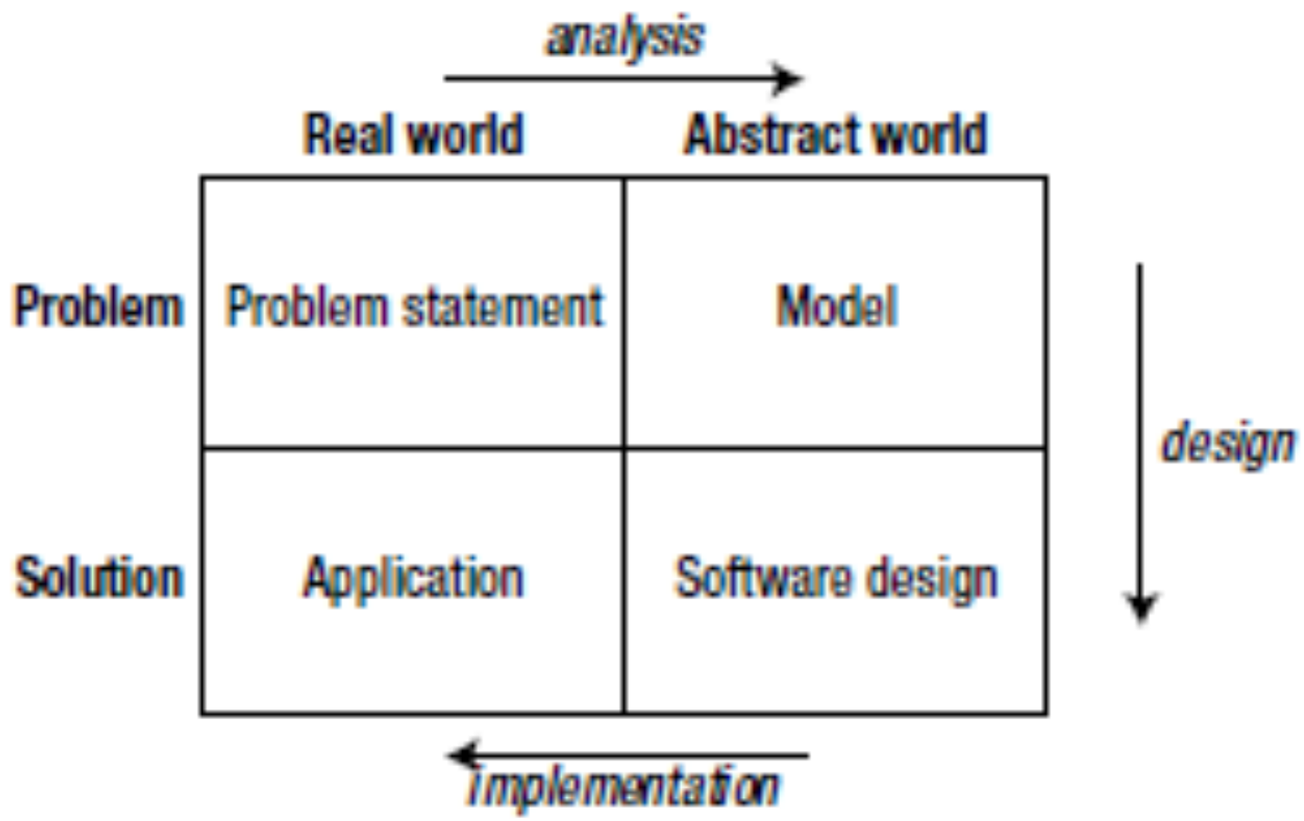
5. Development Process

Fall 2018

- The decision to set up a small database usually arises because there is some specific task in mind
- Not to focus solely on the immediate task at hand but to try to understand the data that are going to support that task *and other likely tasks* referred to as *data independence*.
- Designing a database to reflect the type of data involved, rather than what you currently think is the main use for the data, will be more advantageous in the long term.
- A small business may want to send invoices and statements to its customers. Rather than thinking in terms of a statement and what goes on it, it is important to think about the underlying data items.
- Some abstraction of the problem so that the possibilities become clear.



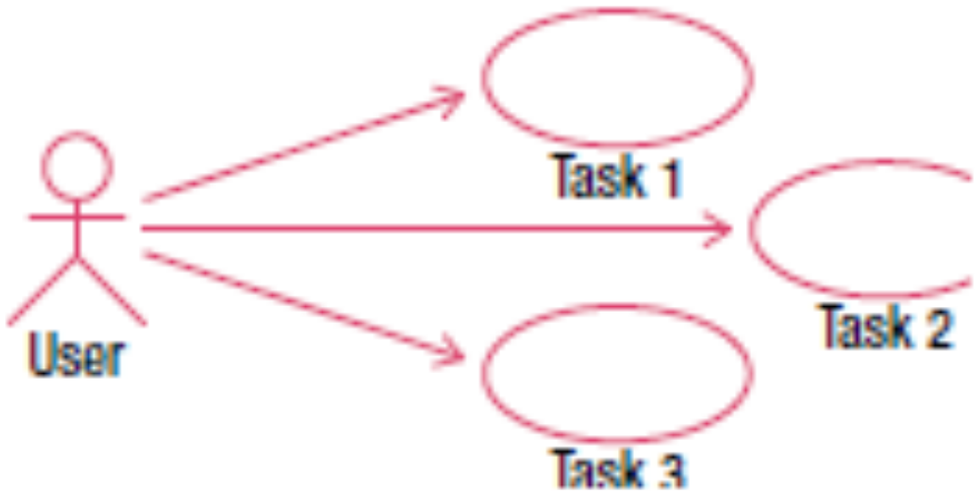
The software process (based on Zelkowitz et al., 19791)

Today's Lecture

1. Initial Problem Statement
2. Analysis and Simple Data Model
3. Further Analysis: Revisiting the Use Cases
4. Design
5. Implementation
6. Excercises
7. Summary

Initial Problem Statement

- One way to represent a description is with *use cases*, part of the *Unified Modeling Language* (UML).
- Use cases are descriptions of how different types of users (more formally known as *actors*) might interact with the system.



