# Building Spectro Lab GUI



CUC 2001

Scene Graph:



# SpectroLab GUI Programming in XML

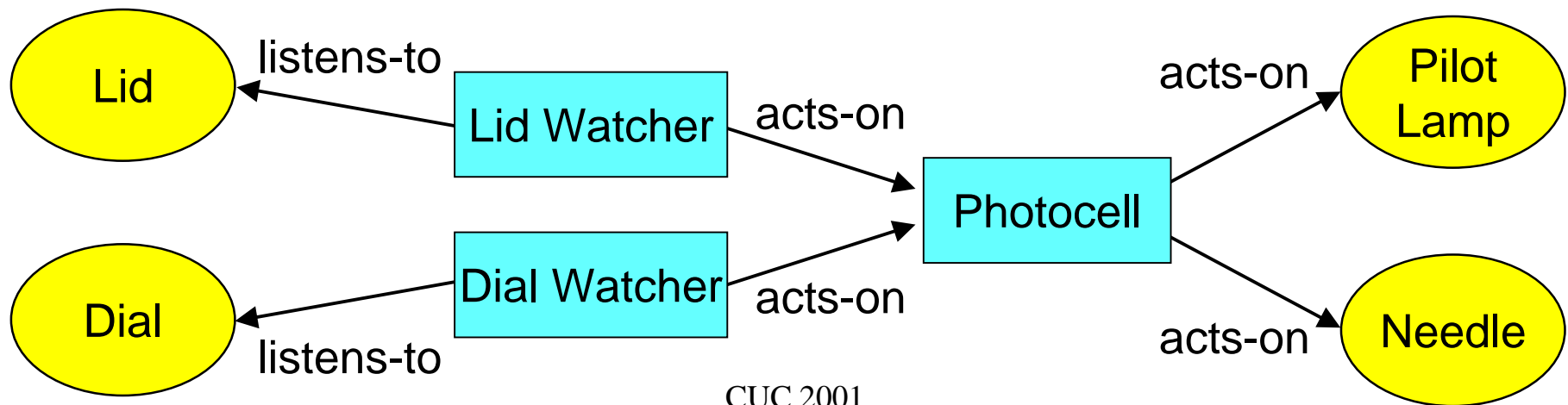
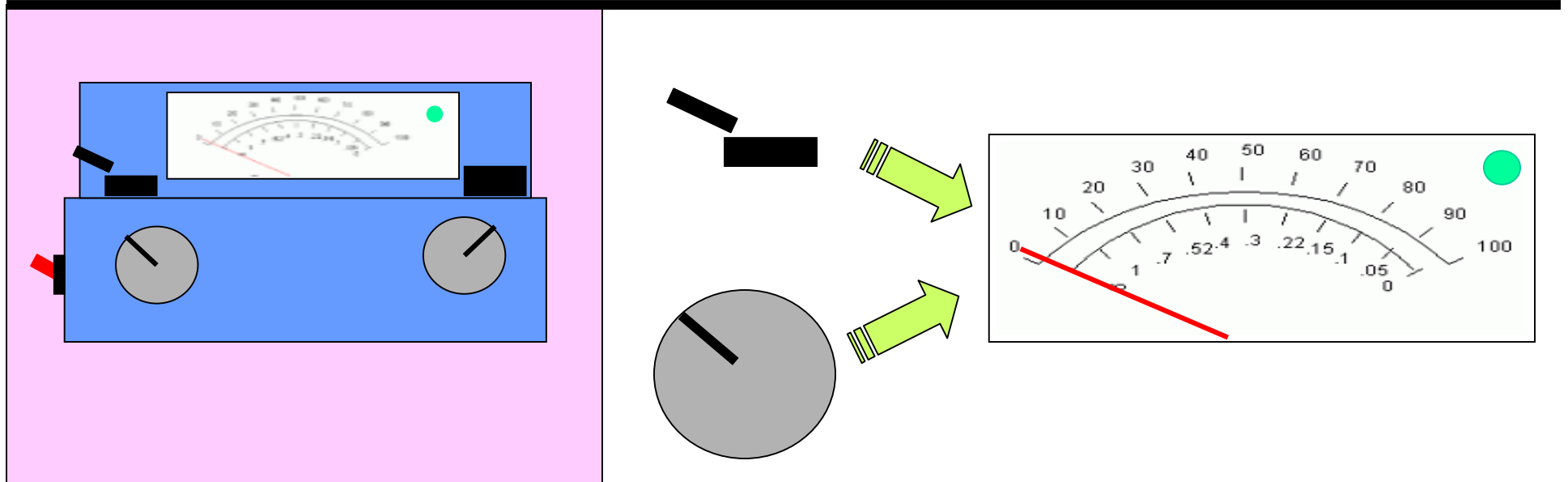
---

```
<POLYGLYPH id="zeroDial" type="flatscape.domain.PolyGlyph2D">
  <PROPERTY name="glyph.permittedUserTransform"
    type="java.lang.String" value="rotate" />
  <PROPERTY name="glyph.dialType" type="java.lang.String" value="zer
  <TRANSFORMATION type="flatscape.domain.Transform2D"
    value="79.0 495.0 1.0 1.0 0.0 0.0 6.5" />

<GLYPH id="zeroDialKnob" type="flatscape.domain.EllipseFigure">
  <PROPERTY name="glyph.height" type="java.lang.Double" value="42.0" /
  <PROPERTY name="glyph.width" type="java.lang.Double" value="42.0" />
  <PROPERTY name="fill.color" type="java.awt.Color"
    value="java.awt.Color[r=150,g=150,b=150]" />
  <TRANSFORMATION type="flatscape.domain.Transform2D"
    value="0.0 0.0 1.0 1.0 0.0 " />
</GLYPH>

<GLYPH id="zeroDialReferenceMark" type="flatscape.domain.LineFigure
  <PROPERTY name="glyph.length" type="java.lang.Double" value="13.5" /
  <TRANSFORMATION type="flatscape.domain.Transform2D"
    value="0.0 -13.0 1.0 1.0 -1.57 0.0 0.0"/>
</GLYPH>
</POLYGLYPH>
```

# Specifying Spectro Behavior



# Spectro Behavior Programming in XML

---

```
<BEHAVIOR id="photocell" type="biology.spectro.domain.Photocell">
  <TARGET name="pilotLamp" ref="pilotLamp" />
  <TARGET name="needle" ref="needle" />
</BEHAVIOR>

<BEHAVIOR id="opening" type="biology.spectro.domain.LidWatcher">
  <LISTENER type="manifold.domain.PropertyChangeListener"
    source="sampleHolder" />
  <TARGET name="lightMeasure" ref="photocell" />
</BEHAVIOR>

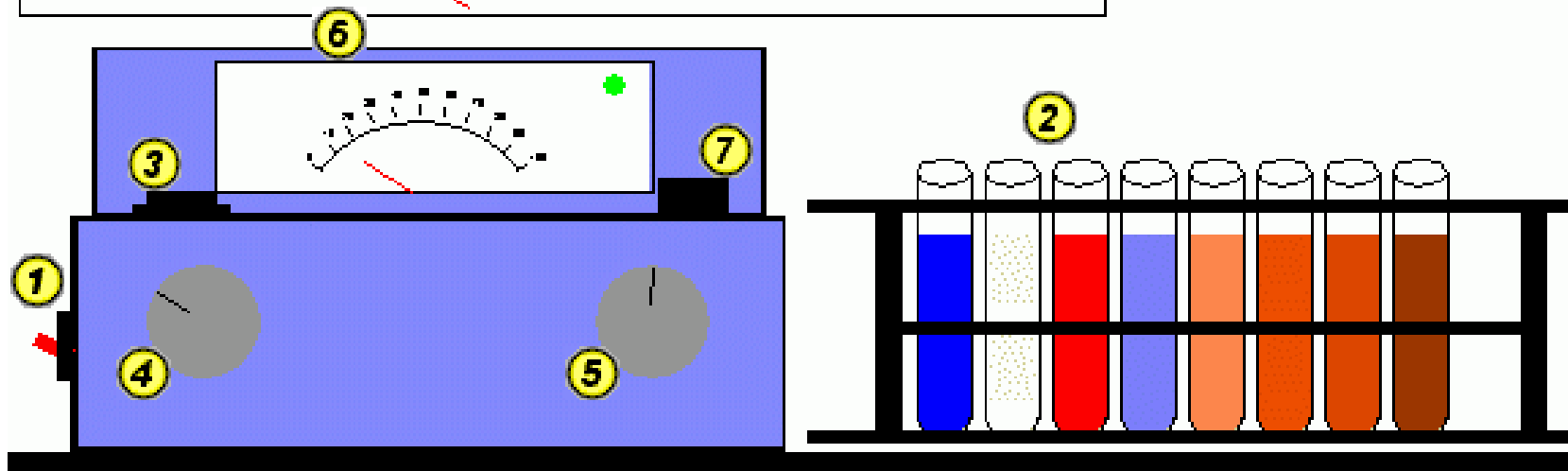
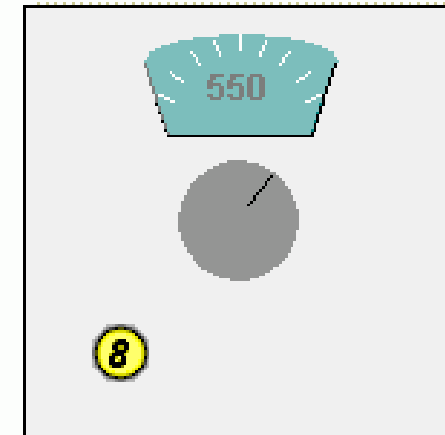
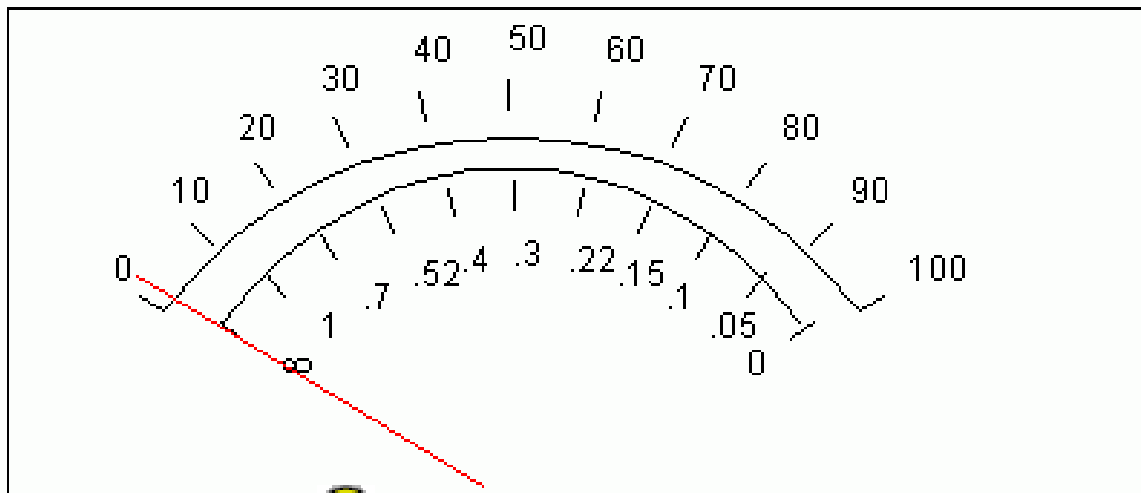
<BEHAVIOR id="turning" type="biology.spectro.domain.DialWatcher">
  <LISTENER type="manifold.domain.TransformListener" source="lightDial" />
  <LISTENER type="manifold.domain.TransformListener" source="zeroDial" />
  <TARGET name="lightMeasure" ref="photocell" />
</BEHAVIOR>
```

