

2008 Fall

# Software Modeling & Analysis

Introduction to OSP

OSP Stage 1000

Lecturer: JUNBEOM YOO  
jbyoo@konkuk.ac.kr

# What is OSP?

- OSP (Object Space Process)
  - A software process based on RUP
  - Tailored to SE classes in universities
- Characteristics of OSP
  1. 3 Stages
  2. Iterative : Multiple development cycles
  3. Incremental : System grows incrementally as each cycle is completed
  4. Architecture : Stage > Cycle > Phase > Activity



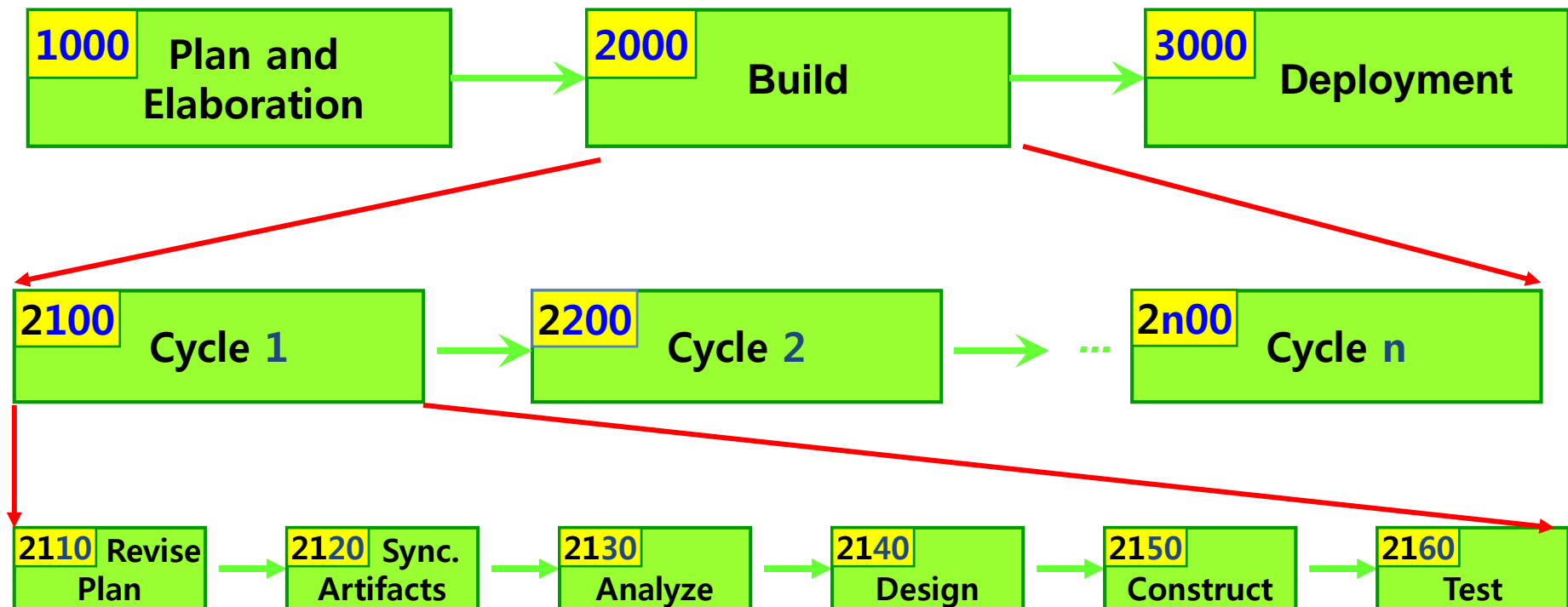
# 1. 3 Stages



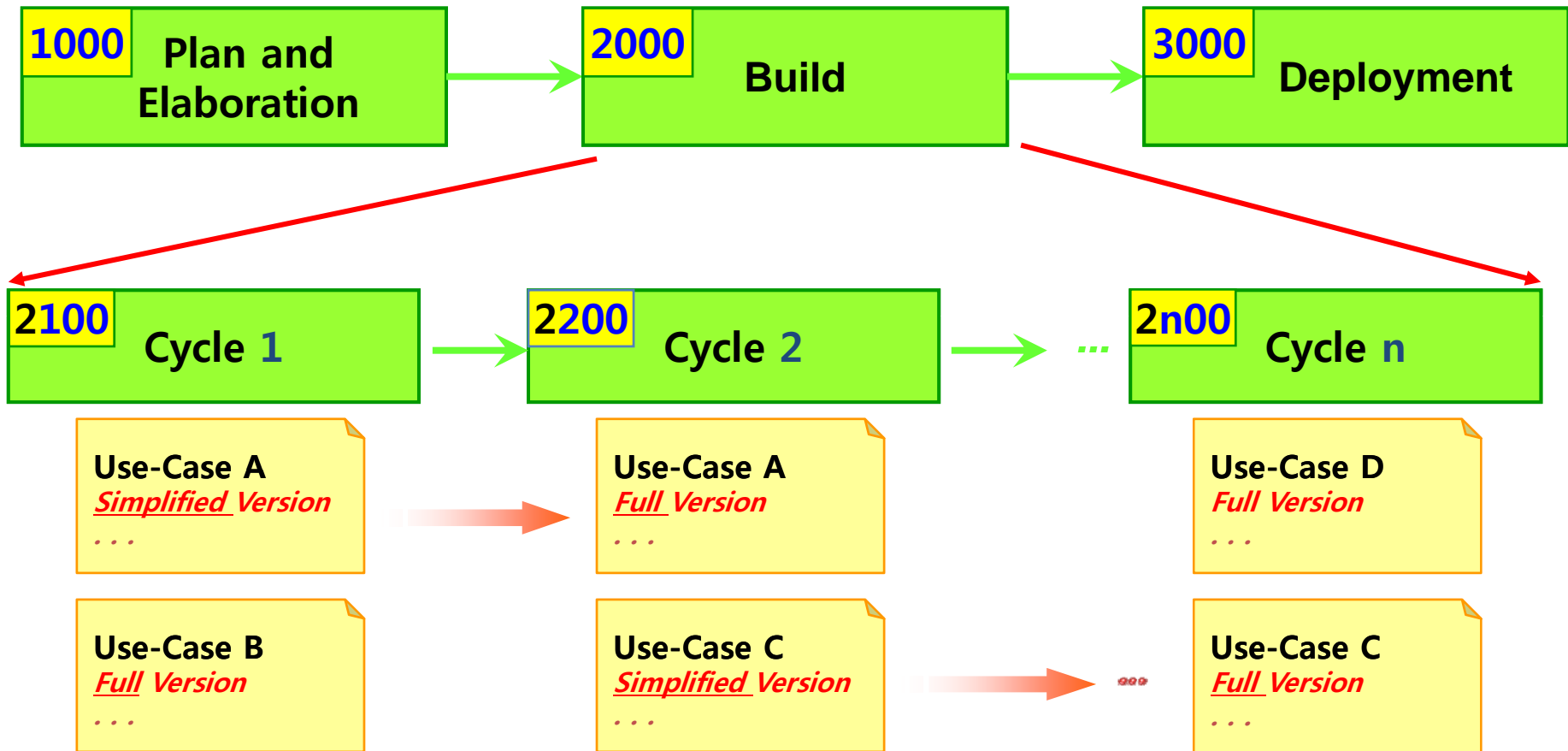
- Stage 1000 : Plan and Elaboration
  - Planning, defining requirements, building prototyping, etc
  - Corresponding to Inception/Elaboration phases in the RUP
- Stage 2000 : Build
  - Construction of the system
  - Corresponding to Construct phase in the RUP
- Stage 3000 : Deployment
  - Implementation of the system into use
  - Corresponding to Transition phase in the RUP

