An NFR Pattern Approach to Dealing with Non-Functional Requirements

Presenter: Sam Supakkul

Outline

- Motivation
- The Approach
  - NFR Patterns
  - Pattern Organization
  - Pattern Reuse
  - Tool Support
- Case Study
- Conclusion

Is it beautiful to you?

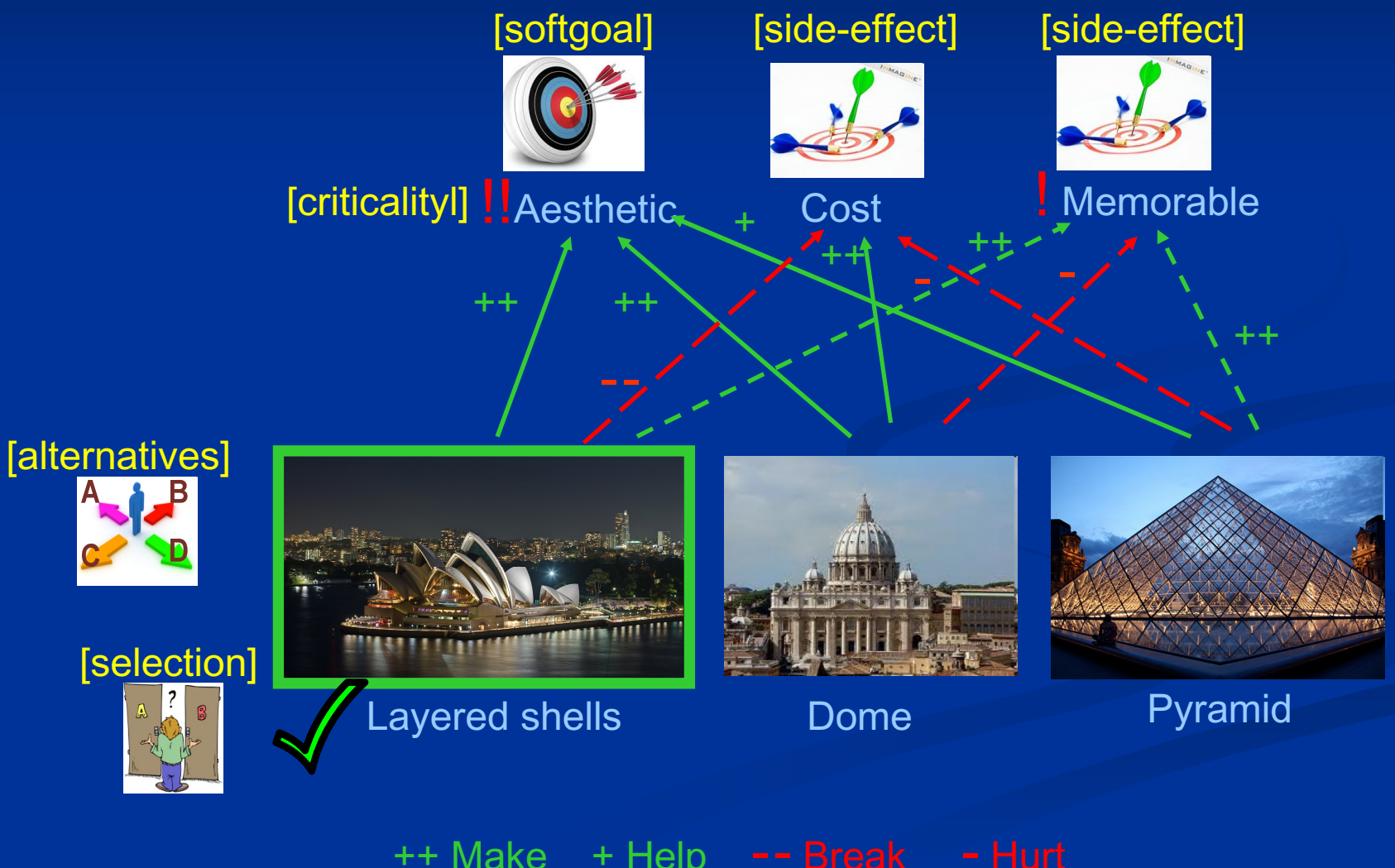
Authors:

Sam Supakkul
Tom Hill
Lawrence Chung
The Univ. of Texas at Dallas

Thein Than Tun
The Open University, UK

Julio CSP Leite
PUC-Rio, Brazil

Dealing with NFRs involves many concepts and activities



Some aspects of NFRs are achieved by mitigating known problems

!! Security of credit card info

\- / --

Break-in wireless network
Masquerade user login
Steal credit card info

! Trustworthiness

Cost

\- / --

++/+/-/--

++/+/-/--

Password encryption
Biometric authentication
2-factor authentication

++ Make + Help -- Break - Hurt

Having insufficient knowledge of NFRs can lead to dire consequences

TJX unable to prevent the hacker

1. Break-in wireless network
2. Masquerade user login
3. Steal credit card info

(2nd) Biggest credit card theft

- 45.7M credit cards stolen
- $20M in fraudulent transactions

TJX used security measures

- ID/password authentication
- Data encryption

But TJX did not know enough

- Potential security problems
- Applicable mitigations
- Proper tradeoff among NFRs

Having sufficient knowledge of NFRs is difficult because NFR knowledge is

Difficult to capture

- Problems, solutions, domain
- Complete, correct
- Conceptual modeling and reasoning

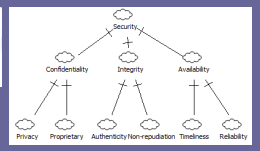
Difficult to organize

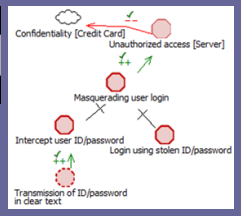
- Cataloging knowledge
- Relating similar knowledge
  - General – Specific
  - Class – Instance
- Combining knowledge

Difficult to reuse

- Choosing appropriate knowledge
- (Re-)creating visual models

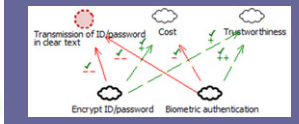
This talk presents a pattern-based approach to capturing, organizing, and reusing NFR knowledge



