

## Middleware for Heterogeneous and Distributed Information Systems – Exercise Sheet 12

Wednesday, January 28, 2009 – 10:00 to 11:30 – Room 48-379

### Asynchronous Transaction Processing

Asynchronous transaction processing decouples request entry, request processing, and response delivery using three separate transactions as depicted in Figure 1.

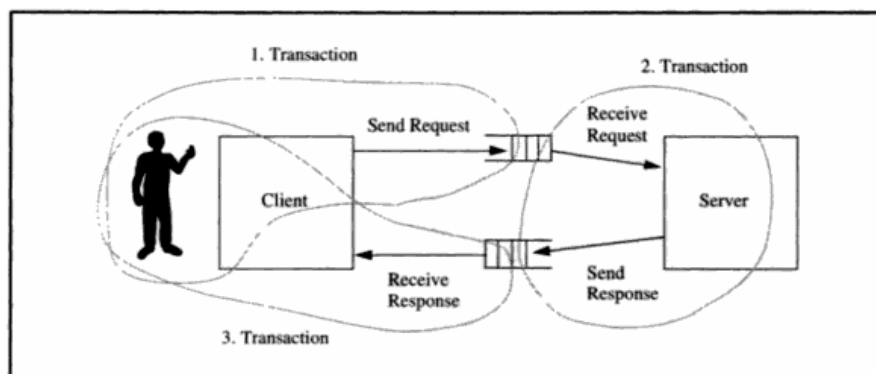


Figure 1: Asynchronous transaction processing<sup>1</sup>

Consider the following scenario: A client application attempts to order goods from a server. Therefore it assembles an order and submits it. The server processes the order, prepares a response that includes billing data and a packet tracking number, and sends it back to the client. Finally, the client processes the response and persistently stores the contained information.

1. Assume that one global transaction (not asynchronous transaction processing) is used in the above scenario. What resource managers are involved? What happens if a resource manager waits for locks to be released? What happens if a resource manager crashes? What determines the transaction response time?
2. Assume that asynchronous transaction processing is used in the above scenario. What actions are performed by the first, the second, and the third transaction? What determines the transaction response time? Do you see advantages compared to the global transaction approach?

<sup>1</sup> Gray, J., Reuter, A.: Transaction Processing: Concepts und Techniques, Morgan Kaufmann, San Mateo, California, 1993

3. How are the following failures being handled during asynchronous transaction processing?
  - a. The client crashes while attempting to send an order (transaction 1).
  - b. The server crashes in the middle of processing an order (transaction 2).
  - c. The client crashes while it writes the billing data and the packet tracking number to its local database (transaction 3).
  - d. Transaction 2 is aborted because of a deadlock situation.
  - e. Transaction 2 is aborted because consistency constraints are violated.

### **Multiple-Transactions Requests**

Assume that order processing in the above scenario involves several steps: The accounting department is asked to prepare a bill, the inventory database is updated, and a packet tracking number is queried from the shipping department. Finally, the billing data and the package tracking number are returned to the client that placed the order.

1. How could this process be realized using multi-transaction requests (also referred to as chained transactions)? In what way do multi-transaction requests generalize asynchronous transaction processing?
2. In class, stratified transactions have been discussed. In what way do stratified transactions generalize multi-transaction requests?

### **Message Routing**

To increase flexibility, message routing and message transformation logic is often separated from application logic. Reconsider the purchase order scenario above and compose a suitable message routing script!