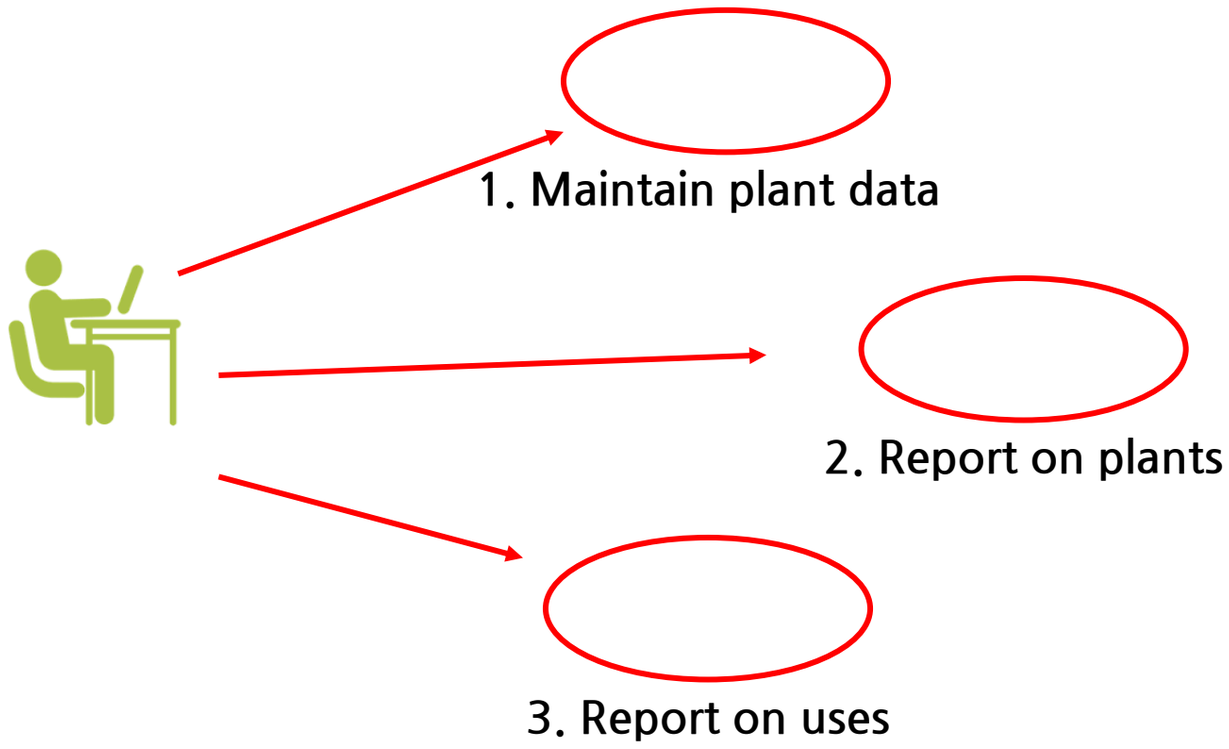
Example: Initial Use Case for Plant DB

A small portion of a database table recording information about plants. Along with the botanical and common names of each plant, the developer decides it would be convenient to keep information on the uses for each plant. This is to help prospective buyers decide whether a plant is appropriate for their requirements.

“I need to store all the info I have about each plant.”

“What uses does this plant have?”

plantID	genus	species	common_name	use1	use2	use3
1	Dodonaea	viscosa	Akeake	shelter	hedging	soil stability
2	Cedrus	atlantica	Atlas cedar	shelter		
3	Alnus	glutinosa	Black alder	soil stability	shelter	firewood
4	Eucalyptus	nichollii	Black peppermint gum	shelter	coppicing	bird food
5	Juglans	nigra	Black walnut	timber		
6	Acacia	mearnsii	Black wattle	firewood	shelter	soil stability



Use case 1: Enter (or edit) all the data we have about each plant; that is, plant ID, genus, species, common name, and uses.

Use case 2: Find or report information about a plant (or every plant) and see what it is useful for.

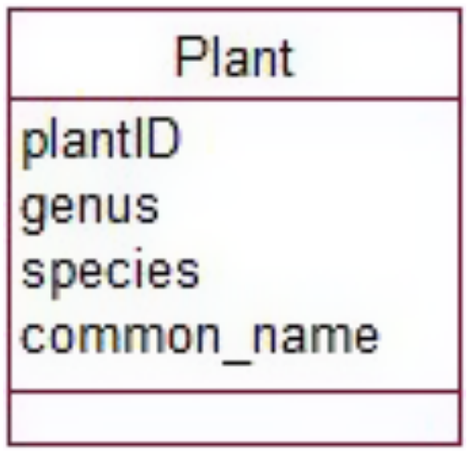
Use case 3: Specify a use and find the appropriate plants (or report for all uses).

