# CSCI 604 Distributed Computer Systems Architecture

# Unit 3

|  |  |  |
| --- | --- | --- |
| Teacher: | George Rudolph |  |
| Subject: | Remote Invocation & Publish-Subscribe Communication |  |
| Objective 1: | Study request-reply protocols |  |
| Objective 2: | Look at RPC and RMI under the hood |  |
| Objective 3: | Study Group & Publish-Subscribe Communication |  |
| Objective 4: | Study Message Queues & Shared Memory approaches to Communication |  |

## C Level Maximum 50 points

1. Create a flyer that describes a request-reply protocol like the one in the chapter. (10 pts)
2. List the 3 request-reply exchange protocols. Explain how they differ. (10 pts)
3. List the 3 request-reply exchange protocols. Explain what they are best used for. (10 pts)
4. TCP has overhead that UDP does not have. Why would we choose to use TCP streams then, to implement a Protocol? (10 pts)
5. Give an example of a well-known protocol that uses UDP. Give an example of one that uses TCP. (5 pts)
6. Create a flyer that briefly describes the HTTP protocol as discussed in chapter 5 of your text. Be sure to include items like timeouts and failure models. (10 pts)
7. List the HTTP methods. Give a one-or two-sentence description of what each one is for. (10 pts)
8. Explain the difference between asynchronous communication, and time uncoupling. Give an example illustrating each one. (10 pts)
9. Define time uncoupling and space uncoupling, and give examples of each one. (10 pts)
10. List 4 key issues in group Communication. For each, briefly explain (1 or 2 sentences) the problem and how it is handled. (10 pts)
11. Explain how JGroups provides reliable group communication. (10 pts)
12. List 5 application domains for publish-subscribe systems. (5 points)
13. How does publish-subscribe differ from group communication? (5 pts)
14. Create a flyer that outlines the programming model for message queues described in Figure 6.3 of your text.
15. List 3 kinds of shared memory approaches to communication. Choose one of the three and create a flyer that outlines its programming model. (10 pts)
16. Define the interface to the Election service in CORBA IDL and in Java RMI. Note that CORBA provides long type for 32-bit integers. Compare the two languages in how they specify in/out parameters to methods (10 pts)

## B Level 30 points—Choose Two

1. Suggest a design for a notification mailbox system that is intended to store notifications on behalf of multiple subscribers, allowing subscribers to specify when they require notifications to be delivered. Explain how subscribers who are not always active can make use of this service. How will the system deal with subscribers that crash while they have delivery turned on?
2. Construct a step-by-step guide that explains the operation of the alternative rendezvous-based routing algorithm shown in Figure 6.12 of your text.
3. Using Mule, find or construct and execute an example of indirect group communication. Briefly explain the structure and behavior of your example.
4. Design a remote object table that can support remote garbage collection as well as translate between local and remote references. Give an example involving several remote objects and proxies at two sites to illustrate use of the table. Show changes in the table when an invocation causes a proxy to be created. Show changes when in the table when one proxy becomes unreachable.

## A Level 20 points—Choose One

1. How would you implement the equivalent of a remote procedure call using a Tuple Space? What are the advantages and disadvantages of implementing RPC-style calls this way? Do you think Tuple Spaces are a solution looking for a problem?
2. Do you prefer message-passing or DSM for fault-tolerant systems?
3. Is CORBA better than Java RMI? Or is there something better than both?

F: < 60 D: < 70 C: < 80 B: < 90 A: >= 90