CSC323 2010S *Software Design*

Class 07: Analysis

Held: Tuesday, February 16, 2010

Summary: We consider the basic tools of object-oriented analysis.

Related Pages:

* EBoard.

Notes:

* Surprise! It's *bad writing* day in CSC 323. The primary question: How do you learn from things that don't have a clear point?
* EC for CS Table Friday.
* EC for Rosenfield Symposium this week.
* Readings for next class: HFOOAD 5 and BC 31.

Overview:

* BC 30: When A Button Is All That Connects You To The World.
* Object-Oriented Analysis, The Head-First Approach.
* Putting The Two Together.

Dealing with Readings

Part of the growing process of being a computer scientist is learning how to deal with materials of varying quality and depth.

What do you do when assigned a reading that you find problematic? (For example, it may be obvious, it may be too dense, or it may just be poorly written.)

Snark doesn't cut it.

In every case, it is still your responsibility to do your best to extract something useful from the reading.

For most readings in CS, you should be able to find a bit of code or a bit of design to understand and critique.

For every reading, you should be able to find a main point. (It may be hidden, but we have to assume that if it's published, it has a point.)

Beautiful Code 30: When a Button Is All That Connects You to the World

- "Incidentally the Beautiful Code reading for today cites wikipedia as one of its sources. Apparently what is not so acceptable for high school history classes is acceptable for CS academia." Why might the cases be different?

- What makes the code beautiful?
- What lessons can you derive from this chapter?

- Your questions ....

OO Analysis, The Head-First Approach

- What did you see as the main points of the chapter?
- What advantages/disadvantages would you see to building a `DogInfo` object to store the list of barks?

The One-Button Problem, Revisited

- So, let's consider how we might use what we know of OO analysis so far to study the "one button" problem.