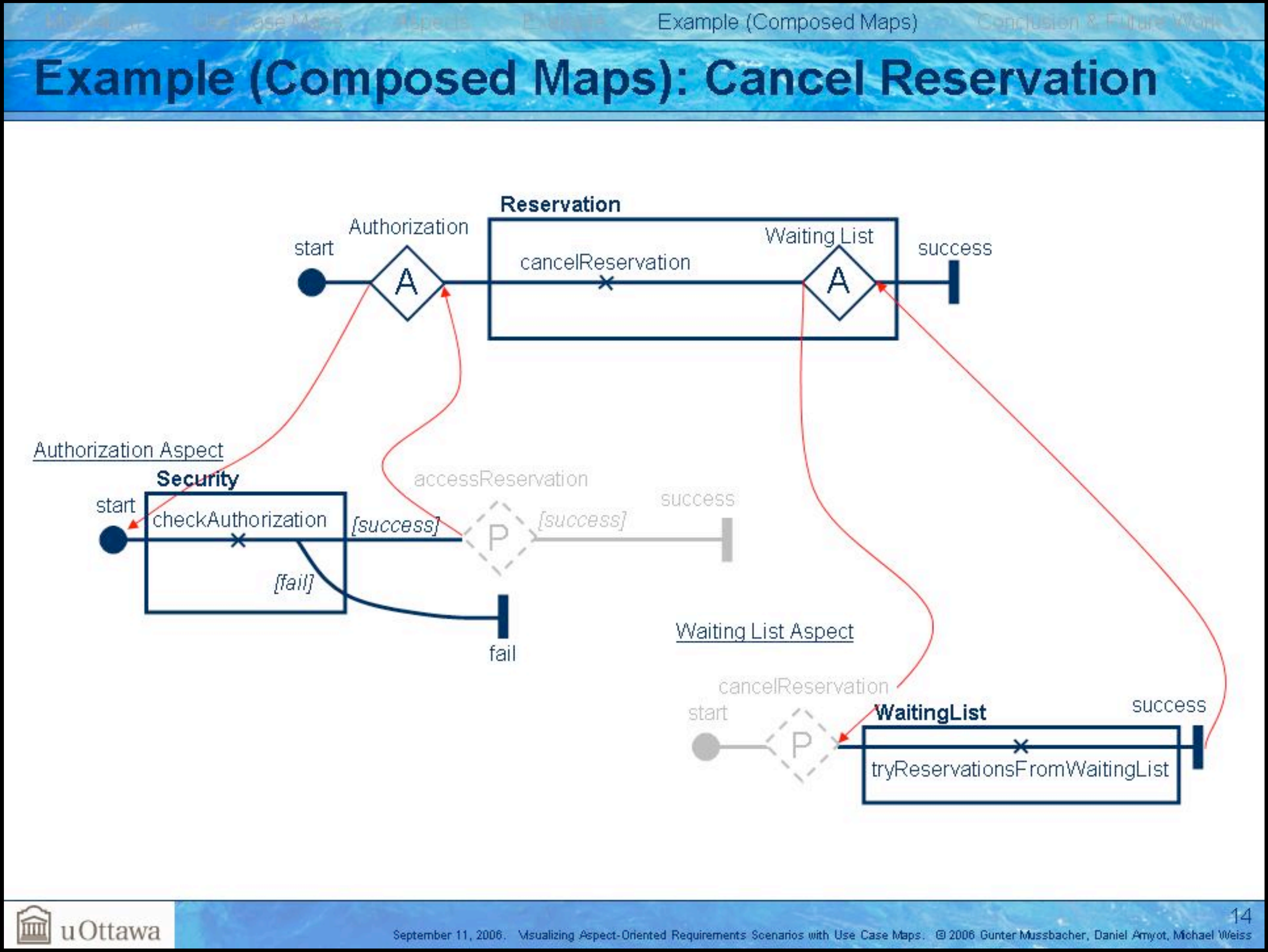
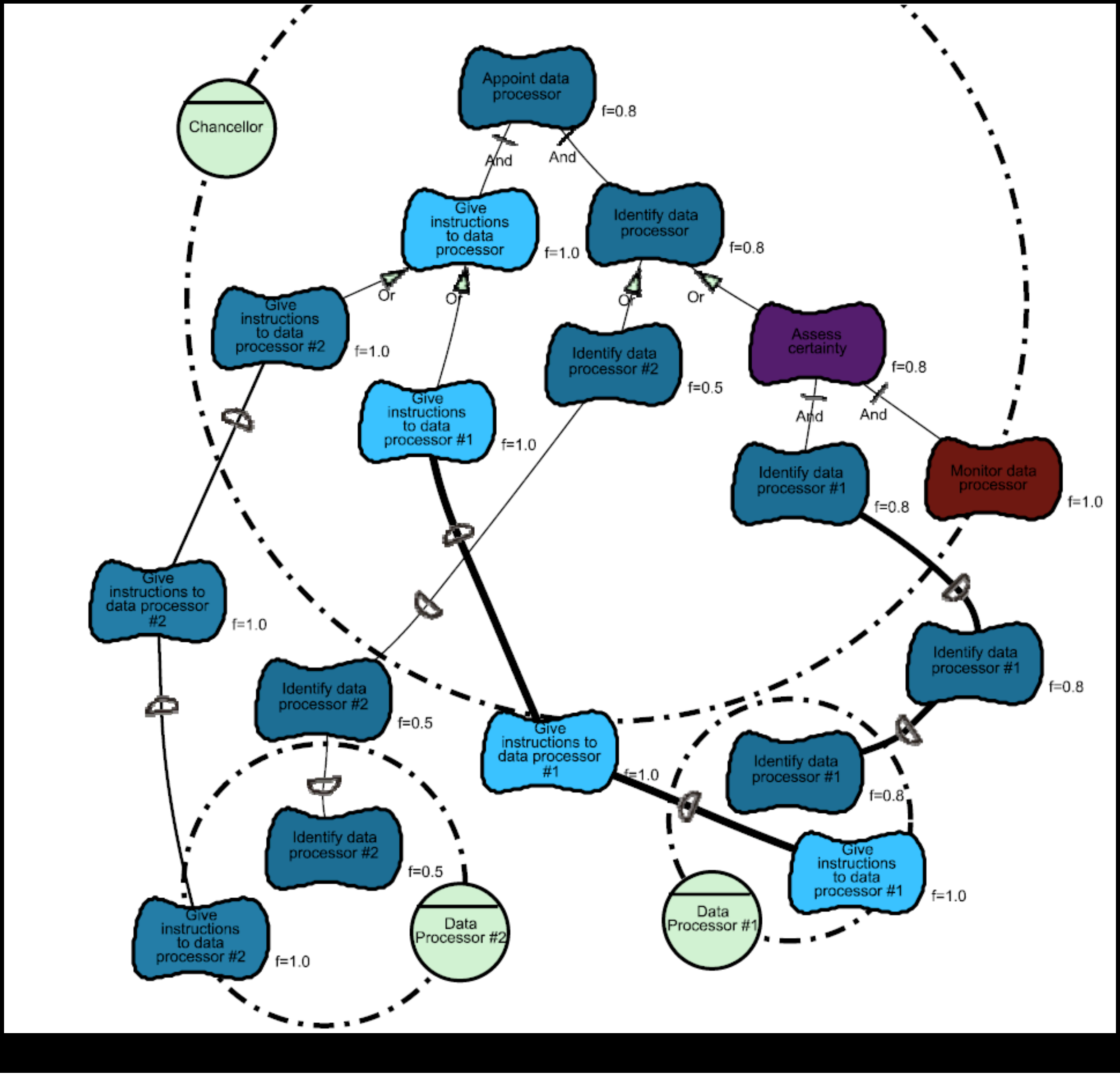
REV'06 VISUALIZATIONS

Gunter Mussbacher (University of Ottawa), Daniel Amyot (University of Ottawa) and Michael Weiss (Carleton University), from “Visualizing Aspect-Oriented Requirements Scenarios With Use Case Maps”

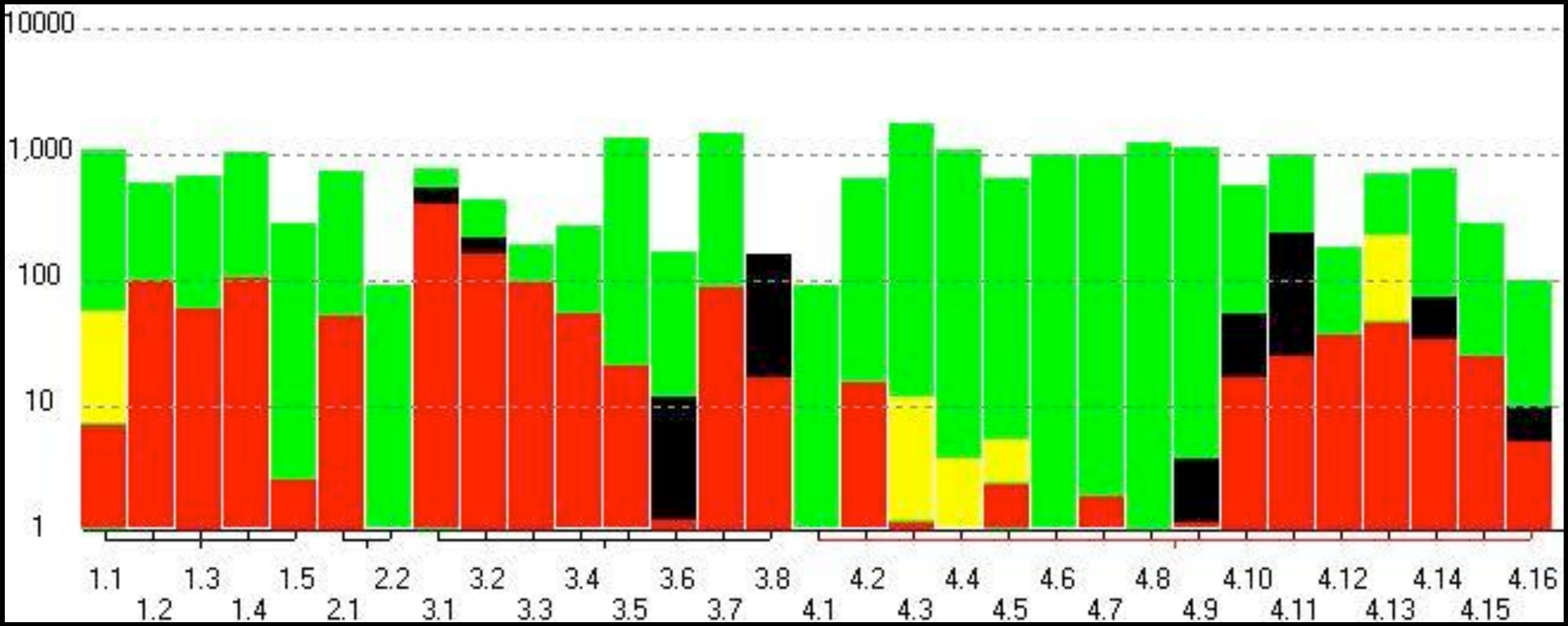


Example (Composed Maps): Cancel Reservation

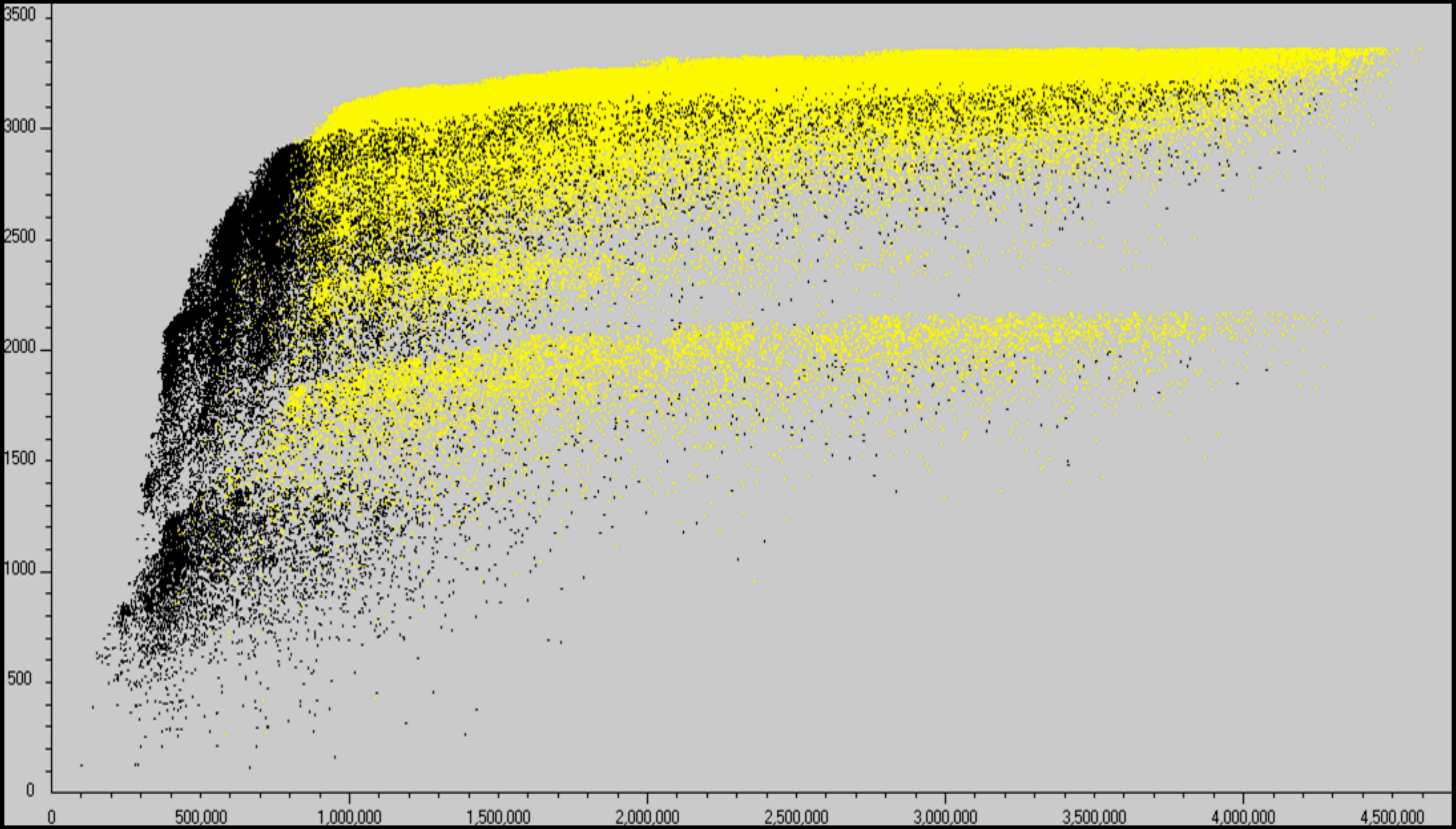
Neil A. Ernst, Yijun Yu and John Mylopoulos (University of Toronto), from “Visualizing Non-Functional Requirements”


