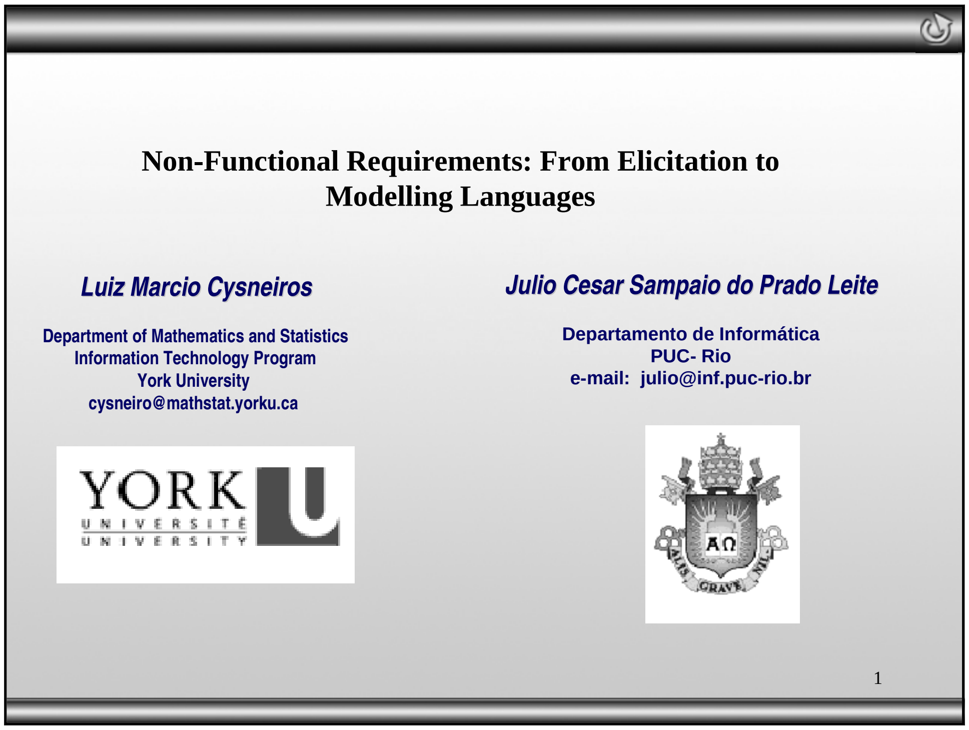


Non-Functional Requirements: From Elicitation to Modelling Languages

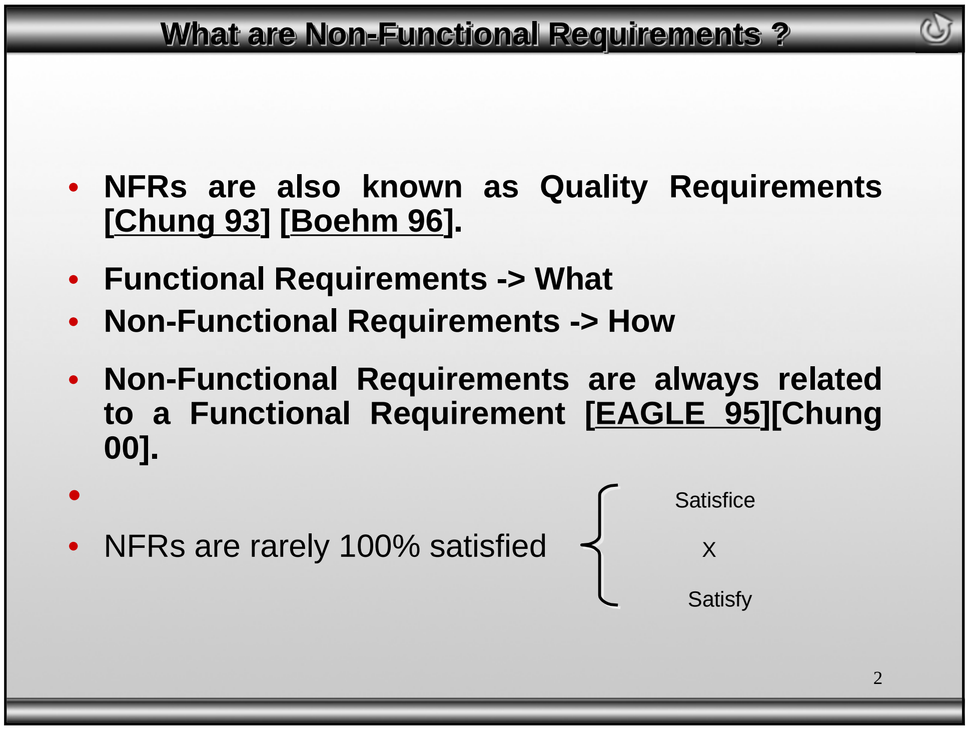
Luiz Marcio Cysneiros

Department of Mathematics and Statistics
Information Technology Program
York University
cysneiro@mathstat.yorku.ca

Julio Cesar Sampaio do Prado Leite

Departamento de Informática
PUC- Rio
e-mail: julio@inf.puc-rio.br

1



What are Non-Functional Requirements ?

- **NFRs are also known as Quality Requirements [Chung 93] [Boehm 96].**
- **Functional Requirements -> What**
- **Non-Functional Requirements -> How**
- **Non-Functional Requirements are always related to a Functional Requirement [EAGLE 95][Chung 00].**
-
- NFRs are rarely 100% satisfied $\left\{ \begin{array}{c} \text{Satisfice} \\ \text{x} \\ \text{Satisfy} \end{array} \right.$

2

Non-Functional Requirements examples

- Adaptability
- Architectural integrity
- Cost
- Configurability
- Efficiency
- Maintainability
- Flexibility
- Profitability
- Performance
- Usability
- Understandability
- Risk
- Resilience
- Reusability
- Time to Market
- Reliability
- Security
- Modularity
- Portability
- Size
- Safety
- Testability
- Mobility
- Standard compliant
- Robustness
- Complexity
- Learnability

Why Care About Non-Functional Requirements

Why Non-Functional Requirements ?

- **Nowadays, the market demands more and more non-functional aspects to be implemented in information systems besides its functionality.**
- **Recent works [Dardene 93] [Mylopoulos 92] [Chung 00] have shown that complex conceptual models must deal with non-functional requirements.**
- **Errors due to omission of NFRs or to not properly dealing with them are among the most expensive type and most difficult to correct (Mylopoulos 92) (Ebert 97) (Cysneiros 99).**

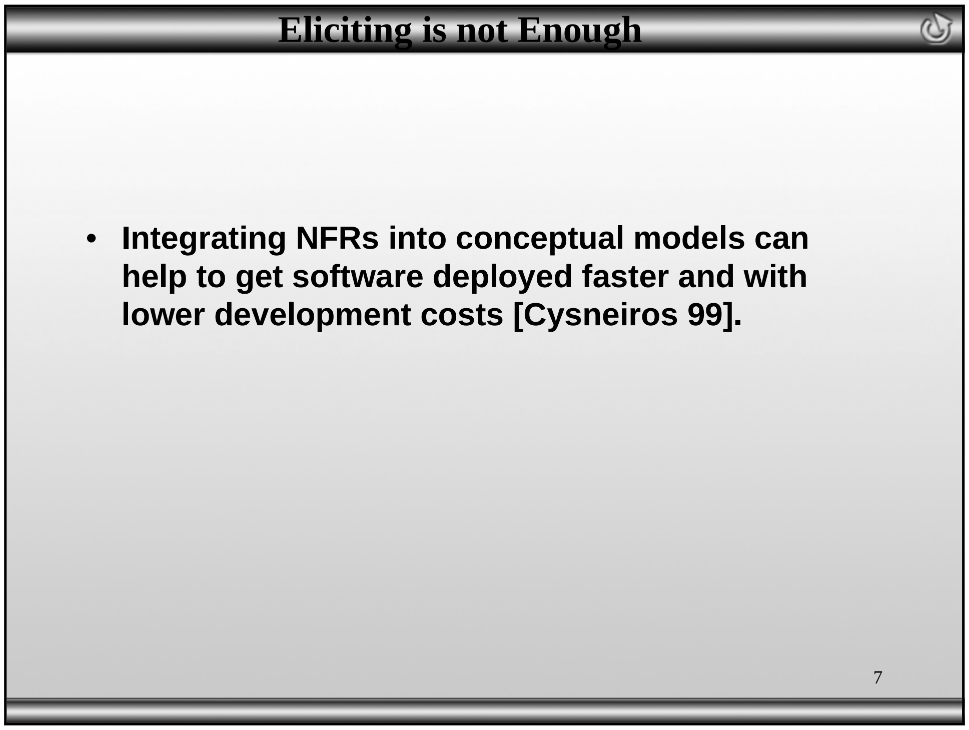
5

One case of Failure Due to neglecting non-functional requirements (NFRs)

The London Ambulance System was deactivated just after its deployment because, among other reasons, many ***non-functional requirements*** were neglected during the system development

e.g. ***reliability*** *(vehicles location)*, ***cost*** *(emphasis on the best price)*, ***usability*** *(poor control of information on the screen)*, ***performance*** *(the system did what was supposed to do but he performance was unacceptable)*, [Finkelstein 96] [Breitman et all 99].

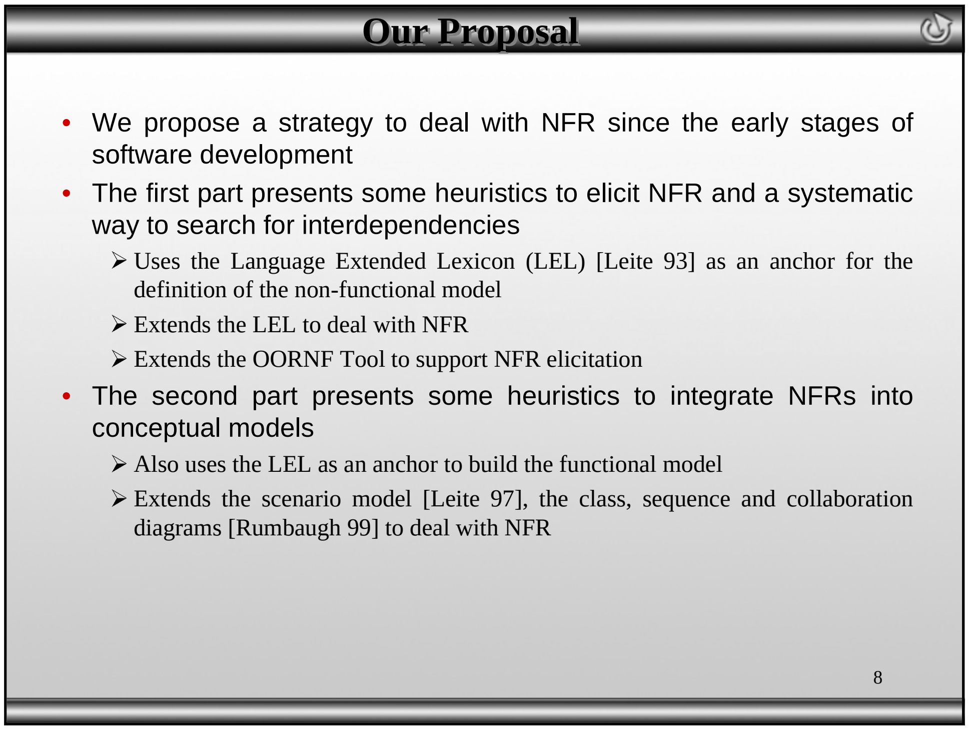
6



Eliciting is not Enough

- **Integrating NFRs into conceptual models can help to get software deployed faster and with lower development costs [Cysneiros 99].**

