

Sean Skinner 011521843

4/25/2021

CptS 322

Homework 5

- 1) We have studied four architectural styles: data-centered, data-flow, callreturn, and layered. Now for each style, please give an example project that would be a good fit for the style. Briefly justify your choice.

Data-Centered

A good example of when to use this would be in a game for a player driven market. The central database would be where all the listings are stored, and players can search the market for items put up by other players. When an item is bought, the item is removed from the central database and given to the player and whoever originally put the item up for sale will be compensated for the items listing price. This is a good choice for this project as there are many people trying to access the same data source (the market) and the market needs to be able to communicate with every player.

Data-Flow

This would be good for a sort of search engine as it takes in data in the form of a string and needs to apply filters from the received data to display the desired result.

Call-Return

This is the type of architecture that supports the use of a controller. For example, you may want to use a model view controller in a text based application to keep track of the commands being input by the user. Then you can use a stack of those commands to keep track of an undo/redo system.

Layered

A spreadsheet application can make good use of a layered architecture. The UI of the spreadsheet would be able to request information from a spreadsheet layer which then gets information from a cell layer. This means the display will not need to worry about anything past the spreadsheet layer.

- 2) There are two measures of functional independence in software design: cohesion and coupling. Explain in your words what each of these measures means. Is it better to have higher/lower cohesion? Why? Is it better to have higher/lower coupling? Why?

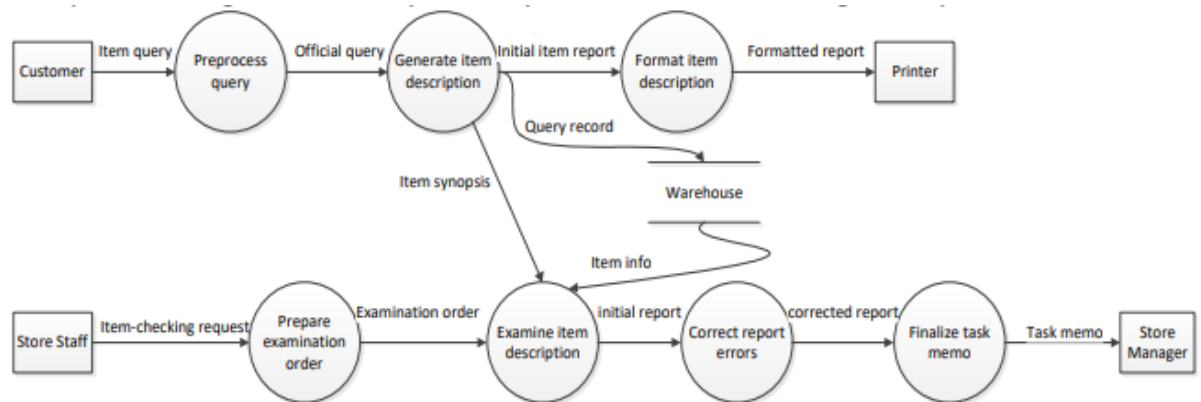
Cohesion is the degree to which elements come together to perform a task for a component. Low cohesion would mean that the component is very broad and unfocused for its functionality. High cohesion means the component is very specific and focused on its functionality.

Coupling is the level of dependency between components or in other words how many connections it has and therefore how many things a component may be dependent on. High coupling would mean a component is very interdependent on other components while low

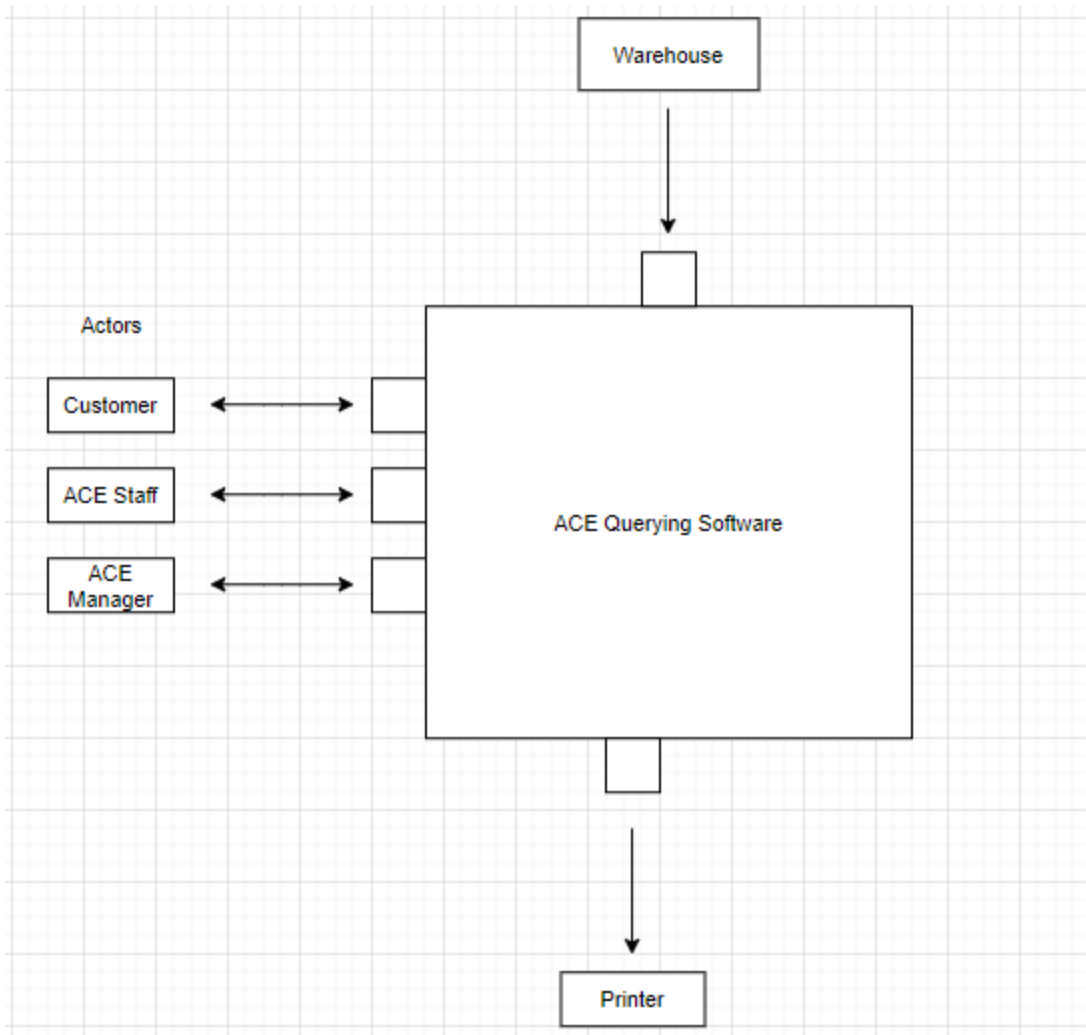
coupling would mean the component has very few interdependencies.

The most ideal set up would be to have High cohesion and low coupling, that way your classes are easy to understand and maintain, and with very interdependencies your project itself will be easier to maintain.

- 3) The following data flow diagram (DFD) is part of the flow models for the requirements of the ACE store management software. It describes the process of responding to a customer request for querying about an item. Derive the architecture design from this DFD, and draw the (brief) component diagrams that represent your architectural design.



Architecture Diagram:



Brief Component Diagrams:

