Beyond Documents:
Visualizing Informal Communication

Kurt Schneider, Eric Knauss, Kai Stapel
<firstname.lastname>@inf.uni-hannover.de

Software Engineering Group
Overview

- Motivation: beyond documents

- FLOW modeling concepts
  - *Fluid information*
  - *Experience*
  - *Visualization*

- Example: Security requirements

- Comparison with other notations
The problem

• Practitioners frustrated: RE is never perfect

• „All requirements are specified …“?
  – Conversations
  – Phone calls
  – E-mails
  – Meetings
  – Wikis
  – Blogs
  – Tradition

Often ignored as „irrelevant“

Why?
Information forgotten
No distribution
No reconstruction

Why popular?
Fast
Flexible
Less effort, enjoyable
Fluid and Solid information

• **Solid** representation: documents and stores
  
  (1) that can be retrieved by others
  (2) without the author or source
  (3) even after some time
  (4) in a form that supports dissemination.

• All other representations are called **fluid**.
## Syntax of FLOW notation

### Metaphor

<table>
<thead>
<tr>
<th>Information state</th>
<th>Store</th>
<th>Information flow</th>
<th>Experience flow</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>solid</td>
<td><img src="doc.png" alt="Solid Store" /></td>
<td><img src="doc.png" alt="Solid Information Flow" /></td>
<td><img src="doc.png" alt="Solid Experience Flow" /></td>
<td><img src="doc.png" alt="Solid Activity" /></td>
</tr>
<tr>
<td>fluid</td>
<td><img src="doc.png" alt="Fluid Person" /></td>
<td><img src="doc.png" alt="Fluid Information Flow" /></td>
<td><img src="doc.png" alt="Fluid Experience Flow" /></td>
<td><img src="doc.png" alt="Fluid Activity" /></td>
</tr>
</tbody>
</table>

- Solid: Represents a stable, rigid environment where information is stored and flows are controlled explicitly.
- Fluid: Represents a dynamic, flexible environment where information flows naturally, often resembling human interactions.

### Typical pros & cons

- **Pros**:
  - Flexibility and adaptability.
  - Enhanced collaboration and communication.

- **Cons**:
  - Potential for information overload.
  - Difficulty in maintaining control.

### Transformations

- There are many „fluid cultures“ (ok)
- Explicit control of flows
Visualization goals

- Visualize solid and fluid
- Intuitive, use without training
  - Small set of simple symbols
  - on white-boards
  - in tools like Power-Point
  - In custom-built editors

- Link flow models to processes
  
  Documents, Activities

- Reuse well-known notations
  
  Comparison →
Modeling purposes and applications

- Awareness and Overview
- Improving Requirements Processes and Practices
- Defining and Tailoring Communication
- Tools & Techniques

Information flow model of desired situation

Customer

Improved First Interview
using
Fast Feedback Technique

feedback

Analyst

Use cases

UI sketches

animation

record of use

Approx. 1.5-2.5 hours

Time

FastFeedback using a TabletPC

Saving weeks of calendar time

Compared to initial uniPro interviews (Fig. 2)

see: REFSQ’07

Kurt Schneider: How FLOW can be used to discuss SecReq
For better visibility: green experience

- Stakeholders
- Security Engineer
- Security Experience
- UMLsec Tool
- Secure System

SecReq
Technique for handling security requirements
Initial elicitation: three alternative flows

1. Stakeholders write
   - Stakeholders write
   - Sec Eng reads
   - Sec Eng integrates

2. Stakeholders discuss
   - Stakeholders discuss
   - Sec Eng moderates
   - Sec Eng listens
   - Sec Eng summarizes

3. Instructor explains security
   - Instructor explains security
   - Sec Eng elicits
   - one-by-one or all together
   - Guided by experience

Kurt Schneider: How FLOW can be used to discuss SecReq
Elaborating on experience exploitation

SecReq Core
Elicitation and Validation of Security Reqs.

Security Requirements Elicitation
Security Engineer
NLP
Common Criteria
Experience Engineer
Security Experience
Individual Experience

Stakeholders
Security Instructor
(Elicitation Tool)

Improved SecReqs
Construct System
UMLsec
Secure System

Kurt Schneider: How FLOW can be used to discuss SecReq
Comparing with DFDs

**Related**
- Data/information flow focus
- Persistent storage modeled
- Dependencies via data only
- Context diagrams

**Different**
- Stores do not transform
- Data dictionary rules vs. FLOW
- No intuitive symbols
- No concept of fluid

Diagram:
- Security Instructor to Stakeholder with Instruction
- Stakeholder to Elicitation Experience with Requirements
- Elicitation Experience to Security Engineer with Requirements
- Security Instructor to 1a. Instruct on Secrecy Issues with Instruction
- 1a. Instruct on Secrecy Issues to 1. Security Requirements Elicitation with Requirements
- Security Engineer to 1. Security Requirements Elicitation with Requirements
UML Activity Diagrams: Control flow

**Related**
- Storage can be modeled

**Different**
- Synchronisation of control
  - No intuitive symbols
  - Nothing fluid
  - No data flow

**Stereotypes** can do almost everything - even emulate FLOW
Little JILL: Process programming

**Related**
- Dependencies modeled
- Information represented

**Different**
- Complex, detailed notation
- Flows only implicit
- Fluid not defined
RCSN: Observing real interaction

Related
- Information flow
- Including fluid/informal flows

Different
- Automated collection
- Description, not prescription
- Observing, not designing flows
- Proposal: not yet applied (2006)

Legend:
Tina is a Security Instructor
Tim and Christine are Stakeholders
John is a Security Engineer
Summary of related notations

• We do not rank notations!

• We try to express our key concepts
  – Information flow
  – Experience
  – Fluid information

• Surprise: Often difficult to express in “related notation“
  Different purpose \(\rightarrow\) subtle differences \(\rightarrow\) hard to express concepts

• Conclusion: It is worth-while considering FLOW
Conclusions

• Let‘s face it: Not all requirements are documented

• That‘s fine!

• If we model fluid representations, too

• Relax: No formal notation required

• Modeling is a crucial learning process

• It requires good (=simple) visualizations

We propose: FLOW!
Modeling purpose and applications

Awareness and Overview

Improving Requirements Processes and Practices

Defining and Tailoring Communication

see: Software Engineering (SE’07)