

# SketchUML

## A Tablet PC-based e-Learning Tool for UML Syntax using a Minimalistic Interface

Bastian Tenbergen\*, Colleen Grieshaber, Lisa Lazzaro, Rick Buck

{ tenberge | lazzaro | buck } @ oswego.edu  
colleen\_grieshaber @ yahoo.com

\* presenting and corresponding author

Human-Computer Interaction Laboratory  
Interactive Learning Technology Laboratory  
Oswego State University, NY, USA

Project Websites:

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Outline

- Traditional Interfaces
- Minimalistic Interfaces
- Naturalistic Interfaces
  
- Tablet PC-based Education
  
- SketchUML
  
- Usability Study
- Results
- Discussion

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Traditional Interfaces

- Computers play an important role in education
- Computer-aided design helps in understanding development processes
  - Also eases instruction of certain (mostly computer science and graphic design related) topics
- Mostly traditional screen-and-mouse interfaces
  - Packed screens

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>



Pack... SVN ...

← → ↶ ↷ ↺ ↻

- DSS
- FamilyTree
- PySelectiveDownloader
- WebStats
  - edu.oswego.hci.hci530
  - edu.oswego.hci.hci530
    - HttpServer.java
    - Request.java
    - Response.java
      - Response
        - BUFFER
        - http204
        - http404
        - output
        - request
        - Respons
        - sendStat
        - sendStat
        - setRequ

```

package edu.oswego.hci.hci530.betrecommender;

import java.io.File;

public class Engine { //implements MLPs {

    private Layer input;
    private Layer hidden;
    private Layer output;
    private Vector features;

    public Engine(String file) throws FileNotFoundException {
        features = new Vector();
        int maxValue = 0;
        //load & parse file
        //parse each row in file
        Scanner sr = new Scanner(new File(file)).useDelimiter("\r\n");
        while (sr.hasNext()) {
            //now parse each column in row
            String row = sr.next();
            StringTokenizer sc = new StringTokenizer(row);
            int[] featureVector = new int[sc.countTokens()];

```

Outline

- Undo (Ctrl+Z)
- Revert File
- Save
- Open Declaration (F3)
- Open Type Hierarchy (F4)
- Open Call Hierarchy (Ctrl+Alt+H)
- Quick Outline (Ctrl+O)
- Quick Type Hierarchy (Ctrl+T)
- Show In (Alt+Shift+W)
- Cut (Ctrl+X)
- Copy (Ctrl+C)
- Paste (Ctrl+V)
- Source (Alt+Shift+S)
- Refactor (Alt+Shift+T)
- Local History
- References
- Declarations
- Add to Snippets...
- Find Bugs
- Edit in JSEclipse
- Run As
- Debug As
- Profile As
- Validate
- Team
- Compare With
- Replace With
- Preferences...

NUM LOCK: OFF

Writable Smart Insert 1:1



# Traditional Interfaces

- **Problem:**

Many applications to aid in development/design processes like

- IDEs (Eclipse, Netbeans, #Develop, ...)
- Design Tools (Illustrator, Flash, Photoshop, ...)
- CGI Applications (AutoCAD, Inventor, ...)

have no support for learners!

- No “Beginner Mode”
- No Wizards
- No feedback when a design concept is wrong
- No context-sensitive assistance  
(with few exceptions)

- **High Learning Curve!**

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Minimalistic Interfaces

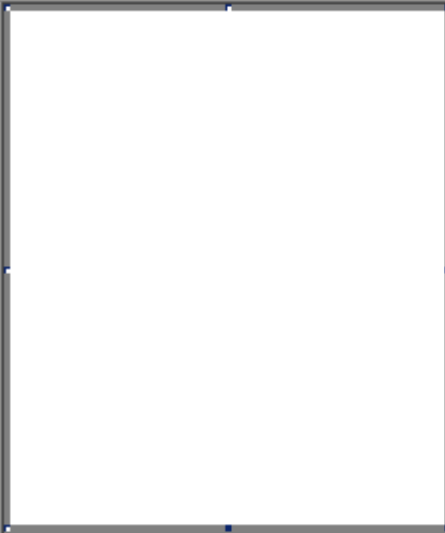
- Minimalistic Interfaces can do a better job
  - No confusion on “what means what”
  - Clear outline of elements
  - Easy-to-understand symbols
- Allow for
  - Learning by doing
  - Undoing the previous step
  - Incremental learning process
- But, potentially still some problems...

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Minimalistic Interfaces

- Potential Problems:
  - Learning curve still high (although not as high as with fully-featured applications)
  - Wrong concepts might be learned
  - No context-sensitive help
- Most significant:
  - Lack of features in most applications with minimalistic interfaces!





