Proposed solution to the following exam:

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Exam in - INF5150 – Unassailable IT-systems
Day of exam: 5. December 2005
Exam hours: 14.30 – 17.30
This examination paper consists of 4 page(s).
Appendices: 0
Permitted materials: All written documents can be applied

Make sure that your copy of this examination paper is complete before answering.

NB: This exam text is only given in English since the course has been given in English this year. The candidate may, however, choose to answer in Bokmål or Nynorsk if he or she wants.

Santa’s Xmas Breaks

A company has the following business idea. They see that people need a break in their Christmas preparations and shopping efforts. They have also seen that people often fear that the nice cafes are crammed when they finally decide to take a break. They have decided to offer customers scheduled breaks in the vicinity of where they are. They agree with a number of nice cafes and restaurants to arrange such breaks, and guarantee a minimum number of participants.

The following is the context of Santa’s Xmas Breaks:

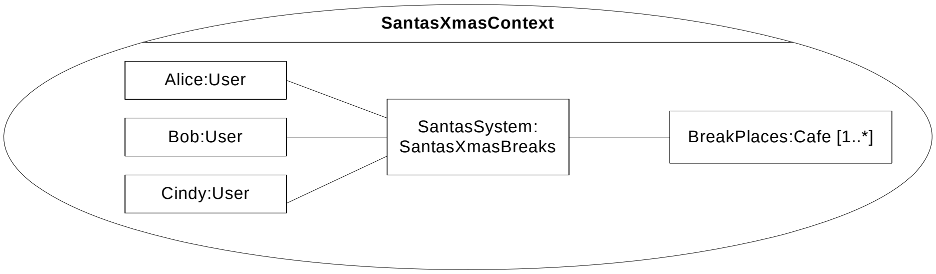


Figure 1 Santa's Context

For simplicity we have omitted the ports in the collaboration composite structure in Figure 1.

The following details the usage of the Xmas Break service.

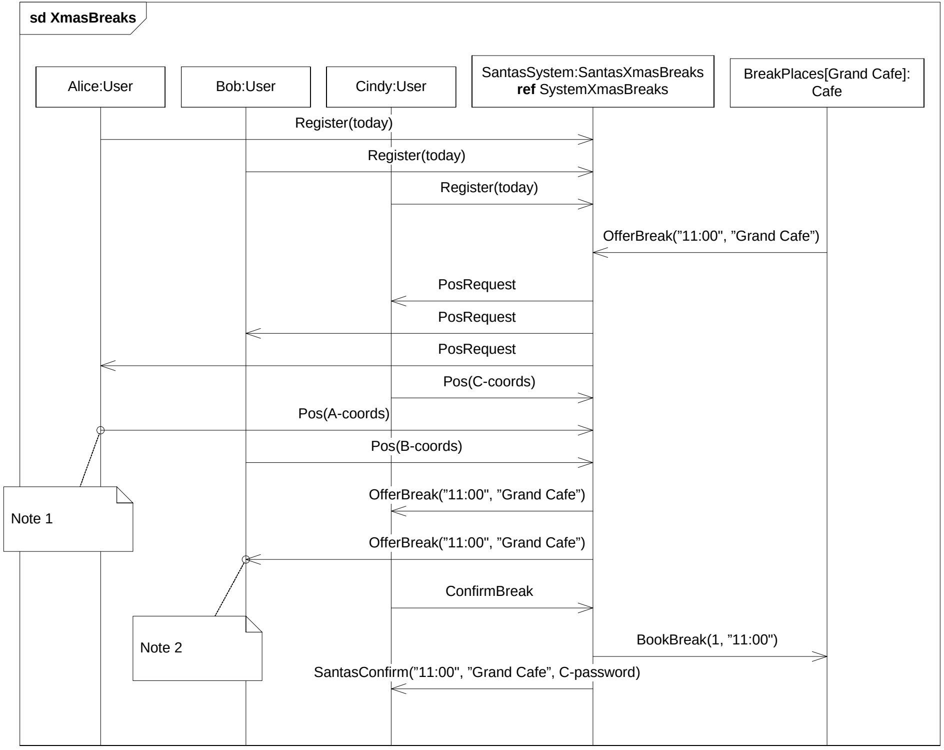


Figure 2 Xmas Breaks service

Alice, Bob and Cindy are just three examples of users of this system. In reality there may be hundreds of users.