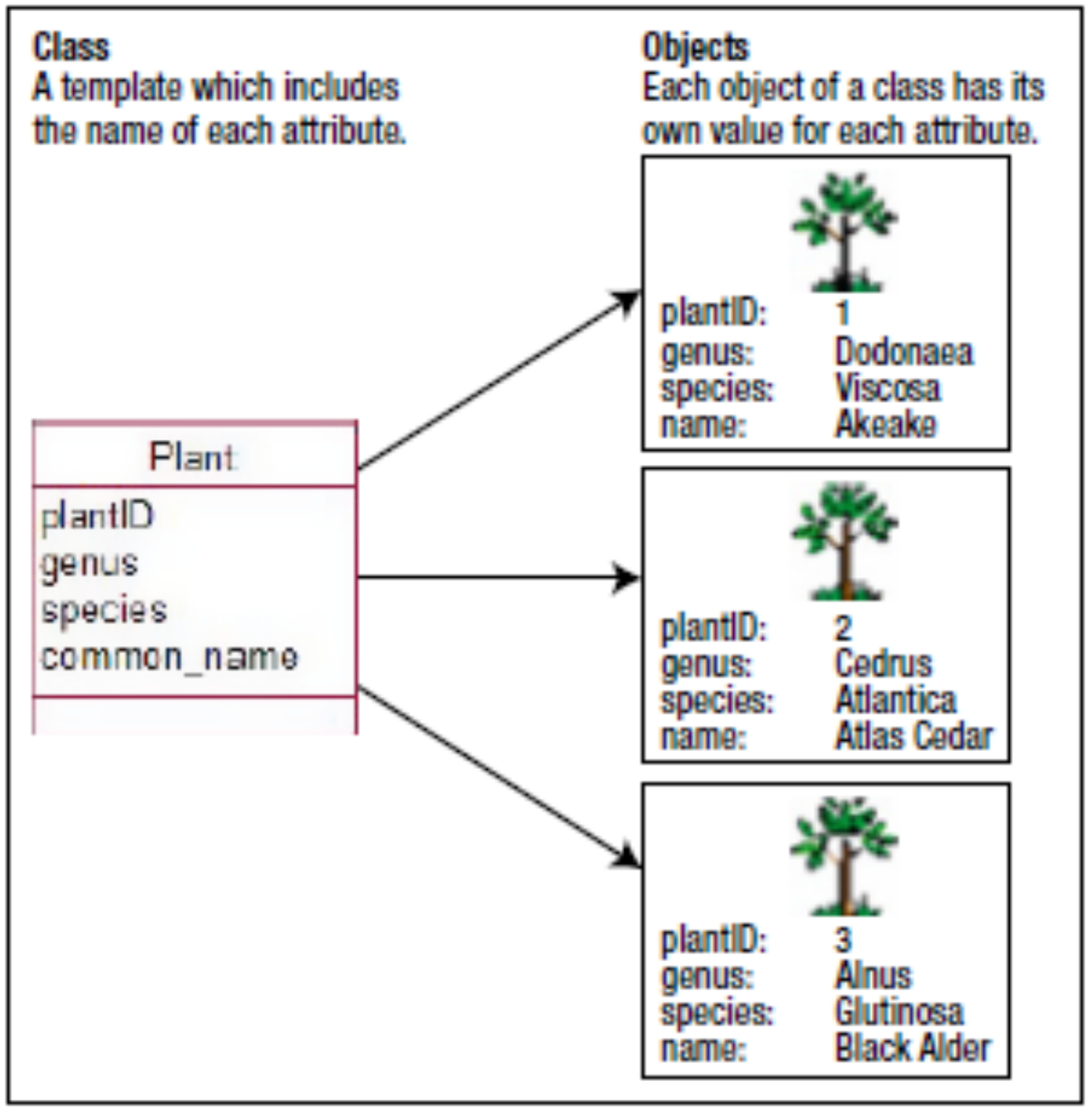
Analysis and Simple Data Model

- Start to get a feel for what the data involves is to sketch an initial data model that is a representation of how the different types of data interact.

Classes and Objects

- Each *class* can be considered a template for storing data about a set of similar things.
- An obvious candidate for our first class is the idea of a **Plant**. Each plant can be described in a similar way in that each has a **genus**, a **species**, a **common_name**, and perhaps a **plantID** number.
- These pieces of information, that we will keep about each plant, are referred to as the ***attributes*** (or ***properties***) of the class.

