

Information Systems Analysis & Design (M8748)

Tutorial 13 Answer

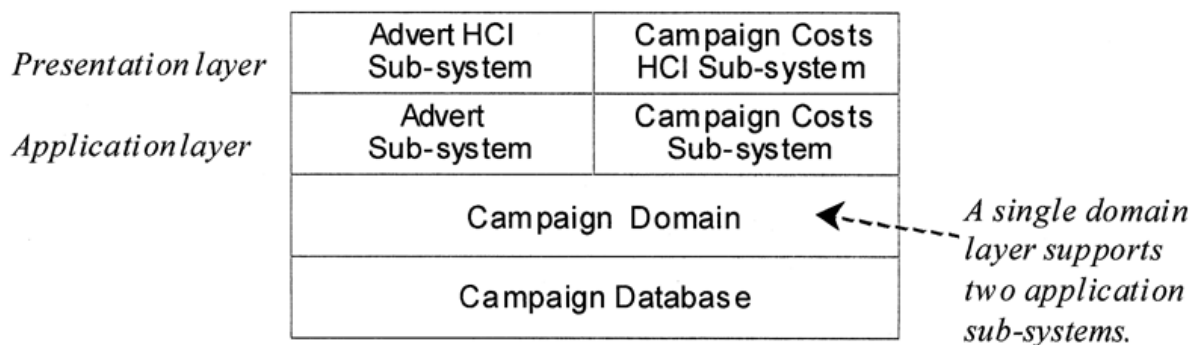
1. Why is an open layered architecture more difficult to maintain?

Open layered architectures are more difficult to maintain because each layer may communicate with all lower layers hence increasing the degree of coupling in the architecture. A change to one layer may ripple to many layers.

2. What are the disadvantages of the closed layer architecture?

A closed layer architecture may require more processing, as messages have to be passed through intervening layers.

3. What advantages would there be if the Advert HCI sub-system in the figure were designed to have direct access to the Campaign Database layer?



This would result in more compact code and perhaps improve the performance of the application. However, as mentioned above, it is likely to reduce the extensibility and maintainability of the application.

4. What are the main differences between the MVC architecture and the layered and partitioned architecture?

The main differences between the MVC and the layered architecture include the update propagation mechanism and the separation of the presentation layer into the View and Controller components in the MVC.

5. Explain how the update propagation mechanism works in the MVC architecture.

In the MVC when a change to the model data is detected by the model it informs each registered view/controller that some change has been made but does not provide any detail of the change (in the standard use of the architecture at least). It is then up to each view/controller pair to request from the model the latest version of data they require so

that they can update themselves.

6. In what sense does a broker decouple two sub-systems that need to communicate with each other? How does this work?

A broker decouples sub-systems by acting as an intermediate messaging-passing component through which all messages are passed. As a result a sub-system is aware of the broker and not directly in communication with the other sub-systems. This makes it easier to move the sub-systems to distributed computers.

7. Why is it sometimes necessary to design information systems that have explicitly concurrent behavior?

User requirements may dictate concurrent behavior that cannot be supported using facilities such as multi-tasking but requires the use of custom-built scheduler component. For example concurrent behavior may be required because:

- A use case indicates that different events require concurrent response.
- A statechart highlights the need for concurrent substates.
- An interaction diagram may show a single thread of control splitting into two concurrent threads.

8. What ways are there of simulating concurrency in the execution of a system?

Concurrency may be simulated by

- Utilizing multi-tasking capabilities in the operating system.
- Using a software development environment that supports multi-threading.
- Using a scheduler to serialize concurrent threads.
- Using a multi-processor environment.

9. How should you go about allocating system tasks to processors?

Each process should be analyzed to determine its processing requirements—see Section 13.5 for a detailed description.

10. What facilities are typically offered by a DBMS??

A DBMS provides various facilities that are useful across a wide range of different applications. These facilities include multiple views, multi-user access, security, data recovery and data access via query languages. For a full list see Section 13.6.