# Naturalistic Interfaces

- Non-Naturalistic Interfaces
  - Don't introduce mapping between action and reaction
    - on/off switch for a pump, furnace, etc
    - Eject button on your VCR
- Naturalistic Interfaces
  - Allow for direct or indirect mapping of action and re-action
- Inferred Naturalism vs Direct Naturalism

# Naturalistic Interfaces

- Inferred Naturalism

Indirect mapping of action and reaction

- Arrow Keys
- Mouse
- Joystick

Map an action on the interface to some corresponding re-action on the display:

- Right arrow means right
- Mouse up means up
- Joystick down-right means down-right

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Naturalistic Interfaces

- Direct Naturalism

Direct mapping of action and reaction

Mostly used in Direct Manipulation Paradigms

- Virtual Reality Glove
- Nintendo Wii Controller
- Touch Screens
- PDA Stylus
- Tablet PC Pen

Map an action on the interface to identical action on the display

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Tablet PC-based Education

- Pen/Stylus is a direct naturalistic interface
  - Only when used to draw ink in ink-enabled applications
- Allows for a multitude in educational advances for:
  - Faculty
    - Can interact with screen content more directly to show complex interaction of concepts
      - For instance: TabCon (Concurrency Teaching Utility for Tablet PCs)

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Tablet PC-based Education

- Allows for a multitude in educational advances for:
  - Students
    - Can **explore** the program without having to figure out the controls first
    - **Collaborative learning** allows feedback from peers and instructors
    - **Mobility** allows to learn anywhere (as opposed to tablet-peripheries)

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# SketchUML

- Goal:
  - Combine the advantages of
    - Minimalistic Interfaces
    - Naturalistic Interfaces
    - Tablet PCs
  - to provide a good learning experience when learning complex design tasks.

# SketchUML

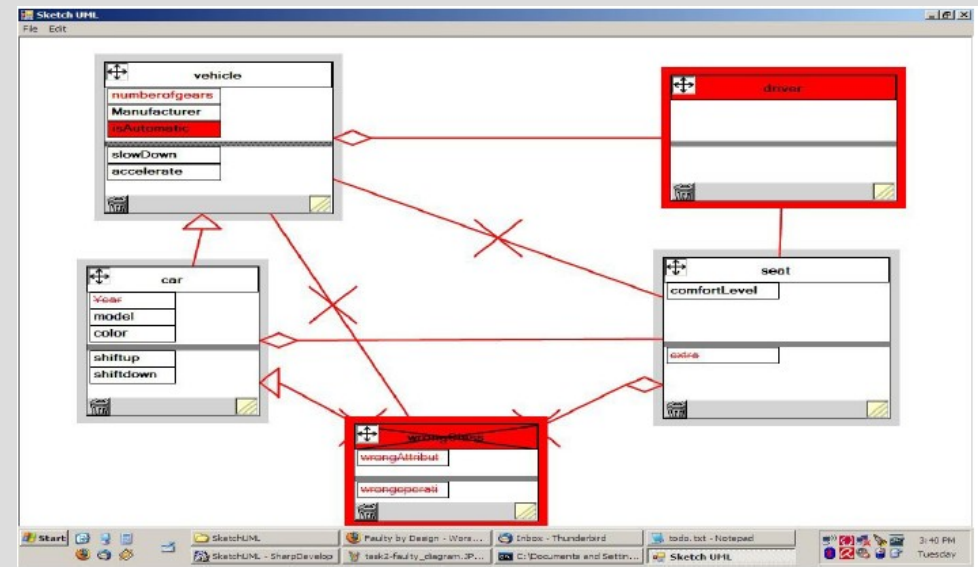
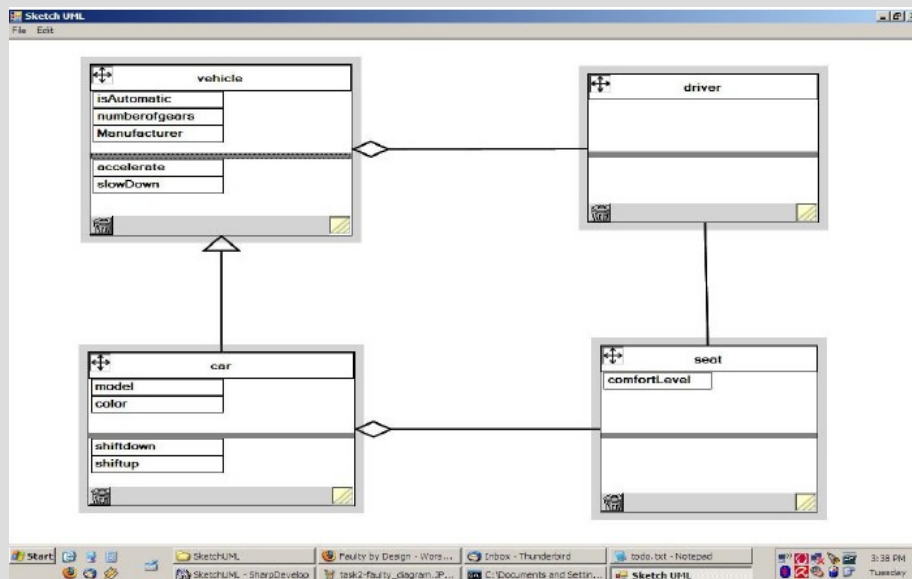
- Designed to facilitate learning the UML syntax
- Allows to draw ink on blank canvas
- Program converts it into valid UML
  - Only if gesture corresponds to UML component
- Context Sensitive help
- Allows for self-guided learning
  - Without complex algorithms to supervise learning
  - Can easily be ported to other design concepts

