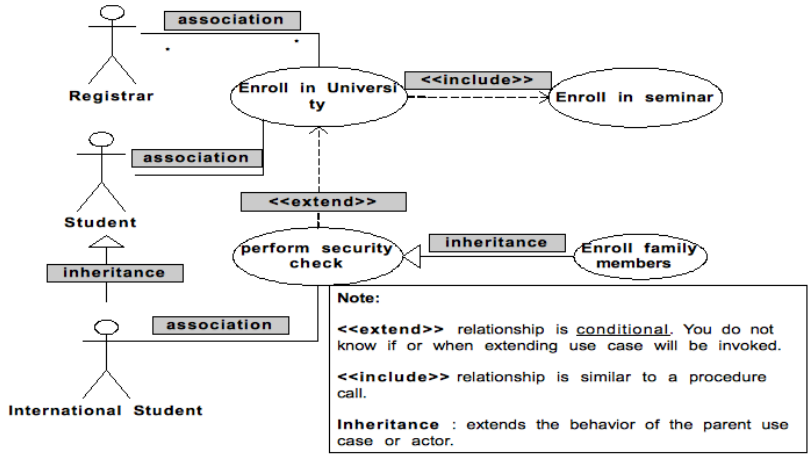
♦ 12 UML interview Questions & Answers

Posted on August 20, 2014 by Arulkumaran Kumaraswamipillai

Q1. What are the different types of UML diagrams?
A1. Use case diagrams, Class diagrams, Package diagrams, Object diagrams, Sequence diagrams, Collaboration diagrams, and State chart diagrams.

Q2. What is a use case diagram, and when will you use it?
A2. Depicts the typical interaction between external users (i.e. actors) and the system. The emphasis is on what a system does rather than how it does it. A use case is a summary of scenarios for a single task or goal. An actor is responsible for initiating a task. The connection between actor and use case is a communication association.



UML – use case diagram

Capturing use cases is one of the primary tasks of the elaboration phase of RUP. In its simplest usage, you capture a use case by talking to your users and discussing the various things they might want to do with the system.

When to use 'use case' diagrams?

1) Determining user requirements. New use cases often generate new requirements.
2) Communicating with clients. The simplicity of the diagram makes use case diagrams a good way for designers and developers to communicate with clients.
3) Generating test cases. Each scenario for the use case may suggest a suite of test cases.

Q3. What are the different types of associations in class diagrams, and when will you use it?
A3. Class diagram technique is vital within Object Oriented methods. Class diagrams describe the types of objects in the system and the various static relationships among them. Class diagrams also show the attributes and the methods. Class diagrams have the following possible relationships:

Association: A relationship between instances of 2 classes.

Aggregation: An association in which one class belongs to a collection (does not always have to be a collection. You can

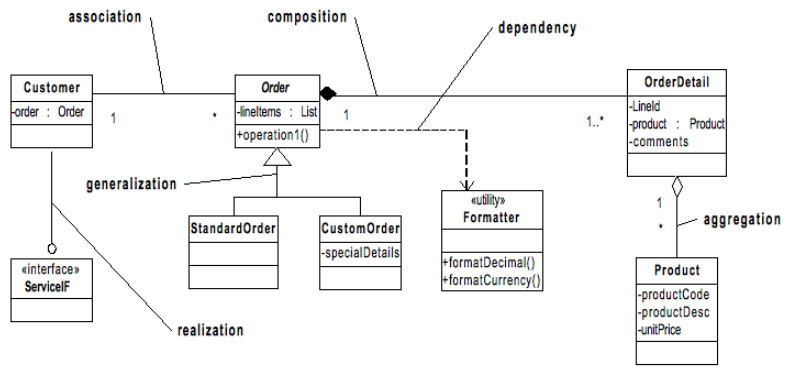
also have cardinality of “1”). This is a part of a whole relationship where the part can exist without the whole. For example: A line item is whole and the products are the parts. If a line item is deleted then the products need not be deleted.

Composition: An association in which one class belongs to a collection (does not always have to be a collection. You can also have cardinality of “1”). This is a part of a whole relationship where the part cannot exist without the whole. If the whole is deleted then the parts are deleted. For example: An Order is a whole and the line items are the parts. If an order is deleted then all the line items should be deleted as well (i.e. cascade deletes).

Generalization: An inheritance link indicating that one class is a superclass of the other. The Generalization expresses the “is a” relationship whereas the association, aggregation and composition express the “has a” relationship.

Realization: Implementation of an interface.

Dependency: A dependency is a weak relationship where one class requires another class. The dependency expresses the “uses” relationship. For example: A domain model class uses a utility class like Formatter etc.

