Software Visualization or How to See and Explore the Intangible

Houari Sahraoui
, Université de Montréal
Acknowledgement

- Omar Benomar
- Simon Bouvier
- Karim Dhambri
- Guillaume Langelier
- Pierre Poulin
- Ahmed Sfayhi
Maintenance Tasks

- Maintenance problems

Diagram:
- Maintenance
  - Corrective
  - Adaptive
  - Perfective
  - Preventive
    - Reverse engineering
    - Feature location
    - Evolution analysis
    - Anomaly detection/correction
    - Quality assessment
    - Migration
Maintenance Tasks

• Many maintenance tasks difficult to automate
  – Multi-criteria decision making
  – Difficulty to capture/encode contextual information
  – Complexity/scalability
Interactive Visualization

• Semi-automatic approach
• Maintenance task = Set of reasoning and calculation modules
  – Set of automated modules AMs (explicit knowledge)
  – Human analyst module HM
  – Interactive visualization = Interface between AMs and HM
Visualization Tools for Maintenance

- Dozens of tools proposed each year in VLHCC, SOFTVIS, VISSOFT, SE journals, conferences and workshops, etc.
Visualization Tools for Maintenance

• Not used outside the community that developed them

• What’s wrong?
  – Not tailored for specific tasks
  – Effort and efficiency
    • Intrinsic complexity
    • Suitability
Modeling Maintenance as Interactive Visualization

• Maintenance world
  – Maintenance task
    • Exploring data extracted from software artifacts
    • Modifying software artifacts
  – Data
    • Entities (at different levels) with Properties, having Relationships and Structures
    • Viewpoints
    • Time
  – Operations
    • Aggregation, clustering, identification, etc.
Modeling Maintenance as Interactive Visualization

- Example: detecting anomalies
  - Example of anomaly definition
    - A Blob is a controller class, abnormally large, with almost no parents and no children. It mainly uses data classes, i.e. very small classes with almost no parents and no children.
  - Model
    - Entities: classes, methods
    - Properties: coupling, cohesion, complexity
    - Relationships: invocations, inheritance
    - Structure: architecture
    - Viewpoints: code, metrics
    - Time: multiple version
    - Operations: evaluate conditions, etc.
Modeling Maintenance as Interactive Visualization

- Visualization world

Views
Movements
Interactions
Modeling Maintenance as Interactive Visualization

• Visualization world
  – Interactive visualization
    • Processing large sets of multidimensional data
    • Mainly a perception problem
  – Human brain hard-wired to perceive things in a certain way
  – Understanding human perception reduces complexity and increases efficiency
Defining Views

- Software is intangible
- Representations of
  - Entities by shapes
  - Properties by graphical attributes of shapes
  - Relationships by connecting shapes
  - Structures by spatially organizing shapes
Defining Views

• Representing entities (principles)
  – Gestalt Rules of Perception
Defining Views

• Representing entities (principles)
  – Simplicity
Defining Views

• Representing entities (principles)
  – Continuity
Defining Views

• Representing entities (principles)
  – Continuity
Defining Views

• Representing entities
  – Example of VERSO

Interface  Class/method
Defining Views

- Representing properties (principles)
  - Interaction between visual properties

Size vs depth
Defining Views

- Representing properties (principles)
  - Interaction between visual properties

Size vs depth
Defining Views

- Representing properties (principles)
  - Interaction between visual properties

Size vs orientation
Defining Views

- Representing properties (principles)
  - Interaction between visual properties

Size vs orientation
Defining Views

• Representing properties (principles)
  – Interaction between visual properties

Color & contrast
Defining Views

• Representing properties (principles)
  – Interaction between visual properties

Color & contrast
Defining Views

• Representing properties
  – Example of VERSO
Defining Views

- Representing relationships (principles)
  - Explicit representation vs size
Defining Views

- Representing relationships (principles)
  - Explicit representation vs size
Defining Views