## 2. Iterative Development

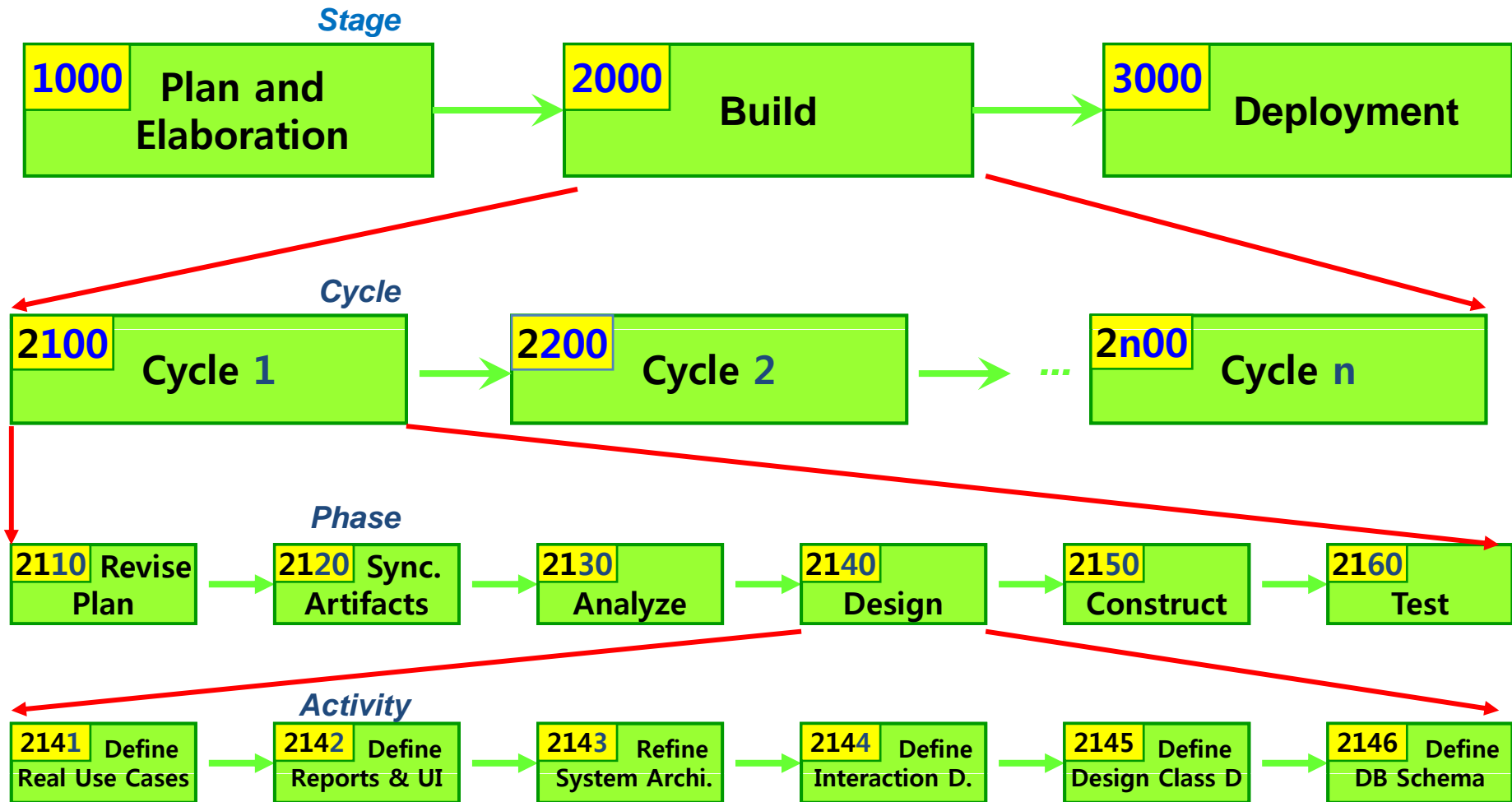
- Multiple iterations in the Build stage
- Each iteration took about 2 to 8 weeks