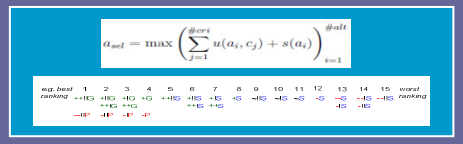


Objective pattern

Problem pattern

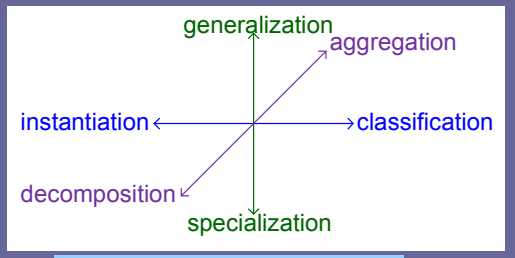


Alternatives pattern

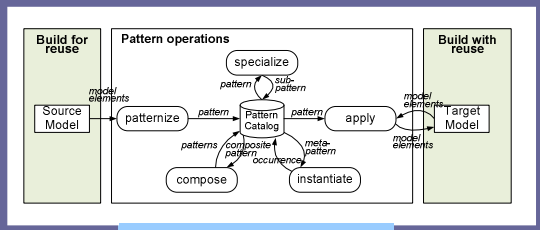


Selection pattern

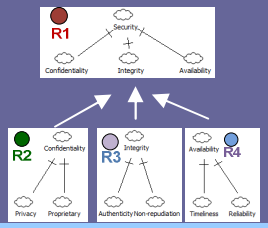
Capturing



Organizing

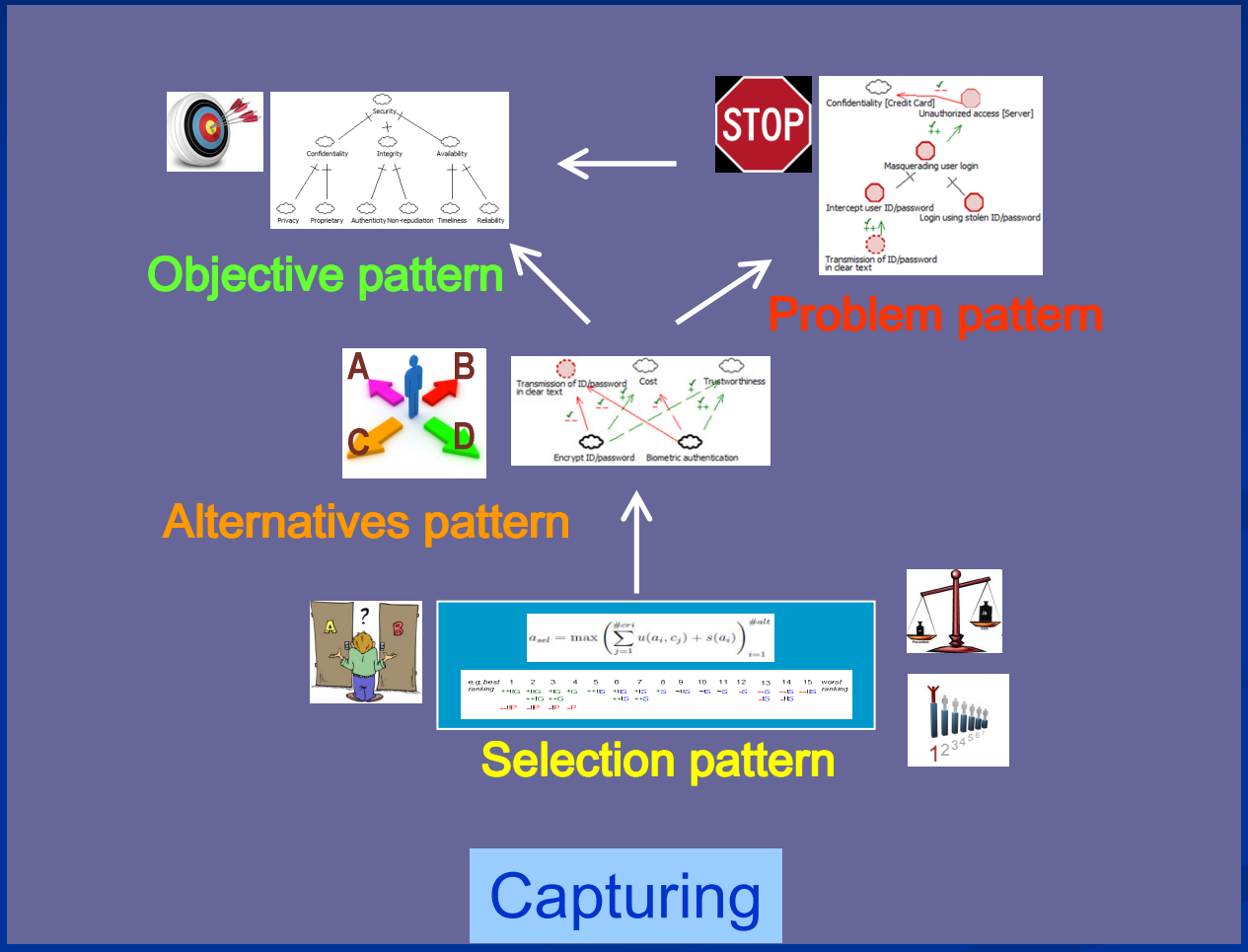


Reusing



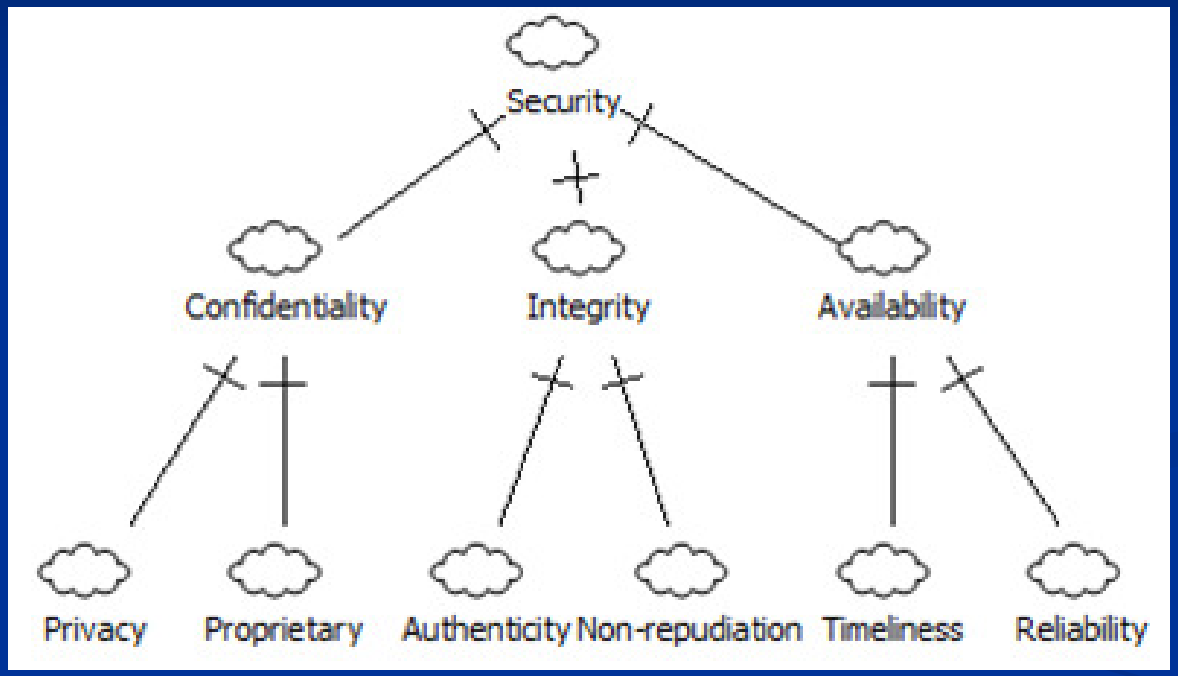
Tool support

4 kinds of NFR patterns for capturing different kinds of NFR knowledge




An objective pattern captures a definition of an NFR as a softgoal (and sub-goals) to be achieved

An objective pattern

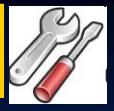


Knowledge

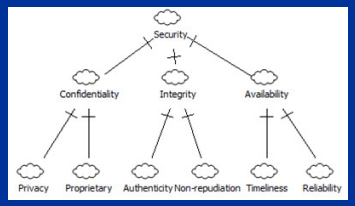
Name: FISMA Security Objectives

Credential
Sources: US FISMA Act of 2002
Authors: Sam Supakkul
Endorsements:
Known uses: US government agencies

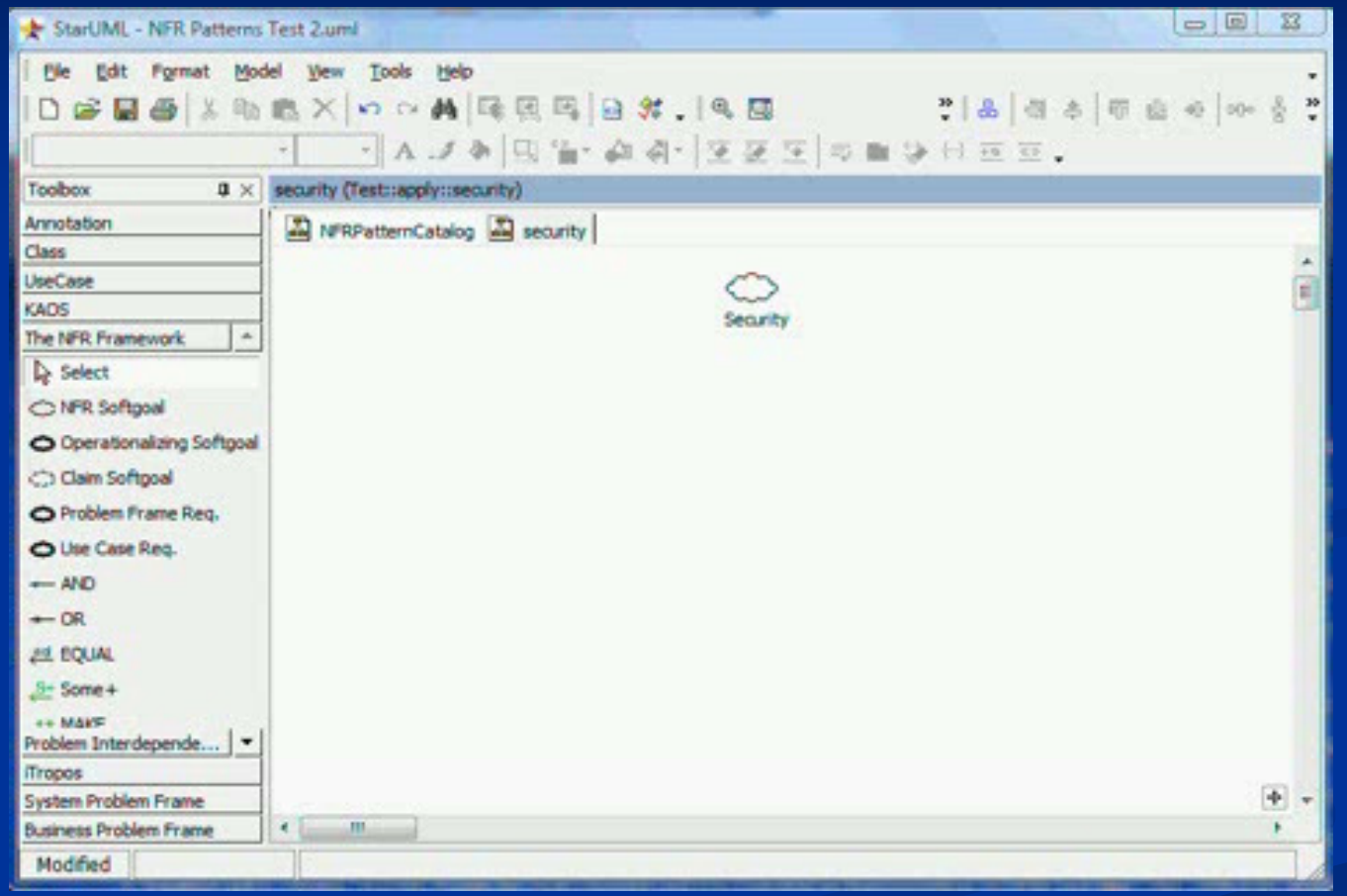
Applicability (5W2H questions)
Domain (Who): Government
Topic (What): Information, data
Type (Why): Security
Phase (When): Requirements
Artifact (Where): World [per the WRSPM ref. model]
Application (How): Automated
Implication (How much): Regulation


