



Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

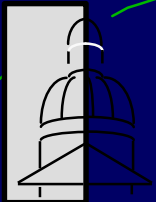
Les Approches Dirigées par les buts en Ingénierie des Besoins



Goal driven approaches in
requirements engineering



Colette Rolland
Université Paris Panthéon Sorbonne



Université de
Paris 1 - Sorbonne

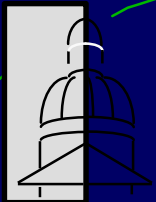


Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

Les Approches Dirigées par les buts en Ingénierie des Besoins

Outline

- ❖ UP1 research Centre in Informatics
- ❖ About Goals in Requirements Engineering
- ❖ L' Ecritoire overview



Université de
Paris 1 - Sorbonne

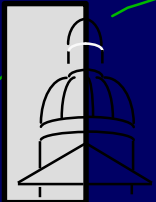


Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

CRI : Centre de Recherche en Informatique

<http://crinfo.univ-paris1.fr>

**Conceptual modelling,
OO, components & e-services development,
Methods & Case tools, Method Engineering & CAME tools,
Requirements Engineering,
System evolution and change handling,
Business process modelling and change management**



Université de
Paris 1 - Sorbonne

**300 publications, 82 PhD students, 20 European research projects
& industrial projects in France, Europe, USA, Japan, IFIP, CAiSE**

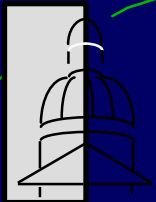


Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

The L' Ecritoire Approach to Requirements Elicitation

Outline

- UP1 research Centre in Informatics
- About Goals in Requirements Engineering
- L' Ecritoire overview



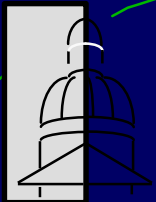
Université de
Paris 1 - Sorbonne



Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

Requirements Engineering

***“The hardest single part of building
a software system
is deciding precisely what to build”- F.Brooks***



Université de
Paris 1 - Sorbonne



Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

What is RE about?

- **The WHY question**

Why the system needs to be developed?

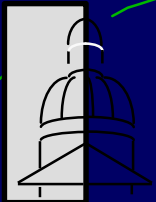
- **The WHAT question**

What the system shall do?

« Requirements definition must say

- why a system is needed, based on current or foreseen conditions,
- what system features will satisfy this context,
.....» (Ross77)

**IEEE Computer 85, IEEE SE 91-92, Bubenko94, Mylopoulos92,
Dardenne93, Loucopoulos95, Rolland98 etc..**



Université de
Paris 1 - Sorbonne



Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

What is RE About?

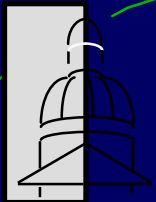


**Focusing on the the WHAT question
poses problems :**

- ◆ **Poor requirements are major source of failures (Standish95) 8000 projects, 350 US companies : 1/3 of projects never completed & 50% succeeded only partially**
- ◆ **Most perceived problems are related to requirements specification (>50%) - (ESI96) :3800 organisations in 17 European countries**



**Tackling the WHY question
gives hope for more purposeful systems
to be developed**



Université de
Paris 1 - Sorbonne

What is RE about?

The Requirements Engineering process

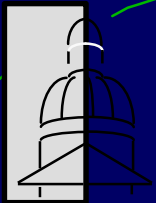
WHY ?

**Mission statement,
goals**

"RE is concerned with the identification of goals to be achieved by the envisioned system, the operationalisation of such goals into services and constraints..."
(Lamsweerde00)

WHAT ?

**Requirements
Specification**





Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

Goal Driven RE

Goals are recognised to play an important role in RE

**Goals are optative statements (as opposed to descriptive),
(Jackson95), expressions of intents**

Ex : Transport passengers safely

Assure customer loyalty

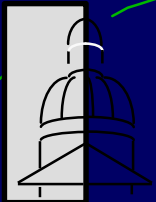
- **Avoid to deal with details and help focusing of the essentials**

Goals can be expressed at different level of abstraction

Ex : Keep doors close when moving

Keep record of loyal customers favourites!

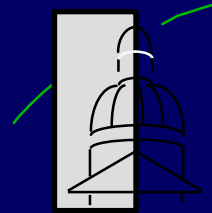
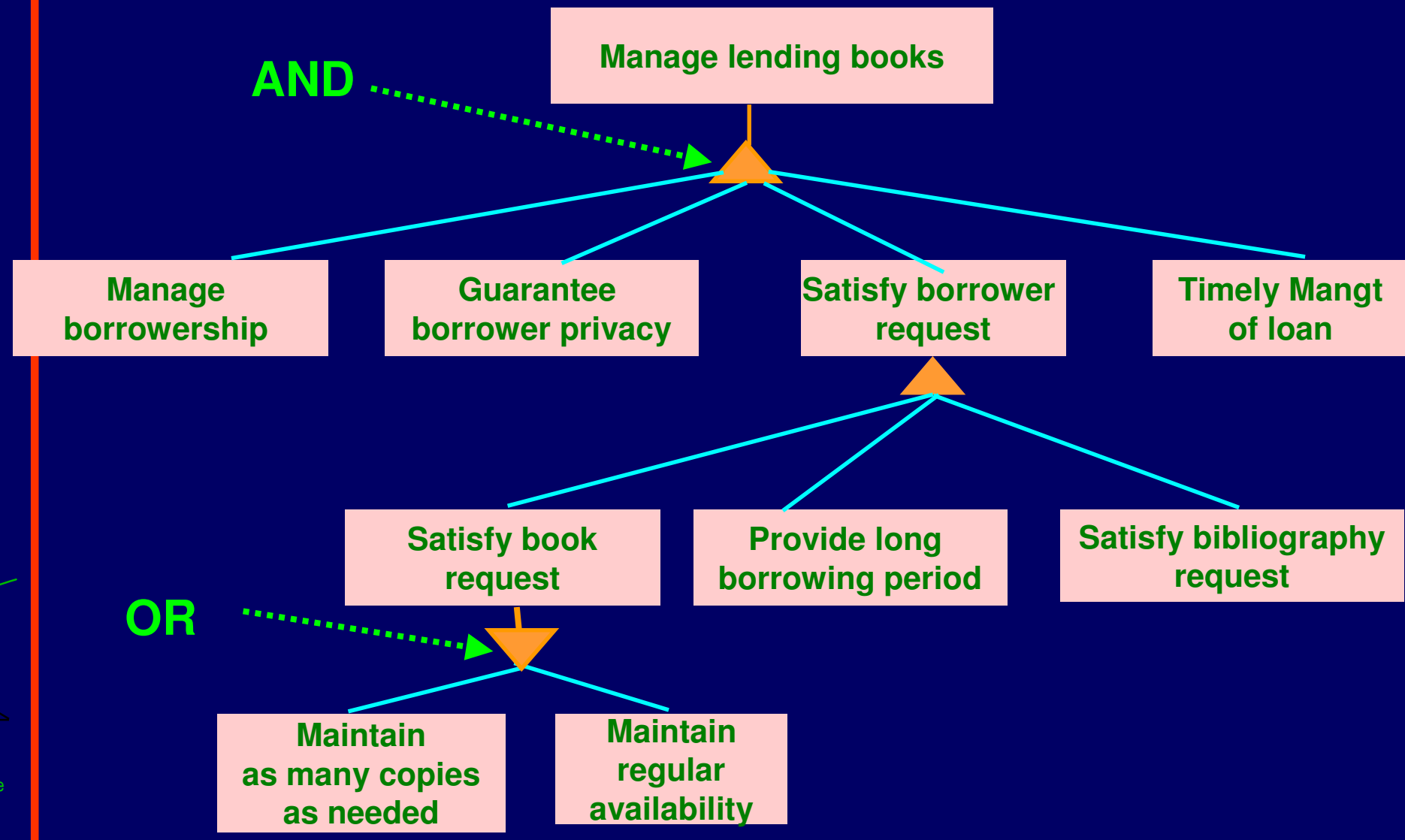
- **Goals drive the elaboration of requirements to support them**