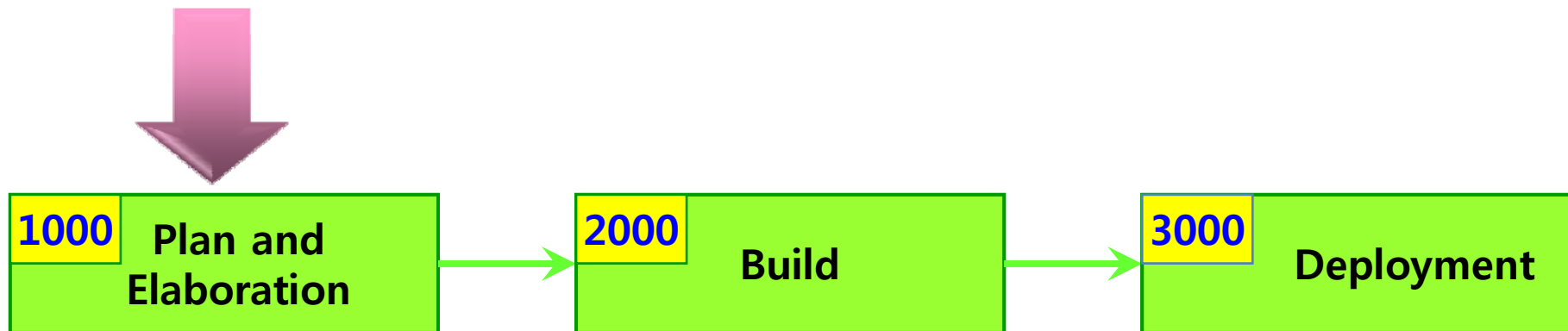
# 3. Incremental Development



# 4. Architecture of OSP



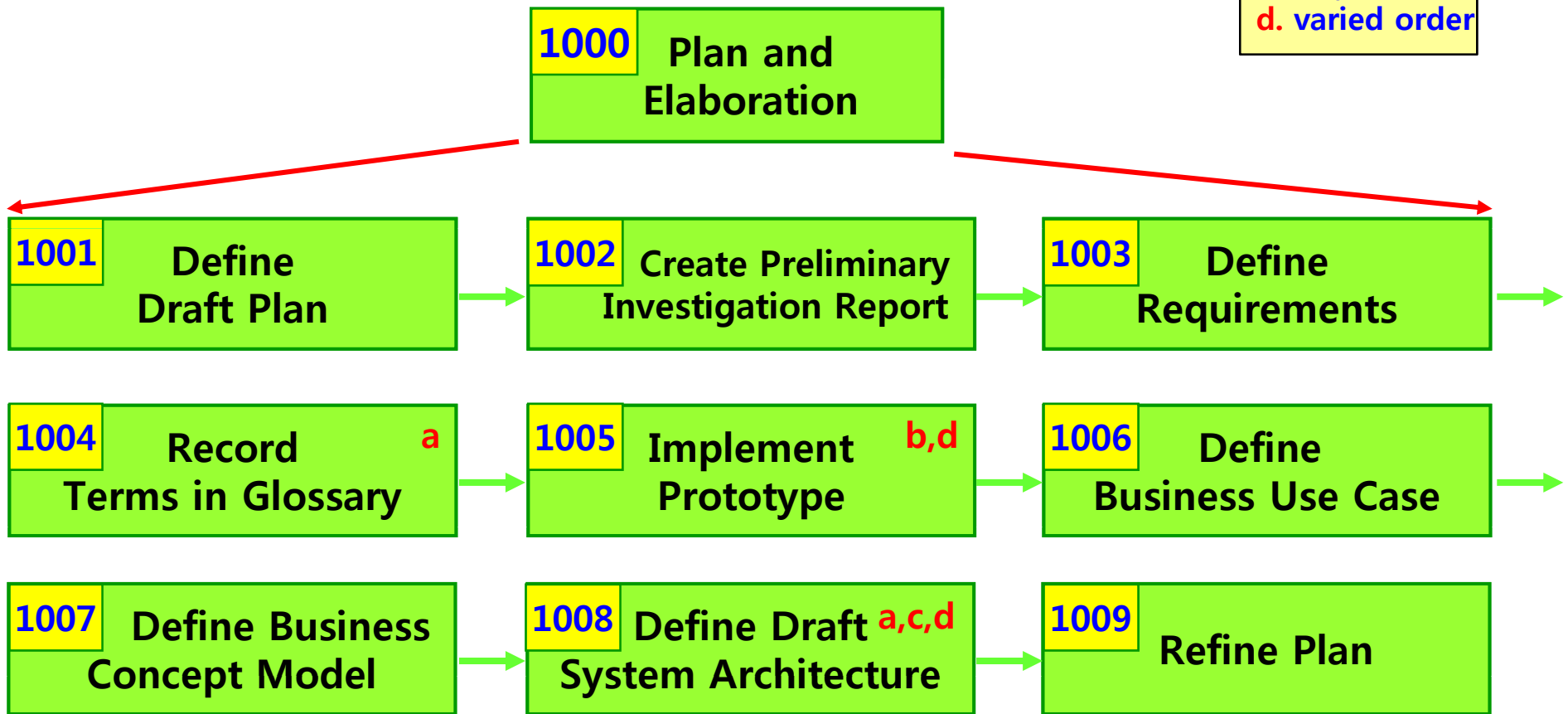
# Stage 1000. Plan and Elaboration



# Stage 1000. Plan and Elaboration

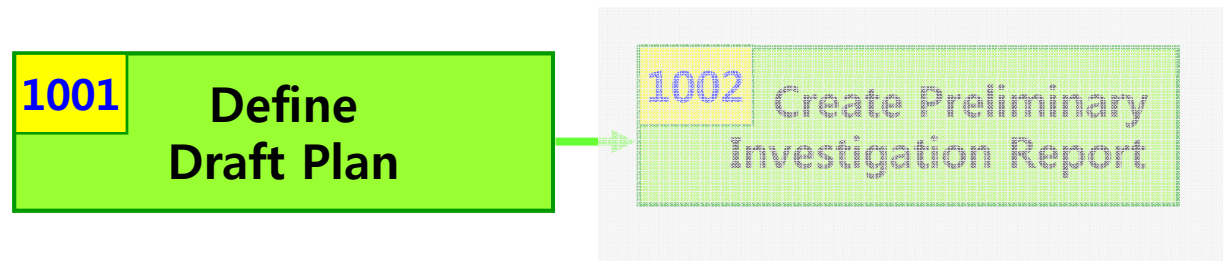
- Stage 1000 Activities

a. ongoing  
 b. optional  
 c. may defer  
 d. varied order





# Activity 1001. Define Draft Plan



- Description
  - Write a draft plan for schedule, resources, budget, objective, etc
  - Input : related documents of previous similar projects
  - Output : a draft project plan
- Steps
  1. Write motivation and objective of project
  2. Write scope of project
  3. Identify and write functional requirements
  4. Identify and write non-functional requirements
  5. Estimate resources (human efforts(M/M), human resources, duration, budget)

