Requirements Engineering

RE Activities

Conclusion

### How to Make a Tree Swing?



#### Picture from projectcartoon.com

		-) 40
Introduction to Requirements Engineering	(O) PUBLICDOMAIN	1/62
Matthieu Vergne vergne@is.naist.jp		NAIST

# Introduction to Requirements Engineering

#### Matthieu Vergne

Nara Institute of Science and Technology



#### March 1st, 2017

Introduction to Req	uirements Engineering		DUBLICDOMAIN	2/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

◆□ > ◆□ > ◆三 > ◆三 > 三日 のへで

### Seminar Objectives

Understand what is a requirement

troduction to Requ	uirements Engineering	3/6
atthieu Vergne	vergne@is.naist.jp	NAIS

< □ > < □ > < 三 > < 三 > < 三 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □

# Seminar Objectives

- Understand what is a requirement
- Understand what is Requirements Engineering (RE)

Introduction to Rec	juirements Engineering		62
Matthieu Vergne	vergne@is.naist.jp	NAI	sт

# Seminar Objectives

- Understand what is a requirement
- Understand what is Requirements Engineering (RE)
- Understand how RE impacts/is impacted by software projects

Introduction to Req	uirements Engineering	() PUBLECOMARI	3/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

# Seminar Objectives

- Understand what is a requirement
- Understand what is Requirements Engineering (RE)
- Understand how RE impacts/is impacted by software projects
- Get a broad overview of existing RE techniques

Introduction to Req	uirements Engineering	3/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

< 口 > < 同 >

(E)

# Seminar Objectives

- Understand what is a requirement
- Understand what is Requirements Engineering (RE)
- Understand how RE impacts/is impacted by software projects
- Get a broad overview of existing RE techniques

This course is inspired from:

- RE course of Anna Perini, University of Trento, Italy (2014)
- Guide to the Software Engineering Body of Knowledge v3 (Bourque et al. [2014])

ELE DQO

# Seminar Objectives

- Understand what is a requirement
- Understand what is Requirements Engineering (RE)
- Understand how RE impacts/is impacted by software projects
- Get a broad overview of existing RE techniques

This course is inspired from:

- RE course of Anna Perini, University of Trento, Italy (2014)
- Guide to the Software Engineering Body of Knowledge v3 (Bourque et al. [2014])

Slides available on the Web:

https://www.matthieu-vergne.fr/?page=teaching

Introduction to Req	uirements Engineering		2
Matthieu Vergne	vergne@is.naist.jp	NAIS	Г

4/62 NAIST

# Outline

#### 1 Requirements

2 Requirements Engineering

#### **3** RE Activities

- Requirements Elicitation
- Requirements Modelling
- Requirements Analysis
- Requirements Prioritisation
- Requirements Management

### 4 Conclusion

		_	_	
Introduction to Requ	uirements Engineering			DUBLICDOMAIN
Matthieu Vergne	vergne@is.naist.jp			

# Outline

#### **1** Requirements

2 Requirements Engineering

### 3 RE Activities

- Requirements Elicitation
- Requirements Modelling
- Requirements Analysis
- Requirements Prioritisation
- Requirements Management

#### 4 Conclusion

			문 가 문 문	*)40
Introduction to Require	ements Engineering	[	D PUBLICDOMAIN	5/62
Matthieu Vergne v	ergne@is.naist.jp			NAIST

Requirements ●0000	Requirements Engineering	RE Activities 000000000000000000000000000000000000	Conclusion 0000
Definiti	ons		
IEEE	Standards Board [19	990]:	

1 A condition or capacity needed by a user to solve a problem or

achieve an objective.

Introduction to Requ	uirements Engineering	PUBLICDOMAIN	6/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

イロト イヨト イヨト イヨト ショー クタイ

Requirements	
00000	

イロト 不得 トイヨト イヨト ヨヨ ろくで

# Definitions

IEEE Standards Board [1990]:

- A condition or capacity <u>needed by a user</u> to solve a problem or achieve an objective.
- A condition or capability that <u>must be met or possessed by a</u> <u>system or system component</u> to satisfy a contract, standard, specification, or other formally imposed documents.

Introduction to Req	uirements Engineering	(0) PUBLEDOMAIN 6/6	62
Matthieu Vergne	vergne@is.naist.jp	NAIS	Т

Requirements	
00000	

< 口 > < 同 >

# Definitions

IEEE Standards Board [1990]:

- A condition or capacity <u>needed by a user</u> to solve a problem or achieve an objective.
- A condition or capability that <u>must be met or possessed by a</u> <u>system or system component</u> to satisfy a contract, standard, specification, or other formally imposed documents.
- A documented representation of a condition or capability as in (1) or (2).

# Definitions

IEEE Standards Board [1990]:

- A condition or capacity <u>needed by a user</u> to solve a problem or achieve an objective.
- A condition or capability that <u>must be met or possessed by a</u> <u>system or system component</u> to satisfy a contract, standard, specification, or other formally imposed documents.
- A documented representation of a condition or capability as in (1) or (2).

### SWEBOK v3 [2014]:

A software requirement is a property that must be exhibited by something in order to solve some problem in the real world.

Introduction to Requ	irements Engineering	(0) PUBLICOMAIN 6/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

# Definitions

IEEE Standards Board [1990]:

- A condition or capacity <u>needed by a user</u> to solve a problem or achieve an objective.
- A condition or capability that <u>must be met or possessed by a</u> <u>system or system component</u> to satisfy a contract, standard, specification, or other formally imposed documents.
- A documented representation of a condition or capability as in (1) or (2).

### SWEBOK v3 [2014]:

A software requirement is a property that must be exhibited by something in order to solve some problem in the real world. (user? document?)

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements 0●000	Requirements Engineering	RE Activities ০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০	Conclusion
Classifica	ations		

Various kinds of requirements:

Product Need or constraint on the product to be developed.

The software shall verify that a student meets all pre-requisites before he or she registers for a course.

Introduction to Req	uirements Engineering	O PUBLICDOMAIN 7/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□ > ◆□ > ◆三 > ◆三 > 三日 のへで

Requirements	Requirements Engineering	RE Activities	Conclusion
	0000000		0000

### Classifications

Various kinds of requirements:

Product Need or constraint on the product to be developed.

• The software shall verify that a student meets all pre-requisites before he or she registers for a course.

Process Constraint on the development of the product.

• The software shall be developed using Agile methods.

Introduction to Req	uirements Engineering	<b>Ø</b> PUBLIECOMAIN 7/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

### Classifications

Various kinds of requirements:

Product Need or constraint on the product to be developed.

• The software shall verify that a student meets all pre-requisites before he or she registers for a course.

Process Constraint on the development of the product.

- The software shall be developed using Agile methods.
- Functional Functions that the product is to execute.
  - formatting some text, modulating a signal

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

### Classifications

Various kinds of requirements:

Product Need or constraint on the product to be developed.

• The software shall verify that a student meets all pre-requisites before he or she registers for a course.

Process Constraint on the development of the product.

• The software shall be developed using Agile methods.

Functional Functions that the product is to execute.

■ formatting some text, modulating a signal

Quality Constrain the solution.

performance, reliability, safety, security, maintainability, etc.

Introduction to Requi	rements Engineering		MAIN	7/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

### Classifications

Various kinds of requirements:

Product Need or constraint on the product to be developed.

• The software shall verify that a student meets all pre-requisites before he or she registers for a course.

Process Constraint on the development of the product.

• The software shall be developed using Agile methods.

Functional Functions that the product is to execute.

■ formatting some text, modulating a signal

Quality Constrain the solution.

performance, reliability, safety, security, maintainability, etc.

System Requirements for the whole system.

■ Software + hardware + people + information + etc.

Introduction to Requ	uirements Engineering	(0) PUBLEDOMAIN 7/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements 0●000	Requirements Engineering	RE Activities	Conclusion
Classifi	cations		
Vario	us kinds of requireme	nts:	
Product	Need or constraint of	n the product to be developed.	
	The software shall	verify that a student meets all pre-requisites	;
	before he or she re	egisters for a course.	
Process	Constraint on the de	evelopment of the product.	
	The software shall	be developed using Agile methods.	
Functional	Functions that the p	roduct is to execute.	
	formatting some to	ext, modulating a signal	
Quality	Constrain the solution	on.	
	performance, relia	bility, safety, security, maintainability, etc.	
System	Requirements for the	e whole system.	
	Software + hardware	are $+$ people $+$ information $+$ etc.	
Software	Reqs for the softwar	e part, derived from the system reqs.	

	・ロ・・日・・ 日・	<	୬୯୯
Introduction to Requirements Engineering		DUBLICDOMAIN	7/62
Matthieu Vergne vergne@is.naist.jp			NAIST

Requirements 0●000	Requirements Engineering	RE Activities 000000000000000000000000000000000000	Conclusion 0000
Classifi	cations		
Vario	us kinds of requireme	nts:	
Product	Need or constraint of	n the product to be developed.	
	The software shall before he or she re	verify that a student meets all pre-requisites gisters for a course.	i
Process	Constraint on the de	evelopment of the product.	
	The software shall	be developed using Agile methods.	
Functional	Functions that the p	roduct is to execute.	
	formatting some to	ext, modulating a signal	
Quality	Constrain the solution	on.	
	performance, relia	bility, safety, security, maintainability, etc.	
System	Requirements for the	e whole system.	
	Software + hardware	are $+$ people $+$ information $+$ etc.	
Software	Reqs for the softwar	e part, derived from the system reqs.	

Etc. ...

	티트 《로》《로》《唱》《日》	500
Introduction to Requirements Engine	eering (0) PUBLICDOMAIN	7/62
Matthieu Vergne vergne@is.nai	ist.jp	NAIST

# Classifications Usage

Requirements classifications are:

Supports for identifying relevant requirements.