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# SketchUML

- A short demonstration...



- You can also check out our demo video:  
[http://www.youtube.com/watch?v=clF6\\_S-xJq](http://www.youtube.com/watch?v=clF6_S-xJq)

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>  
<http://sketchuml.tenbergen.org>



# Usability Study

- Purpose:
  - Investigate the performance and potential pitfalls of applications that employ minimalistic and naturalistic interfaces
  - Investigate the ability of naturalistic interfaces to mimic non-digital interfaces
  - Find out how computer-aided design principles can be exploited to facilitate learning when employed in these interfaces

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Usability Study

- Participants:
  - 11 students of Northeastern US college
  - Enrolled in software engineering class
  - Mediocre knowledge of UML
- Apparatus:
  - 6 HP Compaq TC4200 Tablet PCs
  - 1.7GHz CPU
  - 512mb Memory
  - SketchUML pre-release version

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Usability Study

- Procedure:
  - 1. pre-test demographic questionnaire
  - 2. introduction to SketchUML on projector
  - 3. separated participants into two groups
  - 4. handed out task description depending on participants group
    - Two different tasks
    - Tasks designed to incorporate every feature of SketchUML at least once
    - Tasks were text only – students had to invent diagram given the task description!
  - 5. Post-test questionnaire on perceived experience

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

# Usability Study

- Experiment was not timed
- Participants were free to start over as often as necessary
- When finished, diagrams were stored using SketchUML's built-in export function
- Performance was recorded by conducting a task performance analysis
  - Measure number of erroneous steps made in the final diagram against an ideal solution.

# Results

First condition – Group 1  
(participants asked to create a diagram)

- Overall of 9 mistakes in 6 participants
- 21 steps to complete the task (minimum)
- Only 1 mistake in symbol manipulation
- Majority of 8 mistakes when manipulating labels with handwriting!!

# Results

Second condition – Group 2  
(participants asked to critique diagram)

- 16 mistakes by 6 participants  
(3.2 mistakes/participant)
- 7 steps minimally to complete the task
- Majority of mistakes made during label and connector manipulation

# Results

## Post-Test Questionnaire

- Students reported high satisfaction with product interaction
- Frustration only occurred when handwriting was recognized incorrectly
- Context-sensitive gesture recognition was perceived as easy to understand and learn

# Discussion

- Gesture recognition did not cause problems!
  - Minimalistic naturalistic interfaces can rely on this mode of interaction
- Context-sensitive functionality increased learning experience
- Poor handwriting recognition performance caused frustration and inhibited learning
  - That's Microsoft's Problem... not ours ;-)



# Discussion

- Minimalistic Interfaces do not inhibit functionality and learning!
- Naturalistic Interfaces can increase learning experience
  - In complex design concepts
  - Help to focus on design task rather than the interface
  - Collaborative learning can enhance this effect
  - Unobtrusive!

# Acknowledgements

- This project is made possible by the generous support of
  - Dr. Gary Klatsky
  - Dr. Douglas Lea (both at SUNY Oswego)
  - Dr. Lin Qiu (formerly at SUNY Oswego)
  - The Research Foundation of SUNY
- And through the generous funding of the Hewlett-Packard Development Company, L.P., HP Technology for Teaching Grant Number U06TFH0037C.

# SketchUML Project Site

Be sure to check out:

<http://moxie.cs.oswego.edu/~tenberge/SketchUML>

<http://sketchuml.tenbergen.org>

Demo Video:

[http://www.youtube.com/watch?v=clF6\\_S-xJqs](http://www.youtube.com/watch?v=clF6_S-xJqs)