Demo video: applying an objective pattern



In the catalog



During the requirements engineering of a project

Tools used

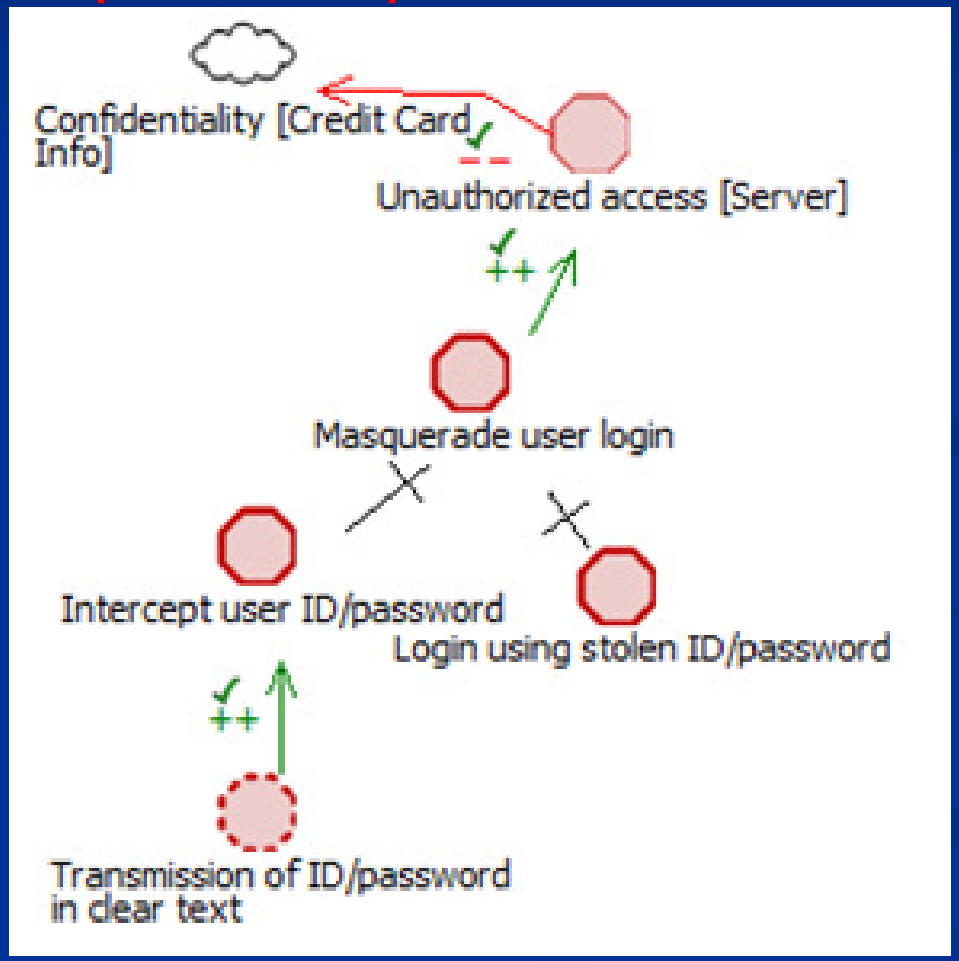
The NFR Pattern Assistant (utdallas.edu/~supakkul/tools/NFRPassist)

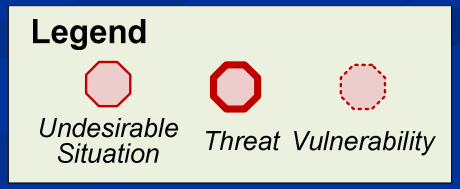
The RE-Tools (utdallas.edu/~supakkul/tools/RE-Tools)



A problem pattern captures soft-problems or obstacles to achieving an NFR softgoal

A problem pattern

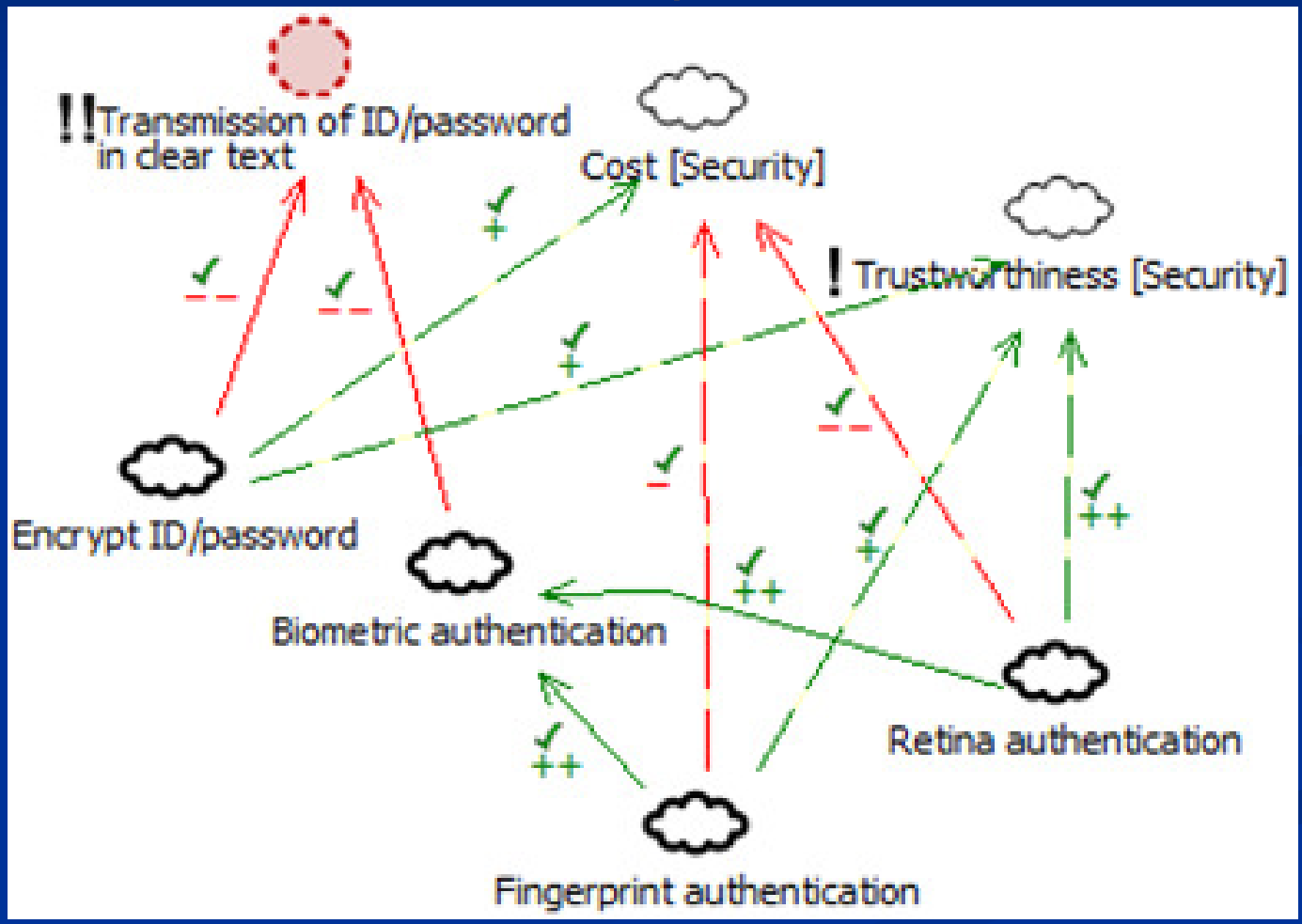






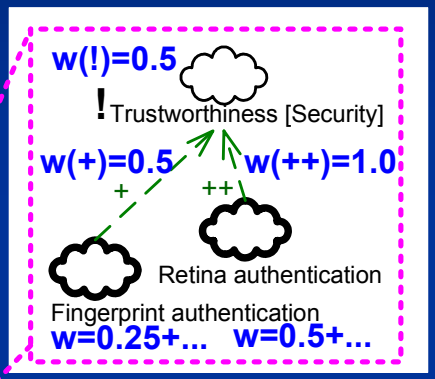
An alternatives pattern captures alternative means or alternative solutions with side-effect information

An alternative-solutions pattern



A selection pattern captures an application independent selection scheme

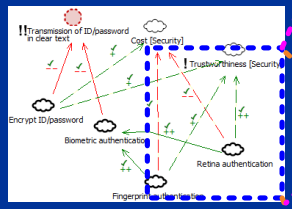
Weight-based quantitative selection



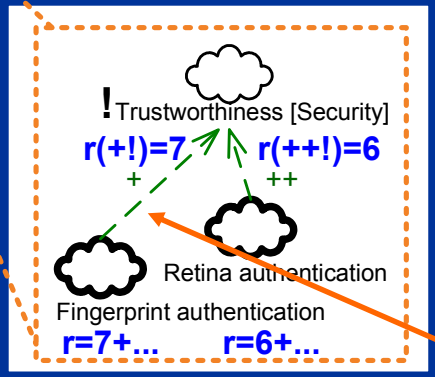
Weight-based
Selection = Highest cumulative **weight**

weight(Fingerprint) = w(!Trust.) x w(+) + …
= 0.25 + …

Widely used, but subjective



alternatives

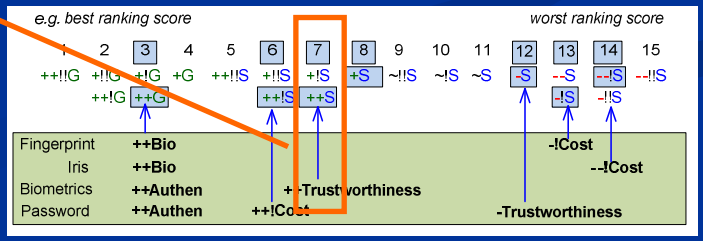


Rank-based qualitative selection

Rank-based
Selection = Best cumulative **ranking**

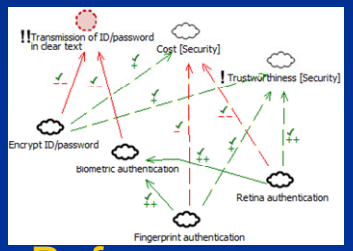
rank(Fingerprint) = r(+!Trust.) + …
= 7 + …