The implementation of SantasSystem will detail the service provider class with the following composite structure.

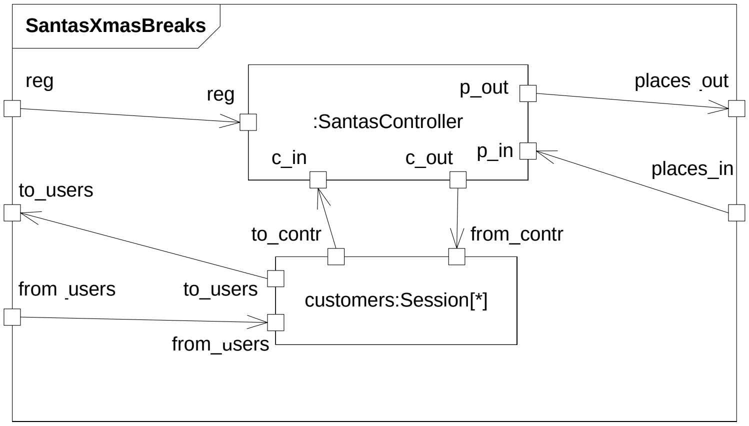


Figure 3 The composite structure of the service provider SantasXmasBreaks

Exercise 1 Modeling (35 %)

a) Give a textual explanation of what happens in Figure 2

We have the SantaSystem which is responsible for handling café reservations. The system is able to handle many users in parallel.

In figure 2, this is shown by three concurrent users. The first thing that happens is that Alice, Bob and Cindy (which are users of the system) send Registration messages to the system with the time for the wanted registration is included (today). After receiving these messages, the system receives a message from one of its cooperating cafes (GrandCafe) where the message contains the time where there are tables available and the name of the café.

When the time is right, the system sends out PosRequest messages which will provide the system with the coordinates of the different registered users.

The system gets the reply message from each of the users (this is an abstraction from the real world, happens automatically without interaction from the users and includes communication with i.e. PATS). From the sequence diagram we can see that the order of the receiving signals is different from the order of the sending of the requests - even though Bob is asked for his location prior to Alice the replies can come in different order back to the system.

After the Pos messages have been received the system sends out OfferBreak messages containing time and café name. This message is only sent to Cindy and Bob. Why Alice does not get an offer can only be guessed at this stage. Potential reasons can be: Wrong position, lower ranking in the system or something similar.