7

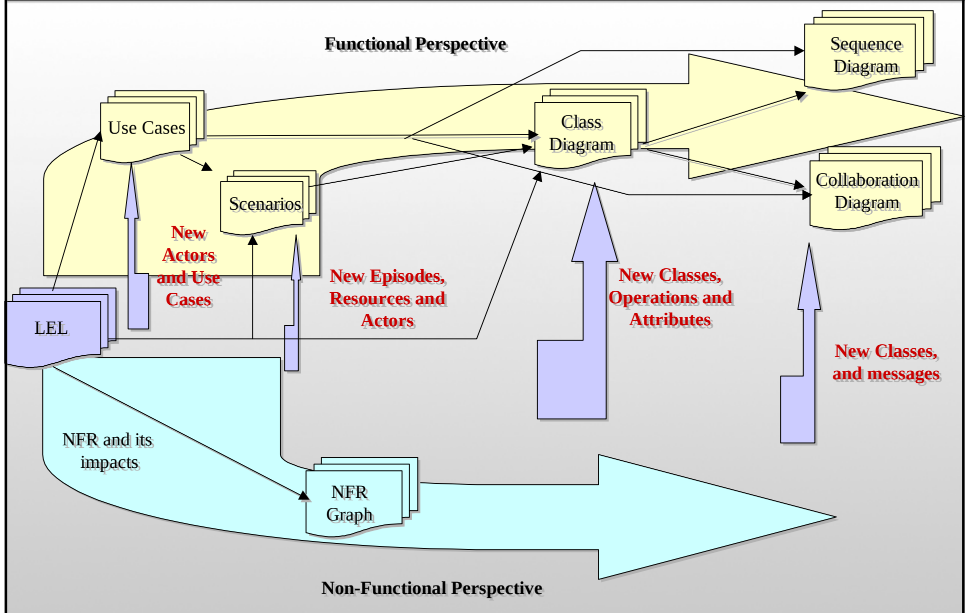


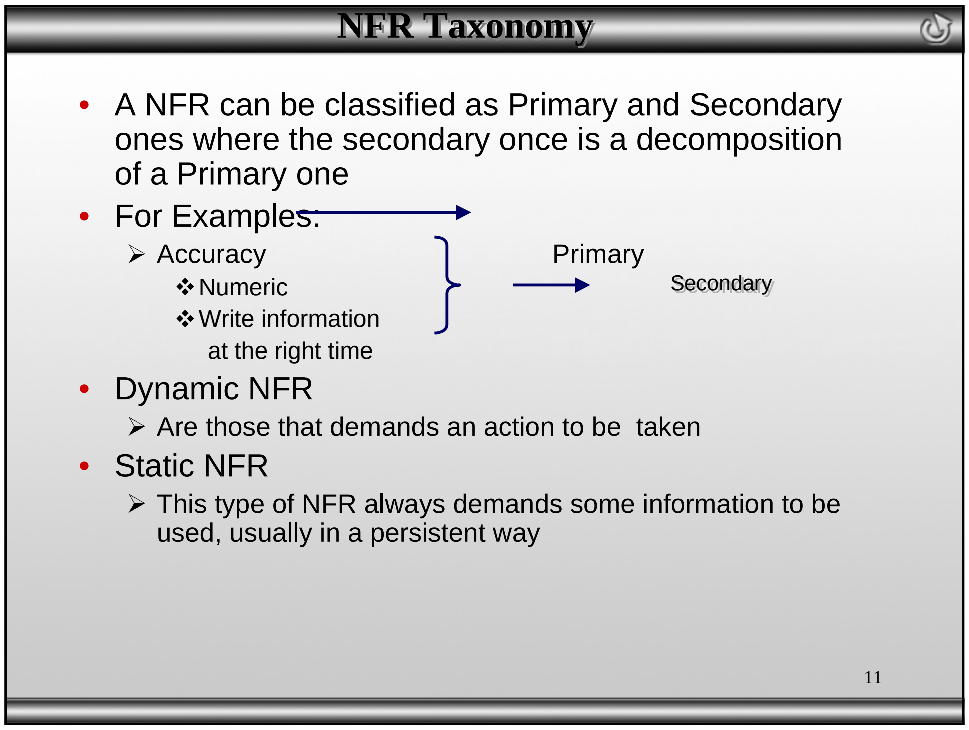
Our Proposal

- We propose a strategy to deal with NFR since the early stages of software development
- The first part presents some heuristics to elicit NFR and a systematic way to search for interdependencies
  - ➢ Uses the Language Extended Lexicon (LEL) [Leite 93] as an anchor for the definition of the non-functional model
  - ➢ Extends the LEL to deal with NFR
  - ➢ Extends the OORNF Tool to support NFR elicitation
- The second part presents some heuristics to integrate NFRs into conceptual models
  - ➢ Also uses the LEL as an anchor to build the functional model
  - ➢ Extends the scenario model [Leite 97], the class, sequence and collaboration diagrams [Rumbaugh 99] to deal with NFR

8

The Proposed Strategy



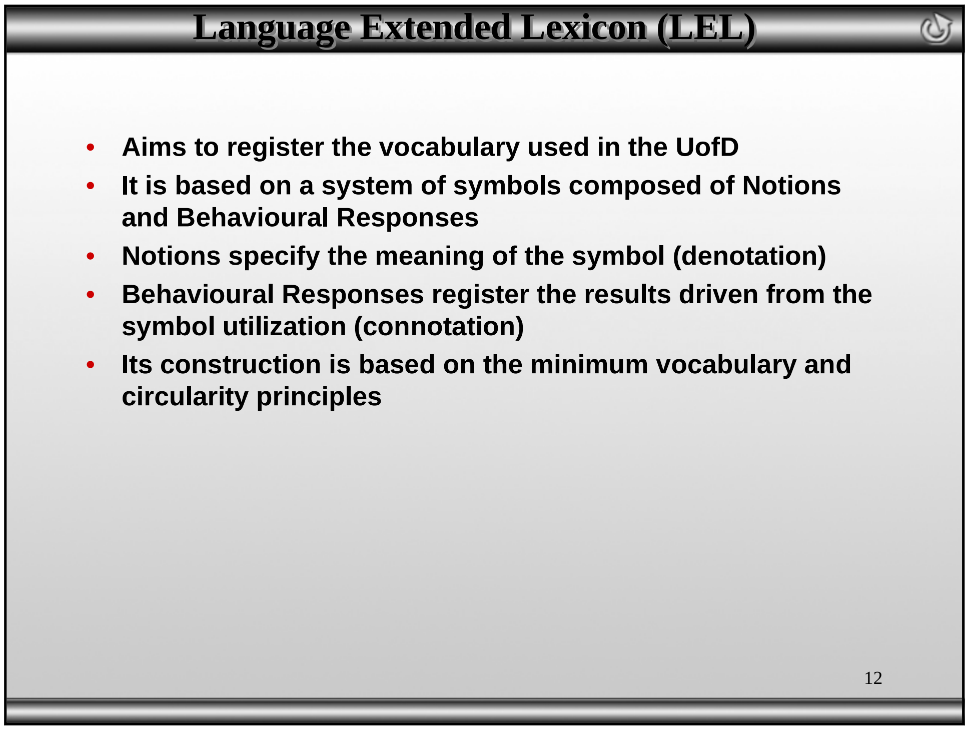


NFR Taxonomy

- A NFR can be classified as Primary and Secondary ones where the secondary once is a decomposition of a Primary one
- For Examples:
  - Accuracy
    - Numeric
    - Write information at the right time
- Dynamic NFR
  - Are those that demands an action to be taken
- Static NFR
  - This type of NFR always demands some information to be used, usually in a persistent way

Primary
Secondary

11



Language Extended Lexicon (LEL)

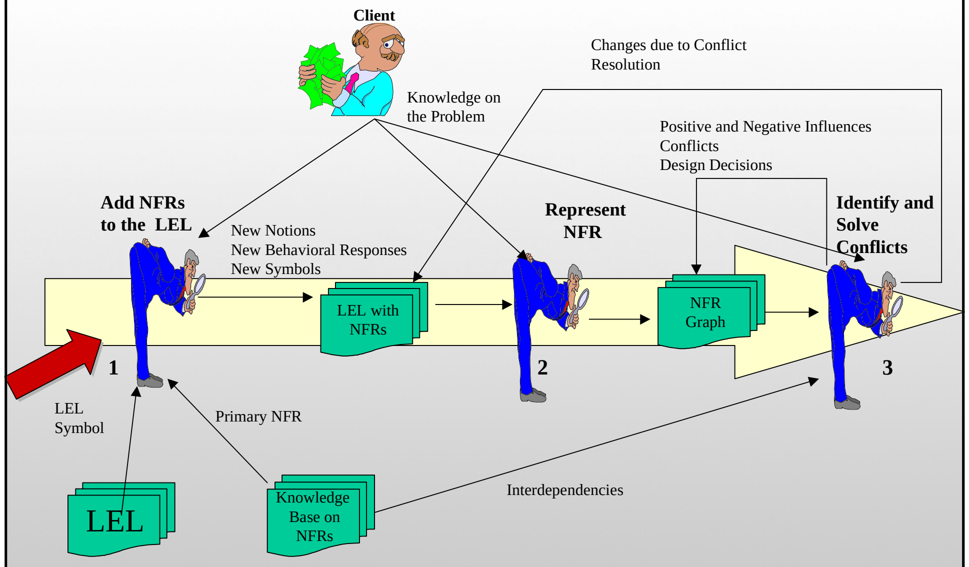
- **Aims to register the vocabulary used in the UofD**
- **It is based on a system of symbols composed of Notions and Behavioural Responses**
- **Notions specify the meaning of the symbol (denotation)**
- **Behavioural Responses register the results driven from the symbol utilization (connotation)**
- **Its construction is based on the minimum vocabulary and circularity principles**

12

LEL – Proposed Extension

- We now register Primary NFR in the notion of the symbols
- We register the fact that a notion or a behavioural response is stated for satisficing an NFR from another symbol (eventually from the same symbol)
- We can now show what notions and behavioural responses are necessary to satisfy an NFR

Building the Non-Functional View

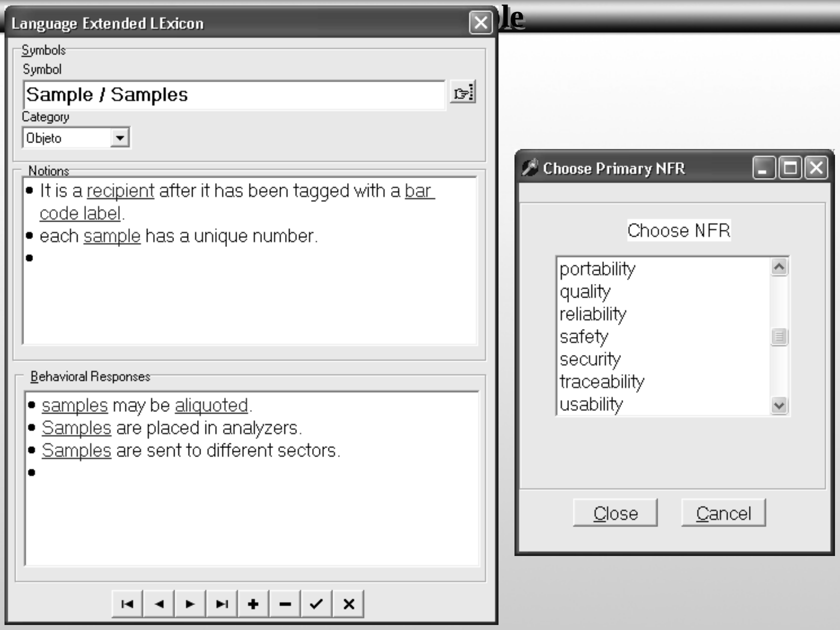


Add NFRs to the LEL

- **To each symbol of the LEL :**
  - Check if any NFR belonging to the Knowledge Base may be necessary in this symbol
  - If it is true, represent it in the Notion
  - Evaluate the possible consequences in this symbol and in other symbols due to NFR satisfaction
  - Establish a dependency link between these notion and behavioural Reponses to this NFR

15

le



Symbol: Sample / Samples

Category: Objeto

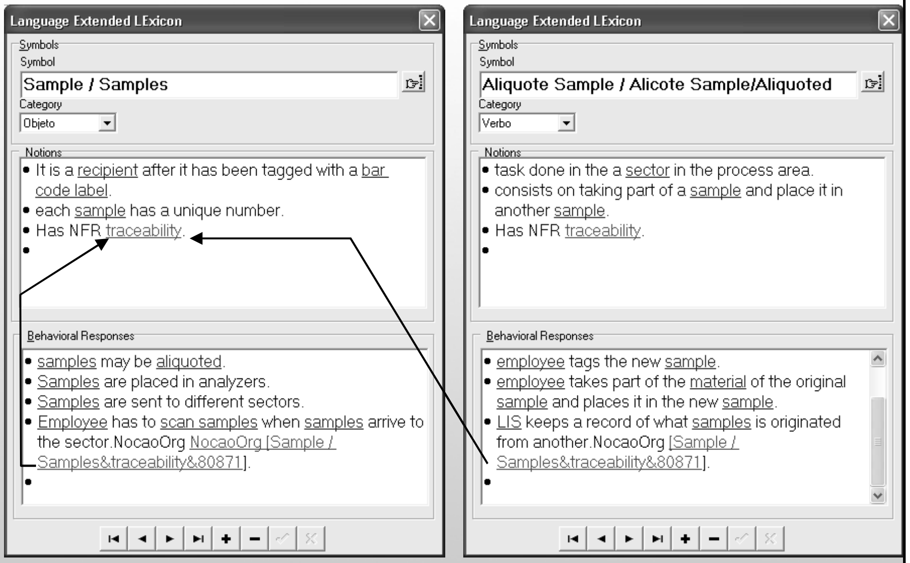
Notions
- It is a recipient after it has been tagged with a bar code label.
- each sample has a unique number.