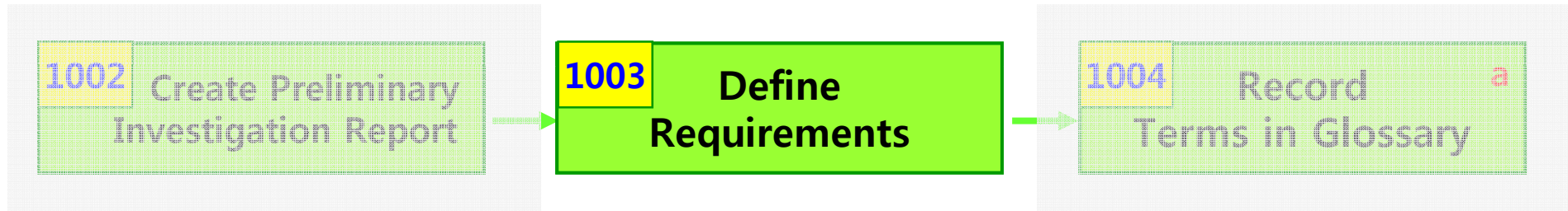
# Activity 1002.

## Create Preliminary Investigation Report



- Description
  - Write an investigation report on alternatives, business needs, risk, etc
  - Input : draft project plan
  - Output : an investigation report
- Steps
  1. Write alternative solutions
  2. Write project's justification (business needs)
  3. Identify and manage risks, and write risk reduction plans
  4. Analyze business market
  5. Write managerial issues

# Activity 1003. Define Requirements



- Description
  - Write a requirement specification for a product
  - Input : draft project plan, investigation report
  - Output : a requirement specification
- What is a requirement? (IEEE Std 610.12-1990)
  - A condition or capability needed by a user to solve a problem or achieve an objective.
  - A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.
  - A documented representation of a condition or capabilities as in (1) or (2)

# Activity 1003. Define Requirements

- Functional requirements
  - A requirement that specifies a function that a system or system component must be able to perform
  - Analyzed and Realized in Use-Case model
- Non-functional requirements
  - Constraints on the services or functions offered by the system as timing constraints, constraints on the development process, standards, etc.
  - Portability, Reliability, Usability, Efficiency(Space, Performance)
  - Delivery, Implementation, Standards
  - Ethical, Interoperability, Legislative(Safety, Privacy)
- Recommended reference : IEEE Std. 830-1998

# Activity 1003. Define Requirements

- Steps
  1. Gather all kinds of useful documents
  2. Write an overview statement (objective and name of the system, etc.)
  3. Determine customers who use the product
  4. Write goals of the project
  5. Identify system functions
    - Functional requirements
    - Add function references(such as R1.1, ...) into the identified functions
    - Categorize identified functions into Event, Hidden, and Frill
  6. Identify system attributes
    - Non-functional requirements
  7. Identify other requirements (Optional)
    - Assumptions, Risks, Glossary, etc.

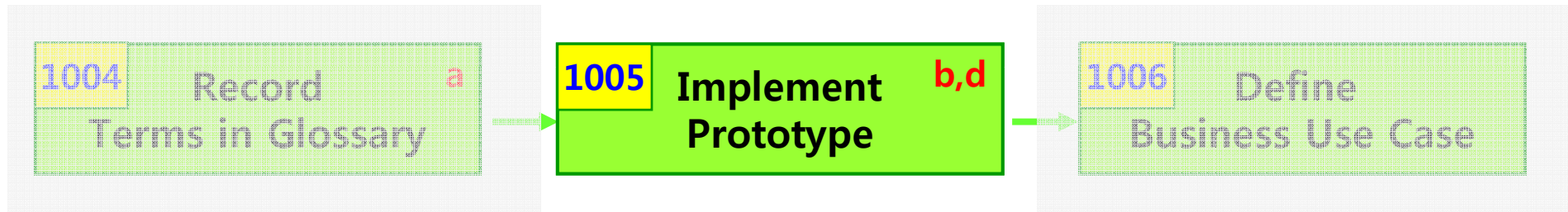
Event:	should perform / visible to users
Hidden:	should performs / invisible to users
Frill:	optional

# Activity 1004. Record Terms in Glossary



- Description
  - Similar to “Data Dictionary”
  - Dictionary of terms and any associated information(constraints and rules)
  - Input : requirements specification
  - Output : a term dictionary
- Steps
  1. Describe meaning of terms specified in requirements specification
  2. Write alias of each term

# Activity 1005. Implement Prototype



- Description
  - Develop a prototype system to permit use feedback, determine feasibility, or investigate timing or other issues
  - Input : requirements specification
  - Output : a prototype
- Steps
  1. Develop a prototype

