

# Homework #3

30+ points  
Due Wed, Feb 13, 2002

Assigned by: Ed Colbert

Grader: Ying Li

## Description

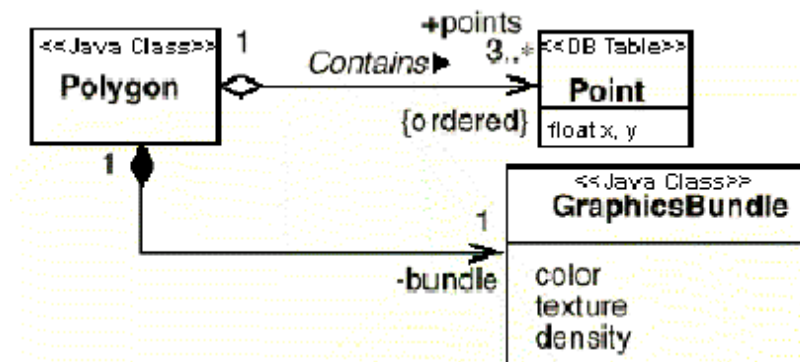
This is an individual assignment and you may not collaborate with other students except as specified in the homework assignment. You may use some of the same material as other 577b students; however, you must submit work that is significantly different from theirs.

Pick a 577b project whose LCA SSAD you are going to review. Each person on a team should pick different project. You cannot pick your own team's project, unless you did not take 577a last semester.

- 1) [20 points] Pick a *non-trivial* example diagram from each of the following sections of the SSAD and explain in detail what the diagrams communicates (with respect to implementation)
  - a. SSAD 3.4 Implementation Class Model (if has been deferred, use a diagram from 3.1.5 Logical Class Model)
  - b. SSAD 3.2 Object-Structure Model
  - c. SSAD 3.3 Interaction Model

A non-trivial class diagram would contain 5 or more classes. A non-trivial Object-Structure Diagram would contain 5 or more objects. A non-trivial Interaction Diagram would have 3 or more objects and 3 or more messages. If you cannot find a diagram in each section that meets these criteria, then use the most complex diagram available. (If no diagrams exist in a section, write a note saying that no diagrams exist. See extra credit below.)

Example:



The class diagram shown above tells the implementer to

1. Create a database table named “Point” with two floating–point valued columns named “x” and “y”.
2. Create a Java class called “GraphicsBundle” that has attribute fields “color”, “texture”, and “density” of undefined types.
3. Create a Java class named “Polygon”, that has
  - a. A public member named “point” that accesses 3 or more points from the Point DB table. The member maintains the points in some order that is not specified on the diagram.

No implementation is specified, but point member could be implemented double–scripted float array, or array of objects of a “Point” class. Either way the array would have to have at least 3 points. The Polygon’s points would have to be saved to the Point DB table at some time determined by the logic in the behavior model. The points of a Polygon could also be read from the Point table at some time determined by the logic in the behavior model.

- b. A private member named “bundle” that is an instance of the GraphicsBundle class. The composition symbol implies that Polygon class is also responsible for creating and destroying the instance of the GraphicsBundle. In Java, the bundle member could be defined at the class scope by the declaration “private GraphicsBundle bundle = new GraphicsBundle([color], [texture], [density]);”. In Java, destruction (“finalization” & clean up) is automatic.

*Note: the above example is not intended to be an example of an ideal design or a perfect interpretation of the diagram. It is just a class diagram of some design and how someone may interpret it.*

- 2) [10 points] Meet with the architect from the team whose project you chose examples from above and explain your answers to question (1). After this, have them explain to you what they intended to communicate for the examples then answer the following:
  - a. List the name of the architect that you met with along with a date and approximate time of day.
  - b. Explain in detail (for each example) where your understanding was different that the architects intentions.
  - c. What changes should the architect make to ensure clear understanding and avoid such differences?

## Extra Credit

**Missing Diagram Bonus:** If the LCA SSAD does not have any diagrams in a section, you can earn up to 7 points by creating correct diagram of the appropriate type for the system and discussing with the architect of the SSAD. (Be sure to give the architect a copy of your diagram. ;->)

**Architect's Bonus:** The architect of each team can earn extra points as follows:

- 1) List the name of reviewers with whom you met along with the date and approximate time of day.
- 2) Explaining how your design was mis-understood by the reviewers with whom you met, and what changes you will make to clarify your intent or to fix problems. For each, mis-understanding, list the reviewer or reviewers.

## Grading:

- 1) Question 1: up to 20 points
  - a. -1 point for each unexplained class, object, relation, or message.
  - b. -1/4 point for each unexplained attribute, operation, role, etc.
- 2) Question 2: up to 10 points
  - a. 2 points for meeting with the architect & including the architect's name in your report. (Note: we can check for consistency with the architect's report.)
  - b. 1/2 point for each identified inconsistency.
  - c. 1/2 point for each recommended change that the grader agrees would improve the description.
- 3) Missing Diagram Bonus: up to 7 points
  - a. 2 points for creating an appropriate diagram for an SSAD section that does not have one.
  - b. 2 points for getting the architect's agreement that the diagram accurately describes a possible design for that part of the system.
  - c. Up to 3 points based on the grader's evaluation of the correct use of UML.
- 4) Architect's Bonus:
  - a. 2 points for each reviewer that with whom you meet and whose name you include in your report. (Note: we can check for consistency with the reviewer report(s).)
  - b. 1/2 point for each recommended change that the grader agrees would improve the diagram. Maximum 8 points for this activity.