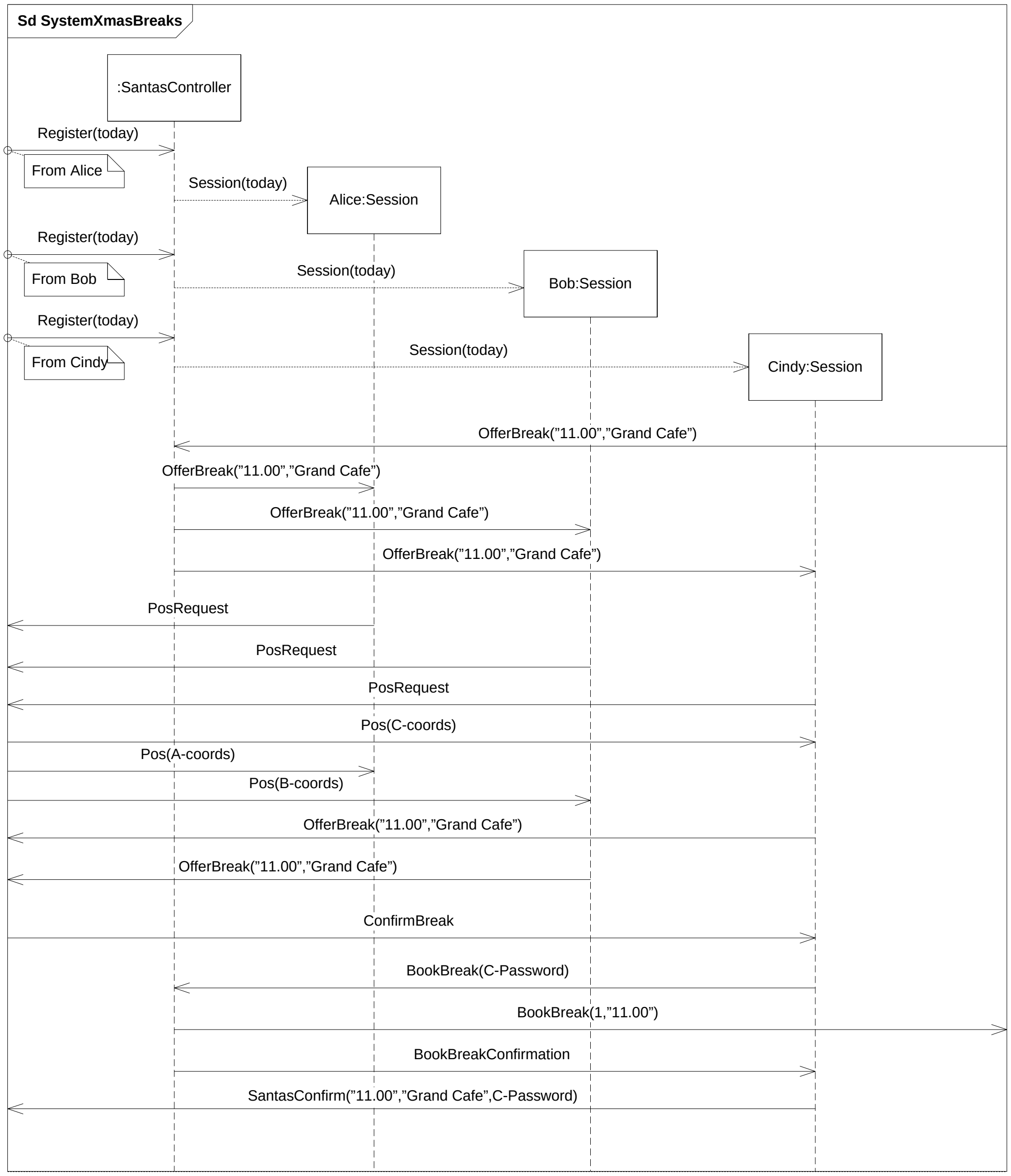
The next thing that happens is that Cindy replies with a ConfirmBreak message. Then the SantaSystem sends a BookBreak message containing number of reservations and at which time, in this case 1 reservation at 11.00.

The SantaSystem then sends a SantasConfirm to Cindy. This message includes the time, café name and a unique password.

b) In Figure 2 there are two Notes in the diagram. Give suggestions to what the designers would write in each of the two notes. The notes are supposed to give information about what makes Alice, Bob and Cindy different from each other.

