University of Toronto

Faculty of Arts and Science
Dept of Computer Science

CSC340F – Information Systems Analysis and Design

December 2006
Instructor: Steve Easterbrook

No Aids Allowed
Duration: 2 hours
Answer all questions.

Make sure your examination booklet has 10 pages (including this one).
Write your answers in the space provided.

This examination counts for 35% of your final grade

Name: ____
(Please underline last name)

Student Number: ____

Question Marks

1 ______________/20

2 ______________/20

3 ______________/20

4 ______________/20

5 ______________/20

Total ___________/100

1. [Short Questions; 20 marks total]

(a) [Systems Theory – 5 marks] In systems theory, a system may be described as a soft system or as a hard system. Explain each type of system, and give an example of each.

Soft systems are systems that are difficult to define precisely, because the system depends on the viewpoint of the person describing it. If it is difficult or impossible to come to agreement on the boundaries of the system and its behavior, then the system is considered to be soft. All human activity systems are soft systems. For example, a banking system is a soft system.

Hard systems are well-defined and it is relatively easy to get agreement on where the boundaries to the system are, and what the purpose of the system is. The key difference between soft and hard systems is the amount of consensus that can be reached. The mechanical operation of a car is an example of a hard system.

(b) [Requirements specifications – 5 marks] Specification documents often fail to describe the requirements clearly enough for the various people who need to read them. Describe five different types of mistakes made by specification authors.

Many different answers are possible. Here are some examples:

Ambiguity – text that may be interpreted in two or more different ways

Noise – extra text that is irrelevant to the problem

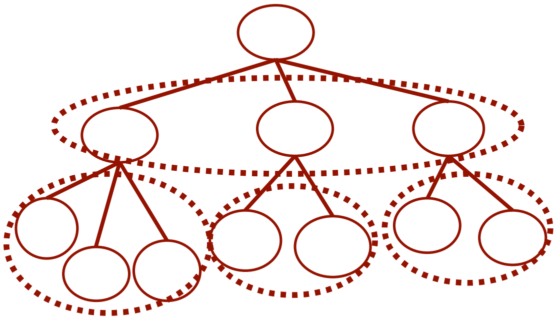
Duckspeak – text that is included only to conform to standards, but is not actually useful

Unnecessary invention of terminology – lots of new jargon invented just for this document

Wishful thinking – describing requirements for which it will be impossible to determine whether they have been satisfied.

(c) [Requirements Prioritization – 5 marks] The Analytical Hierarchy Process (AHP) simplifies the task of deciding which requirements should have higher priority, by grouping requirements into subsets, so that each requirement only has to be compared to others in the same subset. It then computes the relative weight for each requirement in the subset. Explain how the subsets are chosen, and how the relative weights are calculated.

AHP groups requirements into subsets by arranging them in a hierarchy (e.g. a goal tree), and only comparing nodes at the same level on a single branch of the tree in the tree. We can sort the nodes into priority order by first sorting the leaves of each node, then sorting the branches, etc., as shown by the dotted circles



Relative weights in each subset are calculated by asking stakeholder to compare each pair within the subset, using numbers from 1 (equal priority) to 9 (greatest difference in priority). These numbers are then entered into an nxn matrix, so each entry (x,y) has a number from 1 to 9 if x is preferred more than y, or the reciprocals of these if y is preferred more than x. Example:
The eigenvalues for the matrix are then estimated to calculate the relative weight of each row of the table. This gives an estimate of the percentage contribution of each requirement to the overall set.

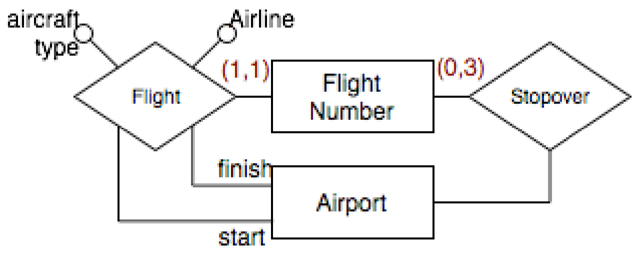
	A	B	C
A	1	9	1/3
B	1/9	1	1/5
C	3	5	1

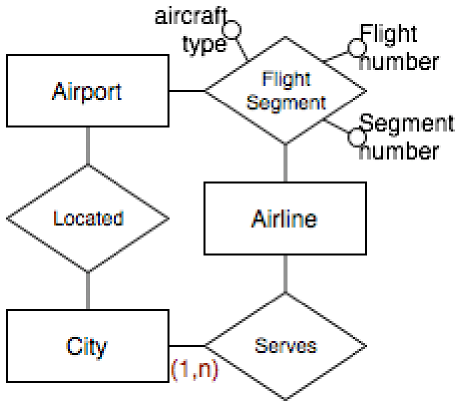
(d) [Verification and Validation – 5 marks] Explain the difference between verification and validation. Give an example of a technique that can be used for each.