# Activity 1006. Define Business Use Case

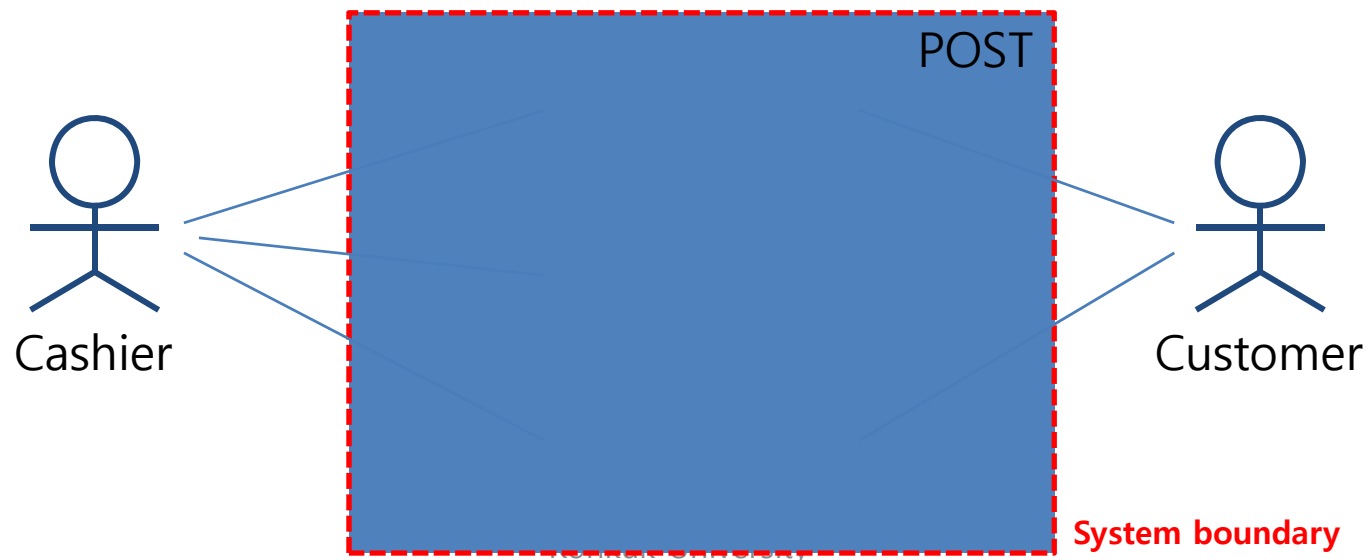


- Description
  - To obtain a deeper understanding of the processes and requirements identified so far
  - Identify business tasks as business use cases, and illustrate their relationships in use case diagrams
  - Input : requirements specification
  - Output : a business use case model (High-level use case)
    - Business Use Case Diagram
    - Business Use Case Description



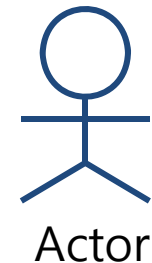
# Activity 1006. Define Business Use Case

- Steps
  1. Determine system boundary in order to identify what is external versus internal, and what the responsibilities of the system are
    - Typical system boundary includes:
      - Hardware/Software boundary of a device / computer system
      - Department of an organization
      - Entire organization



# Activity 1006. Define Business Use Case

2. Identify the actors related to a system or organization
  - An actor is anything with behavior, including the system under discussion(SuD) itself when it calls upon the services of other systems
  - Actors are not only the roles played by people, but also organizations, software, and machines
  - Primary Actors
    - Have user's goals fulfilled through using services the system provides
    - Primary actors can be other computer systems (i.e. watchdog)
  - Supporting Actors
    - Provide services to the system under design
    - Often a computer system could be a supporting actor



# Activity 1006. Define Business Use Case

3. Identify user goals for each actor
4. Record the primary actors and their goals in an actor-goal list

Actor	Goal
Cashier	Process sales Process rentals Handle returns Cash in Cash out ...
System Admin.	Add users Modify users Delete users Manage securities ...