# Spectrophotometer Lab (2)

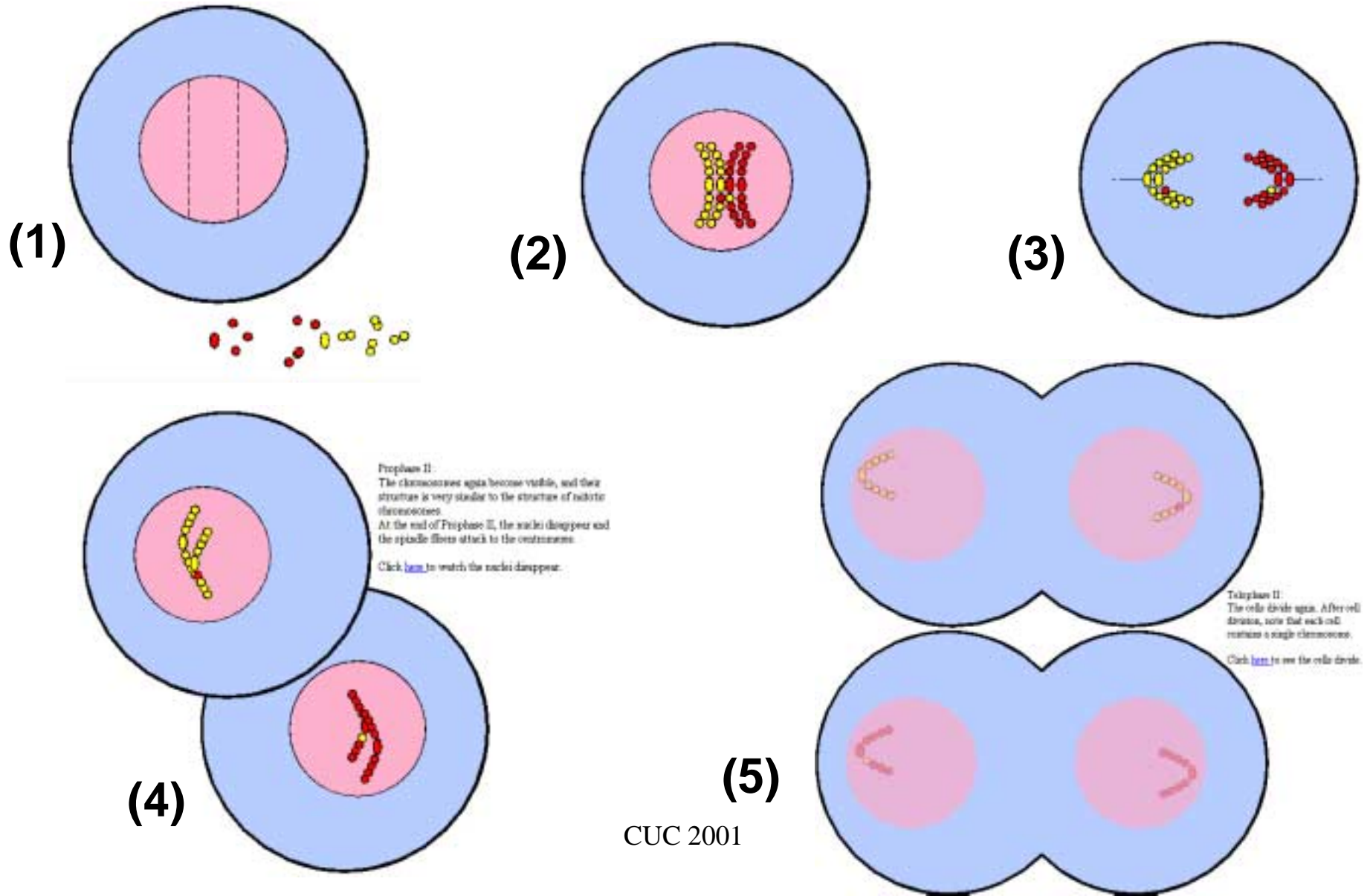
What ho! Welcome to Virtual Spectrophotometry. Begin by setting the Wavelength. For this, click on the wave dial to obtain a top view. Rotate the wavelength control until the desired wavelength (in nanometers) is indicated by the wavelength scale. Now click on the switch to turn it on. Bring the meter needle to infinity on the absorbance scale by adjusting the dark control.