Introduction to Rec	uirements Engineering		8/62
Matthieu Vergne	vergne@is.naist.jp	N.	AIST

# Classifications Usage

Requirements classifications are:

- Supports for identifying relevant requirements.
- Not strict rules to be followed.

Introduction to Req	uirements Engineering		8/62
Matthieu Vergne	vergne@is.naist.jp	Ν	IAIST

# **Classifications** Usage

Requirements classifications are:

- Supports for identifying relevant requirements.
- Not strict rules to be followed.



Introduction to Requ	irements Engineering		8/62
Matthieu Vergne	vergne@is.naist.jp	М	NAIST

# **Classifications** Usage

Requirements classifications are:

- Supports for identifying relevant requirements.
- Not strict rules to be followed.



Introduction to Requ	uirements Engineering	DIBLICDOMAIN	8/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

イロト イヨト イヨト イヨト ショー クタイ

- What is needed (requirement *per se*)
  - Register participants for an event

Introduction to Red	uirements Engineering	9/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

- What is needed (requirement *per se*)
  - Register participants for an event
- Who needs it
  - Event organisers

Introduction to Rec	uirements Engineering		9/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

- What is needed (requirement per se)
  - Register participants for an event
- Who needs it
  - Event organisers
- Why it is needed
  - Identify participants
  - Store contact information
  - Measure event interest
  - Evaluate food requirements

Introduction to Req	uirements Engineering	9/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

- What is needed (requirement per se)
  - Register participants for an event
- Who needs it
  - Event organisers
- Why it is needed
  - Identify participants
  - Store contact information
  - Measure event interest
  - Evaluate food requirements
- How it is fulfilled
  - Registration form on a website
  - Registration phone call
  - On-site registration desk

Introduction to Req	uirements Engineering		9/62
Matthieu Vergne	vergne@is.naist.jp	1	NAIST

Requirements 0000●	Requirements Engineering	RE Activities ০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০০	Conclusic 0000
	21 11		

### Many Challenges

What is needed (requirement per se)

- Could be ambiguous ("track" instead of "register")
- Could be subjective (no reliable test procedure)

Introduction to Rec	quirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

# Many Challenges

- What is needed (requirement per se)
  - Could be ambiguous ("track" instead of "register")
  - Could be subjective (no reliable test procedure)
- Who needs it
  - Could focus on irrelevant people (not directly involved)
  - Could forget important people (some organisers)
  - Could forget authorities (regulators, law)

Introduction to Req	uirements Engineering	10/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト イタト イモト イモト 三日 のうろ

# Many Challenges

- What is needed (requirement per se)
  - Could be ambiguous ("track" instead of "register")
  - Could be subjective (no reliable test procedure)
- Who needs it
  - Could focus on irrelevant people (not directly involved)
  - Could forget important people (some organisers)
  - Could forget authorities (regulators, law)
- Why it is needed
  - Could be unknown (no access to the decision team)
  - Could be wrong (failed guess)
  - Could be conflictual (disagreements)

			1.15
Introduction to Req	uirements Engineering	(0) PUBLICDOMAIN 10	0/62
Matthieu Vergne	vergne@is.naist.jp	NA	IST

# Many Challenges

- What is needed (requirement per se)
  - Could be ambiguous ("track" instead of "register")
  - Could be subjective (no reliable test procedure)
- Who needs it
  - Could focus on irrelevant people (not directly involved)
  - Could forget important people (some organisers)
  - Could forget authorities (regulators, law)
- Why it is needed
  - Could be unknown (no access to the decision team)
  - Could be wrong (failed guess)
  - Could be conflictual (disagreements)
- How it is fulfilled
  - Could be unknown (lack expertise)
  - Could be inadapted (lack alternatives)
  - Could be infeasible (conflicts, law restrictions)

Introduction to Req	uirements Engineering	PUBLICDOMAIN	10/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

# Outline

#### 1 Requirements

#### 2 Requirements Engineering

#### 3 RE Activities

- Requirements Elicitation
- Requirements Modelling
- Requirements Analysis
- Requirements Prioritisation
- Requirements Management

#### 4 Conclusion

	문제 제품 제품(문	*)4(*
Introduction to Requirements Engineering	DUBLICDOMAIN	11/62
Matthieu Vergne vergne@is.naist.jp		NAIST

Requirements 00000	Requirements Engineering ●000000	RE Activities 000000000000000000000000000000000000	Conclusion	
What is RE?				

Several detailed definitions depending on emphasis, but in short:

	・ 耳(叫) スポット 御マ スロッ	୬ବନ
Introduction to Requirements Engineering		12/62
Matthieu Vergne vergne@is.naist.jp	Ν	NAIST
Requirements 00000

イロト 不得 トイヨト イヨト ヨヨ ろくで

#### What is RE?

Several detailed definitions depending on emphasis, but in short:

Requirements Engineering

Systematic handling of requirements. (SWEBOK v3 [2014])

Introduction to Req	uirements Engineering	12/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

< 口 > < 同 >

• = • •

12/62

NAIST

#### What is RE?

Several detailed definitions depending on emphasis, but in short:

Requirements Engineering

Systematic handling of requirements. (SWEBOK v3 [2014])

More precisely, we can speak about systematic methods to discover, model, improve, and exploit requirements.

#### RE in Software Projects - In Theory



Diagram by Peter Kemp & Paul Smith (Wikimedia).

Introduction to Req	uirements Engineering	DUBLICDOMAN 13/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

#### RE in Software Projects - In Theory



After first step, requirements don't change anymore. The rest of the work is about satisfying them as best as possible.

Diagram by Peter Kemp & Paul Smith (Wikimedia).

Introduction to Requ	uirements Engineering	(0) PUBLICDOMAIN 13/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

#### RE in Software Projects - Wake up!

Assumptions:

All requirements are known after the first phase

Introduction to Requirements Engineering	(Ø) PUBLICDOMAIN	14/62
Matthieu Vergne vergne@is.naist.jp		NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

# RE in Software Projects - Wake up!

Assumptions:

- All requirements are known after the first phase
- Requirements are perfectly documented

Introduction to Req	uirements Engineering	14/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

# RE in Software Projects - Wake up!

Assumptions:

- All requirements are known after the first phase
- Requirements are perfectly documented
- If tests do not pass, the developer is at fault

Introduction to Req	uirements Engineering	14/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

# RE in Software Projects - Wake up!

Assumptions:

- All requirements are known after the first phase
- Requirements are perfectly documented
- If tests do not pass, the developer is at fault

Introduction to Req	uirements Engineering	(Ø) PUBLICDOMAIN	14/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

#### RE in Software Projects - Wake up!

Assumptions:

- All requirements are known after the first phase
- Requirements are perfectly documented
- If tests do not pass, the developer is at fault

Reality:

 New requirements discovered during implementation/verification

Introduction to Req	uirements Engineering	PUBLICDOMAIN	14/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

#### RE in Software Projects - Wake up!

Assumptions:

- All requirements are known after the first phase
- Requirements are perfectly documented
- If tests do not pass, the developer is at fault

- New requirements discovered during implementation/verification
- All requirements cannot be clarified immediately

Introduction to Requ	irements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆∃▶ ◆∃▶ ∃|目 のQの

#### RE in Software Projects - Wake up!

Assumptions:

- All requirements are known after the first phase
- Requirements are perfectly documented
- If tests do not pass, the developer is at fault

- New requirements discovered during implementation/verification
- All requirements cannot be clarified immediately
- Changes in requirements impact the rest of the chain

Introduction to Req	uirements Engineering	14/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

(0) PUBLICDOMAIN

14/62 NAIST

#### RE in Software Projects - Wake up!

Assumptions:

- All requirements are known after the first phase
- Requirements are perfectly documented
- If tests do not pass, the developer is at fault

- New requirements discovered during implementation/verification
- All requirements cannot be clarified immediately
- Changes in requirements impact the rest of the chain
- Later the change, higher the cost

Introduction to Rec	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	

イロト イヨト イヨト イヨト ショー クタイ

#### RE in Software Projects - In Practice

#### **Iterative Development**

Business value is delivered incrementally in time-boxed cross-discipline iterations.



#### Diagram by Dutchguilder (Wikimedia).

Introduction to Requ	uirements Engineering	PUBLICDOMAIN	15/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

## RE in Software Projects - In Practice

#### **Iterative Development**



Business value is delivered incrementally in time-boxed cross-discipline iterations.

> In iterative and incremental software development, RE runs through the whole project.

#### Diagram by Dutchguilder (Wikimedia).

Introduction to Req	uirements Engineering	PUBLICDOMAIN	15/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

#### Successful RE is Knowledge-Intensive

Having requirements is not enough:

 $\blacksquare$  Unclear requirements  $\rightarrow$  Plan discussions

Introduction to Red	uirements Engineering	16/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

## Successful RE is Knowledge-Intensive

Having requirements is not enough:

- Unclear requirements  $\rightarrow$  Plan discussions
- Few requirements  $\rightarrow$  Expect new requests

Introduction to Req	uirements Engineering		16/62
Matthieu Vergne	vergne@is.naist.jp	N	AIST

#### Successful RE is Knowledge-Intensive

Having requirements is not enough:

- Unclear requirements  $\rightarrow$  Plan discussions
- Few requirements  $\rightarrow$  Expect new requests
- Lot of requirements  $\rightarrow$  Prioritise them

Introduction to Requ	uirements Engineering		/62
Matthieu Vergne	vergne@is.naist.jp	NAI	SТ

## Successful RE is Knowledge-Intensive

Having requirements is not enough:

- Unclear requirements  $\rightarrow$  Plan discussions
- Few requirements  $\rightarrow$  Expect new requests
- Lot of requirements  $\rightarrow$  Prioritise them
- $\blacksquare$  Conflicts may occur  $\rightarrow$  Agree on priority criteria

Introduction to Req	uirements Engineering		16/62
Matthieu Vergne	vergne@is.naist.jp	1	NAIST

## Successful RE is Knowledge-Intensive

Having requirements is not enough:

- Unclear requirements  $\rightarrow$  Plan discussions
- Few requirements  $\rightarrow$  Expect new requests
- Lot of requirements  $\rightarrow$  Prioritise them
- Conflicts may occur  $\rightarrow$  Agree on priority criteria

Etc.