Verification is the process of checking that software meets its specification (i.e. did we build the system right). One way to do this is by writing test cases that check the program behaves in the way specified

Validation is the process of checking that we understood the stakeholder's problem domain correctly (i.e. are we building the right system). One way to do this is by building models that capture our understanding of the problem, and discussing the properties of these models with the stakeholders to check that they accurately represent the problem.

2. [Entity-Relationship Diagrams – 20 marks] The following two alternative Entity-Relationship models have been proposed as the basis for a database to hold information for a flight booking system:





a. [8 marks] An instance of a relationship is a tuple of entity instances (and attribute values where applicable). Write down one example instance for each of the five relationships shown above.

Flight: AC001, Toronto (start), Tokyo (end), Air Canada(Airline) B747(aircraft type),

Stopover: AC001, Vancouver

Serves: Air Canada, Toronto

Located: Pearson, Toronto

Flight Segment: Air Canada, Pearson, AC001, 1 (segment #), B747 (aircraft type)

b. [4 marks] Give an example of a situation that is captured in the first model, but not the second, and an example of a situation that is captured in the second model but not the first.

The second model doesn't have an easy way to represent where flights start and end, except by careful use of the segment numbers – e.g. could use the convention that segment number=0 is the flight origin.

First model doesn't represent where each airport is located, so if there is more than one airport in a city, we cannot look them up as alternatives for a booking.

NOTE: other answers are possible

c. [8 marks] Add multiplicities to both diagrams to show the following constraints. If a constraint cannot be expressed using multiplicities on one or both models, say so below.

i. A flight can stop over at no more than three different airports.

Shown by (0,3) on first model, cannot be represented on second

ii. Each city must be served by at least one airline.

Shown by (1,n) on second model, cannot be represented on first

iii. Two different routes cannot have the same flight number.

Shown by (1,1) on first model, cannot be represented on second.

iv. Sometime airlines share flights, by giving different flight numbers to the same flight.

