“Airport” Modeling Example

- We will model an airport expressing relationships between
  - Vehicles
    - Airplanes, Service vehicles (e.g., fuel trucks, tugs, food trucks)
  - Locations
    - Gates, Locations, Taxiways, ...
- Focus on how to extract the model from a textual description
CIS 771 -- Airport Model

For you to do (pause here)

- Read the description of the airport problem
- Don’t look ahead at the model (yet)
- Think about how you would construct the model
  - Domains?
  - Sets?
  - Relations?
  - Constraints?

CIS 771 -- Airport Model

For the purposes of this problem we can consider an airport to have two basic kinds of resources: fixed resources (special locations) and mobile resources. Fixed resources include gates, runways, and taxi-ways. Airports have mobile resources like planes and service vehicles (trucks for delivering fuel/food/etc.).
Identify Sets

For the purposes of this problem we can consider an airport to have two basic kinds of resources: fixed resources (special locations) and mobile resources. Fixed resources include gates, runways, and taxi-ways. Airports have mobile resources like planes and service vehicles (trucks for delivering fuel/food/etc.).

Identify Relations

It is often useful to distinguish between modeling of the "environment" or physical conditions in which a system will execute and modeling of the intended behavioral policies of the system. In this problem there is a clear distinction. For example, we clearly should model the physical condition that mobile resources can only be one place at a time. On the other hand, we would like "ground traffic" to behave according to certain guidelines or policies which our ground traffic control system will work to enforce. For example, one such policy is that only planes (and not service vehicles) can be on runways. In any case the model must encode both types of
**Basic Model**

```plaintext
model GroundControl {
    domain {Equipment, Location}

    state {
        partition Plane, Vehicle : Equipment
        partition Runway, Taxiway, Gate : Location
        equipment (~location) : Location -> Equipment
    }
}
```

Fixed resources are termed locations

Fixed-mobile resource association

---

**For you to do (pause here)**

- Load airport-1.all
- Generate some instances
- Do they look reasonable?
- Type in some conditions to probe the possible instances that can be generated.
Identify Constraints

Here are some physical environment constraints and policy constraints on resources that you must incorporate into your model:

1. Resources like planes and vehicles can be at one place at a time.
2. Only planes can be on runways.
3. Gates can have a single plane.
4. Gates can have multiple service vehicles.
5. If a plane is at a gate, then a vehicle should be there to service it.

Encode Constraints

1. Resources like planes and vehicles can be at one place at a time.

Declaration option? Multiplicity? Invariant?

Resources like planes and vehicles = Equipment
Place = Location

Equipment can be at one Location at a time

For each piece of Equipment there is exactly one Location that it occupies

equipment : Location! -> Equipment
**Identify Constraints**

2. Only planes can be on runways.

   Equipment and Location are generic and we cannot constrain selected subsets of the equipment relation using multiplicities

   \[
   \text{all } r : \text{Runway} \mid r.\text{equipment} \in \text{Plane}
   \]

**Identify Constraints**

3. Gates can have a single plane.

   Constrain the equipment relation for a particular class of Locations

   \[
   \text{all } g : \text{Gate} \mid \text{sole} (g.\text{equipment} \& \text{Plane})
   \]
Identify Constraints

4. Gates can have multiple service vehicles.

Equipment is partitioned into Planes and Vehicles

3. Constrained the number of planes at a Gate

How many Vehicles are allowed by the declaration of equipment?

Identify Constraints

5. If a plane is at a gate, then a vehicle should be there to service it.

This constraint is conditional so we use an implication

Exploit partition to express Vehicle by differencing Plane

\[
\text{all } g : \text{Gate} \mid \\
\quad \text{some } (g.\text{equipment } \& \text{ Plane}) \rightarrow \\
\quad \text{some } (g.\text{equipment } - \text{ Plane})
\]
For you to do (pause here)

- Load airport-2.all
- Generate some instances
- Do they look reasonable?
- Type in some conditions to probe the possible instances that can be generated.
- Do we need to state that every piece of equipment is at a location?

Airport “Realism” Constraints

- Some of each of the sets

```cond Realism {
    some Gate && some Runway &&
    some Taxiway && some Plane &&
    some Vehicle
}
```
Conditions as Queries

- Can the model generate an instance that satisfies property $P$?

5) ... Can service vehicles be present at a gate when a plane is not? ...

```plaintext
some v : Vehicle |
  v.location in Gate &&
  no (v.location.equipment & Plane)
```

For you to do (pause here)

- Load airport-3.all
- Generate some instances of the constraints
- Does the last condition answer the question in part 5 of the airport description?
- How would you go about adding to the model to check that all 5 constraints were satisfied?
Airport Assertions

- For each constraint define an assertion
  - Assertions should formulate the constraint in a different way than the constraint is stated if possible
  - e.g., for a positive constraint try formulating the negative version for an assertion
- This adds redundancy to the specification
  - ACA can check redundant specifications against each other to see if they are equivalent
  - This can be very helpful in finding errors in either the model or your statement of the redundant specification

For you to do

- Load airport-4.all
- Check all the constraints
- Do you believe that there are no counterexamples in larger scopes? If not, increase the scope and recheck.
- Can you think of ways to reformulate the redundant assertions?