Chancellor
Appoint data processor
f=0.8
And
And
Give instructions to data processor
f=1.0
Identify data processor
f=0.8
Give instructions to data processor #2
f=1.0
Or
Or
Or
Or
Identify data processor #2
f=0.5
Assess certainty
f=0.8
Give instructions to data processor #1
f=1.0
And
And
Identify data processor #1
f=0.8
Monitor data processor
f=1.0
Give instructions to data processor #2
f=1.0
Identify data processor #1
f=0.8
Identify data processor #2
f=0.5
Give instructions to data processor #1
f=1.0
Identify data processor #1
f=0.8
Identify data processor #2
f=0.5
Give instructions to data processor #1
f=1.0
Data Processor #1
Data Processor #2
Give instructions to data processor #2
f=1.0

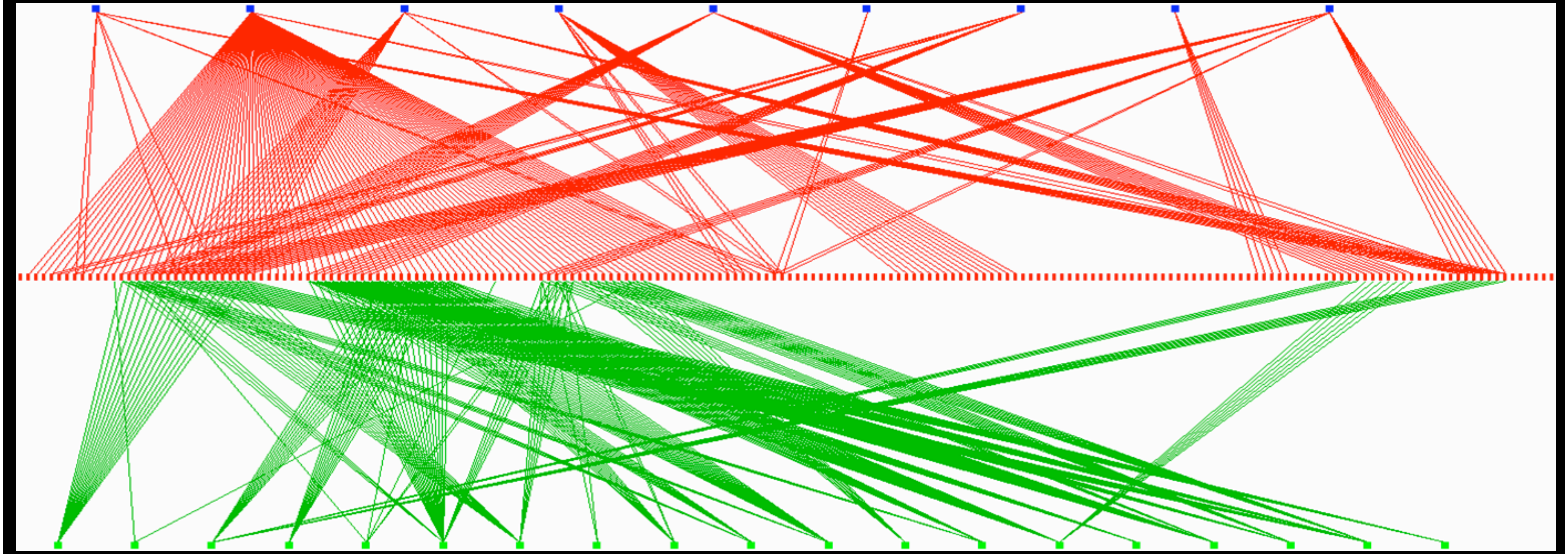
Martin S. Feather (California Institute of Technology), Steven L. Cornford (California Institute of Technology), James D. Kiper (Miami University) and Tim Menzies (West Virginia University), from “Experiences using Visualization Techniques to Present Requirements, Risks to Them and Options for Risk Mitigation”



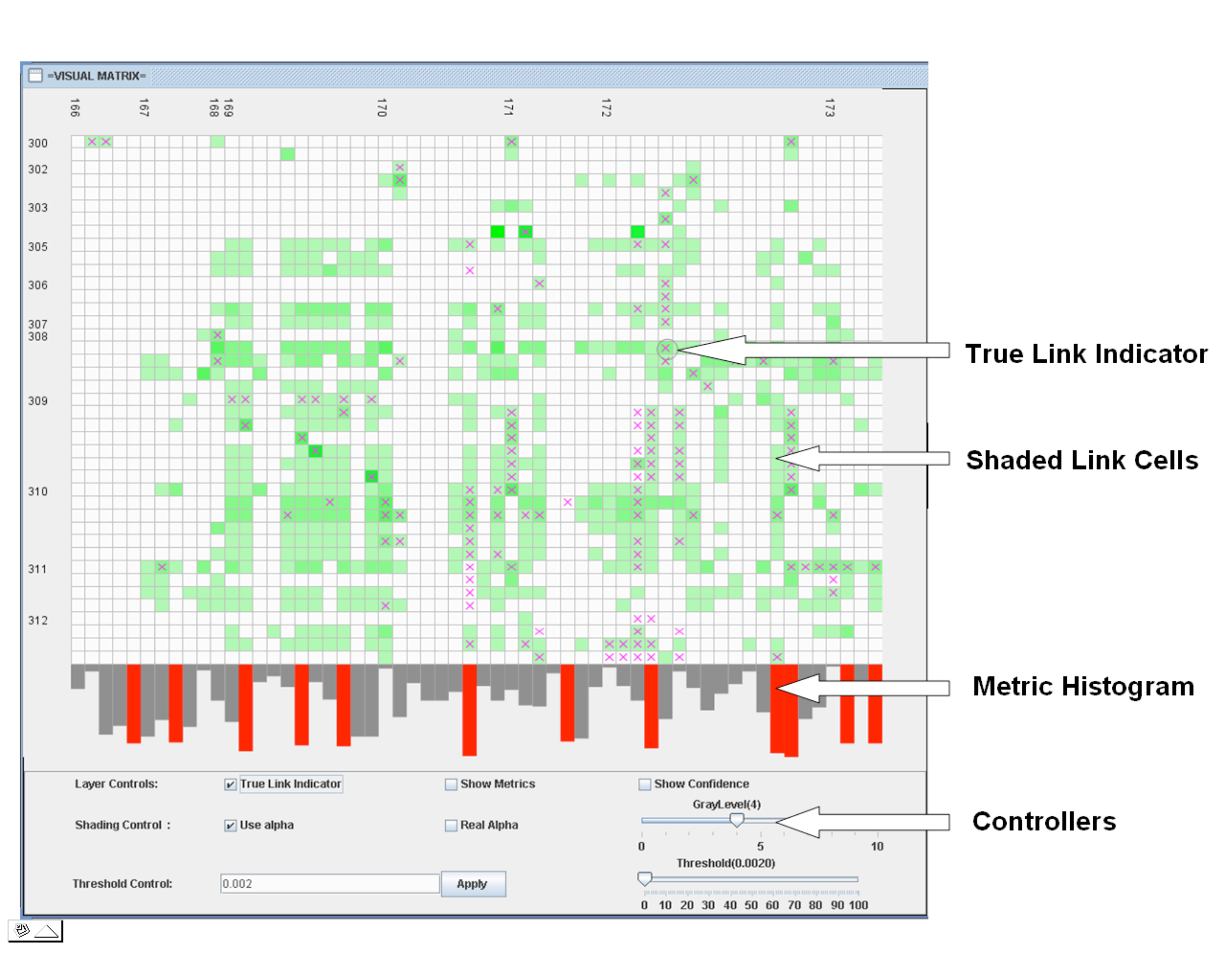
Martin S. Feather (California Institute of Technology), Steven L. Cornford (California Institute of Technology), James D. Kiper (Miami University) and Tim Menzies (West Virginia University), from “Experiences using Visualization Techniques to Present Requirements, Risks to Them and Options for Risk Mitigation”



Martin S. Feather (California Institute of Technology), Steven L. Cornford (California Institute of Technology), James D. Kiper (Miami University) and Tim Menzies (West Virginia University), from “Experiences using Visualization Techniques to Present Requirements, Risks to Them and Options for Risk Mitigation”



Chuan Duan and Jane Cleland-Huang (DePaul University), from "Visualization and Analysis In Automated Trace Retrieval"



=VISUAL MATRIX=

Layer Controls: ☑ True Link Indicator □ Show Metrics □ Show Confidence

Shading Control : ☑ Use alpha □ Real Alpha

GrayLevel(4)

Threshold Control: 0.002 Apply

Threshold(0.0020)

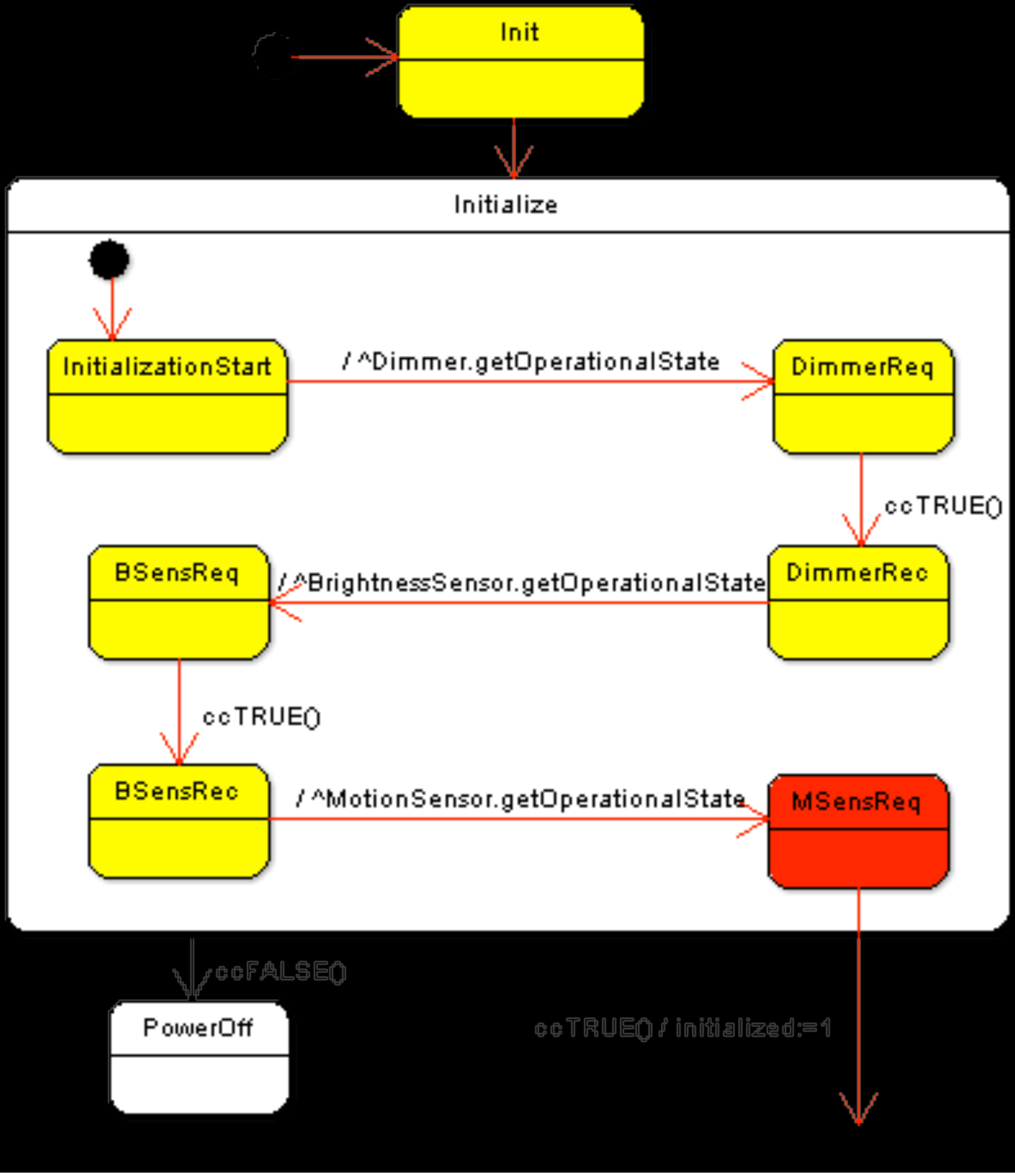
True Link Indicator

Shaded Link Cells

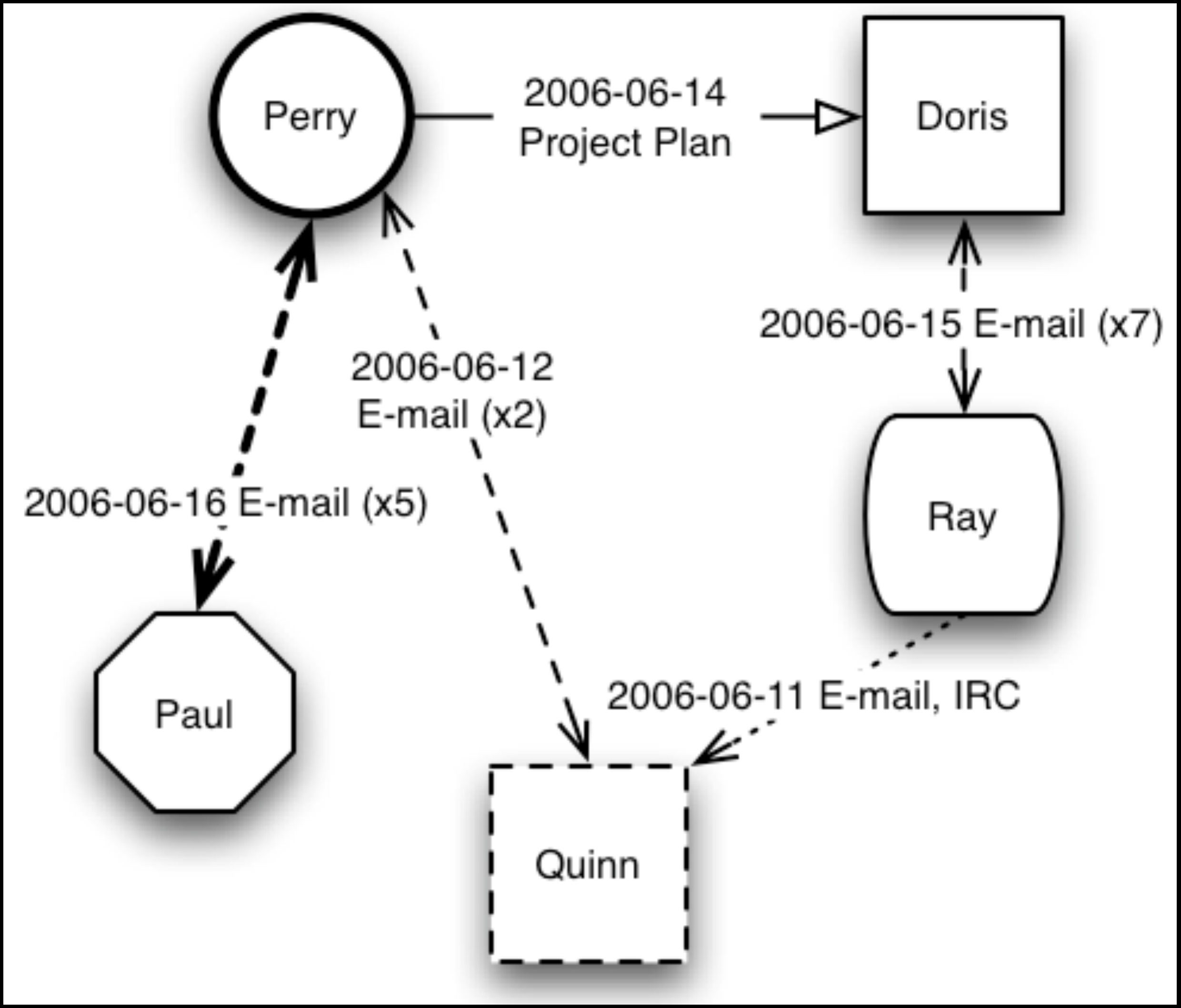
Metric Histogram

Controllers

Sascha Konrad, Heather Goldsby, Karli Lopez and Betty H.C. Cheng (Michigan State University), from “Visualizing Requirements in UML Models”

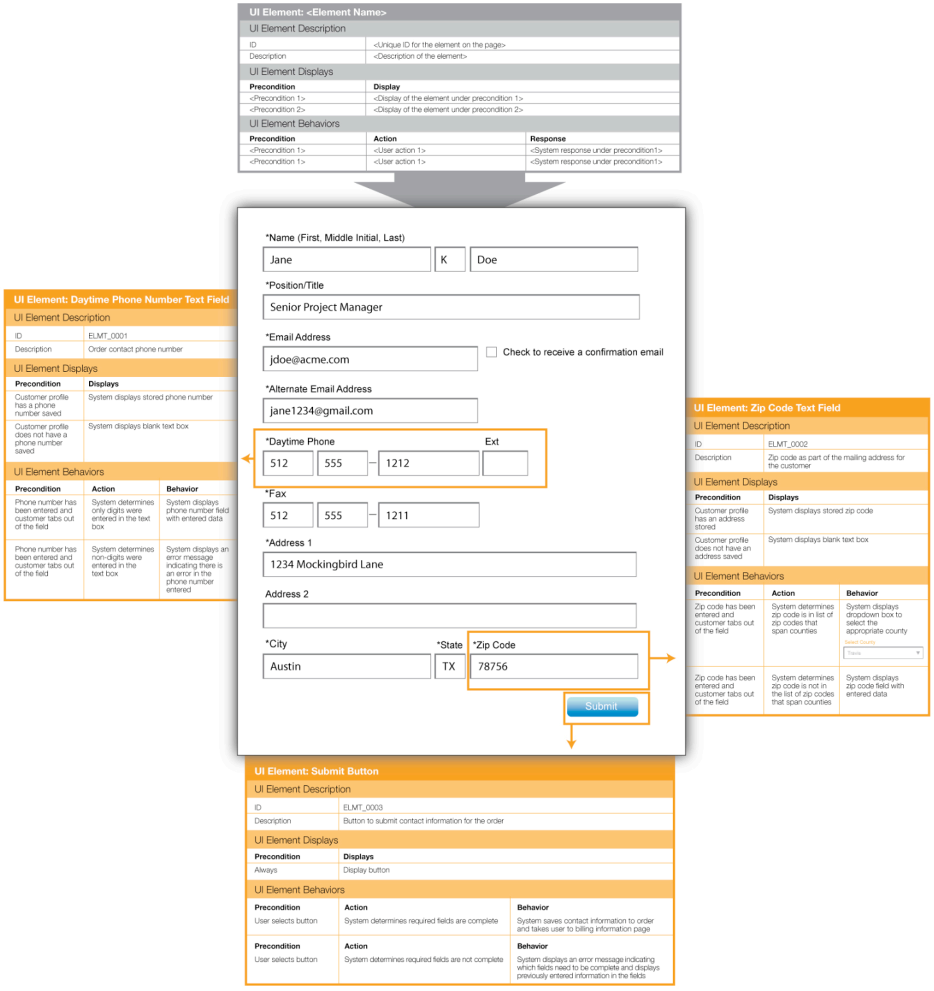

Init
Initialize
InitializationStart
/ ^Dimmer.getOperationalState
DimmerReq
ccTRUE()
DimmerRec
/ ^BrightnessSensor.getOperationalState
BSensReq
ccTRUE()
BSensRec
/ ^MotionSensor.getOperationalState
MSensReq
ccFALSE()
PowerOff
ccTRUE() / initialized:=1

Irwin Kwan, Daniela Damian and Margaret-Anne Storey (University of Victoria), from “Visualizing Requirements-Centered Social Network to Maintain Awareness within Development Teams”



REV'07 VISUALIZATIONS

Display-Action-Response (DAR) Model for User Interface Requirements



UI Element: <Element Name>		
UI Element Description		
ID	<Unique ID for the element on the page>	
Description	<Description of the element>	
UI Element Displays		
Precondition	**Display**	
<Precondition 1>	<Display of the element under precondition 1>	
<Precondition 2>	<Display of the element under precondition 2>	
UI Element Behaviors		
Precondition	**Action**	**Response**
<Precondition 1>	<User action 1>	<System response under precondition1>
<Precondition 1>	<User action 1>	<System response under precondition1>

*Name (First, Middle Initial, Last)
Jane | K | Doe

*Position/Title
Senior Project Manager

*Email Address
jdoe@acme.com
□ Check to receive a confirmation email

*Alternate Email Address
jane1234@gmail.com

*Daytime Phone: 512 555 – 1212 Ext

*Fax: 512 555 – 1211

*Address 1
1234 Mockingbird Lane

Address 2

*City: Austin
*State: TX
*Zip Code: 78756

UI Element: Daytime Phone Number Text Field		
UI Element Description		
ID	ELMT_0001	
Description	Order contact phone number	
UI Element Displays		
Precondition	**Displays**	
Customer profile has a phone number saved	System displays stored phone number	
Customer profile does not have a phone number saved	System displays blank text box	
UI Element Behaviors		
Precondition	**Action**	**Behavior**
Phone number has been entered and customer tabs out of the field	System determines only digits were entered in the text box	System displays phone number field with entered data
Phone number has been entered and customer tabs out of the field	System determines non-digits were entered in the text box	System displays an error message indicating there is an error in the phone number entered

UI Element: Zip Code Text Field		
UI Element Description		
ID	ELMT_0002	
Description	Zip code as part of the mailing address for the customer	
UI Element Displays		
Precondition	**Displays**	
Customer profile has an address stored	System displays stored zip code	
Customer profile does not have an address saved	System displays blank text box	
UI Element Behaviors		
Precondition	**Action**	**Behavior**
Zip code has been entered and customer tabs out of the field	System determines zip code is in list of zip codes that span counties	System displays dropdown box to select the appropriate county Select County Travis ▾
Zip code has been entered and customer tabs out of the field	System determines zip code is not in the list of zip codes that span counties	System displays zip code field with entered data

UI Element: Submit Button		
UI Element Description		
ID	ELMT_0003	
Description	Button to submit contact information for the order	
UI Element Displays		
Precondition	**Displays**	
Always	Display button	
UI Element Behaviors		
Precondition	**Action**	**Behavior**
User selects button	System determines required fields are complete	System saves contact information to order and takes user to billing information page
Precondition	**Action**	**Behavior**
User selects button	System determines required fields are not complete	System displays an error message indicating which fields need to be complete and displays previously entered information in the fields

Joy Beatty and Mike Alexander

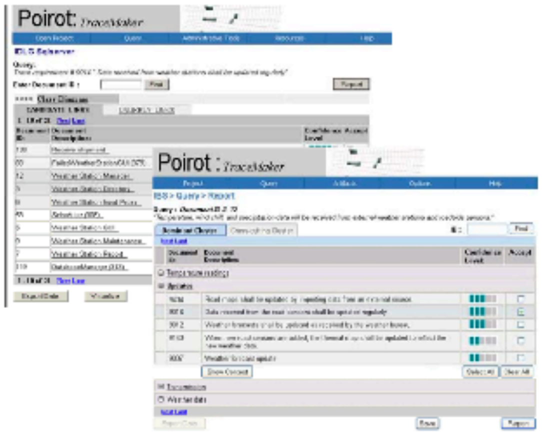
Seilevel
requirements defined

joy.beatty@seilevel.com
mike.alexander@seilevel.com

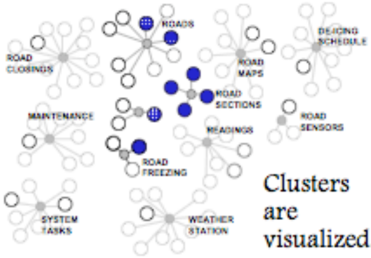
Visual Support In Automated Tracing

Automated traceability generates candidate links between requirements and other software artifacts.

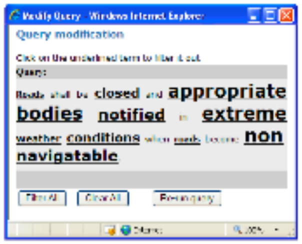
An analyst must manually evaluate the results.



Results are clustered to increase comprehension.



Clusters are visualized.



Query modification

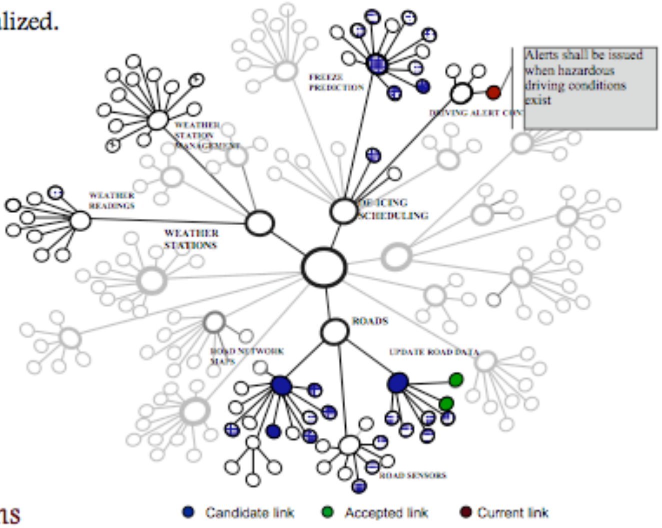
Click on the underlined term to filter it out

Query:

Roads shall be closed and appropriate bodies notified in extreme weather conditions when roads become non navigatable

Tag clouds help an analyst understand why a particular artifact is linked to the query.

Visualizations help analysts to make decisions during trace analysis.



Links are displayed within the context of a physical hierarchy.

Jane Cleland-Huang and Rafal Habrat, DePaul University

Visual Analytics for Requirements-driven Risk Assessment

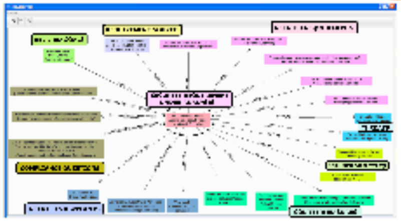
Robin A. Gandhi, Seok-Won Lee

Knowledge-Intensive Software Engineering (NISE) Research Group, College of Computing and Informatics, UNC Charlotte, NC, USA

1 The Position

- **Requirements Visualization** plays an important role in dealing with the complexities from
  - Numerous Dependability Requirements
  - Software-intensive Systems
  - Socio-technical Environments
- **Visual Analytics for Requirements** is the creation of abstract *visual metaphors* based on quantitative and qualitative requirements metrics that help to systematically reason about the related software behavior in a large information space

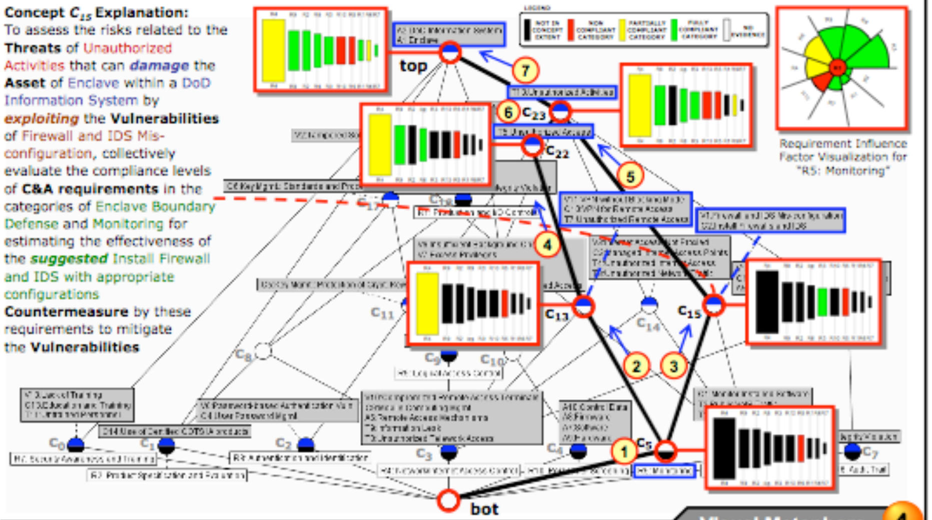
Multi-dimensional Requirements Visualization 2



3 Visualizing Non-compliance Impact Analysis for Risk Assessment

- **Step 1 (Localization):** Locate the most specific formal concept that characterizes the non-compliant requirement categories (**Concept C_5 for "R5: Monitoring" labeled as 1**)
- **Step 2 (Traversal):** The impact of non-compliance is gradually propagated at higher levels of abstraction in the concept lattice. (Paths labeled as **2, 3, 4, 5, 6, & 7**)

Concept C_{15} Explanation:
To assess the risks related to the **Threats** of Unauthorized Activities that can ***damage*** the **Asset** of Enclave within a DoD Information System by ***exploiting*** the **Vulnerabilities** of Firewall and IDS Mis-configuration, collectively evaluate the compliance levels of **C&A requirements** in the categories of Enclave Boundary Defense and Monitoring for estimating the effectiveness of the ***suggested*** Install Firewall and IDS with appropriate configurations **Countermeasure** by these requirements to mitigate the **Vulnerabilities**

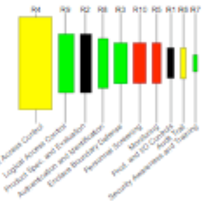


Requirement Influence Factor Visualization for "R5: Monitoring"

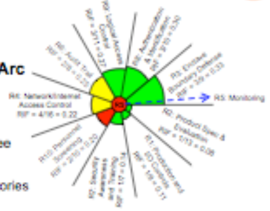
Visual Metaphors 4

- **Cohesive Bar Graph**
  - **Visualization Context:** Formal Concept
- **Visual Features of a Bar**
  - **Color:** Requirement category presence and compliance level
  - **Height:** Requirement category correlation Index
  - **Width:** Risk coverage of a requirement category
  - **Order:** Relative criticality

Concept C_{22}



- **Cohesive Arc Graph**
  - **Visualization Context:** Requirement Category
- **Visual Features of an Arc**
  - **Color:** Requirement category compliance level
  - **Arc Radius:** *Requirement influence factor* is the degree of similarity between two requirement categories in correlating with other categories in the given scenario



"R5: Monitoring" Requirement Category

5 References

- Gandhi, R.A., Lee, S.W.: Discovering and Understanding Multi-dimensional Correlations among Certification Requirements with application to Risk Assessment, India, RE 07
- Lee, S.W., Gandhi, R.A.: Requirements as Enablers for Software Assurance, CrossTalk: The Journal of Defense Software Engineering, December Issue, 19(12), 2006, pp. 20-24.

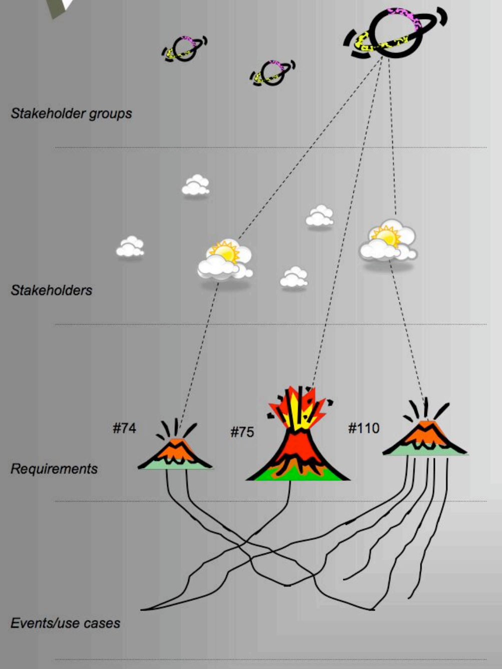
PACE UNIVERSITY

CITY

On Requirements Visualization

Orlena C.Z. Gotel, Francis T. Marchese and Stephen J. Morris

Big picture

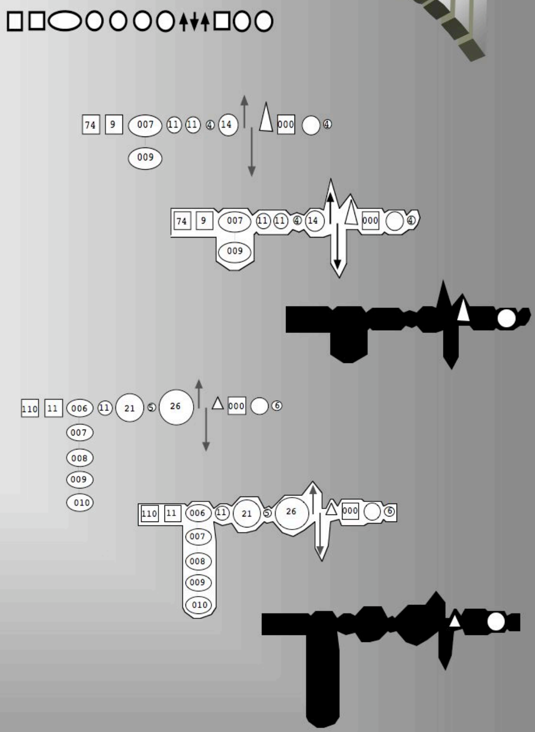


From page 157 of [1] :
Req #: 75
Req Type: 9 (functional requirement)
Event/Use Case #: 6
Description: The product shall issue an alert if a weather station fails to transmit readings.
Rationale: Failure to transmit readings might indicate that the weather station is faulty and needs maintenance, and that the data used to predict freezing roads may be incomplete.
Source: Road Engineers
Fit Criterion: For each weather station the product shall communicate to the user when the recorded number of each type of reading per hour is not within the manufacturer's specified range of the expected number of readings per hour.
Customer Satisfaction: 3
Customer Dissatisfaction: 5
Dependencies: None
Conflicts: None
Supporting Materials: Specification of Rosa Weather Station
History: Raised by GBS, 28 July 99

From web site of [1] :
Req #: 74
Req Type: 9 (functional requirement)
Event/Use Case #: 7, 9
Description: The product shall record all the roads that have been treated.
Rationale: To be able to schedule untreated roads and highlight potential danger.
Source: Arnold Snow, Chief Engineer
Fit Criterion: The recorded treated and untreated roads shall agree with the drivers' road treatment logs.
Customer Satisfaction: 3
Customer Dissatisfaction: 5
Dependencies: None
Conflicts: None
Supporting Materials: None
History: Created February 29, 2006

From page 159 of [1]:
Req #: 110
Req Type: 11 (non-functional requirement - usability)
Event/Use Case #: 6, 7, 8, 9, 10
Description: The product shall be easy for the road engineers to use.
Rationale: It should not be necessary for the engineers to attend training classes in order to be able to use the product.
Source: Sonia Henning, Road Engineering Supervisor
Fit Criterion: A road engineer shall be able to use the product to successfully carry out the cited use cases within 1 hour of first encountering the product.
Customer Satisfaction: 3
Customer Dissatisfaction: 5
Dependencies: None
Conflicts: None
Supporting Materials: None
History: Raised by AG, 25 Aug 99

Detail



Health

REQ	Value	Source	Rationale	Fit
# 74	☹	☺	☺	☺
# 75	☹	😐	☺	☺
# 110	☹	☺	☺	☺

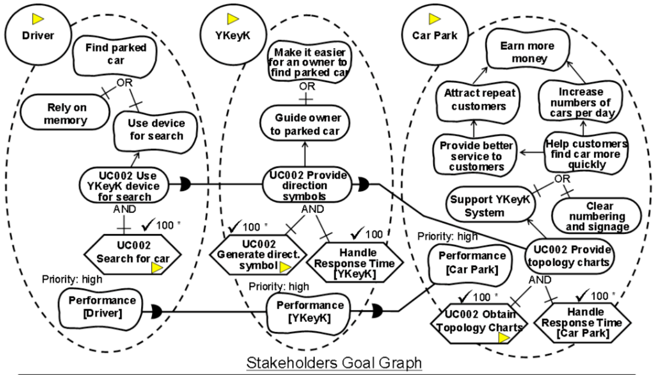
[1] Robertson, S. and Roberson, J. *Mastering the Requirements Process*, ACM Press, 1999 *www.systemsguild.com/GuildSite/Robs/Template.html*

Contacts: *ogotel@pace.edu*, *fmarchese@pace.edu*, *sjm@soi.city.ac.uk*

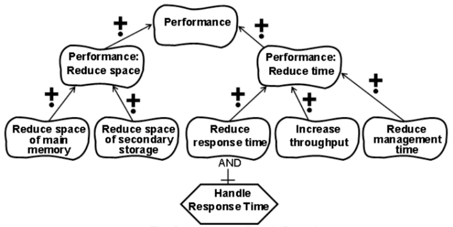
Visualizing Aspect-Oriented Goal Models with AoGRL

© 2007, Gunter Mussbacher, SITE, University of Ottawa, Canada; gunterm@site.uottawa.ca

GRL Model:

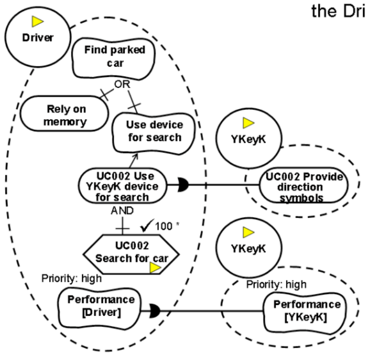

Driver
Find parked car
OR
Rely on memory
Use device for search
UC002 Use YKeyK device for search
AND
100
UC002 Search for car
Priority: high
Performance [Driver]
YKeyK
Make it easier for an owner to find parked car
OR
Guide owner to parked car
UC002 Provide direction symbols
AND
100
UC002 Generate direct. symbol
100
Handle Response Time [YKeyK]
Priority: high
Performance [YKeyK]
Car Park
Earn more money
Attract repeat customers
Increase numbers of cars per day
Provide better service to customers
Help customers find car more quickly
OR
Support YKeyK System
Clear numbering and signage
Priority: high
Performance [Car Park]
UC002 Provide topology charts
AND
100
UC002 Obtain Topology Charts
100
Handle Response Time [Car Park]

Stakeholders Goal Graph



Performance Goal Graph

The Driver goal graph shown below is an alternative to the Stakeholders goal graph shown above, focusing on one stakeholder at the time (only the Driver is shown).



Driver Goal Graph

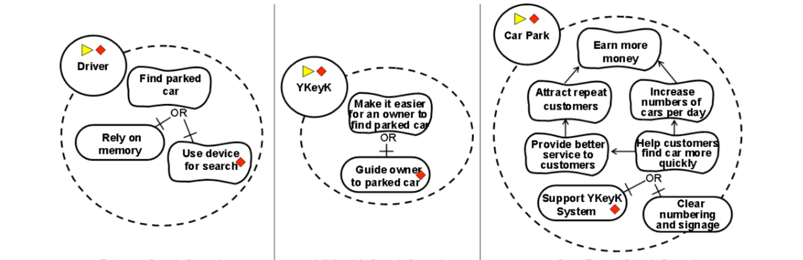
Restructure!

With the help of aspect-oriented techniques the major concerns in the GRL model are now better encapsulated.

Typical major concerns are stakeholders, non-functional requirements, and solutions (i.e., use cases).

Note that the details of use cases are not shown because they are modeled with Use Case Maps (UCMs) and only linked from the GRL model with URN links.

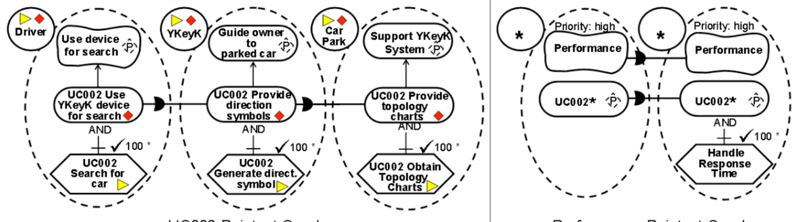
AoGRL Model:


Driver
Find parked car
OR
Rely on memory
Use device for search
YKeyK
Make it easier for an owner to find parked car
OR
Guide owner to parked car
Car Park
Earn more money
Attract repeat customers
Increase numbers of cars per day
Provide better service to customers
Help customers find car more quickly
OR
Support YKeyK System
Clear numbering and signage

Driver Goal Graph

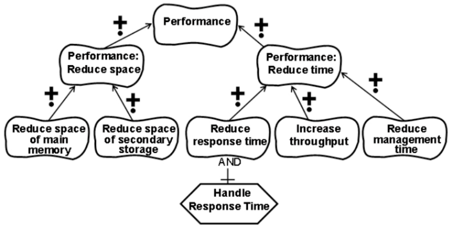
YKeyK Goal Graph

Car Park Goal Graph



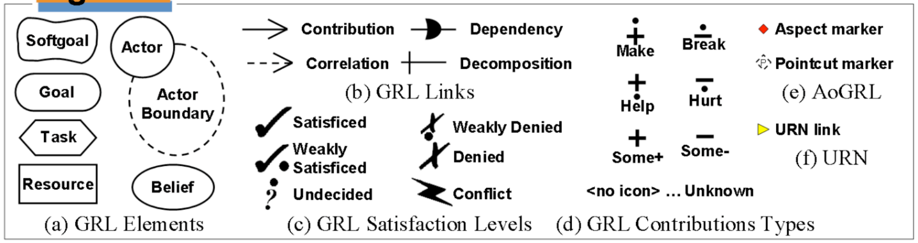
UC002 Pointcut Graph

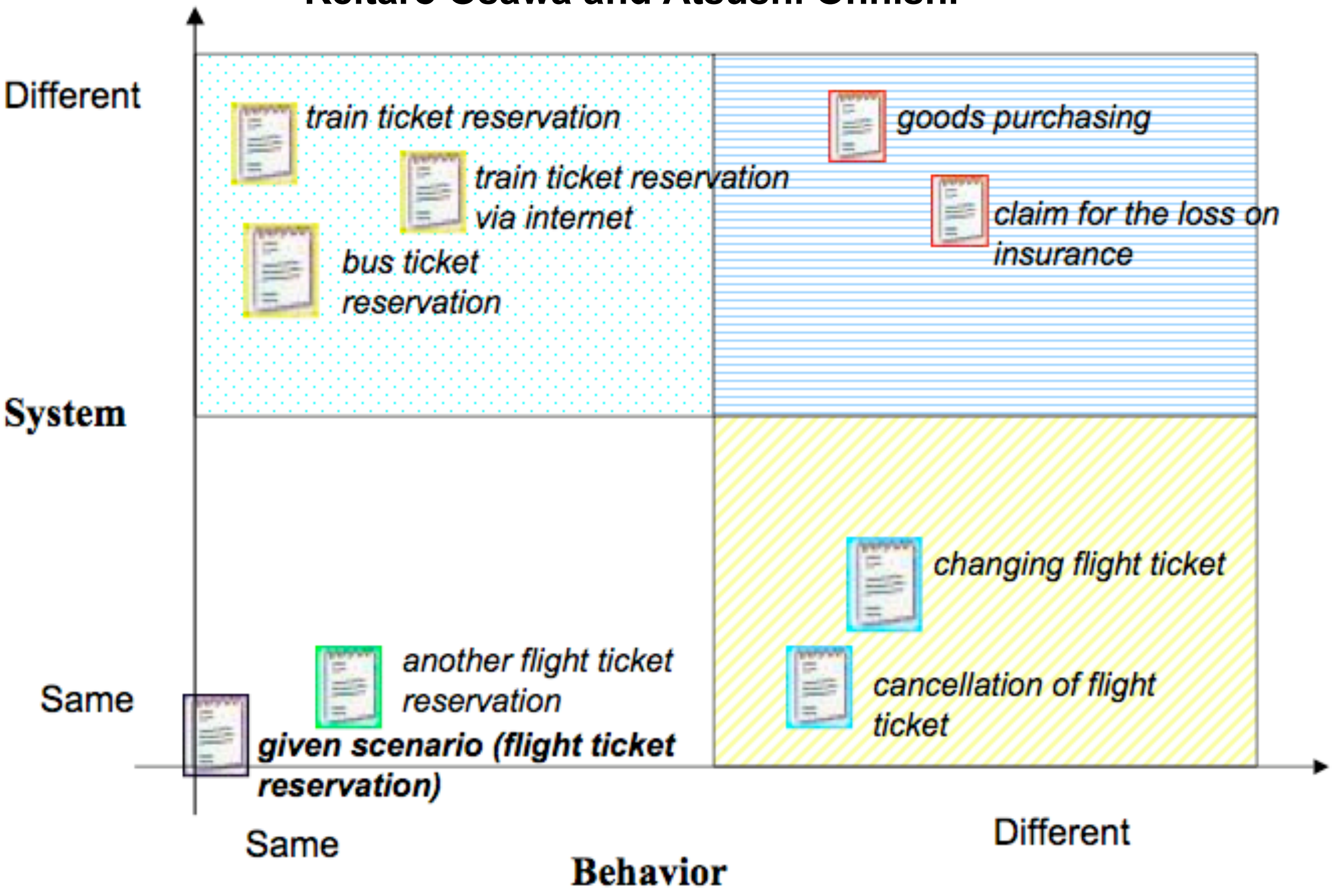
Performance Pointcut Graph



Performance Goal Graph

Legend:

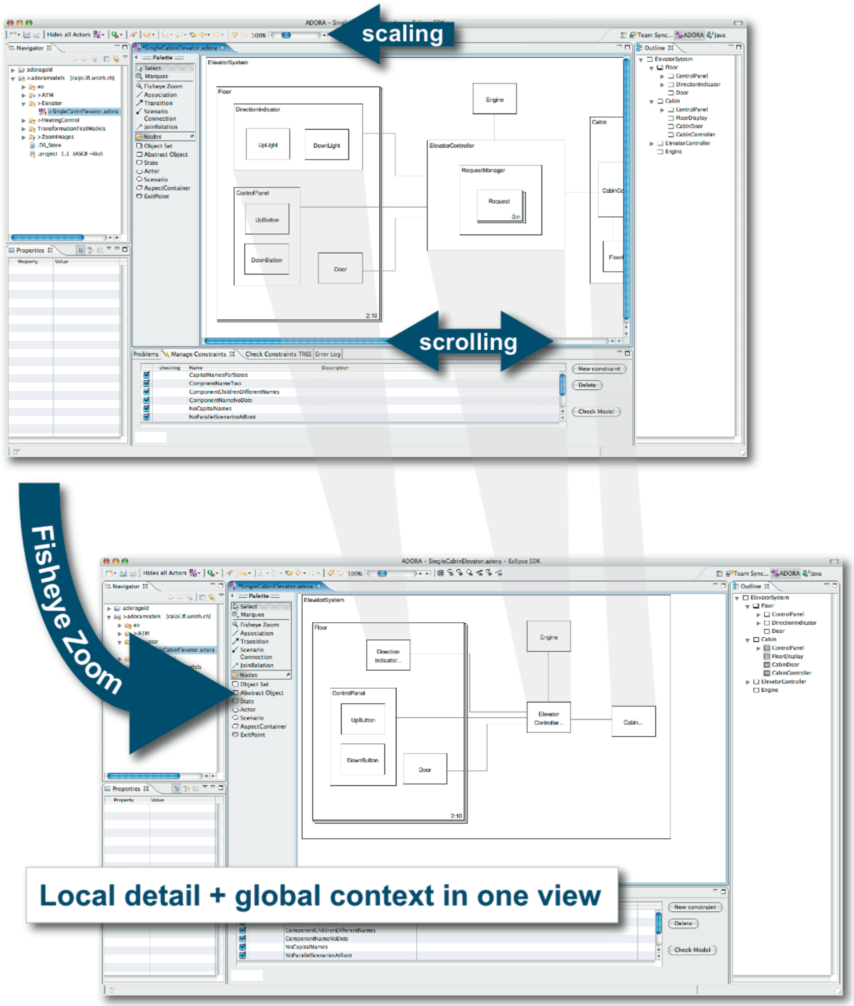



Different
train ticket reservation
train ticket reservation via internet
bus ticket reservation
goods purchasing
claim for the loss on insurance
System
changing flight ticket
another flight ticket reservation
Same
cancellation of flight ticket
given scenario (flight ticket reservation)
Same
Different
Behavior

Scenario Similarity Map

An Improved Fisheye Zoom Algorithm for Visualizing and Editing Hierarchical Models

Tobias Reinhard, Silvio Meier, Martin Glinz
University of Zurich



Local detail + global context in one view

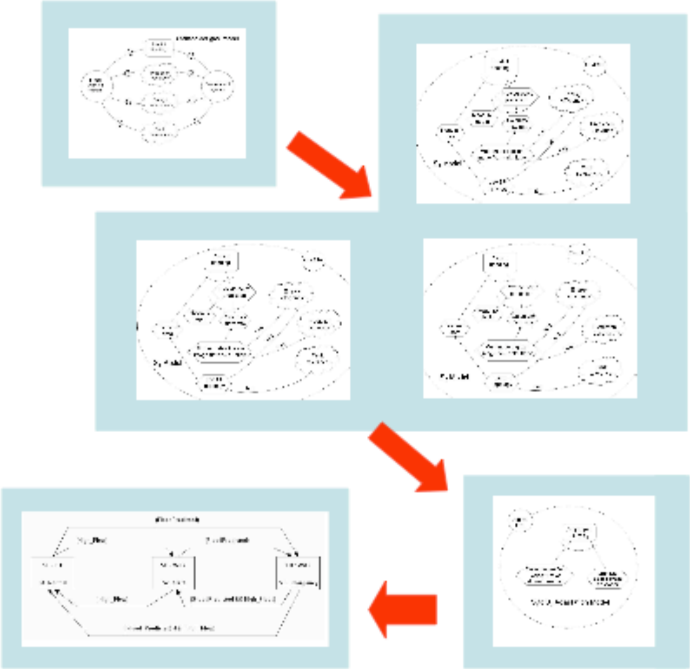
REV'07

University of Zurich
RERG

Visualizing the Analysis of Dynamically Adaptive Systems Using *i** and DSLs

Peter Sawyer[1], Nelly Bencomo[1], Heather J. Goldsby[3], Betty H.C. Cheng[3], Danny Hughes[1], Paul Grace[2]

[1] Computing department, InfoLab21, Lancaster University, LA1 4WA, UK
[2] Departement Computerwetenschappen, Katholieke Universiteit Leuven, B-3001 Heverlee, Belgium
[3] Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824, USA
(sawyer, nelly danny) @comp.lancs.ac.uk; Paul.Grace@cs.kuleuven.be; (hjg, chengb)@cse.msu.edu



This work has been supported in part by NSF grants EIA-0000433, EIA-0130724, CDA-9700732, CCR-9901017, Department of the Navy, Office of Naval Research under Grant No. N00014-01-1-0744, Eaton Corporation, Siemens Corporate Research, a grant from Michigan State University's Quality Fund and EPSRC projectEP/C010345/1 "The Divergent Grid"

MICHIGAN STATE UNIVERSITY

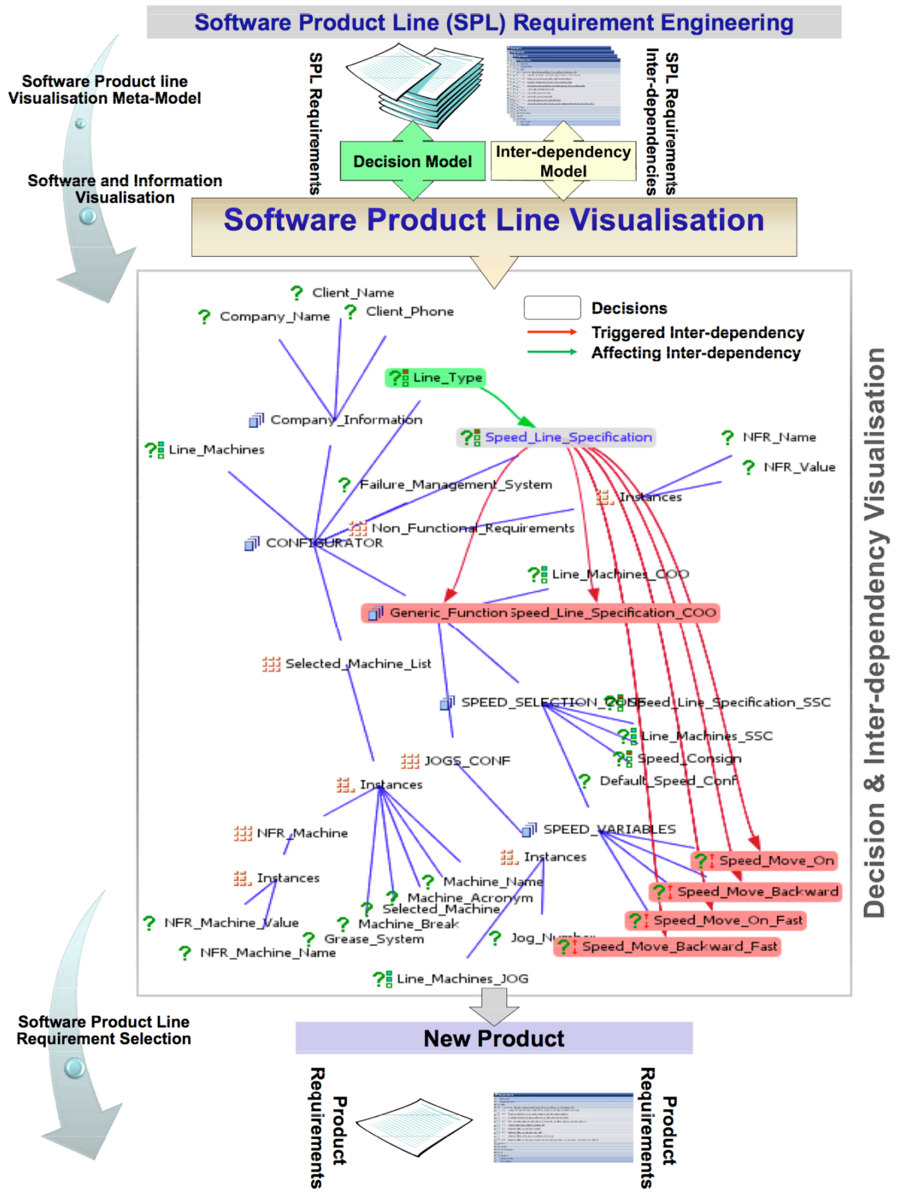
LANCASTER UNIVERSITY

Visualising Product Line Requirement Selection Decision Inter-dependencies

David H. Sellier, Mike Mannion

Glasgow Caledonian University,

(David.Sellier@gcal.ac.uk, M.A.G.Mannion@gcal.ac.uk)



Software Product Line (SPL) Requirement Engineering

Software Product line Visualisation Meta-Model

Software and Information Visualisation

SPL Requirements

Decision Model

Inter-dependency Model

SPL Requirements Inter-dependencies

Software Product Line Visualisation

Decision & Inter-dependency Visualisation

Decisions

Triggered Inter-dependency

Affecting Inter-dependency

Software Product Line Requirement Selection

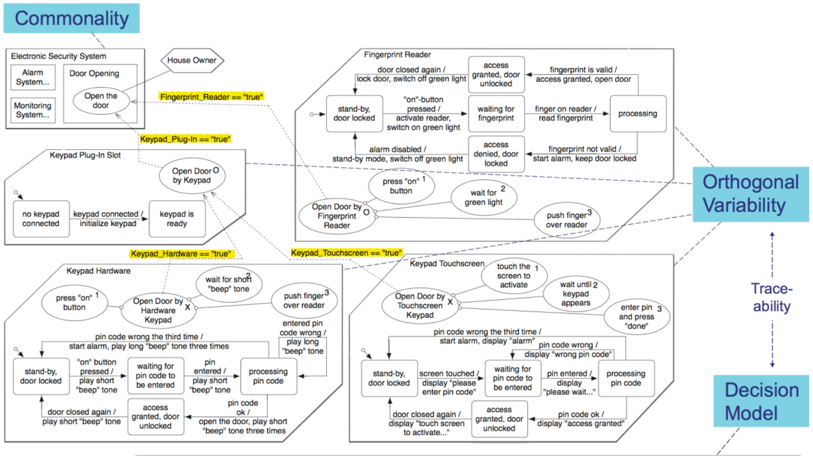
New Product

Product Requirements

Product Requirements

Visualizing Product Line Domain Variability by Aspect-Oriented Modeling

Reinhard Stoiber, Silvio Meier, Martin Glinz
University of Zurich


Commonality
Electronic Security System
Alarm System...
Monitoring System...
Door Opening
Open the door
House Owner
Fingerprint_Reader == "true"
Keypad_Plug-In == "true"
Keypad Plug-In Slot
Open Door by Keypad
no keypad connected
keypad connected / initialize keypad
keypad is ready
Fingerprint Reader
door closed again / lock door, switch off green light
access granted, door unlocked
fingerprint is valid / access granted, open door
stand-by, door locked
"on"-button pressed / activate reader, switch on green light
waiting for fingerprint
finger on reader / read fingerprint
processing
alarm disabled / stand-by mode, switch off green light
access denied, door locked
fingerprint not valid / start alarm, keep door locked
press "on" button 1
wait for green light 2
push finger over reader 3
Open Door by Fingerprint Reader
Orthogonal Variability
Keypad_Hardware == "true"
Keypad_Touchscreen == "true"
Keypad Hardware
press "on" button 1
Open Door by Hardware Keypad
wait for short "beep" tone 2
push finger over reader 3
entered pin code wrong / play long "beep" tone
pin code wrong the third time / start alarm, play long "beep" tone three times
stand-by, door locked
"on" button pressed / play short "beep" tone
waiting for pin code to be entered
pin entered / play short "beep" tone
processing pin code
door closed again / play short "beep" tone
access granted, door unlocked
pin code ok / open the door, play short "beep" tone three times
Keypad Touchscreen
touch the screen to activate 1
wait until keypad appears 2
enter pin and press "done" 3
Open Door by Touchscreen Keypad
pin code wrong the third time / start alarm, display "alarm"
pin code wrong / display "wrong pin code"
stand-by, door locked
screen touched / display "please enter pin code"
waiting for pin code to be entered
pin entered / display "please wait..."
processing pin code
door closed again / display "touch screen to activate..."
access granted, door unlocked
pin code ok / display "access granted"
Trace-ability
Decision Model

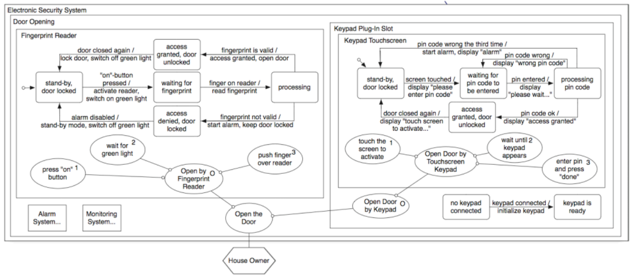
Name	Relevance	Description	Range	Multiplicity	Constraints	Binding Time
Fingerprint_Reader		Is there a fingerprint scanner?	true, false	1	Fingerprint_Reader = false -> Keypad_Plug-In = true	Installation
Keypad_Plug-In		Is there a Keypad for a numeral code?	true, false	1	Keypad_Plug-In = false -> Fingerprint_Reader = true	Installation
Keypad_Hardware	Keypad_Plug-In == true	Is it a Hardware Keypad?	true, false	1	Keypad_ Touchscreen = true -> Keypad_HW = false	Installation
Keypad_Touchscreen	Keypad_Plug-In == true	Is it a Touchscreen Keypad?	true, false	1	Keypad_ HW = true -> Keypad_ Touchscreen = false	Installation

... and how to use your Domain:

1. Do your Product Decisions

Fingerprint_Reader = "true"
Keypad_Plug-In = "true"
Keypad_Hardware = "false"
Keypad_Touchscreen = "true" ✓

2. Receive an automatically generated Application Product


Electronic Security System
Door Opening
Fingerprint Reader
door closed again / lock door, switch off green light
access granted, door unlocked
fingerprint is valid / access granted, open door
stand-by, door locked
"on"-button pressed / activate reader, switch on green light
waiting for fingerprint
finger on reader / read fingerprint
processing
alarm disabled / stand-by mode, switch off green light
access denied, door locked
fingerprint not valid / start alarm, keep door locked
wait for green light 2
push finger over reader 3
press "on" button 1
Open by Fingerprint Reader
Alarm System...
Monitoring System...
Open the Door
House Owner
Keypad Plug-In Slot
Keypad Touchscreen
pin code wrong the third time / start alarm, display "alarm"
pin code wrong / display "wrong pin code"
stand-by, door locked
screen touched / display "please enter pin code"
waiting for pin code to be entered
pin entered / display "please wait..."
processing pin code
door closed again / display "touch screen to activate..."
access granted, door unlocked
pin code ok / display "access granted"
touch the screen to activate 1
Open Door by Touchscreen Keypad
wait until keypad appears 2
enter pin and press "done" 3
Open Door by Keypad
no keypad connected
keypad connected / initialize keypad
keypad is ready

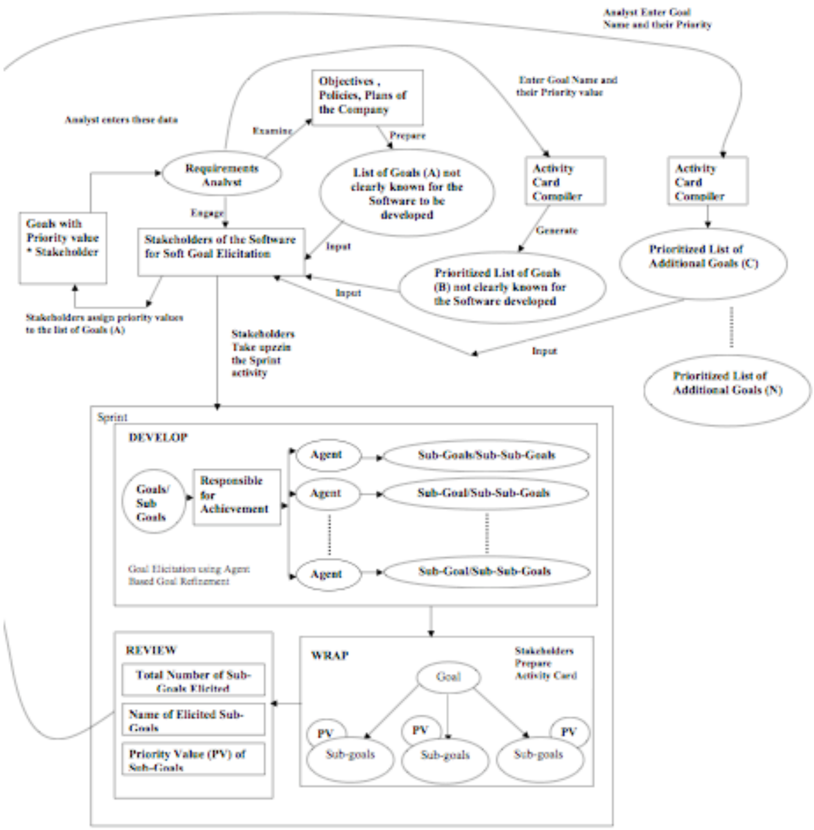
REV'07

Requirements Engineering Research Group
Department of Informatics
University of Zurich, Switzerland

http://www.ifi.uzh.ch/rerg

RE
RG

A Visual Technique for Agent Based Goal Refinement to Elicit Soft Goals in Goal oriented Requirements Engineering


Analyst Enter Goal Name and their Priority
Enter Goal Name and their Priority value
Analyst enters these data
Objectives , Policies, Plans of the Company
Examine
Prepare
Requirements Analyst
List of Goals (A) not clearly known for the Software to be developed
Activity Card Compiler
Activity Card Compiler
Engage
Goals with Priority value * Stakeholder
Stakeholders of the Software for Soft Goal Elicitation
Input
Generate
Prioritized List of Additional Goals (C)
Input
Prioritized List of Goals (B) not clearly known for the Software developed
Stakeholders assign priority values to the list of Goals (A)
Stakeholders Take upzzin the Sprint activity
Input
Prioritized List of Additional Goals (N)
Sprint
DEVELOP
Agent
Sub-Goals/Sub-Sub-Goals
Goals/ Sub Goals
Responsible for Achievement
Agent
Sub-Goal/Sub-Sub-Goals
Goal Elicitation using Agent Based Goal Refinement
Agent
Sub-Goal/Sub-Sub-Goals
REVIEW
Total Number of Sub-Goals Elicited
Name of Elicited Sub-Goals
Priority Value (PV) of Sub-Goals
WRAP
Stakeholders Prepare Activity Card
Goal
PV
PV
PV
Sub-goals
Sub-goals
Sub-goals

A Visualization Technique for Agent Based Goal Refinement to Elicit Soft Goals in Goal Oriented Requirements Engineering
A.M. Sen and S.K. Jain

REV'08 VISUALIZATIONS

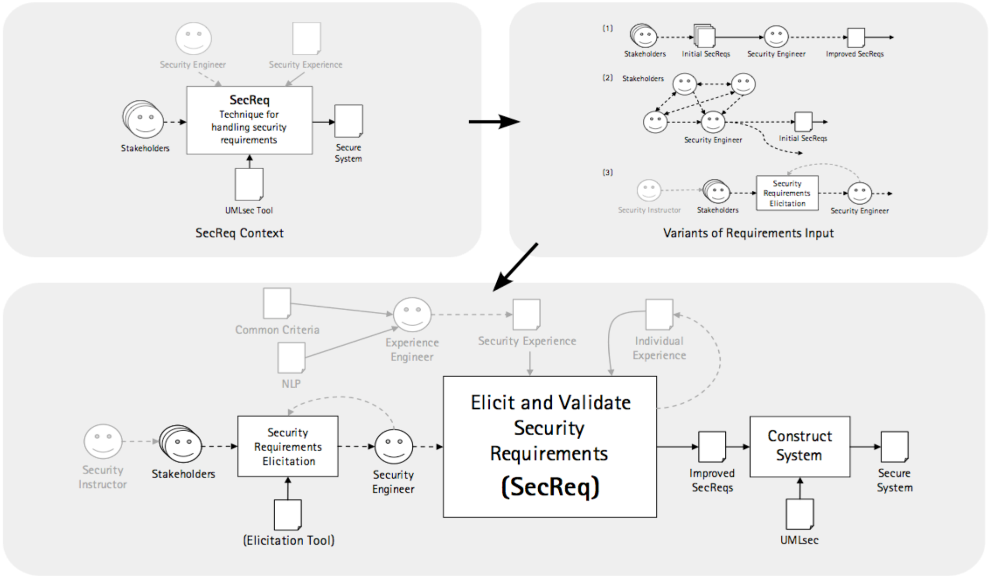
Visualizing Informal Communication

with

FLOW

by *Kurt Schneider*, *Kai Stapel*, and *Eric Knauss*

Security Requirements Example



Information Flow in Related Notations

Table 1. Symbols to Visualize Informa tion Flow Aspects (Syntax)

Concept	FLOW	DFD	UML Activity Diagram	Little-JIL	RCSN
Information store	Person (fluid), Document (solid)	Data store, External	Data store stereotype	Parameters, Agents	Persons
Information flow	Dashed arrows (fluid) Solid arrows (solid)	Data flow arrow	Data/object flow edge	Parameters with control flow	Communication flow
Distinction of solid and fluid	Different symbols (see above)	No	Through Stereotypes	No	Style of arrow (color, etc.)
Explicit experience	Explicit edge color	No	Association stereotypes	Parameter type	No
Challenges when used for information flow modeling	No	Stakeholder as process, data store, or external? Labeling rules violated.	Requires stereotyping for symbols, extended meaning etc. Will look like FLOW when stereotyped.	Very fine-grained symbols, very detailed and abstract models due to its roots as a process programming language	Single req. flowing as observed in reality. Notation not fully defined, many symbols.