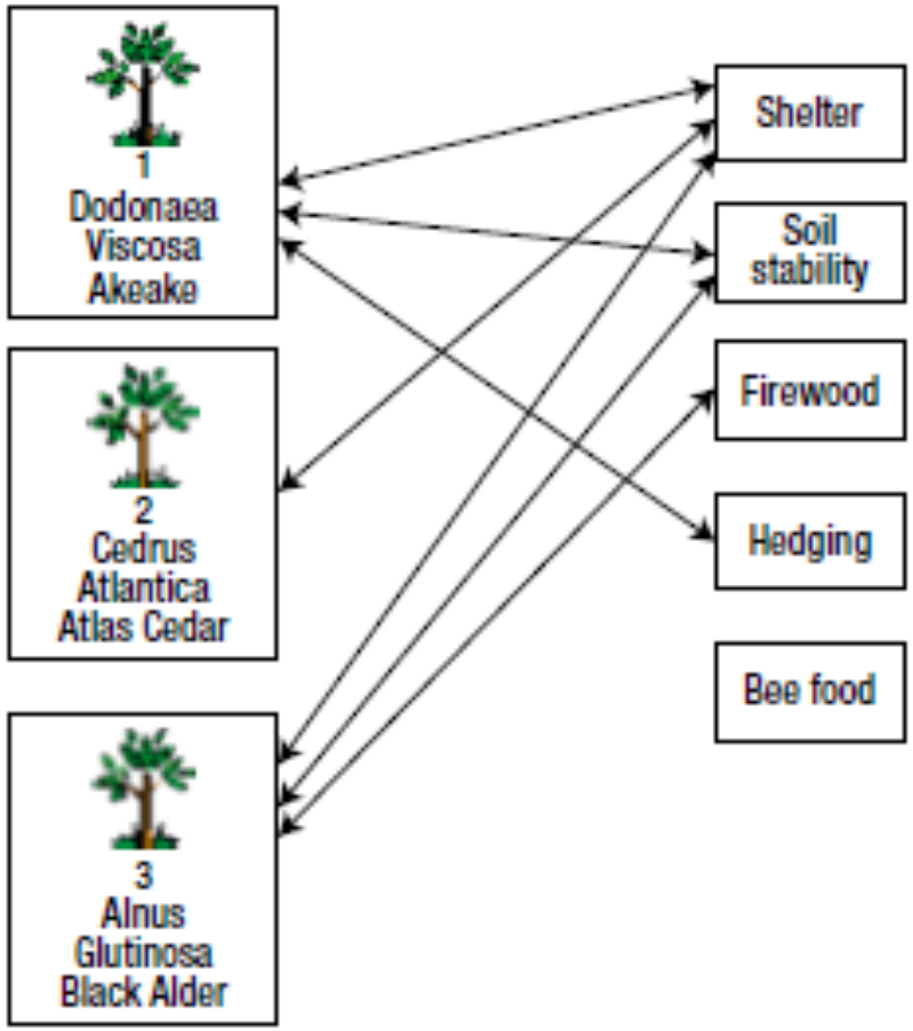



Class
A template which includes the name of each attribute.
Objects
Each object of a class has its own value for each attribute.
Plant
plantID
genus
species
common_name
plantID: 1
genus: Dodonaea
species: Viscosa
name: Akeake
plantID: 2
genus: Cedrus
species: Atlantica
name: Atlas Cedar
plantID: 3
genus: Alnus
species: Glutinosa
name: Black Alder

Another candidate for a class: **Use,** Each object of the **Use** class will have a value for **name** such as “hedging,” “shelter,” or “bird food.”

Relationships

- Akeake can be used for soil stability, hedging, and shelter. We can think of this as a relationship (or association) between particular objects of the Plant class and objects of the Use class.



- We would usually create a table for each class, and the information about each object would be recorded as a row in that table.
- The information about the specific relationship instances would also be recorded in a table.

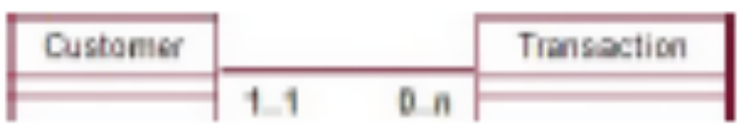
plantID	genus	species	common_name
1	Dodonaea	viscosa	Akeake
2	Cedrus	atlantica	Atlas cedar
3	Alnus	glutinosa	Black alder
4	Eucalyptus	nichollii	Black peppermint gum
5	Juglans	nigra	Black walnut
6	Acacia	mearnsii	Black wattle

Table Plants

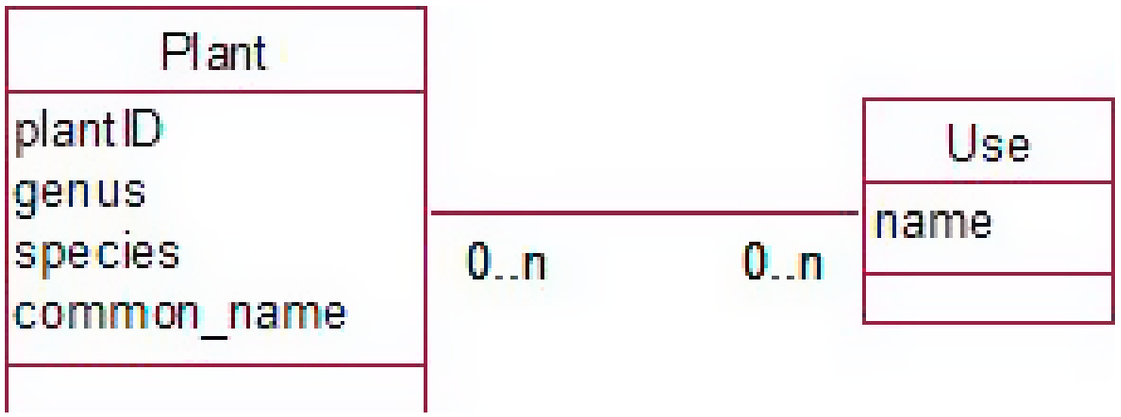
plant	use
1	soil stability
1	hedging
1	shelter
2	shelter
3	firewood
3	soil stability
3	shelter

Table Uses

- A relationship is represented by a line between two class rectangles. The line can be named to make it clear what the relationship.
- The pair of numbers at each end of the line indicates how many objects of one class can be associated with a particular object of the other class.
- The first number is the minimum number. This is usually 0 or 1 and is therefore sometimes known as the ***optionality***.
- The second number is the greatest number of related objects. It is usually 1 or many (denoted n), although other numbers are possible. These numbers can be referred to as the ***cardinality*** or the ***multiplicity*** of the relationship.

	Left to Right	Right to Left
	One particular plant may have no uses or it could have any number	One particular use may have no plants associated with it, or it may have many plants
	One person may have lots of interests or may have none	Each interest has at least one person associated with it and maybe several
	One customer may have several transactions but might not have any	Each transaction is associated with exactly one customer
	A visit has at least one sample associated with it and maybe many	Each sample comes from a single visit

Further Analysis: Revisiting the Use Cases



- We now need to check whether this model is able to satisfy the requirements of the three use cases:

Use case 1: Maintain plant information.

We can create objects for each plant and record the attributes we might require now or in the future. We can create use objects, and we can specify relationship instances between particular plant and use objects.

Use case 2: Report on plants.

We can take a particular plant object (or each one in turn) and find the values of its attributes. We can then find all the use objects related to that plant object.

Use case 3: Report on uses.

We can take a particular use object and find all the plant objects that are related to it.

Use case 1: Maintain plant information.

- Two or maybe three separate tasks:
- We consider how the database will actually work in practice, it seems likely that the different uses (hedging, shelter, etc.) would be entered right at the start of the project and be updated from time to time.
- Entering information about uses is a task that a user might want to perform independently of any specific plant information. At some later time, the same user, or someone else, may want to enter details of a plant and relate it to the uses that are already recorded..

How will it be done in practice? Will different people be involved? Will bits of the data be entered at different times?

For data entry or editing, separate the tasks done by different people or at different times into their own use cases.

Use case 2: Report on plants.

