

# Rational Unified Process

Requirements Engineering

Johannes Passing, 10.07.2006



# Agenda

---

- **Motivation**
- **Rational Unified Process**
- **Requirements Capture**
  - **Overview**
  - **Business Modelling Workflow**
  - **Requirements Workflow**
- **Tools Overview**

# Motivation

---

## ➤ Problems in Requirements Capture

- Users do not know what they need
- ‚Classic Requirements specification‘
  - Describes WHAT and HOW
  - Lacks WHY
  - Conceals the original problems and intents

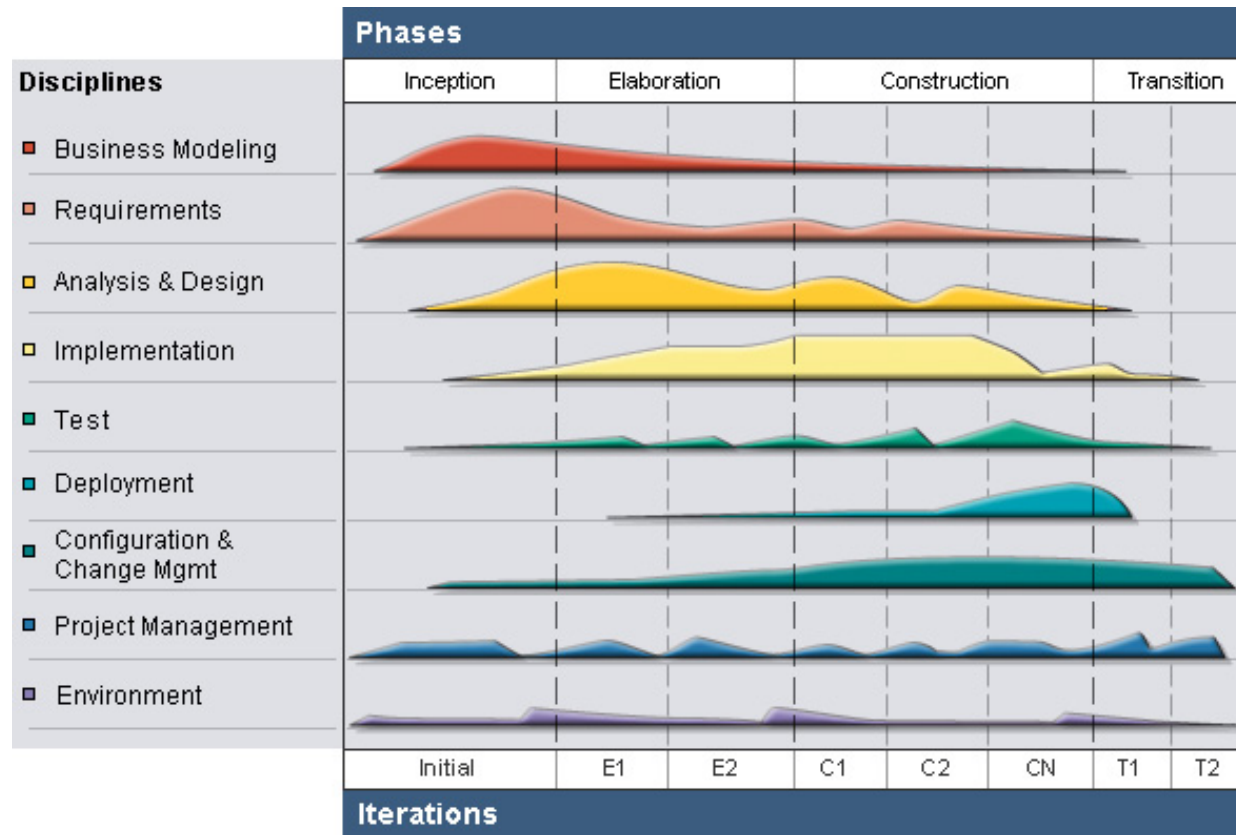
## ➤ Shortcomings of Waterfall Approach

- Wrong Assumptions:
  - Requirements will be frozen
  - Stable requirements as foundation

→ Users' needs and technology change

→ Works only if same people, same tools and same problem domain involved

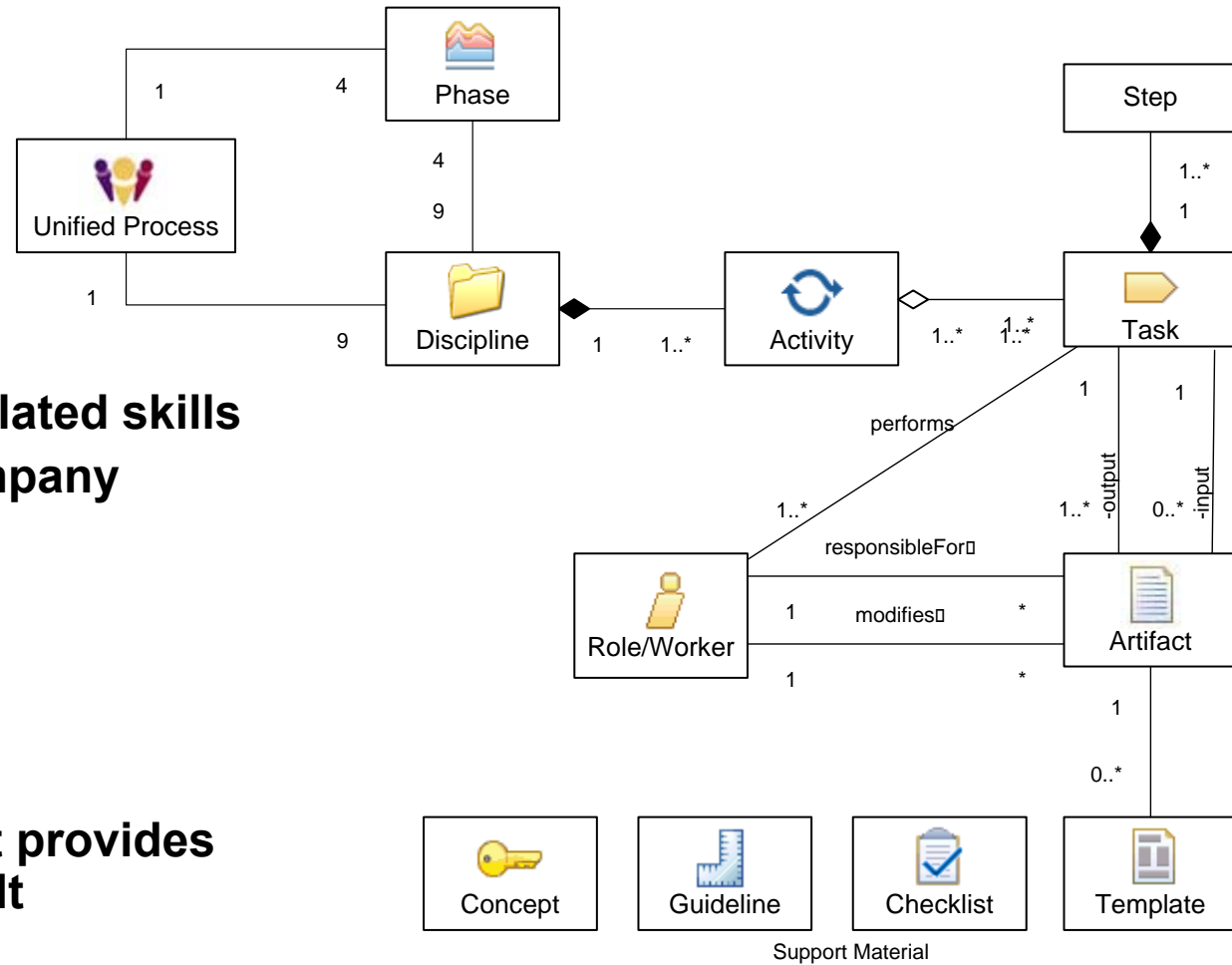
# RUP – Process



## ➔ Rational Unified Process (RUP)

- Iterative process
  - Emphasis on visual modelling
  - Covers complete development lifecycle
    - Supports Tailoring
- Helps avoiding 'Methodology Madness'

# RUP – Structure



## ➤ Role/Worker

- Defines set of related skills
- Internal to a company

## ➤ Activity

- Group of tasks

## ➤ Task

- Unit of work that provides meaningful result

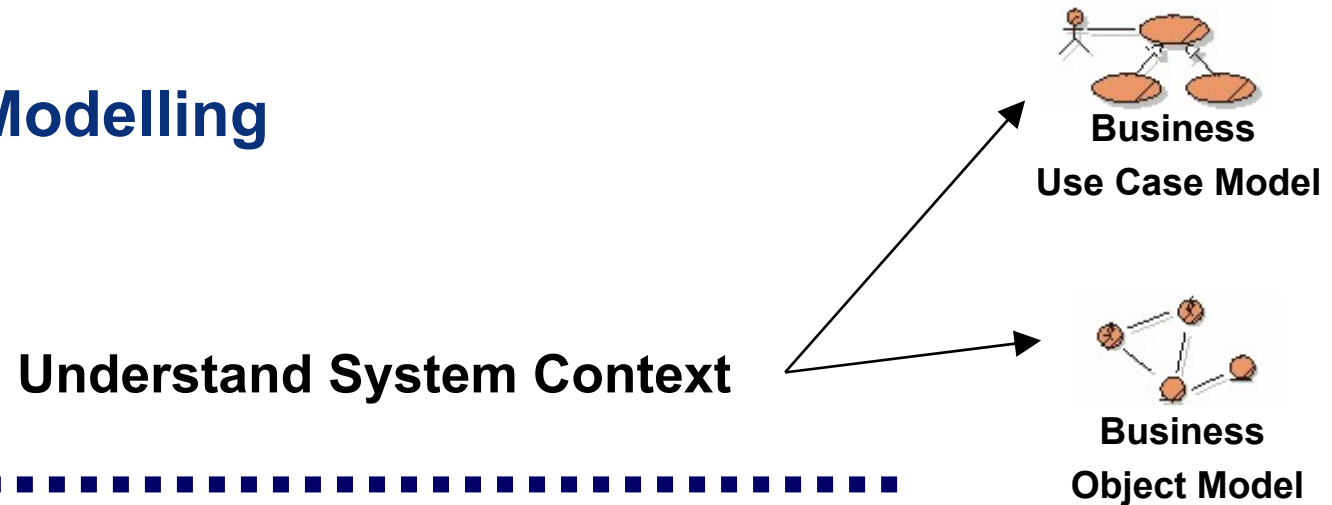
## ➤ Artifact

- Result of a task
- *Not necessarily a document*

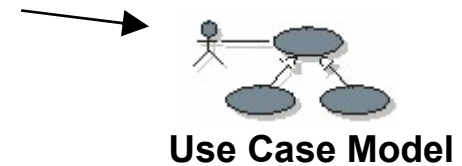
# Requirements Capture Overview

---

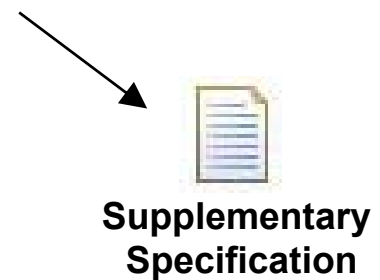
## Discipline: Business Modelling



**Capture functional requirements**



**Capture non-functional requirements**



## Discipline: Requirements

# Business Modelling – Discipline

---

## ➤ Business Modelling

- Performed mainly during Inception Phase
- For larger projects only
  - Not contained in ‚Small Configuration Template‘

## ➤ Goals

- Reflect on how the business is run
- Understand structure and dynamics of organisation
- Understand problems and identify improvement potentials
- Establish common understanding of organisation among customers, end users and developers
- Develop a vision of new *target* organisation