Université de
Paris 1 - Sorbonne

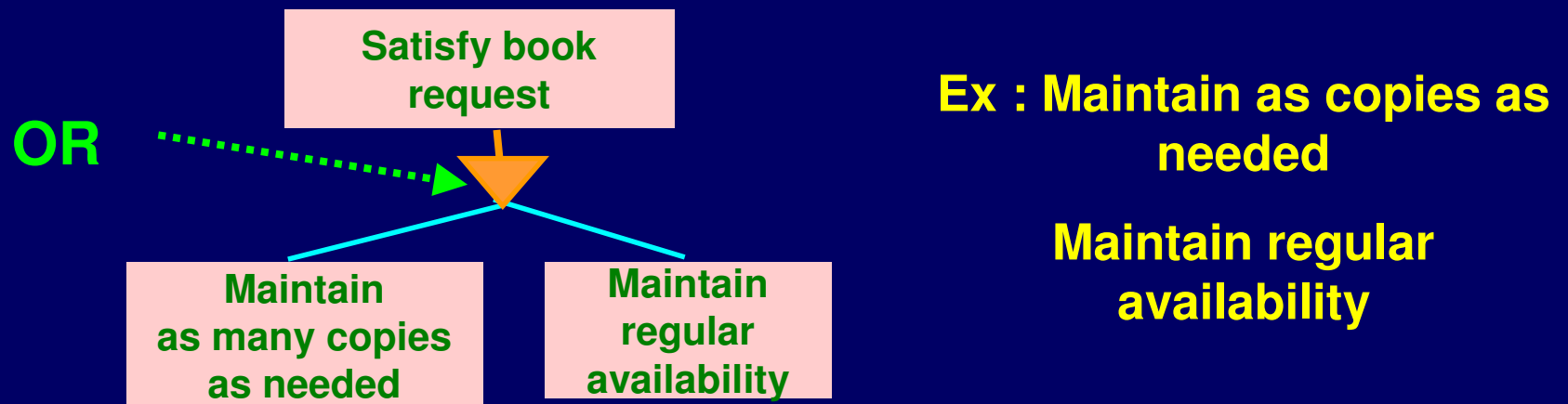
Goal Driven RE

- Goals provide rich structuring mechanism (AND/OR refinement)



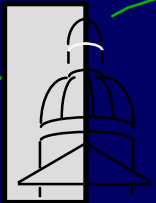
Goal Driven RE

- Goals allow to make explicit alternative design options



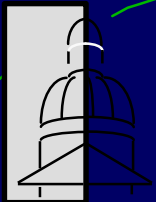
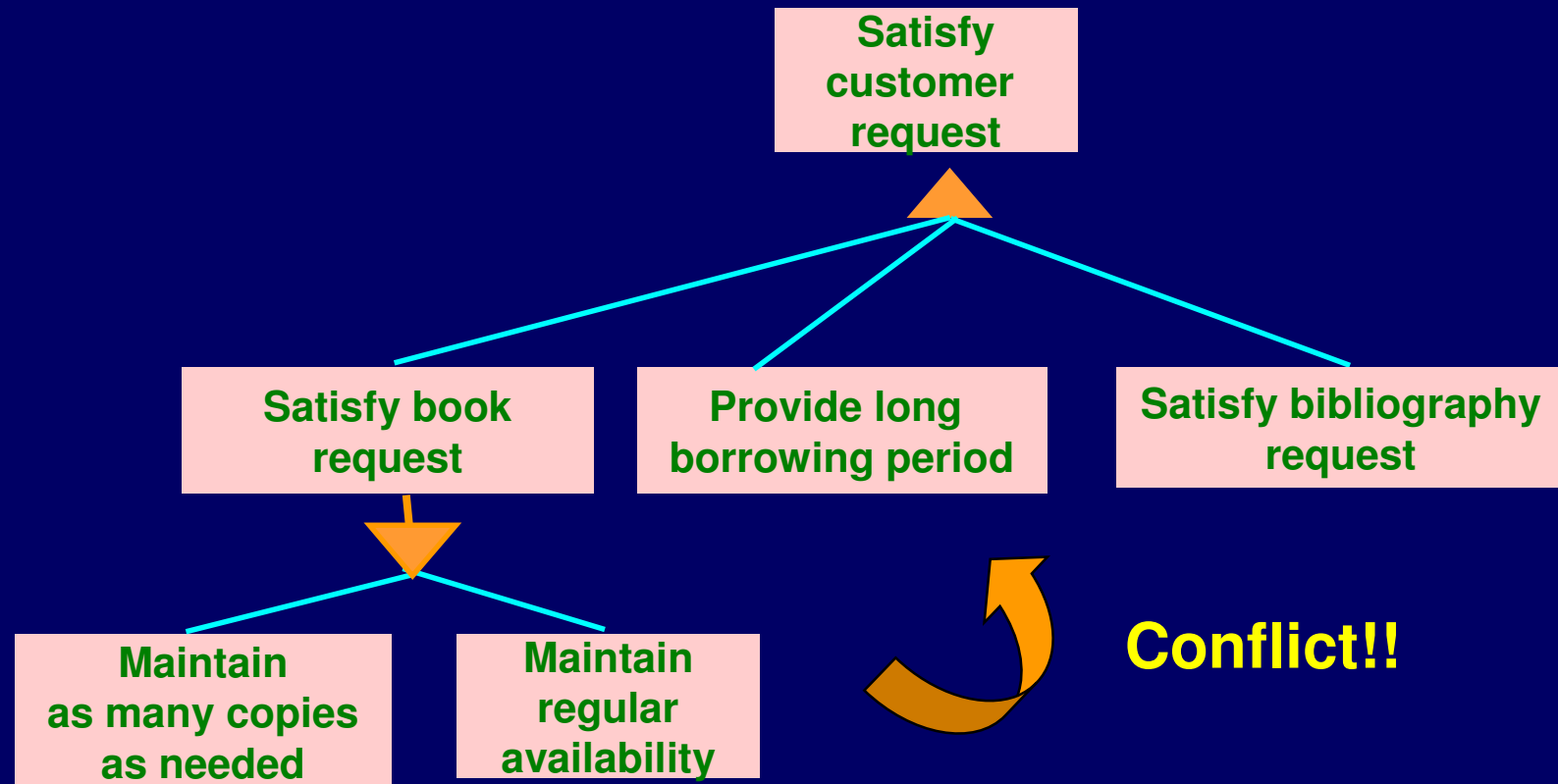
- Operationalisable goals seek for requirements

Ex: the system shall restrict the loan duration to 1 month



Goal Driven RE

- Goals are roots for conflict detection & resolution





Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

Goal Driven RE

Goals are recognised to play an important role in RE

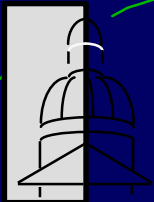
But : difficult to use in practice !!!

