

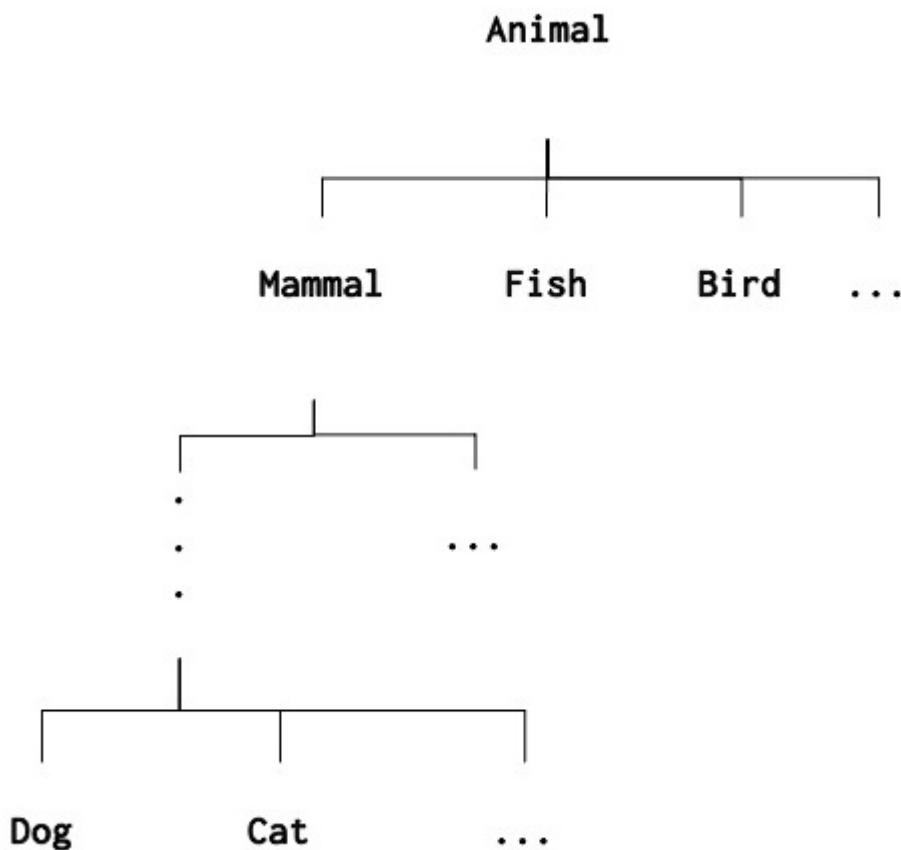
## 8.6 Representing classification hierarchies

*Classification* is a fundamental means for understanding and organizing knowledge about the real world, and is used in many disciplines like biology, physics, health, etc. Classification is a means for organising concepts in a hierarchy called a *classification hierarchy*.

Class `Account` and the subclasses `SavingsAccount` and `CreditAccount` and class `Booking` and its subclasses `TravelBooking`, `FlightBooking`, `TrainBooking` and `HotelBooking` are examples representing classification hierarchies.



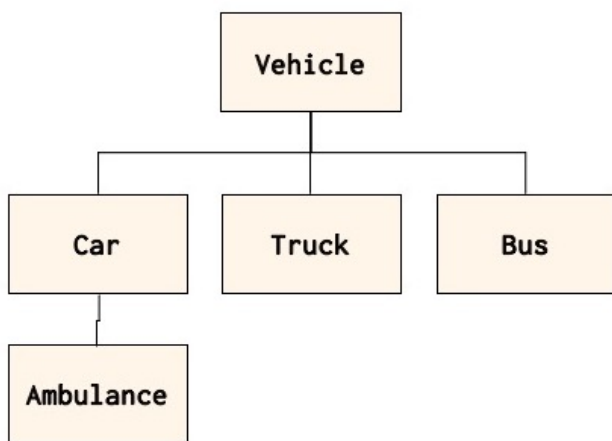
We use classification in our everyday language when we deal with various types of concepts. We use general terms like animal, and more specific terms like mammal, fish, bird, and insects. We may use terms like dog and cat to be more specific than just mammal. Even though dogs like golden retriever, poodle, and greyhound are breeds of dogs, one may also put these into such a classification. The figure to the right shows a partial animal classification hierarchy.



The terms *generalization* and *specialization* are often associated with classification hierarchies.

A *generalization* is a more general concept that captures the commonalities of and suppresses some of the differences of a number of more special concepts. A *specialization* is the inverse of a generalization. In the above figure, Account is a generalization of SavingsAccount and CreditAccount, and these are specializations of Account. Booking is a generalisation of TravelBooking, and HotelBooking, and FlightBooking, and TrainBooking are specializations of TravelBooking, etc.

The vehicle hierarchy is another example of a classification hierarchy. The need for being able to represent such a hierarchy of vehicles lead the designers of SIMULA, Ole-Johan Dahl and Kristen Nygaard to invent the subclass mechanism of SIMULA.