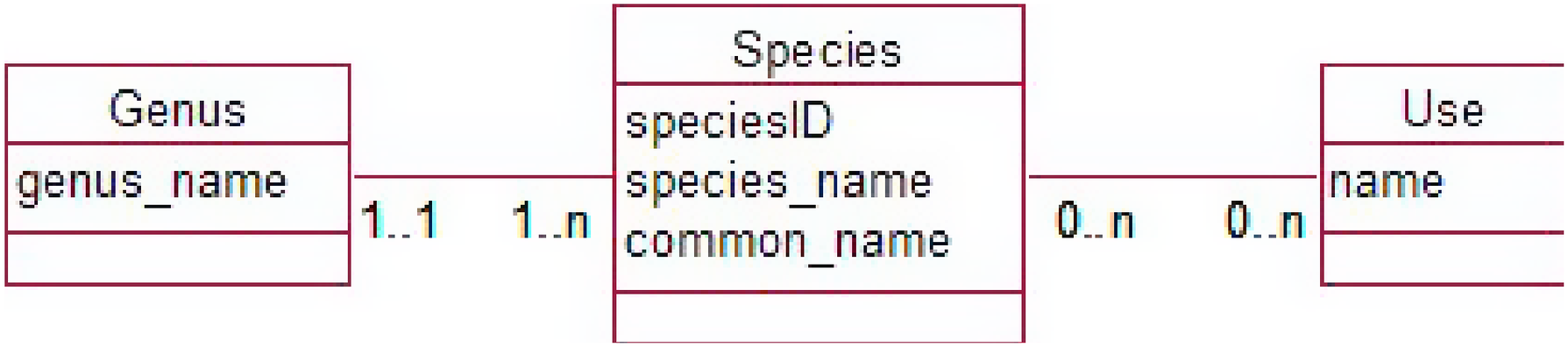
Think about the following dialog:
You: Would you like to be able to print out a list of all your plants to put in a folder or send to people?
User: That would be good.
You: What order would you like the plants to be listed in?
User: By their genus, I guess. Alphabetical?
You: Genus? So you'd like, for example, all the Eucalyptus plants together.
User: Yep, that would be good.

We see another level of the problem. If we look carefully at the data in the original table, we can see that it appears that each genus includes a number of species, and each of these species can have many uses.

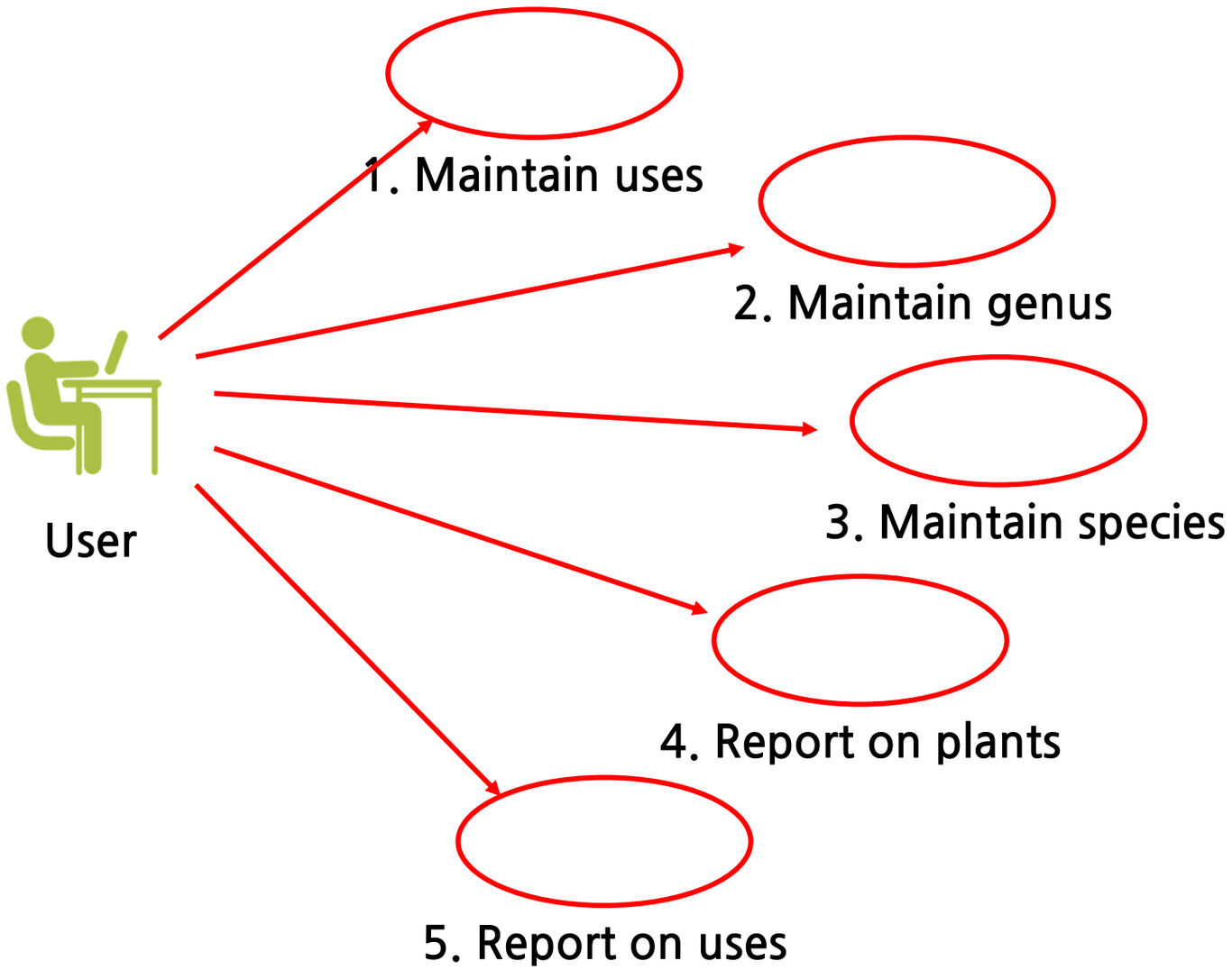
You: So each species belongs to just one genus? Is that right?
User: That's right.