Relating goals to other concepts

- problem, opportunity, thread, (EKD,Bubenko94)
- objects, (Dardenne91, Mylopoulos92, Prat97, Rolland98)

The golden triplet : goal, scenario & agent

- scenarios, (Fickas92, Potts95, Leite97, Sutcliffe 98, Haumer98, Rolland98, Anton98)
- agents (responsibility- Kaos, Leiter01), (goal dependency- I*,Yu93)

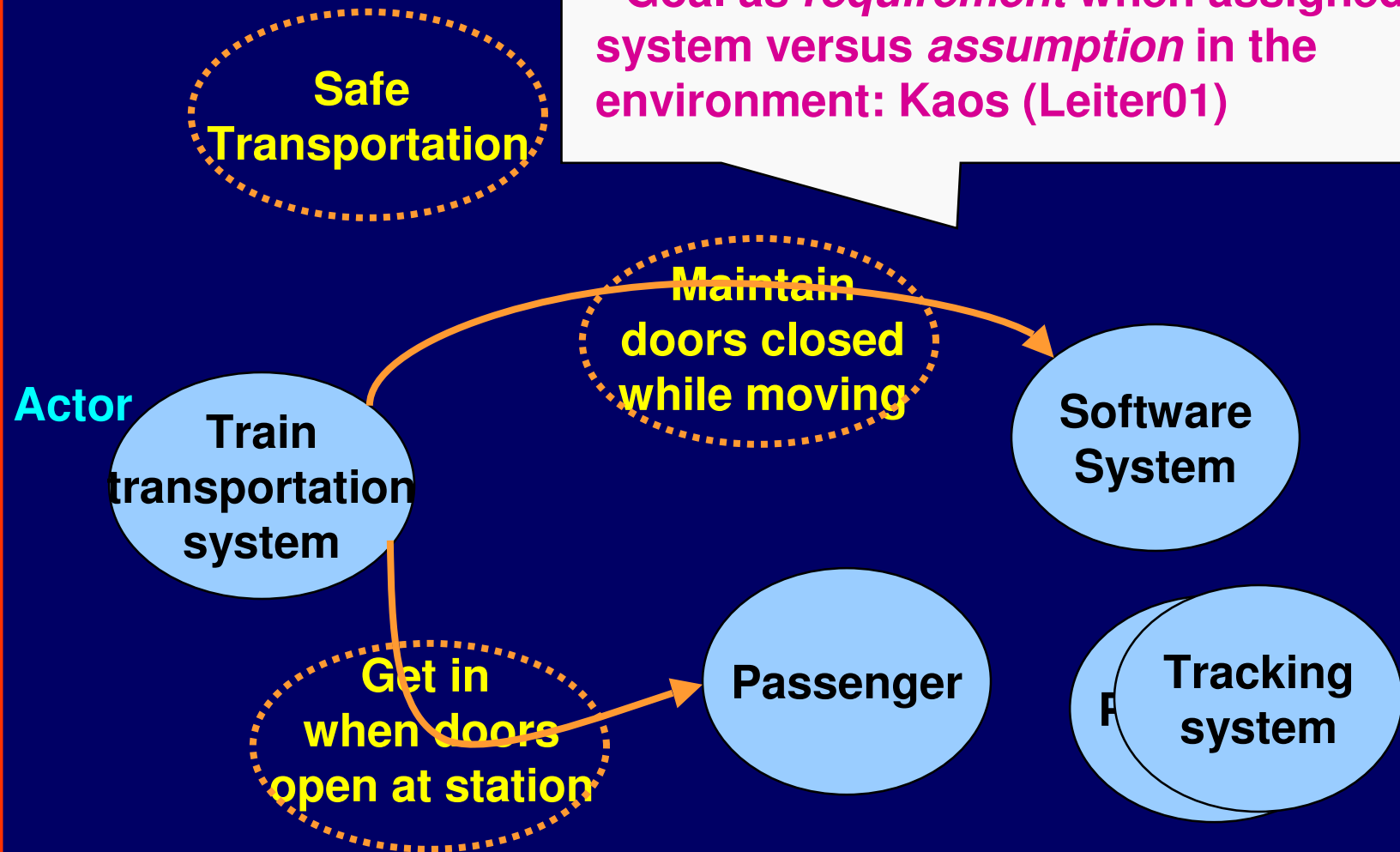


Université de
Paris 1 - Sorbonne

GoalAgent Driven RE

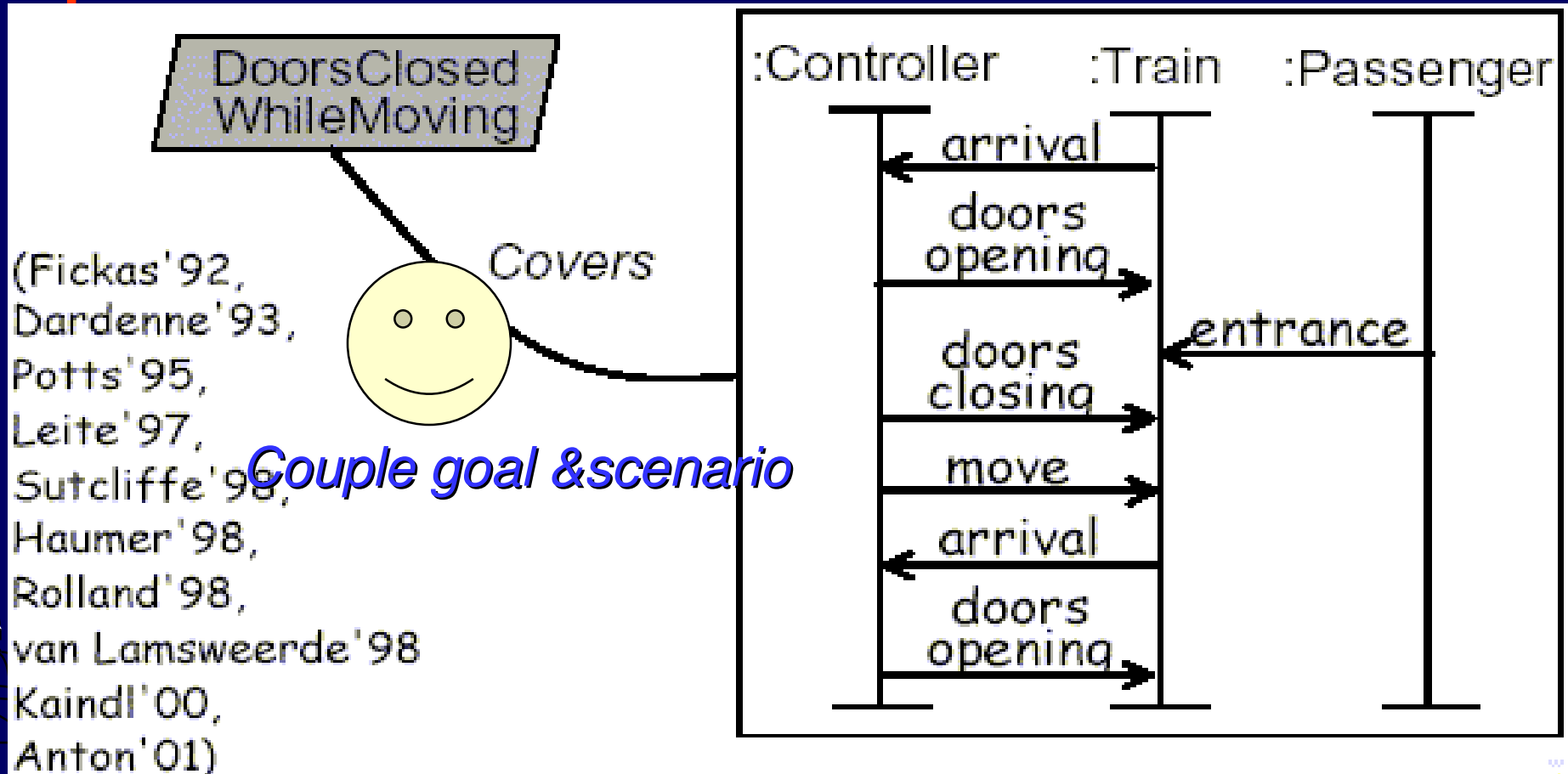
- Goals require co operati

- Actor responsible for goal (Feather87)
- Actor dependencies: I*(Yu94)
- Goal as *requirement* when assigned to system versus *assumption* in the environment: Kaos (Leiter01)