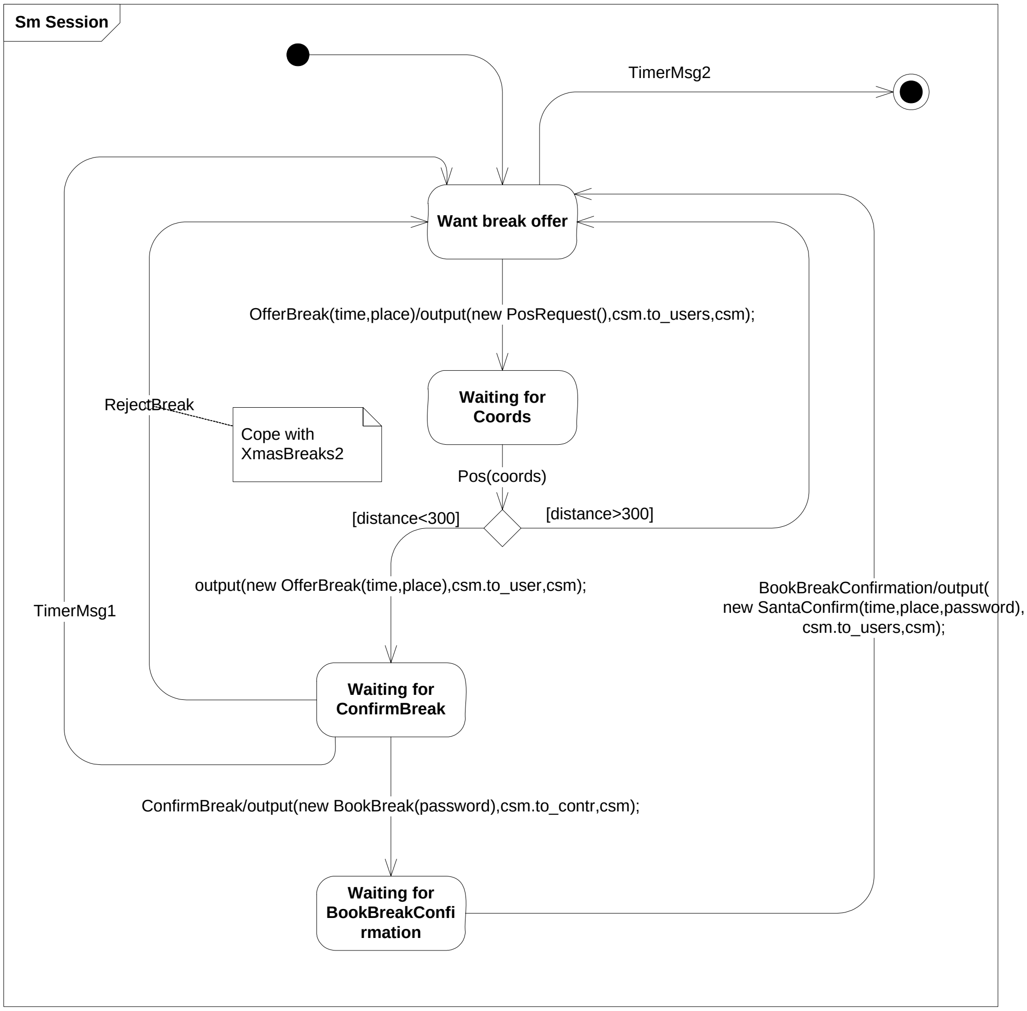
Note on Alice: Alice is not in a location that satisfies the requirements. She might be on the other side of the town at the time when she is located. Therefore she will not receive an OfferBreak message.
Note on Bob: Bob does not reply to the OfferBreak message (perhaps he has left his location, or he might not be interested in a reservation anymore). Therefore he will not be included among the users that will be expected at Grand Cafe.

c) Model the sequence diagram SystemXmasBreaks. (You find the identifier SystemXmasBreaks in the XmasBreaks diagram in Figure 2)



(The textual notation for the create messages is somewhat unclear. We will accept either using “session” or just the parameters or even using the stereotype <<create>>.

d) Model a state machine for Session (which you may find used in Figure 3). You should use the vocabulary from JavaFrame inside the transitions. Please also add explanatory comments when appropriate.