For data retrieval or reporting tasks, ask questions about which attributes might be used for sorting, grouping, or selecting data. These attributes may be candidates for additional classes.

- We now realize that we have a new class, Genus, to add to our data model. Why is it important to include this new class?
- The fact that our user has mentioned that grouping by genus would be useful means that it is important to get the genus data stored appropriately.



Example: Revised Use Cases for the PLANT DB



Use case 1: Maintain uses. Create or update a use object. Enter (or update) the name.

Use case 2: Maintain genus. Create or update a genus object. Enter the name.

Use case 3: Maintain species. Create a species object. Generate a unique ID, and enter the species and common name. Associate the new species object with one of the existing genus objects and optionally associate it with any number of the existing uses.

Use case 4: Report plant information. For each genus object, write out the name and find all the associated species objects. For each species object, write out the species and common name. Find all the associated uses and write out their names.

Use case 5: Report use information. For each use object, write out the name. Find all the associated species objects, and write out for each the associated genus name and the species and common names.

Design

- Consider what type of software would be suitable for implementing the project.

Implementation

genus_name
Acacia
Aesculus
Alnus
Aristotelia
Betula
Boronia
Brachyglottis
Cedrus
Chaenomeles
Chionochloa
Clianthus

Table Genus

speciesID	species_name	common_name	genus
1	viscosa	ake-ake	Dodonaea
2	atlantica	atlas cedar	Cedrus
3	nigra	black walnut	Juglans
4	melanoxylon	Tasmanian blackwood	Acacia
5	hippocastanum	Horse Chestnut	Aesculus
6	glutinosa	Black alder	Alnus
7	incana	grey alder	Alnus
8	cordata	Italian alder	Alnus
9	serrata	Wineberry ; Mako Mako	Aristotelia
10	pendula	Silver birch	Betula

Table Genus

use
bee food
bird food
coppicing
firewood
hedging
shelter
soil stability
timber

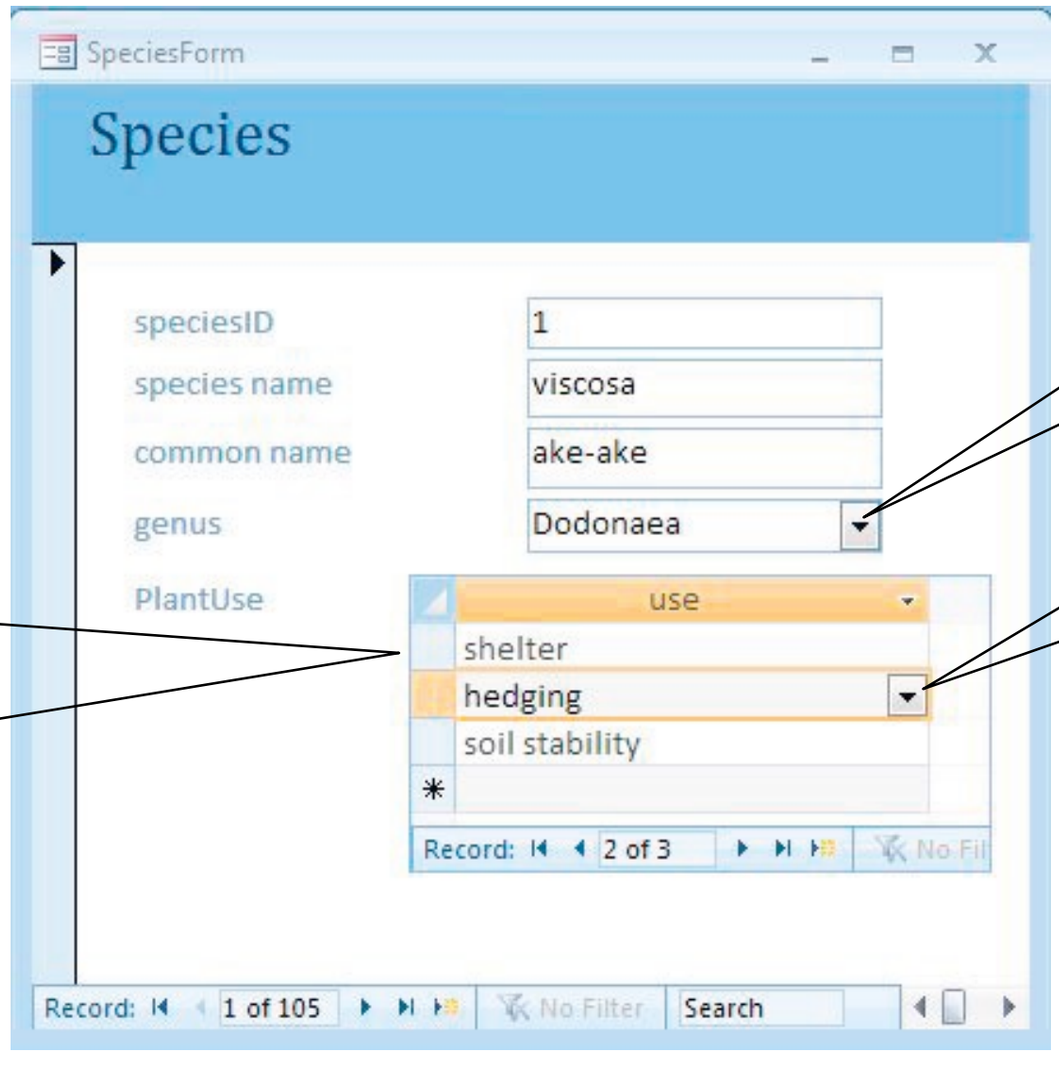
Table Use

plant	use
1	hedging
1	shelter
1	soil stability
2	shelter
3	firewood
3	shelter
3	soil stability
4	bird food

Table PlantUse

Interfaces for Input Use Cases

- The use case for maintaining species information is a little trickier. We need to update two tables: **Species** (for the data about each species) and **PlantUse** (because we need to specify which uses each species is associated with).



