Identifying Objects

A Case Study and
Class Exercise

Ward Cunningham
Computer Research Laboratory
Tektronix, Inc
• Outline

I. Division of Responsibility
   Waveform Example

II. A Method of Design
    Drawing Editor Example

III. Class Exercise
    Bank Machine Problem

IV. A Design Tool
    HyperCard Stack
• Object-Oriented Programming

1. Identify Objects
2. Design Protocol
3. Factor Hierarchy
4. Implement Methods

increasing
difficulty
• Identifying Objects

  • Model Computation as an activity of collaborating agents (objects).

  • Define the classes to which objects will belong.

  • Distribute responsibility (to meet requirements) over classes.
• Division of Responsibility Example

Traces

Ref 3

Waveforms

Configurations

Freq 1000
Gain 300
Offset 20
Polarity +

Vectors

97
99
105
120
122
130
133
139
140
145
• Trace Responsibilities

  • Render waveforms as visible image. Record display position, scale and emphasis.

• Waveform Responsibilities

  • Interpret samples as a representation of a real signal. Manage signal processing.

• Vector Responsibilities

  • Store sample data. Perform numerical computations. Access signal processor.

• Configuration Responsibilities

  • Configure acquisition circuitry. Record acquisition circumstances.
• Questions

• What objects need be duplicated when a signal is displayed both normal and magnified?

• What objects must (should) change when a new trigger occurs?

• Exactly when does a configuration update after a user-requested change in acquisition parameters?
- Grade School Example
- Object-oriented description

<table>
<thead>
<tr>
<th>Responsibility</th>
<th>Collaborators</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teacher</td>
<td>Teaches Lessons</td>
</tr>
<tr>
<td></td>
<td>Evaluates Students</td>
</tr>
<tr>
<td></td>
<td>Secretary</td>
</tr>
<tr>
<td></td>
<td>Student</td>
</tr>
<tr>
<td></td>
<td>Principal</td>
</tr>
<tr>
<td>Student</td>
<td>Learns Lessons</td>
</tr>
<tr>
<td></td>
<td>Teacher</td>
</tr>
<tr>
<td></td>
<td>Principal</td>
</tr>
<tr>
<td>Principal</td>
<td>Administers Funds</td>
</tr>
<tr>
<td></td>
<td>Disciplines Students</td>
</tr>
<tr>
<td></td>
<td>Hires Staff</td>
</tr>
<tr>
<td></td>
<td>Teacher</td>
</tr>
<tr>
<td></td>
<td>Secretary</td>
</tr>
<tr>
<td></td>
<td>Student</td>
</tr>
<tr>
<td>Nurse</td>
<td>Gives First Aid</td>
</tr>
<tr>
<td></td>
<td>Gives Vaccinations</td>
</tr>
<tr>
<td></td>
<td>Students</td>
</tr>
<tr>
<td></td>
<td>Teachers</td>
</tr>
<tr>
<td>Secretary</td>
<td>Answers Phone</td>
</tr>
<tr>
<td></td>
<td>Prints Handouts</td>
</tr>
<tr>
<td></td>
<td>Teacher</td>
</tr>
<tr>
<td></td>
<td>Principal</td>
</tr>
<tr>
<td>Janitor</td>
<td>Cleans Building</td>
</tr>
<tr>
<td></td>
<td>Fixes Equipment</td>
</tr>
<tr>
<td></td>
<td>Teacher</td>
</tr>
<tr>
<td></td>
<td>Secretary</td>
</tr>
<tr>
<td>Cook</td>
<td>Prepares Meals</td>
</tr>
<tr>
<td></td>
<td>Janitor</td>
</tr>
</tbody>
</table>
• Grade School Example
• Process oriented description

Money

Direction

Administer

Teachers

Books

Educate

Students

Graduates
• A Method of Design

  • Make decisions based on wisdom and experience.

  • Record decisions in a structured design document.

  • Test design for adequacy and completeness.

  • Maintain and refer to the design throughout implementation.
• Design Representation

  • Enumerates all (new) classes.

  • Defines responsibilities assumed by members of each class.

  • Describes collaborations through which responsibilities are discharged.
• Introducing CRC Cards

**Drawing**

- Holds & displays Figures
- Accumulates region of change.

**Class**

- Drawing

**Responsibility**

- DrawingView
- DrawingController
- Figure

**Collaborators**
Step 1: Start With Knowns

A Drawing is composed of Figures
Figures come in several kinds
Step 2: Hypothesize Support

- A line may connect to other figures
- A "smart" point does the work
Step 3: Test with Scenarios

- Figures notify the Drawing and dependent Locators when moved.
- Change propagates through Locators
Step 4: Try Various Groupings

- A Handle is like a Tool ...
- Locators are quite unique ...
Step 5: Redistribute Responsibilities

- Selections are kept in the View
- Selections won't be saved with a Drawing
Step 6: Rewrite for Clarity

• It is important that Figures are ordered
• Design Exercise (Requirements)
• Design Exercise (Methodology)

1. Start with Knowns
2. Hypothesize Support
3. Test with Scenarios
4. Try Various Groupings
5. Redistribute Responsibility
6. Rewrite for Clarity
### Design Exercise (Schedule)

- Find partner and design system: **1 Hr.**
- Find new partner and review design: **15 Min.**
• CRC Stack

A HyperCard stack for browsing and editing an object-oriented design

A Machine-readable design aids distribution, maintenance and validation.
• Browsing Collaborators

• Click to browse a collaborator, press and hold to create and link a new collaborator card

• Use HyperCard browsing (Find, Recent, etc)
• Category Cards

User Interface Services

Distinct user interface paradigms are supported by objects providing such a service to other scope components.

- BezeiMenu
- Help

• Categories organize a completed design as an aid to learning and understanding