Sm Session
TimerMsg2
Want break offer
OfferBreak(time,place)/output(new PosRequest(),csm.to_users,csm);
Waiting for Coords
RejectBreak
Cope with XmasBreaks2
Pos(coords)
[distance<300]
[distance>300]
output(new OfferBreak(time,place),csm.to_user,csm);
BookBreakConfirmation/output(new SantaConfirm(time,place,password), csm.to_users,csm);
TimerMsg1
Waiting for ConfirmBreak
ConfirmBreak/output(new BookBreak(password),csm.to_contr,csm);
Waiting for BookBreakConfirmation

If the user does not reply; "TimerMsg1" will trig a transition which brings the session sm from the state "Waiting for ConfirmBreak" to the state "Want Break Offer". The session is now ready to handle a new "OfferBreak" message.

This solution also gives the registered users the ability to receive several "OfferBreak" messages during the day. The "TimerMsg2" will trig at the end of the day and the session sm will be destroyed.

Exercise 2 Verification (30 %)

a) Give explicitly one trace included in the positive traces defined by XmasBreaks in Figure 2. above.

The following is an example of a positive trace of XmasBreaks. We show the transmitter/receiver of the message by appending [tr=…]/[re=…] to the event for messages that are sent to/from more than one lifeline:

<!Register(today)[tr=Alice]
!Register(today)[tr= Cindy]
!Register(today)[tr=Bob]
!OfferBreak(“11:00”,”Grand Cafe”)[tr=Grand Café, re=SantasSystem]
?Register(today)[tr=Alice]
?Register(today)[tr=Bob]
?Register(today)[tr= Cindy]
?OfferBreak(“11:00”,”Grand Cafe”) [tr=Grand Café, re=SantasSystem]
!PosRequest[re=Cindy]
?PosRequest[re=Cindy]
!PosRequest[re=Bob]
?PosRequest[re= Bob]
!PosRequest[re=Alice]
?PosRequest[re= Alice]
!Pos(C-coords)
?Pos(C-coords)
!Pos(A-coords)
?Pos(A-coords)
!Pos(B-coords)
?Pos(B-coords)
!OfferBreak(“11:00”,”Grand Cafe”)[tr= SantasSystem, re=Cindy]
!OfferBreak(“11:00”,”Grand Cafe”)[tr= SantasSystem, re=Bob]
?OfferBreak(“11:00”,”Grand Cafe”)[tr= SantasSystem, re=Cindy]
?OfferBreak(“11:00”,”Grand Cafe”)[tr= SantasSystem, re=Bob]
!ConfirmBreak[tr=Cindy]
?ConfirmBreak[tr=Cindy]
!BookBreak(1,”11:00”)
?BookBreak(1,”11:00”)
!SantasConfirm(“11:00”,”Grand Cafe”,C-password)
?SantasConfirm(“11:00”,”Grand Cafe”,C-password)>