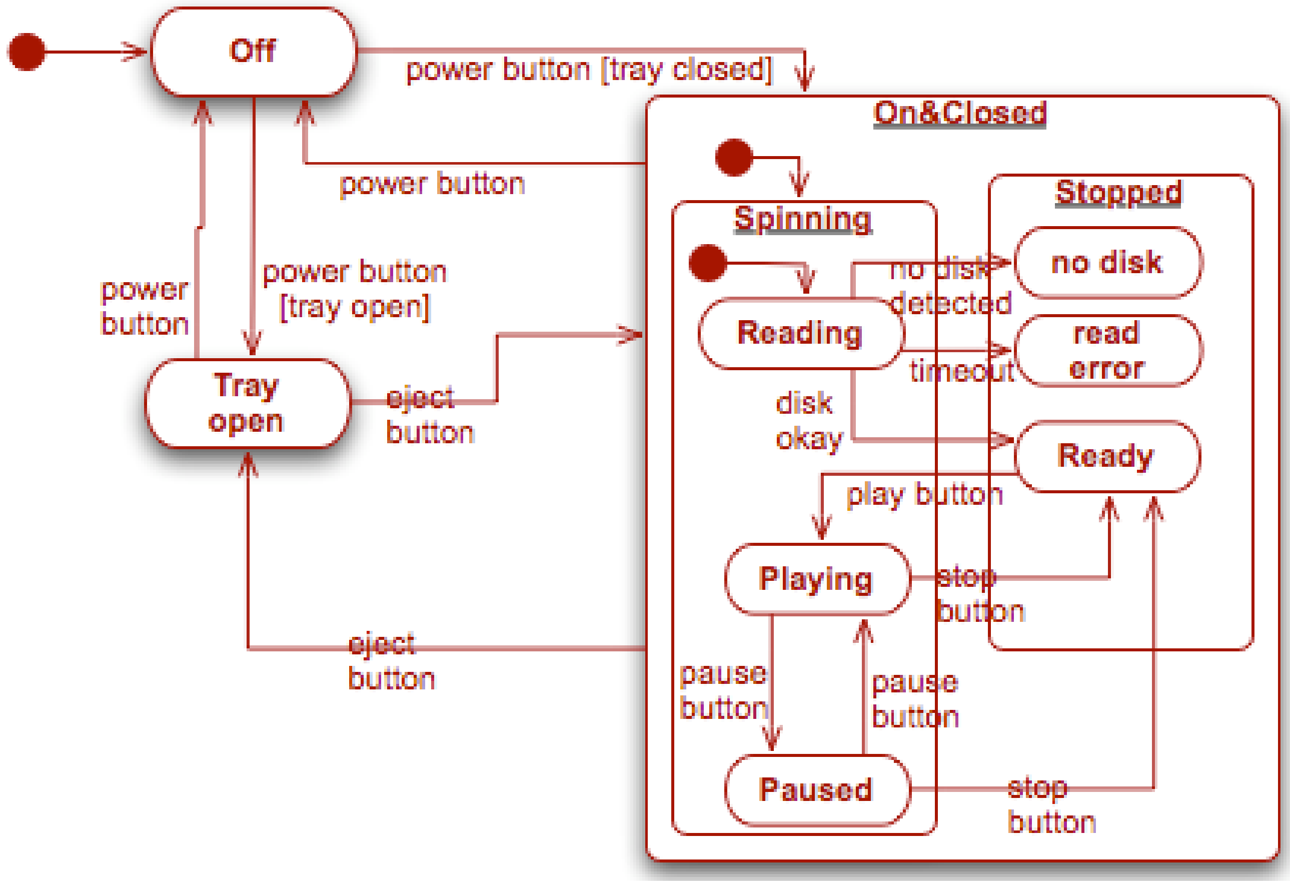
Cannot be represented on either model. Note that first model excludes this by requiring each flight to have exactly one flight number.

3. [Event Modelling – 20 marks] Draw a StateChart diagram to show the following behaviours of a DVD player. Be sure to label all transitions with the events that cause them. The DVD player is initially off. It can turned on or off by pressing the **power** button. When it is off, none of the other buttons do anything. When it is on, the tray can be opened or closed by pressing the **eject** button. When it is open, only the **power** and **eject** buttons will do anything. When the tray is closed, the motor is either spinning or stopped. When you turn on the power or close the tray, the machine first spins up the motor to read the disk. It then gets one of three signals:

(a) no disk – it stops the motor, and disables all buttons other than **eject** and **power**

