1.264 Midterm Exam
Fall, 2005

Name: _____

Exam guidelines:

1. 80 minutes are allowed to complete the exam.
2. Open notes, open book.
3. No laptop computers or calculators are allowed.
4. No cell phones or messaging devices are allowed. Please turn off any that you have brought.
5. Short answer questions: Your answers are limited to a maximum of 2-4 sentences or phrases. Two sentences or phrases are completely adequate to answer any of these questions; 4 sentences are not necessary. Demonstrate that you understand the principles and key points. You will receive full credit for an answer if you make the principal observation(s) that the question is asking for. Details are not necessary.

1. **Software process (33 points)**

Your management has asked you to write a warehouse management system based on RFID tags and readers for high value electronic equipment. This equipment, such as automated fareboxes, passenger counters, and train control systems, is sold by manufacturers to public transportation operators. There are many models and variations of each item; large numbers of spare items are carried in inventory in your warehouse and shipped overnight to public transit operators when a unit fails. Some inventory owned by the manufacturers is also kept at public transportation operator facilities for rapid replacement of failed units. It is very difficult to keep track of all of this expensive inventory. While all inventory is supposed to be barcode scanned monthly, this is a low priority task for the staff, and many expensive units are lost or otherwise written off. Also, with poor data on what the inventory levels are, much more inventory is held than needed to support the service levels and repair times promised by the manufacturers.

RFID tags are electronic transponders that can be read automatically by an RFID reader, without human intervention. The RFID-enabled warehouse will read all tags every few minutes, 24 hours per day, to provide continuous data on inventory levels. The warehouse is run by your organization, who acts as a third party logistics provider for 20 manufacturers of these devices. Each public transit operator who is a customer of yours will be required to place all manufacturer-owned inventory in a room also equipped with an RFID reader that, again, will automatically read inventory every few minutes.

RFID technology is somewhat untested in this environment. The functional needs of the manufacturers, your company and the public transportation operators are not well understood: the inventory levels, ownership, business terms, data flows and overall use of the new system are not well understood, since it will create substantial changes from current practice.

1. You have been asked by your management to provide a short discussion of the system development/acquisition/customization approach you would take, as follows:

a. What lifecycle model would you use, and why? Also name at least two lifecycle models you would not use, and say why. (11 points)

Spiral: Requirements, design, user interface hard to define at start, must be discovered. Time and resource estimates highly uncertain. New technology, which is risky. Likely changes in business process, from barcoding to RFID. Likely new interfaces with different identifiers (primary keys) between manufacturers and you. And others.

Evolutionary prototyping: If risks are in performance of RFID tags, readers and whether the system works, starting a prototype and getting it to work at all first, and then refining it, is appropriate. The software risks may be less than the hardware/new technology risks here.

Pure waterfall is not appropriate: requirements not knowable at start, locking in design early will be wrong, changes in business process will occur, many risks exist. Perception of little progress until end would not be acceptable to management in this situation.

Modified waterfall: not ideal for the same reasons as waterfall.

Staged delivery: again, not ideal for same reasons as waterfall. Requirements and design are unlikely to be fixed at the start, or to be modified only slightly. A full spiral or evolutionary prototype are much more appropriate.

Code and fix: unacceptable in any project: No control, poor quality, no progress estimates

b. Your management asked how long it would take to prepare a first release of the system. Briefly describe how you would estimate this time. (11 points)

1. Write an initial requirements document, probably as use cases, sequence diagrams, state models and other UML documents to define the system.
2. Count function points; plan significant extra function points in software related to the RFID hardware, since it will be revised many times as the system proceeds. This is close to 'software research', which is hard to estimate. Treat RFID-related functions as complex. If off-the-shelf software is used for components, estimate the modifications needed to get the function points (table changes, user interface changes, interface changes).
3. Treat all RFID-related software as systems software; treat your team as nominal even if highly skilled, since no one has much RFID experience.