			_	
Introduction to Requ	irements Engineering		MAIN	16/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

# Successful RE is Knowledge-Intensive

Having requirements is not enough:

- Unclear requirements  $\rightarrow$  Plan discussions
- Few requirements  $\rightarrow$  Expect new requests
- Lot of requirements  $\rightarrow$  Prioritise them
- $\blacksquare$  Conflicts may occur  $\rightarrow$  Agree on priority criteria

Etc.

RE needs a lot of domain knowledge: find a balance between certainty and flexibility.

Introduction to Rec	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

Agile methods on a game is feasible (e.g. extension packs)

Introduction to Red	uirements Engineering	(Ø) PUBLICDOMAIN	17/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Introduction to Req	uirements Engineering	17/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Some projects are more certain by nature:

			2.25
Introduction to Rec	uirements Engineering		7/62
Matthieu Vergne	vergne@is.naist.jp	N.	AIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Some projects are more certain by nature:

An aircraft is constrained by physics and regulation

Introduction to Req	uirements Engineering		17/62
Matthieu Vergne	vergne@is.naist.jp	N	AIST

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Some projects are more certain by nature:

- An aircraft is constrained by physics and regulation
- While a game is a fully creative process

Introduction to Requ	uirements Engineering	(0) PUBLICDOMAIN	17/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Some projects are more certain by nature:

- An aircraft is constrained by physics and regulation
- While a game is a fully creative process

But it also depends on decision makers:

Introduction to Rec	uirements Engineering		/62
Matthieu Vergne	vergne@is.naist.jp	NAI	IST

< 口 > < 同 >

글 에 에 글 에 크

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Some projects are more certain by nature:

- An aircraft is constrained by physics and regulation
- While a game is a fully creative process

But it also depends on decision makers:

Is it really necessary?

イロト イヨト イヨト イヨト ショウ

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Some projects are more certain by nature:

- An aircraft is constrained by physics and regulation
- While a game is a fully creative process

But it also depends on decision makers:

- Is it really necessary?
- For how much cost/benefice in money/time/...?

Introduction to Req	uirements Engineering	17/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト イヨト イヨト イヨト ショウ

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Some projects are more certain by nature:

- An aircraft is constrained by physics and regulation
- While a game is a fully creative process

But it also depends on decision makers:

- Is it really necessary?
- For how much cost/benefice in money/time/...?
- With which impact on our image?

Introduction to Requ	irements Engineering	17/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト イヨト イヨト イヨト ショウ

#### Certainty/Flexibility Depends on Projects

Some projects are more flexible by nature:

- Agile methods on a game is feasible (e.g. extension packs)
- But costly for a critical aircraft or hospital device

Some projects are more certain by nature:

- An aircraft is constrained by physics and regulation
- While a game is a fully creative process

But it also depends on decision makers:

- Is it really necessary?
- For how much cost/benefice in money/time/...?
- With which impact on our image?

#### Etc.

Introduction to Requirements Engineering		0 PUBLICDOMAIN	17/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements	

#### RE in a Nutshell

RE is knowledge-intensive

		0.40
Introduction to Req	uirements Engineering	18/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

<ロ> <四> < 回> < 回> < 回> < 回> < 回> < 回</p>

- RE is knowledge-intensive
- RE is not a one shot activity, but a continuous one

Introduction to Rec	quirements Engineering	(Ø) PUBLICDOMAIN	18/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

- RE is knowledge-intensive
- RE is not a one shot activity, but a continuous one
- Requirements can be handled in many ways

Introduction to Req	uirements Engineering	18/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

- RE is knowledge-intensive
- RE is not a one shot activity, but a continuous one
- Requirements can be handled in many ways
- Requirements engineers should adapt to project and people

Introduction to Requirements Engineering		
Matthieu Vergne	vergne@is.naist.jp	NAIST

- RE is knowledge-intensive
- RE is not a one shot activity, but a continuous one
- Requirements can be handled in many ways
- Requirements engineers should adapt to project and people
- Doing so requires familiarity with various activities:
  - Requirements elicitation
  - Requirements modelling
  - Requirements analysis
  - Requirements prioritisation
  - Requirements management
  - Etc.

Introduction to Requ	uirements Engineering	PUBLICDOMAIN	18/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

## Outline

#### 1 Requirements

- 2 Requirements Engineering
- 3 RE Activities
  - Requirements Elicitation
  - Requirements Modelling
  - Requirements Analysis
  - Requirements Prioritisation
  - Requirements Management

#### 4 Conclusion

			240
Introduction to Req	uirements Engineering	(0) PUBLICDOMAN 1	9/62
Matthieu Vergne	vergne@is.naist.jp	N/	AIST
#### Requirements Elicitation



### 1 Requirements

2 Requirements Engineering

### 3 RE Activities

### Requirements Elicitation

- Requirements Modelling
- Requirements Analysis
- Requirements Prioritisation
- Requirements Management

### 4 Conclusion

		1040
Introduction to Requ	uirements Engineering	20/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion 0000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シスペ

Requirements Elicitation

### **Requirements Elicitation**

Goal: Gather new or revised requirements.

Introduction to Rec	uirements Engineering	(0) PUBLICDOMAIN	21/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

RE Activities

Conclusion 0000

Requirements Elicitation

### **Requirements Elicitation**

Goal: Gather new or revised requirements.

			040
Introduction to Requ	uirements Engineering	(0) PUBLICDOMAIN	21/62
Matthieu Vergne	vergne@is.naist.jp	N	IAIST

Requirements Engineering

RE Activities

Conclusion

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Elicitation

### **Requirements Elicitation**

Goal: Gather new or revised requirements.

Main tasks:

Identify stakeholders

Introduction to Req	uirements Engineering	21/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Elicitation

### **Requirements Elicitation**

Goal: Gather new or revised requirements.

- Identify stakeholders
- Understand their goals

Introduction to Req	uirements Engineering	21/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

Requirements Elicitation

### **Requirements Elicitation**

Goal: Gather new or revised requirements.

- Identify stakeholders
- Understand their goals
- Understand domain and environment (system-as-is)

Introduction to Rea	quirements Engineering	(0) PUBLICDOMAN 21	L/62
Matthieu Vergne	vergne@is.naist.jp	NA	IST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Elicitation

## **Requirements Elicitation**

Goal: Gather new or revised requirements.

- Identify stakeholders
- Understand their goals
- Understand domain and environment (system-as-is)
- Draw requirements (system-to-be)

			_	
Introduction to Requ	uirements Engineering		MAN	21/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

Requirements Engineering

< 口 > < 同 >

Requirements Elicitation

## **Requirements Elicitation**

Goal: Gather new or revised requirements.

- Identify stakeholders
- Understand their goals
- Understand domain and environment (system-as-is)
- Draw requirements (system-to-be)
- Document for reuse

Requirements Engineering

RE Activities

Conclusion 0000

◆□ > ◆□ > ◆三 > ◆三 > 三日 のへで

Requirements Elicitation

### **Requirements Elicitation**

Introduction to Requ	irements Engineering	22/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering 0000000 RE Activities

Conclusion

Requirements Elicitation

## **Requirements Elicitation**

Various stakeholders:

Customers

		-		
Introduction to Requireme	ents Engineering		(0) PUBLICDOMAIN	22/62
Matthieu Vergne verg	gne@is.naist.jp			NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Elicitation

## **Requirements Elicitation**

- Customers
- Final users

			2.40
Introduction to Req	uirements Engineering		22/62
Matthieu Vergne	vergne@is.naist.jp	1	NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Elicitation

### Requirements Elicitation

- Customers
- Final users
- Domain experts

Introduction to Rec	juirements Engineering	22/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Elicitation

### **Requirements Elicitation**

- Customers
- Final users
- Domain experts
- Regulatory authorities

Introduction to Requ	irements Engineering	22/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Elicitation

## **Requirements Elicitation**

- Customers
- Final users
- Domain experts
- Regulatory authorities
- Developers

Introduction to Requ	uirements Engineering	DUBLICDOMAIN	22/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Elicitation

## **Requirements Elicitation**

- Customers
- Final users
- Domain experts
- Regulatory authorities
- Developers
- Etc.

Introduction to Requ	uirements Engineering	PUBLICDOMAIN	22/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Elicitation

## **Requirements Elicitation**

### Various stakeholders:

- Customers
- Final users
- Domain experts
- Regulatory authorities
- Developers
- Etc.

Introduction to Req	uirements Engineering	22/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Elicitation

## **Requirements Elicitation**

### Various stakeholders:

- Customers
- Final users
- Domain experts
- Regulatory authorities
- Developers
- Etc.

Other sources can help:

Existing specifications

Introduction to Req	uirements Engineering	22/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Elicitation

## **Requirements Elicitation**

### Various stakeholders:

- Customers
- Final users
- Domain experts
- Regulatory authorities
- Developers
- Etc.

- Existing specifications
- Similar projects

			2.40
Introduction to Requ	irements Engineering		22/62
Matthieu Vergne	vergne@is.naist.jp	N	IAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Elicitation

# **Requirements Elicitation**

### Various stakeholders:

- Customers
- Final users
- Domain experts
- Regulatory authorities
- Developers
- Etc.

- Existing specifications
- Similar projects
- Standards

Introduction to Requ	uirements Engineering	22/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Elicitation

# **Requirements Elicitation**

### Various stakeholders:

- Customers
- Final users
- Domain experts
- Regulatory authorities
- Developers
- Etc.

- Existing specifications
- Similar projects
- Standards
- Etc.