# Activity 1006. Define Business Use Case

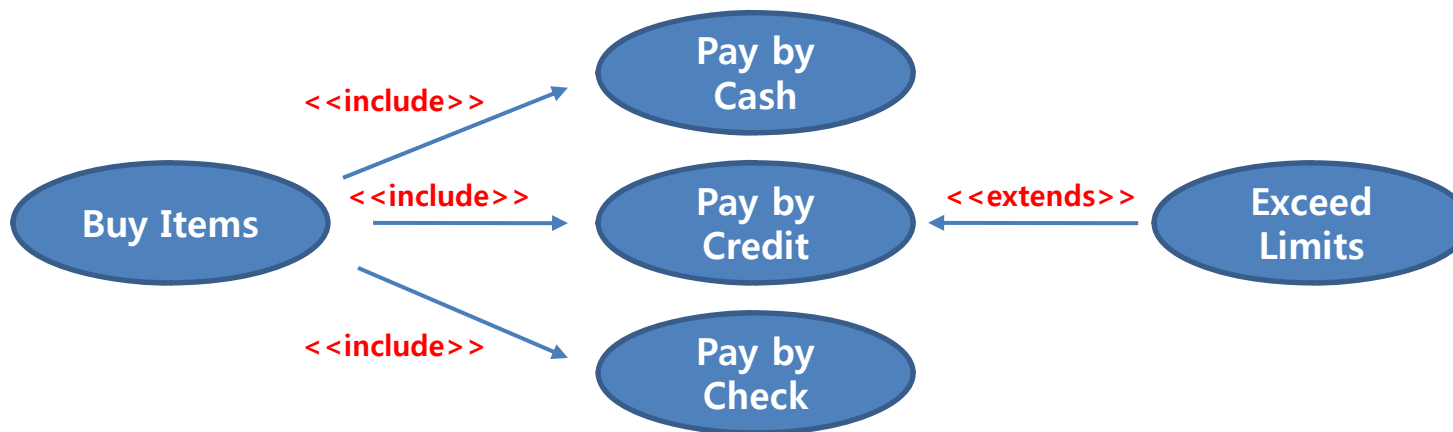
5. Define use cases that satisfy user goals
  - Identify use cases by actor-based
    - For each actor, identify the processes they initiate or participate in
  - Identify use cases by event-based
    - Identify the external events that a system must respond to
    - Related the events to actors and use cases
  - Name them according to their goals
6. Allocate system functions identified during the requirements specification into related use cases
7. Categorize identified use cases into primary, secondary, and optional use cases
  - Primary use cases : major common processes
  - Secondary use cases : minor or rare processes
  - Optional use cases : processes that may not be tackled



Use Case

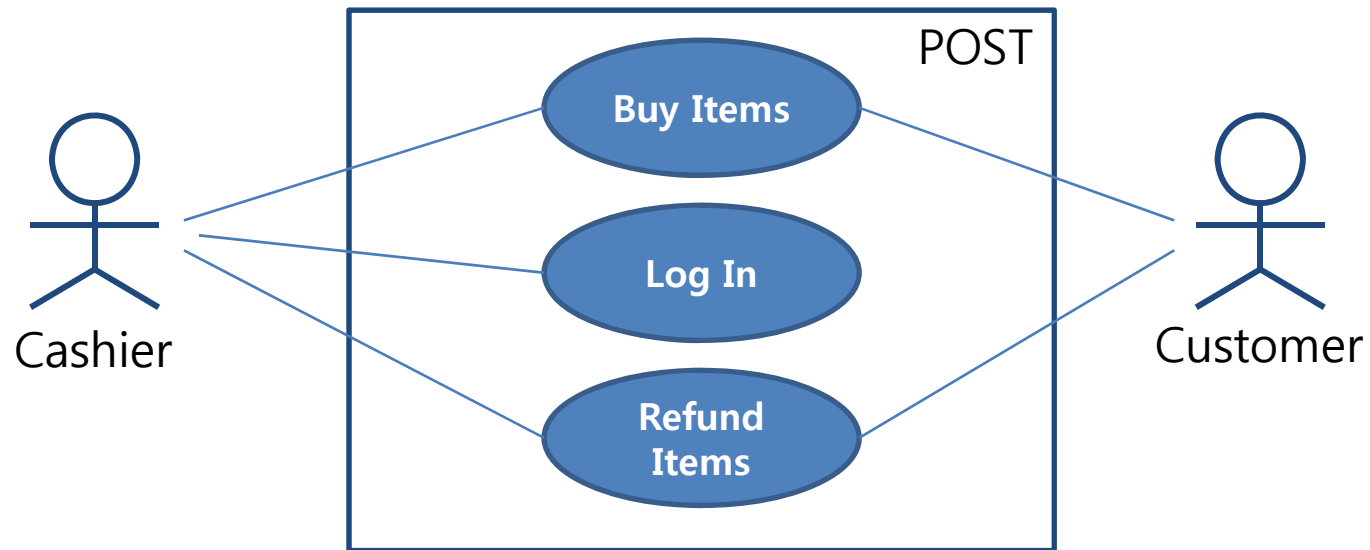
# Activity 1006. Define Business Use Case

8. Identify relationships between use cases
  - Write major steps or branching activities of one use case as several separate use cases using "include" relationship, when they are too complex, long, and duplicated to understand
  - Use "extends" relationship when an exceptional activity is occurred in use case



# Activity 1006. Define Business Use Case

9. Draw a use case diagram



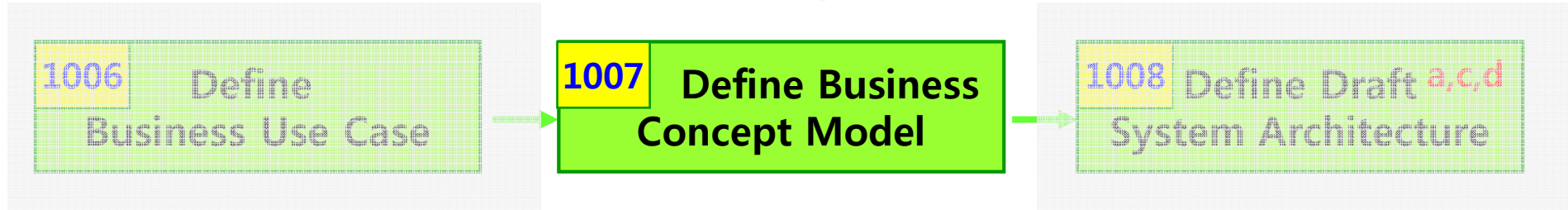
# Activity 1006. Define Business Use Case