Goal/Scenario Driven RE

Scenarios are sequence of events between the user and the system



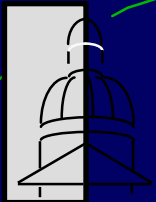


Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

Les Approches Dirigées par les buts en Ingénierie des Besoins

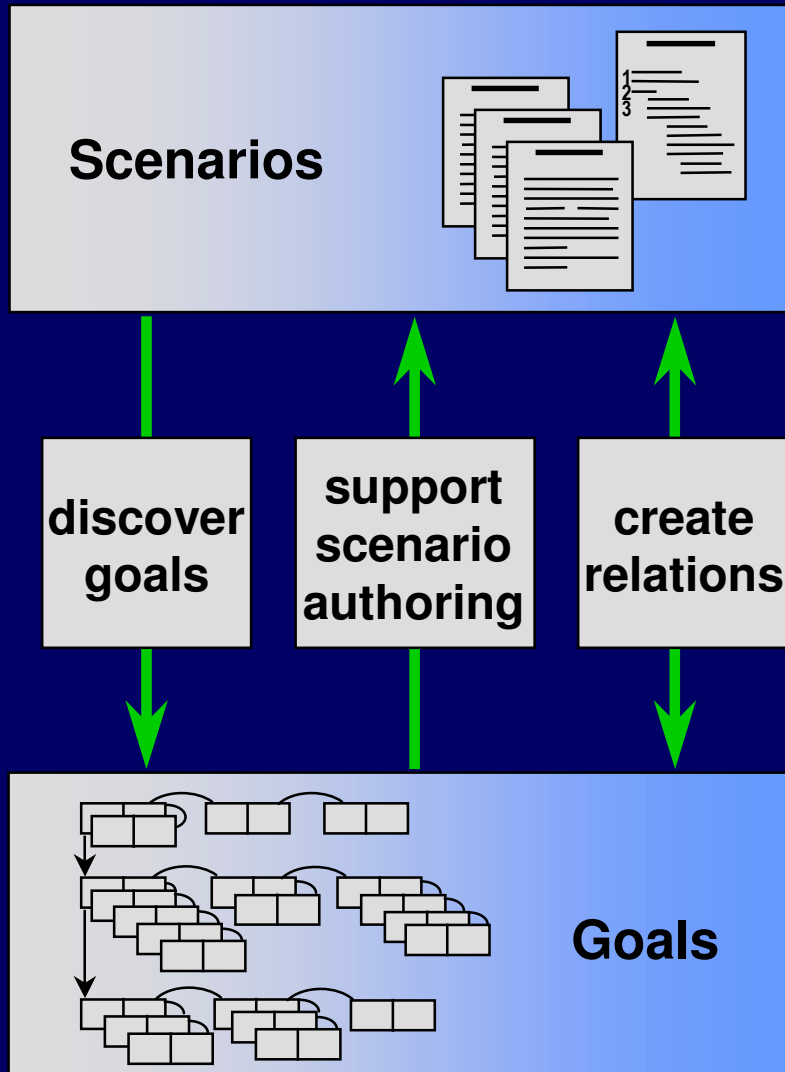
Outline

- ❖ UP1 research Centre in Informatics
- ❖ About Goals in Requirements Engineering
- ❖ L' Ecratoire overview



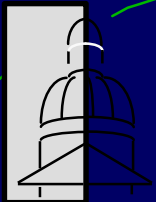
Université de
Paris 1 - Sorbonne

Overview

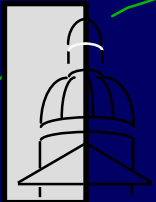
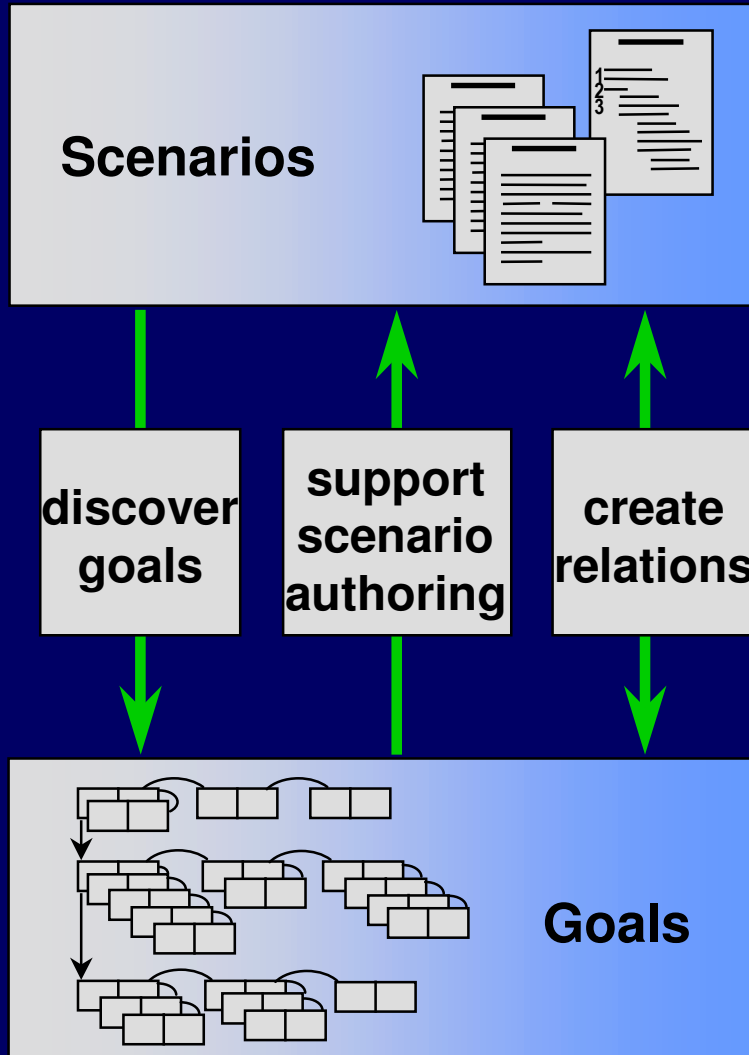


Guided elicitation of requirements from textual scenarios through bi-directional goal/scenario coupling :

- Support textual scenario authoring and processing
- Discover requirements/goals
- Organise (scenario, goal) collections
- Ground elicitation on methodological guidance and automated support