In section , we introduced the notions of extension, intension and designation of a concept. For a concept and its specializations, the following properties hold:

1. The extension of a specialized concept is a *subset* of the extension of the corresponding generalized concept.
2. The intension of a specialized concept *extends* the intension of the corresponding generalized concept by adding additional properties

With respect to (1), all phenomena covered by a specialised concept are also phenomena belonging to the more general concept. All saving accounts are accounts, all flight bookings are booking, and all dogs are animals.

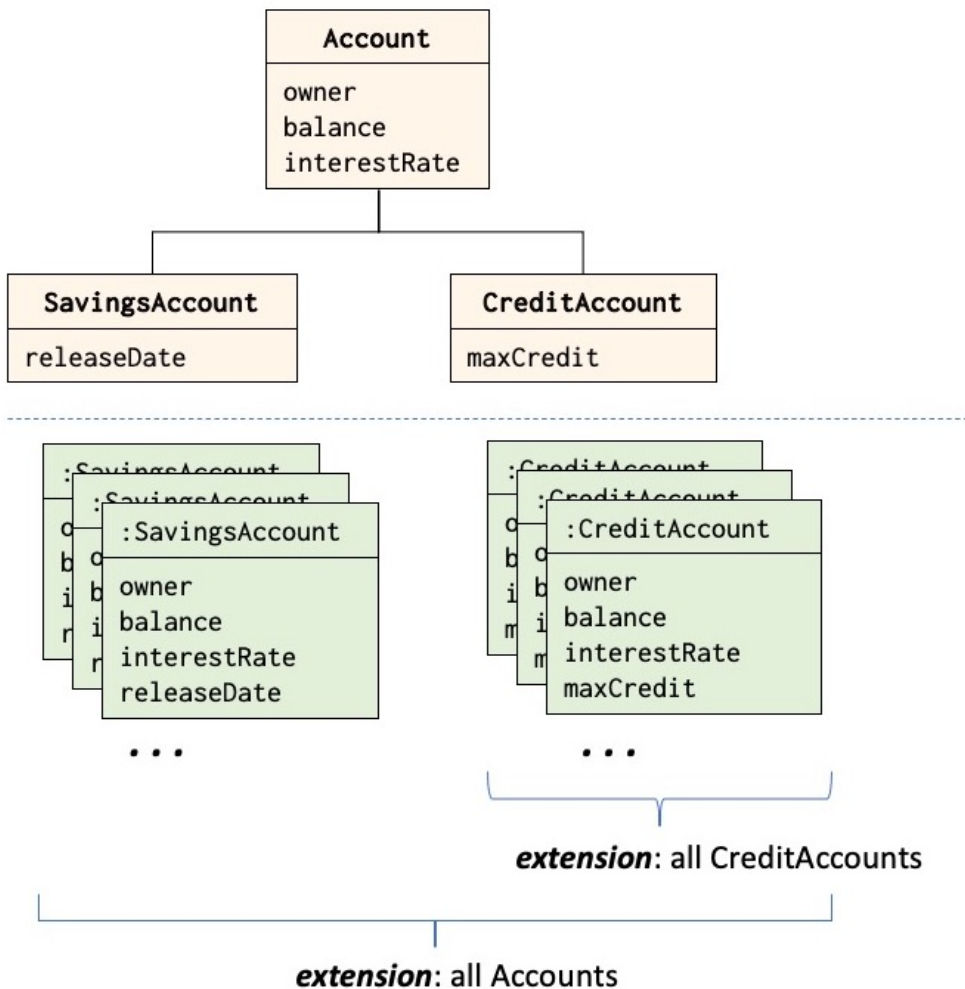
With respect to (2), a general concept is characterised by a number of properties. Specializations of a concept have further properties in addition to this of their generalisation. The properties of a booking are also properties of a flight booking, etc. The properties of a dog are also properties of a cocker spaniel, etc.

#### *Extension of a superclass.*

As said in section , the extension of a class is all objects generated from the class. The extension of a class also includes all objects generated from subclasses of the class.

- The extension of SavingsAccount is all objects generated from class SavingsAccount.
- The extension of CreditAccount is all objects generated from class CreditAccount.
- The extension of Account is all objects generated from class Account and it includes the extensions of SavingsAccount and CreditAccount

The relationships between the extensions of the Account classes are illustrated by the figure below:



The term *subclass* was used in SIMULA and refers to the fact that the extension of a subclass is a subset of the extension of the superclass.

For C++, Bjarne Stroustrup found the terms subclass and superclass to be confusing and he thus introduced the terms *base class* and *derived class* where derived class corresponds to subclass and base class to superclass.

Some languages, like Java, use the term *extend* to define a subclass/derived class - extend refers to the fact that the intension of the superclass is extended.

The class/subclass mechanism as presented here can only be used to represent *tree-structured classification hierarchies*, also referred to as *single inheritance*. In a tree-structured classification hierarchy a concept has at most only one direct super generalisation and an arbitrary number of specialization. When represented by classes this means that a class has at most only one direct superclass and an arbitrary number of subclasses.

There are domains where there is a need for representing non-tree-structured classification hierarchies where a concept/class may have more than one generalization/superclass. We return to this in chapter on aspects and in chapter 1.