Less subjective, but need a ranking scale

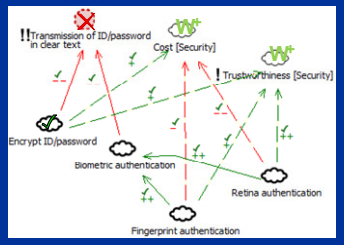


Demo video: applying a weight-based selection pattern

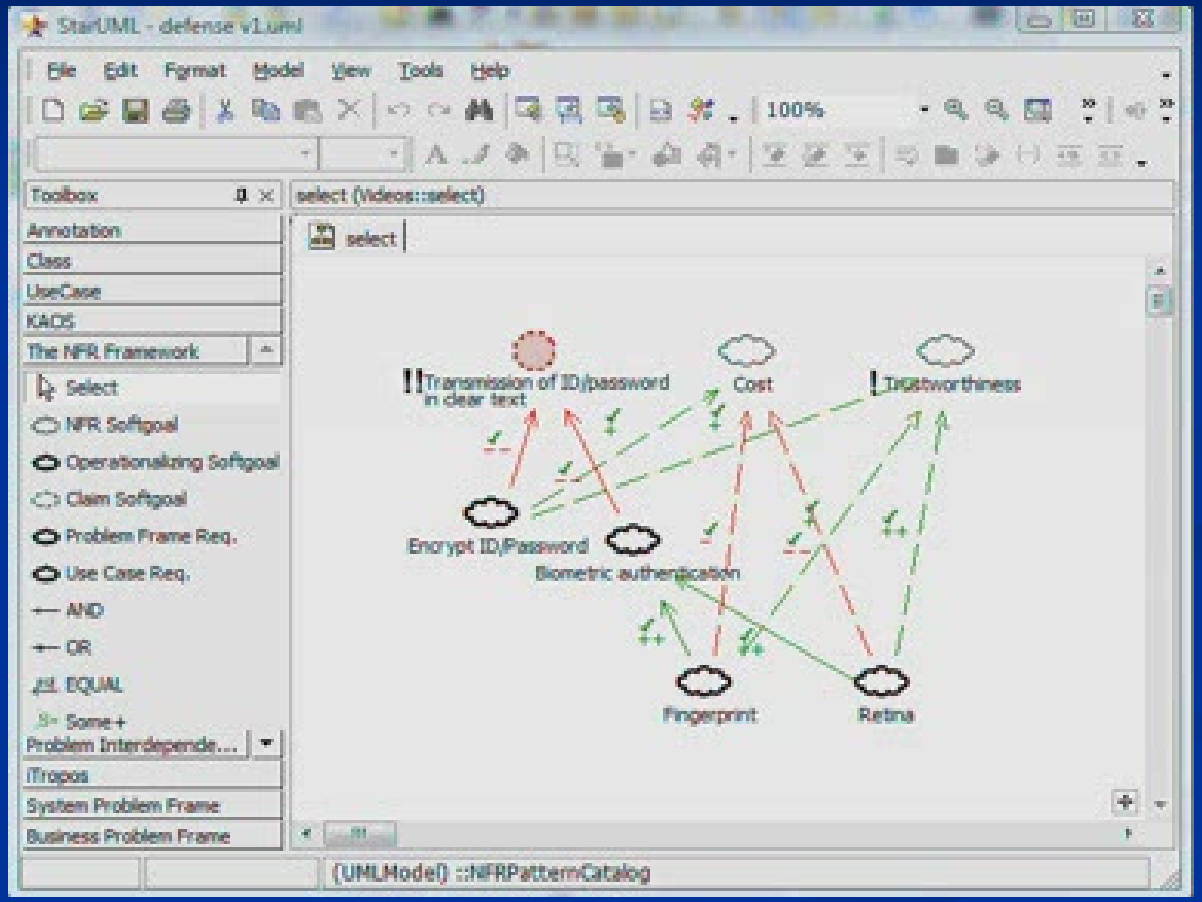




Before



After

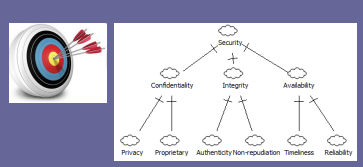


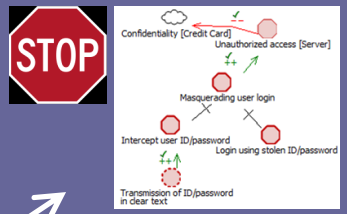
Tools used

The NFR Pattern Assistant (utdallas.edu/~supakkul/tools/NFRPassist)

The RE-Tools (utdallas.edu/~supakkul/tools/RE-Tools)

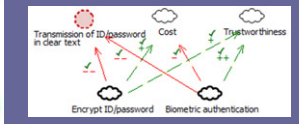
Patterns may be organized along the generalization, aggregation, and classification dim.



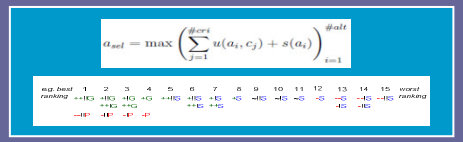


Objective pattern

Problem pattern

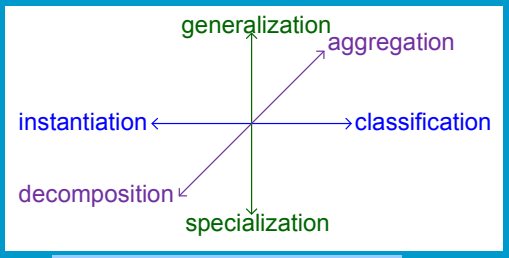


Alternatives pattern

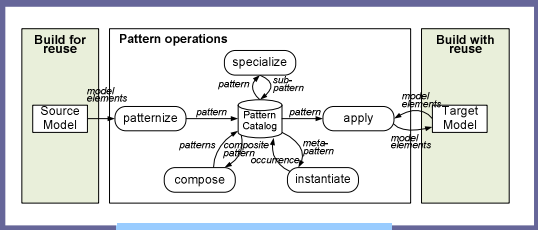


Selection pattern

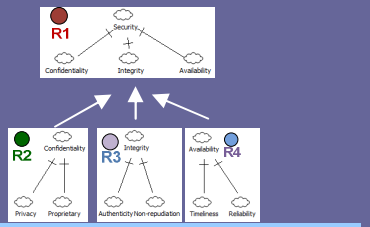
Capturing



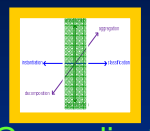
Organizing



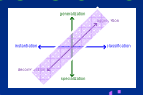
Reusing



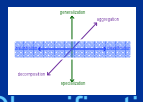
Tool support



Generalization

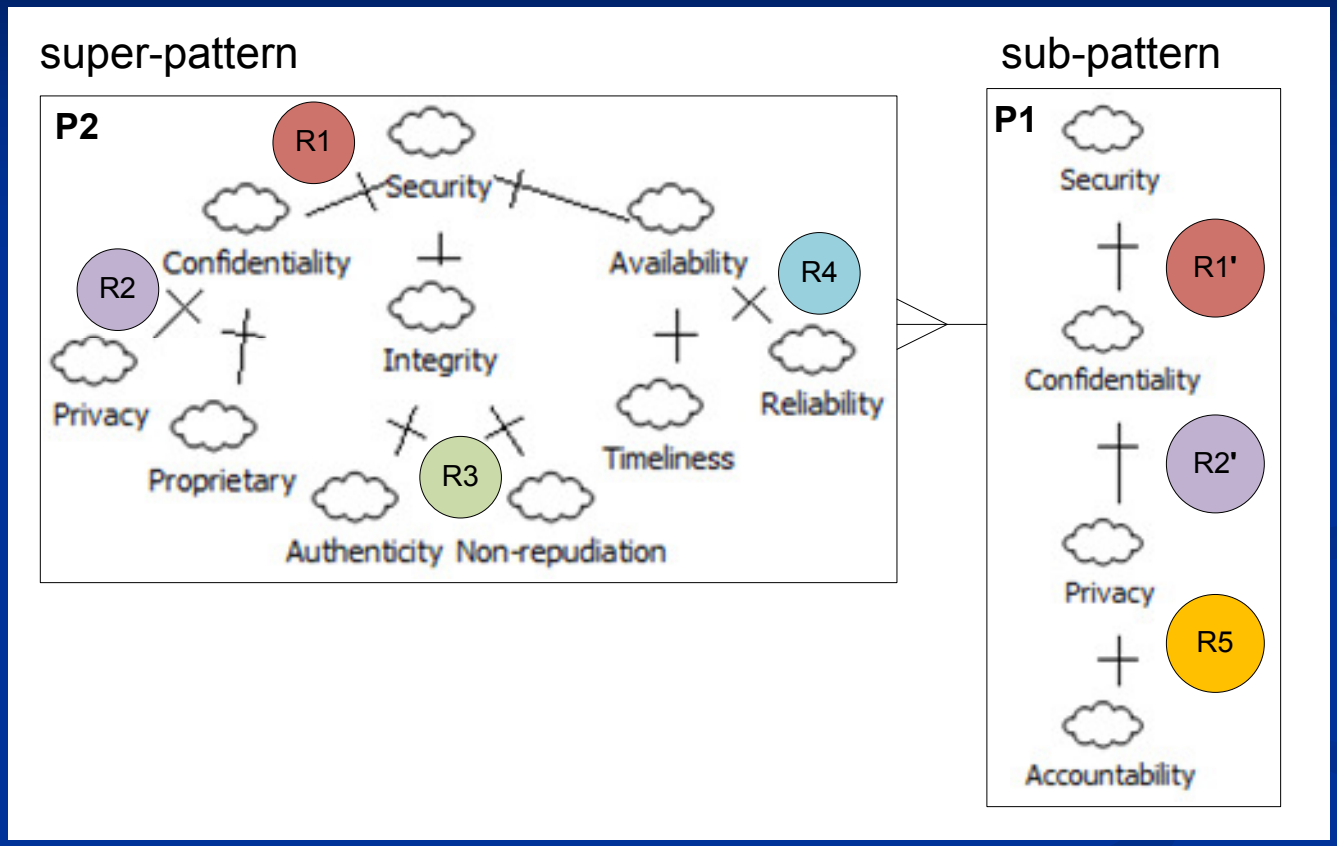


Aggregation



Classification

A specialized pattern captures more specific knowledge than that of the generalized pattern



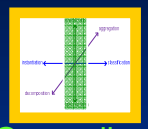
[US FISMA Law]

[Payment Card Industry]

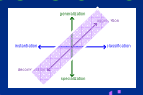
P1 specializes P2

$$P_1 \prec P_2$$

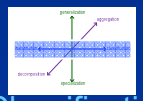
The specialized pattern is more specific in breadth or in depth