Introduction to Req	uirements Engineering	PUBLICDOMAIN	22/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

Requirements Elicitation

### Elicitation Technique - Interviews/Questionnaires

Goal: Gather information and opinions from isolated stakeholders.

Introduction to Re	quirements Engineering	23/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

#### Requirements Elicitation

### Elicitation Technique - Interviews/Questionnaires

Goal: Gather information and opinions from isolated stakeholders.

Pros:

- Direct feedback with personal perspectives
- Can be close (strictly follow questions) or open (adapt questions on the fly)
- Isolation allows customisation

Introduction to Rec	uirements Engineering	23/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆∃▶ ◆∃▶ ∃|目 のQの

#### Requirements Elicitation

### Elicitation Technique - Interviews/Questionnaires

Goal: Gather information and opinions from isolated stakeholders.

Pros:

- Direct feedback with personal perspectives
- Can be close (strictly follow questions) or open (adapt questions on the fly)
- Isolation allows customisation

Cons:

- Driven by interviewer more than stakeholders
- Does not exploit synergies (isolated interviews)
- Different interviews may contradict each other

Introduction to Requ	irements Engineering	23/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Elicitation

## Elicitation Technique - Meeting/Focus Group

Goal: Gather information and opinions from groups of stakeholders.

Introduction to Re	quirements Engineering	24/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Elicitation

# Elicitation Technique - Meeting/Focus Group

Goal: Gather information and opinions from groups of stakeholders.

Pros:

- Exploit synergies (e.g. brainstorming)
- Help to identify conflicts and agreements

Introduction to Requ	uirements Engineering		62
Matthieu Vergne	vergne@is.naist.jp	NAIS	SТ

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Elicitation

# Elicitation Technique - Meeting/Focus Group

Goal: Gather information and opinions from groups of stakeholders.

Pros:

- Exploit synergies (e.g. brainstorming)
- Help to identify conflicts and agreements

Cons:

- Discussions can be limited by power (leaders vs. subordinates)
- Discussions can be limited by personality (extroverts vs. introverts)
- Cannot involve too many stakeholders
- Hard to apply for timely/geographically spread teams

Introduction to Req	uirements Engineering	PUBLICDOMAIN	24/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

RE Activities

Conclusion 0000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

**Requirements Elicitation** 

## Elicitation Technique - Scenarios/Use Cases/Personas

Goal: Draw behavioural patterns.

Introduction to Red	uirements Engineering	25/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion 0000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Elicitation

## Elicitation Technique - Scenarios/Use Cases/Personas

Goal: Draw behavioural patterns.

Pros:

- Helps to understand the system-as-is
- Easy to get from domain experts

Introduction to Requ	uirements Engineering	(0) PUBLICOOMAN	25/62
Matthieu Vergne	vergne@is.naist.jp	l i i i i i i i i i i i i i i i i i i i	NAIST

Requirements Engineering

< 口 > < 同 >

∃ ► < ∃ ►</p>

Requirements Elicitation

# Elicitation Technique - Scenarios/Use Cases/Personas

Goal: Draw behavioural patterns.

Pros:

- Helps to understand the system-as-is
- Easy to get from domain experts

Cons:

- Arguable for describing the system-to-be (bias)
- Not easy to identify inter-dependencies (isolated behaviours)

Requirements Engineering

RE Activities

Conclusion 0000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シスペ

**Requirements Elicitation** 

### Elicitation Technique - Ethnographic Study

Goal: Analyse people's behaviours on site.

Introduction to Rec	uirements Engineering	PUBLICDOMAIN	26/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

#### Requirements Elicitation

## Elicitation Technique - Ethnographic Study

Goal: Analyse people's behaviours on site.

Pros:

- Direct observation of actual behaviours
- Can observe subtle behaviours and variants
- Can ask for explanations on the fly

Introduction to Req	uirements Engineering	26/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

#### Requirements Elicitation

## Elicitation Technique - Ethnographic Study

Goal: Analyse people's behaviours on site.

Pros:

- Direct observation of actual behaviours
- Can observe subtle behaviours and variants
- Can ask for explanations on the fly

Cons:

- Impact on the workflow might be significant
- Require a lot of time
- Not applicable to all tasks (e.g. emergency)
- Rare but important behaviours might be missing

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements	Requirements Engineering	RE Activities
		000000000000000000000000000000000000000

イロト イヨト イヨト イヨト ショー クタイ

Requirements Elicitation



Requirements elicitation is a hard task

Introduction to Requ	uirements Engineering		27/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

#### Requirements Elicitation



- Requirements elicitation is a hard task
- Subject to communication failures
  - Stakeholders can be unavailable
  - Stakeholders can overlook relevant details
  - Stakeholders can be unable to explain themselves
  - Stakeholders can disagree with each others
  - Stakeholders can be unwilling to share or lie

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

#### Requirements Elicitation



- Requirements elicitation is a hard task
- Subject to communication failures
  - Stakeholders can be unavailable
  - Stakeholders can overlook relevant details
  - Stakeholders can be unable to explain themselves
  - Stakeholders can disagree with each others
  - Stakeholders can be unwilling to share or lie
- Other techniques not presented:
  - Prototypes, user stories, competitors' analysis, data mining, etc.

Introduction to Requirem	ents Engineering		DOMAIN	27/62
Matthieu Vergne ver	gne@is.naist.jp			NAIST

Requirements Engineering

< D > < A >

-

#### Requirements Elicitation



- Requirements elicitation is a hard task
- Subject to communication failures
  - Stakeholders can be unavailable
  - Stakeholders can overlook relevant details
  - Stakeholders can be unable to explain themselves
  - Stakeholders can disagree with each others
  - Stakeholders can be unwilling to share or lie
- Other techniques not presented:
  - Prototypes, user stories, competitors' analysis, data mining, etc.
- Different techniques can be complementary, but come with their own costs
Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

#### Requirements Elicitation



- Requirements elicitation is a hard task
- Subject to communication failures
  - Stakeholders can be unavailable
  - Stakeholders can overlook relevant details
  - Stakeholders can be unable to explain themselves
  - Stakeholders can disagree with each others
  - Stakeholders can be unwilling to share or lie
- Other techniques not presented:
  - Prototypes, user stories, competitors' analysis, data mining, etc.
- Different techniques can be complementary, but come with their own costs
- Choosing the right techniques, applied with the right amounts, and combined in the right way is a skill by itself

Introduction to Requ	uirements Engineering	DUBLICDOMAN 27/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

≡ •ી લ.ઉ 28/62 NAIST

#### Requirements Modelling

### Outline

### 1 Requirements

2 Requirements Engineering

### 3 RE Activities

Requirements Elicitation

### Requirements Modelling

- Requirements Analysis
- Requirements Prioritisation
- Requirements Management

### 4 Conclusion

Introduction to Requ	irements Engineering	
Matthieu Vergne	vergne@is.naist.jp	

Requirements Engineering

◆□ > ◆□ > ◆三 > ◆三 > 三日 のへで

Requirements Modelling

### Requirements Modelling

Goal: Represent relevant requirements properties.

Introduction to Rec	uirements Engineering	29/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering 0000000

Requirements Modelling

### Requirements Modelling

Goal: Represent relevant requirements properties.

Main tasks:

Introduction to Rec	uirements Engineering		/62
Matthieu Vergne	vergne@is.naist.jp	NAI	SΤ

Requirements Engineering 0000000

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Modelling

### **Requirements Modelling**

Goal: Represent relevant requirements properties.

Main tasks:

Identify domain terminology

Introduction to Requ	irements Engineering	(0) PUBLICDOMAIN	29/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering 0000000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Modelling

### **Requirements Modelling**

Goal: Represent relevant requirements properties.

Main tasks:

- Identify domain terminology
- Structure concepts and relationships
  - System, users, goals, interactions, resources, etc.

Introduction to Req	uirements Engineering	29/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering 0000000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Modelling

### **Requirements Modelling**

Goal: Represent relevant requirements properties.

Main tasks:

- Identify domain terminology
- Structure concepts and relationships
  - System, users, goals, interactions, resources, etc.
- Select types/levels of formalisation

Introduction to Req	uirements Engineering	PUBLICDOMAN	29/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering 0000000 RE Activities

Conclusion 0000

イロト イヨト イヨト イヨト ショー クタイ

Requirements Modelling

### Requirements Modelling

Visual models:

Introduction to Red	uirements Engineering	30/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Modelling

### Requirements Modelling

Visual models:

Help to understand/explain the domain

Introduction to Requ	uirements Engineering		DOMAIN	30/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

Requirements Engineering

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Modelling

### **Requirements Modelling**

Visual models:

- Help to understand/explain the domain
- Help to communicate with stakeholders

Introduction to Req	uirements Engineering	(O) PUBLICDOMAIN	30/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Modelling

### **Requirements Modelling**

Visual models:

- Help to understand/explain the domain
- Help to communicate with stakeholders
- Help to build requirements documents

Introduction to Req	uirements Engineering	O PUBLICDOMAN 3	0/62
Matthieu Vergne	vergne@is.naist.jp	NA	AIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Modelling

### **Requirements Modelling**

Visual models:

- Help to understand/explain the domain
- Help to communicate with stakeholders
- Help to build requirements documents

Formal models:

Introduction to Req	uirements Engineering	() PUBLICDOMAN	30/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Modelling

### **Requirements Modelling**

Visual models:

- Help to understand/explain the domain
- Help to communicate with stakeholders
- Help to build requirements documents

Formal models:

Help to spot errors/inconsistencies

Introduction to Requ	uirements Engineering		DOMAIN	30/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

Requirements Modelling

### **Requirements Modelling**

Visual models:

- Help to understand/explain the domain
- Help to communicate with stakeholders
- Help to build requirements documents

Formal models:

- Help to spot errors/inconsistencies
- Help to scale requirements

			1940
Introduction to Req	uirements Engineering		30/62
Matthieu Vergne	vergne@is.naist.jp	1	VAIST

Requirements Modelling

### **Requirements Modelling**

Visual models:

- Help to understand/explain the domain
- Help to communicate with stakeholders
- Help to build requirements documents

Formal models:

- Help to spot errors/inconsistencies
- Help to scale requirements
- Allows for automated reasoning

・ロト ・ 日 ・ ・ ヨ ト ・ 日 ト ・ の へ の

Requirements Modelling

## Requirements Modelling

Visual models:

- Help to understand/explain the domain
- Help to communicate with stakeholders
- Help to build requirements documents

Formal models:

- Help to spot errors/inconsistencies
- Help to scale requirements
- Allows for automated reasoning

A model can be both visual and formal.