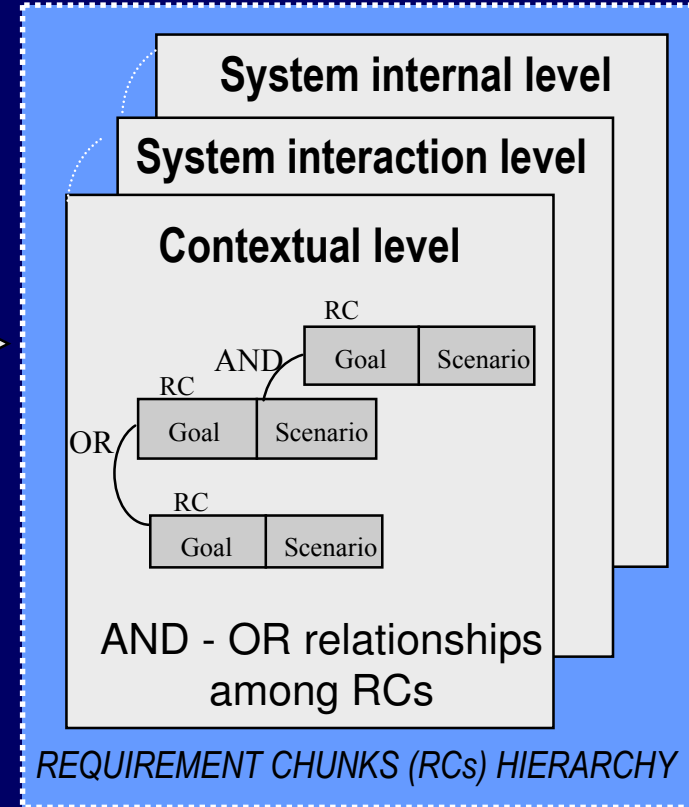
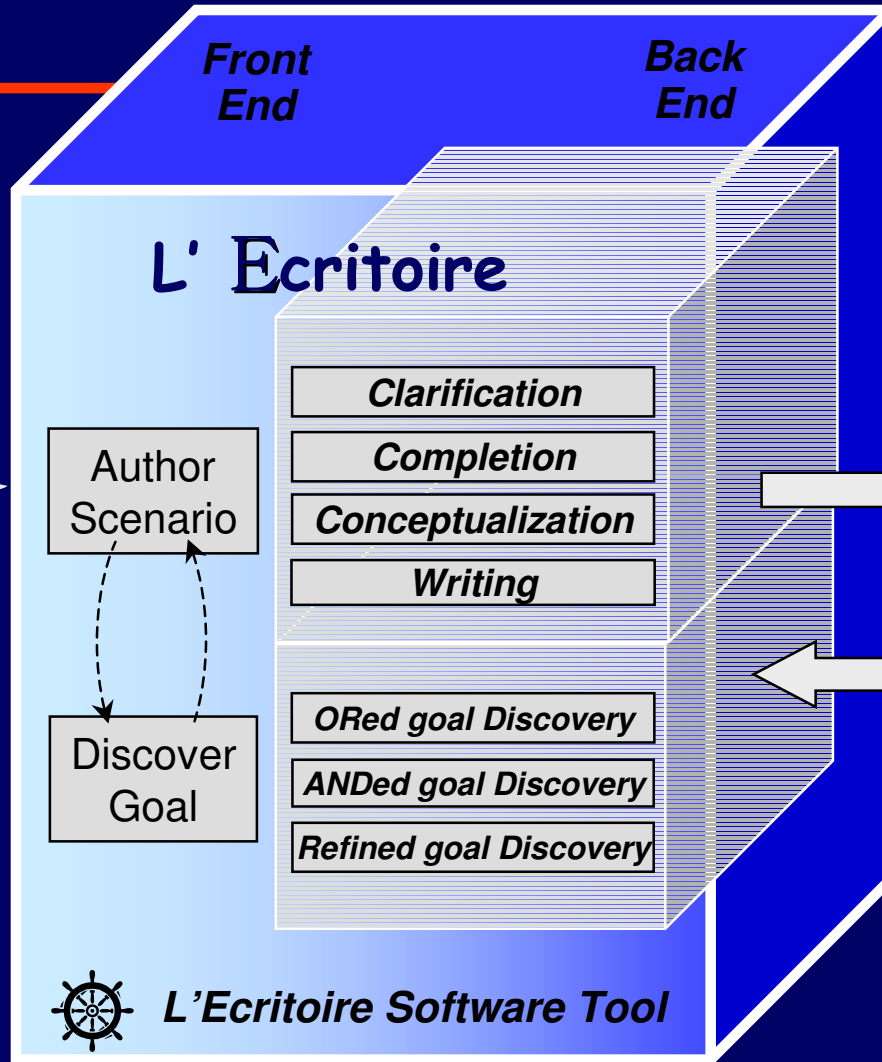


Overview





Overview



Textual scenarios



Menu of discovered goals

A Computer-based environment guiding a **systematic process** for eliciting requirements through **scenario authoring and analysis**



Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)

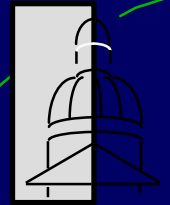
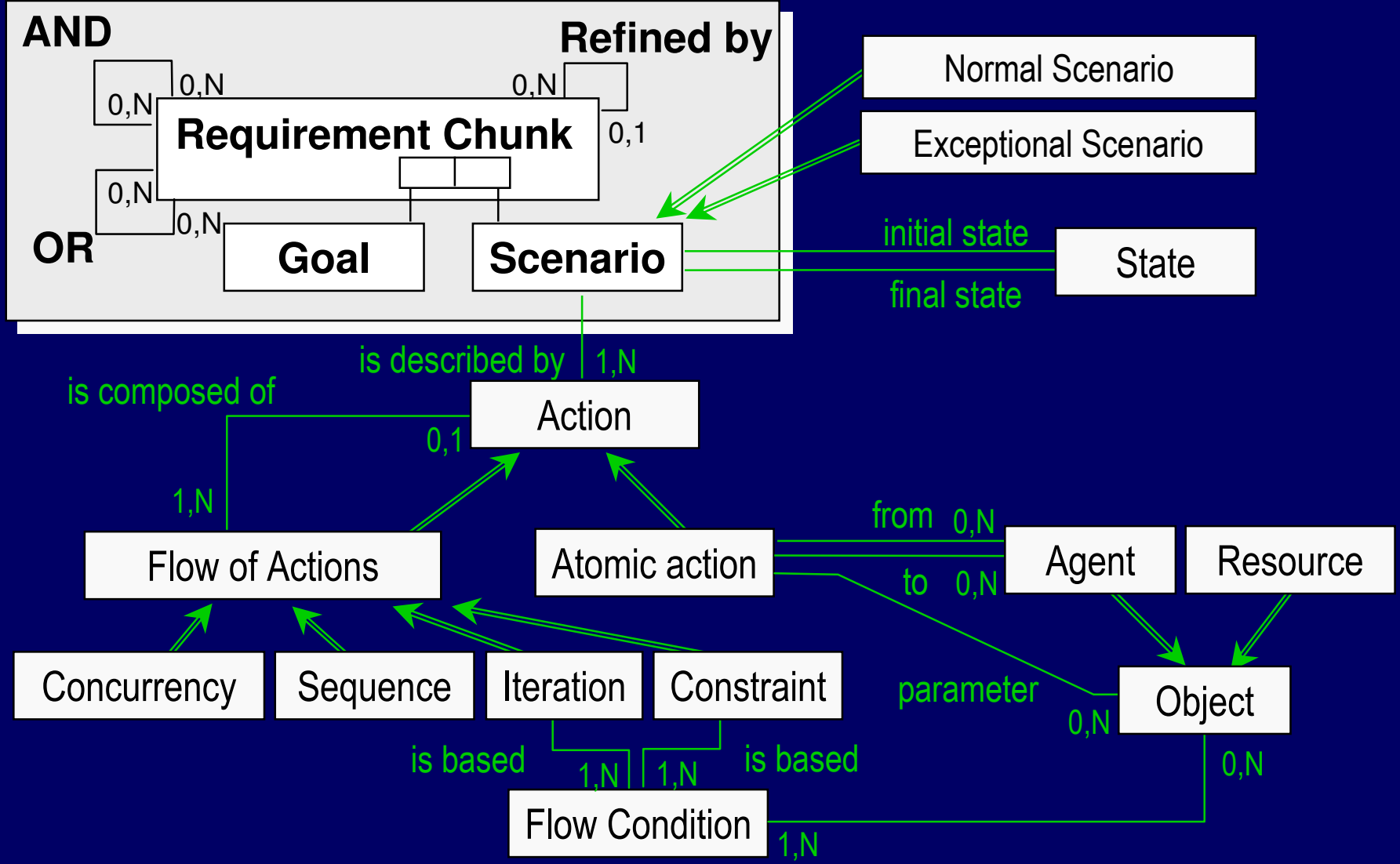
RE 05 A PARIS SORBONNE



Réservez les dates dans votre agenda!
Du Lundi 29 Août
au Vendredi 2 Septembre



Requirement Chunk Model



RC Example

A Requirement Chunk is a <Goal, Scenario> couple

Goal:

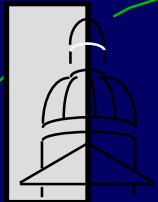
*Withdraw cash
from ATM in a
normal way*

*A specific
manner*

Scenario :

- 1- The user inserts a card in the ATM
- 2- The ATM checks if the card is valid
- 3- If the card is valid , then
- 4- A prompt for code is displayed by the ATM to the user
- 5- The user inputs a code in the ATM
- 6- If the code is valid then
- 7- The ATM displays a prompt for amount to the user
- 8- The user enters an amount in the ATM
- 9- If the amount is valid then
- 10- The ATM ejects the card
- 11- The ATM delivers the cash to the user

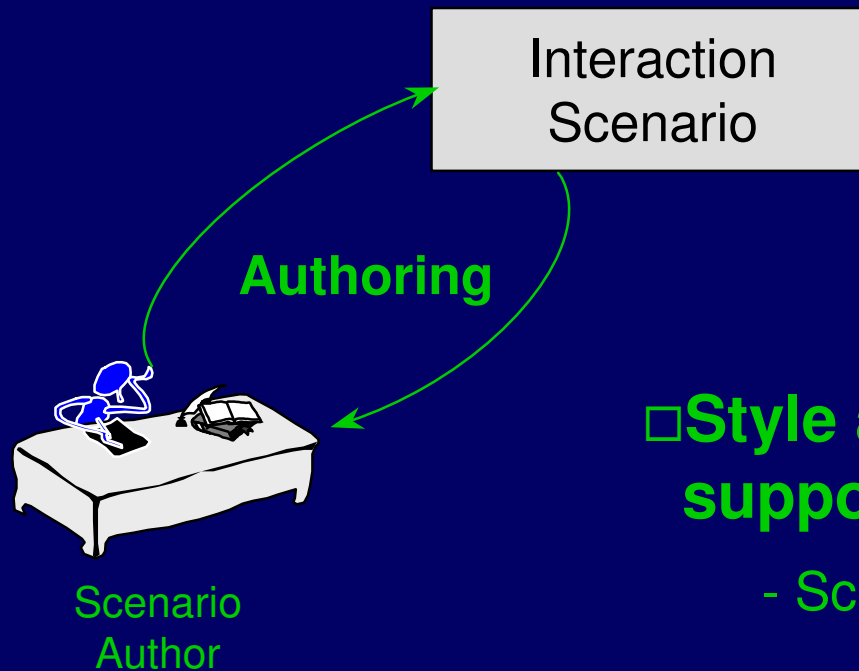
A single thread



Supporting Scenario Authoring



Cooperative Requirements
Engineering With Scenarios
(ESPRIT 21.903)



□ **Style and contents guidelines to support**

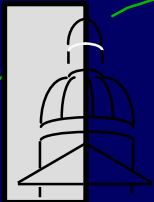
- Scenario writing

□ **Linguistic devices embodied in rules to support**

- Scenario clarification

- Scenario completion

- Scenario conceptualisation



Université de
Paris 1 - Sorbonne

Elicit Scenario

Chunk Name Chunk Type Goal: Manner:

States

Initial states

Final States

Checked Sentence

Scenario Description

The user inserts his card in the ATM. The system checks if the card is valid. Then, the prompt for code is given, and the user inputs the code. If the code is valid, a prompt for amount is displayed. Otherwise, a message is displayed to the user and the code is requested again.

RC States

RCs States

Goal Elicited

Withdraw cash from the ATM (In a normal way)

Guidelines

Type

Guidelines

Style Guideline :

- You should be consistent in the scenario terminology. Therefore, avoid the use of synonyms (one object with two different names) and homonyms (two different objects with the same name).

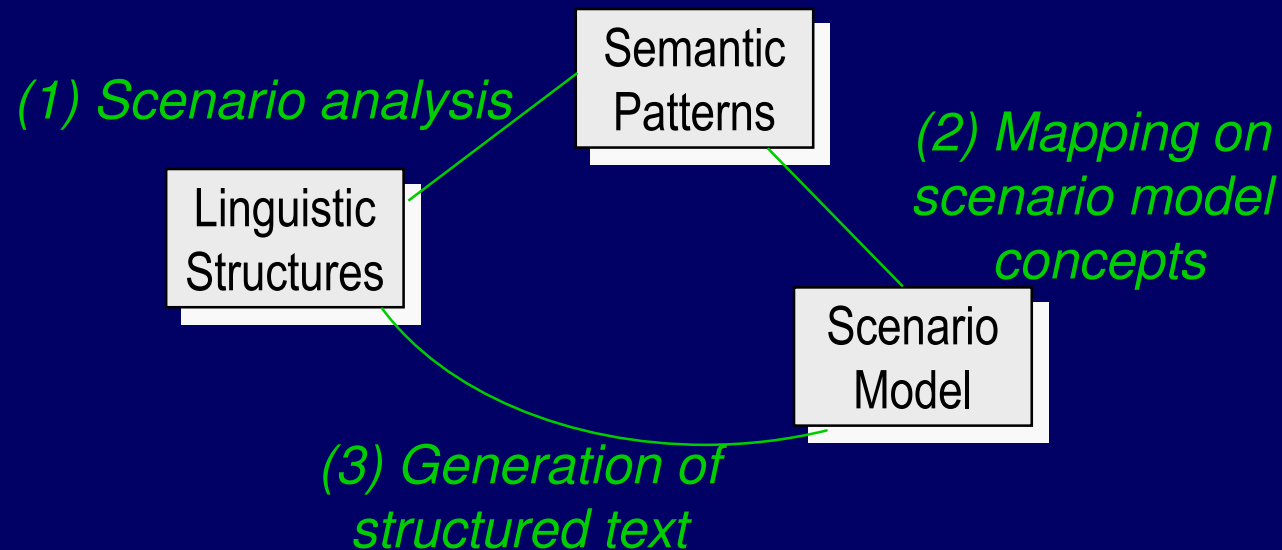
Contents Guideline :

- The expected scenario prose is a description of a single course of actions. Alternative scenarios, interruptions or exceptional treatments are described separately.