Behavioral Responses
- samples may be aliquoted.
- Samples are placed in analyzers.
- Samples are sent to different sectors.

portability
quality
reliability
safety
security
traceability
usability

16

Example (Cont.)



Language Extended LExicon

Symbols

Symbol: Sample / Samples

Category: Objeto

Notions
- It is a recipient after it has been tagged with a bar code label.
- each sample has a unique number.
- Has NFR traceability.
-

Behavioral Responses
- samples may be aliquoted.
- Samples are placed in analyzers.
- Samples are sent to different sectors.
- Employee has to scan samples when samples arrive to the sector.NocaoOrg NocaoOrg [Sample / Samples&traceability&80871].
-

Language Extended LExicon

Symbols

Symbol: Aliquote Sample / Alicote Sample/Aliquoted

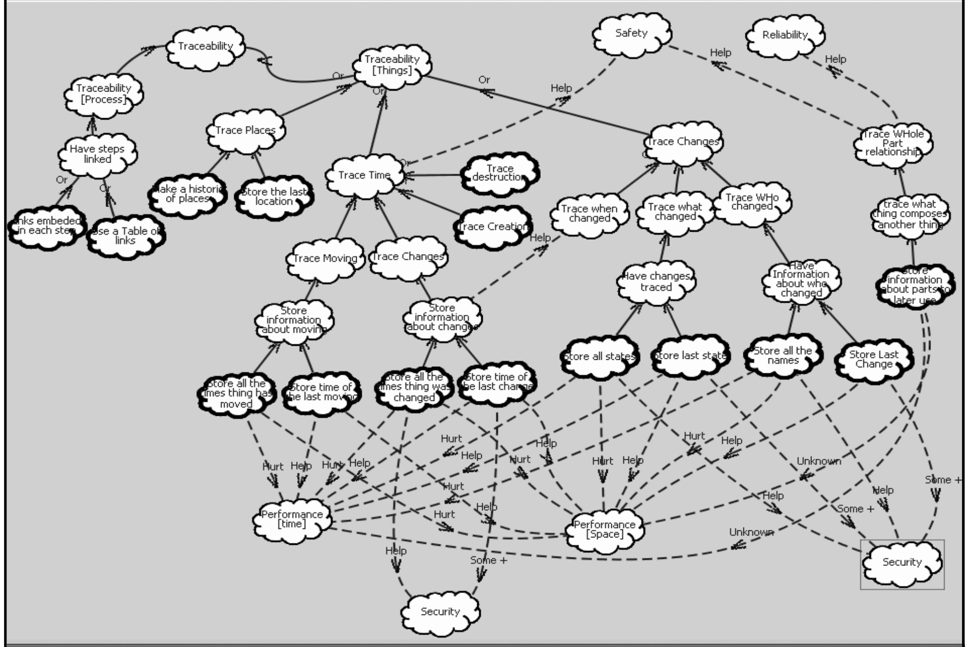
Category: Verbo

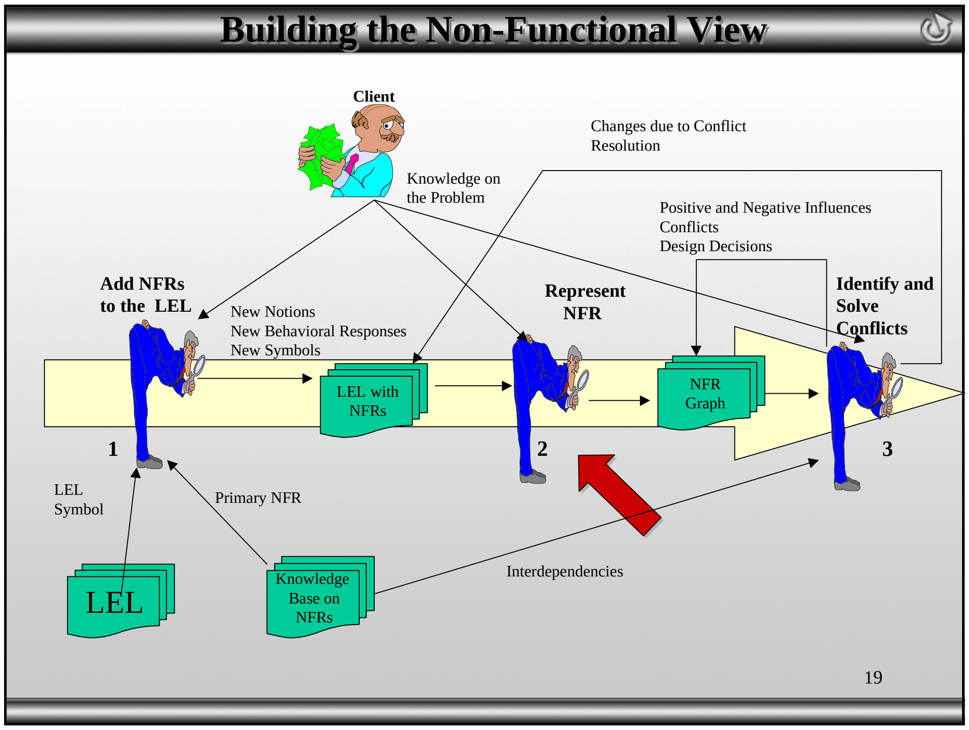
Notions
- task done in the a sector in the process area.
- consists on taking part of a sample and place it in another sample.
- Has NFR traceability.
-

Behavioral Responses
- employee tags the new sample.
- employee takes part of the material of the original sample and places it in the new sample.
- LIS keeps a record of what samples is originated from another.NocaoOrg [Sample / Samples&traceability&80871].
-

17

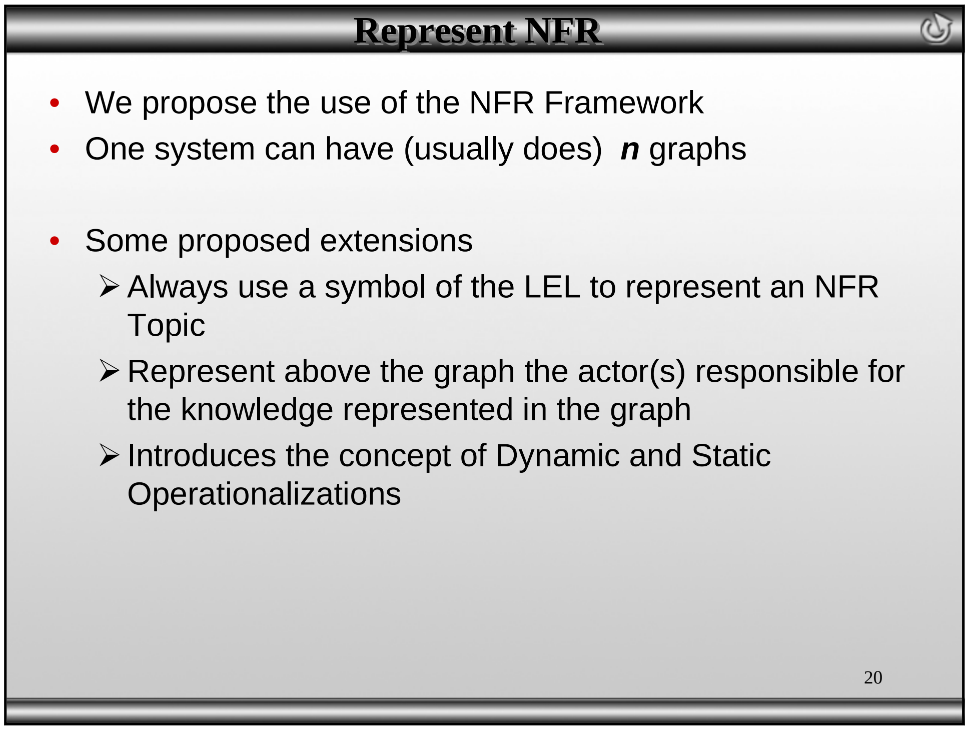
Using Catalogues





Building the Non-Functional View

19

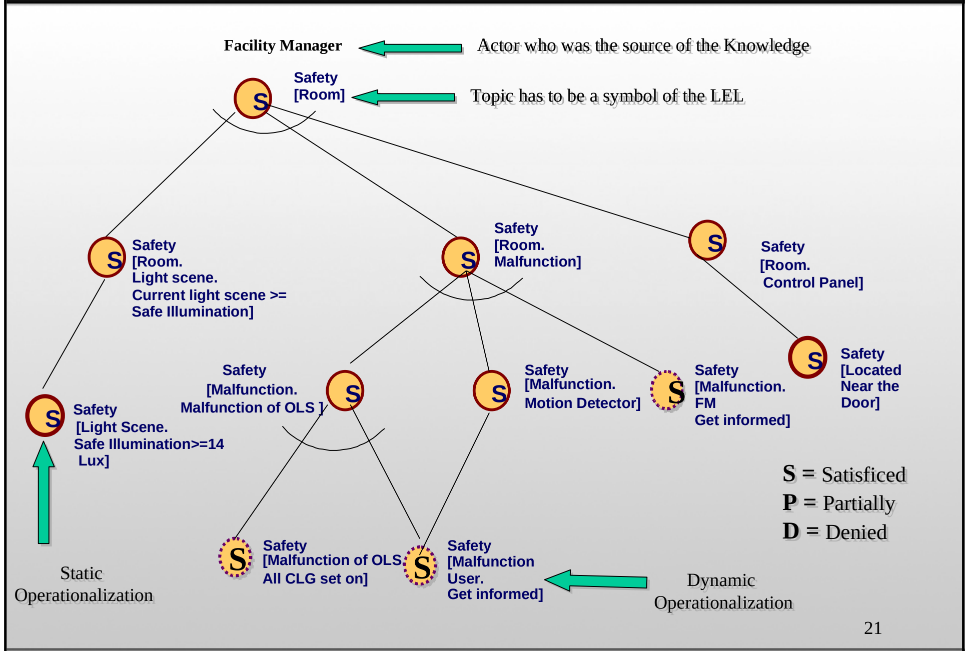


Represent NFR

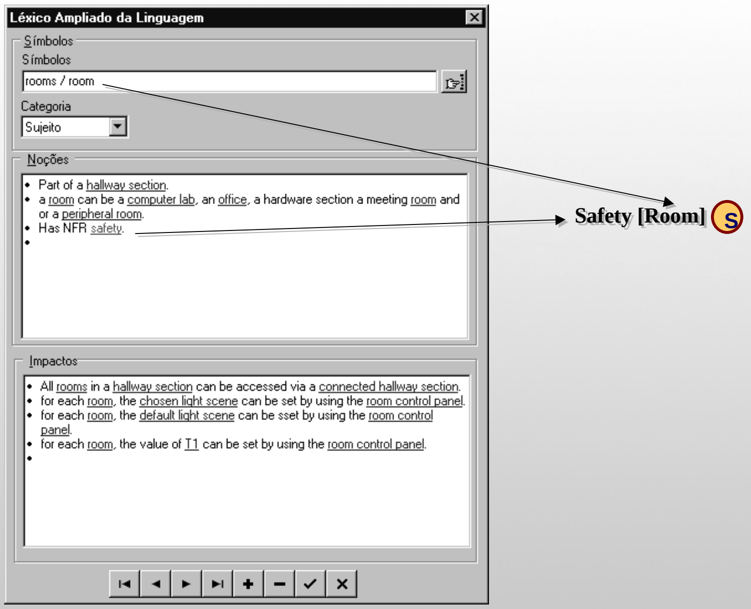
- We propose the use of the NFR Framework
- One system can have (usually does) ***n*** graphs

