TUIML Visual Editor

Prof. Robert Jacob
Orit Shaer
Amy von Holten
What is a Tangible User Interface (TUI)?

- Use physical objects to represent digital data.
- Aim to break away from the traditional computer screen and bring reality based interaction into interfaces.
- EXAMPLE TUI: Urban Planning
Why Are TUIs Important?

- TUIs allow for not only human-computer interaction, but human-human interaction.
- TUIs are the future of interfaces and facilitate a new way of thinking.
The Problem & Our Project

• **Problem:**
  – Currently, TUI systems are difficult to build; each developer has their own method.

• **Our Project:**
  – Create a software solution for designing TUIs.

• This is made possible by a new language developed here at Tufts called TUIML (TUI Modeling Language).
  – TUIML is a visual language like UML.
  – TUIML fully describes a TUI and all its interactions
  – TUIML uses three diagrams to describe a TUI:
Brief Explanation of TUIML

• **TAC: Tokens and Constraints**
  – **Tokens:** Physical objects that represent digital data.
  – **Constraints:** Physical objects that provide context to tokens manipulation by constraining the ways in which tokens can be manipulated.

• **Dialog Diagram**
  – Squares represent all possible states the system can be in.
  – Triangles represent all possible interactions in that state.
  – Arrows represent transitions.
More TUIML

- **Interaction Diagram**
  - From Dialog diagram, this explains one of the triangles in a state.
  - Physical and Digital areas.

- Rectangles are transitions.
- Rounded rectangles are TACs.
- Arrows into a transition are preconditions needed for the transition.
- Arrows leaving the transition are post-conditions, resulting from the transition.
- Hexagons are continuous manipulations, also making use of bi-directional arrows.
- Ovals are digital data.
- The recycler symbol indicates the diagram returns to its initial state.
addWindTool

removeWindTool

moveWindTool

windSimOn-true

displayWind()

windSimOn-false
The Software and the Future

• Realized as a plug-in for Eclipse, an IDE.

• **Future Work:**
  – Freely sketch the diagrams
  – Prototyping; creating a virtual testing ground for the fully specified TUI.
  – Integrating TUIML’s built in support for XML.

• **Final Statement:**
  – There is much room for addition
  – Not meant to be a complete, commercial solution
  – Hoped that it will complement the newly created language (TUIML) and propel it forward as the standard language of TUIs.