

From Domain to Design Models

Triangle World

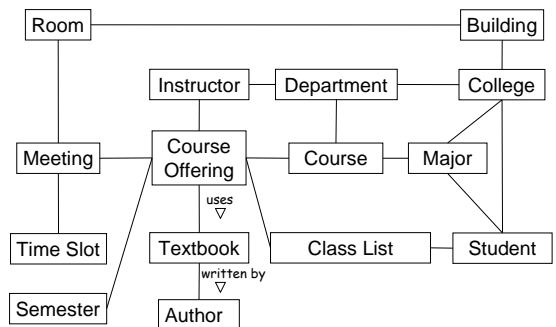
I have a sealed room in which some triangles are magically suspended in the air. Their position and orientation is random. There is also a ball in the room. It has some position and velocity at time t . (It does not intersect any triangles at time t .)

Describe the room and the ball's movement between time t and $t+\Delta t$. (In other words, write the dt -timestep use case.)

Triangle World

Build a domain model for triangle world.

Domain Model Example



Domain Model → Design Model

Assign "responsibilities" to classes

Method: CRC Cards

CRC Cards Technique (Responsibility-Driven Design)

- Informal, non-detailed
- Used for group brain-storming
- End result is a first cut at **classes** for an object-oriented model
- Not intended to provide a **complete** design

CRC

- **C:** Classes
- **R:** Responsibilities
- **C:** Collaborations

The Basic Idea

- Develop set of index cards.
- Each card represents one class.
- A card contains:
 - The name of the class.
 - The responsibilities of the class.
 - Collaborations: other classes with which this class interacts and its responsibilities in the interaction

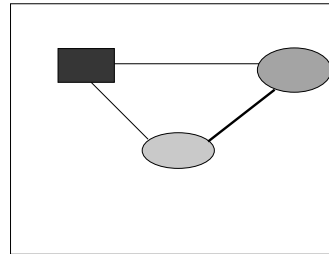
Format of CRC cards

Class Name	super class sub-classes
Responsibilities	Collaborations
_____	_____
_____	_____
_____	_____

Limiting the size of a card is an attempt at preventing the class from becoming too complex.

Sample Application: A drawing program

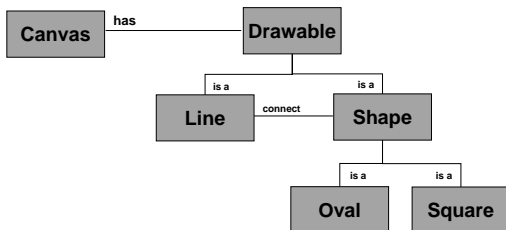
Possible screen image



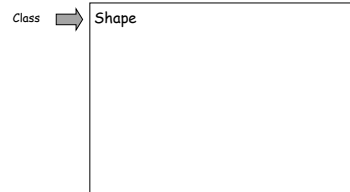
Typical Application Use-Cases:

- Draw shape
- Move shape
- Resize shape
- Connect shapes
- Erase shape
- Erase connector

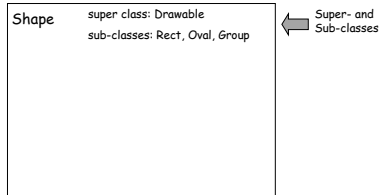
Possible Domain Model



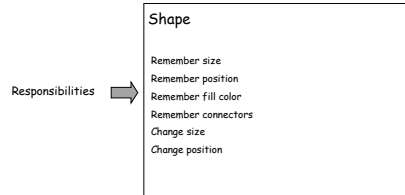
Example of CRC card for a graph-drawing program (1)



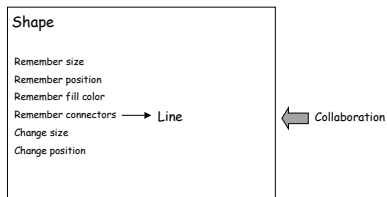
Example of CRC card for a graph-drawing program (2)



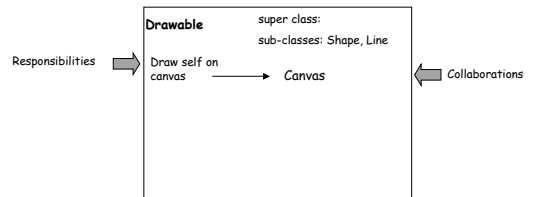
Example of CRC card for a graph-drawing program (3)



Example of CRC card for a graph-drawing program (4)

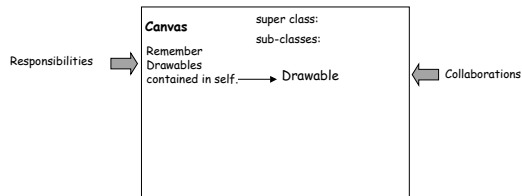


Example of CRC card for a graph-drawing program (5)



Note: The Drawable doesn't necessarily need to *remember* a Canvas, since the Canvas could be passed as an argument to the *draw* method.

Example of CRC card for a graph-drawing program (6)



Attribute vs. Collaboration

- An object of a class typically has one or more **attributes**.
- Attributes have **values** that specify or describe the object.
- A value might or might not deserve the distinction of being an object itself.
- A would-be attribute that is object-valued is actually a **collaboration**.

Triangle World

Build CRC cards for Triangle World.

Once the CRC cards are constructed ...

- Team can engage in **role-playing** to verify that use-case **scenarios** make sense for chosen CRC.
- Each person can role-play one or more class cards.
- If something doesn't work, change the class accordingly.
- Revision of use-cases might also be indicated.

Triangle World

Role play your dt-timestep use case.

Revise/redraw your class diagram.