- Some proposed extensions
  - ➢ Always use a symbol of the LEL to represent an NFR Topic
  - ➢ Represent above the graph the actor(s) responsible for the knowledge represented in the graph
  - ➢ Introduces the concept of Dynamic and Static Operationalizations

20


Facility Manager
Actor who was the source of the Knowledge
Safety [Room]
Topic has to be a symbol of the LEL
Safety [Room. Light scene. Current light scene >= Safe Illumination]
Safety [Room. Malfunction]
Safety [Room. Control Panel]
Safety [Light Scene. Safe Illumination>=14 Lux]
Safety [Malfunction. Malfunction of OLS]
Safety [Malfunction. Motion Detector]
Safety [Malfunction. FM Get informed]
Safety [Located Near the Door]
S = Satisficed
P = Partially
D = Denied
Safety [Malfunction of OLS. All CLG set on]
Safety [Malfunction User. Get informed]
Static Operationalization
Dynamic Operationalization

How to Create a Graph

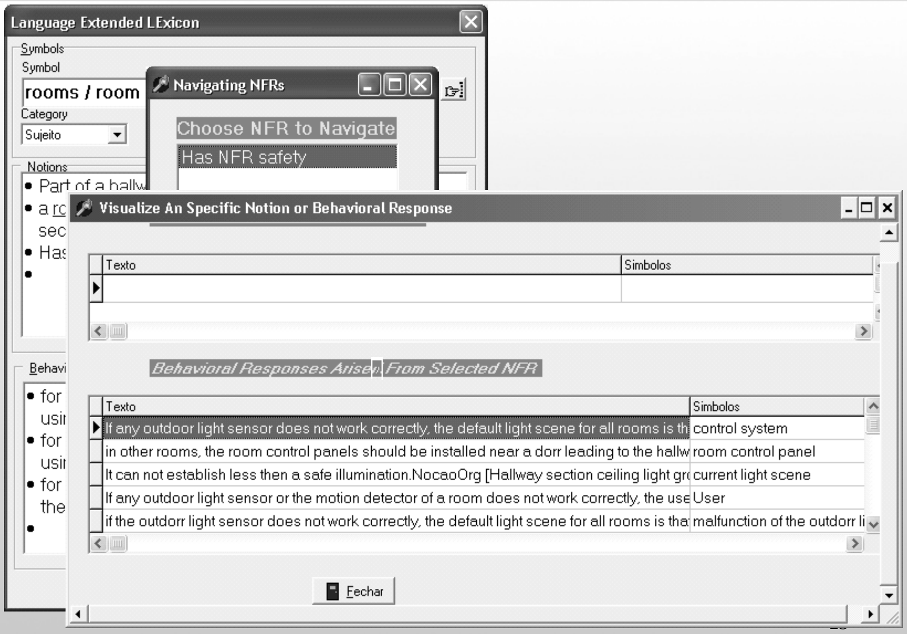


rooms / room

Sujeito

- Part of a hallway section.
- a room can be a computer lab, an office, a hardware section a meeting room and or a peripheral room.
- Has NFR safety.

- All rooms in a hallway section can be accessed via a connected hallway section.
- for each room, the chosen light scene can be set by using the room control panel.
- for each room, the default light scene can be sset by using the room control panel.
- for each room, the value of T1 can be set by using the room control panel.


Language Extended LExicon

Symbols

Symbol: rooms / room

Category: Sujeito

Notions

- Part of a hallw
- a rc sec
- Has
-

Behavi

- for usi
- for usi
- for the
-

Navigating NFRs

Choose NFR to Navigate

Has NFR safety

Visualize An Specific Notion or Behavioral Response

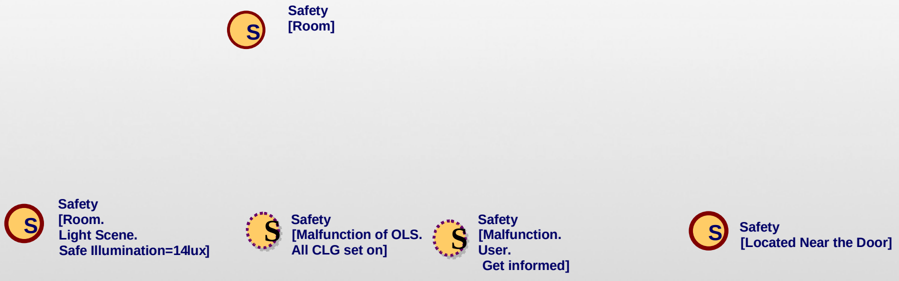
Texto	Simbolos

Behavioral Responses Arise From Selected NFR

Texto	Simbolos
If any outdoor light sensor does not work correctly, the default light scene for all rooms is th	control system
in other rooms, the room control panels should be installed near a dorr leading to the hallw	room control panel
It can not establish less then a safe illumination.NocaoOrg [Hallway section ceiling light gr	current light scene
If any outdoor light sensor or the motion detector of a room does not work correctly, the use	User
if the outdorr light sensor does not work correctly, the default light scene for all rooms is tha	malfunction of the outdorr li

Fechar

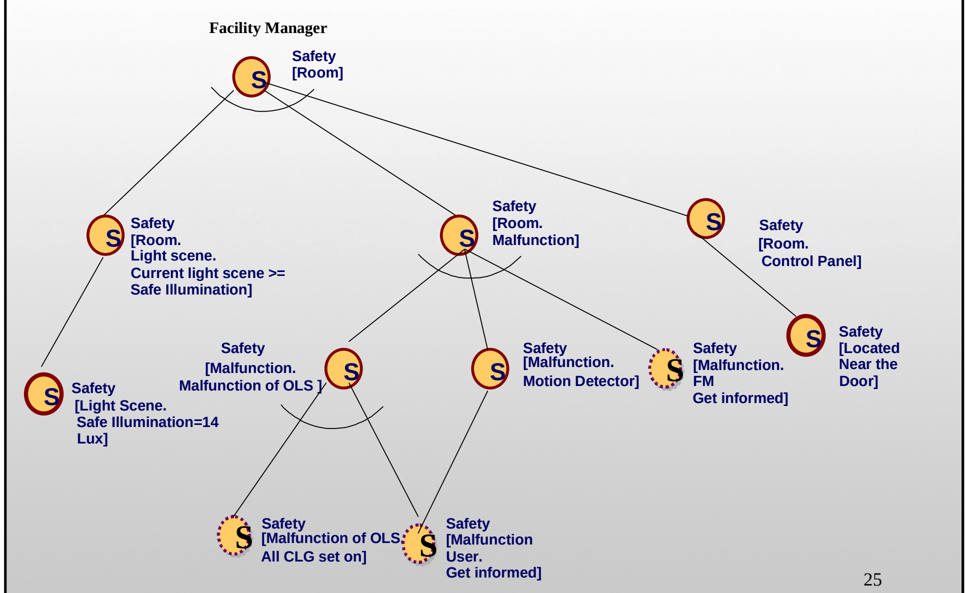
A First Decomposition



Safety [Room]
Safety [Room. Light Scene. Safe Illumination=14lux]
Safety [Malfunction of OLS. All CLG set on]
Safety [Malfunction. User. Get informed]
Safety [Located Near the Door]

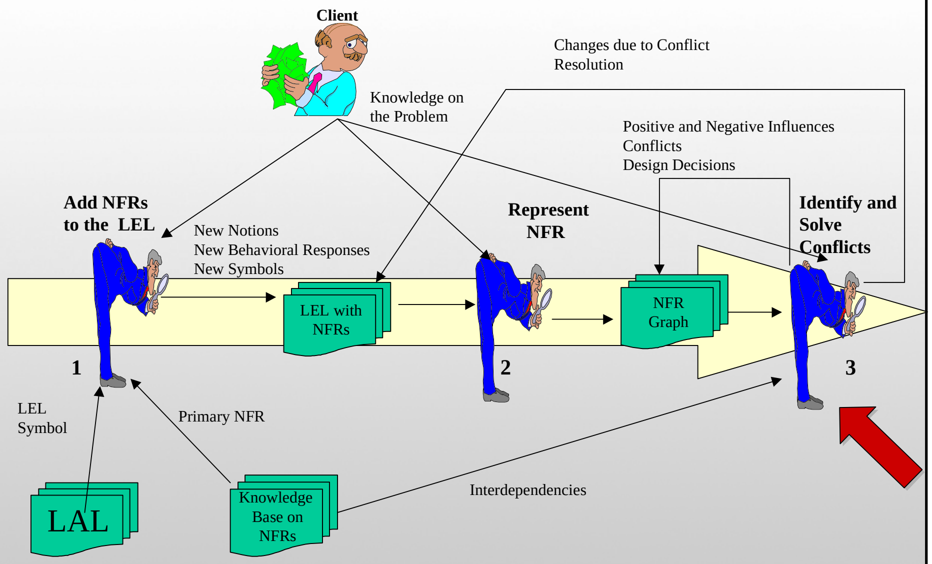
24

Final Version


Facility Manager
Safety [Room]
Safety [Room. Light scene. Current light scene >= Safe Illumination]
Safety [Room. Malfunction]
Safety [Room. Control Panel]
Safety [Light Scene. Safe Illumination=14 Lux]
Safety [Malfunction. Malfunction of OLS]
Safety [Malfunction. Motion Detector]
Safety [Malfunction. FM Get informed]
Safety [Located Near the Door]
Safety [Malfunction of OLS. All CLG set on]
Safety [Malfunction User. Get informed]
S

25

Building the Non-Functional View



26

Identify and Solve Conflicts

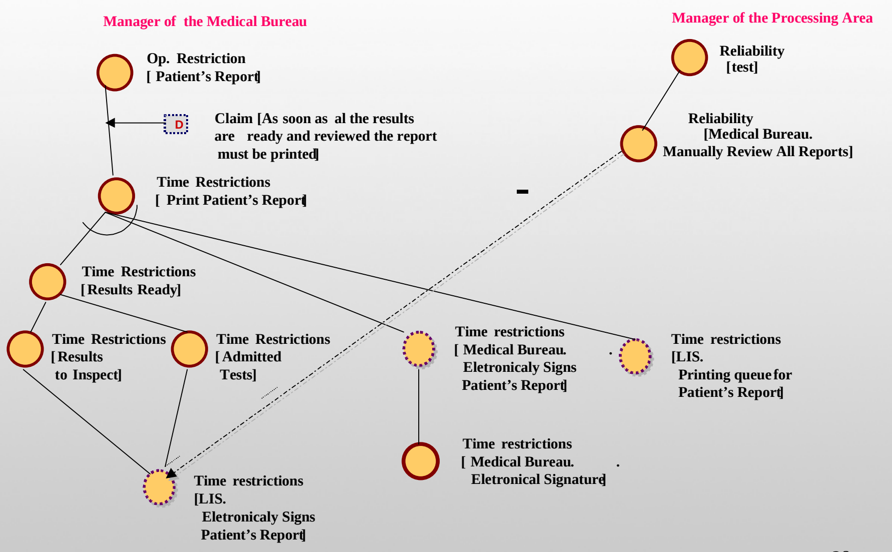
- Compare graphs with the same type (ex: Performance)
- Compare graphs with conflicting types (Ex: Security and Performance or Usability)
- Pair wise graphs

For each of the above heuristics:

- Register positive and negative interdependencies
- Try to solve possible conflicts (negative interdependencies)

27

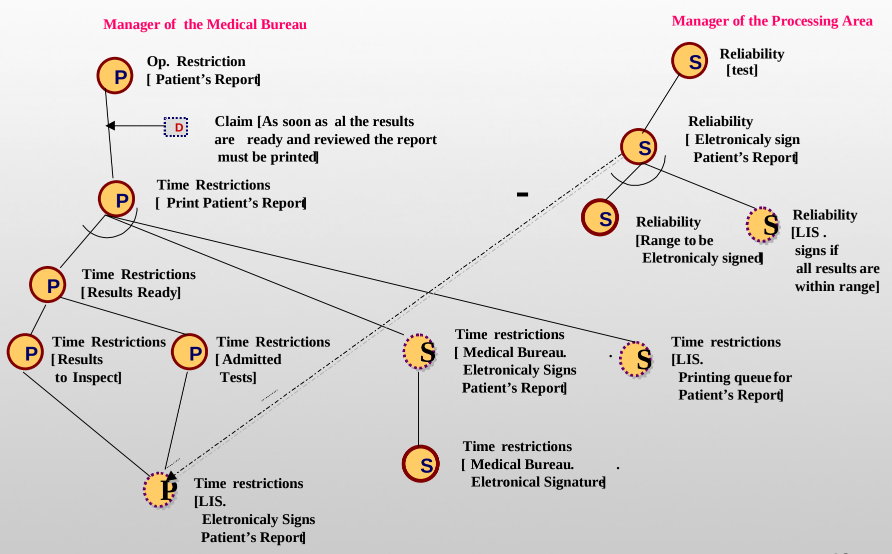
Example



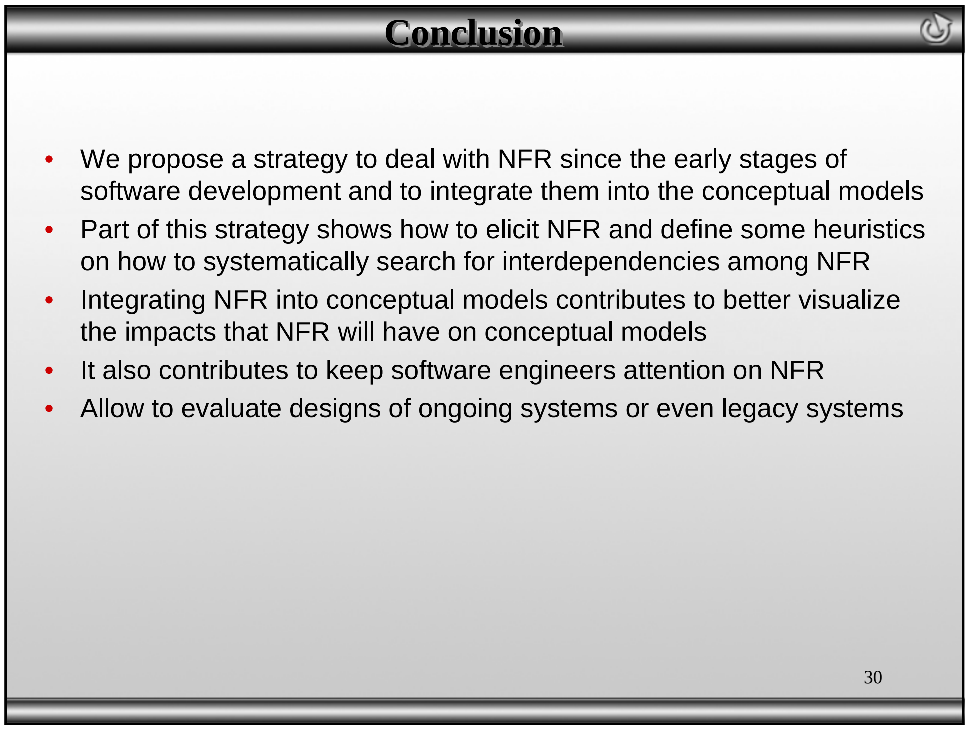
Manager of the Medical Bureau
Manager of the Processing Area
Op. Restriction [Patient's Report]
D
Claim [As soon as al the results are ready and reviewed the report must be printed]
Time Restrictions [Print Patient's Report]
Reliability [test]
Reliability [Medical Bureau. Manually Review All Reports]
-
Time Restrictions [Results Ready]
Time Restrictions [Results to Inspect]
Time Restrictions [Admitted Tests]
Time restrictions [Medical Bureau. Eletronicaly Signs Patient's Report]
Time restrictions [LIS. Printing queue for Patient's Report]
Time restrictions [Medical Bureau. Eletronical Signature]
Time restrictions [LIS. Eletronicaly Signs Patient's Report]

28

Example


Manager of the Medical Bureau
Manager of the Processing Area
P Op. Restriction [Patient's Report]
D Claim [As soon as al the results are ready and reviewed the report must be printed]
P Time Restrictions [Print Patient's Report]
P Time Restrictions [Results Ready]
P Time Restrictions [Results to Inspect]
P Time Restrictions [Admitted Tests]
P Time restrictions [LIS. Eletronicaly Signs Patient's Report]
S Reliability [test]
S Reliability [Eletronicaly sign Patient's Report]
S Reliability [Range to be Eletronicaly signed]
S Reliability [LIS . signs if all results are within range]
S Time restrictions [Medical Bureau. Eletronicaly Signs Patient's Report]
S Time restrictions [LIS. Printing queue for Patient's Report]
S Time restrictions [Medical Bureau. Eletronical Signature]
-

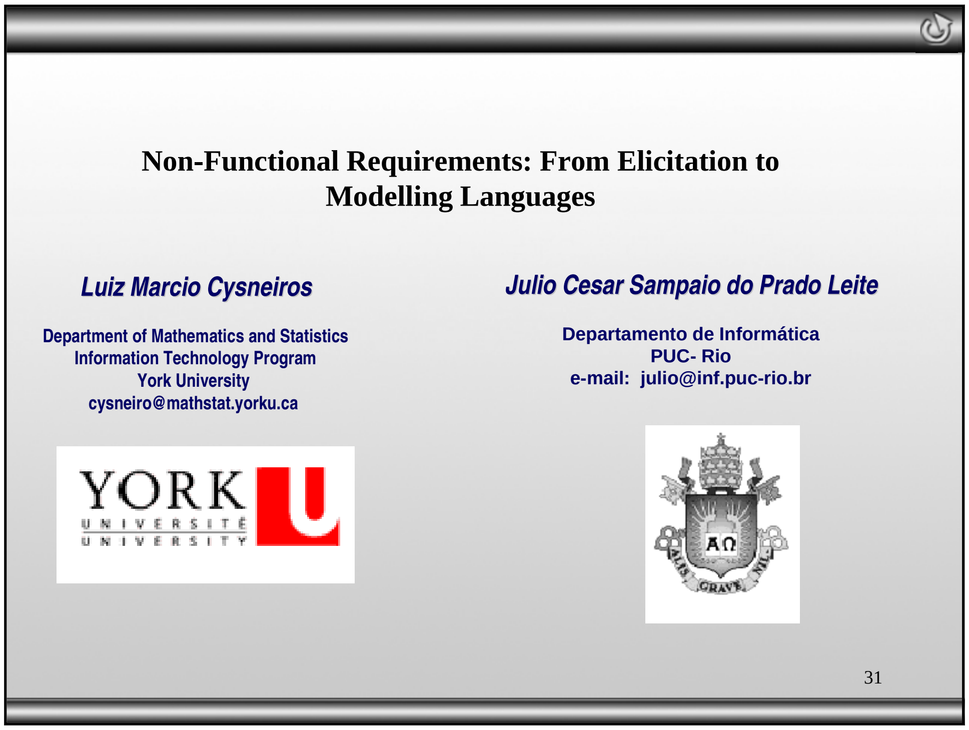
29



Conclusion

- We propose a strategy to deal with NFR since the early stages of software development and to integrate them into the conceptual models
- Part of this strategy shows how to elicit NFR and define some heuristics on how to systematically search for interdependencies among NFR
- Integrating NFR into conceptual models contributes to better visualize the impacts that NFR will have on conceptual models
- It also contributes to keep software engineers attention on NFR
- Allow to evaluate designs of ongoing systems or even legacy systems

30



Non-Functional Requirements: From Elicitation to Modelling Languages

Luiz Marcio Cysneiros

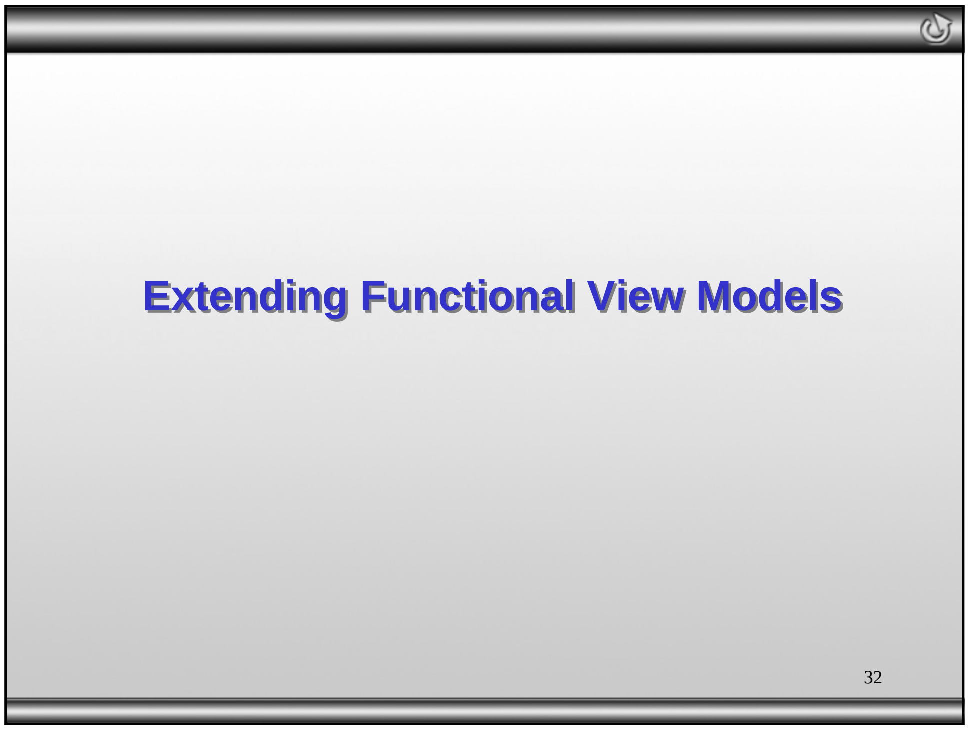
Department of Mathematics and Statistics
Information Technology Program
York University
cysneiro@mathstat.yorku.ca

Julio Cesar Sampaio do Prado Leite

Departamento de Informática
PUC- Rio
e-mail: julio@inf.puc-rio.br

YORK U
UNIVERSITÉ
UNIVERSITY

31



Extending Functional View Models

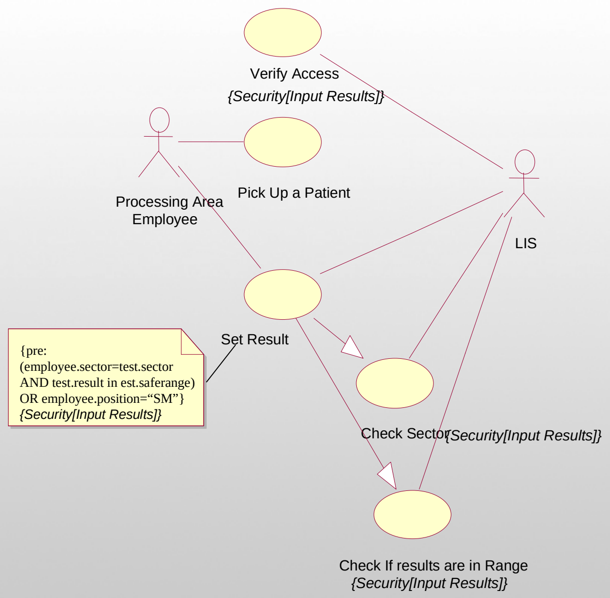
32

Extending Use Cases

- Every use case or actor included, due to NFR satisfaction, must be followed by an expression using the pattern: ***{NFR_Type[NFR_topic]}*.**
- Represent Special Conditions that must prevail to an Use Case as a note linked to this Use Case, using whenever possible OCL to describe these conditions
  - Ex: In a Light Control System
  - The Use Case Turn Off Lights after T3 Sec must have a special condition saying that this will only happen if the room is empty
  - We should them represent it as: Pre: room.number_people=0