Table 2. Comparison of Notations: Purpose and Focus

Diagram type	FLOW	DFD	UML Activity Diagram	Little-JIL	RCSN
Main purpose	Process improvement by considering solid and fluid flows alike	System design	Process design	Process programming	RE awareness
Main object of interest	Information, in particular requirements	Data	Activities	Steps	Social network

SOFTWARE ENGINEERING

Leibniz Universität Hannover
Software Engineering Group
Welfengarten 1, 30167 Hannover, Germany
E-Mail: kurt.schneider@inf.uni-hannover.de
Web: www.se.uni-hannover.de

Leibniz Universität Hannover

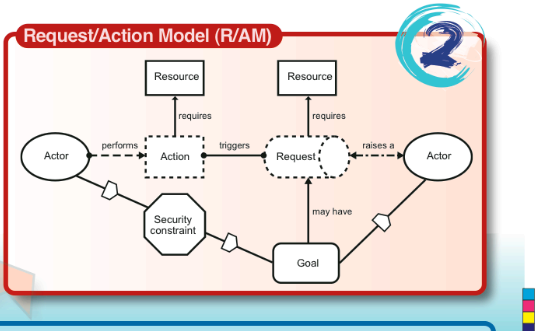
Modelling Trust Requirements by Means of a Visualization Language

Innovative Informatics, School of Computing and Technology,
University of East London, UK
Kamaljit Kaur Bimrah, Haralambos Mouratidis, David Preston
{bimrah, h.mouratidis, d.preston}@uel.ac.uk

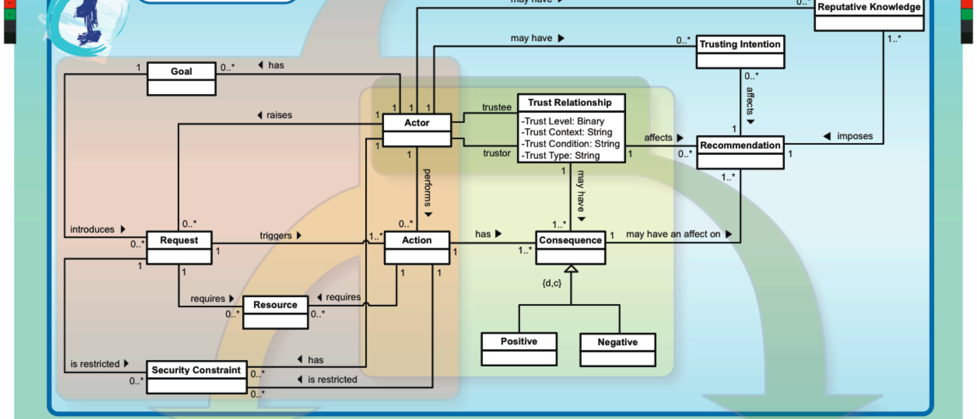
AIM:

To develop a reasoning and modelling framework that will enable information system developers to consider trust and its related concepts collectively during the development of information systems.

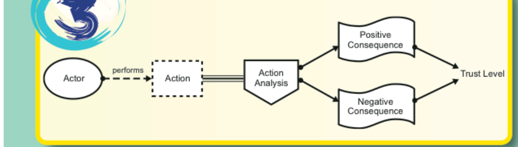
Request/Action Model (R/AM)



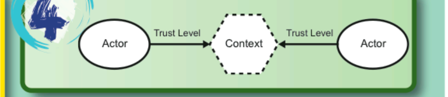
Metamodel



Consequence Model (CONM)



Trust Relationship Model (TRM)

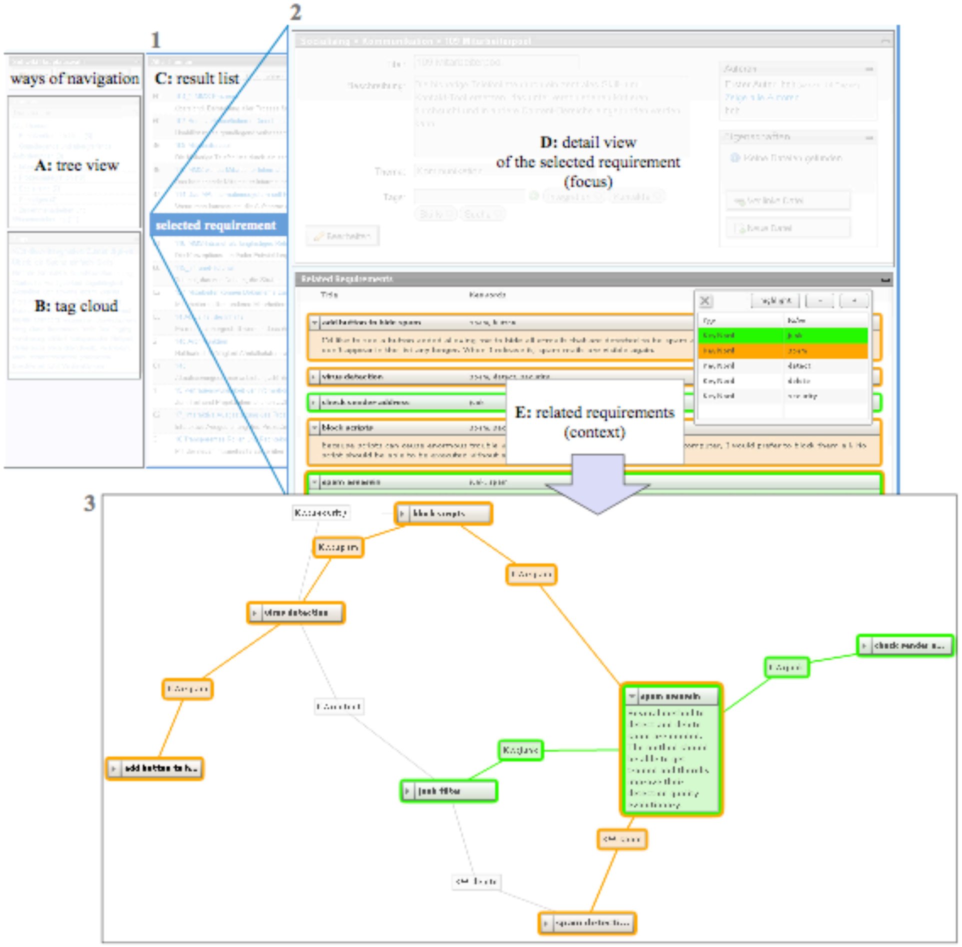


UEL University of East London SCoT School of Computing and Technology

Graph-based Visualization of Requirements Relationships

Philipp Heim, Steffen Lohmann
Interactive Systems and Interaction Design
Department of Computational and Cognitive Science
University of Duisburg-Essen, 47057 Duisburg, Germany
{lohmann, heim}@interactivesystems.info
http://www.interactivesystems.info

Kim Lauenroth
Software Systems Engineering
Institute for Computer Science and Business Information Systems
University of Duisburg-Essen, 45117 Essen, Germany
kim.lauenroth@sse.uni-due.de
http://www.sse.uni-due.de/



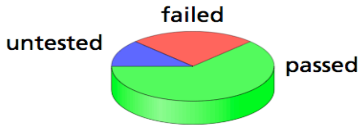
Funded by the German Federal Ministry of Education and Research (BMBF).

On Usability in Requirements Trace Visualizations

Stefan Winkler <stefan.winkler-et@fernuni-hagen.de>

FernUniversität in Hagen

Charts and Tables



	#Req	LOC
Module 1	56	15823
Module 1.1	26	10536
Module 1.2	30	5287
Module 2	24	3578

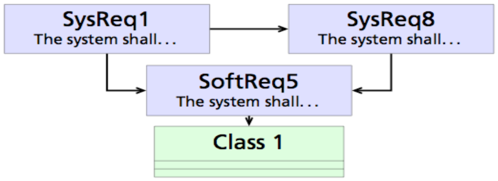
Cross-References

id	content	traceability links
Req 1	The system shall. . .	▶ Req2 ▶ Req3 ▶ Class1
Req 2	The system shall. . .	▶ Req3 ◀ Req1 ◀ Req3
Req 3	The system shall. . .	▶ Req2 ◀ Req1 ◀ Req2

Traceability Matrices

	Req1	Req2	Req3
Class1	■		■
Class2		■	
Class3		■	■
Class4	■		

Trace Networks / Graphs



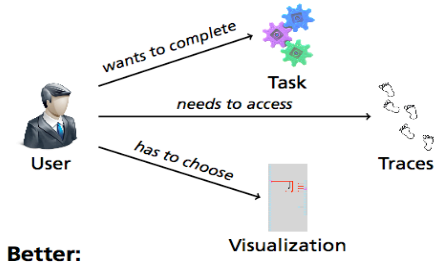
User roles, tasks, and access modes

	user role							access		
	Process Mgr.	Customer	Project Mgr.	Auditor	Req. Engineer	Designer	Maintainer	Report	Search	Browse
Prioritize requirements		×	×		×					×
Estimate change impact		×	×		×	×	×	×		×
Prove adequateness		×	×	×				×		
Validate artifacts			×	×	×	×			×	×
Test the system		×		×				×		
Support audits		×		×					×	
Improve changeability					×	×	×			×
Extract metrics	×		×					×		
Monitor progress			×					×		
Assess the process	×		×					×	×	
Track rationale					×	×	×			×
Understand the system						×	×	×	×	×
Establish accountability			×		×	×	×			×
Document reengineering					×	×				×
Reuse elements	×		×		×	×				×
Extract best practices	×							×	×	

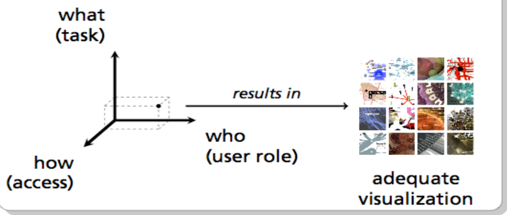
Preliminary data – derived from literature

. . . should drive visualizations

Current situation:



Better:



Making visualizations usable will better expose the benefits of traceability.

Visualizing Use Case Sets as BPMN Processes

SOFTWARE ENGINEERING

Problem Description

Especially in SOA projects but in other projects as well, software has to support business processes. These processes are part of the software requirements as well as the Use Case descriptions. **Both models overlap which wastes time for creating and maintaining the software requirements.**

Solution

Generate BPMN processes from textual Use Cases

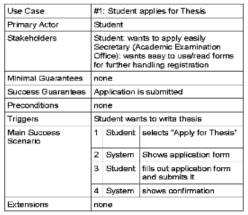
- Generate sequential flows for scenario
- Create decision points for extensions
- Match Pre- and Postconditions of Use Cases
- Join Use Cases

Advantages

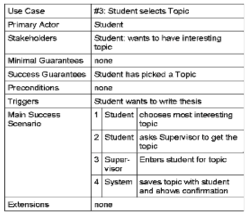
- Useful to generate skeleton business processes
- Helps visualizing and identifying dependencies between Use Cases
- Helps validating dependencies between Use Cases
- Helps validating pre- and postconditions of Use Cases

Open Questions for Future Work

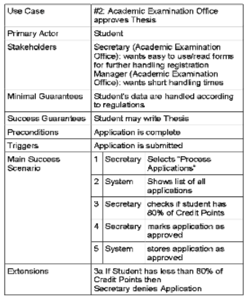
- Non-Literal matching of pre- and postconditions
- Also useful for comparing business processes with given Use Cases?
- Advanced Tool-Support



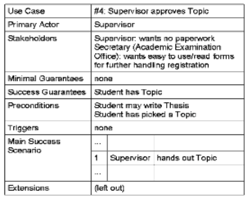
Use Case	#1: Student applies for Thesis		
Primary Actor	Student		
Stakeholders	Student: wants to apply easily Secretary (Academic Examination Office): wants easy to use/read forms for further handling registration		
Minimal Guarantees	none		
Success Guarantees	Application is submitted		
Preconditions	none		
Triggers	Student wants to write thesis		
Main Success Scenario	1	Student	selects "Apply for Thesis"
	2	System	Shows application form
	3	Student	fills out application form and submits it
	4	System	shows confirmation
Extensions	none		