Introduction to Req	uirements Engineering	(0) PUBLICDOMAN 3	30/62
Matthieu Vergne	vergne@is.naist.jp	N.	AIST

Requirements Engineering

Requirements Modelling

# Modelling Technique - Unified Modelling Language (UML)

Goal: Provide a standard way to visualise the design of a software system.

Source: http://www.uml.org, Picture: Kishorekumar 62 (Wikimedia)

Introduction to Requ	uirements Engineering	DUBLICOOMAN 31/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Modelling

# Modelling Technique - Unified Modelling Language (UML)

Goal: Provide a standard way to visualise the design of a software system.

Pros:

- Cover several aspects of software design
  - e.g. Use case diagrams can model goals:
- Broadly used standard
- Many tools available



Source: http://www.uml.org, Picture: Kishorekumar 62 (Wikimedia)

Introduction to Requ	uirements Engineering	31/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Modelling

# Modelling Technique - Unified Modelling Language (UML)

Goal: Provide a standard way to visualise the design of a software system.

Pros:

- Cover several aspects of software design
  - e.g. Use case diagrams can model goals:
- Broadly used standard
- Many tools available

Cons

- Designed for software modelling, not RE
- Miss relevant features for RE

Source: http://www.uml.org, Picture: Kishorekumar 62 (Wikimedia)



(0) PUBLICDOMAIN

31/62

NAIST

Introduction to Requirements Engineering Matthieu Vergne



Requirements Engineering

Requirements Modelling

## Modelling Technique - i\* (i Star)

Goal: Represent stakeholders' goals and how they relate to each other and to the system.



Introduction to Requ	irements Engineering	(0) PUBLICOOMAN 32/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Modelling

## Modelling Technique - i\* (i Star)

Goal: Represent stakeholders' goals and how they relate to each other and to the system.



Source: i\* Wiki (http://istar.rwth-aachen.de)

Introduction to Requ	uirements Engineering	O PUBLICDOMAIN 32/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Modelling

# Modelling Technique - i\* (i Star)

Goal: Represent stakeholders' goals and how they relate to each other and to the system.



Pros: Offer a more comprehensive representation of stakeholders and their interactions with the system

Actors, goals, tasks, resources, dependencies, etc.

Source: i	i* Wiki	(http:/	/istar	.rwth-aa	chen.d	le)	)										
						- I		•	< A	4.3	E 🖌	- A - E	£ ⊳.	315	. 4	na	0

Introduction to Requ	uirements Engineering	32/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Modelling

# Modelling Technique - i\* (i Star)

Goal: Represent stakeholders' goals and how they relate to each other and to the system.



Pros: Offer a more comprehensive representation of stakeholders and their interactions with the system

Actors, goals, tasks, resources, dependencies, etc.

Cons: Limited to general aspects (mitigated by variants)

Source: i* V	Viki (http://is	tar.rwth-aachen	.de	)										
					κ.	< A	1 B	<ul> <li>E</li> </ul>	•	< E.	•	315	50	0

ntroduction to Req	uirements Engineering	32/62
/latthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering 0000000

Requirements Modelling

### Modelling Technique - Gherkin

Goal: Enforce firm, unambiguous requirements.

```
Given the car is a <type>
When we calculate the price
Then the price should be $<price>
```

Examples:

	type		price	
T	Ferrari		1,000,000	

Requirements Engineering 0000000 RE Activities

Conclusion 0000

Requirements Modelling

### Modelling Technique - Gherkin

Goal: Enforce firm, unambiguous requirements. Given the car is a <type> When we calculate the price Then the price should be \$<price>

◆□ > ◆□ > ◆三 > ◆三 > 三日 のへで

Examples:

| type | price | | Ferrari | 1,000,000 |

Introduction to Req	irements Engineering	(0) PUBLICDOMAIN	33/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Modelling

### Modelling Technique - Gherkin

Goal: Enforce firm, unambiguous requirements. Given the car is a <type> When we calculate the price Then the price should be \$<price>

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Examples: | type | price | | Ferrari | 1,000,000 |

Pros:

- Balance formal and informal specifications
- Feature acceptance test generation
  - Behaviour Driven Development, Test Driven Development
- Support many natural + programming languages

Introduction to Requ	irements Engineering	(0)	PUBLICDOMAIN	33/62
Matthieu Vergne	vergne@is.naist.jp		N	IAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Modelling

# Modelling Technique - Gherkin

Goal: Enforce firm, unambiguous requirements. Given the car is a <type> When we calculate the price Then the price should be \$<price>

Examples: | type | price | | Ferrari | 1,000,000 |

Pros:

- Balance formal and informal specifications
- Feature acceptance test generation
  - Behaviour Driven Development, Test Driven Development
- Support many natural + programming languages

Cons:

- Not RE method, tool only
- Not applicable to vague reqs.

Introduction to Requ	uirements Engineering	DUBLICDOMAIN	33/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

< 口 > < 同 >

34/62

Requirements Modelling

# Modelling Technique - Boolean Equations

Goal: Represent requirements as logical formula to satisfy.

customer(Alice) seller(Bob) product(Car)  $\forall c, customer(c) \Rightarrow \exists (p, s), seller(s) \land product(p) \land buy(c, p, s)$ 

Introduction to Req	uirements Engineering
Matthieu Vergne	vergne@is_naist_in

Requirements Engineering

RE Activities

Conclusion

Requirements Modelling

### Modelling Technique - Boolean Equations

Goal: Represent requirements as logical formula to satisfy.

customer(Alice) seller(Bob) product(Car)  $\forall c, customer(c) \Rightarrow \exists (p, s), seller(s) \land$   $product(p) \land buy(c, p, s)$ 

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Modelling

# Modelling Technique - Boolean Equations

Goal: Represent requirements as logical formula to satisfy.

```
customer(Alice)
seller(Bob)
product(Car)
\forall c, customer(c) \Rightarrow \exists (p, s), seller(s) \land
product(p) \land buy(c, p, s)
```

Pros:

- Allow to use formal reasoning (e.g. SAT solvers)
- Exact meaning (no ambiguity)

			-) 40
Introduction to Requ	irements Engineering	(O) PUBLICDOMAN	34/62
Matthieu Vergne	vergne@is.naist.jp	1	JAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Modelling

# Modelling Technique - Boolean Equations

Goal: Represent requirements as logical formula to satisfy.

```
customer(Alice)
seller(Bob)
product(Car)
\forall c, customer(c) \Rightarrow \exists (p, s), seller(s) \land
product(p) \land buy(c, p, s)
```

・ロト ・ 日 ・ ・ ヨ ト ・ 日 ト ・ の へ の

Pros:

- Allow to use formal reasoning (e.g. SAT solvers)
- Exact meaning (no ambiguity)

Cons:

- Hard to understand for non-logicians
- Not applicable to vague requirements

Introduction to Requ	irements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

Requirements Modelling



Different Models emphasise different properties

		- 240
Introduction to Req	uirements Engineering	35/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Modelling



- Different Models emphasise different properties
- Various techniques exist, not only the ones presented:
  - Data Flow Diagram and RML, KAOS, extensions of i\* (e.g. Tropos for multi-agents, Nòmos for laws, Zanshin [残心] for self-adaptation), ReqIF & OSLC standards, etc.

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

< 口 > < 同 >

= 990

36/62 NAIST

#### Requirements Analysis

### Outline

### 1 Requirements

2 Requirements Engineering

### 3 RE Activities

- Requirements Elicitation
- Requirements Modelling
- Requirements Analysis
- Requirements Prioritisation
- Requirements Management

### 4 Conclusion

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	

Requirements Engineering 0000000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Analysis

### Requirements Analysis

Goal: Validate requirements through their model.

Introduction to Req	uirements Engineering	PUBLICDOMAIN	37/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering 0000000

Requirements Analysis

### Requirements Analysis

Goal: Validate requirements through their model.

Main tasks:

		-) 40-
Introduction to Req	uirements Engineering	37/62
Matthieu Vergne	vergne@is.naist.jp	NAIST
Requirements Engineering 0000000

Requirements Analysis

### Requirements Analysis

Goal: Validate requirements through their model.

Main tasks:

Identify & fix inconsistencies

Introduction to Requ	uirements Engineering		OMAIN	37/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

Requirements Engineering 0000000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Analysis

### Requirements Analysis

Goal: Validate requirements through their model.

- Identify & fix inconsistencies
- Identify & fix incompleteness

Introduction to Requ	irements Engineering	37/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering 0000000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Analysis

## Requirements Analysis

Goal: Validate requirements through their model.

- Identify & fix inconsistencies
- Identify & fix incompleteness
- Identify & fix ambiguities

Introduction to Req	uirements Engineering	(Ø) PUBLICDOMAIN	37/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering 0000000

Requirements Analysis

## Requirements Analysis

Goal: Validate requirements through their model.

- Identify & fix inconsistencies
- Identify & fix incompleteness
- Identify & fix ambiguities
- Risk analysis

		2.40
roduction to Req	uirements Engineering	37/62
atthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering 0000000

ъ.

Requirements Analysis

## Requirements Analysis

Goal: Validate requirements through their model.