## ➤ Two Approaches

- Domain Modelling
- Business Modelling

# Business Modelling – Domain Modelling

## ➤ Approach: Domain Modelling

### ● Describe important concepts as Domain Classes

- E.g. Order, Invoice, Account
- Few classes (10-50)
- UML Class Diagram

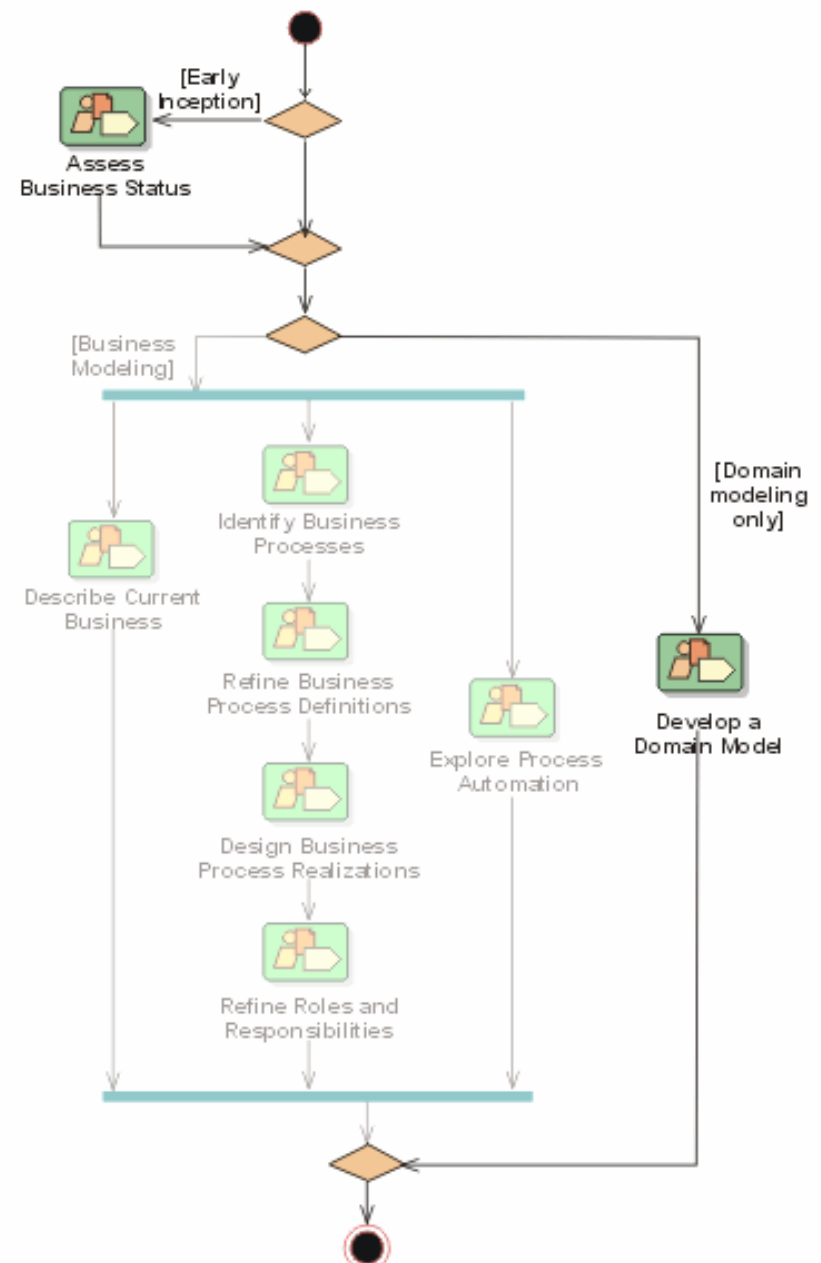
→ Classes not suited for implementation

→ No internal parts,  
no solution to problem

### ● Performed by Domain Experts

→ Establishes common vocabulary

→ Helps to derive glossary



# Business Modelling – Business Modelling

## ➤ Approach: Business Modelling

- Superset of Domain Modelling

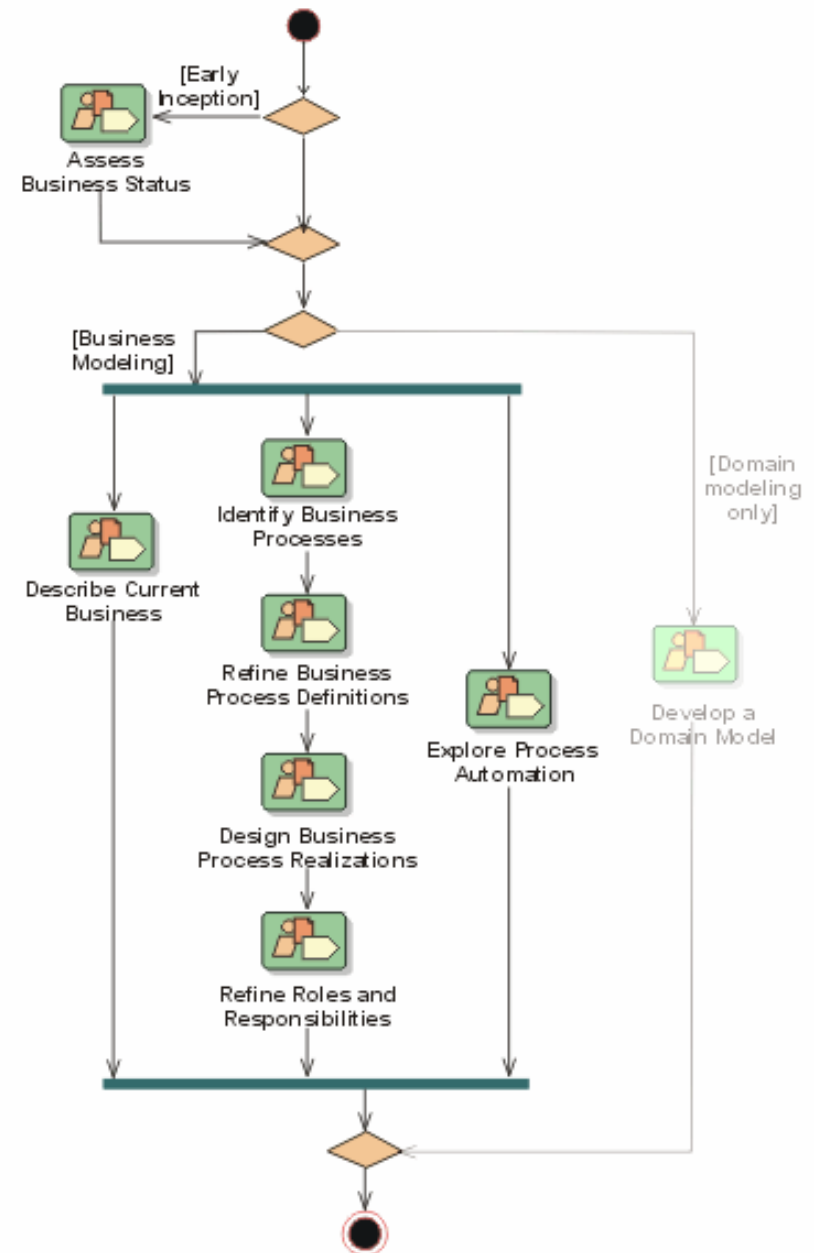
- Describe business from usage perspective

- Assess

- Business Workers
- Business Entities
- Business Use Cases

➔ Understand business and its processes

➔ Helps identifying Use Cases



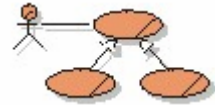
# Business Modelling – Comparing Approaches

---

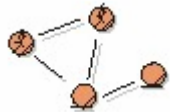
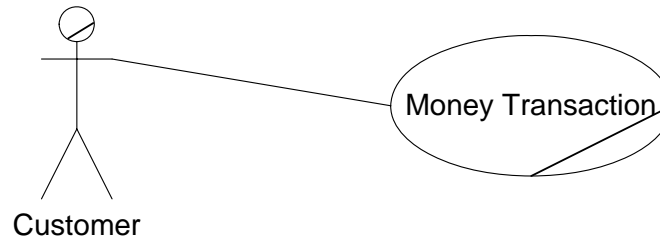
- **Comparison between Business Modelling and Domain Modelling**
  - **Business Modelling more ‘complete’**
  - **Business Entities  $\approx$  Domain Classes**

	<b>Domain Model</b>	<b>Business Model</b>
<b>Classes/ Entities</b>	<ul style="list-style-type: none"> <li>➤ <b>Created by few experts</b></li> <li>→ <b>dependent on experience</b></li> </ul>	<ul style="list-style-type: none"> <li>➤ <b>Derived from Use Cases</b></li> <li>→ <b>Each entity has a rationale</b></li> <li>→ <b>Backtracing to customer need</b></li> </ul>
	<ul style="list-style-type: none"> <li>➤ <b>Few operations</b></li> </ul>	<ul style="list-style-type: none"> <li>➤ <b>Operations derived from how workers use entities</b></li> </ul>