4. Estimate lines of code, schedule time, person-months and team size from tables. 5. Apply a very wide range of convergence in the first spirals or prototypes, since the product definition will not be as stable as in better-understood technologies.
6. Communicate ranges to management, not point estimates.

c. List at least four major risks in this project, and how you lessen them. (11 points)

1. Hardware not working. Prototype early to understand how well it works.
2. Changing requirements. With new process and new information previously unavailable, there will be many changes to requirements, desired reports and outputs, frequency of data transmission, etc. Spiral model to plan for evolving requirements.
3. Changing business process due to project. New actors, new reports, disputes among parties whose inventory control or costs will change. Spiral model to plan for requirements changes driven by these issues.
4. Software churn due to changing requirements. Throwaway prototype might be best approach. Throw it away once all is understood and rewrite requirements, redo the design and write/configure/modify the software and hardware from scratch.
5. Schedule visibility and control. It will be difficult to keep on schedule. Principled negotiation based on estimate convergence graph can be used to communicate and control.
6. People problems. This will be an exciting but difficult project. Insist on best staff.
7. Process problems. Many pressures will exist to derail your software process. Stay with it carefully. Don't shortchange requirements, design, QA steps.

2. Data model (34 points)

Each electronic device (e.g., farebox) in your system has a serial number (which is unique within each manufacturer but not across manufacturers), manufacturer (e.g. GE), and model number (e.g., RS4325). Each individual device has two RFID tags placed on it; in case one fails, the other one can still be read and the product's presence or absence in the warehouse can still be known. Each RFID tag has a unique serial number; this is its only attribute.

Each manufacturer has a name and address. (You may make the address a single column for simplicity in any entity in this question.) Each product has a model number, manufacturer, model name and unit cost. Each public transit agency (your customers) has a name and billing address. Each transit agency has one or more ship-to addresses. Each ship-to address has a list of the devices (models) that are to be sent to it. A device may be sent to multiple ship-to addresses for each transit agency. For example, fare boxes or passenger counters may be sent to more than one ship-to address in a transit agency.

You must draw a <u>fully normalized</u> data model that corresponds to this set of business rules. Follow these steps. You only need to turn in one drawing that includes all the elements listed in steps a-e.

a. **Draw a box for each entity: give each an appropriate name. If there are any many-to-many relationships in your model, show the intermediate/associative entity explicitly in your diagram (8 points)**

