Understanding Code

David A. Mellis

Advisors
Neil Churcher
Yaniv Steiner

Overview

Redesign of a debugger: software tool for observing, annotating, and analyzing the behavior of software.
Agenda

context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact
next steps
Agenda

context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact
next steps
Context
What we consider a simple program today is much more complex to build. It sucks up programmer’s mental processing power. They need better tools.
What it’s not: a tool for non-programmers

Pablo, a visual programming environment (MIT Media Lab).
Agenda

context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact
next steps
Frustrations today

I can’t keep it all in my head.
I can’t tell what connects to what.
I’m getting lost in the details.
I can’t see what’s happening.
I don’t know where in the code to look.
I don’t know what will happen if I change this.

Gathered from extended interviews with 2 professional programmers (mid-20’s) and informal conversation with others.
Agenda
context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact
next steps
Today: freezing time in snapshots
Need: continuous timeline
Today: abstract instructions, separated data

\[ a + b + c + d = ? \]

\[ a = 5 \quad b = 10 \quad c = -2 \quad d = 3 \]
Need: concrete operation

\[
\begin{array}{c}
5 + 10 + (-2) + 3 = ?
\end{array}
\]
Today: only see static structure

**GRAPHICS**
- Images.java
- Textures.java
- Lighting.java
- Camera.java

**PHYSICS**
- Gravity.java
- Collision.java

**AI / GAMEPLAY**
Need: dynamic structure

```plaintext
main()
  ➔ startGame()
    ➔ createPlayer()
    ➔ loadLevel()
    ➔ turnOnGraphics()
  ➔ playLevel()
    ➔ handlePlayerInput()
    ➔ moveMonsters()
    ➔ drawScreen()
    ➔ checkForGameEnd()
    ➔ insertBonusItems()
```
Need: connections

main()
  → Start Game()
  → create Player()
  → play Level()
  →
  →
  →

GRAPHICS

PHYSICS

AI
Agenda

context
problems today
visual metaphors
**design goals**
scenario from last review
interface specification
prototype for testing
impact
next steps
Knowledge in the world vs. in your head

Tangible variables

\[ \text{a} \quad \text{b} \quad \text{c} \]

Tangible time
Tangible connections
Supporting tasks

Debugging: focused goal, observing details
Maintenance: overall understanding, which parts do what
Complexity: details and flows across components
Agenda

context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact
next steps
output(\text{answer})
\begin{array}{l}
\text{draw}(\text{answer});
\end{array}
output (answer)

3
draw

% FB^/**
value should be

17
output (answer)

3
draw

% / FB^/ ***

show me
where this
came from
calculate()

3

answer = "% / FIB^/* *;"

3

someone should probably fix this.
calculate ()

\[
\text{answer} = g(1)
\]
Agenda

context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact
next steps
class Graphics {
    void start()
    {
        if (first_game)
            loadEverything();
        else
            resetEverything();

        for (int i = 0; i < shapes.length; i = i + 1)
        {
            draw(shapes[i]);
        }
    }

    void draw(Shape shape)
    {
        print(shape.type); // Rectangle
        print("width: "); // 100
        print("height: "); // 30
        print(shape.vertices);
        4: [<Vertex, Vertex, Vertex, Vertex>]
    }

    void drawBorder(Shape shape)
    {
    }

    void drawBackground(Shape shape)
    {
    }

    void drawForeground(Shape shape)
    {
    }

    void loadEverything()
    {
    }
}
Program files and dependencies

- Graphics.java
- Physics.java
- Gameplay.java
- Compiler.java
- URL.java
- Compiler.java
- System.java
- Networking.java

“Children”

“Parents”
Timeline of program execution ("trace")

- main
  - startup
    - Graphics.start
      - Graphics.load... 22
      - Graphics.draw 3
      - Graphics.draw 3
      - Graphics.draw 3
      - Graphics.draw 3
      - Graphics.draw 3
      - Graphics.draw 3
      - Physics.start 5
      - Gameplay.start 123
    - run 3456
    - cleanup 291
  - Clock time
  - Hierarchical time
class Graphics {
    void start()
    {
        if (first_game)
            loadEverything();
        else
            resetEverything();

        for (int i = 0; i < shapes.length; i = i + 1)
            draw(shapes[i]);
    }

    void draw(Shape shape) {
        print(shape.type)
        print("width: ")
        print("height: ")
        print(shape.vertices);
    }

    void drawBorder(Shape shape)
    {
    }

    void drawBackground(Shape shape)
    {
    }

    void drawForeground(Shape shape)
    {
    }

    void loadEverything()
    {
    }
}
Live function: code operating on data

```java
void draw(Shape shape <Rect:100x30px, (30, 25)>)
{
    print(shape.type Rectangle);
    print("width: " + shape.width 100);
    print("height: " + shape.height 30);
    print(shape.vertices);
    4:[<Vertex, Vertex, Vertex, Vertex>]
}
```
Live function: unreached code

```cpp
if (first_game
true
)
    loadEverything();
else
    resetEverything();
```
Live function: loops

```python
for (int i = 0; i < 10; i = i + 1):
    print(i);
```
for (int i = 0; i < shapes.length; i++)
    draw(shapes[i]);

Returns to calling function

void draw(Shape shape)
{
    drawBorder(shape);
    drawBackground(shape);
    drawForeground(shape);
}
Flagging incorrect values

```python
print("width: " + shape.width 100); 
print("height: " + shape.height); 
print(shape.vertices);
```

Previous change
Find in trace
Flag

```python
print("width: " + shape.width 100 correct value );
```

```python
print("width: " + shape.width 100 25 );
```
class Graphics {
    void start() {
        if (first_game)
            loadEverything();
        else
            resetEverything();
    
        for (int i = 0; i < shapes.length; i = i + 1)
            draw(shapes[i]);
    }

    void draw(Shape shape) {
        print(shape.type "Rectangle");
        print("width: " + shape.width 100 correct value");
        print("height: " + shape.height 30");
        print(shape.vertices);
        4: <Vertex, Vertex, Vertex, Vertex>
    }

    void drawBorder(Shape shape) {
    }

    void drawBackground(Shape shape) {
    }

    void drawForeground(Shape shape) {
    }

    void loadEverything() {
    }
}
Flags in the program timeline

- main
  - startup
    - Graphics.start
      - Graphics.load... 22
      - Graphics.draw 3
    - Graphics.draw 3
    - Graphics.draw 3
    - Graphics.draw 3
    - Graphics.draw 3
    - Graphics.draw 3
    - Physics.start 5
    - Gameplay.start 123
  - run 3456
  - cleanup 291
Search: file, function, variable
Variable traces

```
main
  \ startup
  100
   \ Graphics.start
      24
        \ Graphics.draw
        \ Graphics.draw
        \ Graphics.draw
        \ Graphics.draw
        \ Graphics.draw
        \ Graphics.draw
        \ Physics.start
        \ Gameplay.start
        \ run
        \ cleanup
  32
    \ Graphics.load...
    22
```
Programmer feedback

Early draft of wireframes. Reviewed by 7 professional programmers. Approval of general approach and structure. Requests for specific features. Many of which have been incorporated.
Agenda

context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact
next steps
Agenda

context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact
next steps
Impact

More productive programmers.
Better and more complex software.
Agenda

context
problems today
visual metaphors
design goals
scenario from last review
interface specification
prototype for testing
impact

next steps
Next steps

User testing and validation.

Complete interface specification.

Exhibition preparation.
Thank you.