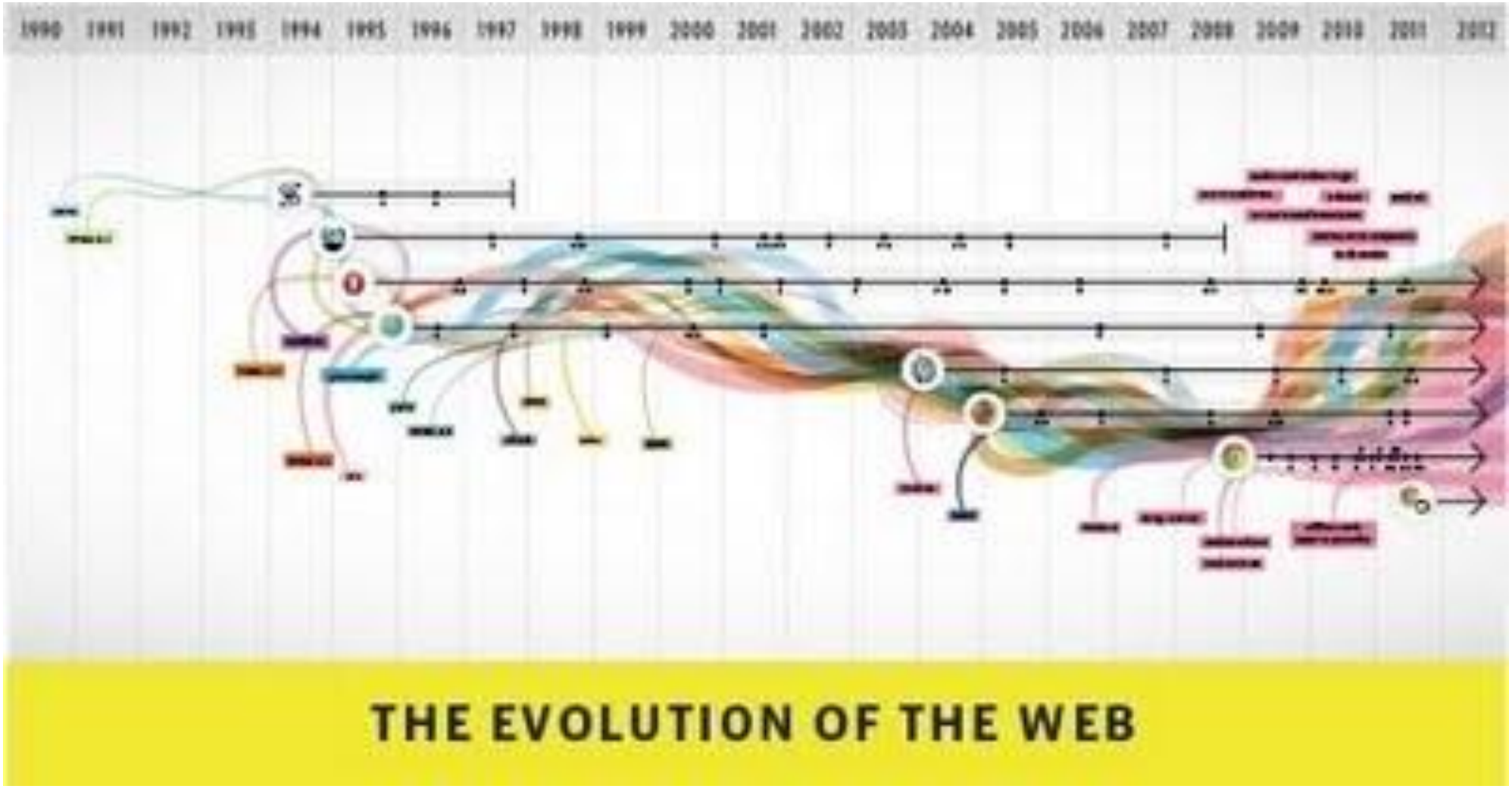
Evolving Software Evolution

This talk provides a brief overview of my learning process at the **University of Trento**, working on the **Lucretius project**. During this time, I have identified gaps in dealing with software evolution. Of those, I have focused on four topics.

1) Species as Central in Software Evolution.
2) Evolving Service Product Line based on Commitments Performance.
3) Modeling of Feedback Mining with Explicit Actors.
4) Security Evolution Based on Interaction of Quality Patterns.

For each of the four topics I will briefly report on working ideas as to improve perceived opportunities.

Talk Given at the *Lucretius Seminar* – 26/07/12 by **Julio Cesar Leite**



http://www.evolutionoftheweb.com

Agenda

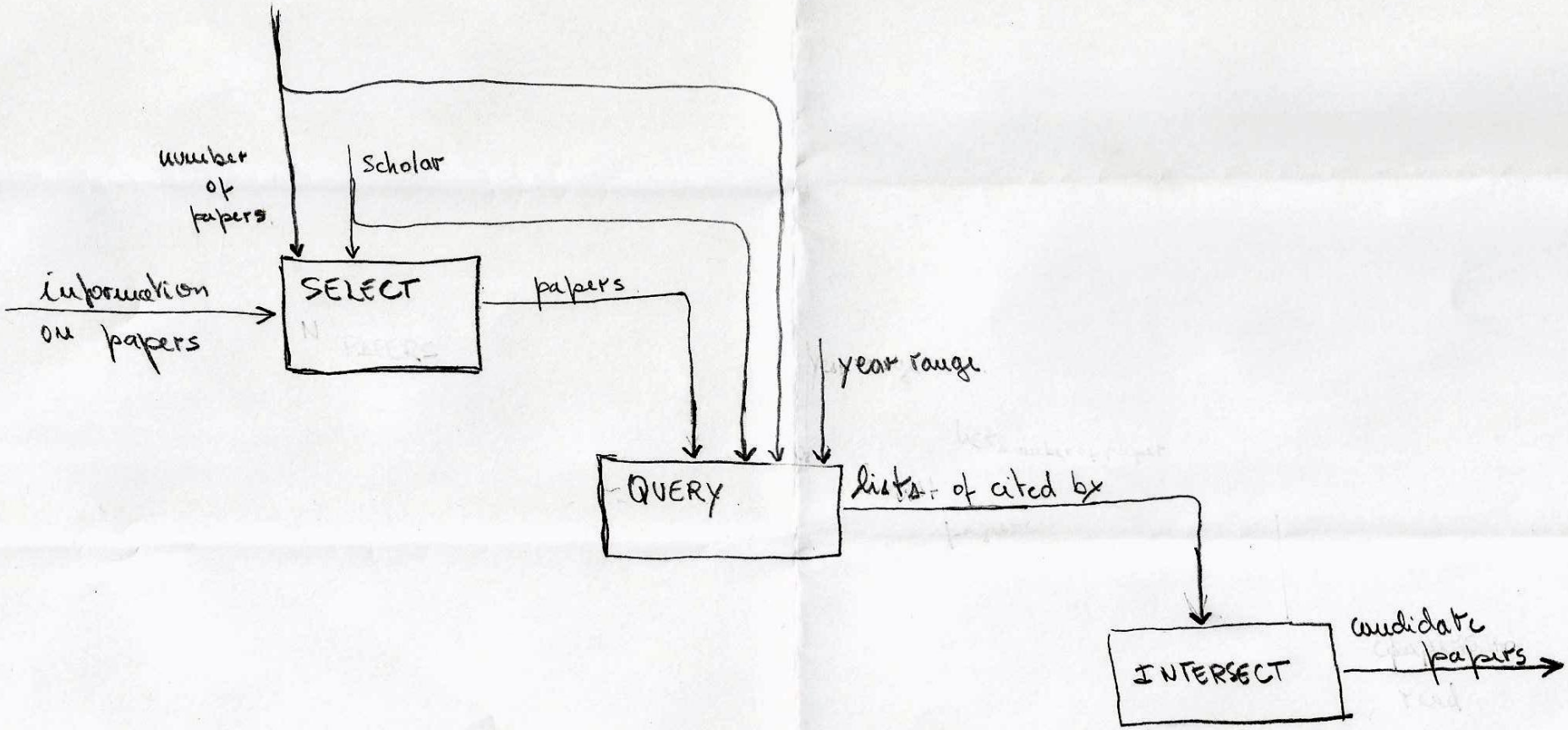
- Information Sources
- Different Elicitation Strategies
- The problem of software evolution definition
- Services also need to evolve
- Internal feedback loop is not enough
- The problem of dealing with security as a functional requirements