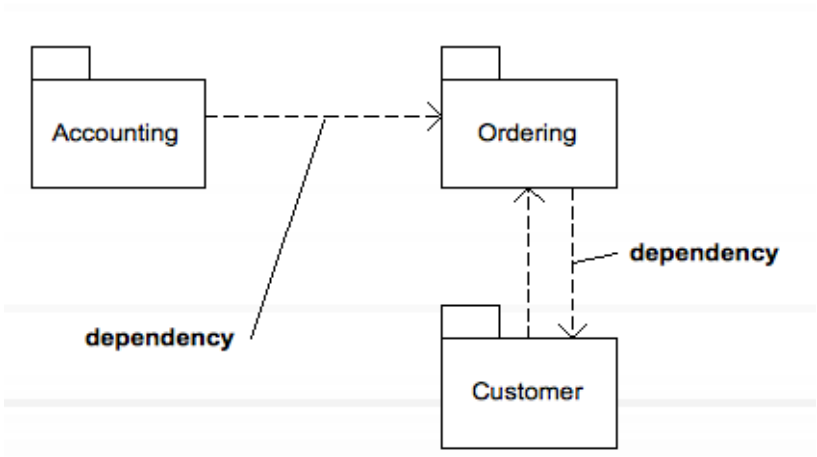


UML – class diagram

When to use class diagrams?

1) Class diagrams are the backbone of Object Oriented methods. So they are used frequently.
2)Class diagrams can have a conceptual perspective and an implementation perspective. During the analysis draw the conceptual model and during implementation draw the implementation model.

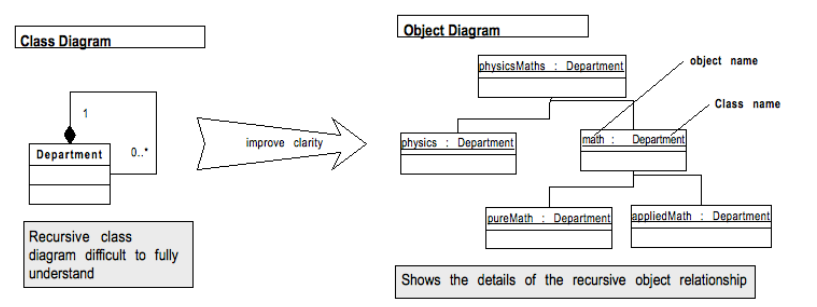
Q4. What is a package diagram, and when will you use it?
A4. Used to simplify complex class diagrams by grouping classes into packages.



UML – package diagram

When to use package diagrams? Package diagrams are vital for large projects.

Q5. What is an object diagram, and when will you use it?
A5. Object diagrams show instances instead of classes. They are useful for explaining some complicated objects in detail about their recursive relationships etc.

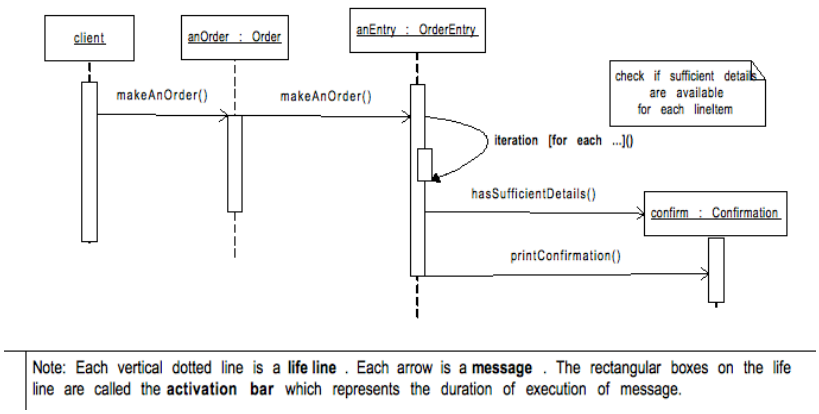


UML – object diagram

When to use object diagrams?

1) Object diagrams are a vital for large projects.
2) They are useful for explaining structural relationships in detail for complex objects.

Q6. What is a sequence diagram, and when will you use it?
A6. Sequence diagrams are interaction diagrams which detail what messages are sent and when. The sequence diagrams are organized according to time. The time progresses as you move from top to bottom of the diagram. The objects involved in the diagram are shown from left to right according to when they take part.