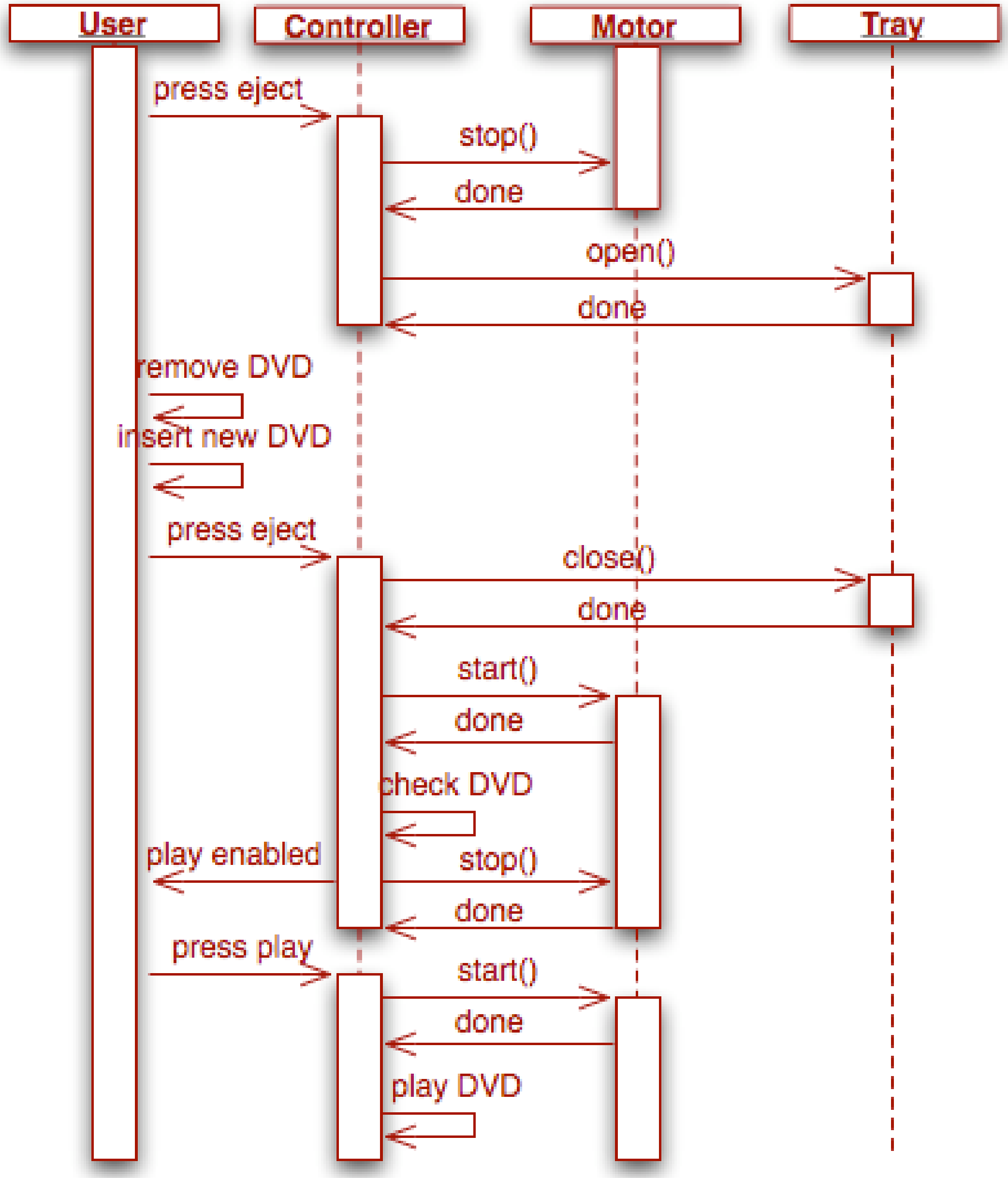
(b) timeout, indicating it was unable to read the disk – it stops the motor, displays an error message, and disables all buttons other than **eject** and **power**.

(c) disk okay – it stops the motor, but enables the play control buttons. There are three play control buttons. **Play** starts the DVD playing, spinning up the motor if necessary. **Pause** toggles between playing and paused, without stopping the motor. **Pause** does nothing if pressed when the motor is stopped. **Stop** stops the motor, if it was spinning.