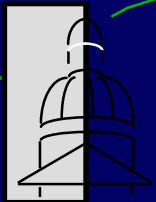
Select Rc by Clicking The Mouse Right Button

Linguistic Devices

Linguistic devices to address the *problem of the mapping from informal to formal scenarios*



- to identify ambiguities and incompleteness
- to help correcting them
- to transform narrative scenarios into well-structured and semantically correct scenarios



Verify Scenario Using completion Rules

Scenario Description

Scenario Description

Initial States

The user inserts his card in the ATM. The system checks if the card is given. The user inputs the code. If the code is valid, a prompt is entered. If it is valid, the ATM ejects the card to the user. If a receipt is asked, the ATM delivers the cash and a receipt is printed.

Instantiated Patterns

Scenario Description

Initial States

```
communication ( enter ) [ agent : ? ; object : an amount ; source : the user ; condition : it is valid ; constrained : communication ( eject ) [ agent : the ATM ; destination : the user ] ]
constraint [ condition : a receipt is asked by the user ; constrained : communication ( deliver ) [ agent : the ATM ; object : the cash ; after : communication ( print ) [ agent : ? ; object : a receipt ; source : the ATM ] ]
```

Errors List

Scenario Description

Initial States

```
communication ( insert ) [ agent : the user ; object : his card ; source : the user ]
```

Contains the Ambiguous Reference 'his ' Please Rep

```
action ( check ) [ agent : the system ; object : ? ]
```

Generate Patterns

Linguistic Check

Conceptualize Scenario

Instantiated Patterns

```
communication ( insert ) [ agent : the user ; object : a card ; source : the user ; destination : the ATM ] | action ( check ) [ agent : the ATM ; object : the card validity ] | constraint [ condition : the card is valid ; constrained : communication ( give ) [ agent : the ATM ; object : a prompt ; source : the ATM ] ]
```

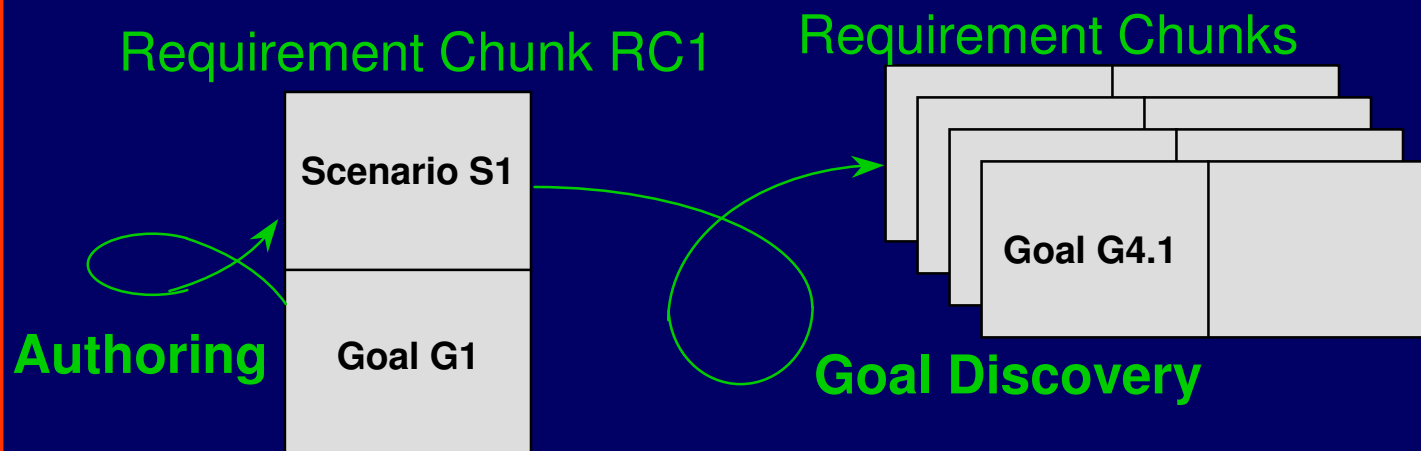
Conceptualized Scenario

1. the user inserts a card in the ATM
2. the ATM checks the card validity
3. If the card is valid Then
 4. a prompt for code is given by the ATM to the user
 5. the user inputs the code in the ATM
 6. the ATM checks the code validity
 7. If the code is valid Then
 8. a prompt for amount is displayed by the ATM to the user
 9. an amount is entered by the user to the ATM
 10. the ATM checks the amount validity
 11. If the amount is valid Then
 12. the ATM ejects the card to the user
 13. the ATM proposes a receipt to the user
 14. the user enters a choice in the ATM
 15. If a receipt is asked by the user to the ATM Then
 16. the ATM delivers the cash to the user
 17. a receipt is printed by the ATM to the user

Conceptualise

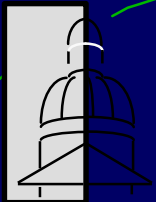
OK

Eliciting Requirements as Goals



□ Goal Discovery strategies embodied in scenario analysis rules to elicit :

- Complementary goals (ANDed),
- Alternative goals (ORed),
- Refined goals (Refined).



Elicit Ored Goals

Applied On (RC) : RC1

Missing Cases:

- 2. C 1 and not C 2
- 3. C 1 and C 2 and not C 3
- 4. C 1 and C 2 and C 3 and not C 4

Discovered Goals

Withdraw cash from the ATM with an invalid card

OK

Cancel

Alternative strategy

RC States

- RCs States
 - Goal Elicited
 - Scenario Elicited
 - Scenario Verified
 - Scenario Conceptualised
 - RC1 : Withdraw cash from the ATM (In a normal way)

Conditions List

- C 1= the card is valid
- C 2= the code is valid
- C 3= the amount is valid
- C 4= a receipt is asked by the user to the ATM

OK

Select Rc by Clicking The Mouse Right Button

Elicit Ored Goals

Applied On (RC) : RC1

Missing Cases:

- 2. C 1 and not C 2
- 3. C 1 and C 2 and not C 3
- 4. C 1 and C 2 and C 3 and not C 4

Discovered Goals

Withdraw cash from the ATM with an invalid card

OK

Cancel

RC States

RCs States

normal way)

Elicit Goal

Phrasing Manner

Missing Case:

C 1 and not C 2

Goal :

Withdraw cash from the ATM

Manner: with one code error correction Normal

Type: functional Exceptional

Ok

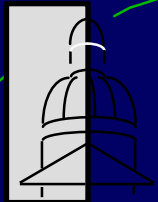
Cancel

Conditions List

- C 1= the card is valid
- C 2= the code is valid
- C 3= the amount is valid
- C 4= a receipt is asked by the user to the ATM

OK

Select Rc by Clicking The Mouse Right Button



L'ECRITOIRE [Window Title Bar]

Glossary Pending RC Goals Discovery Rc Hierarchy Reset Exit

Elicit Anded Goals [Dialog Title Bar]

RCs List : RC5 RC ID : Withdraw cash from the ATM

Actions List :

- 8. the ATM checks the amount validity
- 9. the ATM ejects the card to the user**
- 10. the ATM proposes a receipt to the user
- 11. the user enters a choice in the ATM
- 12. the ATM delivers the cash to the user

Add to Producing Add to Consuming

Producing :

- 12. the ATM delivers the cash to the user
- 13. the receipt is printed by the ATM to the u
- 9. the ATM ejects the card to the user**

Remove Add to Consuming

Consuming :

- 1. the user inserts a card in the ATM**

Remove Add to Producing

Couple

Producing : Consuming

Generate Goals Remove

Elicited Goals

Remove Rephrase

OK Cancel

Select RC by Clicking The Mouse Right Button

Composition strategy [Callout Box]



Cooperative Requirements Engineering With Scenarios (ESPRIT 21.903)

L'ECRITOIRE
 Glossary Pending RC Goals Discovery Rc Hierarchy Reset Exit

Elicit Anded Goals

RCs List : RC5 RC Goal : Withdraw cash from the ATM

Actions List :

