Modelling Concepts

Based on Chapter 5
Bennett, McRobb and Farmer

Object Oriented Systems Analysis and Design Using UML

4th Edition, McGraw Hill, 2010

In This Lecture You Will Learn:

- What is meant by a model
- The distinction between a model and a diagram
- The UML concept of a model

What is a Model

- Like a map, a model represents something else
- A useful model has the right level of detail and represents only what is important for the task in hand
- Many things can be modelled: bridges, traffic flow, buildings, economic policy

Why Use a Model?

- A model is quicker and easier to build than the real thing
- A model can be used in a simulation
- A model can evolve as we learn
- We can choose which details to include in a model
- A model can represent real or imaginary things from any domain

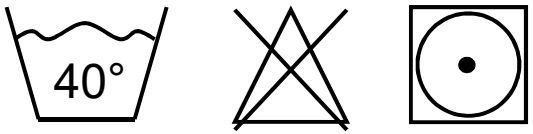
Modelling Organizations

Organizations are human activity systems.

- The situation is complex
- Stakeholders have different views
- We have to model requirements accurately, completely and unambiguously
- The model must not prejudge the solution

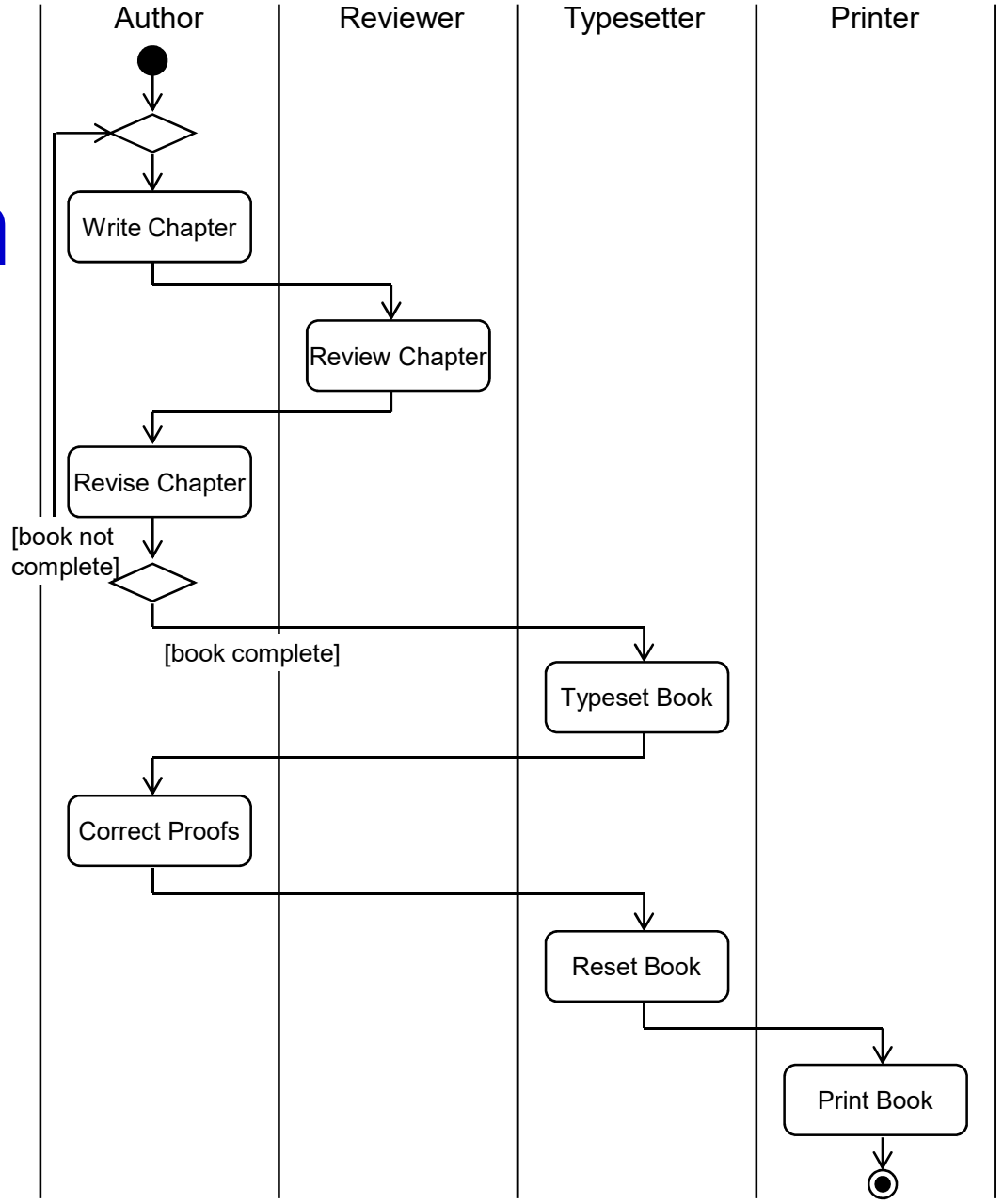
What is a Diagram?

- Abstract shapes are used to represent things or actions from the real world
- Diagrams follow rules or standards
- The standards make sure that different people will interpret the diagram in the same way

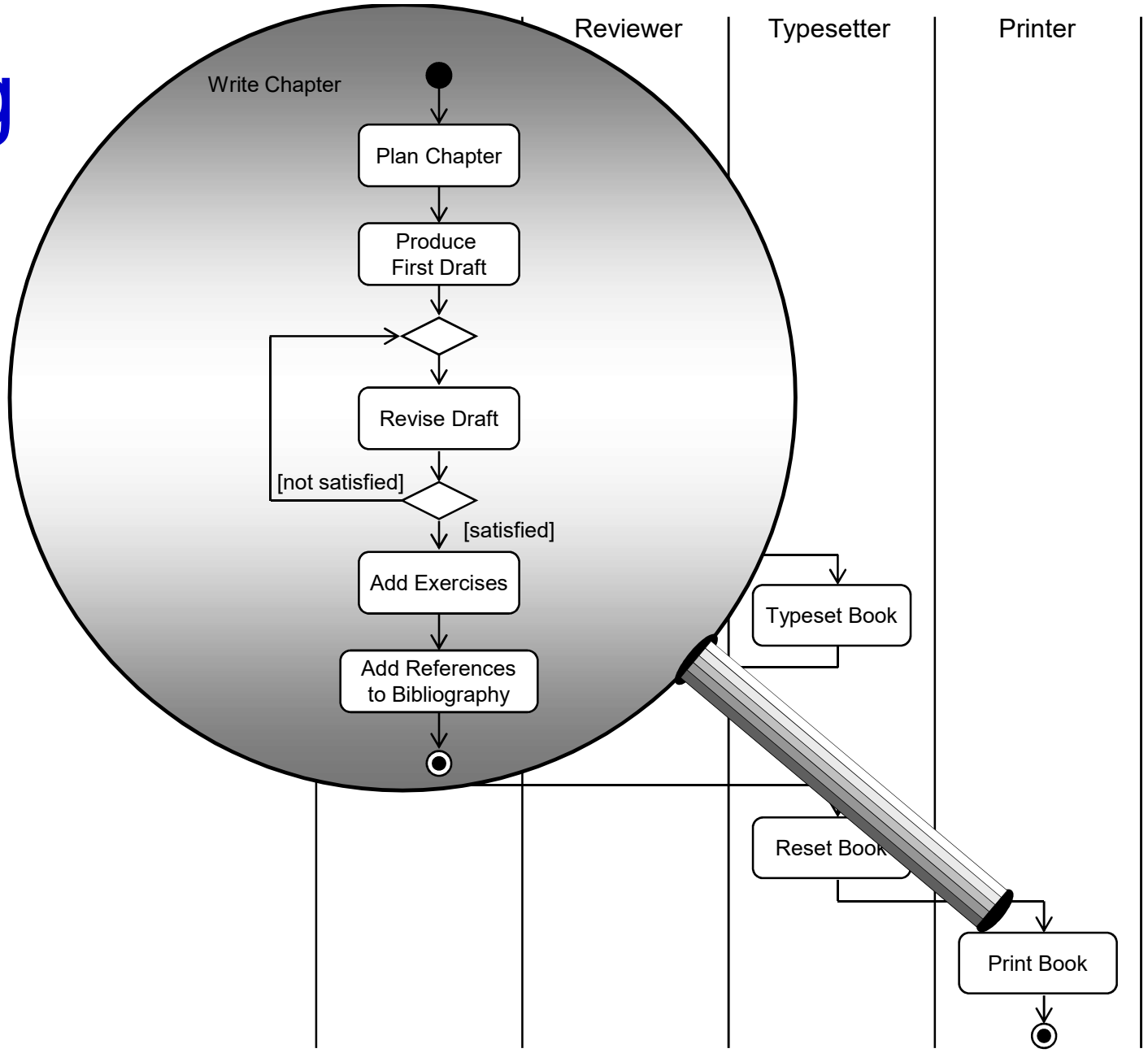


An Example of a Diagram

- An activity diagram of the tasks involved in producing a book.