Reports for Output Use Cases

- The reporting and querying facilities in database products make extracting (simple) information.

PlantUse

Use	ID	Genus	Species Name	Common Name
bird food				
	4	Acacia	melanoxylon	Tasmanian blackwood
	7	Alnus	incana	grey alder
	28	Eucalyptus	nichollii	Black peppermint gum
coppicing				
	30	Eucalyptus	gunnii	cider gum
	4	Acacia	melanoxylon	Tasmanian blackwood
	28	Eucalyptus	nichollii	Black peppermint gum
firewood				
	6	Alnus	glutinosa	Black alder

genus name	speciesID	species name	common name	use
Dodonaea	1	viscosa	ake-ake	shelter
Dodonaea	1	viscosa	ake-ake	hedging
Dodonaea	1	viscosa	ake-ake	soil stability

Dodonaea

Exercise 5-1

A small sports club keeps information about its members and the fees they pay. The secretary wants to be able to **record when members pay and print a report similar to the following figure**.

last_name	first_name	phone	type	gender	fee	date_paid
Smith	Jane	563201	Full	F	220	21/09/2011
Wilson	Harry	375967	Full	M	220	19/09/2011
Green	Bert	439871	MidWeek	M	150	
Jones	Bert	295784	Social	F	80	
Smith	Sharon	387648	MidWeek	F	150	16/08/2011

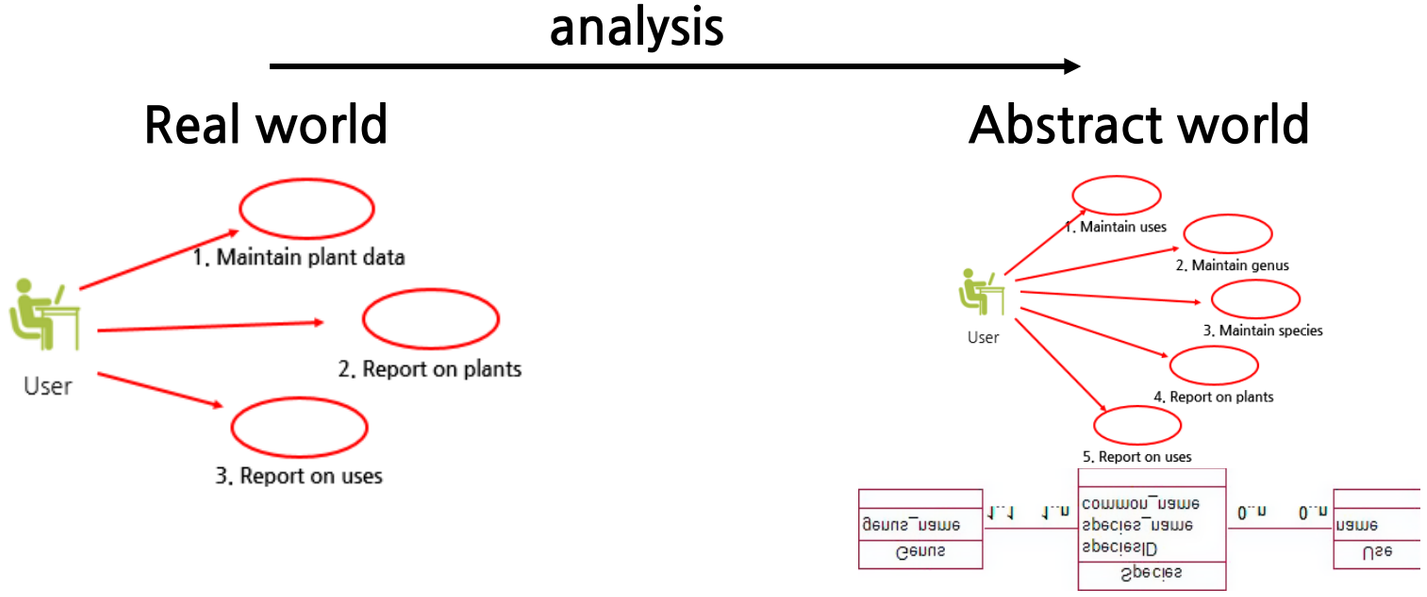
a. Think about when the different pieces of data might be entered. Sketch an initial use case diagram for data entry.

b. Consider what different things you are keeping information about and sketch a simple class diagram.

c. What options could you suggest to the club for different ways a report could be presented? Does your class diagram have the information readily available?

Summary

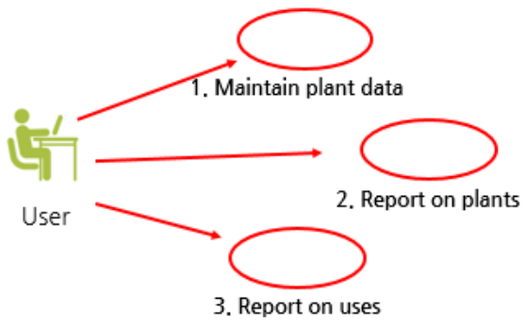
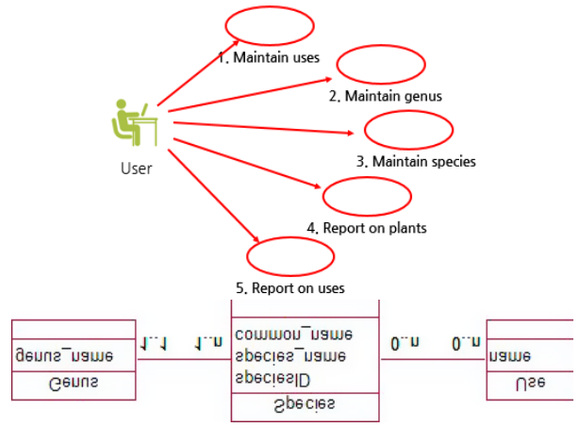
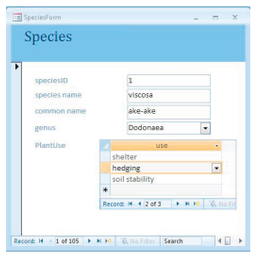
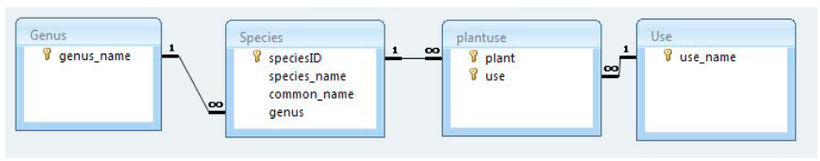
1. Express the problem in terms of what a user might want to achieve. For a database problem, this will typically be in terms of the data to be stored and the information that needs to be retrieved.
2. Think about other possible uses of the information and how the data might be usefully ordered or grouped. Undertake an iterative analysis process of reconsidering the data model and the use cases, until you are satisfied



