UML – Interaction Diagrams

Collaboration and Sequence diagrams

Contents

- Interactions
- Collaboration diagrams
- Communication – what does this mean in programming?
- Sequence diagrams

Interactions

- In most programs, we will have more than one class. These classes must work together to achieve a result.
- This is usually achieved by objects calling the functions of other objects, or sending messages.

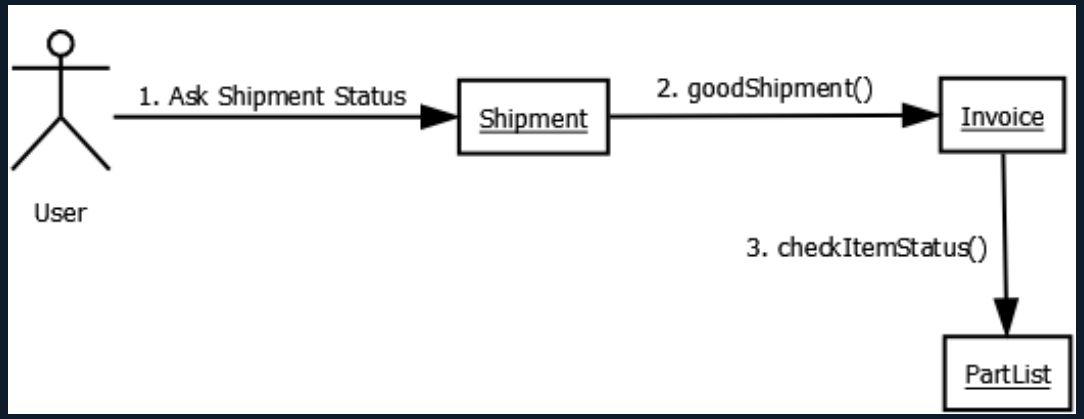
Why Use a UML Interaction Diagram?

- Similar to the reasons for building a class diagram.
- However we need to determine how our classes will work together
- This will help when deciding on the functions to include in each class – and where each of the functions should belong.

What is a UML Interaction Diagram?

- There are 2 types of Interaction diagrams:
  - Collaboration diagrams
  - Sequence diagrams

Collaboration diagram



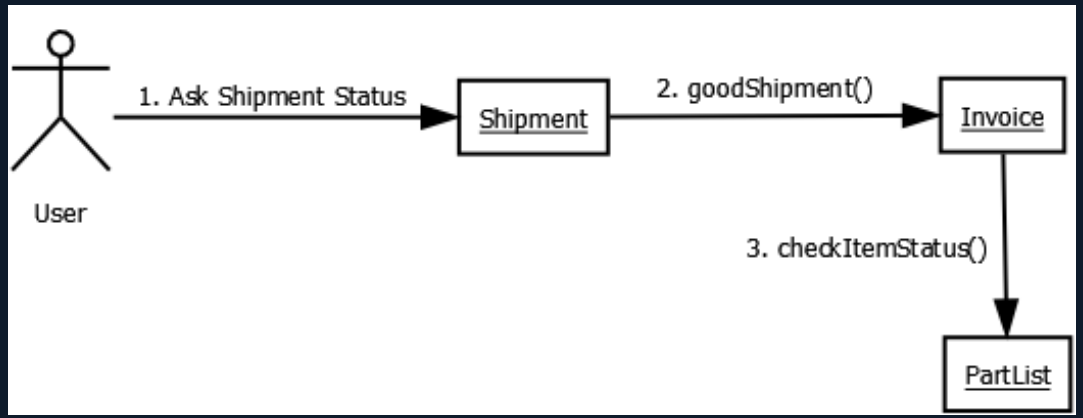
- Collaboration diagrams are used to show communication between objects.
- It’s basically a class diagram with communications superimposed.

Communications?

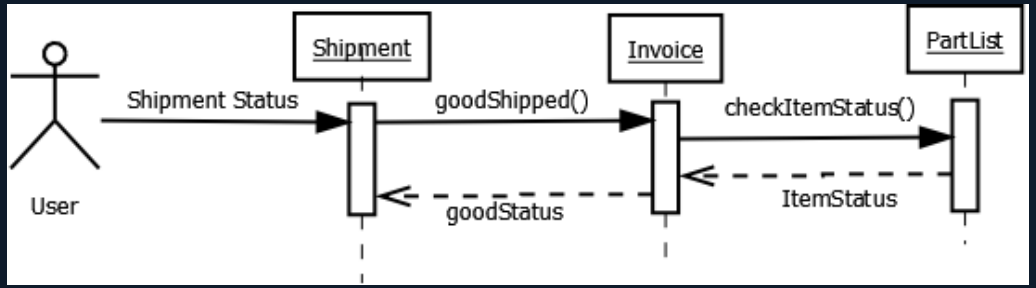
- Communications between objects are usually function calls.
- Communication between a human and the system is normally just described in plain English.
- These are sometimes referred to as messages.
- We use a stick figure (or multiple) to represent the users of our system.

Collaboration Diagram

- Represents a single transaction/option. Usually multiple collaboration diagrams are created for one project.
- The arrows represent links (functions or interactions) between objects. These can be numbered if ordering is important.



Sequence diagram

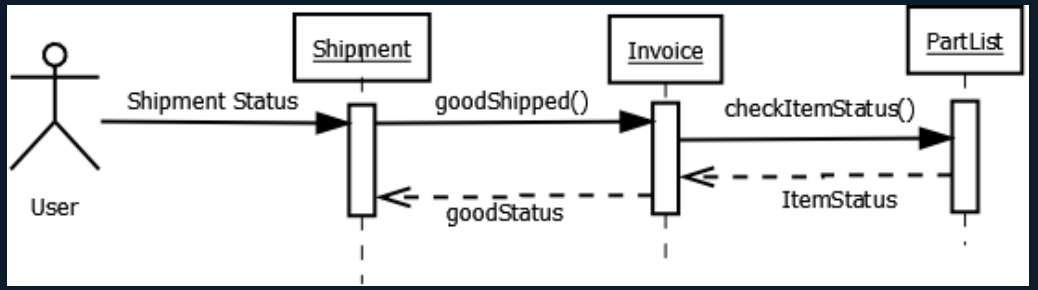


- Sequence diagrams show interactions along a timeline. They basically represent a sequence of messages between objects.

Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Sequence diagram



- The vertical dotted line below each object is it's timeline. Messages are passed from left to right, and then returned data from right to left.
- The timeline is vertical and travels from top to bottom. The return of goodStatus is the last thing to occur in the image above.

Relation to class diagrams

- The objects used in interaction diagrams must be instances of classes listed in your class diagram.
- Any objects which share an interaction in a sequence or collaboration diagram MUST have an association in a class diagram.
- It’s quite normal for class diagrams to be changed once interaction diagrams are created – this is all part of the code design process!

Summary

- Both collaboration diagrams and sequence diagrams can be used to visualise how classes will work together, and how they will pass information around.
- Normally we only include one type of interaction diagram in our projects.

References

- Stevens, P 2006, *Using UML: Software Engineering with Objects and Components*, Addison Wesley

- Brambilla, M 2013, Differences between sequence diagram and collaboration diagram, StackOverflow
  - http://stackoverflow.com/questions/14319015/differences-between-sequence-diagram-and-collaboration-diagram