- Identify & fix inconsistencies
- Identify & fix incompleteness
- Identify & fix ambiguities
- Risk analysis
- Evaluate alternatives

Requirements Engineering

RE Activities

Conclusion 0000

Requirements Analysis

### Requirements Analysis

			1040
Introduction to Req	uirements Engineering	(0) PUBLICDOMAIN	38/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Analysis

#### Requirements Analysis

Informal analysis:

Fit with natural language and visual models

Introduction to Req	uirements Engineering	PUBLICDOMAN	38/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering 0000000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Analysis

## Requirements Analysis

- Fit with natural language and visual models
- Exploit the flexibility of human reasoning

Introduction to Reg	Jirements Engineering	38/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Analysis

## Requirements Analysis

- Fit with natural language and visual models
- Exploit the flexibility of human reasoning
- Good for qualitative analysis

Introduction to Requ	uirements Engineering	DUBLICDOMAIN	38/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Analysis

## Requirements Analysis

- Fit with natural language and visual models
- Exploit the flexibility of human reasoning
- Good for qualitative analysis
- Help for formal modelling/analysis

Introduction to Requ	irements Engineering		38/62
Matthieu Vergne	vergne@is.naist.jp	Ν	JAIST

Requirements Analysis

## Requirements Analysis

Informal analysis:

- Fit with natural language and visual models
- Exploit the flexibility of human reasoning
- Good for qualitative analysis
- Help for formal modelling/analysis

Formal analysis:

Introduction to Req	uirements Engineering	O PUBLICDOMAN 38/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Analysis

## Requirements Analysis

Informal analysis:

- Fit with natural language and visual models
- Exploit the flexibility of human reasoning
- Good for qualitative analysis
- Help for formal modelling/analysis

Formal analysis:

Fit with formal models

Introduction to Requirements	Engineering		(0) РИВЦСО	OMAIN	38/62
Matthieu Vergne vergne@	is.naist.jp				NAIST

◆□▶ ◆□▶ ◆∃▶ ◆∃▶ ∃|目 のQの

Requirements Analysis

## Requirements Analysis

Informal analysis:

- Fit with natural language and visual models
- Exploit the flexibility of human reasoning
- Good for qualitative analysis
- Help for formal modelling/analysis

Formal analysis:

- Fit with formal models
- Exploit the exactitude of formal reasoning

Introduction to Requ	uirements Engineering	(0) PUBLICDOMAIN	38/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Analysis

## Requirements Analysis

Informal analysis:

- Fit with natural language and visual models
- Exploit the flexibility of human reasoning
- Good for qualitative analysis
- Help for formal modelling/analysis

Formal analysis:

- Fit with formal models
- Exploit the exactitude of formal reasoning
- Good for quantitative analysis

Introduction to Requ	uirements Engineering	DUBLICDOMAIN	38/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Analysis

## Analysis Technique - Manual Analysis

Goal: Rely on stakeholders to analyse and refine the models.

Introduction to Red	uirements Engineering	PUBLICDOMAIN	39/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Analysis

# Analysis Technique - Manual Analysis

Goal: Rely on stakeholders to analyse and refine the models.

Pros:

- Build on expertise of the stakeholders
- Requirements engineer acts as a facilitator
- Good when the models are still small

Introduction to Req	uirements Engineering	(O) PUBLICDOMAIN	39/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

イロト イヨト イヨト イヨト ショウ

Requirements Analysis

## Analysis Technique - Manual Analysis

Goal: Rely on stakeholders to analyse and refine the models.

Pros:

- Build on expertise of the stakeholders
- Requirements engineer acts as a facilitator
- Good when the models are still small

Cons:

- Lack systematicity (no guarantee of success)
- Hard to maintain when the models become complex

Introduction to Requ	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Analysis

## Analysis Technique - Formal Tropos

Goal: Translate goal-models into logical formulae for formal analysis.



Requirements Engineering

RE Activities

Conclusion

Requirements Analysis

# Analysis Technique - Formal Tropos

Goal: Translate goal-models into logical formulae for formal analysis.



イロト イポト イヨト イヨト

315

#### Pictures: reworked from Anna Perini

Introduction to Requirements Engineering 00 vergee@is.naist.jp 00 VALCOUND 40/62

Requirements Engineering

RE Activities

Conclusion

Requirements Analysis

## Analysis Technique - Formal Tropos

Goal: Translate goal-models into logical formulae for formal analysis.



Pros:

- Consistency proven or conflicts spotted
- Interesting properties can be proven too
- Can play with the model to analyse different scenarios

Pictures: reworked from Anna Perini		
	・ロト・(四ト・(日下・)田田	9 Q (P
Introduction to Requirements Engineering		40/62
Matthieu Vergne vergne@is.naist.jp		NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Analysis

# Analysis Technique - Formal Tropos

Goal: Translate goal-models into logical formulae for formal analysis.



イロト 不得 トイヨト イヨト ヨヨ ろくで

Pros:

- Consistency proven or conflicts spotted
- Interesting properties can be proven too
- Can play with the model to analyse different scenarios

Cons:

Int

- Not applicable to vague requirements
- Formula hard to understand for non-logicians

Pictures: reworked from Anna Perini

oduction to Requ	irements Engineering	40/62
tthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering 0000000

Requirements Analysis



#### Analysis depends on models

		1940
Introduction to Requ	uirements Engineering	41/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering 0000000

Requirements Analysis



- Analysis depends on models
- Carefully choose models for supporting analysis

Introduction to Requ	uirements Engineering		/62
Matthieu Vergne	vergne@is.naist.jp	NA	IST

< 口 > < 同 >

\* 3 > < 3</p>

三日 のへの

42/62 NAIST

(0) PUBLICDOMAIN

#### **Requirements** Prioritisation

## Outline

#### 1 Requirements

2 Requirements Engineering

#### 3 RE Activities

- Requirements Elicitation
- Requirements Modelling
- Requirements Analysis
- Requirements Prioritisation
- Requirements Management

#### 4 Conclusion

Introduction to Req	uirements Engineering
Matthieu Vergne	vergne@is.naist.jp

Requirements Engineering

RE Activities

Conclusion 0000

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

**Requirements** Prioritisation

#### **Requirements** Prioritisation

Goal: Select the most relevant requirements

Introduction to Rec	uirements Engineering	OPUBLICDOMAN 43/62	
Matthieu Vergne	vergne@is.naist.jp	NAIST	

Requirements Engineering

RE Activities

Conclusion 0000

**Requirements** Prioritisation

#### **Requirements** Prioritisation

Goal: Select the most relevant requirements

Introduction to Requirements Engineering		DOMAIN	43/62
Matthieu Vergne vergne@is.naist.jp			NAIST

Requirements Engineering

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Prioritisation

#### Requirements Prioritisation

Goal: Select the most relevant requirements

Main tasks:

Identify prioritisation criteria (e.g. cost/benefice, risks, popularity)

Introduction to Req	uirements Engineering	(0) PUBLICDOMAN	43/62
Matthieu Vergne	vergne@is.naist.jp	1	NAIST

Requirements Engineering

Requirements Prioritisation

#### **Requirements** Prioritisation

Goal: Select the most relevant requirements

- Identify prioritisation criteria (e.g. cost/benefice, risks, popularity)
- Identify constraints (e.g. deadlines, requirements dependencies)

			2.25
Introduction to Req	uirements Engineering		43/62
Matthieu Vergne	vergne@is.naist.jp	1	NAIST

Requirements Prioritisation

#### **Requirements** Prioritisation

Goal: Select the most relevant requirements

- Identify prioritisation criteria (e.g. cost/benefice, risks, popularity)
- Identify constraints (e.g. deadlines, requirements dependencies)
- Order requirements by priority

troduction to Requ	irements Engineering	43/62
latthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Prioritisation

#### **Requirements** Prioritisation

Goal: Select the most relevant requirements

- Identify prioritisation criteria (e.g. cost/benefice, risks, popularity)
- Identify constraints (e.g. deadlines, requirements dependencies)
- Order requirements by priority
- Filter out unwanted requirements

Introduction to Requ	irements Engineering	43/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion 0000

**Requirements** Prioritisation

#### **Requirements** Prioritisation

Mono vs. multi-objectives:

Introduction to Req	uirements Engineering	44/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

#### **Requirements** Prioritisation

Mono vs. multi-objectives:

 Mono: All criteria are reduced to 1, usually with a weighted average.

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Prioritisation

#### **Requirements** Prioritisation

Mono vs. multi-objectives:

- Mono: All criteria are reduced to 1, usually with a weighted average.
- Multi: Each criterion is evaluated separately, the best solutions form a Pareto front

Introduction to Rec	uirements Engineering	0 PUBLICDOMAIN 44	1/62
Matthieu Vergne	vergne@is.naist.jp	NA	IST

Requirements Prioritisation

#### **Requirements** Prioritisation

Mono vs. multi-objectives:

- Mono: All criteria are reduced to 1, usually with a weighted average.
- Multi: Each criterion is evaluated separately, the best solutions form a Pareto front

Human vs. automated decision:

Introduction to Req	uirements Engineering	44/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Prioritisation

#### **Requirements** Prioritisation

Mono vs. multi-objectives:

- Mono: All criteria are reduced to 1, usually with a weighted average.
- Multi: Each criterion is evaluated separately, the best solutions form a Pareto front

Human vs. automated decision:

Human: Exploit qualitative judgement and expertise of decision makers

Introduction to Requ	uirements Engineering	(0) PUBLICDOMAIN	44/62
Matthieu Vergne	vergne@is.naist.jp	N	VAIST

◆□▶ ◆□▶ ◆∃▶ ◆∃▶ ∃|∃ や��

Requirements Prioritisation

#### **Requirements** Prioritisation

Mono vs. multi-objectives:

- Mono: All criteria are reduced to 1, usually with a weighted average.
- Multi: Each criterion is evaluated separately, the best solutions form a Pareto front

Human vs. automated decision:

- Human: Exploit qualitative judgement and expertise of decision makers
- Automated: Exploit formalised quantitative evaluation

