Software Architecture and Engineering 2017
Project 1 (Part 2: Alloy Modeling Task)

Published: March 16, 2017

Firm deadline for submission of Alloy model: April 02, 2017 (23:59 CET)

Your task is to design a flight booking system based on the description provided in the earlier UML modeling task. The emphasis of your task is on designing the data model, formalizing it in Alloy, and checking whether certain properties hold for your Alloy model.

You should use the UML model from Fig. 1 to guide the design of the Alloy model (i.e., *do not use the UML model you submitted*).

We intentionally made parts of the system description (provided as part of the UML modeling task) ambiguous. It is your responsibility to clarify any ambiguities with the assistants (your "customers"). If you have questions for the customer, send an e-mail to sae-assistants@lists.inf.ethz.ch.

You should base your Alloy model on the (incomplete) Alloy file that we provide. In particular, implement all the provided functions. *Do not change the names of the signatures and do not change the signatures of the functions.*

1   Static Model

Task A. (25 points)   Create an Alloy model of all the data structures representing entities of the flight booking system. Include all the relevant details, relations and facts. Ask your TA in order to clarify any ambiguities. In addition, document any detail that cannot be encoded in the Alloy model.

Make sure to implement all of the provided functions correctly and do not change their names or signatures (the functions for the dynamic part can be omitted for this task). These functions will be used for some automated tests and will significantly affect the number of points obtained for this task.

Task B. (15 points)   Generate the following instances using Alloy, showing that your model is not overly restricted, or explain why they are infeasible if you cannot generate them:

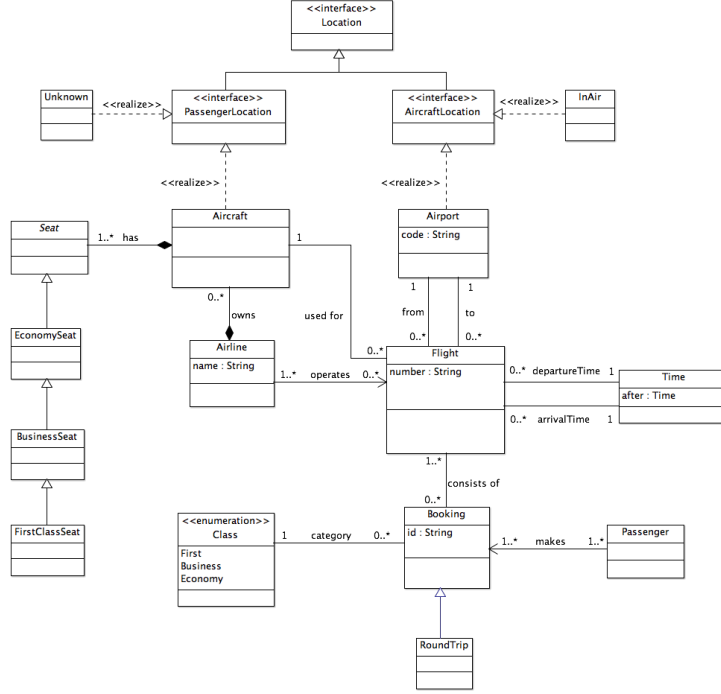1. There is one flight, one aircraft, one airline, one passenger, one seat, and two airports.

Figure 1: UML model

2. There are three bookings with different classes but there are only two seats of the same class (with two passengers, two flights, two airports, and one airline).

3. A round trip has three booked flights (with one passenger, one seat, two airports, and one airline).

4. Two bookings share a flight but have different start and end points and all flights are operated by the same plane (two passengers, two seats, and one airline).

5. A passenger has a booking for three flights, of which the first and the third are on the same aircraft, but the second is not (with one passenger, two aircrafts, two seats, and one airline).

For each instance, create a predicate called `static_instance_n`, with n replaced by the number of the instance, which generates the instance when run with the `run` command. Export all feasible models as images.

2 Dynamic Model

Task C. (21 points) In this task, you should extend your static model from Task A to create a model of the dynamic behavior of the system. More specifically, you should model how the location of a passenger and an aircraft changes over time. For this, you should model time explicitly (see Fig. 1) and model what happens to the locations of passengers and aircrafts when the system transitions from one moment in time to the next one. The location of an aircraft can be either at a given airport or "in the air", while the location of a passenger can be either "unknown" or in an aircraft. It is recommended to complete Tasks A and B first, and then start this task by extending the Alloy model from Task A.

Additionally, you should implement the functions and predicates declared in the supplied Alloy file for the dynamic model, and define the following signatures in your model:

- `State`

- `AircraftLocation`, containing the value `InAir`

- `PassengerLocation`, containing the value `Unknown`

Task D. (9 points) Generate the following instances using Alloy, showing that your model is not overly restricted, or explain why they are infeasible if you cannot generate them:

1. A passenger is on different planes at different points in time (with three flights, one passenger, one roundtrip booking, two airports).

2. There is at exactly one booking for all flights, and at some point an aircraft is in the air while a passenger is in an unknown location (with 2 flights).

3. There are exactly two passengers, and they are in the same location at each point in time (i.e. when the first is on a plane, the other is on the same plane; when the first is in an unknown location, so is the other). One of them has booked a roundtrip and no other bookings, the other hasn't booked a roundtrip (with three bookings, two aircraft, two airports, two passengers)

For each instance, create a predicate called `dynamic_instance_n`, with n replaced by the number of the instance, which generates the instance when run with the `run` command. Export all feasible models as images (projected over time, with one image for each current time).

To generate these models, you may restrict your system only as defined above, e.g. for the first model, your predicate may restrict only the number of flights, passengers, roundtrips, airports, as well as the locations of the passenger at two different times. You are also allowed to restrict the number of elements in all sets when running the corresponding predicates, to keep the solving time within reasonable bounds.

3   Deliverables

Submit your solution by email to sae-assistants@lists.inf.ethz.ch, including:

1. An Alloy file with all the required code from Tasks A–D. The file must include short comments that explain your formalization.

2. An XML file for each generatable instance from Tasks B and D.

3. A PDF file that includes:

   - The Alloy model from Task A
   - The Alloy model from Task C
   - A list of instances from Task B that you were not able to produce, each with a short explanation why the instance is not feasible
   - Image exports of all the generated instances from Task B
   - A list of instances from Task D that you were not able to produce, each with a short explanation why the instance is not feasible
   - Image exports of all the generated instances from Task D, projected over time, for all times.

4   Resources

- Alloy: http://alloy.mit.edu/alloy/
- PDF creation: You can use Microsoft Word or LaTeX to create your document and **export it to PDF**