33

Example


Verify Access
{Security[Input Results]}
Processing Area Employee
Pick Up a Patient
LIS
Set Result
Check Sector*{Security[Input Results]}*
Check If results are in Range
{Security[Input Results]}

{pre:
(employee.sector=test.sector
AND test.result in est.saferange)
OR employee.position="SM"}
{Security[Input Results]}

34

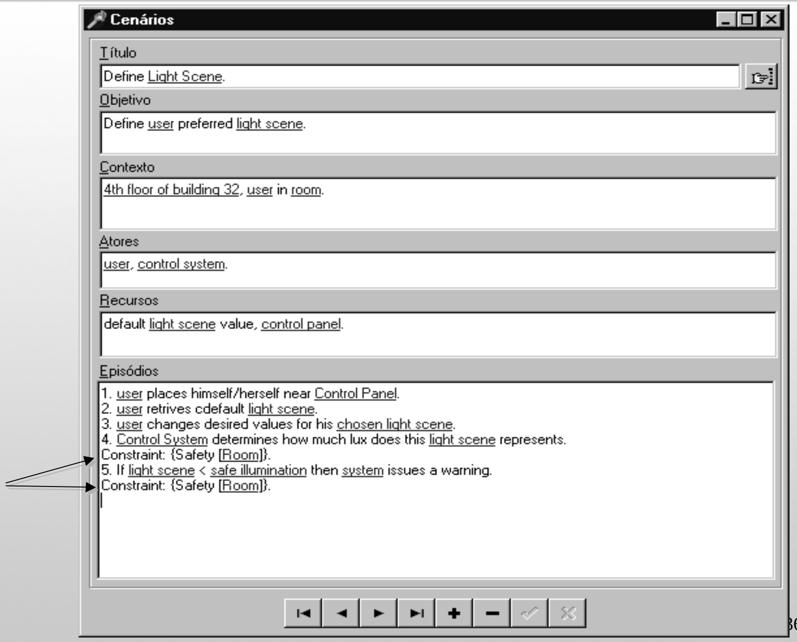
Extending Scenarios [Leite 97]

- Every change in a scenario due to an NFR satisficing must be followed by the expression :

  Constraint: {NFR[Topic]}

- Reason: Traceability between models

35



Título

Define Light Scene.

Objetivo

Define user preferred light scene.

Contexto

4th floor of building 32, user in room.

Atores

user, control system.

Recursos

default light scene value, control panel.

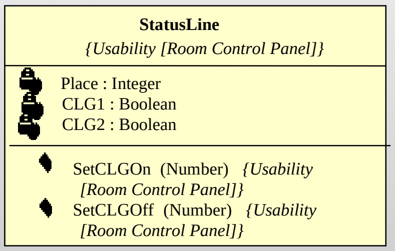
Episódios

1. user places himself/herself near Control Panel.
2. user retrives cdefault light scene.
3. user changes desired values for his chosen light scene.
4. Control System determines how much lux does this light scene represents.
Constraint: {Safety [Room]}.
5. If light scene < safe illumination then system issues a warning.
Constraint: {Safety [Room]}.

36

Extending the Class Diagram

- Every class will be named using a symbol of the LEL
- Classes created due to an NFR satisficing will be follwed by the same traceability pattern used in scenarios.

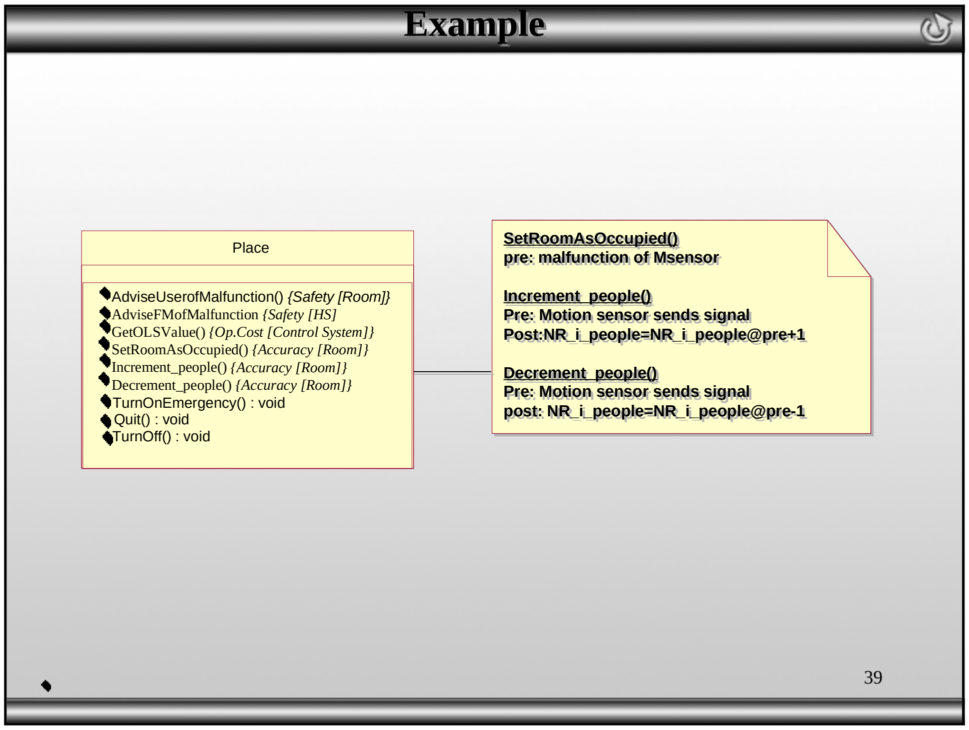

StatusLine
{Usability [Room Control Panel]}
Place : Integer
CLG1 : Boolean
CLG2 : Boolean
SetCLGOn (Number) *{Usability [Room Control Panel]}*
SetCLGOff (Number) *{Usability [Room Control Panel]}*

37

Extending the Class Diagram (cont.)

- Operations that are in the class due to an NFR satisficing will **always** be followed by the same kind of expression used in scenarios : **{NFR[Topic]}**
- We may have to represent special conditions that holds for an operation. These conditions will be represented between {} and must use, whenever possible, expressions written using OCL
- If these conditions impose a significant loss of space, we might represent them inside a note

38



Example

Place

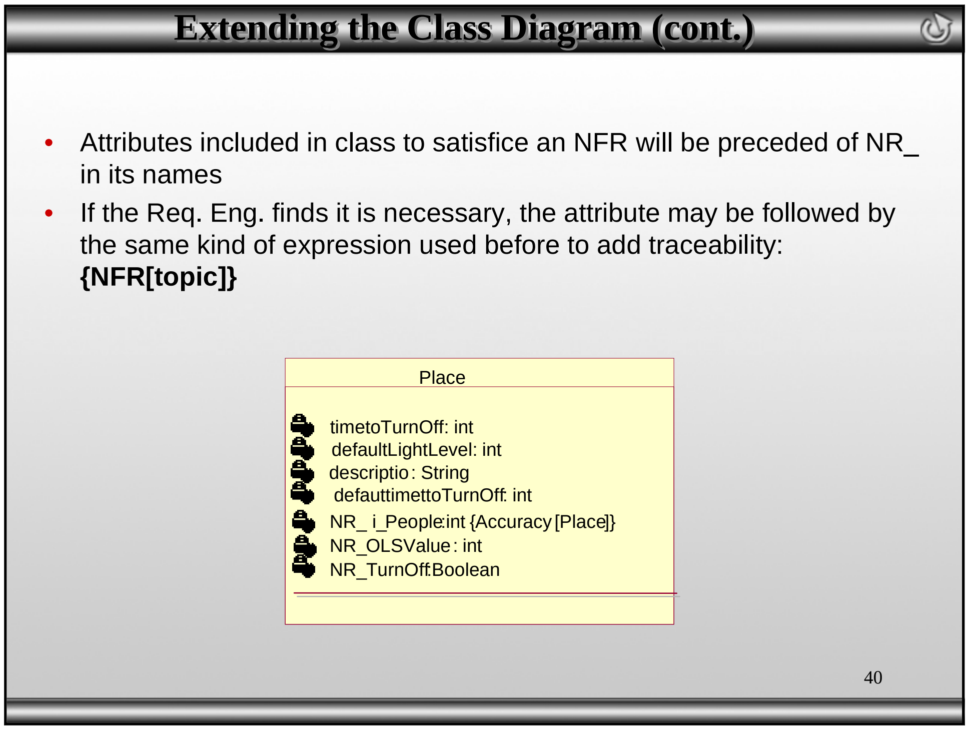
AdviseUserofMalfunction() *{Safety [Room]}*
AdviseFMofMalfunction *{Safety [HS]*
GetOLSValue() *{Op.Cost [Control System]}*
SetRoomAsOccupied() *{Accuracy [Room]}*
Increment_people() *{Accuracy [Room]}*
Decrement_people() *{Accuracy [Room]}*
TurnOnEmergency() : void
Quit() : void
TurnOff() : void

SetRoomAsOccupied()
pre: malfunction of Msensor

Increment_people()
Pre: Motion sensor sends signal
Post:NR_i_people=NR_i_people@pre+1

Decrement_people()
Pre: Motion sensor sends signal
post: NR_i_people=NR_i_people@pre-1

39



Extending the Class Diagram (cont.)

- Attributes included in class to satisfice an NFR will be preceded of NR_ in its names
- If the Req. Eng. finds it is necessary, the attribute may be followed by the same kind of expression used before to add traceability: **{NFR[topic]}**

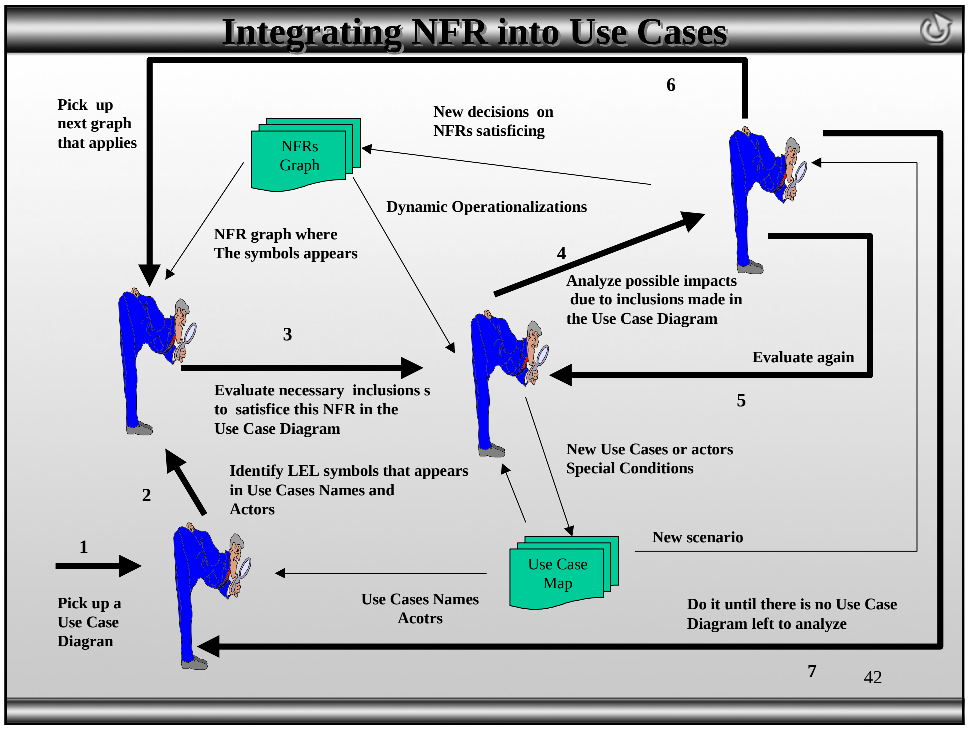
Place

timetoTurnOff: int
defaultLightLevel: int
descriptio: String
defauttimettoTurnOff: int
NR_ i_People:int {Accuracy [Place]}
NR_OLSValue: int
NR_TurnOff:Boolean