• Representing relationships (principles)
  – Explicit representation vs size
Defining Views

• Representing relationships (principles)
  – On-demand representation
Defining Views

• Representing relationships (principles)
  – Flow maps

Minard, C. J. "Carte figurative et approximative des quantités de vin français exportés par mer en 1864".
Defining Views

- Representing relationships
  - Example of VERSO (filters)
Defining Views

• Representing relationships
  – Example of VERSO (Edge bundles)
Defining Views

• Representing structure (principles)
  – Whole vs parts

The unified whole is different from the sum of the parts.
Defining Views

- Representing structure (principles)
  - Law of proximity
Defining Views

- Representing structure
  - Example of VERSO
Defining Views

- Representing structure
  - Example of VERSO
Defining Views

- Representing structure
  - Example of VERSO
Movements between Views

• Within the same level (principles)
  – Change detection mechanisms
    • Change vs difference
  – Multiple-viewpoints management
  – Visual coherence
    • Spatial coherence
    • Temporal coherence
Movements between Views

• Spatial coherence
  – Difference detection
Movements between Views

• Spatial coherence
  – Change detection
Movements between Views

- Spatial coherence
  - Change detection
Movements between Views

• Spatial coherence
Movements between Views

• Spatial coherence
Movements between Views

- Temporal coherence
  - Change Blindness

Movements between Views

- Within the same level
  - Example of VERSO
Movements between Views

• Within the same level
  – Example of VERSO
Movements between Views

• Within the same level
  – Example of VERSO for the evolution
    • Fixed positions
    • Relative positions
Movements between Views

- Within the same level
- Example of VERSO for the evolution
- Fixed positions
- Relative positions
Movements between Views

- Within the same level
- Example of VERSO for the evolution
- Fixed positions
- Relative positions
Movements between Views

- Between levels (principles)
  - Keeping track of the context
  - Semantic zoom
Movements between Views
Movements between Views

• Between levels
  – Example of VERSO
Keeping track of the context
Keeping track of the context
Keeping track of the context
Interactions

• From analysis tasks to interaction scenarios
Example

- Blob Detection
  
  - Task description

Goal($\text{Blob\_detection}$, $\text{BlobSet}$, $\text{System}$)
{
  achieve($\text{Controller\_class\_detect}$, $\text{CDD}$, $\text{System}$)
  achieve($\text{Data\_class\_verif}$, $\text{BlobSet}$, $\text{CDD}$)
}

Goal($\text{Controller\_class\_detect}$, $\text{Cand}$, $\text{Scope}$)
{
  Filter($\text{Scope}$, $\text{Cand}$, $\text{ishigh WMC}$ and $\text{ininterval LOW MEDIUM LCOM5}$ and $\text{islow DIT}$)
}

Goal($\text{Data\_class\_verif}$, $\text{Found}$, $\text{Cand}$)
{
  $\text{for\_each(}$c, $\text{Cand)\{}$
  Filter($\text{System}$, $\text{Rel}$, $\text{iscalled(c)}$)
  Filter($\text{Rel}$, $\text{RelData}$, $\text{islow WMC}$ and $\text{islow DIT}$)
  Compute$_{\text{derived\_value}}$(RelData, count, Num)
  if ($\text{ishigh\ Num}$) {
    operation (+, $\text{Found}$, $\text{Found}$, c)
  }
}
• Blob Detection
  – Interaction scenario

Mapping

Graphic representation
  3-D box >> Class
Graphics attributes
  twist >> LCOM5
  height >> DIT
  color >> WMC
• Blob Detection
  – Interaction scenario

Scenario(Blob_detection)
{
  Run_scenario(Controller_class_detect)
  Run_scenario(Data_class_verif)
}

Scenario(Controller_class_detect)
{
  Overview(Class)
  Block(Class){
    Check_if(Color:Red
    and Twist:0 to 45
    and Height: Medium To High)
    Select(Result)
    Tag(CC, Result)
  }
}