Information Sources

- Books
- Old Papers
- Web via Scholar and DBLP
  - Digital Libraries, Berkeley, Stanford, National Center for Biotechnology Information, DNA Interactive, Wikipedia, Direct Contact with Researchers, …
- Lucretius Seminar
- Software Evolution Classes
- You (The group)
- Meetings, Personal Conversations, Papers

Strategies

- Avoid Systematic Revision
- Good old journals
- Realms (Universe of Discourse)
  - ✓ Biology → LEL
  - ✓ Software Evolution
  - ✓ Requirements
  - ✓ Design (from different pov)
  - ✓ Ontology
  - ✓ Management
- Scholar
  - ✓Number of Citations
  - ✓Find papers to read (intersection (lists of cited by (select papers))



FIND PAPERS TO READ

The Definition Conundrum

- Maintenance encompass Evolution
- Evolution encompass Maintenance
- Evolution is technical; Maintenance is managerial
- Why not looking at Biology?
  - Goodfrey and German
  - But Biology is still debating (species, intersection of genotype/phenotype)*
- Several taxonomies (naturally type oriented)
- Even ontology
- Difficult to stick

* What is a Species? In Journal of the General Philosophy of Science -- “Requirements for a computable connection between genotype and phenotype across evolution” in Phenotype ontology: the bridge between genomics and evolution

Genus

- In the tree of life, genus is the upper hierarchy above species

•“Evolution, simply put, is descent with modification” (Berkeley, 101)

•“Speciation: the process by which species form. This involves the reproductive isolation of different parts of an ancestral species so that they form distinct descendent species.” (Berkeley, 101)

•“Evolution begins with the inheritance of gene variations.” (DNA Learning Center)

•“Populations (animals): It is a group of organisms that interbreed” (Berkeley 101)

• “Micro evolution as a change in gene frequency within a population.” (Berkeley 101)

Berkeley 101

1. Beetles on a diet
Imagine a year or two of drought in which there are few plants that these beetles can eat. All the beetles have the same chances of survival and reproduction, but because of food restrictions, the beetles in the population are a little smaller than the preceding generation of beetles.

2. Beetles of a different color
Most of the beetles in the population (say 90%) have the genes for bright green coloration and a few of them (10%) have a gene that makes them more brown.
Some number of generations later, things have changed: brown beetles are more common than they used to be and make up 70% of the population.

The difference in weight in example 1 came about because of environmental influences — the low food supply — not because of a change in the frequency of genes. Therefore, example 1 is not evolution. Because the small body size in this population was not genetically determined, this generation of small-bodied beetles will produce beetles that will grow to normal size if they have a normal food supply.

The changing color in example 2 is definitely evolution: these two generations of the same population are genetically different

Berkeley 101

Phenotype

the physical features of an organism. Phenotype may refer to any aspect of an organism's morphology, behavior, or physiology. An organism's phenotype is affected by its genotype and by its environment.

Genotype

the set of genes an organism has. Sometimes, genotype refers to the entire genome of an organism and sometimes it refers to the alleles carried at a particular locus.

Software Genus

- The group is working with Professor John Mylopoulos's insight that Version defines Species
- A Software* defines a Genus, which may have different species.
- Descent is achieved by copying.
- Descent with modification establish a new Species (Version) of a Software
- Descent produces new individuals, each individual (copy) has a unique id.
- Individuals may be different, due to change (aging or development**) in the individual phenotype.

* Software Product / Application / Program / System / Software System / Service

** Development is the process that occurs as a living thing grows up (ontogeny). Evolution is change of form or behavior of a population over time (phylogeny). (Berkeley 101)

Software Genus

• Changes within a given Version are seen as analogous to changes in Phenotype, but not in the Genotype.

• As such, a clear separation among what is evolution, leading to a new spices, from what is considered functions* (bio) changes.

• What is the advantage of this concept?

• A clear distinction of evolution, but what determines change in the Genotype (set of genes)?

• Godfrey and German named the software source code as genes, but this definition is problematic. What will be considered the genotype? How the changes will be detected?

• Our proposal relies on Version as standing for Species, so a new Version determines a change in genotype. Engineers control what is the evolution.

*Designed tools have purposes. Structures and behaviors of living things have functions. This is an important distinction in the science classroom. (Berkeley 101)

What is Ahead

• If we take the idea that K, P |-# G, Q, A or the S, K |-- R (S must be such that an implementation of it, connected to the environment ensures that all the properties of R are satisfied.)

• And since for each pair (K, R) or tuple (K, G, Q, A) there may be several possible S or P (solutions to the requirements problem)

• Then which component should we pick to determine the change in the genotype?

• Is the choice depended on the Family (upper hierarchy of Genus)?

• Is it worth building a tree of life or a Phylogeny for Software?

• An Applied Ontology perspective may be of help.

Services Need to Evolve

- Service: a software that is independently provided and which can be combined with other services, if an interaction protocol is shared.
- An interaction protocol is an implementation of a set of commitments that different actors may stablished in their social relationship.
- An Actor in a social relationship may use a service, may provide a service, or may regulate an interaction protocol, or ... .
- If commitments among actors change, so their protocol implementations.

Services Need to Evolve

- As such Services need to be aware (Service Awareness) of possible commitments changes.
- Since Services may engage in a different set of protocols, they may adopt a Service Product Line structure or a Goal Oriented structure, allowing for flexibility.
- It is possible to plan for different protocols, but how to plan for service change according to the possible changes in commitments?
- How commitments may change?

Services Need to Evolve

- Monitoring Services as to Support Commitment change
- Using a case of “failure to deliver”
- Using SA (Strategic Actor Diagram) and SD (Strategy Dependency) to Model a Possible Feedback Loop.

A Real Example

“ To date,
the relationship between ineffective order fulfillment and customer behavior remains unexamined.”