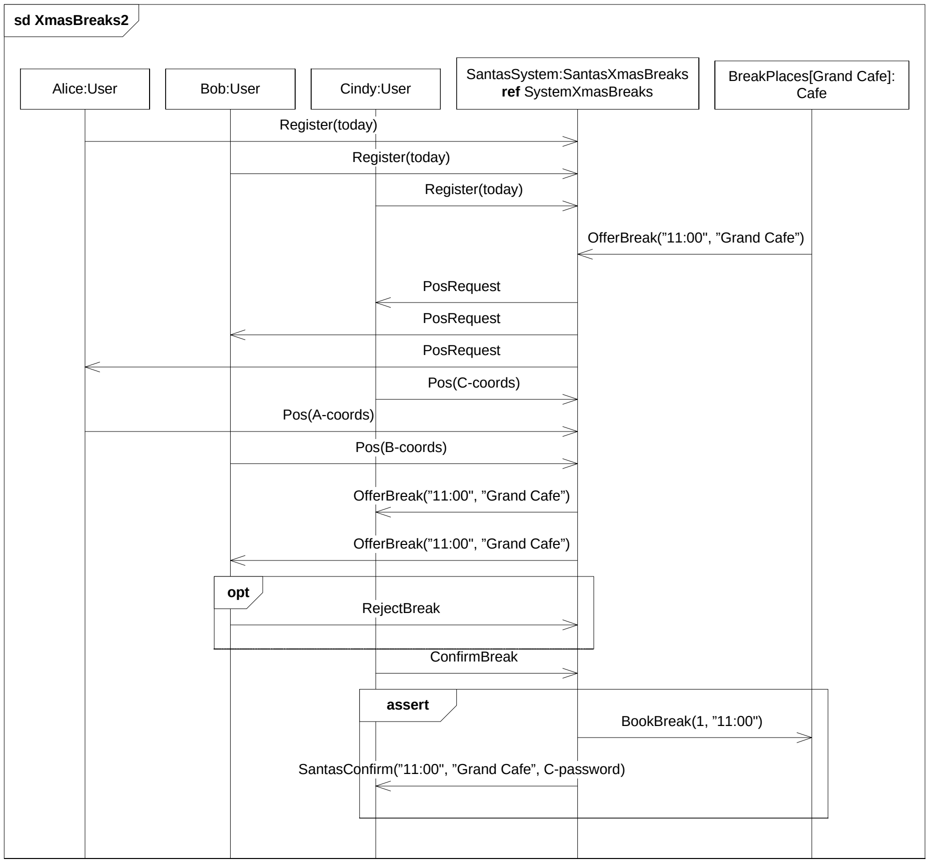


Figure 4 Modified XmasBreaks2

The definition of the **assert** combined fragment is given by the following explanation in the UML 2.0 standard:

> “The interactionOperator **assert** designates that the CombinedFragment represents an assertion. The sequences of the operand of the assertion are the only valid continuations. All other continuations result in an invalid trace.” By “invalid trace” we understand the same as we in STAIRS call a negative trace.

b) Describe one negative and one inconclusive trace of XmasBreaks2.

The following is an example of a negative trace according to XmasBreaks2:

<!Register(today)[tr=Alice]
!Register(today)[tr= Cindy]
!Register(today)[tr=Bob]
!OfferBreak(“11:00”,”Grand Cafe”)[tr=Grand Café, re=SantasSystem]
?Register(today)[tr=Alice]
?Register(today)[tr=Bob]
?Register(today)[tr= Cindy]
?OfferBreak(“11:00”,”Grand Cafe”) [tr=Grand Café, re=SantasSystem]

!PosRequest[re=Cindy]
?PosRequest[re=Cindy]
!PosRequest[re=Bob]
?PosRequest[re= Bob]
!PosRequest[re=Alice]
?PosRequest[re= Alice]
!Pos(C-coords)
?Pos(C-coords)
!Pos(A-coords)
?Pos(A-coords)
!Pos(B-coords)
?Pos(B-coords)
!OfferBreak(“11:00”,”Grand Cafe”)[tr= SantasSystem, re=Cindy]
!OfferBreak(“11:00”,”Grand Cafe”)[tr= SantasSystem, re=Bob]
?OfferBreak(“11:00”,”Grand Cafe”)[tr= SantasSystem, re=Cindy]
?OfferBreak(“11:00”,”Grand Cafe”)[tr= SantasSystem, re=Bob]
!ConfirmBreak[tr=Cindy]
?ConfirmBreak[tr=Cindy]
!SantasConfirm(“11:00”,”Grand Cafe”,C-password)
?SantasConfirm(“11:00”,”Grand Cafe”,C-password)>

This trace is identical with the one given in a), except that the BookBreak(1,”11:00”) message has been removed. This is not allowed due to the *assert* ***operator.***

The following is an inconclusive trace according to XmasBreaks2:

<!PosRequest[re=Alice]
?PosRequest[re= Alice]>

This trace describes the situation where SantasSystem asks for Alice’s position even if she has not registered for today. This situation is not described (either as positive or negative) in XmasBreaks2.

c) XmasBreaks2 (Figure 4) is a refinement of XmasBreaks (Figure 2). Is it a supplementing, a narrowing or a combination? Motivate your answer.