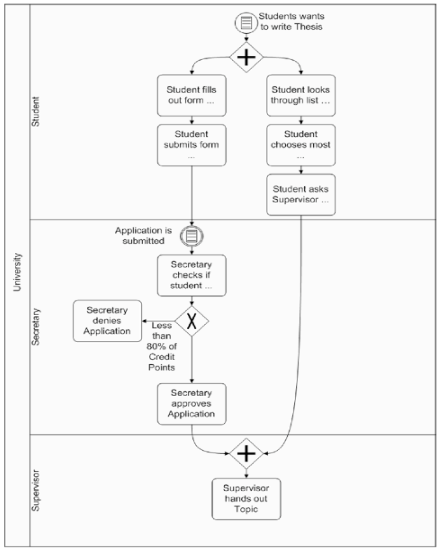
Use Case	#3: Student selects Topic		
Primary Actor	Student		
Stakeholders	Student: wants to have interesting topic		
Minimal Guarantees	none		
Success Guarantees	Student has picked a Topic		
Preconditions	none		
Triggers	Student wants to write thesis		
Main Success Scenario	1	Student	chooses most interesting topic
	2	Student	asks Supervisor to get the topic
	3	Super-visor	Enters student for topic
	4	System	saves topic with student and shows confirmation
Extensions	none		



Use Case	#2: Academic Examination Office approves Thesis		
Primary Actor	Student		
Stakeholders	Secretary (Academic Examination Office): wants easy to use/read forms for further handling registration Manager (Academic Examination Office): wants short handling times		
Minimal Guarantees	Student's data are handled according to regulations.		
Success Guarantees	Student may write Thesis		
Preconditions	Application is complete		
Triggers	Application is submitted		
Main Success Scenario	1	Secretary	Selects "Process Applications"
	2	System	Shows list of all applications
	3	Secretary	checks if student has 80% of Credit Points
	4	Secretary	marks application as approved
	5	System	stores application as approved
Extensions	3a If Student has less than 80% of Credit Points then Secretary denies Application		



Use Case	#4: Supervisor approves Topic		
Primary Actor	Supervisor		
Stakeholders	Supervisor: wants no paperwork Secretary (Academic Examination Office): wants easy to use/read forms for further handling registration		
Minimal Guarantees	none		
Success Guarantees	Student has Topic		
Preconditions	Student may write Thesis Student has picked a Topic		
Triggers	none		
Main Success Scenario	...		
	1	Supervisor	hands out Topic
	...		
Extensions	(left out)		


Students wants to write Thesis
Student
Student fills out form ...
Student submits form ...
Student looks through list ...
Student chooses most ...
Student asks Supervisor ...
University
Secretary
Application is submitted
Secretary checks if student ...
X
Less than 80% of Credit Points
Secretary denies Application
Secretary approves Application
Supervisor
Supervisor hands out Topic

Leibniz Universität Hannover

HPI Hasso Plattner Institut

Daniel Lübke, Kurt Schneider and Matthias Weidlich
FG Software Engineering, Leibniz Universität Hannover
Hasso Plattner Institute, Potsdam
daniel.luebke@inf.uni-hannover.de

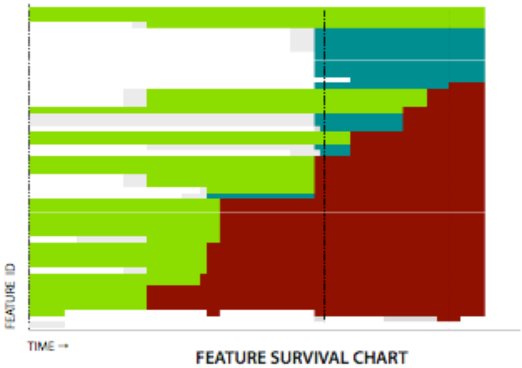
Visualization of Feature Survival in Platform-Based Embedded Systems Development for Improved Understanding of Scope Dynamics

by Krzysztof Wnuk, Björn Regnell, Lund University and Lena Karlsson, DNV Sweden

Visualization can improve real industrial projects.

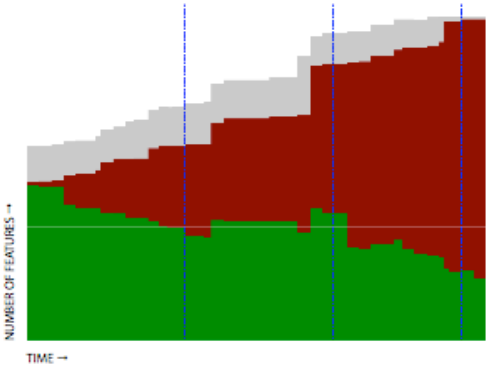
This method for visualizing the scoping process in platform-based development of embedded systems shows the decision process of including or excluding features that are candidates for the next release.

Charts are evaluated in a large-size embedded system platform project and indicates that the visualization of feature survival and scope dynamics can improve the understanding of the decision process of platform scoping in real industrial projects.



FEATURE SURVIVAL CHART

The red lines show out-scoped features. The green lines show features in scope (light green for primary flow features and dark green for secondary flow features). The survivors are placed at the top, as the graph is sorted on duration in scope from last baseline.



FEATURE GROWTH CHART

The green area represents the number of in-scoped features, red the number of out-scoped and gray the number of undecided.

Future work includes improving user interaction.

– PANTHÉON - SORBONNE –
UNIVERSITÉ PARIS 1
Centre de Recherche
en Informatique

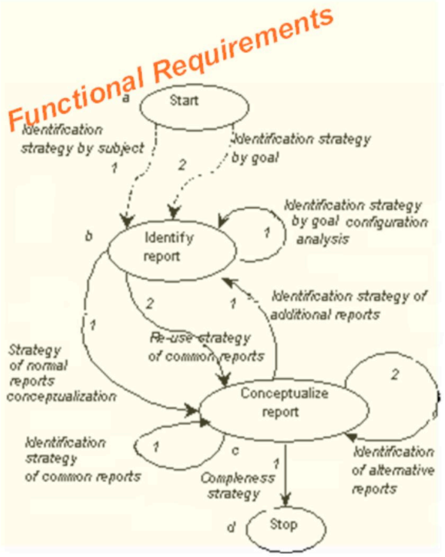
Visualizing the Impact of Non-Functional Requirements on Variants – A Case Study

Clotilde Rohleder
Clotilde.rohleder@siemens.com

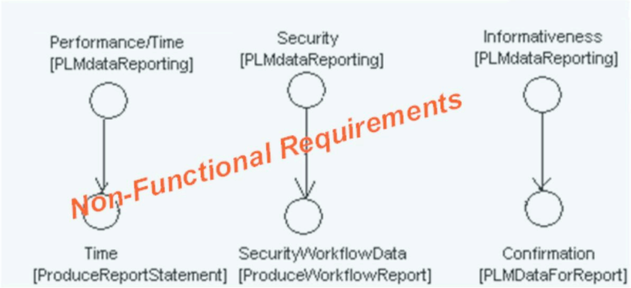
SIEMENS

TEAMCENTER

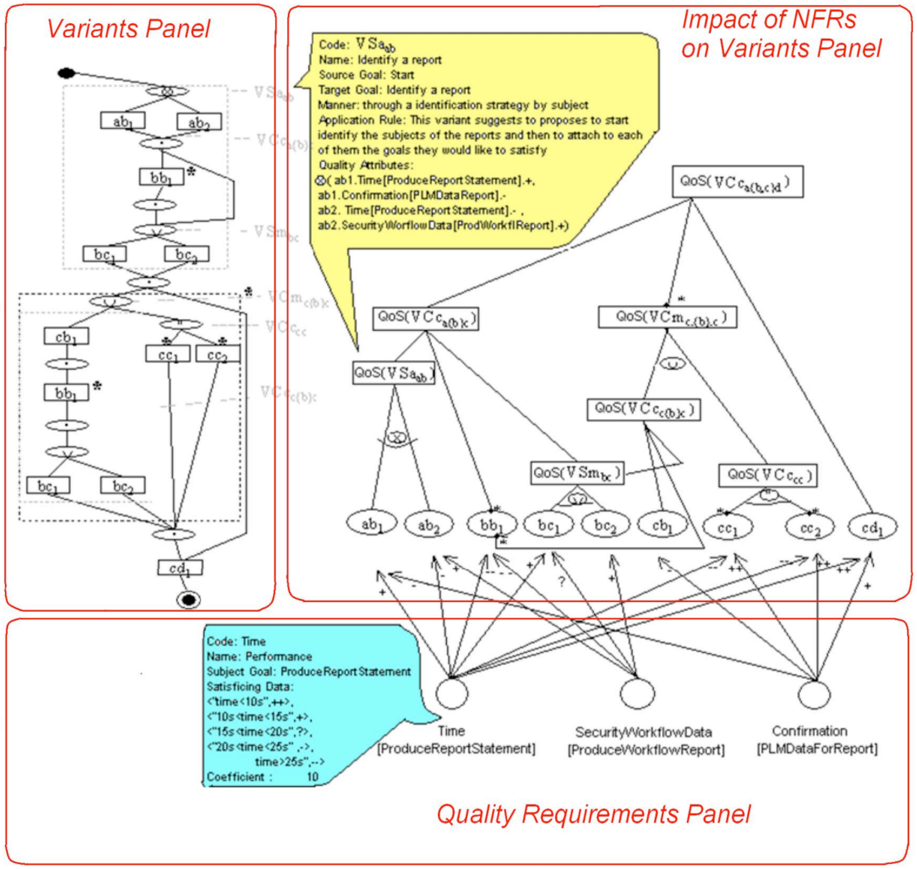
1


Functional Requirements
a
Start
Identification strategy by subject
1
2
Identification strategy by goal
b
Identify report
1
Identification strategy by goal configuration analysis
2
1
Identification strategy of additional reports
Re-use strategy of common reports
1
Strategy of normal reports conceptualization
Conceptualize report
2
Identification strategy of common reports
1
c
1
Completeness strategy
Identification of alternative reports
d
Stop

2



3



Code: VSa_ab
Name: Identify a report
Source Goal: Start
Target Goal: Identify a report
Manner: through a identification strategy by subject
Application Rule: This variant suggests to proposes to start identify the subjects of the reports and then to attach to each of them the goals they would like to satisfy
Quality Attributes:
⊗(ab1.Time[ProduceReportStatement].+,
ab1.Confirmation[PLMDataReport].-
ab2. Time[ProduceReportStatement].- ,
ab2.SecurityWorflowData[ProdWorkflReport].+)

Code: Time
Name: Performance
Subject Goal: ProduceReportStatement
Satisficing Data:
<"time<10s",++>,
<"10s<time<15s",+>,
<"15s<time<20s",?>,
<"20s<time<25s" ,->,
time>25s",-->
Coefficient : 10

Visualizing Rationale

Ian Alexander

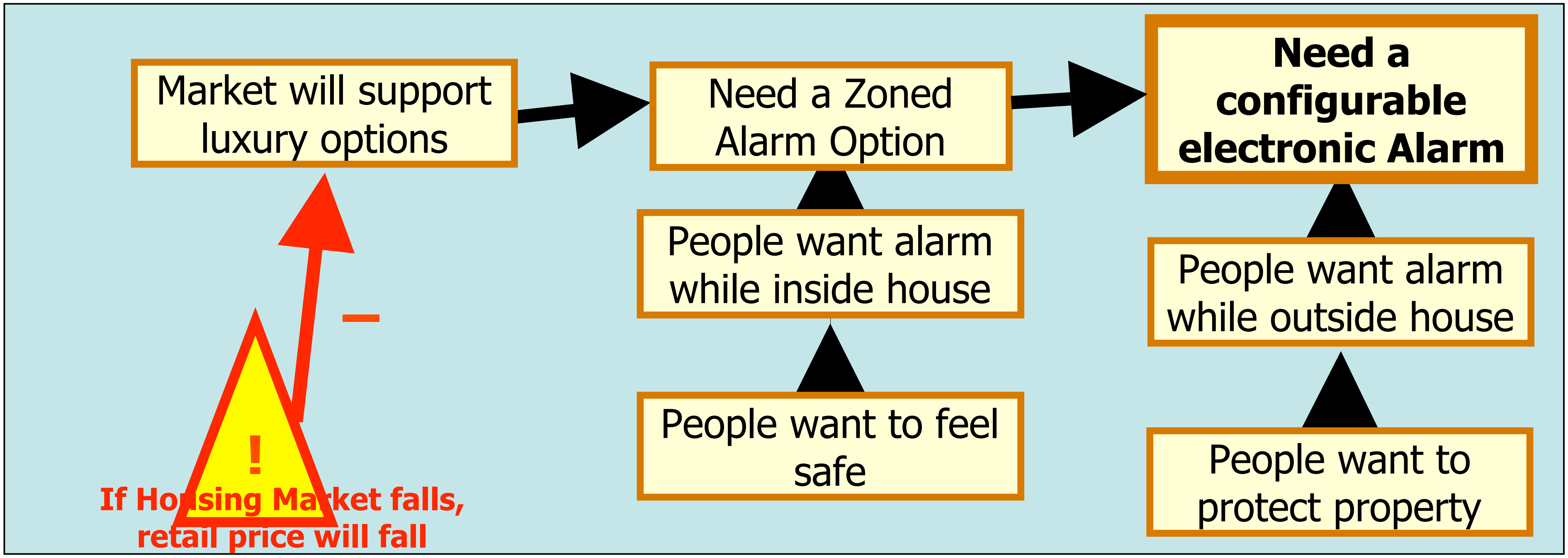
Plato
(c. 424-348 B.C.)

Formal Deduction

Stephen Toulmin

Practical Reasoning

Reasoning as Chains of Assumptions



Market will support luxury options

Need a Zoned Alarm Option

Need a configurable electronic Alarm

People want alarm while inside house

People want to feel safe

People want alarm while outside house

People want to protect property

–

!

If Housing Market falls, retail price will fall

Signposts, Assumptions

James A. Dewar
RAND

Tacit Knowledge

Michael Polanyi
(1891-1976)