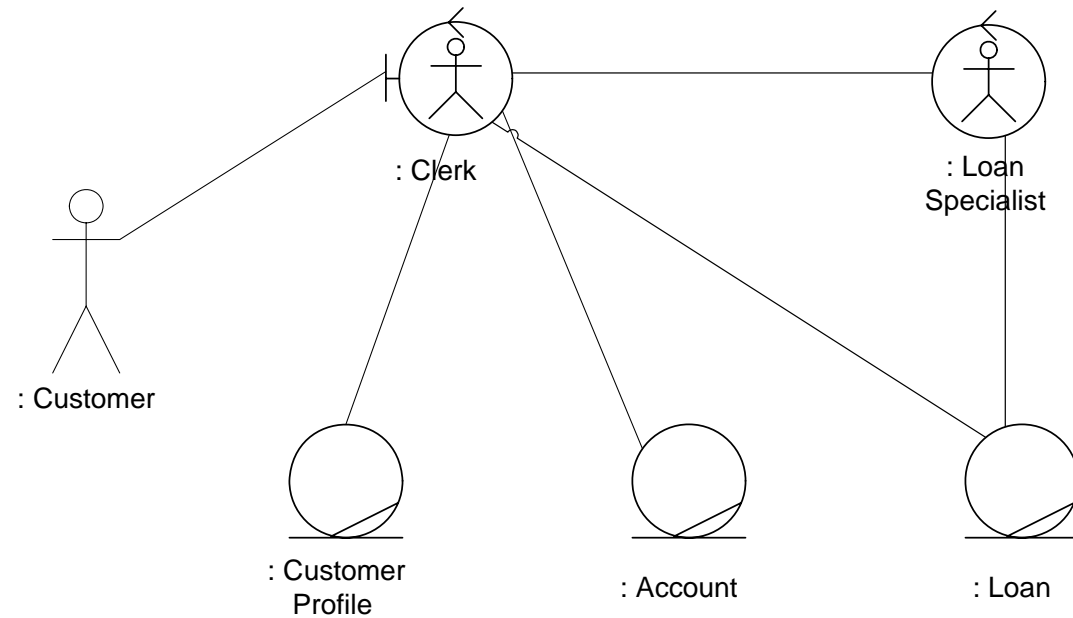
# Business Modelling – Example (1/2)



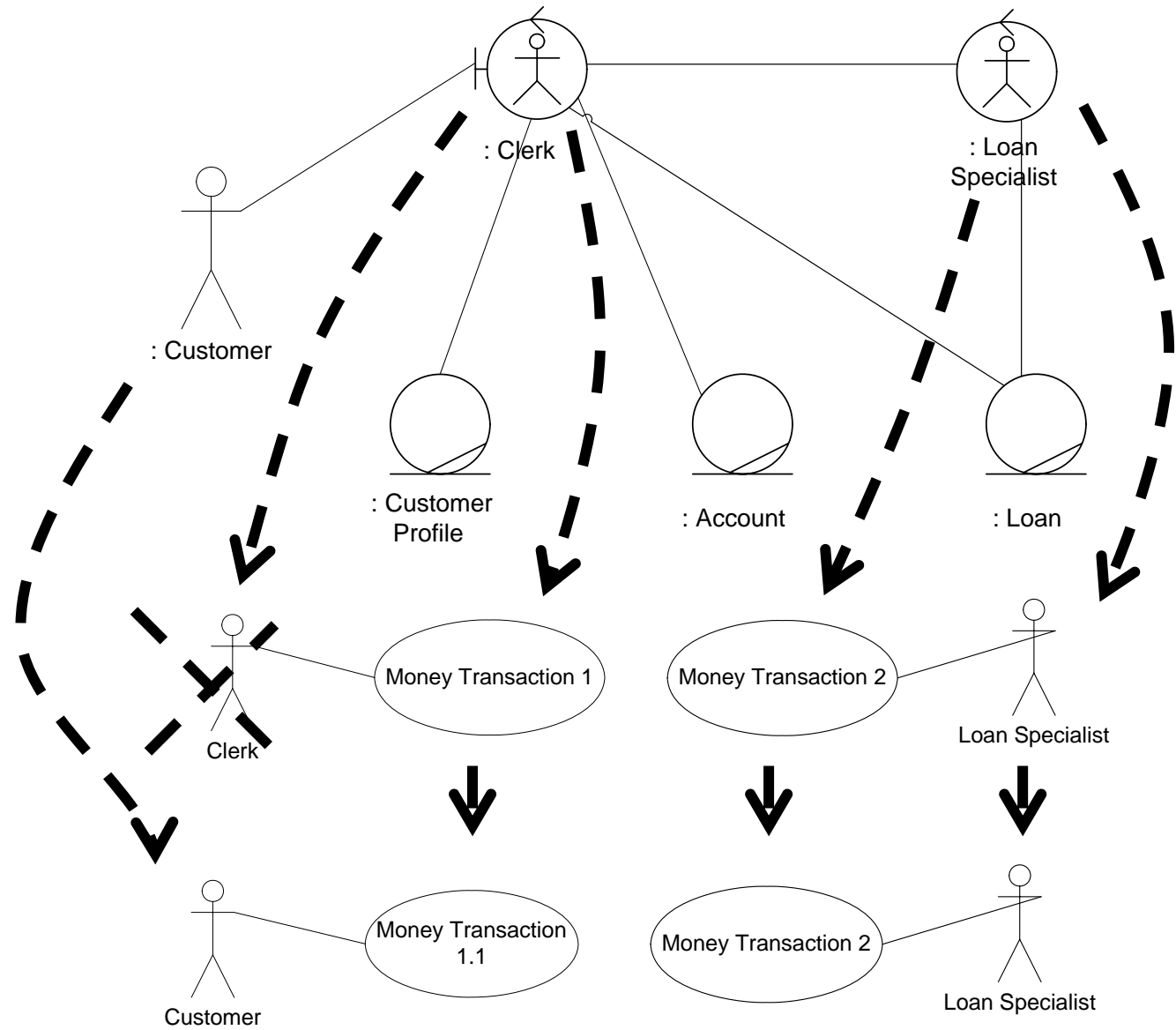
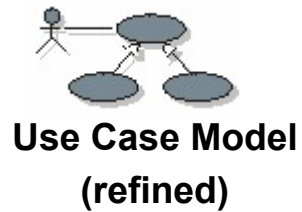
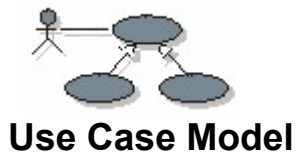
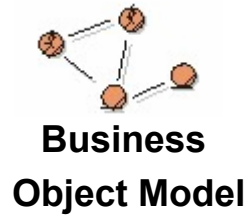
**Business  
Use Case Model**