Summary

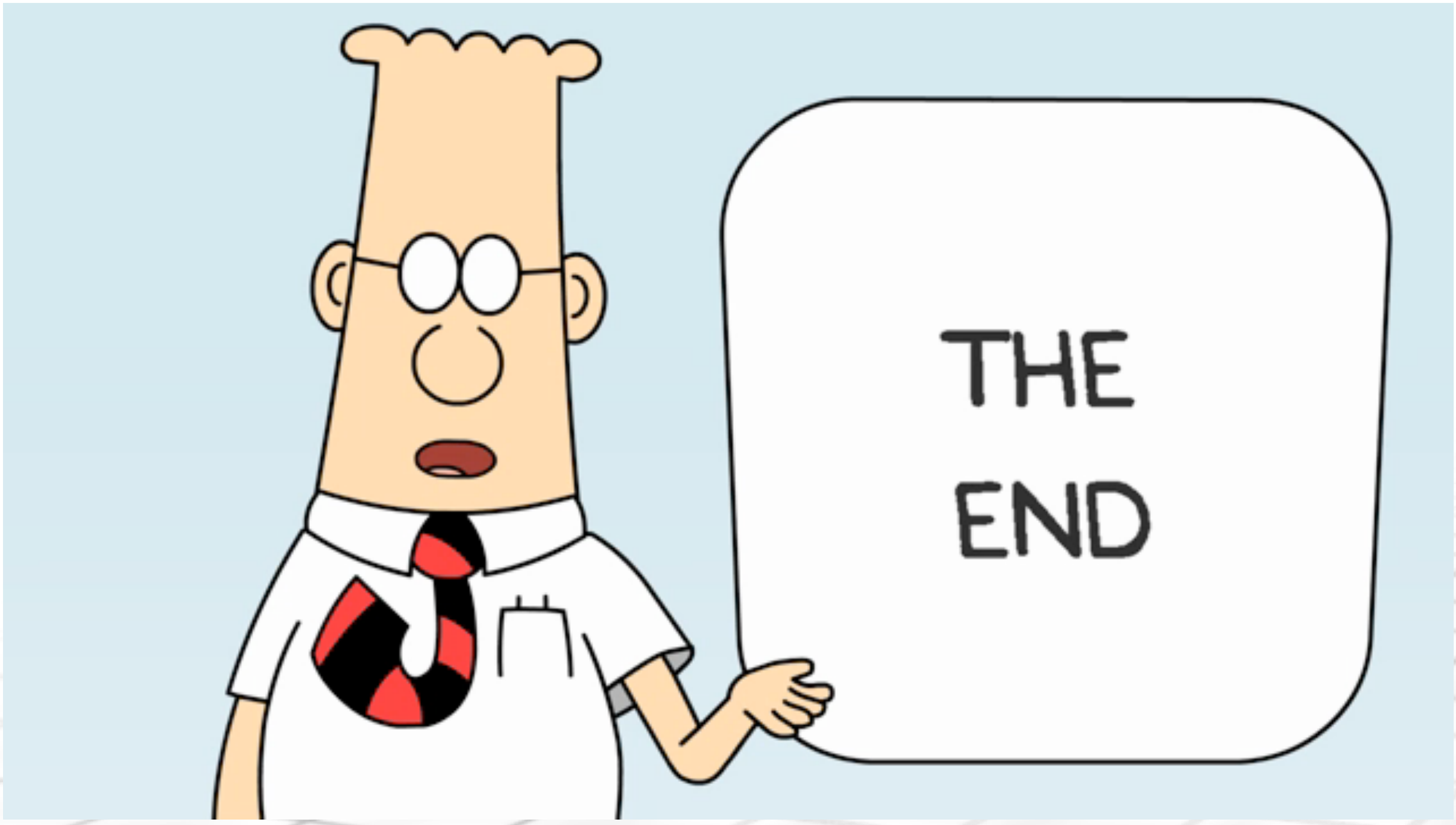
3. Choose the type of product to manage the data and create an appropriate design. For a relational database, this will involve designing tables, keys, and foreign keys.
4. Build the application. For a relational database, this will include setting up the tables and developing forms and reports to satisfy the use cases.

analysis →

	Real world	Abstract world
Problem		
Solution	  PlantUse Use / ID / Genus / Species Name / Common Name bird food 4 Acacia melanoxylon Tasmanian blackwood 7 Alnus incana grey alder 28 Eucalyptus nichollii Black peppermint gum coppicing 30 Eucalyptus gunnii cider gum 4 Acacia melanoxylon Tasmanian blackwood 28 Eucalyptus nichollii Black peppermint gum firewood 6 Alnus glutinosa Black alder 3 Juglans nigra black walnut hedging	

design ↓

← *Implementation*



출처: metachannels.com

Thank you!