b. **List the attributes in the box for each entity (8 points)**

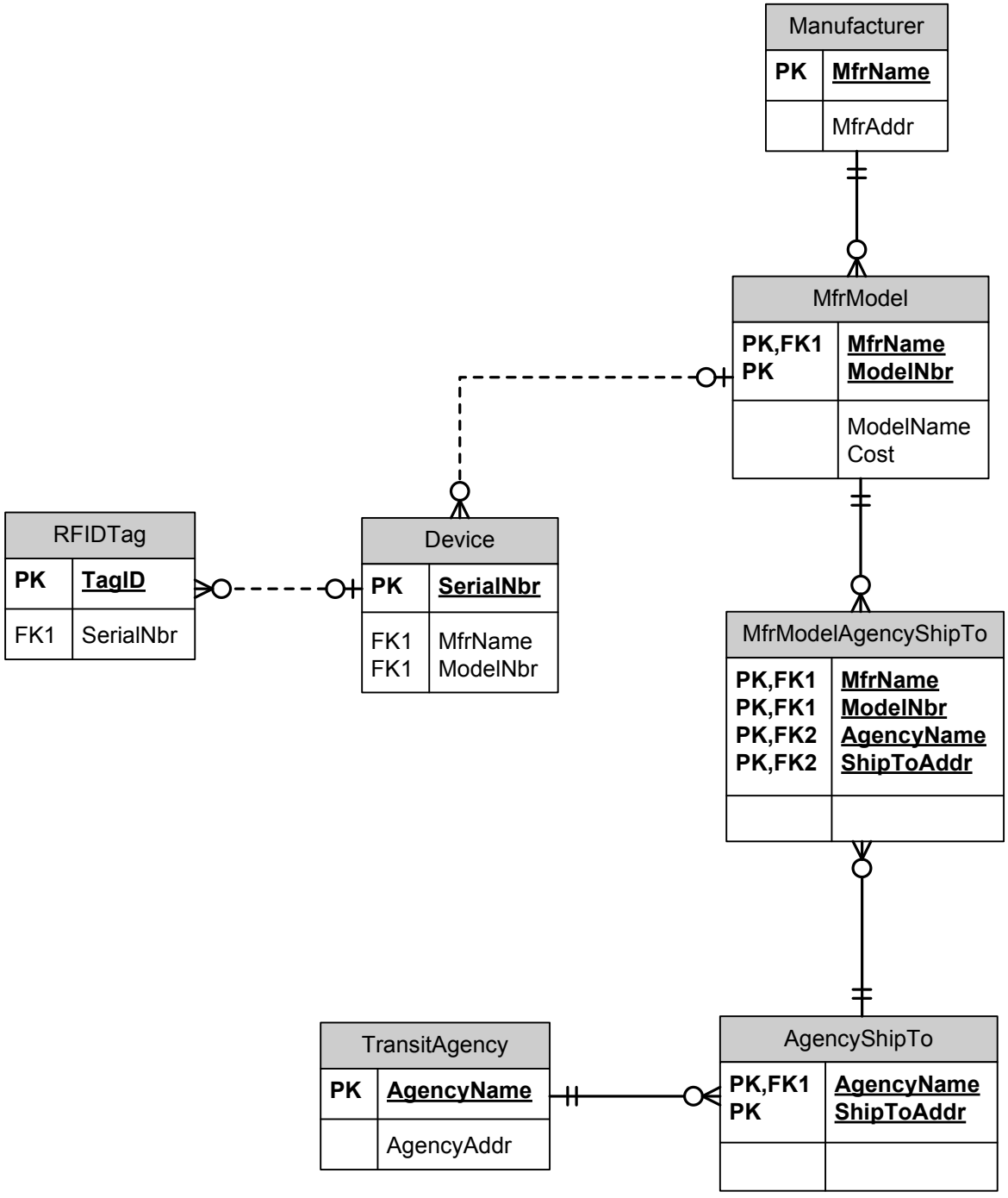
c. **Indicate the primary key for each entity by placing the phrase (PK) next to its name. (4 points)**

d. **Draw all relationships between the entities in the model. Indicate foreign keys by placing the phrase (FK) next to attributes that are foreign keys. (8 points)**

e. **Indicate the cardinality of the relationship: many-many, many-one or one-one. Use crows-foot notation; if you use another notation, define it. (7 points)**

To repeat: You only need to turn in one drawing that includes all the elements listed in steps a-e above.

Draw your data model on this page.



3. SQL (33 points)

a. List the manufacturer names and addresses, whose devices may be received at each transit agency ship-to location. (17 points)

```
SELECT MfrName, MfrAddr, AgencyName, ShipToAddr
FROM AgencyShipTo, MfrModelAgencyShipTo, MfrModel, Manufacturer
WHERE AgencyShipTo.AgencyName= MfrModelAgencyShipTo.AgencyName
  AND AgencyShipTo.ShipToAddr= MfrModelAgencyShipTo.ShipToAddr
  AND MfrModelAgencyShipTo.MfrName= MfrModel.MfrName
  AND MfrModelAgencyShipTo.ModelNbr= MfrModel.ModelNbr
  AND MfrModel.MfrName= Manufacturer.MfrName;
```

b. Increase the cost of all devices of type 'FareCollectionDevice' for manufacturer 'GE' by 25% if none of their serial numbers are less than 10000. (16 points)

```
UPDATE MfrModel
SET Cost= Cost * 1.25
WHERE MfrName= "GE"
  AND 10000 >=
  (SELECT MIN(SerialNbr) FROM Devices
    WHERE  Device.Mfr= MfrModel.Mfr
     AND Device.ModelNbr= MfrModel.ModelNbr);
```