**Business  
Object Model**



# Business Modelling – Example (2/2)



# Requirements – Discipline

---

## ➤ Requirements

### ● Spreads over 3 phases

- Inception: Identify Use Cases
- Elaboration: Capture remaining requirements, estimate effort
- Construction: Capture remaining and changing requirements

### ● Use Case-centered

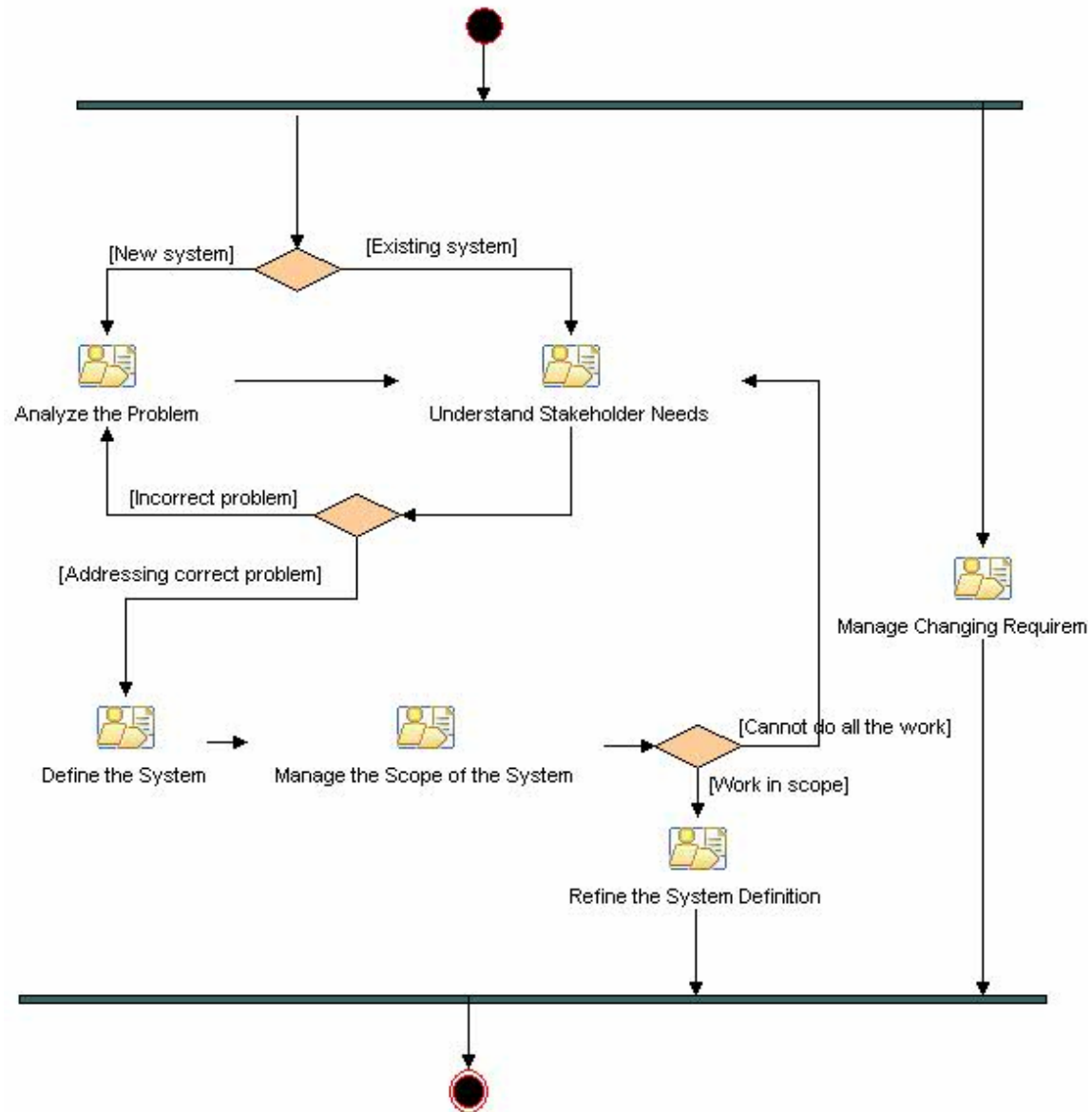
### ● Use language of customer

- Late “Translation to developer language“

## ➤ Goals

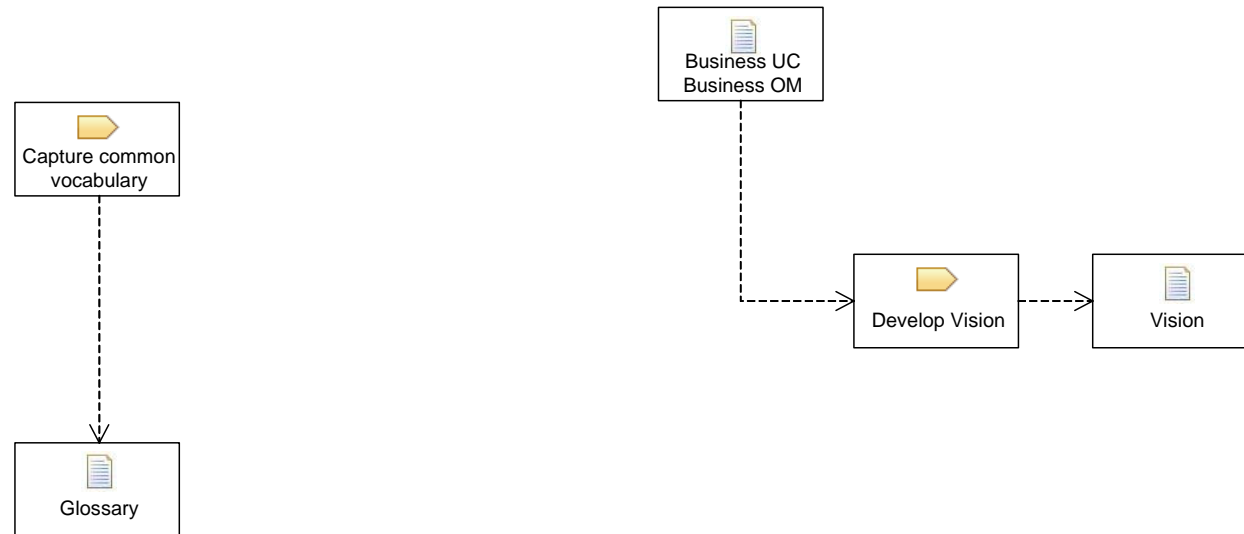
- Provide developers with better understanding of requirements
- Provide basis for estimating cost and effort
- Express WHY requirements have been formulated
  - Better interpretation
  - Easier to estimate risk of not satisfying the requirement