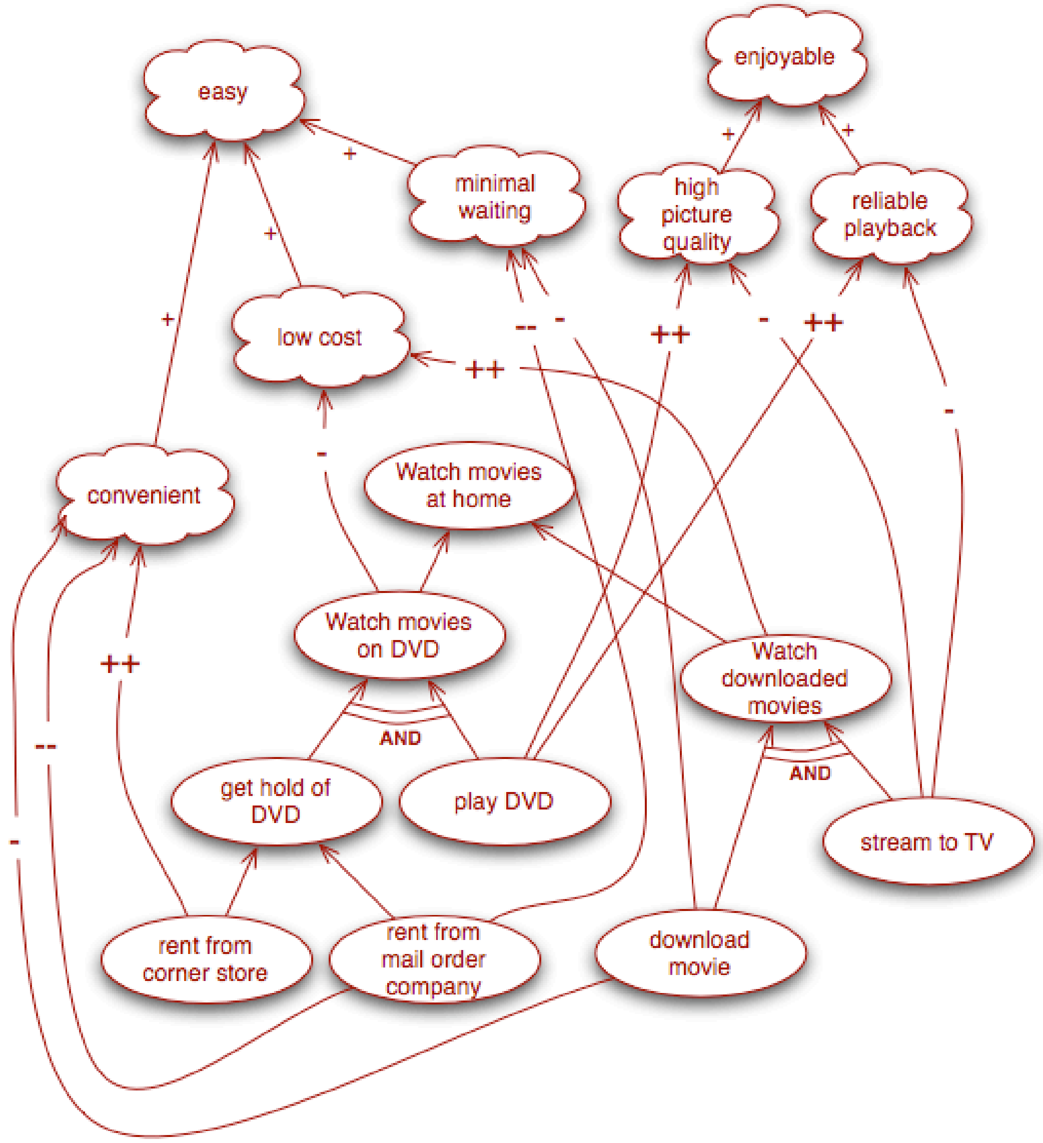
Note: this model assumes that the player can be off with the tray open. In fact most DVD players automatically close the tray before turning off. This can be modeled by adding "close" as an action to the transition from "Tray open".

4. [Sequence Diagrams – 20 marks] Draw a sequence diagram to illustrate the normal operation of the Use Case for “Change DVDs” for the DVD player described in the previous question. Your sequence should start in the ***Playing*** state, and describe the scenario in which the user ejects the current disk, inserts a new one, and selects play again. Assume the DVD player software has to send signals to the motor (to tell it when to spin) and to the tray (to open and close), and in both cases needs to await confirmation from the device before doing anything else. Use activation bars to indicate when the devices are active. State any further assumptions you make.



This answer assumes that the DVD player is unable to detect the actions of removing and inserting a disk while the tray is open, hence these actions are only relevant to the user. It also assumes that the controller is responsible for reading the disk once the motor is spinning, and will enable the play buttons at the same time as it turns off the motor, once it has successfully checked the disk.

5. [Goal Models – 20 marks]. Draw a goal model to represent the following information elicited from Fred about his preferences for watching movies at home. *Fred's main concerns are that watching movies should be easy and enjoyable. By easy, he means that it should be convenient, low cost, and involve minimal waiting. By enjoyable, he means the movie should have a high picture quality, and it should play reliably. He has experienced two ways of watching movies: DVDs and downloads. He finds downloads to be very cheap, while DVDs are not. For DVDs, he first has to rent the DVD, then play it. He can rent DVDs from his corner store, which is very convenient, or from a mail order service, which is very inconvenient, and involves a long wait. For downloads, he has to first download the movie, then stream it to his TV. Downloading is slightly inconvenient, as he has to wait a little while. He finds that playing DVDs gives him high quality picture and a reliable playback, while streaming downloads to his TV is neither high quality nor reliable.* Your goal model can use any suitable notation, but must distinguish softgoals from hardgoals, and different types of goal contribution link must be clearly labeled. State any assumptions.



Note: In this notation, clouds represent softgoals, ovals are hard goals. Goal contribution links are shown as ++ (makes the goal), + (helps the goal), - (hurts the goal), -- (breaks the goal).

(Scratch paper)

(Scratch paper)

(Scratch paper)