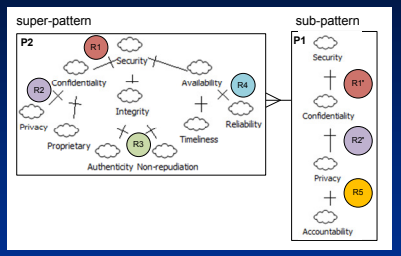
Generalization



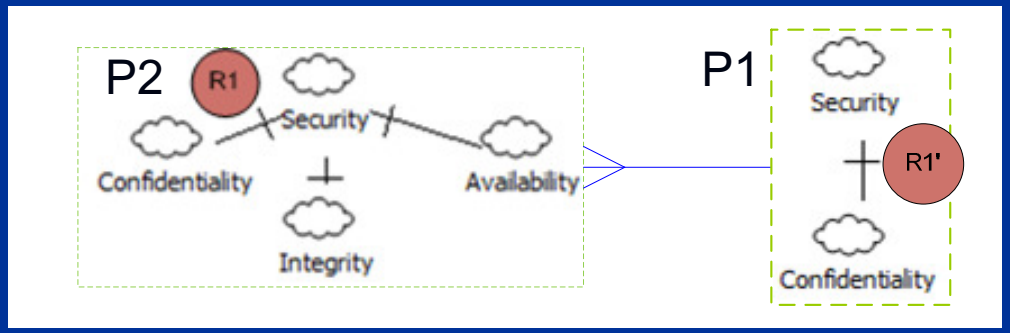
Aggregation



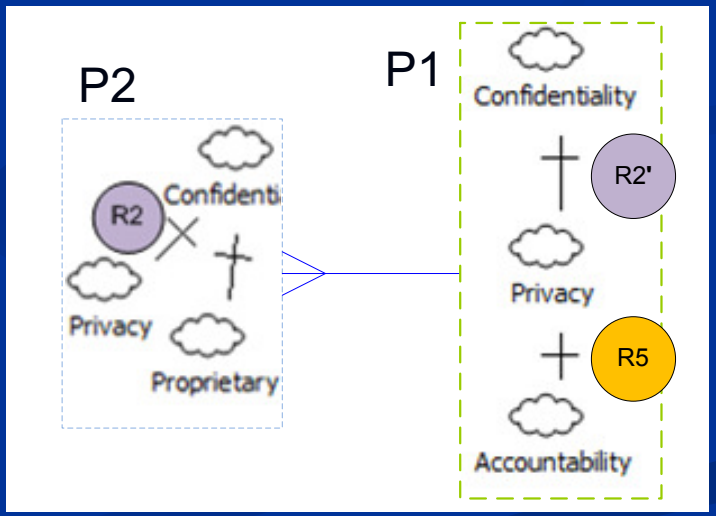
Classification



More specific in breadth



More specific in depth

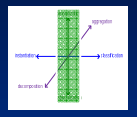


Payment Card Industry (PCI):
Security = Confidentiality

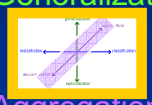
US Law:
Security = Confidentiality, Integrity, Avail.

PCI concerned with
Accountability beyond Privacy

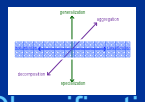
A composite pattern assembles smaller patterns to capture a larger chunk of related knowledge



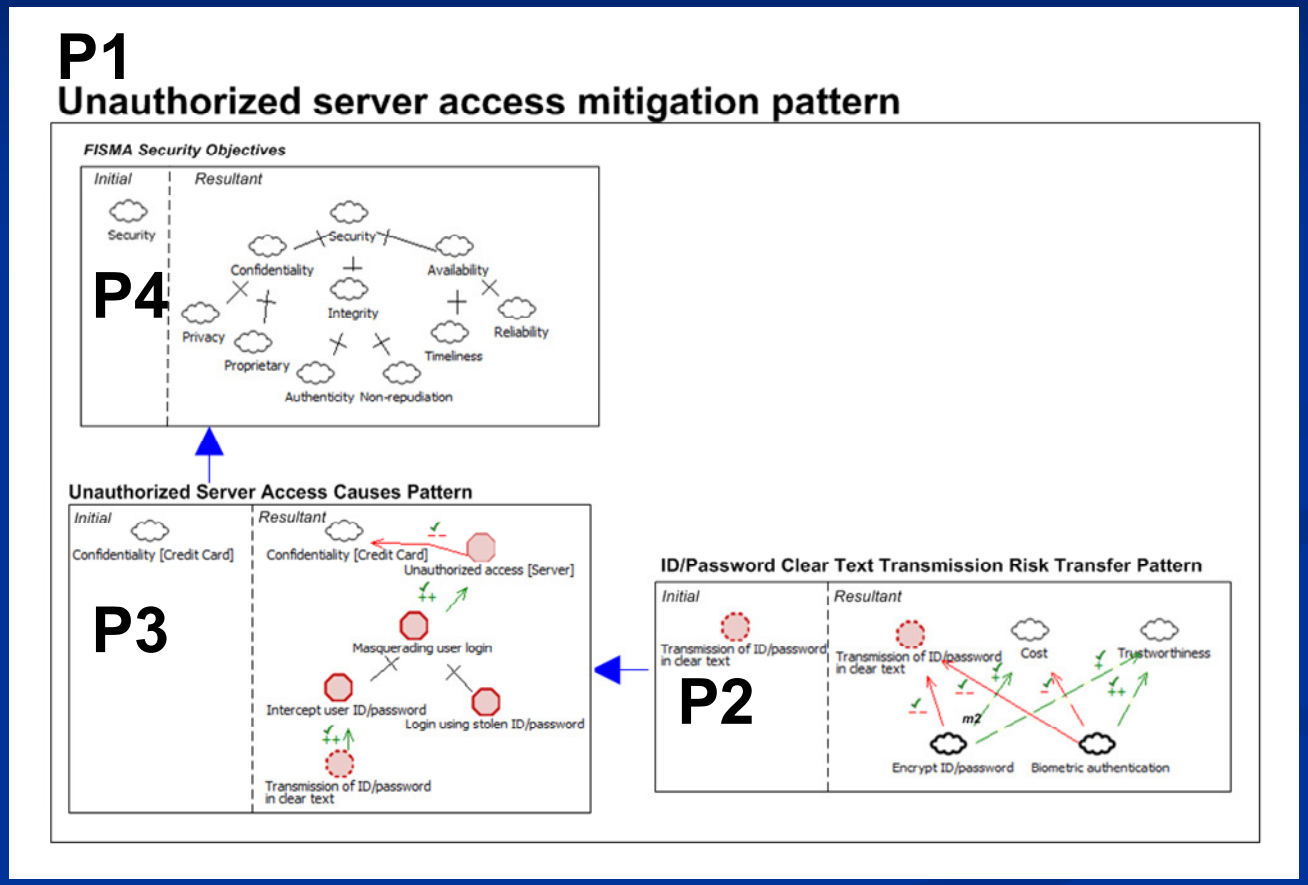
Generalization



Aggregation

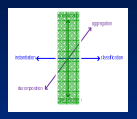


Classification

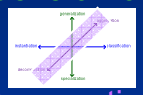


P1 combines P2,P3,P4 where P2 succeeds P3 and P3 succeeds P4

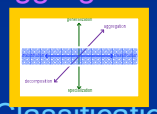
$$P_1 = P_2 \oplus P_3 \oplus P_4$$
$$\text{where } P2 \nearrow P_3 \text{ and } P3 \nearrow P_4$$