# Requirements – Workflow



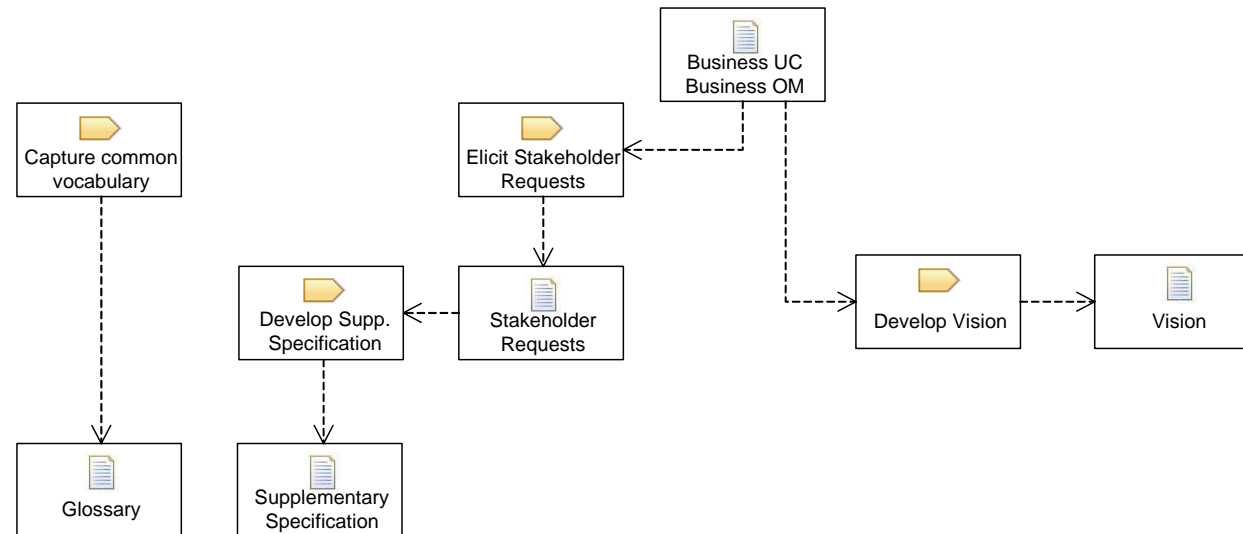
# Requirements – Activity: Analyze the Problem

---



- **Develop Vision**
  - **Outline expected functionality**
- **Identify stakeholders**
- **Gain agreement on problem to be solved**
- **Define Glossary**

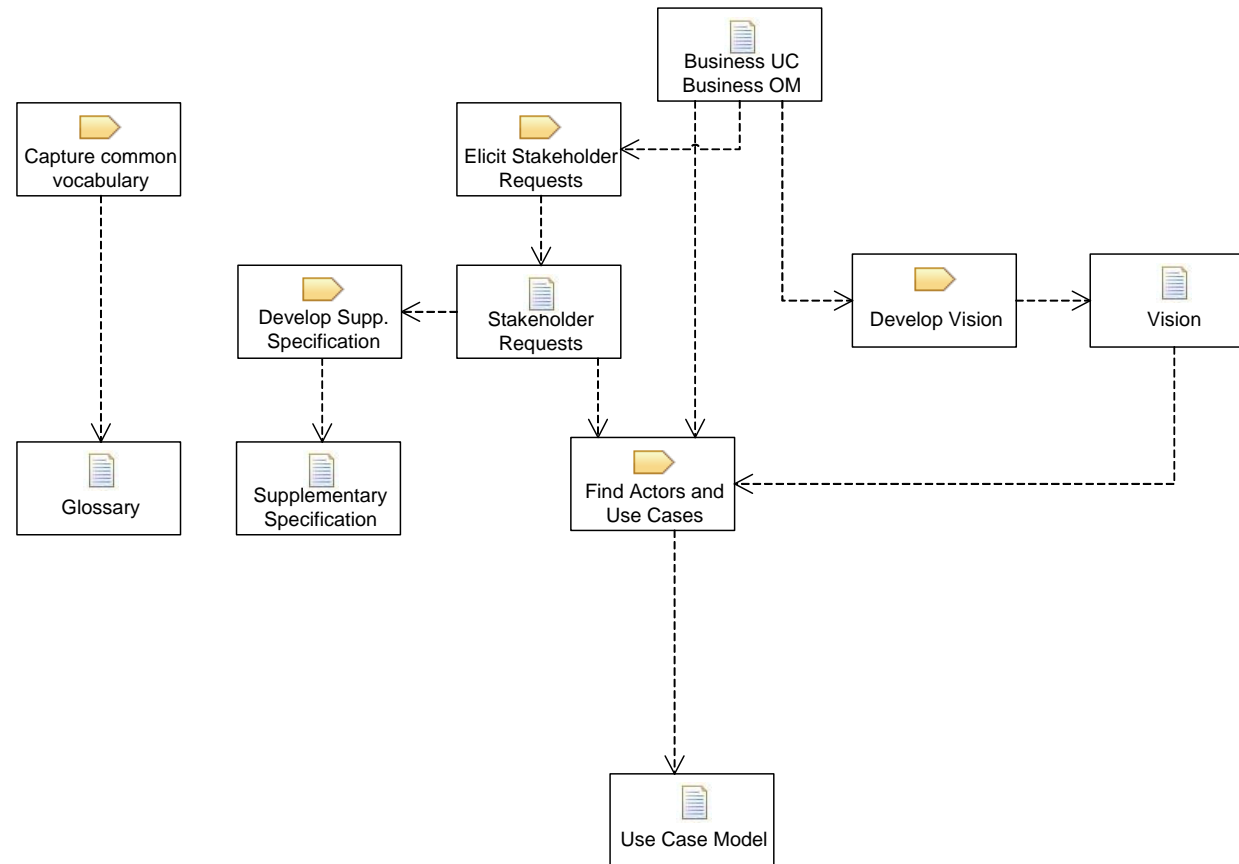
# Requirements – Activity: Understand stakeholder needs



## ➤ Gather stakeholder requests

- Proposals (early), CRs (later) etc.
- Obtain understanding of needs
- Use Interviews, questionnaires etc
- Assemble Requirements List
  - List all high-level requirements
  - Try to capture intent of requests (the 'WHY')