40

The Proposed Integration Method

41

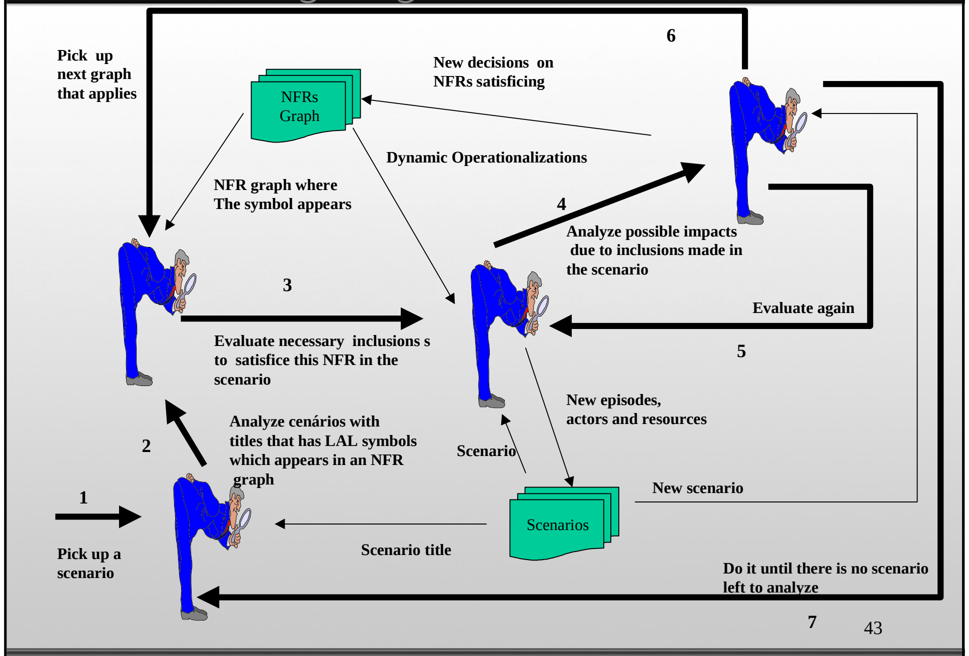


Integrating NFR into Use Cases

Pick up next graph that applies
NFRs Graph
New decisions on NFRs satisficing
6
Dynamic Operationalizations
NFR graph where The symbols appears
4
Analyze possible impacts due to inclusions made in the Use Case Diagram
3
Evaluate again
Evaluate necessary inclusions s to satisfice this NFR in the Use Case Diagram
5
New Use Cases or actors Special Conditions
Identify LEL symbols that appears in Use Cases Names and Actors
2
1
New scenario
Use Case Map
Pick up a Use Case Diagran
Use Cases Names Acotrs
Do it until there is no Use Case Diagram left to analyze
7

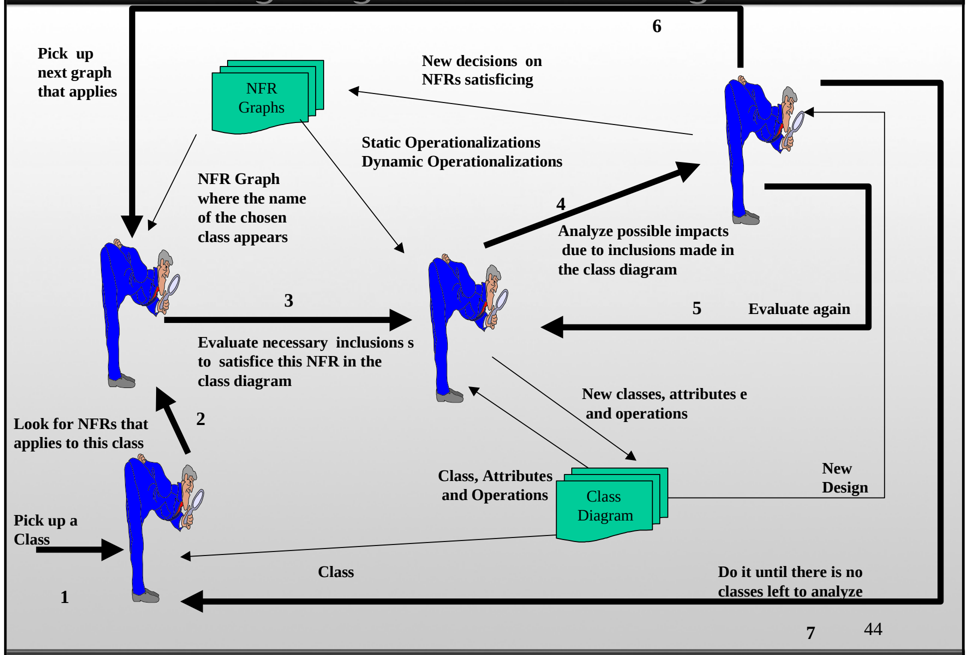
42

Integrating NFR into Scenarios


Pick up next graph that applies
6
New decisions on NFRs satisficing
NFRs Graph
Dynamic Operationalizations
NFR graph where The symbol appears
4
Analyze possible impacts due to inclusions made in the scenario
3
Evaluate again
Evaluate necessary inclusions s to satisfice this NFR in the scenario
5
New episodes, actors and resources
Analyze cenários with titles that has LAL symbols which appears in an NFR graph
2
Scenario
1
New scenario
Scenarios
Pick up a scenario
Scenario title
Do it until there is no scenario left to analyze
7

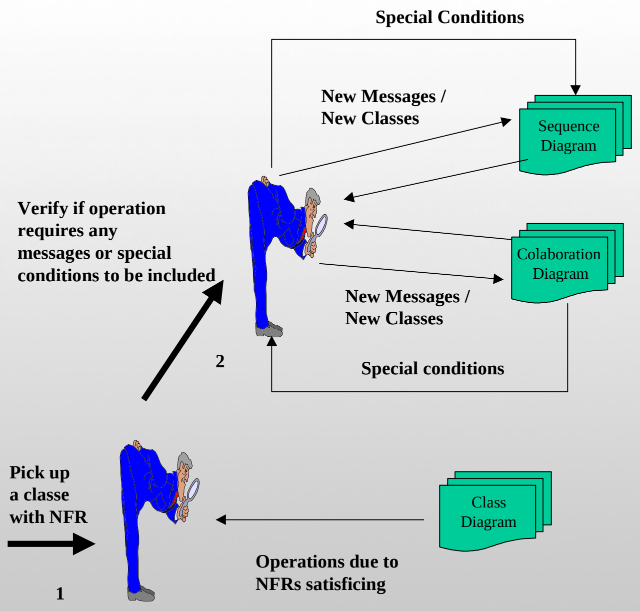
43

Integrating NFR into Class Diagram


Pick up next graph that applies
6
New decisions on NFRs satisficing
NFR Graphs
Static Operationalizations
Dynamic Operationalizations
NFR Graph where the name of the chosen class appears
4
Analyze possible impacts due to inclusions made in the class diagram
3
5
Evaluate again
Evaluate necessary inclusions s to satisfice this NFR in the class diagram
New classes, attributes e and operations
Look for NFRs that applies to this class
2
Class, Attributes and Operations
Class Diagram
New Design
Pick up a Class
Class
1
Do it until there is no classes left to analyze
7

44

Sequence and Collaboration Diagrams


Special Conditions
New Messages / New Classes
Sequence Diagram
Verify if operation requires any messages or special conditions to be included
Colaboration Diagram
New Messages / New Classes
2
Special conditions
Pick up a classe with NFR
Class Diagram
Operations due to NFRs satisficing
1

45

Examples From a Case Study

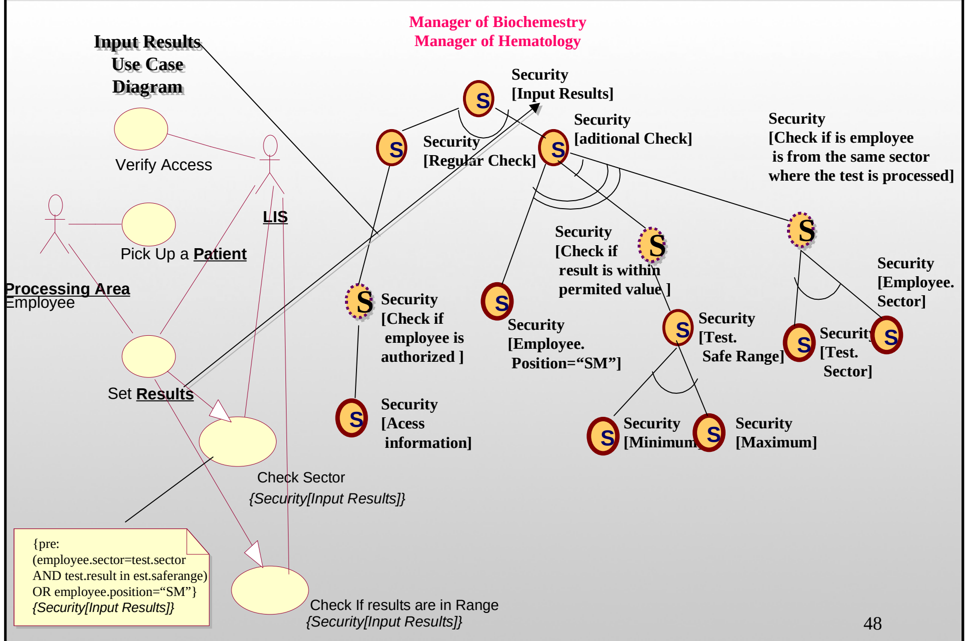
46

Strategy Validation

- Strongly based on the replication project principle [Basili 96]
- 3 different case studies
- 2 in vitro (Controlled Environment)
- 1 in vivo (Real Application Environment)
- In all the 3 cases we used conceptual models built by the other teams and we applied the proposed strategy
  - We built the Non-Functional View
  - We integrated the NFRs from this view into the conceptual models developed by the other teams

47

Integrating NFR into Use Cases


Input Results Use Case Diagram
Manager of Biochemestry
Manager of Hematology
Verify Access
LIS
Pick Up a Patient
Processing Area Employee
Set Results
Check Sector {Security[Input Results]}
Check If results are in Range {Security[Input Results]}
{pre: (employee.sector=test.sector AND test.result in est.saferange) OR employee.position="SM"} {Security[Input Results]}
S
Security [Input Results]
Security [Regular Check]
Security [aditional Check]
Security [Check if is employee is from the same sector where the test is processed]
Security [Check if employee is authorized]
Security [Acess information]
Security [Check if result is within permited value]
Security [Employee. Position="SM"]
Security [Test. Safe Range]
Security [Minimum]
Security [Maximum]
Security [Employee. Sector]
Security [Test. Sector]

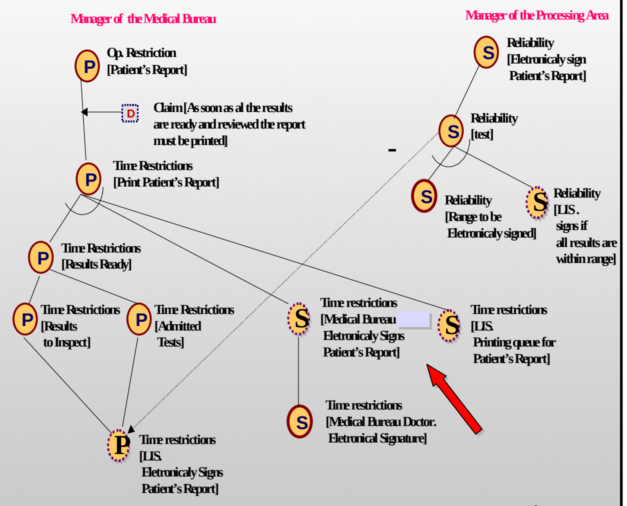
48

Using the Strategy on the Class “Medical Bureau”

Medical Bureau

NR_EletSignature {Op. Restriction [Patient’s Report] }

ReviewPatientReport()
PrintDelayedReport()
PrintPromissedReport()
GetSignature {Op. Restriction [Patient’s Report]}
SetSignature {Op. Restriction [Patient’s Report]}