Generalization

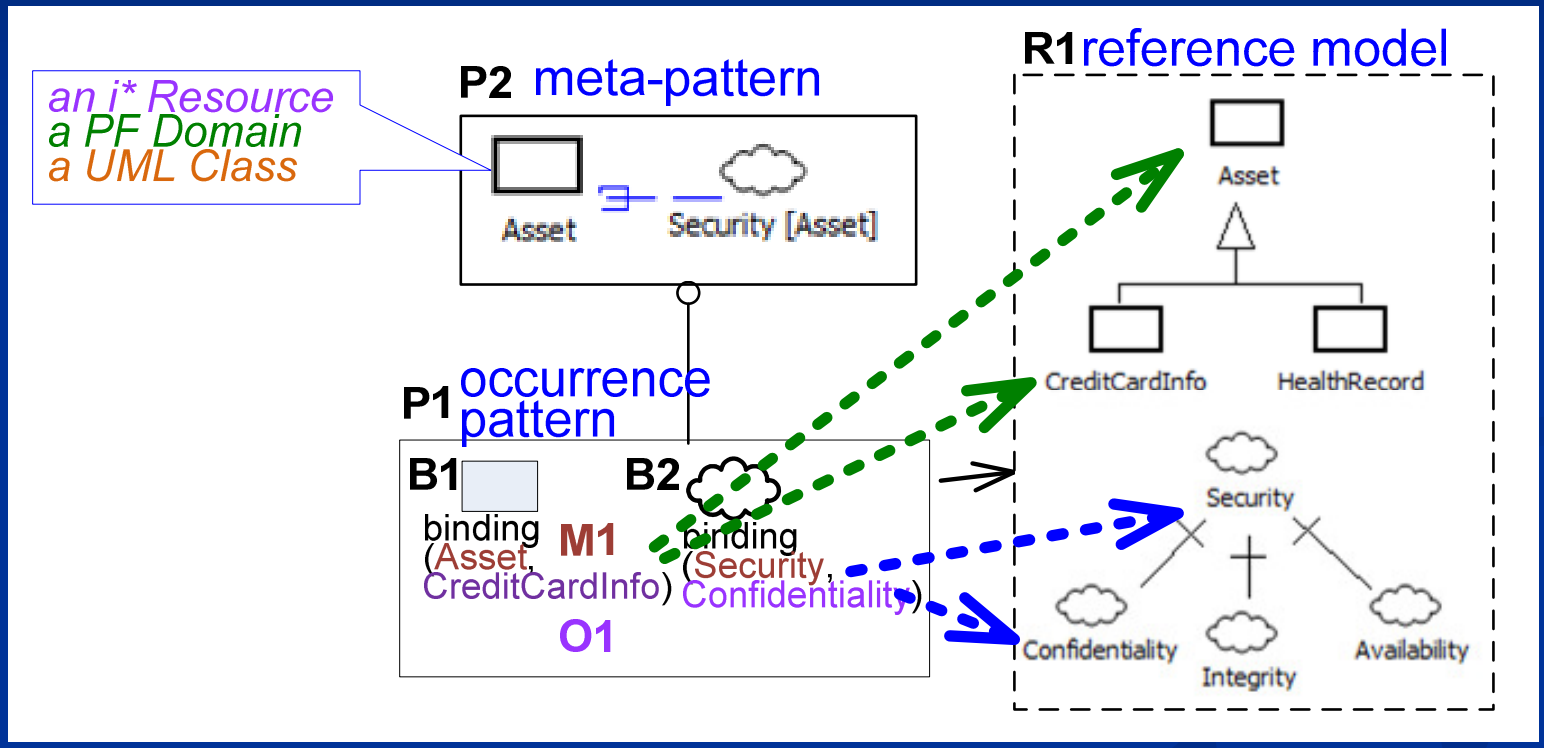


Aggregation



Classification

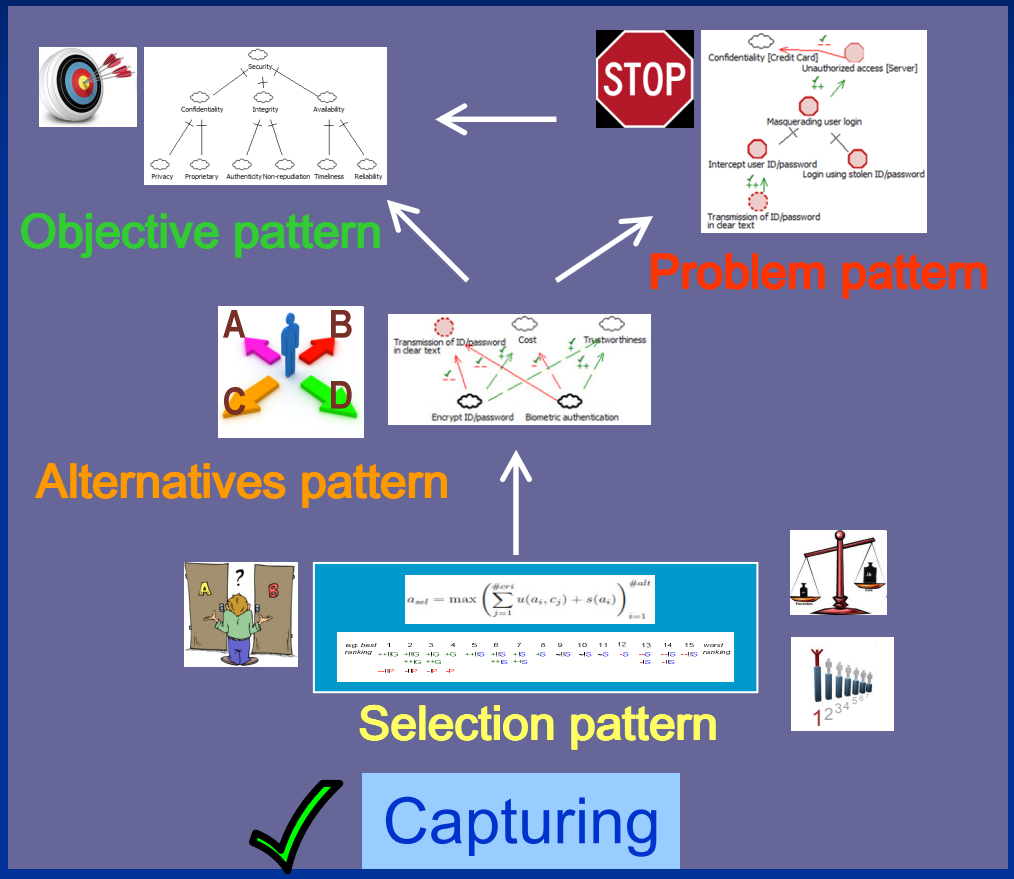
A pattern can be used as a template to instantiate occurrence patterns

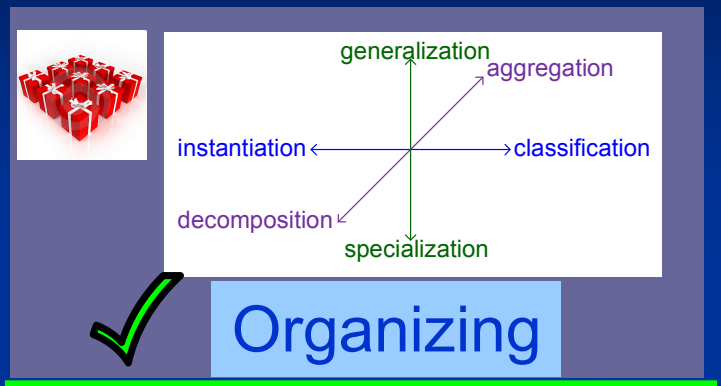


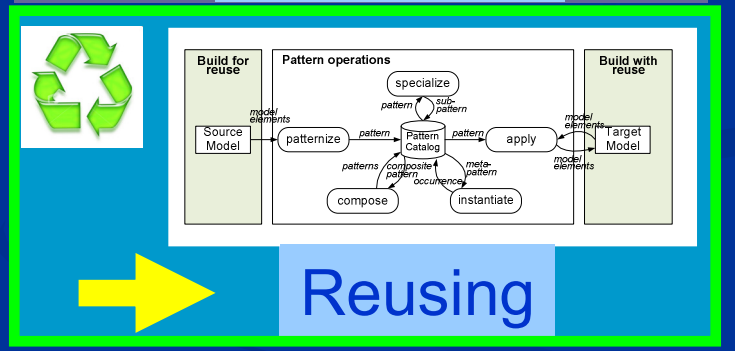
B1 is a binding specification
O1 is a specialization of M1 or
O1 is sub-goal of M1
w.r.t. reference model R1

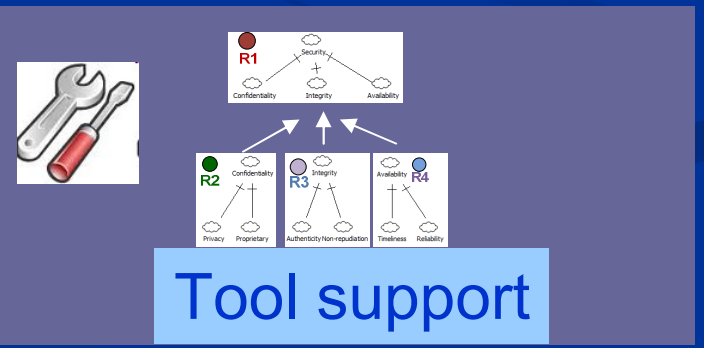
$$P_1 \hookrightarrow P_2$$
where $P_1 = \{B_1, B_2\} \diamond R1$
and $B_1 =< M_1, O_1 >$ and $(O_1 \triangleright M_1 \wr R1$ or $O_1 \angle M_1 \wr R1)$
and $B_2 = ...$

Dealing with NFR knowledge is defined by 5 operations



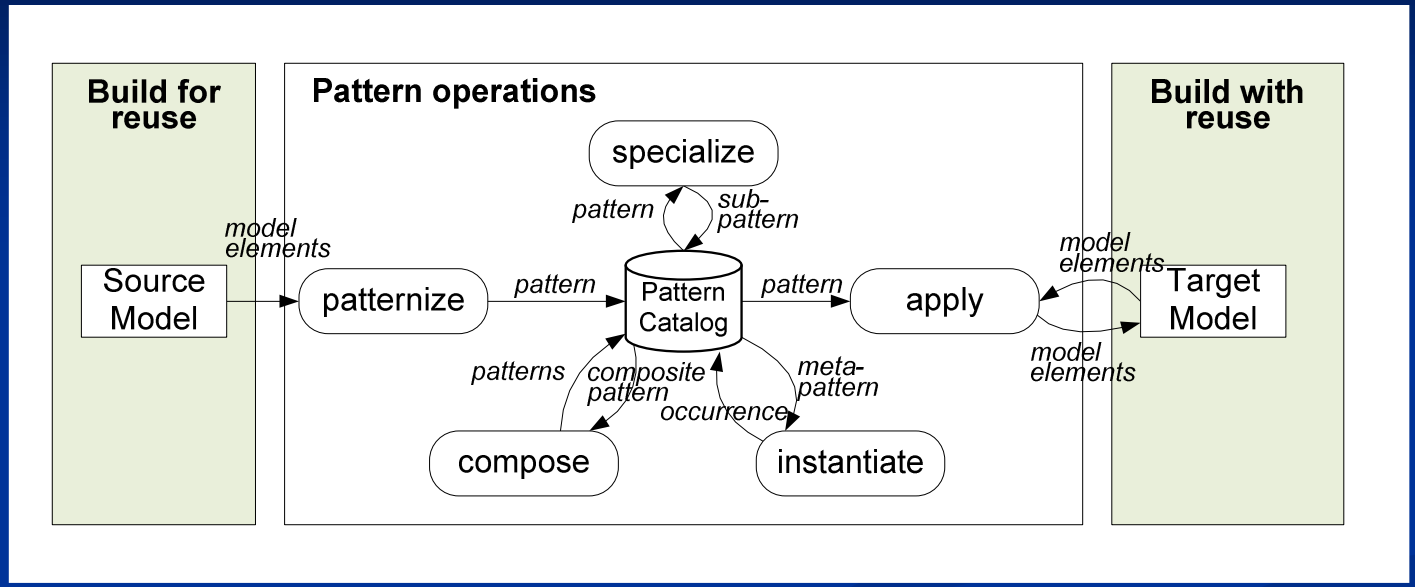






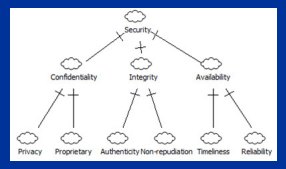
An action-oriented perspective

Pattern operations

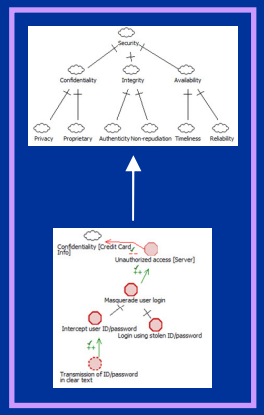


Example

Source Model



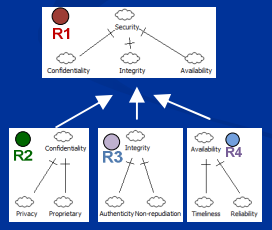
patternize



compose

Credentials
- Authors
- Sources
- Endorsements
- Known Uses

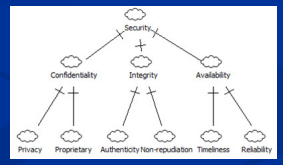
Applicability
- Who
- What
- Why
- When
- Where
- How
- How much