“Our investigation focused upon the relationship between operational failures, and customers’ future buying behavior. The results provide empirical evidence that when Internet retailers fail to deliver upon order fulfillment promises, customers react negatively. Measures of operations performance, such as order fill rate and unit fill rate have been long used as inward looking measures of operations performance (Boyd and Cox, 1997; Johnson and Davis, 1998; Chan et al., 2003; Closs et al., 2010). However, our results suggest that while these types of measures are important, they may not sufficiently predict the full reaction of the customer to operations performance. Historically, operations managers have focused upon minimizing the number of late orders, tracking order and unit fill rates to gauge their performance. This research extends this examination beyond the numbers and percentages of late orders, and reveals the implications of the magnitude by which missed orders are deficient. Rather than view late orders equally, our results suggest that operations managers should endeavor to minimize not only their occurrence, but when they are expected to be late, to consider expending extra effort to minimize that lateness as well, given the observed reaction of the customer to late orders.”

Failure to deliver? Linking online order fulfillment glitches with future purchase behavior
Shashank Raoa, Stanley E. Griffisb, Thomas J. Goldsbyc
Journal of Operations Management 29 (2011) 692–703

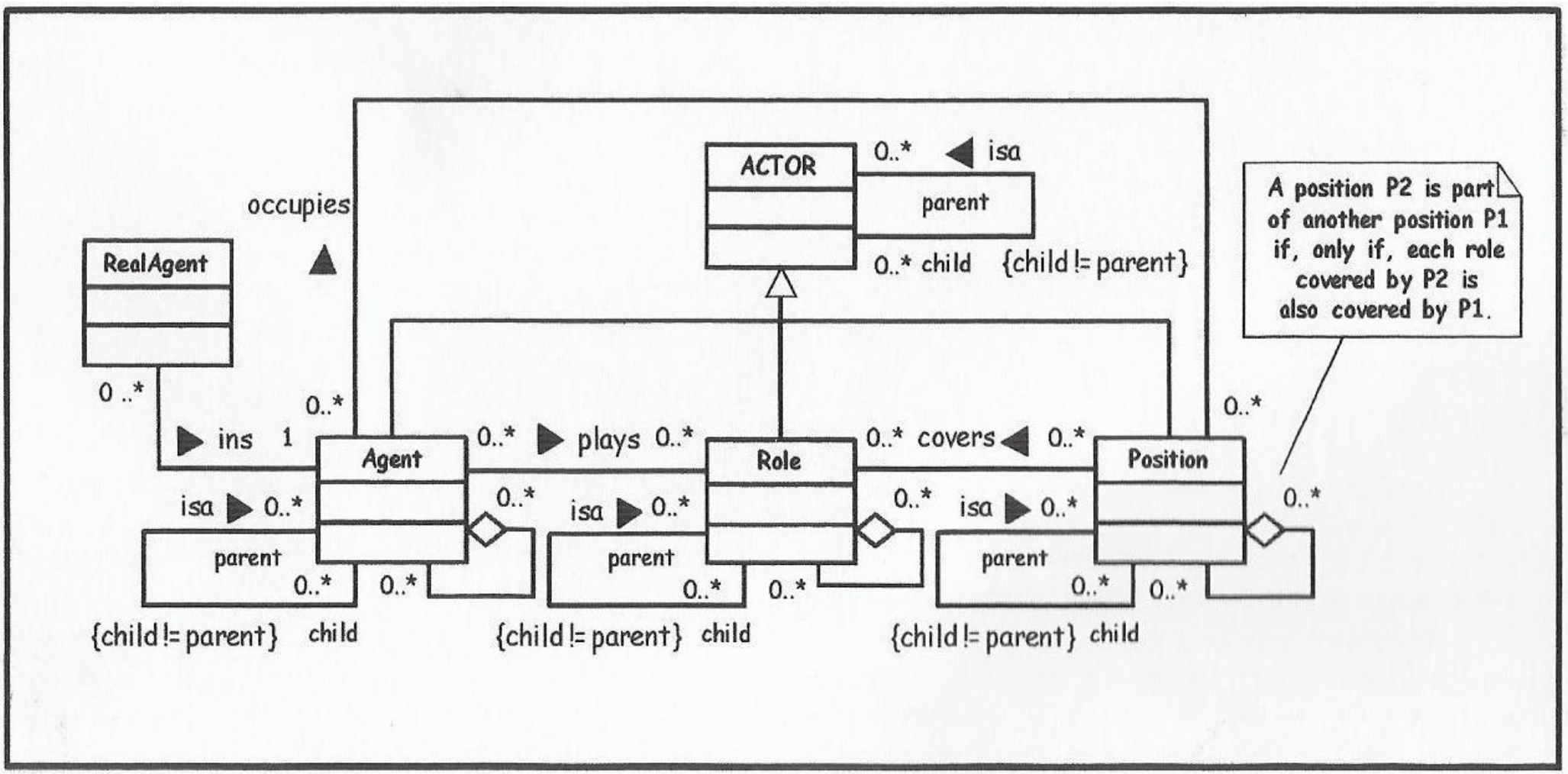
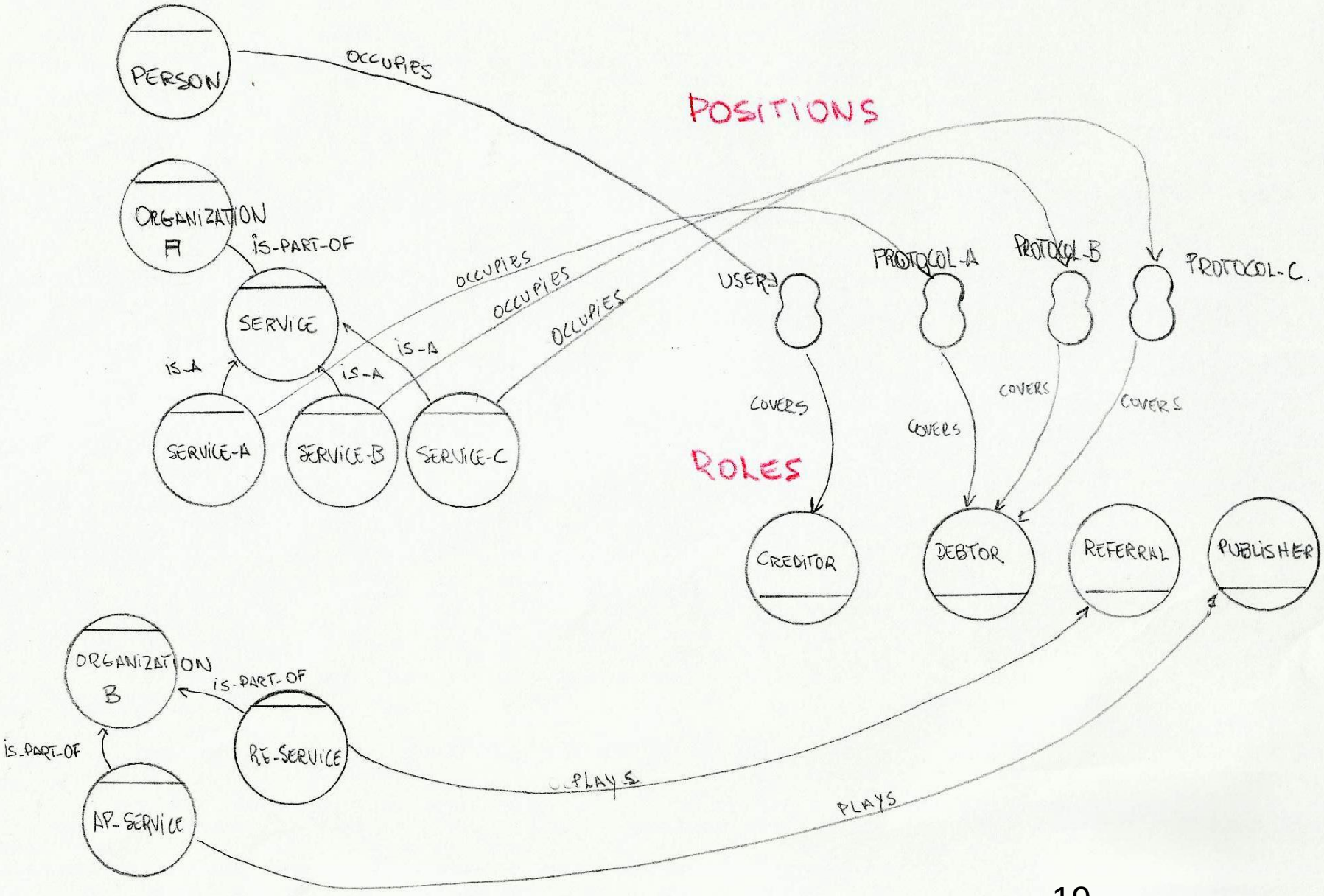
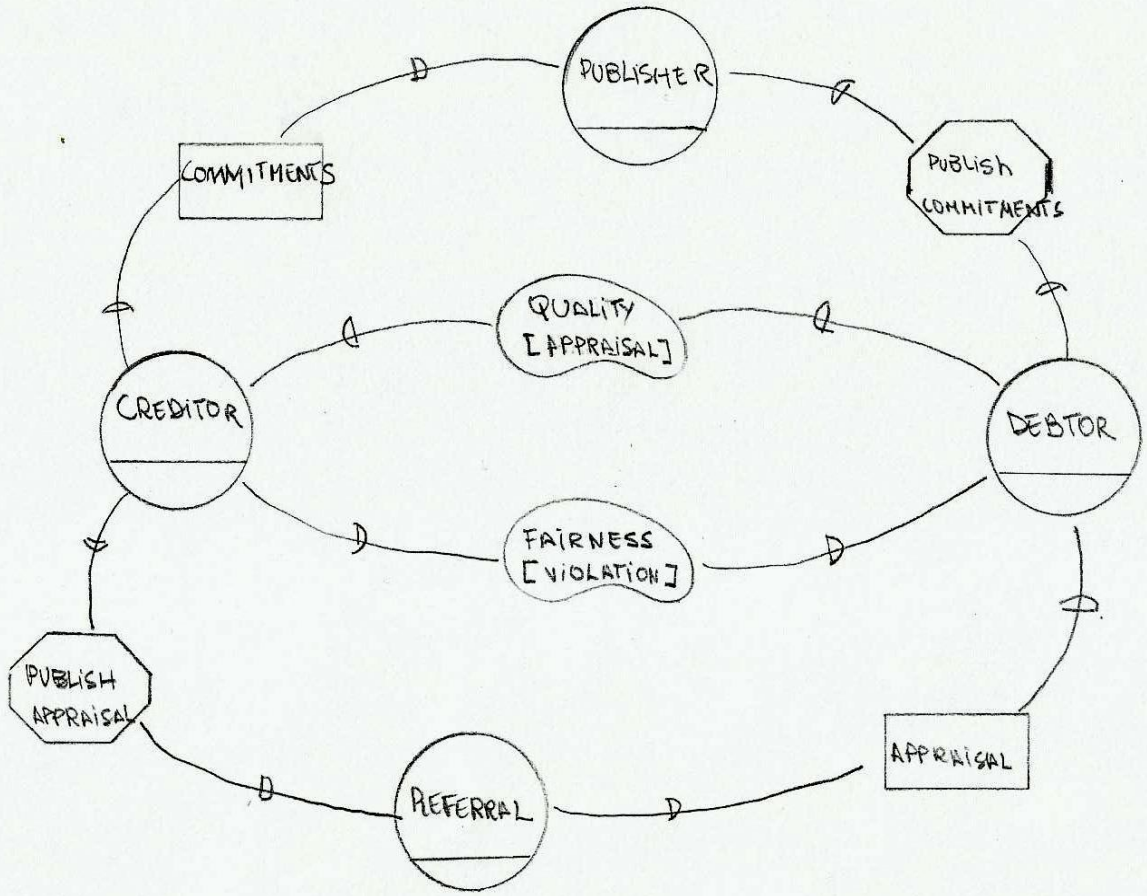
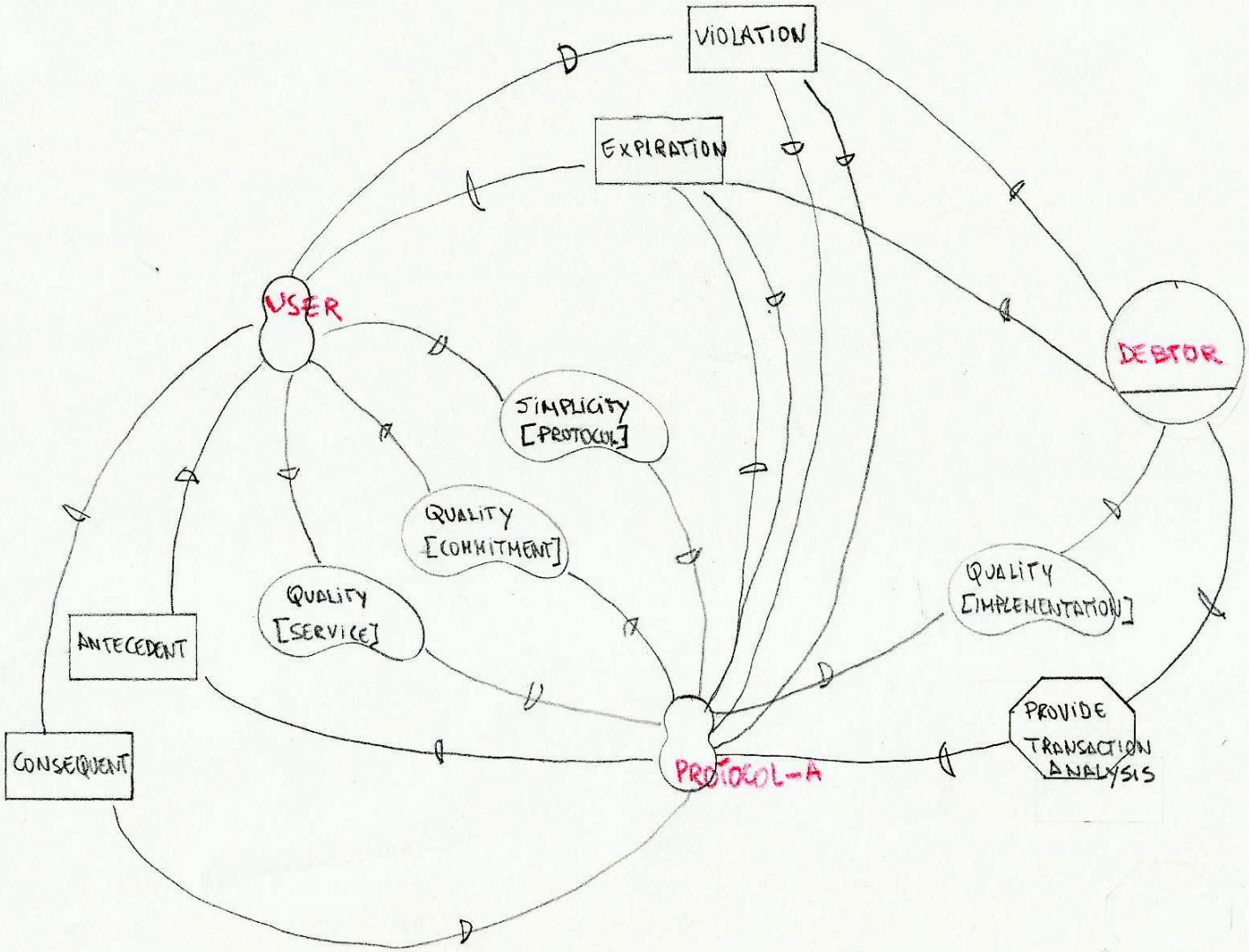


Figure 9: Proposed Model Using UML Specialization and Aggregation

AGENTS

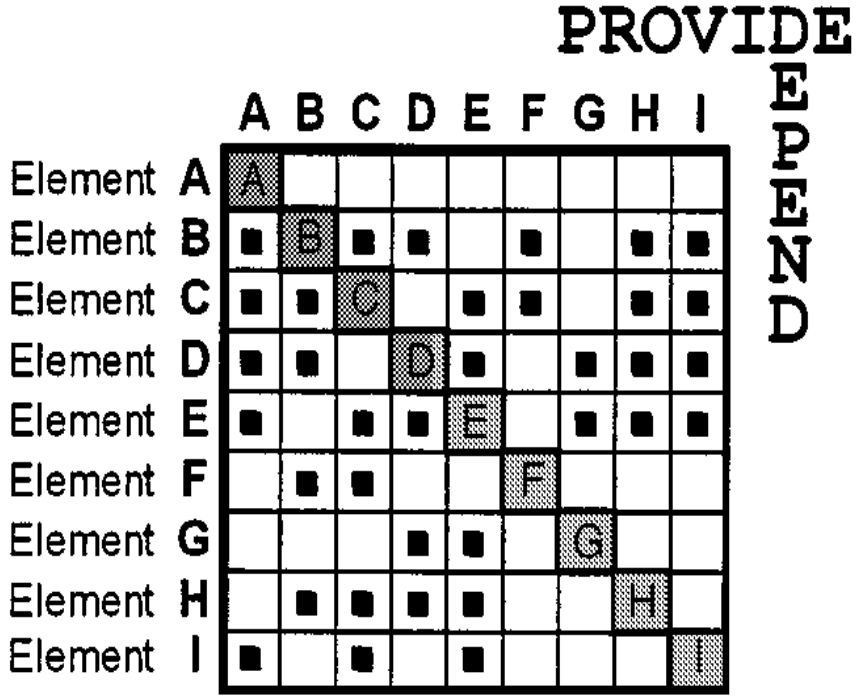

PERSON
OCCUPIES
POSITIONS
ORGANIZATION A
IS-PART-OF
SERVICE
OCCUPIES
OCCUPIES
OCCUPIES
IS-A
IS-A
IS-A
SERVICE-A
SERVICE-B
SERVICE-C
USER
PROTOCOL-A
PROTOCOL-B
PROTOCOL-C
COVERS
COVERS
COVERS
COVERS
ROLES
CREDITOR
DEBTOR
REFERRAL
PUBLISHER
ORGANIZATION B
IS-PART-OF
IS-PART-OF
RE-SERVICE
AP-SERVICE
PLAYS
PLAYS


PUBLISHER
COMMITMENTS
PUBLISH COMMITMENTS
QUALITY [APPRAISAL]
CREDITOR
DEBTOR
FAIRNESS [VIOLATION]
PUBLISH APPRAISAL
REFERRAL
APPRAISAL


VIOLATION
EXPIRATION
USER
DEBTOR
SIMPLICITY [PROTOCOL]
QUALITY [COMMITMENT]
QUALITY [SERVICE]
QUALITY [IMPLEMENTATION]
ANTECEDENT
CONSEQUENT
PROTOCOL-A
PROVIDE TRANSACTION ANALYSIS

What is Ahead

- How to deal when there is a lack of Transparency (Referral, Appraisal)?
- Dealing with complex Antecends and Consequents (can it be modeled as actors – roles?).
- Analyzing Delegations (is the Design Structure Matrix a possibility ?)



Applying the Design Structure Matrix to System Decomposition and Integration Problems:
A Review and New Directions (Tyson R. Browning)

IEEE TRANSACTIONS ON ENGINEERING MANAGEMENT, VOL. 48, NO. 3, AUGUST 2001

Beyond Internal Feedack

- Requirements awareness is great to deal with events under the Span of Control of the Requirements Model.
- How about events out of the span of control?
- Coming back to Jackson and Zave formula (S, K |- - R (S must be such that an implementation of it, connected to the environment ensures that all the properties of R are satisfied.)), that is to say: we know how to deal with R and S (Requirements Awareness), but how about K?

Beyond Internal Feedack

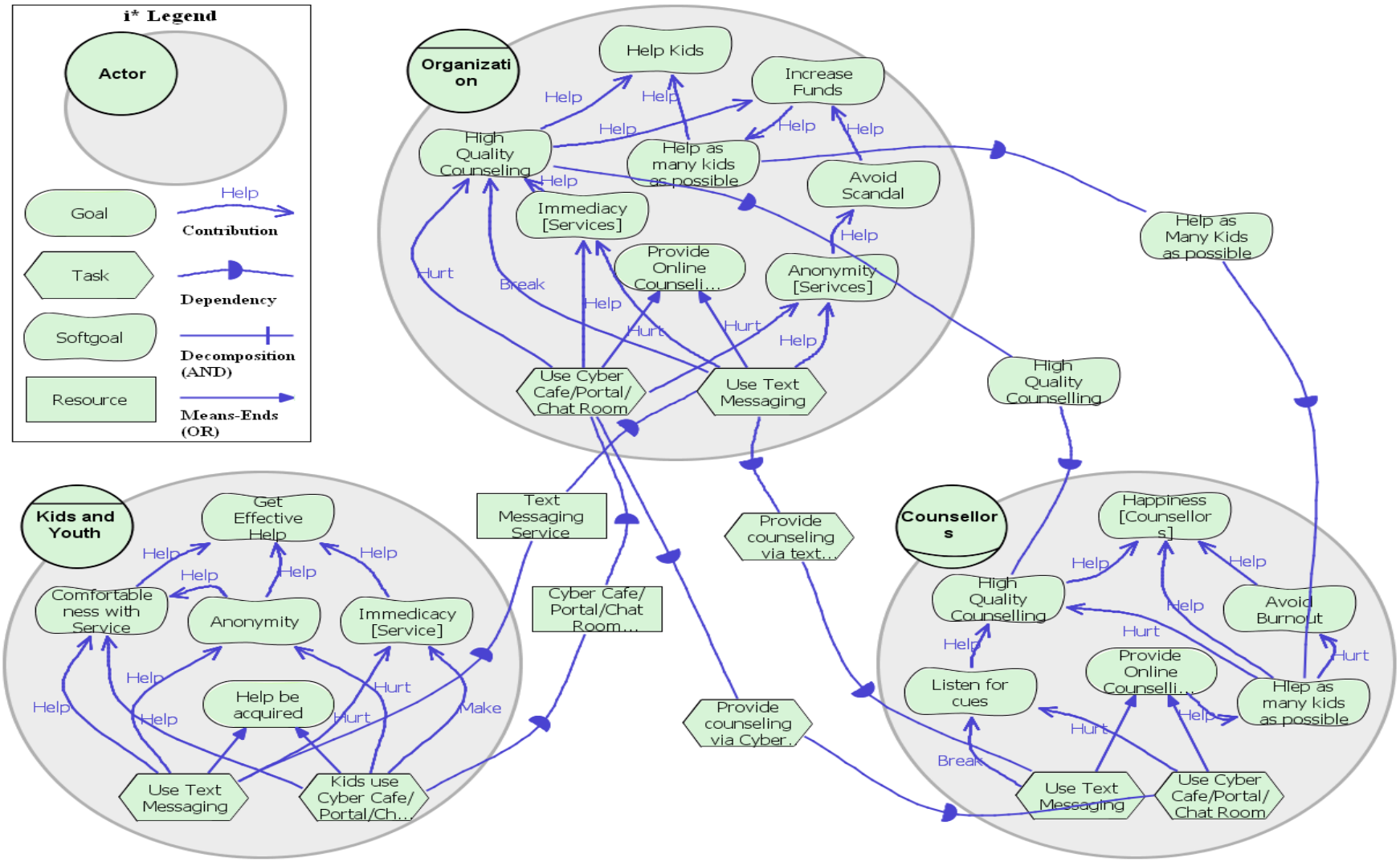
- Does it assumes modeling the world? Opps... we should be smarter (Dynamic Interrogation*).
- A possible way is focusing on Information Sources, that is which Information Sources we have used to model K.
- As such, we have to model the Information Sources that we used to build the requirements of a given system (an artifact designed as a single logical entity).

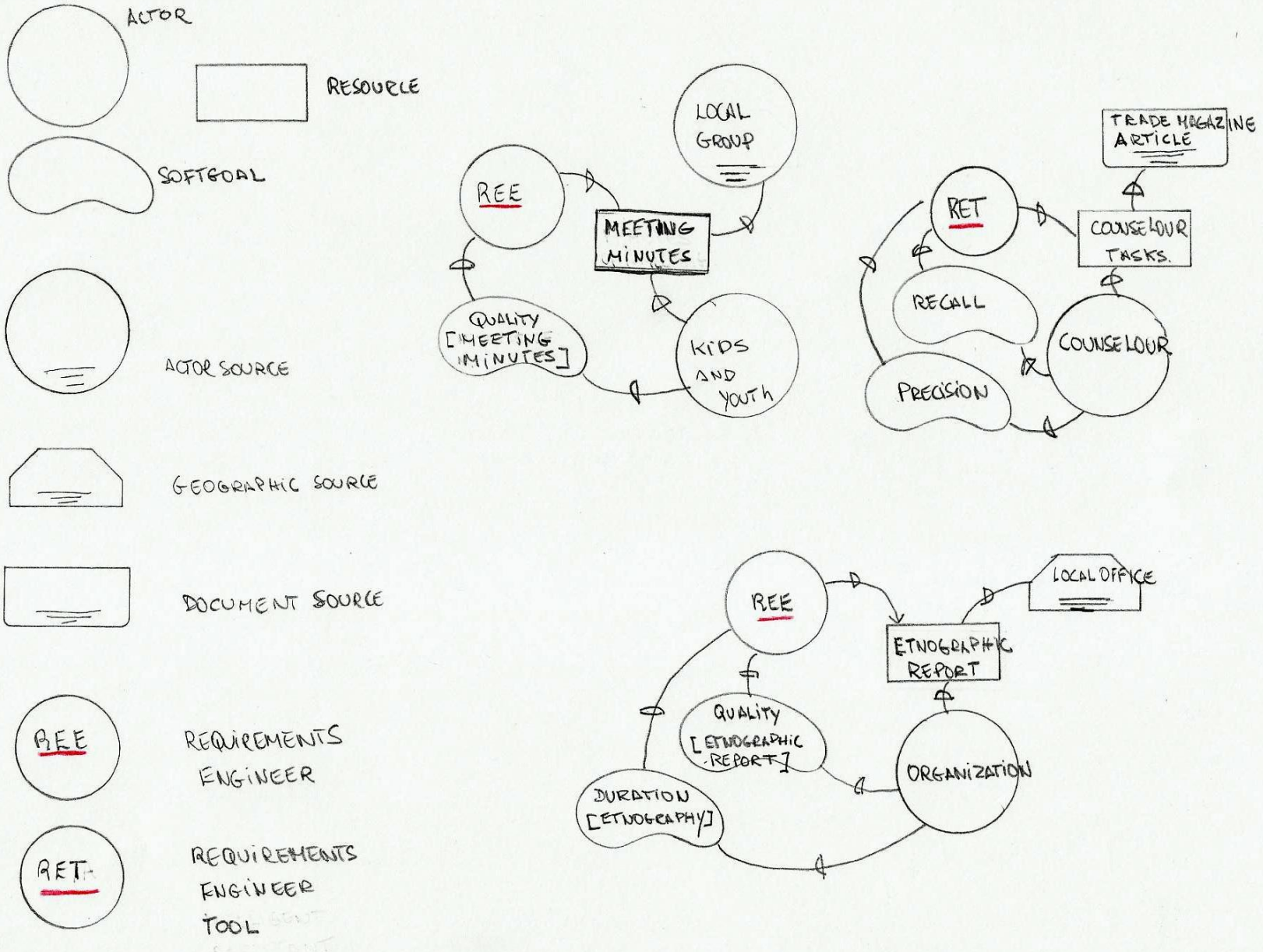
*Dynamic Interrogation: a way out of the environment modeling trap?, Goguen, J, Leite, J.C.S.P., Breitman, K. Draft of a Research Project, 1999 (never published or submitted)

Beyond Internal Feedack

- We propose the Information Source Diagram.
- An ISD maps the dependency of an IS (which can be an actor, a document, or a geographical location) with an eliciting actor (Requirements Engineer(s)) and with the modeled actor (actor, agent, role, position) of an i* SD.
- Dynamic Interrogation works on these relations as to detect possible changes using Mining technology.
- Usual IS include Trade Magazines, or Competitors Systems.