It is a supplementing. We note that neither sequence diagram contain an xalt operator, so both specifications include only one interaction obligation. Therefore, it is sufficient to reason at the level of interaction obligations (and not sets of interaction obligations.)

Only two changes have been made in XmasBreaks2. Firstly, a new message RejectBreak has been inserted and enclosed by an opt operator. The traces we get from XmasBreaks2 that include the RejectBreak message are inconclusive in

XmasBreaks but are not inconclusive in XmasBreaks2. In addition, there are no traces that are inconclusive according to XmasBreaks2 but not according to XmasBreaks. Therefore XmasBreaks2 is a supplementing of XmasBreaks.

The definition of narrowing requires that at least one trace has been moved from the set of positive traces to the set of negative traces. The addition of the assert operator in XmasBreaks2 means that new negative traces have been added, but all traces that are negative according to XmasBreaks2 are inconclusive according to XmasBreaks. Therefore XmasBreaks2 is not a narrowing of XmasBreaks.

d) Is XmasBreaks a refinement of XmasBreaks2?

No, since there are traces that are inconclusive in XmasBreaks, but not in XmasBreaks2.

e) Propose a change to XmasBreaks2 so that XmasBreaks2 is no longer a refinement of XmasBreaks.

This can be achieved for example by removing the OfferBreak("11:00","Grand Cafe") message that goes from BreakPlaces[GrandCafe] to SantasSystem in XmasBreaks2. This would mean that for example the trace given in a) as positive according to XmasBreaks would be inconclusive according to XmasBreaks2.

Comment: This changes the semantics of the diagram considerably.

One could also remove the opt-frame around the RejectBreak. Then the positive traces of XmasBreaks would be inconclusive in XmasBreaks2.

f) Explain in words (or by fragments of diagrams) how you would modify your Session state machine to cope with XmasBreaks2.

This has been handled in the Sm Session diagram by adding the transition with trigger RejectBreak from the Waiting for ConfirmBreak state. In this way we are also able to produce the traces that include the RejectBreak message.

(Note that the addition of this transition is not strictly necessary, since the opt ***operator does not mean that the implementation must be able to produce the relevant traces. The addition of the assert operator requires no change to the Session state machine.)***

Exercise 3: Security Analysis (35%)

a) What is the purpose of asset identification from the perspective of the risk analysis client?

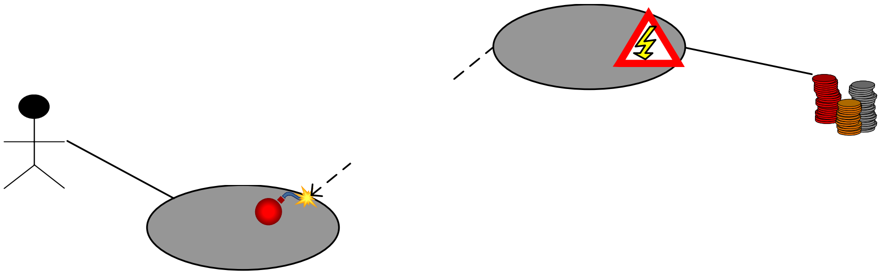
The purpose is to identify the parts and features of the target that have value for the risk analysis client and that we want to protect. Assets are identified early in the analysis process and are the basis for the rest of the analysis.

b) Identify four assets of the Xmas Breaks provider with respect to the Xmas Breaks service.

