


UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

# Requirements Analysis and Negotiation

*Software Engineering and Databases Group  
Department of Computer Languages and Systems  
University of Seville  
November 2015*

La traducción de este material docente ha sido financiada mediante la convocatoria 1.10B - Ayudas de innovación y mejora docente, convocatoria 2013-2014, modalidad B del II Plan Propio de Docencia de la Universidad de Sevilla. No ha habido financiación alguna para este proyecto de otros soportes.



UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

## Requirements Analysis and Negotiation

- Learning objectives
  - Know the objectives, products and **requirements analysis** processes.
  - Know the basics of **modeling** software systems.
  - Know the basics of **conflict** management and requirements **negotiation**.
  - Know the common elements of **UML**.

1. Requirements analysis goals


2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management


6. Unified Modeling Language



November 2015

Requirements Engineering

1



UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

# Requirements Analysis and Negotiation

1. Requirements analysis goals
2. Situation in the RE process
3. Analysis tasks
4. Software system models
5. Problem management
6. Unified Modeling Language

## Requirements analysis goals

- Detecting **problems** in the requirements at the **beginning** of the project, when they are much more inexpensive to solve.
- The most usual way of analysis is the development of a **model** of a software system satisfying product requirements, i.e. of the system to be developed.

November 2015

Requirements Engineering

2

# Requirements Analysis and Negotiation

• Situation in the RE process

The diagram illustrates the Requirements Engineering (RE) process as a continuous cycle. It begins with **Requirements Elicitation**, which leads to **Elicited Information** and **Requirements documentation**. From **Requirements documentation**, the process moves to **Requirements [Draft]**, which then leads to **Requirements Analysis**. **Requirements Analysis** is highlighted with a thick border and leads to **Requirements [Analyzed]**. From **Requirements [Analyzed]**, the process moves to **Requirements Verification**, which leads to **Requirements [Verified]**. From **Requirements [Verified]**, the process moves to **Requirements Validation**, which leads to **Requirements [Validated]**. From **Requirements [Validated]**, the process moves to **Requirements Management**, which leads to **Requirements [Versioned]**. From **Requirements [Versioned]**, the process moves back to **Requirements Elicitation**, completing the cycle. Additionally, there are feedback loops: **Conflicts [Solved]** and **Conflicts [Pending]** lead back to **Requirements Elicitation** and **Requirements [Draft]** respectively. **Defects** lead back to **Requirements Analysis**. The process is divided into two main sections: a top section for elicitation, documentation, and management, and a bottom section for analysis, verification, and validation.

1. Requirements analysis goals  
2. Situation in the RE process  
3. Analysis tasks  
4. Software system models  
5. Problem management  
6. Unified Modeling Language

**Requirements Elicitation**

**Requirements Management**

**Requirements Analysis**

**Requirements Verification**

**Requirements Validation**

**Requirements [Draft]**

**Requirements [Analyzed]**

**Requirements [Verified]**

**Requirements [Validated]**

**Requirements [Versioned]**

**Conflicts [Solved]**

**Conflicts [Pending]**

**Elicited Information**

**Requirements documentation**

**Defects**

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Requirements analysis tasks
  - Develop a **model** from the documented goals and requirements, taking also into account business models, glossaries of terms, etc..
  - Register **problems** identified during model development for later resolution.
  - It is increasingly common during analysis to propose an initial **system architecture** considering **nonfunctional requirements**.

November 2015

Requirements Engineering

4

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language


Requirements Analysis and Negotiation

- Software system models
  - The system offers **services** to the actors.
  - The system **state** are the objects and their links.