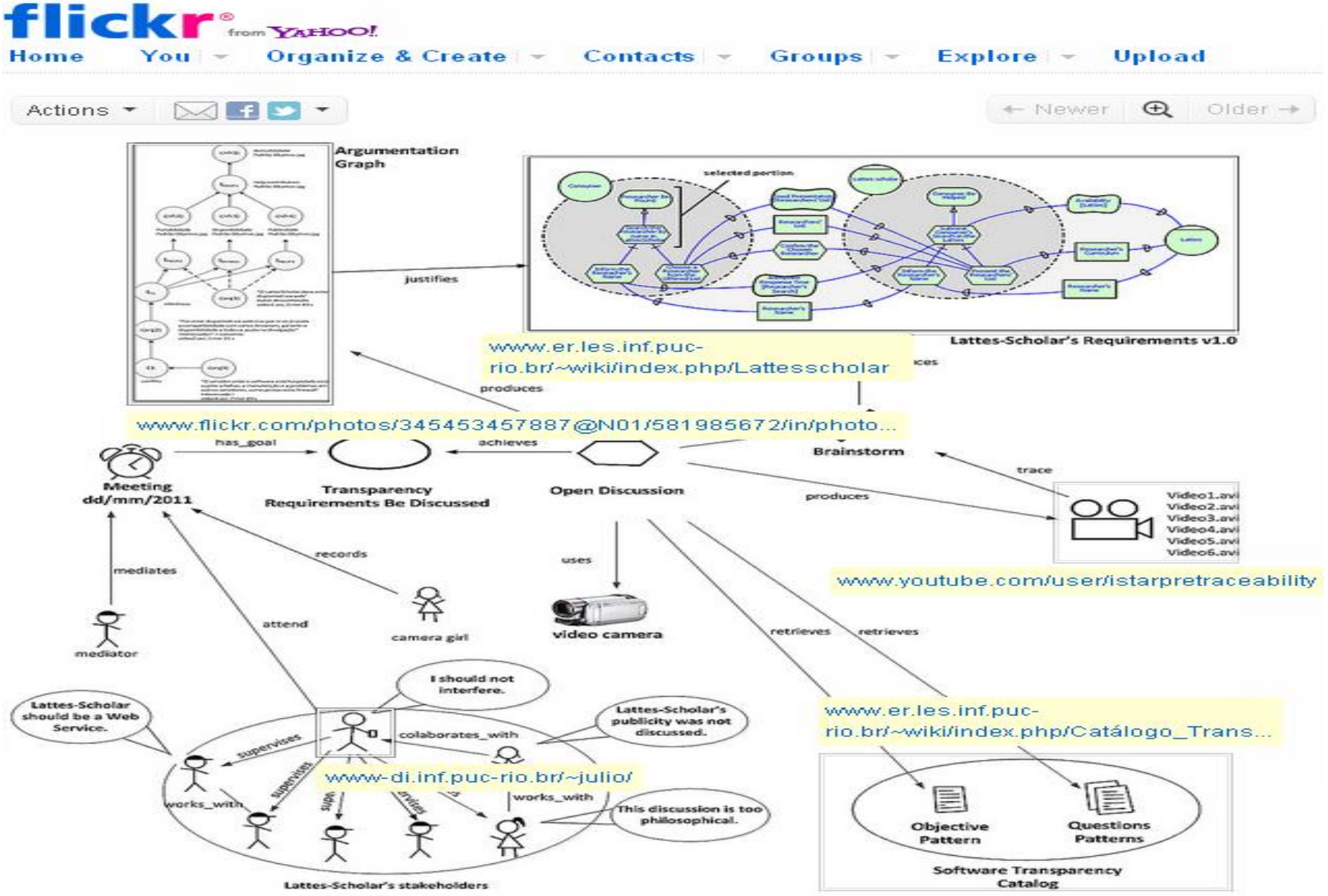
An Example: From Jennifer Horkoff





INFORMATION SOURCE DIAGRAM

An Example of Resource in a ISD



iTrace of meeting dd-mm-2011

What is Ahead

- Dynamic Interrogation is supposed to sample, thus avoiding large models. The question is what to sample over time, from where.
- Dynamic Interrogation of IS can be performed by several different strategies, in particular mining, when the information source has transparent characteristics (access, usability, informativeness...).
- Heuristics for sampling should be based on the current model, where resources and tasks (names) are a reasonable seed. So, how to determine indicators (essential variables)?
- Simpler DI may be made by interviews or questionnaires, which later could be mined.
- However, this is just the monitoring part, controlling is another issue.

Security Evolution

- Security is a general "concern", since its main concept can be applied to a broad range of artifacts.
- However each case has particulars, so there is a need to tailor a security policy to the case at hand.
- If I apply security to a car, to a house, to a mobile phone, or to an organization, I will use a general concept to make the car more secure, the house more secure, the mobile phone more secure and the organization more secure. Ok, so what?

Security Evolution

- So, we can “reuse” the concept of security, for a car, a house, …. However, we can not “reuse” the concept of car for a house or for a mobile phone.
- This is a powerful hint that there are of different nature: one is particular (domain) the other is general (cross domain).
- If they are general, then…Well, how about reusing security? (Quality-Based Software Reuse – Caise 2005).

Security Reuse

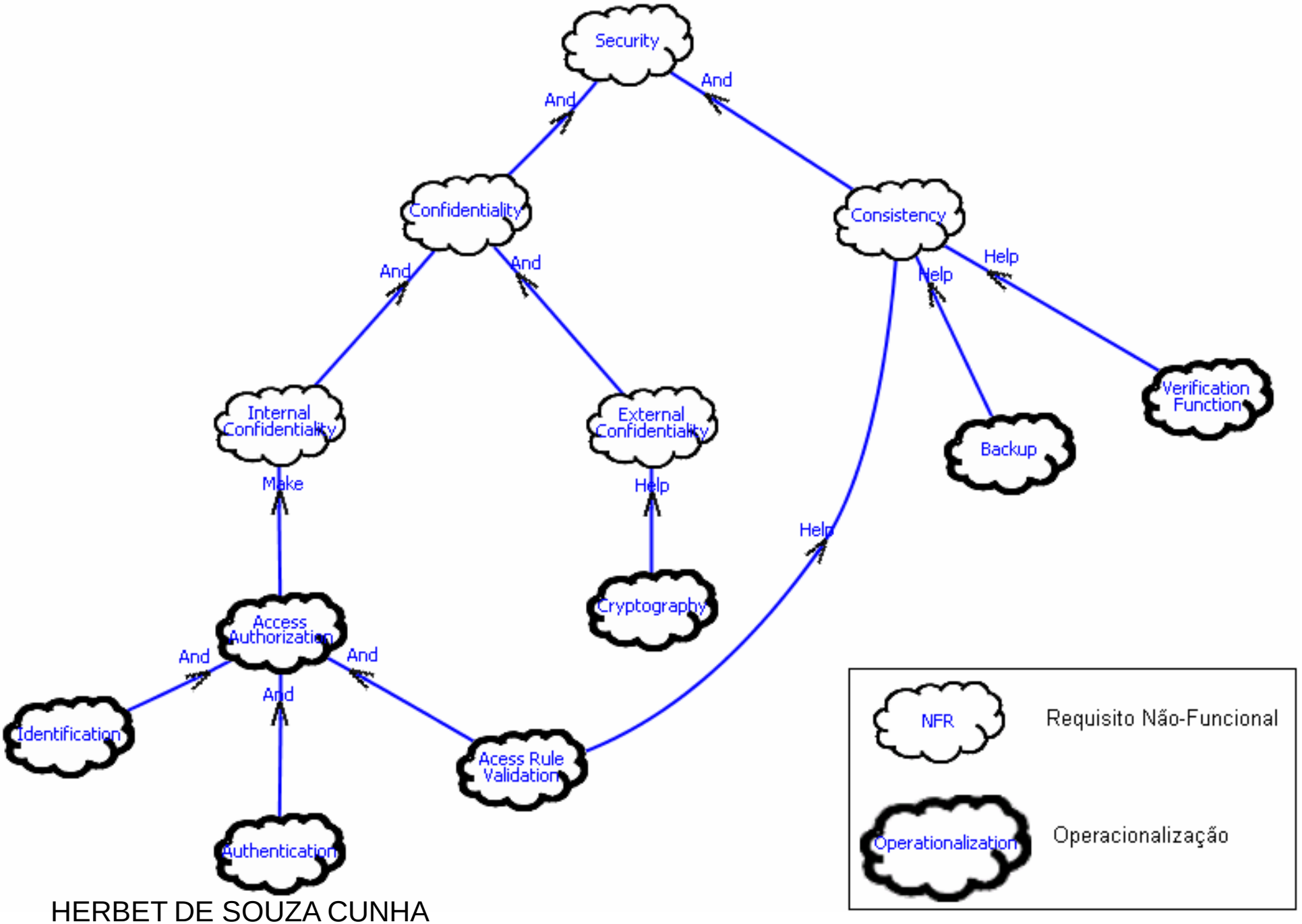
- Building from scratch is most of the time harder than building from previous successful and transparent knowledge.
- So how does one reuse Security? Catalogues, patterns which manage the related knowledge, linking the general concepts to particular instantiations of the concept.
- So, this is great. Well, wait. How about if I find a perfect case (pattern) that needs just a few adjustments, will it bring security to my “system”?