# Requirements – Activity: Define the System (1/2)

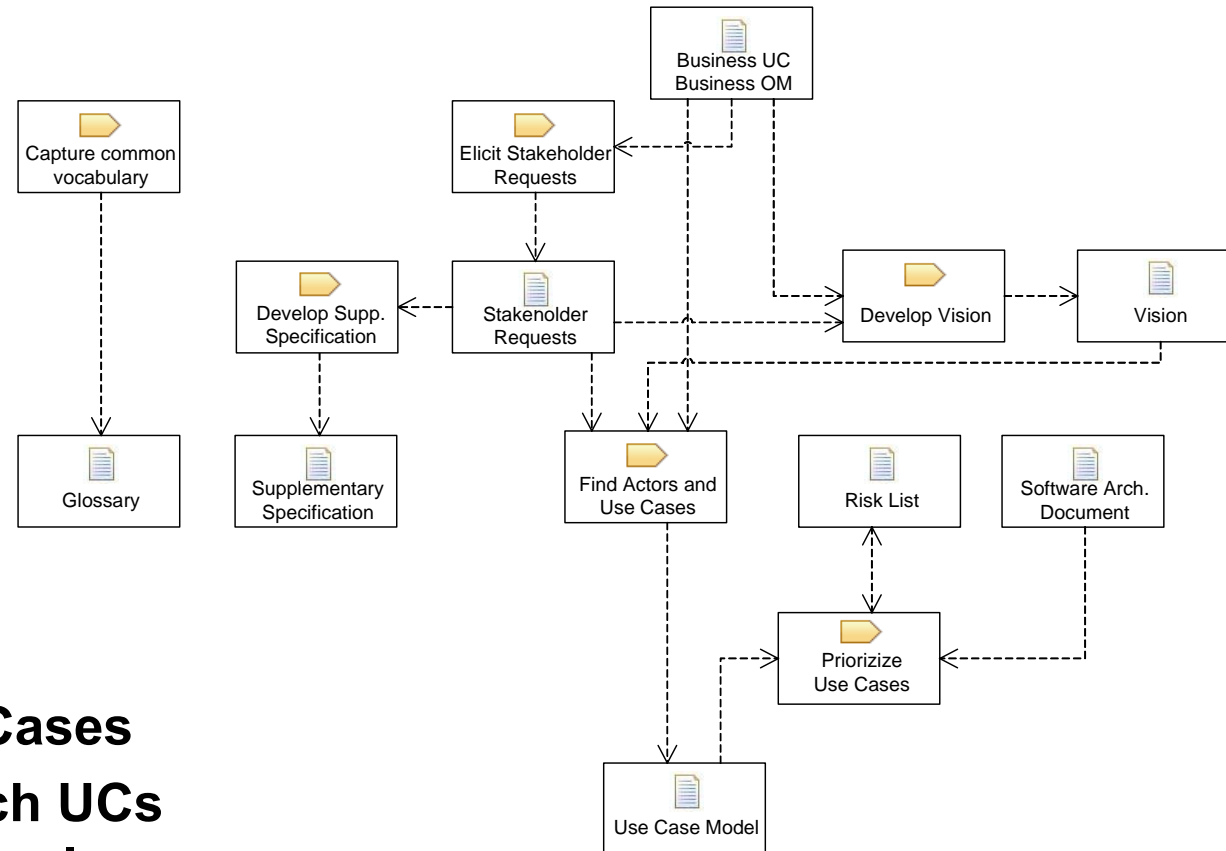


- Define set of system features to be delivered
- Determine feature prioritization criteria
- Identify actors and use cases

# Requirements – Activity: Define the System (2/2)

	With Business Modelling	Without Business Modelling
Find actors	<ul style="list-style-type: none"> <li>➔ 1 actor per worker</li> </ul>	<ul style="list-style-type: none"> <li>➔ Analyst/Customer: identify users</li> <li>➔ Categorize users into actors</li> </ul>
Find Use Cases	<ul style="list-style-type: none"> <li>➔ Derive from Business Use Case Model</li> </ul>	<ul style="list-style-type: none"> <li>➔ Analyst/Customer/Users: Identify Use Cases</li> </ul>
Describe Use Cases	<ul style="list-style-type: none"> <li>➔ Provide brief summary</li> <li>➔ Initial step-by-step description</li> </ul>	
Describe Use Case Model	<ul style="list-style-type: none"> <li>➔ Explain model as a whole</li> <li>➔ Determine whether all requirements have been incorporated</li> <li>➔ May be reviewed</li> </ul>	

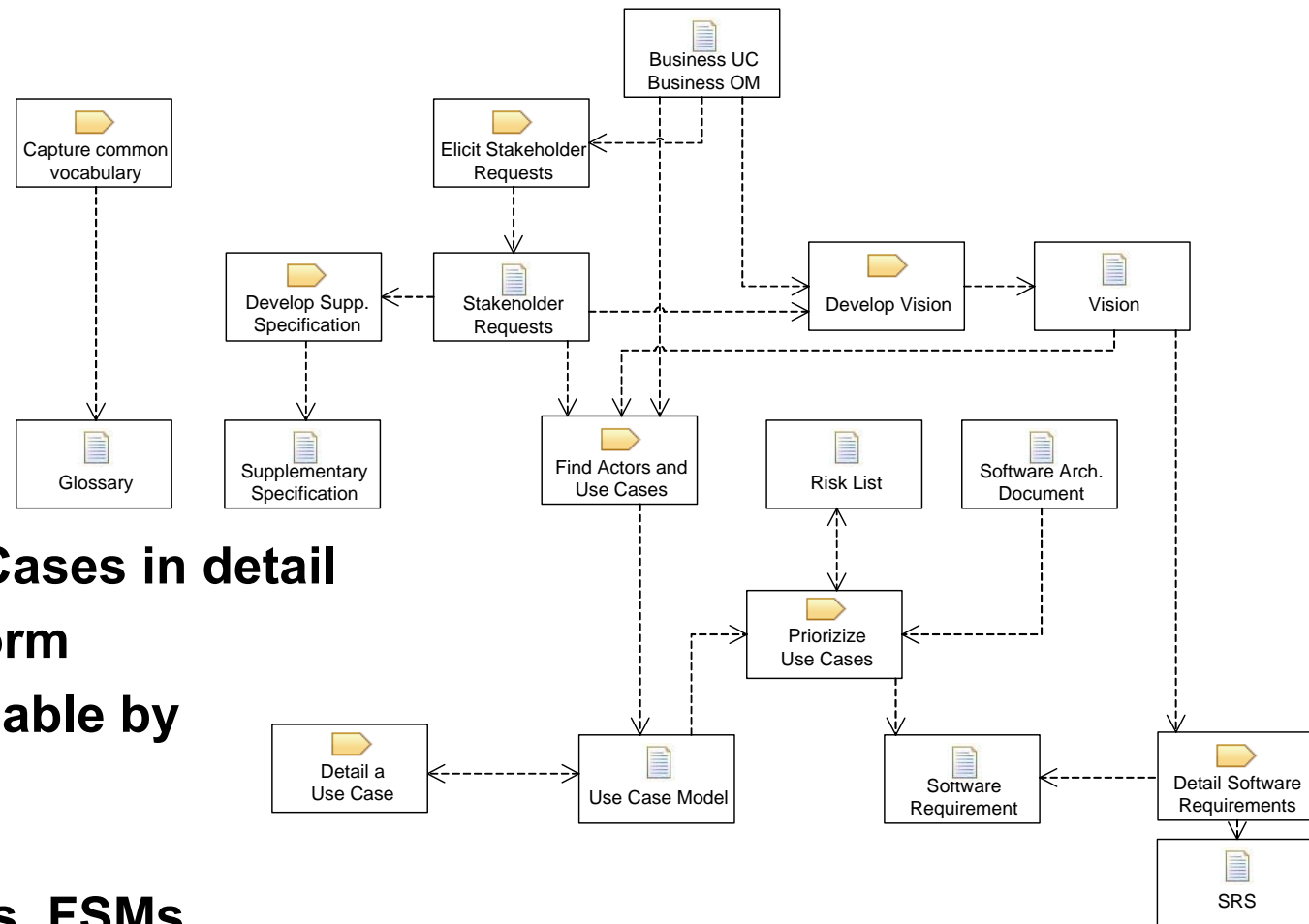
# Requirements – Activity: Manage the scope of the system



## ➤ Prioritise Use Cases

- Decide which UCs to develop early
- Incorporate Architecture and Risks

# Requirements – Activity: Refine the system definition



## Describe Use Cases in detail

- In textual form
- Understandable by customers

## Specify Flow

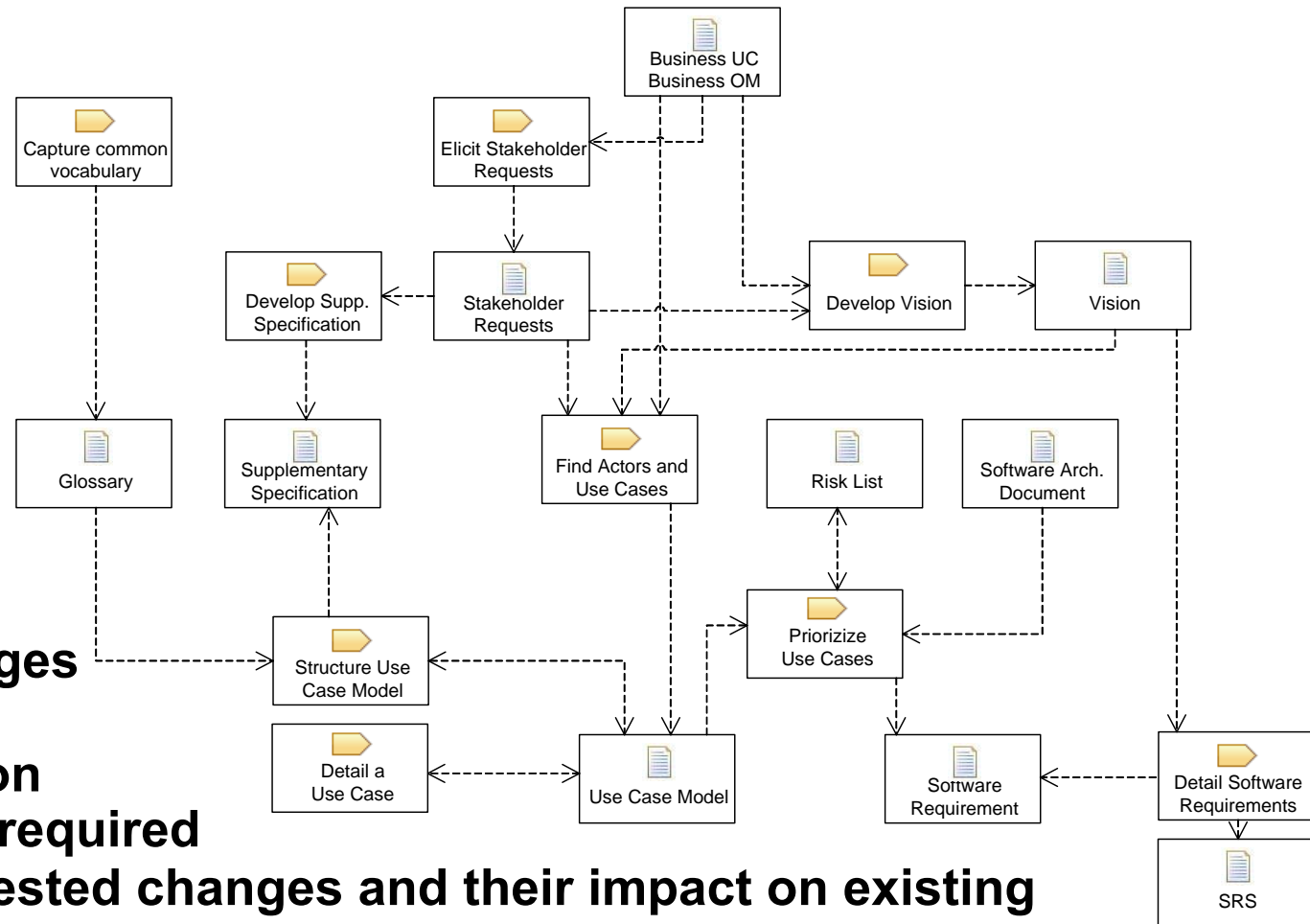
- State Charts, FSMs

## Gather non-functional requirements in Supplementary Spec.

## Derive Software Requirements Specification (SRS)

## Prototype UI

# Requirements – Activity: Manage changing requirements



- **Control and evaluate changes**
- **Set realistic expectations on cost and time required**
- **Evaluate requested changes and their impact on existing requirements**
- **Structure Use Case Model**
  - **Create Generalisations and Extension Points**

# Tools

---

## ➤ Rational Method Composer

- Successor of Rational Process Workbench
- Incorporates RUP
- Helps maintaining the Unified Process
- Commercial Software

## ➤ Eclipse Process Framework Project (“Project Beacon“)

- Based on IBM’s contribution of 15% of RUP
- Streamlined Version of RUP
- Targeted at small projects
  - 3 to 6 persons, 3 to 6 months

# Summary

---

- **Structured process**
- **RE fully incorporated in complete development lifecycle**
  - **Integration with other Disciplines**
- **Interactive**
  - **Facilitates daily usage**
  - **Supports daily work through Guidelines, Templates etc.**
- **Customizable**
  - **Variation points, Tailoring**
- **Comprehensive, but complex**

# References

---

- ➔ [1] P. Kruchten, “The Rational Unified Process”, Addison Wesley, 2001
- ➔ [2] I. Jacobsen, G. Booch, J. Rumbaugh, “The Unified Development Process”, Addison Wesley, 2001
- ➔ [3] IBM Corporation, Rational Unified Process (Evaluation Version), Version 7.0
- ➔ [4] Rational Software Corporation, Rational Unified Process, Version 2002.05.00  
<http://www.ts.mah.se/RUP/RationalUnifiedProcess/>
- ➔ [5] IBM Corporation, IBM Rational Method Composer,  
[ftp://ftp.software.ibm.com/software/rational/web/datasheets/rmc\\_ds.pdf](ftp://ftp.software.ibm.com/software/rational/web/datasheets/rmc_ds.pdf)  
(22.06.2006)
- ➔ [6] M. Fourman, „Rational Unified Process Process Description and Workflows“,  
<http://www.dcs.ed.ac.uk/teaching/cs2/online/Lectures/CS2Ah/SoftEng/se02-slides.PDF> (22.06.2006)
- ➔ [7] Object Management Group, Uml Extensions,  
[http://www.jeckle.de/files/UML1\\_2/4\\_exten.pdf](http://www.jeckle.de/files/UML1_2/4_exten.pdf)
- ➔ Business Modelling Example: Derived from [2]
- ➔ Graphics: Derived from [4], [5]