November 2015

Requirements Engineering

5



UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals


2. Situation in the RE process

3. Analysis tasks

4. Software system models

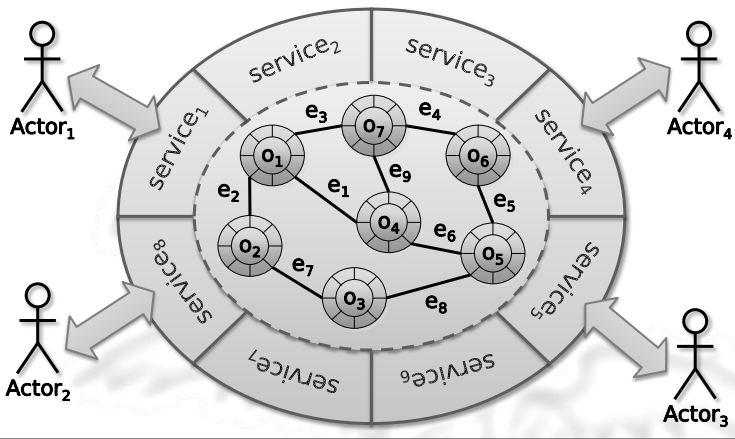
5. Problem management

6. Unified Modeling Language



Requirements Analysis and Negotiation

- Software system models
  - Updating services make the system state change.




The diagram illustrates a software system model. It features a central circular area containing several objects (O1, O2, O3, O4, O5, O6, O7) and edges (e1, e2, e3, e4, e5, e6, e7, e8, e9) representing relationships. This central area is surrounded by eight segments labeled service1 through service8. Four actors (Actor1, Actor2, Actor3, Actor4) are positioned around the perimeter, with arrows indicating interactions with the services. The entire model is enclosed in a dashed circle.

November 2015

Requirements Engineering

6



UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals


2. Situation in the RE process

3. Analysis tasks

4. Software system models

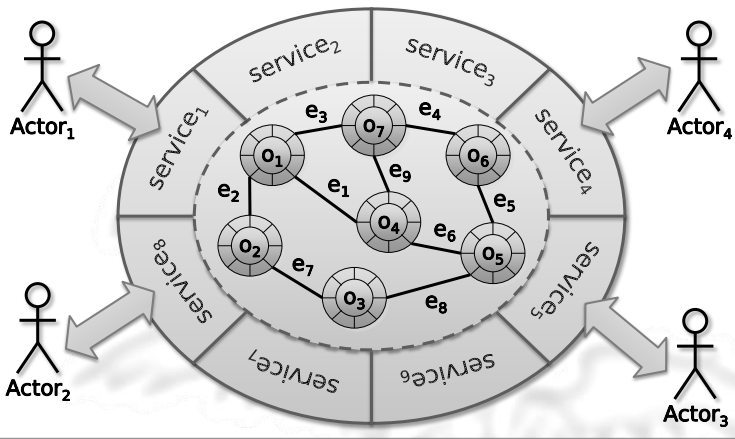
5. Problem management

6. Unified Modeling Language



Requirements Analysis and Negotiation

- Software system models
  - Query services do not change the system state.



The diagram illustrates a software system model. It features a central circular area containing several objects (O1, O2, O3, O4, O5, O6, O7) and edges (e1, e2, e3, e4, e5, e6, e7, e8, e9) representing relationships. This central area is surrounded by eight segments labeled service1 through service8. Four actors (Actor1, Actor2, Actor3, Actor4) are positioned around the perimeter, with arrows indicating interactions with the services. The entire model is enclosed in a dashed circle.

November 2015

Requirements Engineering

7

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Software system models
  - Regarding information system functions...
    - **Memory function:** system environment state  $\approx$  system state.
    - System state is kept synchronized with the environment by the updating services.

November 2015

Requirements Engineering

8

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Software system models
  - Regarding information system functions...
    - **Informative function:** provided by the query services.
    - To be useful, system state must be synchronized with the environment state.\*

\* It does not have to be synchronized immediately, there may be a lag and still be useful for actors.

November 2015

Requirements Engineering

9

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Software system models
  - Regarding **information system functions...**
    - **Active function:** provided through interaction with users and with other systems (actors in general).
    - Interaction with other systems is performed by invoking their services.

November 2015

Requirements Engineering

10

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- A software system model includes...
  - A **static model** (conceptual model)
    - Describes the structure and constraints of the information representing the system state.
  - A **behavior model**
    - Describes how the system interacts with actors and how its state evolves as a result of the interactions.