Refinement Rules

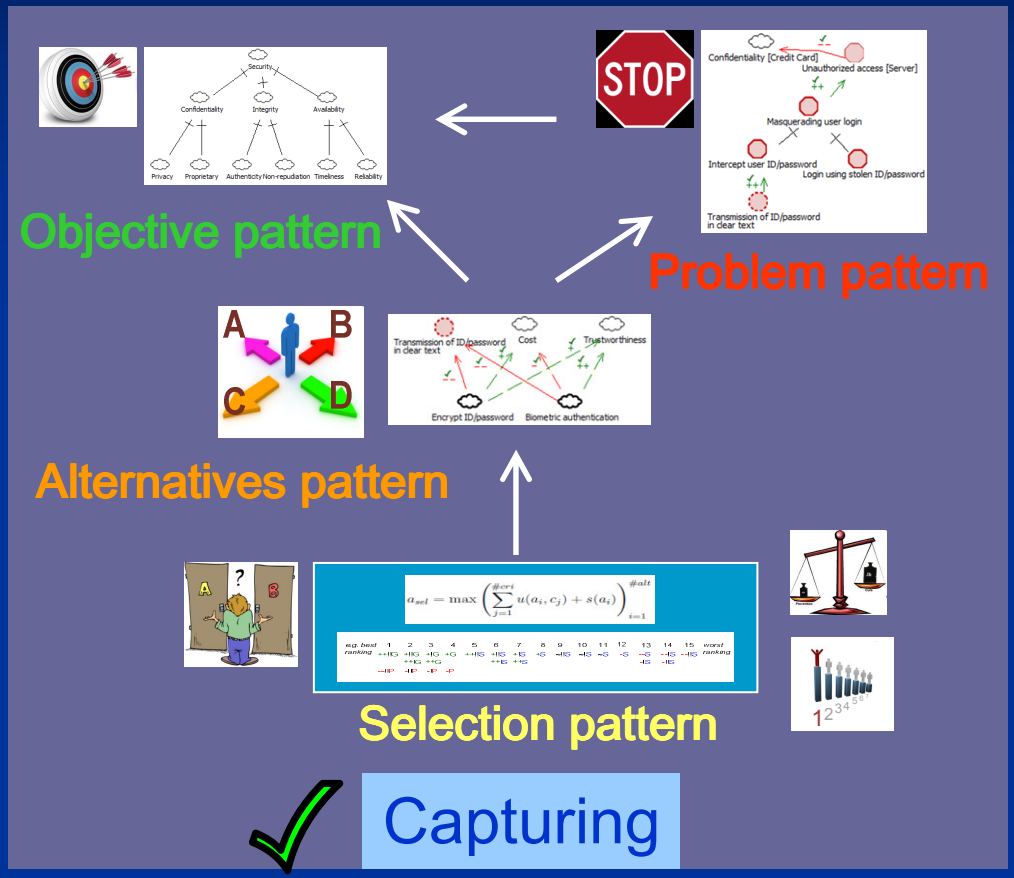
apply

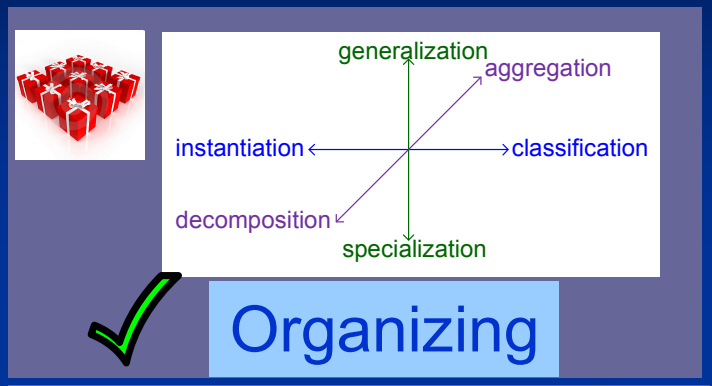
Target Model

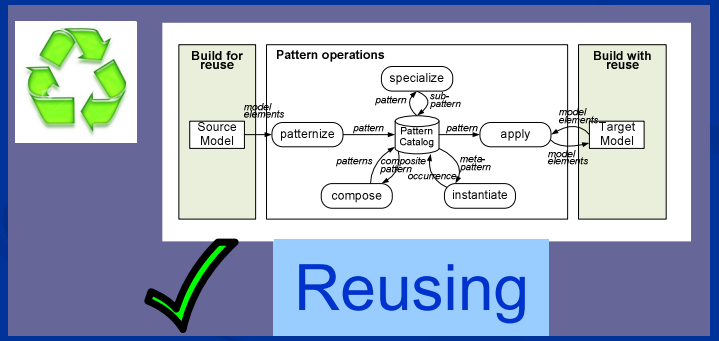


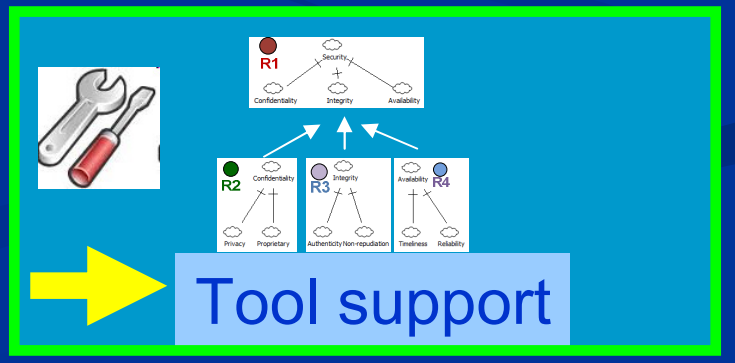
We define 25 refinement rules for tool support