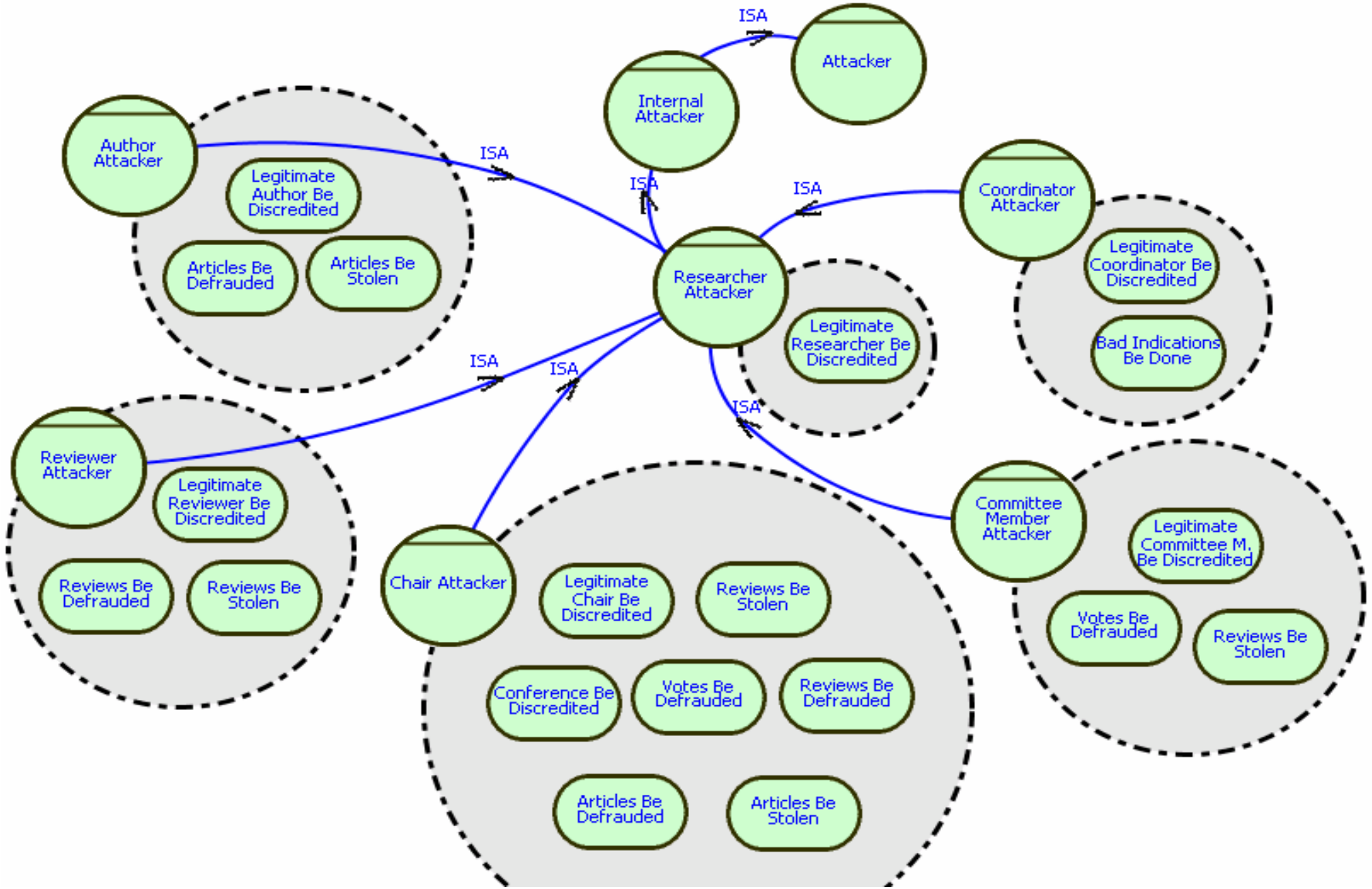
Security Reuse

- Well, it may… but how about the cost, the performance, the usability, the privacy, the …
- So, patterns need to also organize (store) the relationships among those issues, but more, those issues are cross domain.
- As such, those patterns would profit if represented by softgoal based representations, which relies on correlations and contributions.
- But how about evolution? How about have a collection of patterns that evolve? Analogy with Norton

Security Evolution Based on Actors

- The overall idea is to focus on possible actors who may bring insecurity to a given system (see as a set of designed actors).
- Such actors will develop strategies to make the system insecure, so they will have patterns to do so.
- Combining the general patterns (with different instantiations) with patterns for disruption of security (organized by actors) is a general framework for evolving security.


Security
And
And
Confidentiality
Consistency
And
And
Help
Help
Internal Confidentiality
External Confidentiality
Verification Function
Backup
Make
Help
Help
Cryptography
Access Authorization
And
And
And
Identification
Acess Rule Validation
Authentication
NFR
Requisito Não-Funcional
Operationalization
Operacionalização


ISA
Attacker
Internal Attacker
Author Attacker
ISA
Legitimate Author Be Discredited
Articles Be Defrauded
Articles Be Stolen
ISA
ISA
Coordinator Attacker
Researcher Attacker
Legitimate Researcher Be Discredited
Legitimate Coordinator Be Discredited
Bad Indications Be Done
ISA
ISA
ISA
Reviewer Attacker
Legitimate Reviewer Be Discredited
Reviews Be Defrauded
Reviews Be Stolen
Chair Attacker
Legitimate Chair Be Discredited
Reviews Be Stolen
Conference Be Discredited
Votes Be Defrauded
Reviews Be Defrauded
Articles Be Defrauded
Articles Be Stolen
Committee Member Attacker
Legitimate Committee M. Be Discredited
Votes Be Defrauded
Reviews Be Stolen

What is Ahead

- Effort on building set of patterns
- Effort on building attackers strategies
- Is it similar to other softgoals patterns?
- Impact analysis based on change.
- Change management based on early warning.
- What is the interaction among Security and Transparency, are they just antagonist?