1.7 COM/DCOM Issues.

1).Persistence of Code.

2).Sharing.

3).Scalability.

4).Language Independence.

5).Ease of Modification.

6).Versioning.

1).Persistence of Code.

A COM component once created can be used any no. of times, thus making develop a simple task. It is used number of time i.e. Re-usability.

2).Sharing.

This is one of the most wonderful features provided by COM/DCOM. It means that we can design our component to be either In-process, Out-process or Remote-Accessed but the calling application will interact with them in the same manner. The client application is never aware the component using in an In-process or Out-process or it has been accessed remotely because of their issue, a developer can develop and distribute component in a manner that makes the most sense to the client application using them.

3).Scalability.

Scalability refers to the ability of application cop with their increasing load in term of number of user using that application and also data moving across the N/W. A critical factor for a distributed application is its ability to grow with number of users, the amount of data and the required functionality. The application should be small and fast when the demands are minimum. But, it should be able to handle additional demands without Scalability, Performance or Reliability.

4). Language Independence.

COM provides Programming Language independence because thus, component created in one language and can be used in other language.

5). Ease of Modification

Components of COM created in COM are independent of each other because of these fact component can be upgraded without affecting other part of the program or other component.

6). Versioning.

COM component are self-versioning this means that whatever some few functionality is added to an old component. The version of this component changes automatically when this component is distributed to new client they can use new functionality. As well as new functionality provided by the new component while the old client can continue using old functionality.