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST
Requirements Prioritisation

#### **Requirements** Prioritisation

Mono vs. multi-objectives:

- Mono: All criteria are reduced to 1, usually with a weighted average.
- Multi: Each criterion is evaluated separately, the best solutions form a Pareto front

Human vs. automated decision:

- Human: Exploit qualitative judgement and expertise of decision makers
- Automated: Exploit formalised quantitative evaluation
- Hybrid: Automated for preliminary + human for final decision

Introduction to Req	uirements Engineering		4/62
Matthieu Vergne	vergne@is.naist.jp	NA	AIST

Requirements Engineering

RE Activities

Conclusion 0000

イロト イヨト イヨト イヨト ショー クタイ

**Requirements** Prioritisation

## **Requirements** Prioritisation

Pairwise vs. order:

Introduction to Requ	irements Engineering	45/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

# **Requirements** Prioritisation

Pairwise vs. order:

Pairwise: Local decision comparing successively pairs of requirements

Introduction to Rec	quirements Engineering		45/62
Matthieu Vergne	vergne@is.naist.jp	Ν	IAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シスペ

Requirements Prioritisation

# Requirements Prioritisation

#### Pairwise vs. order:

- Pairwise: Local decision comparing successively pairs of requirements
- Order: Global decision

Introduction to Req	uirements Engineering	45/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

# Requirements Prioritisation

#### Pairwise vs. order:

- Pairwise: Local decision comparing successively pairs of requirements
- Order: Global decision

Introduction to Req	uirements Engineering	45/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Prioritisation

#### **Requirements** Prioritisation

Pairwise vs. order:

- Pairwise: Local decision comparing successively pairs of requirements
- Order: Global decision

Various methods:

Fully manual: Rely on stakeholders expertise.

Introduction to Req	uirements Engineering	45/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

#### **Requirements** Prioritisation

#### Pairwise vs. order:

- Pairwise: Local decision comparing successively pairs of requirements
- Order: Global decision

- Fully manual: Rely on stakeholders expertise.
- Machine learning: Evaluate new solutions based on already known ones.

Introduction to Requ	irements Engineering	45/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

#### **Requirements** Prioritisation

#### Pairwise vs. order:

- Pairwise: Local decision comparing successively pairs of requirements
- Order: Global decision

- Fully manual: Rely on stakeholders expertise.
- Machine learning: Evaluate new solutions based on already known ones.
- Search algorithms: Iteratively improve a (set of) solution(s) evaluated through a (set of) fitness function(s).

Introduction to Requ	irements Engineering	(0) PUBLICHOMAIN 45/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Prioritisation

#### **Requirements** Prioritisation

#### Pairwise vs. order:

- Pairwise: Local decision comparing successively pairs of requirements
- Order: Global decision

- Fully manual: Rely on stakeholders expertise.
- Machine learning: Evaluate new solutions based on already known ones.
- Search algorithms: Iteratively improve a (set of) solution(s) evaluated through a (set of) fitness function(s).
- Constraints satisfaction: Find solution(s) satisfying Boolean formulae.

Introduction to Requ	irements Engineering	45/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

◆□ > ◆□ > ◆三 > ◆三 > 三日 のへの

**Requirements** Prioritisation

## Prioritisation Technique - MoSCoW Prioritisation

Goal: Have stakeholders rank all requirements in 4 levels – must have, should have, could have, and won't have.

Introduction to Red	juirements Engineering	(0) PUBLICOOMAIN 46/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

# Prioritisation Technique - MoSCoW Prioritisation

Goal: Have stakeholders rank all requirements in 4 levels – must have, should have, could have, and won't have.

Pros:

- Levels are simple to understand for stakeholders
- Levels focus on practical priorities

Introduction to Req	uirements Engineering	46/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

イロト イタト イモト イモト 三日 のうう

Requirements Prioritisation

# Prioritisation Technique - MoSCoW Prioritisation

Goal: Have stakeholders rank all requirements in 4 levels – must have, should have, could have, and won't have.

Pros:

- Levels are simple to understand for stakeholders
- Levels focus on practical priorities

Cons:

- The ranking is extremely scarce for many requirements
- Lack of rationale behind each choice

		2.25
Introduction to Requ	irements Engineering	46/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

イロト イヨト イヨト イヨト ショー クタイ

Requirements Prioritisation

# Prioritisation Technique - Bubble Sort

Goal: Iteratively improve global order by swapping pairs of requirements.

Introduction to Req	uirements Engineering	47/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

# Prioritisation Technique - Bubble Sort

Goal: Iteratively improve global order by swapping pairs of requirements.

Pros:

- Simple to implement
- Simple to understand

Introduction to Requ	irements Engineering	(0) PUBLICDOMAN	47/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

Requirements Prioritisation

# Prioritisation Technique - Bubble Sort

Goal: Iteratively improve global order by swapping pairs of requirements.

Pros:

- Simple to implement
- Simple to understand

Cons:

- Require a lot of pair comparisons  $(O(n^2))$
- No way to tell that a requirement is "far" below/above another

			_	
Introduction to Requ	uirements Engineering		AN	47/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シスペ

Requirements Prioritisation

# Prioritisation Technique - Analytic Hierarchy Process (AHP)

Goal: Build a matrix of requirements comparisons to infer their global order.

Introduction to Red	uirements Engineering		48/62
Matthieu Vergne	vergne@is.naist.jp	Ν	IAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

**Requirements** Prioritisation

# Prioritisation Technique - Analytic Hierarchy Process (AHP)

Goal: Build a matrix of requirements comparisons to infer their global order.

Pros:

- Comparisons are weighted, thus more informative
- Estimations of the missing values can be computed to ask only the most relevant comparisons

Introduction to Requ	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

**Requirements** Prioritisation

# Prioritisation Technique - Analytic Hierarchy Process (AHP)

Goal: Build a matrix of requirements comparisons to infer their global order.

Pros:

- Comparisons are weighted, thus more informative
- Estimations of the missing values can be computed to ask only the most relevant comparisons

Cons:

Although improved, remain costly in terms of human effort.

Introduction to Req	uirements Engineering	48/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

イロト 不得 トイヨト イヨト ヨヨ ろくで

**Requirements** Prioritisation

# Prioritisation Technique - Case Based Ranking (CBRank)

Goal: Learn from previous comparisons to infer remaining comparisons.

Introduction to Rec	uirements Engineering	(0) PUBLICDOMAIN 49/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

# Prioritisation Technique - Case Based Ranking (CBRank)

Goal: Learn from previous comparisons to infer remaining comparisons.

Pros:

- Machine learning reduces human effort
- Tend to be more efficient than AHP (Avesani et al. [2005])

Introduction to Rec	uirements Engineering	49/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Prioritisation

# Prioritisation Technique - Case Based Ranking (CBRank)

Goal: Learn from previous comparisons to infer remaining comparisons.

Pros:

Machine learning reduces human effort

■ Tend to be more efficient than AHP (Avesani et al. [2005]) Cons:

Critics on learning with anecdotal evidences

Introduction to Requ	uirements Engineering	49/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

**Requirements** Prioritisation

# Prioritisation Technique - Interactive Genetic Algorithms

Goal: Iteratively improve a set of rankings, by recombinations and random mutations, to comply with pre-determined + elicited constraints.



Requirements Engineering

RE Activities

Requirements Prioritisation

#### Prioritisation Technique - Interactive Genetic Algorithms

Goal: Iteratively improve a set of rankings, by recombinations and random mutations, to comply with pre-determined + elicited constraints.



#### Pictures: reworked from Angelo Susi

Introduction to Req	uirements Engineering	50/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Requirements Prioritisation

# Prioritisation Technique - Interactive Genetic Algorithms

Goal: Iteratively improve a set of rankings, by recombinations and random mutations, to comply with pre-determined + elicited constraints.

Pictures: reworked from Angele Suci



Pros:

- Recombinations and mutations motivated by Darwin evolution theory
- The resulting rankings can provide a various set of good alternatives

1 1011	incs. Teworked from Angelo Sus			
		▲□▶ ▲圖▶ ▲厘▶	★ 문 ► (문) =	99
Introduction to Req	uirements Engineering			50/62
Matthieu Vergne	vergne@is.naist.jp			NAIST

Requirements Engineering

RE Activities

Requirements Prioritisation

# Prioritisation Technique - Interactive Genetic Algorithms

Goal: Iteratively improve a set of rankings, by recombinations and random mutations, to comply with pre-determined + elicited constraints.



◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Pros:

- Recombinations and mutations motivated by Darwin evolution theory
- The resulting rankings can provide a various set of good alternatives

Cons:

Intro Mat

- Solutions can converge to local optima only
- Set of solutions more costly than single one

Pictures: reworked from Angelo Susi

duction to Requ	irements Engineering	50/62
hieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

イロト イヨト イヨト イヨト ショー クタイ

**Requirements** Prioritisation



#### Requirements prioritisation usually involves human qualitative judgement

Introduction to Rec	quirements Engineering	51/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

イロト 不得 トイヨト イヨト ヨヨ ろくで

#### **Requirements** Prioritisation



- Requirements prioritisation usually involves human qualitative judgement
- Human efforts makes scalability problems common to all approaches

Introduction to Req	uirements Engineering		51/62
Matthieu Vergne	vergne@is.naist.jp	٩	VAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

#### **Requirements** Prioritisation



- Requirements prioritisation usually involves human qualitative judgement
- Human efforts makes scalability problems common to all approaches
- Other techniques exist
  - Cost-value approach, Binary Search Tree, Planning Game, etc.

