Programming by Demonstration

Terminology

- **PBE**: Programming by example (AKA Example-based programming)
  - User provides one or more concrete examples of the behavior or effect of a more general program
  - System infers intent
- **PBD**: Programming by demonstration—D.C. Smith
  (AKA Demonstrational programming—B. Myers)
  - PBE in which the user demonstrates actions on example data (although PBD and PBE are often used interchangeably)
- **PITUI**: Programming in the User Interface—D. Halbert
  - PBD, emphasizing the use of existing UI commands

- **Inferencing**
  - Determining appropriate generalizations from examples
**Pygmalion** D.C. Smith, 1975

- Introduced icons to computer UIs
- First PBD system
- Designed to work as “blackboard”
- *Programmer* demonstrates algorithm with an example
- No inferencing
- Many relationships not visible

---

**SmallStar 84** D. Halbert

- Reimplementation of Xerox Star subset to support PBD of desktop operations
- **Approach**
  - *User* performs a specific set of tasks
  - *System* records tasks to create a program (at this point, a macro)
  - *System* determines *data description* of any object selected by user (picks one arbitrarily if ambiguous), reuses it throughout program (i.e., no inferencing)
  - *User* can view data description of any object with "properties"
  - *User* must edit program to
    - Modify data descriptions, including generalization
    - Create flow of control

(Xerox Star icons created by DC Smith)
SmallStar 84 D. Halbert

SmallStar desktop

Program to move named document “Treaty”

User selects “Treaty” in the program and presses “Properties” key to display or modify...

Document data description sheet for “Treaty”
SmallStar 84 D. Halbert

Editing data descriptions is used as a substitute for inferencing from multiple examples.

Adding set iteration loop by selecting line(s) and choosing “Repeat” from popup menu.

Creating programs to move all files by copying “Negotiations any” to “everything matching” descriptor.

Editing programs is used to create flow of control.
SmallStar 84  D. Halbert

Mailing form

- Order #123
- Elza F. Dolittle
- London, England
- Weight (pounds): 
- 
- Fourth-class mail

Program

Open Mailing Form.
Direct Mailing Form to First Class.
Type in Text at beginning of Mailing Form to First Class.
Close Mailing Form.
Move Mailing Form to Mailing Room Printer.

SmallStar 84  D. Halbert

Mailing form program with unconditional conversion to first class

Property sheet for conditional testing weight

- First operand: CONSTANT
- Description: Mailing Form [Weight] [Fall]
- Operator: 
- Second operand: CONSTANT
- Description: Mailing Form [Weight] [Fall]

Feiner, COMS W4170, Fall 2017
SmallStar 84  D. Halbert

Mailing form program with conversion to first class conditional on weight

Peridot  B. Myers 1987

- Programming by Example for Real-time Interface Design Obviating Typing
- Creates new GUI widgets
  - User-defined look and feel
- Designed for nonprogrammer users
- One of earliest uses of inferencing for PBD
- Inferencing for
  - Graphical constraints (user gives one example)
  - Iteration (user gives two examples)
  - …
**Procedure PropSheet (Items)**
- **ActiveValue** Selected-Props

Should show user's input: "Selected-Props" and its value "(Italic Underline)"

Note value of "Selected-Props"

User creates gray rectangle
Peridot B. Myers 1987

User creates black rectangle
Peridot proposes constraint

Peridot B. Myers 1987

Peridot establishes constraint:
Black rectangle is now the same size as gray rectangle and at the specified offset
Peridot B. Myers 1987

User creates white rectangle
Peridot proposes constraint

Peridot B. Myers 1987

User creates string “Bold”
Peridot proposes constraint
**Peridot** B. Myers 1987

User selects and copies rectangles and text

---

**Peridot** B. Myers 1987

User changes “Bold” to “Italic”

Peridot infers an iteration over the elements in “Items” and asks user to confirm
Peridot B. Myers 1987

Peridot performs the iteration

User places check mark icon in box

Peridot proposes constraint
User places simulated mouse over check mark with middle button down, indicating a mouse dependency
- Peridot infers dependency on the single active value → check mark for all items in the single active value list (italic, underline)
- User selects mouse action (e.g., toggle value)

Widget created with Peridot can now be used with current “Items” list or with different “Items” list
- Different item strings
- Different number of items
**Chimera**  D. Kurlander 1988–92

- Example-based graphical editing
  - Editable graphical histories
    - Comic-strip/storyboard metaphor
    - Inspect to review
    - Revert to undo
    - Edit to change/redo
    - Select, parameterize, generalize to create “macros by example”

https://www.youtube.com/watch?v=JbrJQW25ekI

---

**Chimera**  D. Kurlander 1988–92

- Graphical history can be used to select past actions

https://www.youtube.com/watch?v=JbrJQW25ekI