Back

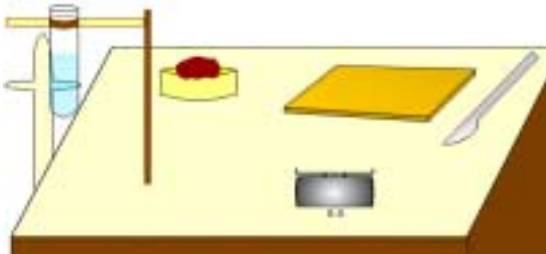
Next



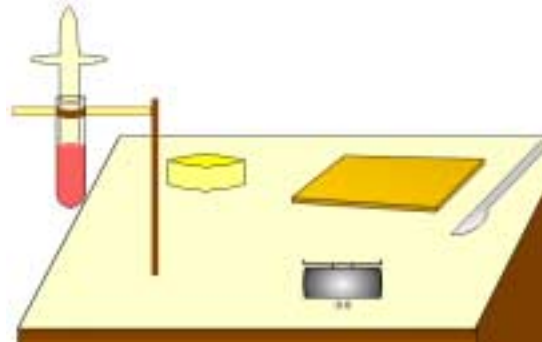
# Meiosis Lab



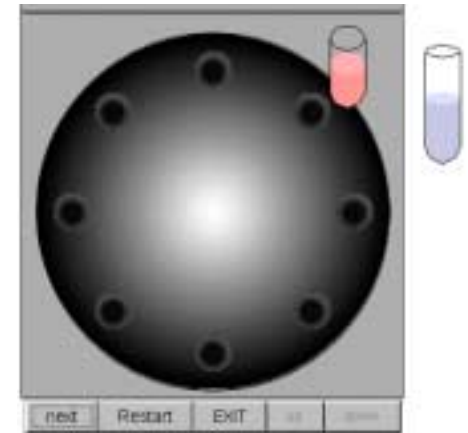
# Differential Centrifugation Lab



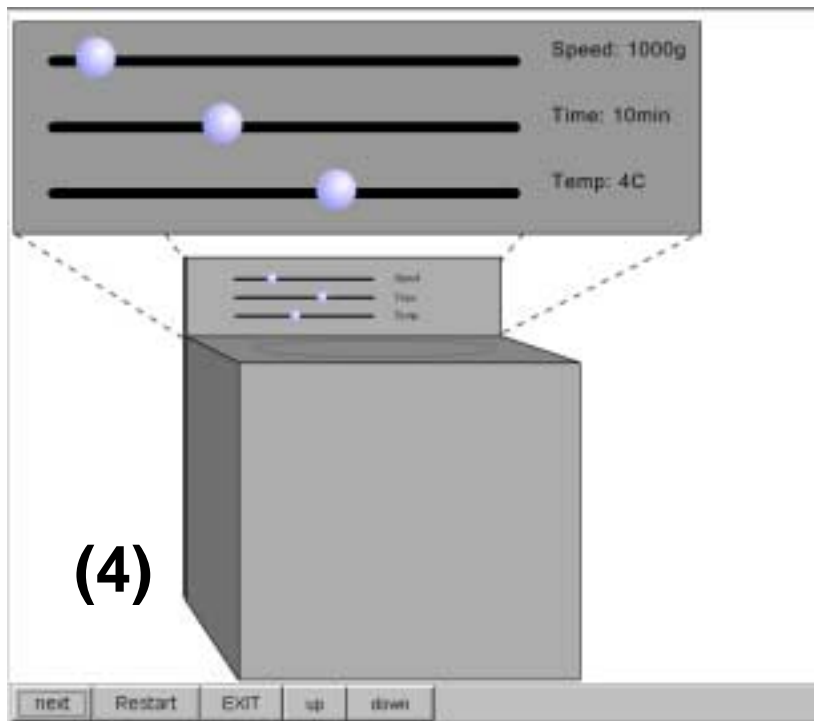
(1)