Scenario(Data_class_verif)
{
  for_each(c in CC){
    Overview(Class)
    Apply_automatic_filter(Class, iscalled(c))
    Tag(REL, Result)
    Overview(Class)
    Block(REL){
      Check_if(color:blue
      and height:low)
      Select(Result)
      Tag(ConData, Result)
    }
    Overview(ConData)
    Do_function(count, ConData, Num)
    Block{
      Check_if(Num,ishigh)
      Tag(Blob, c)
    }
  }
}
- Blob Detection
• Blob Detection
Increasing Tolerance to Complexity

• Problem
  – In the context of large-scale systems
    • Principles mentioned above reduce data exploration complexity
    • Complexity is still overwhelming
    • Much too difficult for a human analyst
  – How can we increase tolerance to complexity?
Increasing Tolerance to Complexity

• Principles
  – Flow state (Csikszentmihalyi)
    *Mental state of operation in which the person is fully immersed in what he or she is doing, characterized by a feeling of energized focus, full involvement, and success in the process of the activity*

• Characteristics
  – Clear goals, distorted sense of time, …

• Applications
  – Education, (video) gaming, sport, …
Increasing Tolerance to Complexity

- **Principles**
  - Neurological theory of aesthetic experience (Ramachandran)
  - 10 universal laws of art:
    1. Peak shift
    2. Grouping
    3. Contrast
    4. Isolation
    5. Perception problem solving
    6. Symmetry
    7. Abhorrence of coincidence/generic viewpoint
    8. Repetition, rhythm and orderliness
    9. Balance
    10. Metaphor
Increasing tolerance to complexity

• Peak shift
  – Exaggerated versions of learned objects easier to interpret by the brain
  – Examples
    • Caricatures
    • Women in art
      – Forms
      – Positions
Increasing tolerance to complexity

- **Peak shift**
  - Exaggerated versions of learned objects easier to interpret by the brain
  - Examples
    - Caricatures
    - Women in art
      - Forms
      - Positions
Increasing tolerance to complexity

• Grouping & Perception problem solving
  – Human visual system is trained to detect regularities in a world of noise
  – Discovery of regularities is rewarding (AHA sensation)
  – Example
Increasing tolerance to complexity

- Symmetry & Repetition, rhythm and orderliness
  - Symmetry is attractive
  - Repetition, rhythm and orderliness are soothing
  - Example
    - Islamic art
    - Western painting
Increasing tolerance to complexity

• Symmetry & Repetition, rhythm and orderliness
  – Symmetry is attractive
  – Repetition, rhythm and orderliness are soothing
  – Example
    • Islamic art
    • Western painting
Increasing tolerance to complexity

• **Metaphor**
  - Generates emotional response even before we understand it
  - Examples
    • Indian art
    • Western painting
Increasing tolerance to complexity

• Metaphor
  – Generates emotional response even before we understand it
  – Examples
    • Indian art
    • Western painting
Increasing Tolerance to Complexity

- Implementation example
  - Differences between entities can be visually amplified
  - City metaphor is used
  - Data exploration tasks are modeled as perceptual problem solving
  - Entities are positioned following a particular order
  - Entity groupings are meaningful
  - Each graphical configuration has a single meaning
Conclusion

Interactive Visualization

- Semi-automatic approach
- Maintenance task = Set of reasoning and calculation modules
  - Set of automated modules AMs (explicit knowledge)
  - Human analyst module HM
  - Visualization = Interface between AMs and HM

Modeling Maintenance as Interactive Visualization

- Visualization world

Defining Views

- Representing structure
  - Example of VERSO

Increasing tolerance to complexity

- Grouping & Perception problem solving
  - Human visual system is trained to detect regularities in a world of noise
  - Discovery of regularities is rewarding (AHA sensation)
  - Example