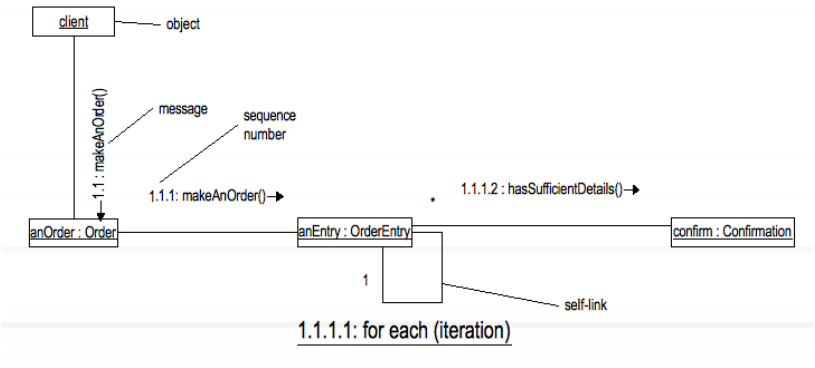


UML – sequence diagram

Q7. What is a collaboration diagram, and when will you use it?
A7. Collaboration diagrams are also interaction diagrams. Collaboration diagrams convey the same message as the sequence diagrams. But the collaboration diagrams focus on the object roles instead of the times at which the messages are sent.

The collaboration diagrams use the decimal sequence numbers as shown in the diagram below to make it clear which operation is calling which other operation, although it can be harder to see the overall sequence. The top-level message is numbered 1. The messages at the same level

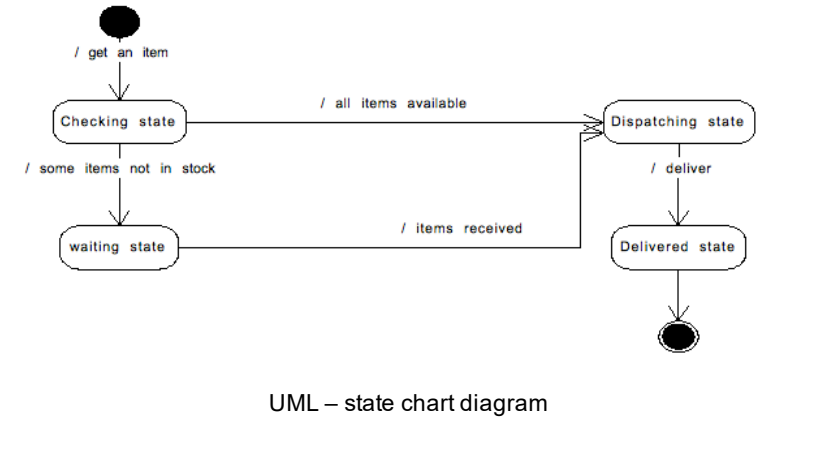
have the same decimal prefix but different suffixes of 1, 2 etc according to when they occur.



UML – collaboration diagram

When to use an interaction diagrams? When you want to look at behavior of several objects within a single use case. If you want to look at a single object across multiple use cases then use state chart diagram as described below.

Q8. What is a state chart diagram, and when will you use it?
A8.Objects have behavior and state. The state of an object depends on its current activity or condition. This diagram shows the possible states of the object and the transitions that cause a change in its state.



UML – state chart diagram

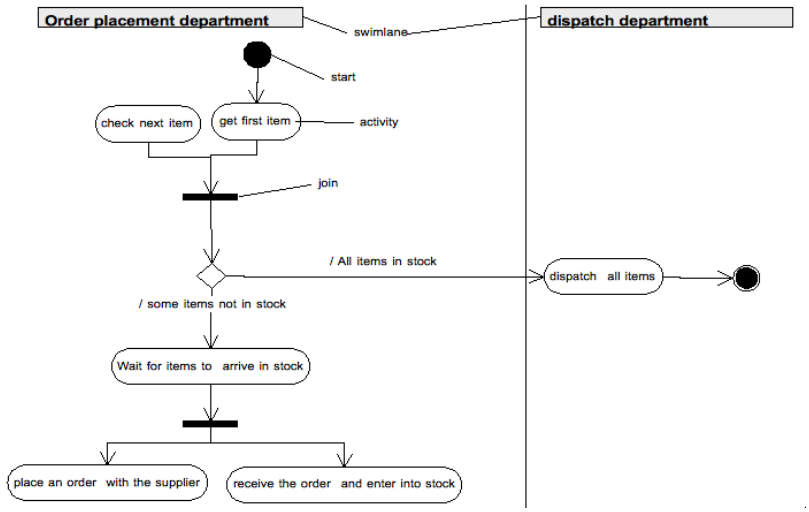
When to use a state chart diagram?

