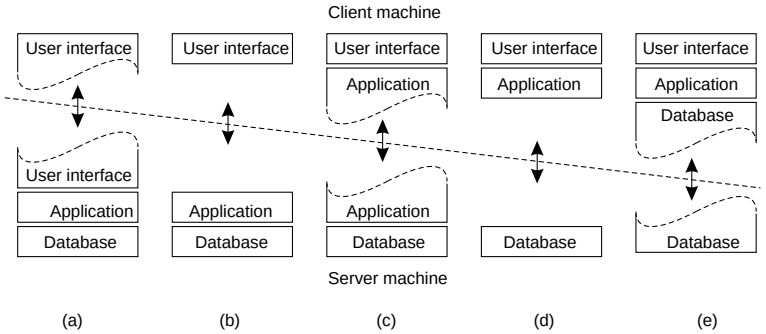
1a The three logical levels in client-server architectures can be distributed across the client and server in five different ways, as shown below. Give an example application for each of the five configurations. *10pt*



1b A server in a client-server architecture can also act as a client. What problem may we bump into when such a server is replicated? Outline a solution to this problem. *10pt*

2a Why are threads so important for developing distributed systems? *5pt*

2b What is the function of an adapter in a general-purpose object server? Give examples of some of the policies that an adapter may implement (hint: think of CORBA) *10pt*

2c Mention a number of design issues for servers. Briefly explain each issue. *10pt*

3a Explain the difference between a totally-ordered and a casually-ordered multicast. *5pt*

3b Explain how Lamport timestamps can be used to implement a totally-ordered multicast. *10pt*

3c What is the difference between sequential consistency and entry consistency? *5pt*

3d The execution of a method of an Orca object is defined as an indivisible operation. Objects may be (internally) replicated. Some of the Orca implementations use the totally-ordered communication facilities of the underlying distributed system to send operations to (possibly replicated) objects. Which data-centric consistency model fits best with Orca. Explain your answer! *10pt*

4a Explain why the remote objects as used in CORBA are harder to scale to wide-area systems than the physically distributed objects as used in Globe (and Orca). *5pt*

4b What is the advantage of separating naming objects and locating objects as is done in Globe? *5pt*

4c What is the difference between an object handle as used in Globe, and the object references as used in CORBA and Java RMI? Does Globe have something similar to CORBA object references? *5pt*

Grading: *The final grade is calculated by accumulating the scores per question (maximum: 90 points), and adding 10 bonus points. The maximum total is therefore 100 points.*