10. Rank use cases according to the followings:
  - a. Significant impact on the architectural design
  - b. Significant information and insight regarding the design
  - c. Include risky, time-critical, or complex functions
  - d. Involve significant research, or new and risky technology
  - e. Represent primary line-of-business processes
  - f. Directly support increased revenue or decreased costs
  - The ranking scheme may use a simply fuzzy classification such as high-medium-low
  - High ranking use cases need to be tackled in early development cycle

Rank	Use case	Justification
High	Buy Items	It's the triggering event of all processes
Medium	Add New Users Log In Refund Items	Affects security ... ...
Low	Cash out Start Up Shut Down	Minimum effect on the architecture ... ...

# Activity 1007.

## Define Business Concept Model

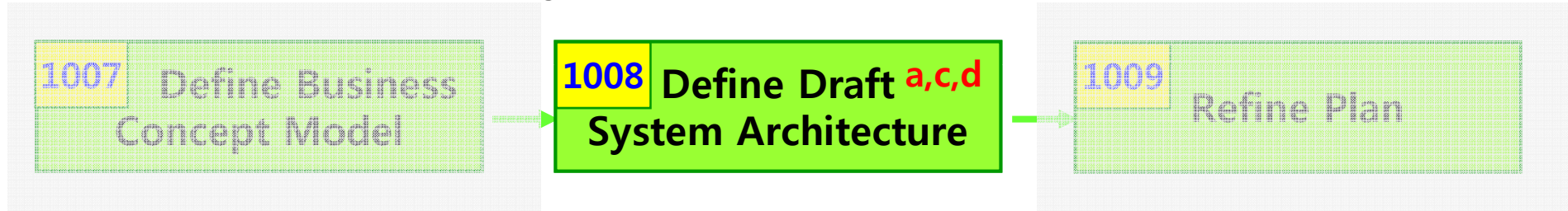


- Description
  - Identify “business concept” in the target domain which can be candidates for “classes”
  - Input : requirements specification  
term dictionary  
business use case model
  - Output : a business concept model
- Steps
  1. Identify business terms or business concepts from requirements specification or through interviews with domain experts
  2. Define identified terms as business concepts
    - Implementation details can’t be business concepts



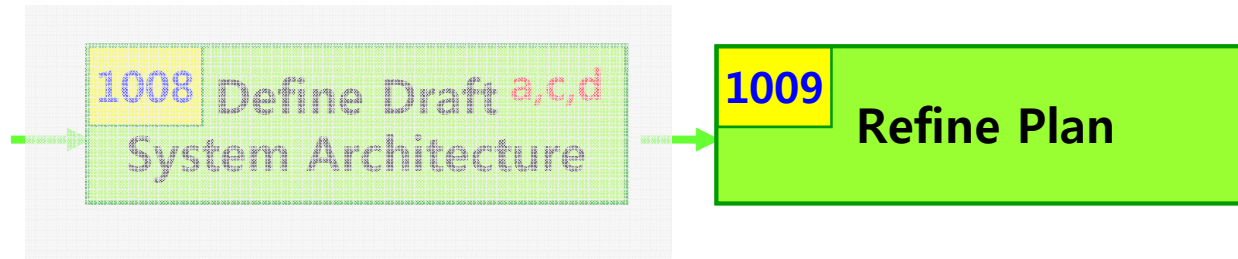
# Activity 1008.

## Define Draft System Architecture



- Description
  - Construct a rough preliminary system architecture model
  - Input : requirements specification  
business use case model
  - Output : a draft system architecture
- Steps
  1. Define logical/physical layers of the target system
  2. Separate the whole system into several subsystems
  3. Assign business use cases into each subsystem
  4. Identify and draw up hardware resources

# Activity 1009. Refine Plan



- Description
  - Refine the draft project plan generated in activity 1001
  - Input : all outputs of OSP stage 1000
  - Output: a refined project report
- Steps
  1. Review draft project plan, based on requirements specification, business use case model, business concept model, and draft system architecture
  2. Refine project's scope, duration, cost, and other resources

# Summary

- Why the name of the process is OSP?
- What is the characteristics of OSP?
  - Can you clarify the difference between RUP and OSP?
- What is the objective of OSP stage 1000?
  - Can you picture the flow of stage 1000?