- 8. the ATM checks the amount validity
- 9. the ATM ejects the card to the user
- 10. the ATM proposes a receipt to the user
- 11. the user enters a choice in the ATM
- 12. the ATM delivers the cash to the user

Add to Producing Add to Consuming

Producing :

- 12. the ATM delivers the cash to the user
- 13. the receipt is printed by the ATM to the u

Remove Add to Consuming

Consuming :

Remove

Couple

Producing : Consuming

9. the ATM ejects the card to the user : 1. the user inserts a card in the ATM

Generate Goals Remove

Elicited Goals

- the user produces the cash (Generated from : 12. the ATM delivers the cash to the user)
- the ATM consumes the cash (Generated from : 12. the ATM delivers the cash to the user)
- the user produces the receipt (Generated from : 13. the receipt is printed by the ATM to the user)

Remove Rephrase

OK Cancel

Elicit Anded Goal:

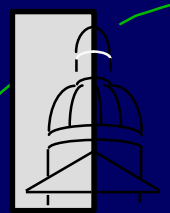
Phrasing Goal

Goal : Insert cash in the ATM

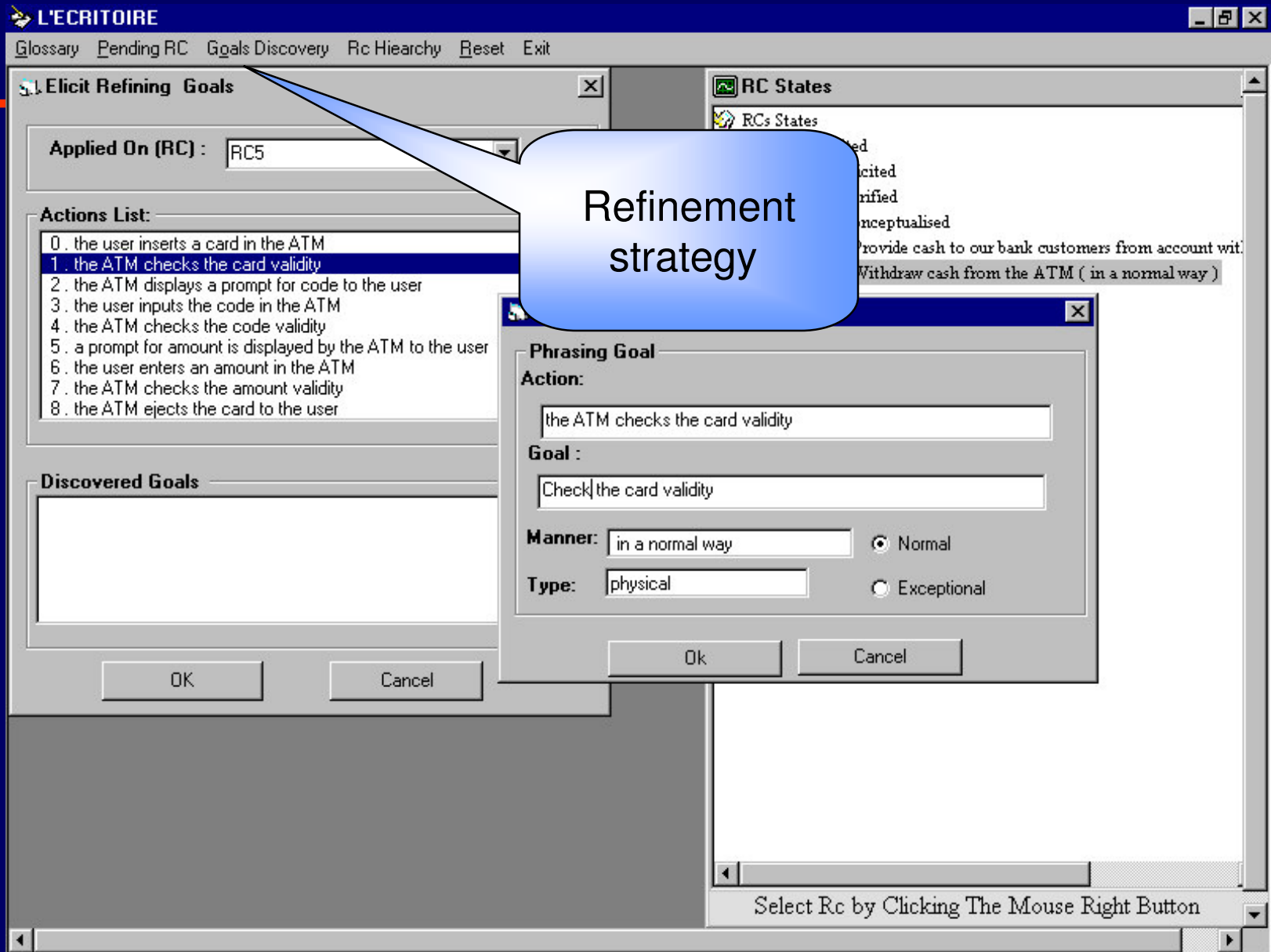
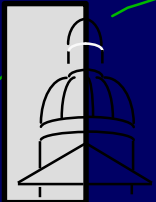
Manner : in a normal way

Ok Cancel

Select RC by Clicking The Mouse Right Button



Université de Paris 1 - Sorbonne



L'ECRITOIRE
Glossary Pending RC Goals Discovery Rc Hierarchy Reset Exit

Elicit Refining Goals

Applied On (RC) : RC5

Actions List:

0. the user inserts a card in the ATM
1. the ATM checks the card validity
2. the ATM displays a prompt for code to the user
3. the user inputs the code in the ATM
4. the ATM checks the code validity
5. a prompt for amount is displayed by the ATM to the user
6. the user enters an amount in the ATM
7. the ATM checks the amount validity
8. the ATM ejects the card to the user

Discovered Goals

Phrasing Goal

Action: the ATM checks the card validity

Goal : Check the card validity

Manner: in a normal way Normal

Type: physical Exceptional

Ok Cancel

Select Rc by Clicking The Mouse Right Button

Organising the RCs Hierarchy

L'ECRITOIRE

Glossary Pending RC Goals Discovery Rc Hierarchy Reset Exit

Requirement Chunks Hierarchy

RC1

Withdraw cash from the

Scenario Description

the user inserts a card in the ATM
the ATM checks the card validity
|| the card is valid || Then
a prompt for code is given by the ATM to the user

RC1 is Anded to :

RC4 : Fill in the ATM with cash (In a normal way)
RC5 : Fill in the ATM with receipt papers (in a normal way)

RC1 is Ored to :

RC2 : Withdraw cash from the ATM (with an invalid card)
RC3 : Withdraw cash from the ATM (with one code error correction)

RC1 is Refined by :

RC6 : Check the card validity (in a normal way)
RC7 : Check the code validity (in a normal way)

OK

Organising the <Goal, Scenario> collection

Organising the
<Goal, Scenario>
collection

normal way

AND

RC4 : Fill in the ATM with cash

AND

RC5 : Fill in the ATM with paper

RC2 : Withdraw cash with an invalid card

OR

RC3 : Withdraw cash with one code error correction

Physical level

RC6 : Check the card validity

AND

RC7 : Check the code validity

Refined by

normal way

AND

RC4 : Fill in the ATM with cash

AND

RC5 : Fill in the ATM with paper

RC2 : Withdraw cash with an invalid card

OR

RC3 : Withdraw cash with one code error correction

Physical level

RC6 : Check the card validity

AND

RC7 : Check the code validity