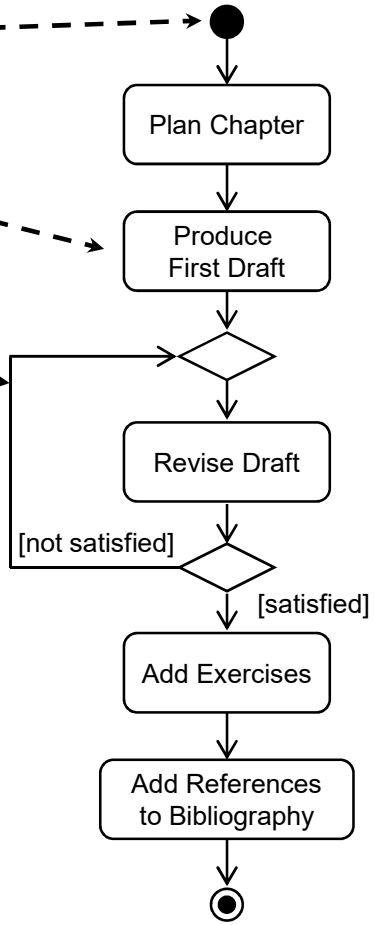


Hiding Detail



Diagrams in UML

- UML diagrams consist of:
  - icons
  - two-dimensional symbols
  - paths
  - Strings
- UML diagrams are defined in the UML specification.

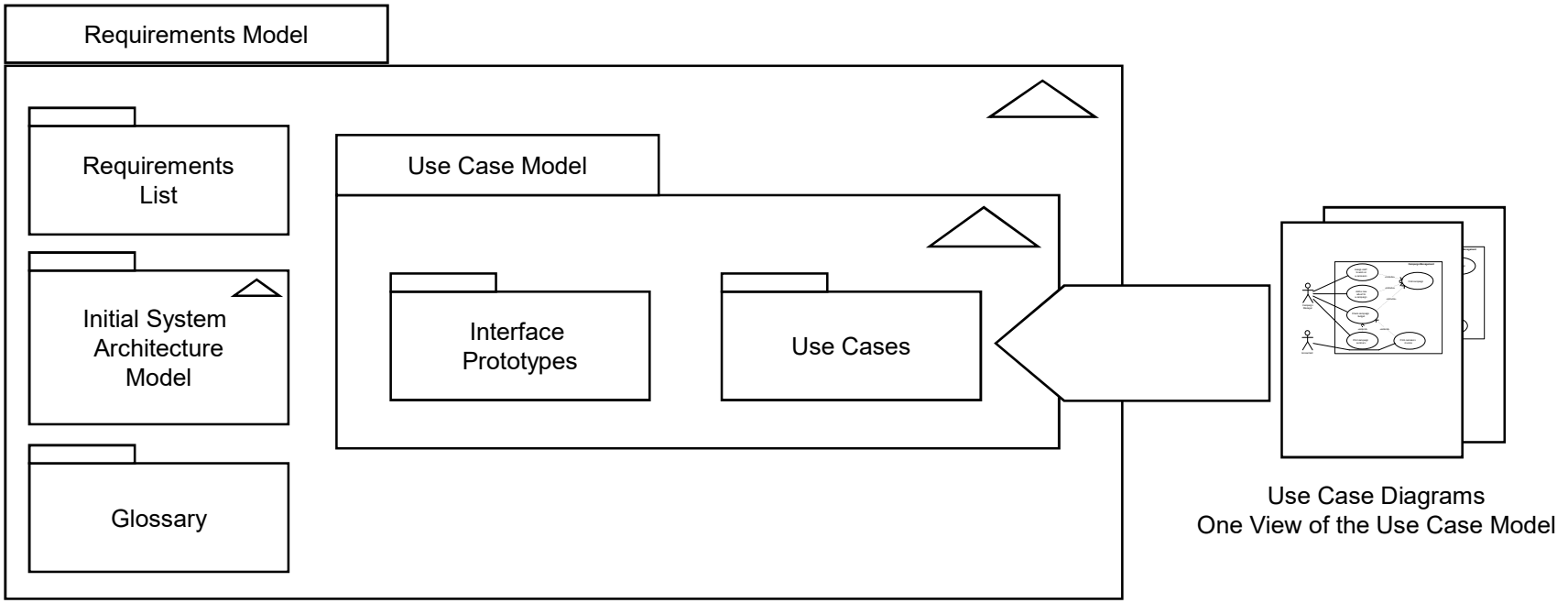


Diagrams vs. Models

- A diagram illustrates some aspect of a system.
- A model provides a complete view of a system at a particular stage and from a particular perspective.
- A model may consist of a single diagram, but most consist of many related diagrams and supporting data and documentation.