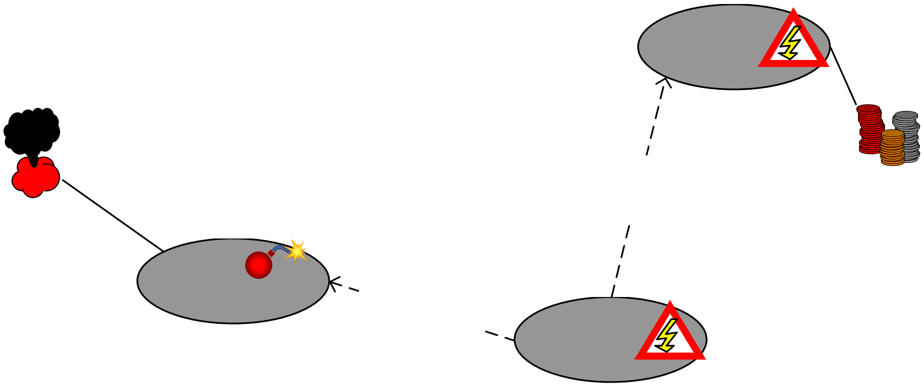
Asset1: SantasSystem's reputation among the customers
Asset2: Break place agreements
Asset3: Computer equipment where SantasSystem is deployed
Asset4: Income

c) Use XmasBreaks2 to identify four different threat scenarios – one for each asset identified under b). Specify the threat scenarios using the CORAS language.

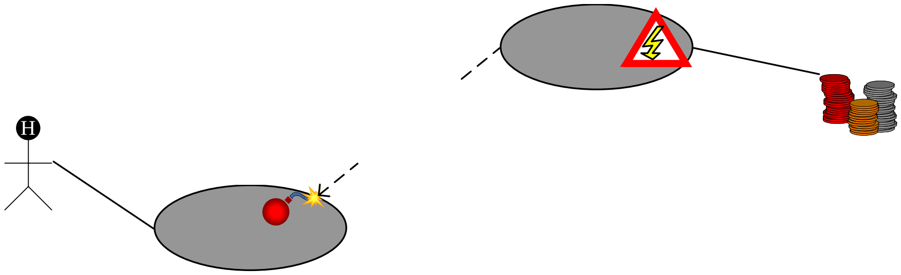
<u>Scenario 1</u> (related to asset1): An unfaithful employee gains access to the SantasSystem and sends false OfferBreak messages to registered customers. The customer arrives at the assumed break place, but no table available.



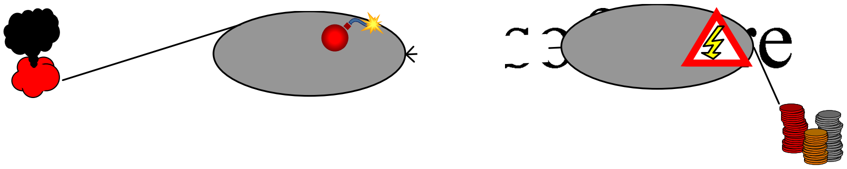
<u>Scenario 2</u> (related to asset2): The software is of poor quality. A BookBreak(n,time) where n is too large is sent to a break place. The break place reserves tables for too many customers of SantasSystem, and therefore rejects other potential customers. The break place owner therefore cancels the agreement.



***Scenario 3** (related to asset3): The computer equipment is placed in an office with poor security. A thief breaks in and steals the equipment.*



***Scenario 4** (related to asset4): The sms communication service is provided by a research laboratory and the communication often fails. Potential registrations are lost, which means that income is lost.*



d) Explain whether the identified threat scenarios compromise confidentiality, integrity, availability or neither.

Scenario 1 compromises integrity since the information sent to the customers is incorrect.

Scenario 2 also compromises integrity since the information sent to the break place is incorrect.

Scenario 3 compromises availability since the system will go down. (It may also be seen as compromising confidentiality since the thief may gain access to information about the registered customers. However, we have not identified customer registrations as an asset).

Scenario 4 compromises availability, since the service will not be available if the sms communication does not work.