Model =

Static model =

Structure

+

Constraints

+

Behavior model =

External interactions

+

Internal evolution

November 2015

Requirements Engineering

11

UNIVERSIDAD DE SEVILLA



Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Requirements problems management
  - **Small problems** such as lack of information or clarification needs can be solved consulting with customers and users in a relatively **informal** way (e.g. a phone call, an email, or a video conference).



November 2015

Requirements Engineering

12

UNIVERSIDAD DE SEVILLA



Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Requirements problems management
  - If **conflicts** requiring nontrivial solutions are detected, **Requirements Negotiation** is needed.



November 2015

Requirements Engineering

13

RE

7

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Requirements negotiation process
  - The common **process** is usually the following:
    - Record the conflict (usually with a tool).
    - Identify affected requirements.
    - Analyze the conflict impact (using requirements traceability).
    - Identify relevant sources (stakeholders).
    - Prepare and conduct a negotiation meeting.
    - Incorporate the conflict solution.

© D. García de la Puente, 2015

November 2015

Requirements Engineering

14

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Unified Modeling Language (UML)
  - Object Management Group (OMG) standard for modeling software systems.

UML version history diagram showing the lineage from Booch's methods to UML 2.4.1. The diagram shows a flow from Booch'91 and Other methods to Unified Method 0.8, then to UML 0.9, and finally to UML 1.0 through UML 2.4.1. Key milestones include the introduction of OOSE and the transition to the current UML 2.x series.

Version	Date
Booch'91	
Booch'93	
Unified Method 0.8	October 1995
UML 0.9	June 1996
UML 1.0	Jan 1997
UML 1.1	Sep 1997
UML 1.2	June 1998
UML 1.3	Mar 2000
UML 1.4	Sep 2001
UML 1.5	Mar 2003
UML 2.0	2005
UML 2.1.1	Aug 2007
UML 2.1.2	Nov 2007
UML 2.2	Feb 2009
UML 2.3	May 2010
UML 2.4	Mar 2010
UML 2.4.1	Aug 2011

November 2015

Requirements Engineering

15

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Unified Modeling Language (UML)
  - Widely used in software industry and supported by many software tools.
  - It defines 14 types of diagrams for modeling software systems (version 2.4.1, August 2011)
  - We will use ...
    - Use case diagrams
    - Class diagrams
    - Object diagrams
    - State diagrams
    - Sequence diagrams

UNIFIED MODELING LANGUAGE

November 2015

Requirements Engineering

16

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

- Organization of UML models
  - In UML, models are organized in **packages**, which may include other packages.
  - Packages should group semantically close items that tend to change together.
  - There are some special packages:
    - «system»: it corresponds to the system being modeled.
    - «subsystem»: it represents a part of the system being modeled.

November 2015

Requirements Engineering

17

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

• Common UML elements

– Notes

• They are represented as boxes with a folded corner and can be associated with any element.

• They can contain either **comments** (any combination of text and graphics) or **constraints**.

– Comments

• They are displayed inside **notes** associated with the element to which the comment refers to.

November 2015

Requirements Engineering

18

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

• Common elements of UML

– Constraints

• They represent **conditions** to be held by the elements to which it is associated to. They are expressed by text in braces and can appear within elements or inside **notes**.


November 2015

Requirements Engineering

19

RE

10



UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks

4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

• Bibliography

– C. Larman, **UML y Patrones**. Ed. Prentice-Hall, 1999.

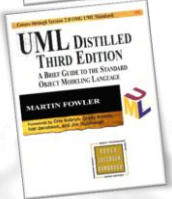

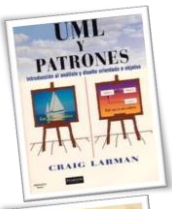
• Chapters 9 to 12

– C. Larman, **UML y Patrones (2ª edición)**. Ed. Prentice-Hall, 2003.

• Chapters 10 to 12

– M. Fowler, **UML Distilled (3rd edition)**. Ed Addison-Wesley, 2004.


• Chapter 3



November 2015

Requirements Engineering

22



UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Requirements analysis goals

2. Situation in the RE process

3. Analysis tasks


4. Software system models

5. Problem management

6. Unified Modeling Language

Requirements Analysis and Negotiation

• Comments, suggestions, ...



November 2015

Requirements Engineering

23

RE

11