Relationship between Models and Diagrams

- Use Case Diagrams are one view of the Use Case Model in the Requirements Model



Examples of Models

- Requirements Model
  - complete view of requirements
  - may include other models, such as a Use Case Model
  - includes textual description as well as sets of diagrams

Examples of Models

- Behavioural Model
  - shows how the system responds to events in the outside world and the passage of time
  - an initial model may just use Communication Diagrams
  - a later model will include Sequence Diagrams and State Machines

Models in UML

- A system is the overall thing that is being modelled
- A subsystem is a part of a system consisting of related elements
- A model is an abstraction of a system or subsystem from a particular perspective
- A model is complete and consistent at the chosen level of abstraction

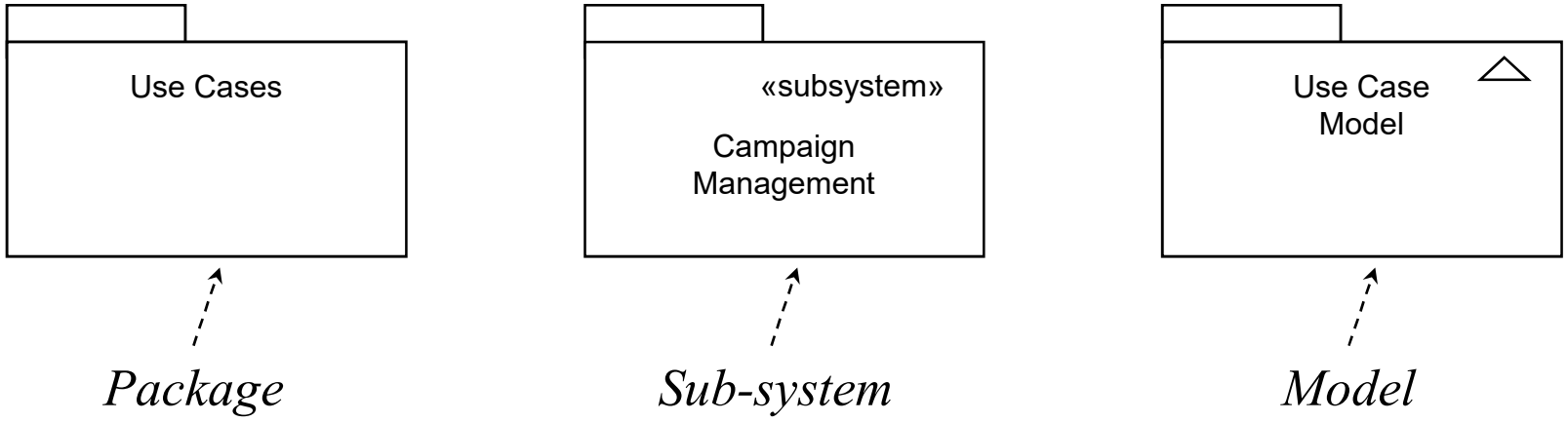
Models in UML

- Different models present different views of the system, for example:
  - use case view
  - design view
  - process view
  - implementation view
  - deployment view

(Booch et al., 1999)

Packages, Sub-systems and Models

- UML has notation for showing subsystems and models, and also for packages, which are a mechanism for organising models (e.g. in CASE tools)

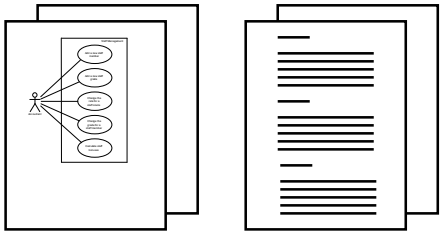


Developing Models

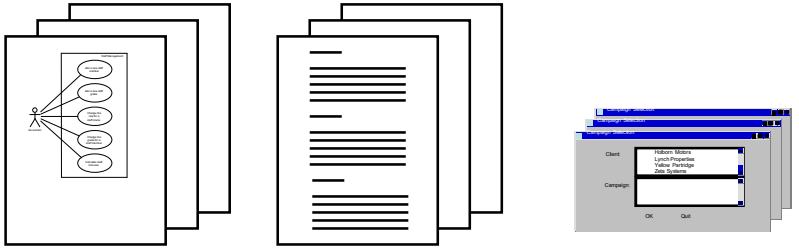
- During the life of a project using an iterative life cycle, models change along the dimensions of:
  - abstraction—they become more concrete
  - formality—they become more formally specified
  - level of detail—additional detail is added as understanding improves

Development of the Use Case Model

Iteration 1
Obvious use cases.
Simple use case descriptions.



Iteration 2
Additional use cases.
Simple use case descriptions.
Prototypes.



Iteration 3
Structured use cases.
Structured use case descriptions.
Prototypes.

Summary

In this lecture you have learned about:

- What is meant by a model
- The distinction between a model and a diagram
- The UML concept of a model

References

- Booch, Rumbaugh and Jacobson (1999)
- Bennett, Skelton and Lunn (2005)

  (For full bibliographic details, see Bennett, McRobb and Farmer)