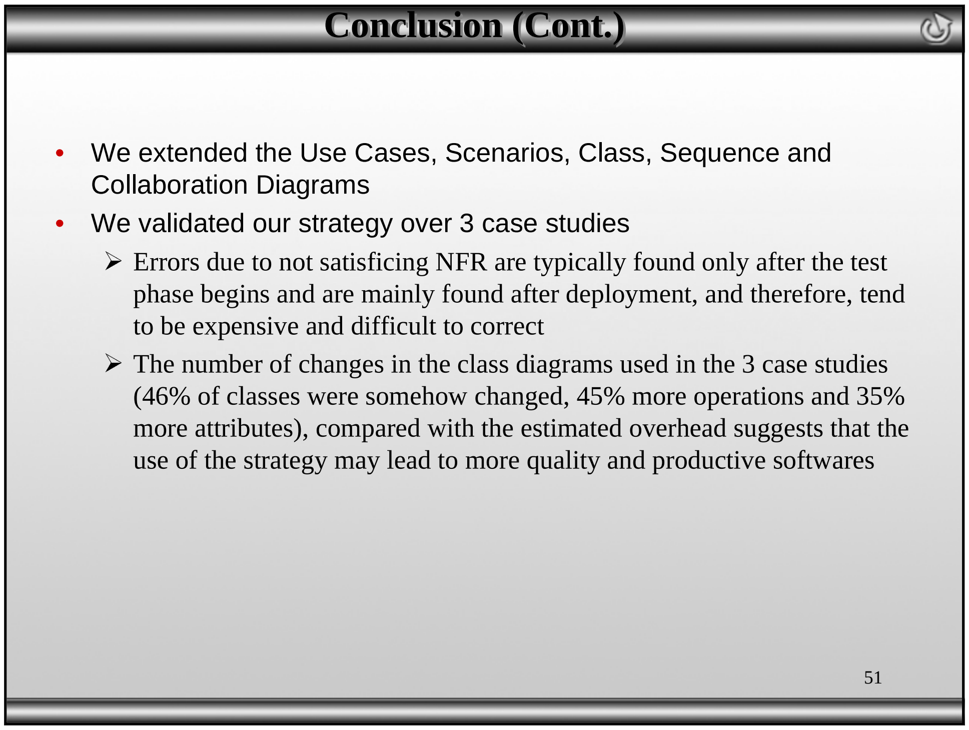

Manager of the Medical Bureau
Manager of the Processing Area
P
Op. Restriction [Patient’s Report]
D
Claim [As soon as al the results are ready and reviewed the report must be printed]
P
Time Restrictions [Print Patient’s Report]
P
Time Restrictions [Results Ready]
P
Time Restrictions [Results to Inspect]
P
Time Restrictions [Admitted Tests]
P
Time restrictions [LIS. Eletronicaly Signs Patient’s Report]
S
Reliability [Eletronicaly sign Patient’s Report]
S
Reliability [test]
-
S
Reliability [Range to be Eletronicaly signed]
S
Reliability [LIS . signs if all results are within range]
S
Time restrictions [Medical Bureau Eletronicaly Signs Patient’s Report]
S
Time restrictions [LIS. Printing queue for Patient’s Report]
S
Time restrictions [Medical Bureau Doctor. Eletronical Signature]

49

Conclusion

- We propose a strategy to deal with NFR since the early stages of software development and to integrate them into the conceptual models
- Part of this strategy shows how to elicit NFR and define some heuristics on how to systematically search for interdependencies among NFR
- Integrating NFR into conceptual models contributes to better visualize the impacts that NFR will have on conceptual models
- It also contributes to keep software engineers attention on NFR
- Allow to evaluate designs of ongoing systems or even legacy systems

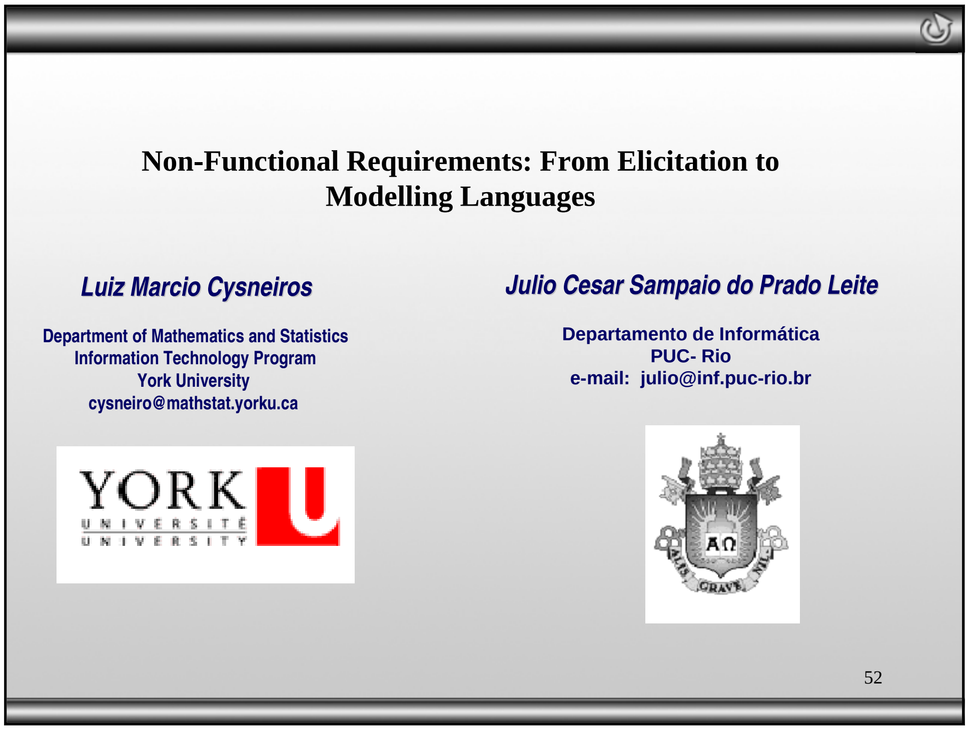
50



Conclusion (Cont.)

- We extended the Use Cases, Scenarios, Class, Sequence and Collaboration Diagrams
- We validated our strategy over 3 case studies
  - Errors due to not satisficing NFR are typically found only after the test phase begins and are mainly found after deployment, and therefore, tend to be expensive and difficult to correct
  - The number of changes in the class diagrams used in the 3 case studies (46% of classes were somehow changed, 45% more operations and 35% more attributes), compared with the estimated overhead suggests that the use of the strategy may lead to more quality and productive softwares

51



Non-Functional Requirements: From Elicitation to Modelling Languages

Luiz Marcio Cysneiros

Department of Mathematics and Statistics
Information Technology Program
York University
cysneiro@mathstat.yorku.ca

Julio Cesar Sampaio do Prado Leite

Departamento de Informática
PUC- Rio
e-mail: julio@inf.puc-rio.br

52

References

- [Basili 91] Basili, V. and Musa, J. "*The Future Engineering of Software a Management Perspective",* IEEE Computer 24 1991 pp:90-96
- [Boehm 76] Boehm, B., Brown, J.R. and Lipow, M. *"Quantitative EVALUATION OF Software Quality"* in Proc. of 2nd International Conference on Soft. Eng. San Francisco, oct 1976, pp:592-605.
- [Boehm78] Boehm, B. *Characteristics of Software Quality*. North Holland Press, 1978
- [Boehm96] Boehm, Barry e In, Hoh. *Identifying Quality-Requirement Conflicts*. IEEE Software, March 1996, pp. 25-35
- [Breitman et al 99] Breitman,K. K., Leite J.C.S.P. e Finkelstein Anthony. *The World's Stage: A Survey on Requirements Engineering Using a Real-Life Case Study*. Journal of the Brazilian Computer Society No 1 Vol. 6 Jul. 1999 pp:13:37.
- [Breitman 00] Breitman, K.K. *"Evolução de Cenários"* Ph.D. Theses at PUC-Rio May 2000.
- [Chung 93]Chung L., "Representing and Using Non-Functional Requirements: A Process Oriented Approach" Ph.D. Thesis, Dept. of Comp.Sci. University of Toronto, June 1993. Also tech. Rep. DKBS-TR-91-1.
- [Chung 95] Chung, L., Nixon, B. "Dealing with Non-Functional Requirements: Three Experimental Studies of a Process-Oriented Approach*" Proc. 17th Int. Con. on Software Eng. Seatle,* Washington, April pp: 24-28, 1995.
- [Chung 00] Chung, L., Nixon, B., Yu, Eric and Mylopoulos, M. "*Non-Functional Requirements in Software Engineering"* Kluwer Academic Publishers 1999.
- [Cysneiros 99] Cysneiros, L.M. and Leite, J.C.S.P. 'Integrating Non-Functional Requirements into data model" 4th International Symposium on Requirements Engineering – Ireland June 1999.
- [Cysneiros 01] Cysneiros,L.M., Leite, J.C.S.P. and Neto, J.S.M. "*A Framework for Integrating Non-Functional Requirements into Conceptual Models*" Requirements Engineering Journal – Vol 6 , Issue 2 Apr. 2001, pp:97-115.
- [Dardenne 93] Dardenne, A.. van Lamsweerde A, Fickas, S.. "Goal Directed Requirements Acquisition". *Science of Computer Programming, Vol. 20 pp: 3-50*, Apr. 1993.
- [Dobson 91] Dobson, J.E. *"On Non-Functional Requirements"* PDCS Techinical Report Series University of Lancaster No 65 May 1991.
- [EAGLE 95] Evaluation of Natural Language Processing Systems, http://www.issco.unige.ch/ewg95 1995.
- [Ebert97]Ebert, Christof. *Dealing with Nonfunctional in Large Software Systems.* Annals of Software Engineering, 3, 1997, pp. 367-395.
- [Finkelstein 96] Finkelstein, A. and Dowell J. "A comedy of Errors: The London Ambulance Service Case Study" *Proceedings of the Eighth International Workshop on Software Specification and Design,* IEEE Computer Society Press pp 2-5 1996

53

- Hauser 88] Hauser, J.R. and Hsiao, K. "*The House of Quality"*harvard Business review, May 1998 pp:63-73
- [Keller 90] Keller, S.E. et al *"Specifying Software Quality Requirements with Metrics"* in Tutorial System and Software Requirements Engineering IEEE Computer Society Press 1990 pp:145-163
- [Kirner 96] Kirner T.G. , Davis A .M. , "Nonfunctional Requirements of Real-Time Systems", *Advances in Computers, Vol 42 pp 1-37* 1996.
- [Leite 93] Leite J.C.S.P. and Franco, A.P.M. *"A Strategy for Conceptual Model Acquisition "* in Proceedings of the First IEEE International Symposium on Requirements Engineering, SanDiego, Ca, IEEE Computer Society Press, pp 243-246 1993.
- [Leite 97] Leite, J.C.S.P. et.al. *" Enhancing a Requirements Baseline with Scenarios*." Requirements Engineering Journal, 2(4):184-198, 1997.
- [Lindstrom93] Lindstrom, D.R. *Five Ways to Destroy a Development Project*. IEEE Software, September 1993, pp. 55-58.
- [Mylopoulos 92] Mylopoulos,J. Chung, L., Yu, E. and Nixon, B., "Representing and Using Non-functional Requirements: A Process-Oriented Approach." *IEEE Trans. on Software Eng, 18(6), pp:483-497,* June 1992.
- [Rumbaugh 99] Rumbaugh, J., Jacobson, I. and Booch,G. *"The Unified Modeling Language Reference Manual"* , Addison-Wesley, 1999.
- [Sommerville 92] Sommerville, I. "*Software Engineering"* fourth edition, Addison-Wesley, 1992.
- [Yu 95] Yu, E., Bois, P. and Mylopoulos, J. *"From Organizational Models to System Requirements"* The 3rd International Conference on Cooperative Information Systems, Vienna, May 1995
- [Yu 97] Yu, Eric *"Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering"* Proc. of the 3rd Interna. Symp. on Requirements Eng. Jan 1997 pp:226-235

54