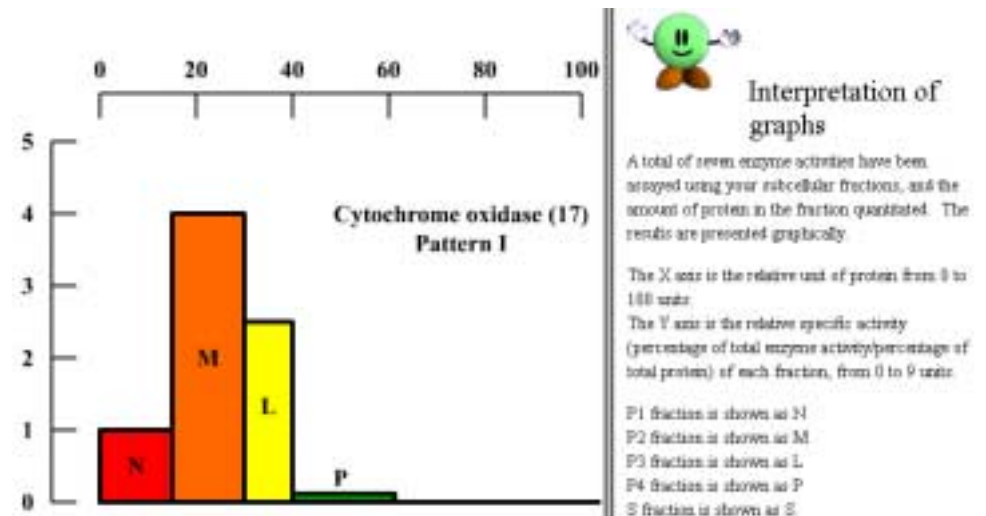
(2)



(3)