2 for Objective, 8 for Problem, 10 for Alternatives, 5 for Selection

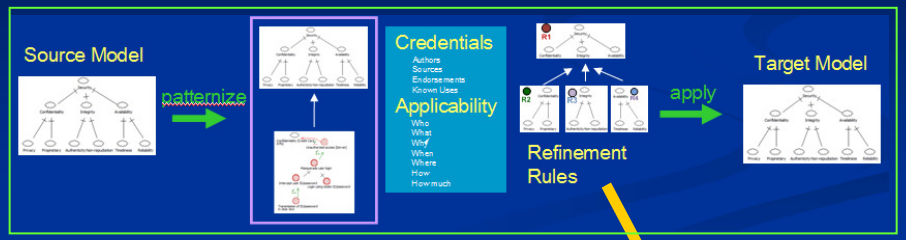


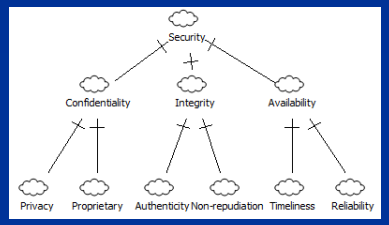


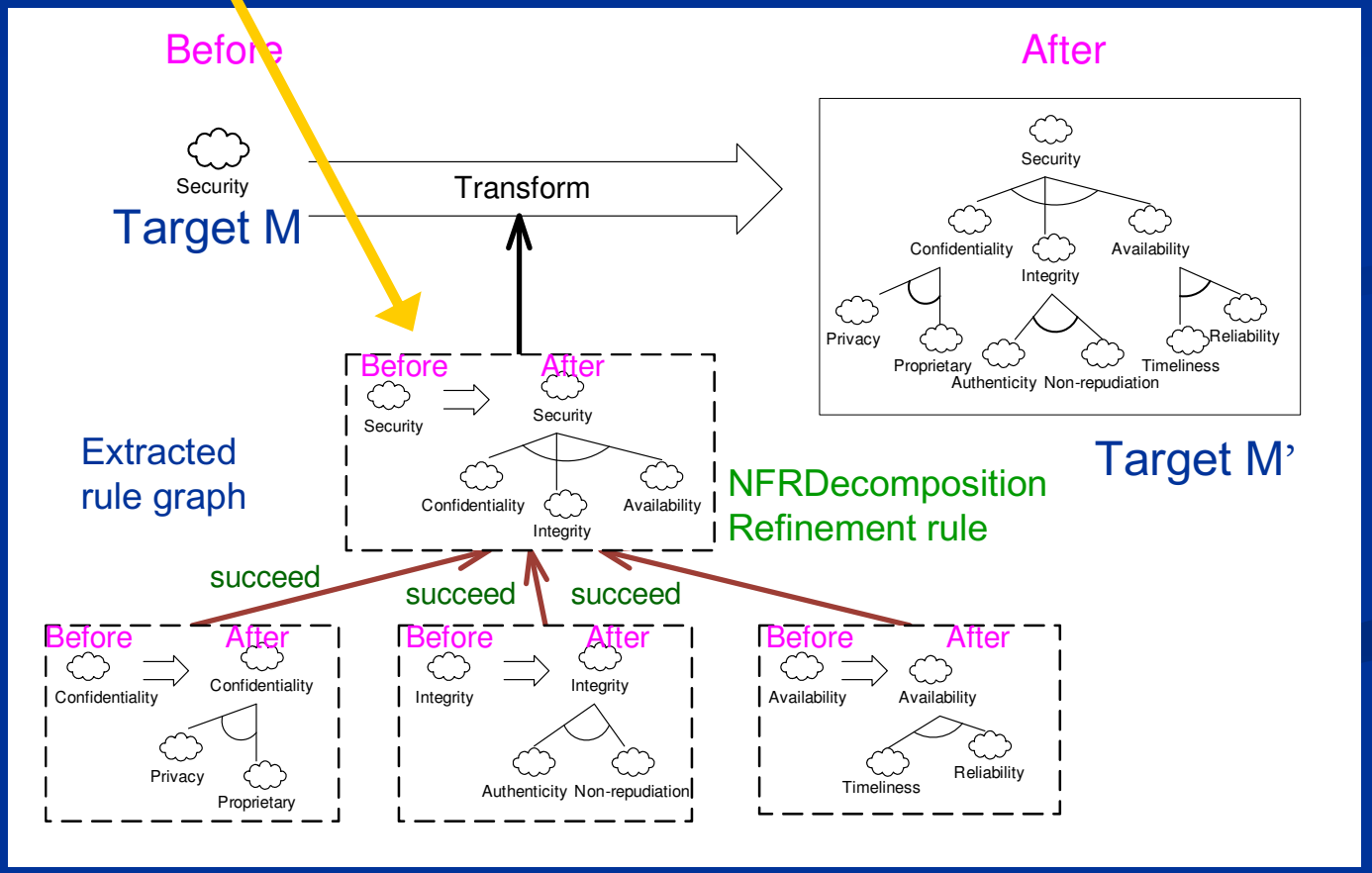




Refinement rules are extracted by “patternize” and used for model transformation by “apply”

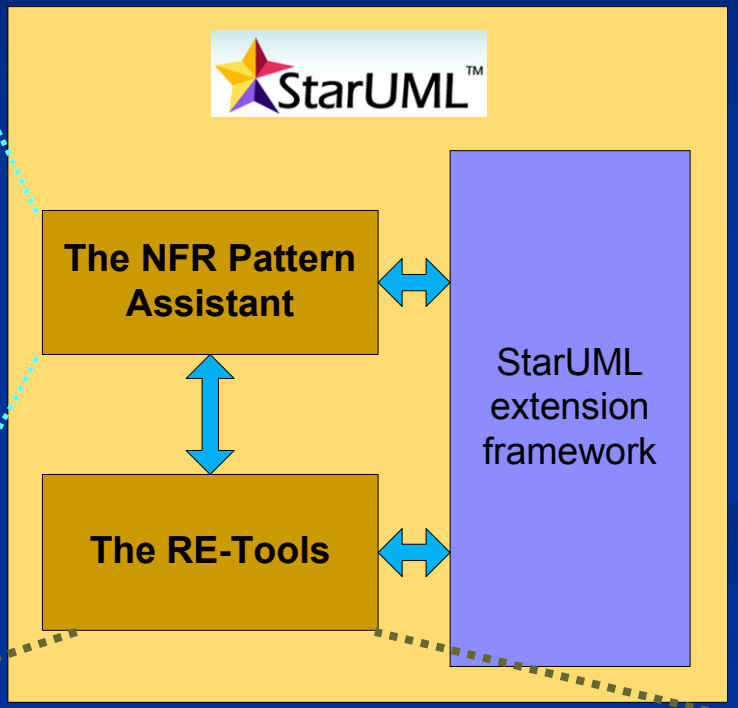


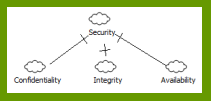




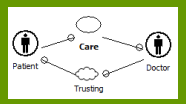
The NFR Pattern Assistant for pattern support
The RE-Tools for knowledge modeling

utdallas.edu/~supakkul/tools/NFRPassist



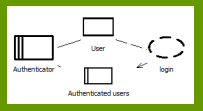


The NFR Framework



The i* Framework

KAOS



Problem Frames

UML

(TBD) . . .

utdallas.edu/~supakkul/tools/RE-Tools

The approach and the tools have been applied to the TJX case

Break-in wireless network
Masquerading user login
Steal credit card info

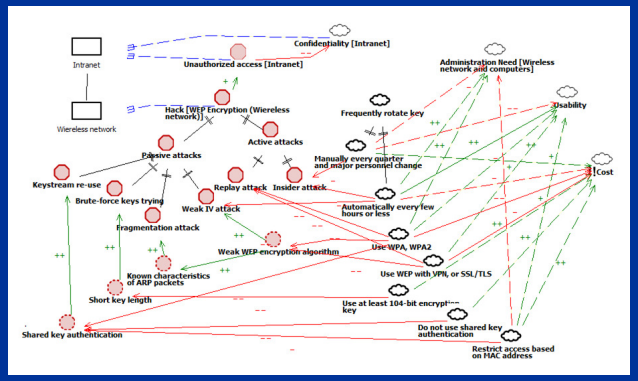
for

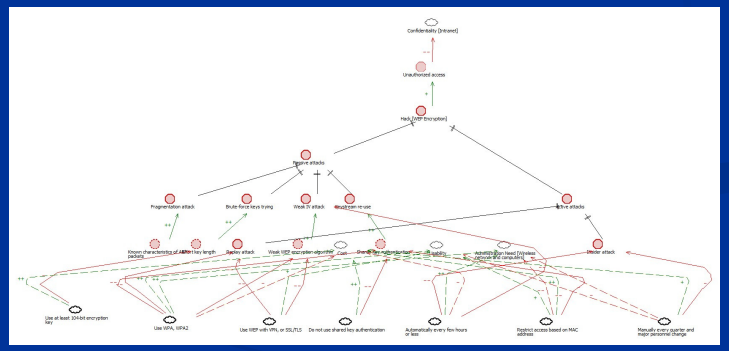
Sample results

reuse knowledge from TJX in a different project

Build for reuse

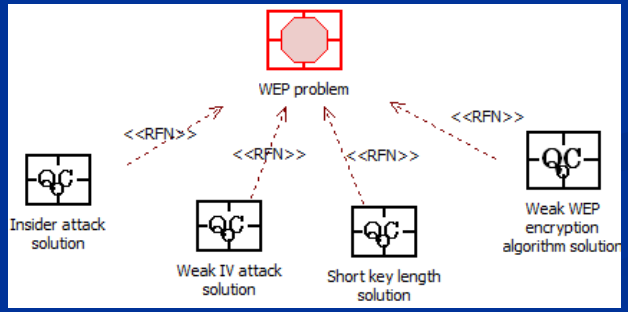
Build with reuse





capture

1 composite, 5 primitive



reuse

Limitations (future work)

Tool/usability related

- Model elements not captured with the original position
- Pattern search and selection are currently manual
- Some knowledge not captured (need 2 more rules)
- Need to support more FRs and NFRs integrated knowledge
- Limited concurrently pattern sharing across groups

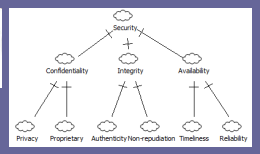
Approach related

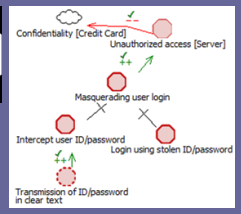
- Costly and time-consuming to learn the notation and the tool
- Need more case studies
- Need to support dealing with NFRs during architecture/design

Summary: The difficulty of capturing, organizing, reusing of NFR knowledge can be alleviated by the approach

	Features in the approach
Difficult to capture	
Problems, solutions, domain	← Objective, problem, alternatives, selection patterns
Complete, correct	← Credentials
Conceptual modeling and reasoning	← Captured softgoal graphs
Difficult to organize	
Cataloging knowledge	← By name, type, applicability
Relating similar knowledge	
General – Specific	← Specialization, composition, instantiation
Class – Instance	
Combining knowledge	
Difficult to reuse	
Choosing appropriate knowledge	← Applicability info
Re-creating visual models	← Refinement rules, tool support

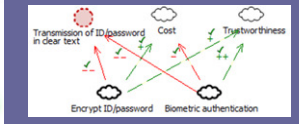
Thank you… Questions & Comments?



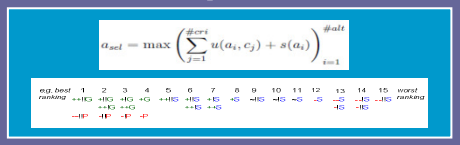


Objective pattern

Problem pattern

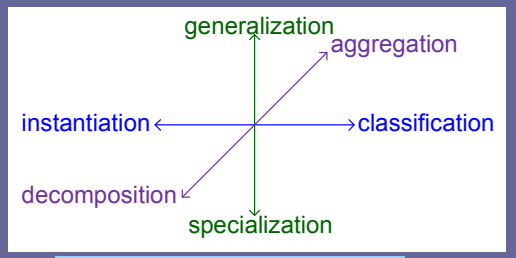


Alternatives pattern

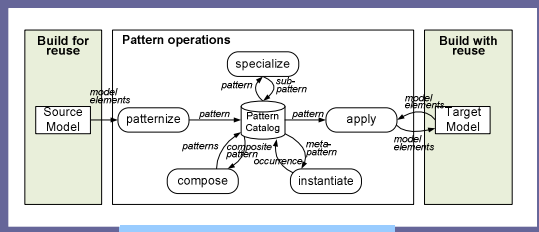


Selection pattern

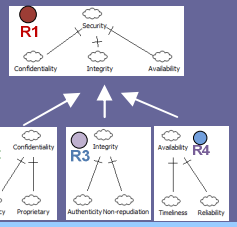
Capturing



Organizing



Reusing



Tool support