Statechart diagrams are good at describing the behavior of an object across several use cases. But they are not good at

describing the interaction or collaboration between many objects. Use interaction and/or activity diagrams in conjunction with a statechart diagram.

Use it only for classes that have complex state changes and behavior. For example: the User Interface (UI) control objects, Objects shared by multi-threaded programs etc.

Q9. What is an activity diagram, and when will you use it?
A9. Activity diagram is really a fancy flow chart. The activity diagram and statechart diagrams are related in a sense that statechart diagram focuses on object undergoing a transition process and an activity diagram focuses on the flow of activities involved in a single transition process.



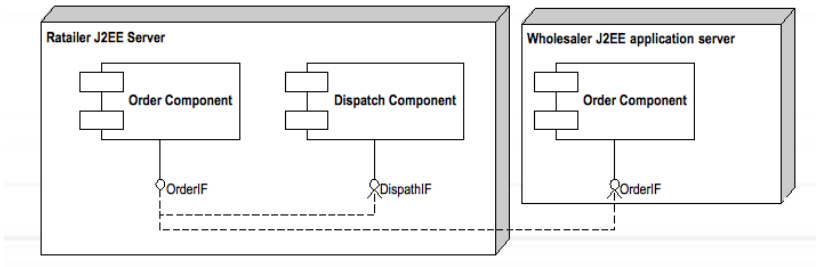
UML -activity diagram

In domain modeling it is imperative that the diagram conveys which object (or class) is responsible for each activity. Activity diagrams can be divided into object swimlanes that determine which object is responsible for which activity. The swim lanes are quite useful because they combine the activity diagram's depiction of logic with the interaction diagram's depiction of responsibility. A single transition comes out of each activity, connecting to the next activity. A transition may join or fork.

When to use activity diagrams?

The activity and statechart diagrams are generally useful to express complex operations. The great strength of activity diagrams is that they support and encourage parallel behavior. The activity and state chart diagrams are beneficial for workflow modeling with multi-threaded programming.

Q10. What is a component and deployment diagram?
A10. A component is a code module. Component diagrams are physical diagrams analogous to a class diagram. The deployment diagrams show the physical configuration of software and hardware components. The physical hardware is made up of nodes. Each component belongs to a node.



UML – component and deployment diagram

Q11. What is the difference between a collaboration diagram and a sequence diagram?
A11. The emphasis of sequence diagram is on the sequence.The emphasis of collaboration diagram is on the object roles.

Q12. What is the difference between aggregation and composition?
A12. **Aggregation:** An association in which one class belongs to another class or a collection. This is a part of a whole relationship where the **part can exist without the whole**. For example: A line item is whole and the products are the parts. If a line item is deleted then the products need not be deleted. (no cascade delete in database terms)

Composition: An association in which one class belongs to another class or a collection. This is a part of a whole relationship where the part cannot exist without the whole. If

the whole is deleted then the parts are deleted. For example: An Order is a whole and the line items are the parts. If an order is deleted then all the line items should be deleted as well (i.e. cascade deletes in database terms).

UML application:

How would you go about designing with UML diagrams.

Popular Posts

♦ 11 Spring boot interview questions & answers

828 views

♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

768 views

18 Java scenarios based interview Questions and Answers

400 views

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

389 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

296 views

♦ 7 Java debugging interview questions & answers

293 views

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

286 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

280 views

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

240 views

001B: ♦ Java architecture & design concepts interview questions & answers

202 views

Bio | **Latest Posts**

Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.**945+** paid members. join my LinkedIn Group. **Reviews**

About Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.**945+** paid members. join my LinkedIn Group. **Reviews**

‹ 04: 5 JAXB interview Questions & Answers

01: ♥♦ 15 Beginner level Java multi-threading interview Q&A ›

Posted in member-paid, UML

Tags: Architect FAQs, Java/JEE FAQs

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ Java generics in no time ☀ Top 6 tips to transforming your thinking from OOP to FP ☀ How does a HashMap internally work? What is a hashing function?

☀ 10+ Java String class interview Q&As ☀ Java auto un/boxing benefits & caveats ☀ Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect

Non-Technical Know Hows

☀ 6 Aspects that can motivate you to fast-track your career & go places ☀ Are you reinventing yourself as a Java developer? ☀ 8 tips to safeguard your Java career against offshoring ☀ My top 5 career mistakes

Prepare to succeed

☀ Turn readers of your Java CV go from “Blah blah” to “Wow”? ☀ How to prepare for Java job interviews? ☀ 16 Technical Key Areas ☀ How to choose from multiple Java job offers?

Select Category

© Disclaimer

 ↑