Introduction to Req	uirements Engineering	DUBLICDOMAIN	51/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

#### Requirements Management

# Outline

#### 1 Requirements

2 Requirements Engineering

#### **3** RE Activities

- Requirements Elicitation
- Requirements Modelling
- Requirements Analysis
- Requirements Prioritisation
- Requirements Management

#### 4 Conclusion

			2.45
Introduction to Req	uirements Engineering		52/62
Matthieu Vergne	vergne@is.naist.jp	1	VAIST

Requirements Engineering

RE Activities

Conclusion 0000

◆□ > ◆□ > ◆三 > ◆三 > 三日 のへで

Requirements Management

#### **Requirements Management**

Goal: Ensure requirements access and update

Introduction to Rec	uirements Engineering	53/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion 0000

Requirements Management

#### **Requirements Management**

#### Goal: Ensure requirements access and update

Main tasks:

Introduction to Req	uirements Engineering	(0) ривисоо	MAN 53/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering 0000000 RE Activities

Conclusion 0000

イロト 不得 トイヨト イヨト ヨヨ ろくで

Requirements Management

#### **Requirements Management**

#### Goal: Ensure requirements access and update

Main tasks:

Store and retrieve requirements

Introduction to Requ	irements Engineering	(0) PUBLICDOMAIN	53/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Management

#### **Requirements Management**

#### Goal: Ensure requirements access and update

Main tasks:

- Store and retrieve requirements
- Relate requirements to other artefacts

Introduction to Requirements Engineering	53/62
Matthieu Vergne vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Management

#### Requirements Management

Goal: Ensure requirements access and update

Main tasks:

- Store and retrieve requirements
- Relate requirements to other artefacts
- Support updates of requirements and their relations

			1.10
Introduction to Req	uirements Engineering		53/62
Matthieu Vergne	vergne@is.naist.jp	١	VAIST

Requirements Engineering

イロト イヨト イヨト イヨト ショー クタイ

Requirements Management

# Management Technique - Dynamic Renumbering

Goal: Identify requirements from their location in requirement documents.

Introduction to Rec	uirements Engineering		62
Matthieu Vergne	vergne@is.naist.jp	NAI	SТ

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Management

# Management Technique - Dynamic Renumbering

Goal: Identify requirements from their location in requirement documents.

Pros:

- Align automatic and manual retrieval
- Provide implicit classification of requirements

Introduction to Req	uirements Engineering	54/62
Matthieu Vergne	vergne@is.naist.jp	NAIST
Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Management

## Management Technique - Dynamic Renumbering

Goal: Identify requirements from their location in requirement documents.

Pros:

- Align automatic and manual retrieval
- Provide implicit classification of requirements

Cons:

Hard to manage without automated tools

troduction to Requ	irements Engineering	54/62
atthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Management

### Management Technique - Database Record Identification

Goal: Exploit IDs in database.

Introduction to Red	quirements Engineering	DUBLICDOMAIN	55/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

**Requirements Management** 

## Management Technique - Database Record Identification

Goal: Exploit IDs in database.

Pros:

- Constant, no need to update
- Naturally generated by automated tools

Introduction to Requ	uirements Engineering	O PUBLICDOMAIN 55/0	52
Matthieu Vergne	vergne@is.naist.jp	NAIS	Т

Requirements Engineering

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

Requirements Management

## Management Technique - Database Record Identification

Goal: Exploit IDs in database.

Pros:

- Constant, no need to update
- Naturally generated by automated tools

Cons:

Hard to generate meaningful document from automatically generated indexes

Introduction to Requ	irements Engineering	(0) PUBLICDOMAIN	55/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

RE Activities

Conclusion

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

**Requirements Management** 

## Management Technique - Traceability

Goal: Link requirements to other requirements and derived resources.



Introduction to Requ	uirements Engineering	56/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

Requirements Management

## Management Technique - Traceability

Goal: Link requirements to other requirements and derived resources.



Picture: IBM Rational DOORS (https://www.doorsng.com/)

Introduction to Requirements Engineering

Matthieu Vergne vergne@is.naist.jp

56/62 NAIST

(0) PUBLICDOMAIN

Requirements Engineering

RE Activities

Conclusion

56/62

NAIST

**Requirements Management** 

# Management Technique - Traceability

Goal: Link requirements to other requirements and derived resources.



Pros:

Matthieu Vergne

Introduction to Requirements Engineering

vergne@is.naist.jp

- Help to update related requirements (horizontal traceability)
  - Requirements dependencies, derived requirements
- Help update other resources (vertical traceability)
  - $\blacksquare \textit{ Interviews} \rightarrow \textit{ requirements} \rightarrow \textit{ tests \& docs}$

Picture: IBM Rational DOORS (https://www.doorsng.com/)

Requirements Engineering

RE Activities

Conclusion

ELE DOG

Requirements Management

## Management Technique - Traceability

Goal: Link requirements to other requirements and derived resources.



Pros:

- Help to update related requirements (horizontal traceability)
  - Requirements dependencies, derived requirements
- Help update other resources (vertical traceability)
  - $\blacksquare \textit{ Interviews} \rightarrow \textit{requirements} \rightarrow \textit{tests} \ \& \ \textit{docs}$

Cons:

Int M:

- Complexity grows with number of requirements
- Hard to handle without automated tools (e.g. IBM Rational DOORS)

Picture: IBM Rational DOORS (https://www.doorsng.com/)

oduction to Req	uirements Engineering	(Ø) PUBLICDOMAIN	56/62
tthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering 0000000 RE Activities

Conclusion 0000

Requirements Management



Usually simple storage and retrieval.

Introduction to Rec	uirements Engineering		52
Matthieu Vergne	vergne@is.naist.jp	NAIS	Т

Requirements Engineering

Requirements Management



- Usually simple storage and retrieval.
- May start with spreadsheets when simple

Introduction to Req	uirements Engineering		/62
Matthieu Vergne	vergne@is.naist.jp	NA	IST

Requirements Engineering

イロト 不得 トイヨト イヨト ヨヨ ろくで

**Requirements Management** 



- Usually simple storage and retrieval.
- May start with spreadsheets when simple
- Quickly need advanced tools for deep management

Introduction to Requ	uirements Engineering	(Ø) PUBLICDOMAIN	57/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

4 E b

ELE DOG

58/62 NAIST

CDOMAIN

## Outline

#### 1 Requirements

- 2 Requirements Engineering
- 3 RE Activities
  - Requirements Elicitation
  - Requirements Modelling
  - Requirements Analysis
  - Requirements Prioritisation
  - Requirements Management

### 4 Conclusion

Introduction to Requ	uirements Engineering	<b>(0)</b> PI
Matthieu Vergne	vergne@is.naist.jp	

Requirements	

イロト 不得 トイヨト イヨト ヨヨ ろくで

## Summary

We have seen:

What is a requirement

		= 1 2 4 2
Introduction to Req	uirements Engineering	omain 59/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

## Summary

We have seen:

- What is a requirement
- What is Requirements Engineering (RE)

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

# Summary

We have seen:

- What is a requirement
- What is Requirements Engineering (RE)
- RE and software development are tightly related

Introduction to Requ	uirements Engineering		62
Matthieu Vergne	vergne@is.naist.jp	NAI	sт

# Summary

We have seen:

- What is a requirement
- What is Requirements Engineering (RE)
- RE and software development are tightly related
- Various RE techniques for different tasks

Introduction to Req	uirements Engineering	DUBLICDOMAIN	59/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

Requirements Engineering

RE Activities

Conclusion

### How to Make a Tree Swing?



		2.25
Introduction to Rec	uirements Engineering	60/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

Requirements Engineering

RE Activities

Conclusion

### How to Make a Tree Swing?



By having everyone agree (on the requirements)!

Introduction to Requirements Engineering 60/62 Matthieu Vergne vergne@is.naist.jp NAIST

◆□▶ ◆□▶ ◆ヨ▶ ◆ヨ▶ ヨヨ シのの

# Going Further

Books about RE:

- Requirements engineering: fundamentals, principles, and techniques (Pohl [2010])
  - Teaching material: https://re-buch.de/en/teaching-material/
- Requirements engineering: from system goals to UML models and software specifications (Lamsweerde [2009])

Introduction to Requ	irements Engineering	61/62
Matthieu Vergne	vergne@is.naist.jp	NAIST

### Thanks for your attention.

Questions?

Introduction to Req	uirements Engineering	(0) PUBLICDOMAIN	62/62
Matthieu Vergne	vergne@is.naist.jp		NAIST

- P. Avesani, C. Bazzanella, A. Perini, and A. Susi. Facing Scalability Issues in Requirements Prioritization with Machine Learning Techniques. In <u>Proceedings of</u> the 13th IEEE International Conference on Requirements Engineering, pages 297–306, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2425-7. doi: 10.1109/RE.2005.30. URL http://dl.acm.org/citation.cfm?id=1099549.1100639.
- P. Bourque, R. E. Fairley, and IEEE Computer Society. <u>Guide to the software</u> engineering body of knowledge. 2014. URL http://www.computer.org/portal/web/swebok/swebokv3. OCLC: 926093687.
- M. Glinz. On Non-Functional Requirements. pages 21–26. IEEE, Oct. 2007. ISBN 978-0-7695-2935-6. doi: 10.1109/RE.2007.45. URL http://ieeexplore.ieee.org/document/4384163/.
- IEEE Standards Board. IEEE standard glossary of software engineering terminology. Institute of Electrical and Electronics Engineers, New York, N.Y, 1990. ISBN 978-1-55937-067-7.
- A. v. Lamsweerde. <u>Requirements engineering : from system goals to UML models and software specifications</u>. Wiley ; John Wiley [distributor], Hoboken, N.J.; Chichester, 2009. ISBN 978-0-470-01270-3 0-470-01270-6. URL http://eu.wiley.com/WileyCDA/WileyTitle/productCd-EHEP000863.html.
- K. Pohl. <u>Requirements engineering:</u> fundamentals, principles, and techniques. Springer, Heidelberg ; New York, 2010. ISBN 978-3-642-12577-5. OCLC: ocn642291082.

Introduction to Req	uirements Engineering	
Matthieu Vergne	vergne@is.naist.jp	NAIST