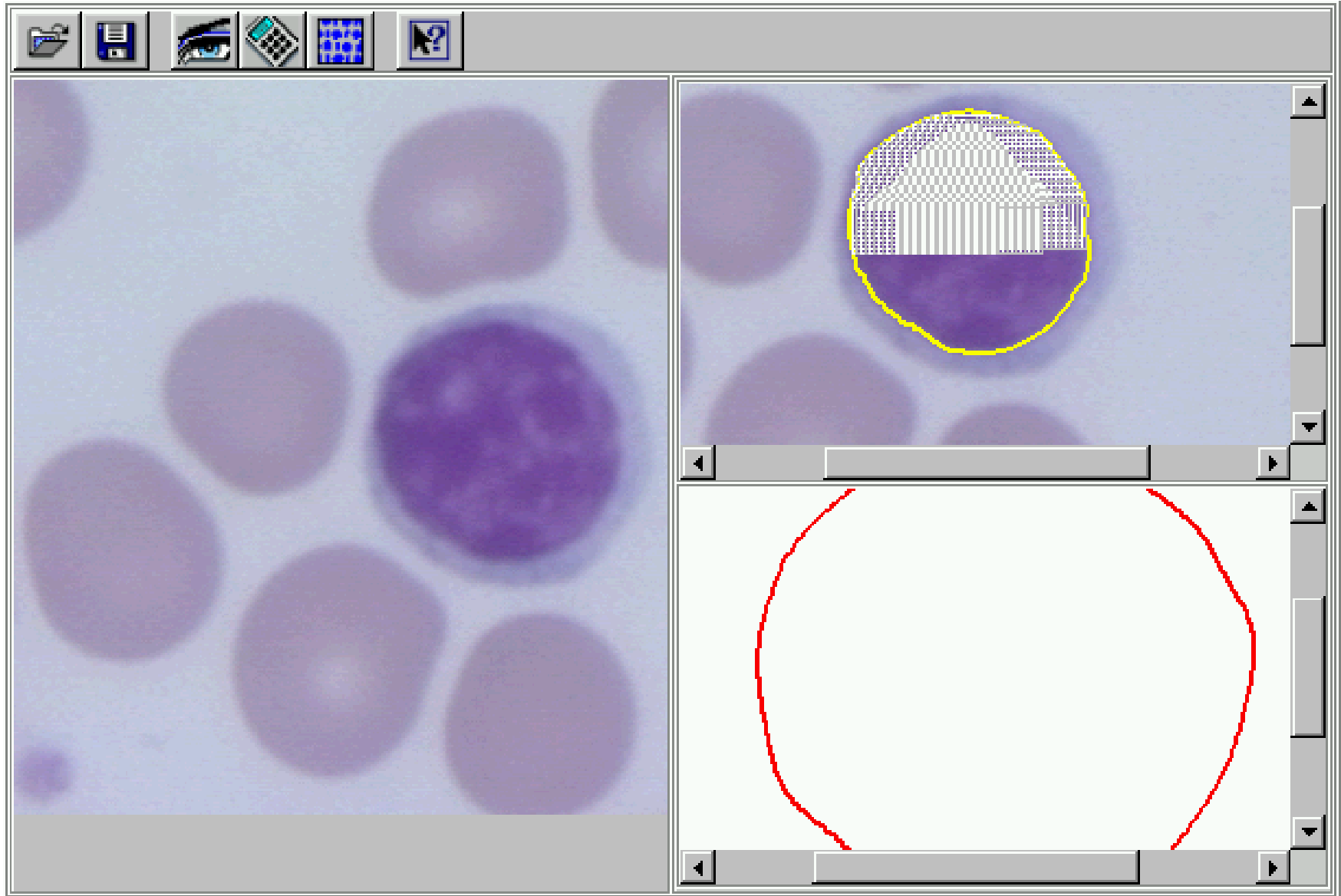


(4)



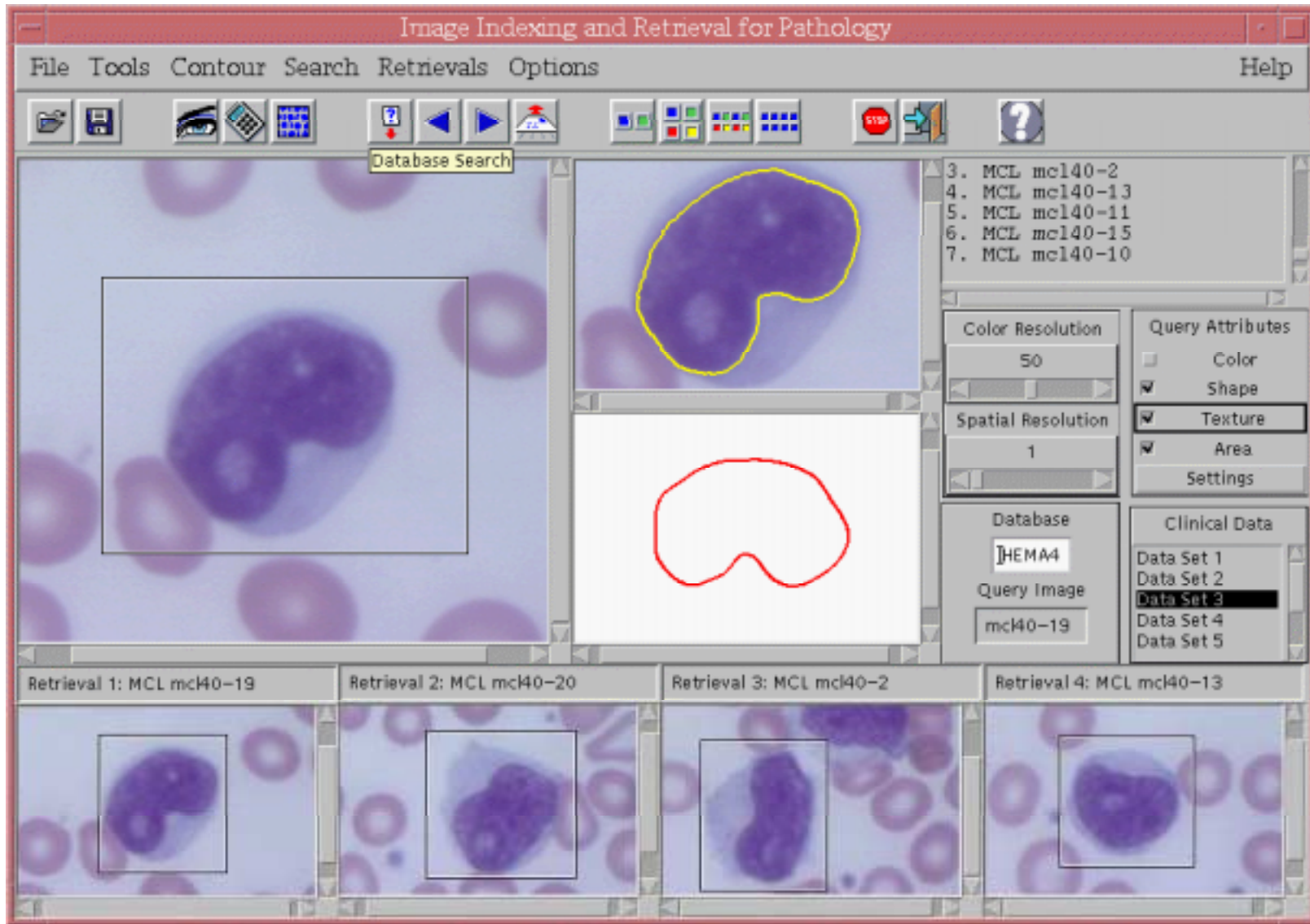
CUC 2001 (5)

# Virtual Microscope Lab





# Medical Diagnosis Support Application



# Classroom Evaluation

---

- Evaluation results encouraging
  - Observations show that interface design choices are mostly correct
  - Student surveys show that students found the labs useful
  - Student performance studies currently done by Rutgers Dept. Education
- Reduced need for teacher intervention
- Students liked interactive engagement
  - Increased student interest and control
  - Passive viewing and listening pre-recorded lectures not popular
- Exposed design problems & missing features
  - Need for user-centered design
  - Need for expert-system-based <sup>CUC 2001</sup> automatic help and guidance

# Conclusions

---

- **Software architecture for rapid development of virtual laboratories**
  - Used to develop five example virtual biology labs
- **Easy programming and modification**
  - XML-based scripting language
- **Labs evaluated in classroom**
  - Evaluation results and student comments demonstrate the value of the labs
  - Currently a supplement to physical labs

# Continuing Work

---

- **Expert System Help**
  - JESS: Java expert system shell
  - Servlets for performance
- **Talking Faces**
  - Increase user engagement
- **Distributed Real-time Collaboration**
- **Behavior Programming**
  - Use procedural scripting language?
- **Classroom Evaluation**
  - Joint work with Rutgers Department of Education