

Ryan Somma

From page 362 in your textbook, please answer the following review questions:

4. How is a data flow diagram different from a flowchart?

A flowchart depicts the flow of processes and is useful in the construction of algorithms in software programs. Only one process can be performed at a time in a flowchart. A data flow diagram depicts only the flow of data through a system. Data processes may be executed in parallel in a data flow diagram.

5. What is decomposition, and why is it needed? What is the tool used to depict the decomposition of the model?

Decomposition is the act of breaking down a system into smaller, more manageable parts. Most systems are too large and complex to be understood as a whole, and decomposition simplifies it by allowing the analyst to focus on its smaller components. A Decomposition Diagram illustrates the breakdown of a system into smaller parts, which form a hierarchy of parent and child processes.

6. What are the three types of logical processes?

Functions are sets of ongoing business activities. Events or transactions are logical units of work that must be completed as a whole, triggered by inputs and completed by providing outputs. Elementary processes are detailed tasks required to complete a response to an event.

11. What is data conservation, and why is it needed?

Data Conservation is the practice of ensuring that a data flow does not include any data not needed by the receiving process. This practice contributes to Business Process Redesign by simplifying the interfaces between processes.

12. What are external agents and why can the external agents of an information system change?

External Agents is anything that interacts with the system, but is outside of the system scope. External agents can become part of a system if the system scope expands to include them, or additional external agents can emerge if the system scope shrinks to exclude them.

13. What are some examples of the event-driven modeling used in systems analysis?

Event partitioning factors a system into subsystems based on business events and responses to those events. An event-response list lists the business events the system must

provide a response to. Event handler processes handles events from the event-response list. An event diagram provides a data flow diagram for a single event handler.

From page 408 in your textbook, please answer the following questions:

1. Since its inception in 1997, the Unified Modeling Language (UML) has quickly gained wide acceptance and usage throughout the world.

- a) **In terms of object modeling, what *does* UML provide to designers? What *doesn't* UML provide?**
- b) **What was the reason that UML was developed?**
- c) **What might object modeling look like today if UML had *not* been developed?**

UML allows designers to describe software in terms of objects; however, UML does not provide a methodology for designing the actual system itself. UML was developed to provide an industry standard for object modeling, and without it, there might not be a standard today, and designers would encounter different object modeling practices on different projects with different notations and methodologies.

2. Object-Oriented analysis (OOA) and object modeling have become familiar terms in many organizations, but their underlying concepts are not always intuitive and can be difficult to understand, especially by nontechnical users who are involved in a systems development project.

- a) **In nontechnical terms, explain what an object is and what the object-oriented analysis approach is.**
- b) **Also in nontechnical terms, explain the technique of object modeling.**
- c) **What are the main differences between object-oriented analysis and traditional systems analysis in how they approach system development?**
- d) **Do you think it would be easier to learn object-oriented analysis methods if you were a systems designer experienced in traditional development methods, or if this was the first analysis method you were learning? Explain your answer.**

An object is a thing, anything. It can be a person, a car, a street, a government agency, a webpage. It can be anything. The Object-oriented analysis approach looks at a system and breaks it down into the objects it deals with, defining the properties of those objects and the methods or services they use to interact with other objects in the system. Object Modeling is a technique for identifying the objects in a system and identifying the relationships between them.

Where traditional process modeling defines a system in terms of its processes, OOA defines a system in terms of its objects, and then defines the processes as methods encapsulated by those objects, which interact with other objects.

In my personal experience, after programming and designing systems for 10 years in traditional, procedural fashion, it has been an incredible paradigm shift to move toward OOP. It has also been a huge shift in thinking for my coworkers who have also had to make the transition. I believe it would be easier to start out thinking in Object Oriented

patterns rather than switch, because our traditional procedural methodologies keep impacting our Object Oriented solutions. We keep treating our Objects as if they were Functions, without truly taking advantage of the Object Oriented methodology.

3. Consider a movie DVD as an example of an object.

- a) Using the textbook's terminology, what type of object is a movie DVD?
- b) What are some of the attributes of a movie DVD?
- c) What is an object instance of a movie DVD?
- d) Represent the class of *Movie DVD* in an object model using UML notation, as shown in Figure 11-2. Include the class name, attributes, and behaviors.
- e) Would the object class of *Movie DVD* be considered a supertype or subtype? Give examples.

A DVD is a thing with attributes such as Title, ISBN, Running Time, Copy Right Date, Distributor, and Genre. An object instance of the DVD class would be individual copies of The Matrix, Lawnmower Man, Ghost in the Shell, or any other actual, physical DVD.

A “DVD” Object Class:

DVD
-title -format -subtitles -language -runningTime -distributor -rating -copyRight
+play() +specialFeature() +deletedScenes() +commentary() +subtitles()

A DVD can be a supertype if we expand my above definition, which I restricted to just movies. There are also software DVDs and video game DVDs, which encapsulate slightly different properties and methods. A DVD can also be a subtype of Media, alongside books, websites, and CDs.