

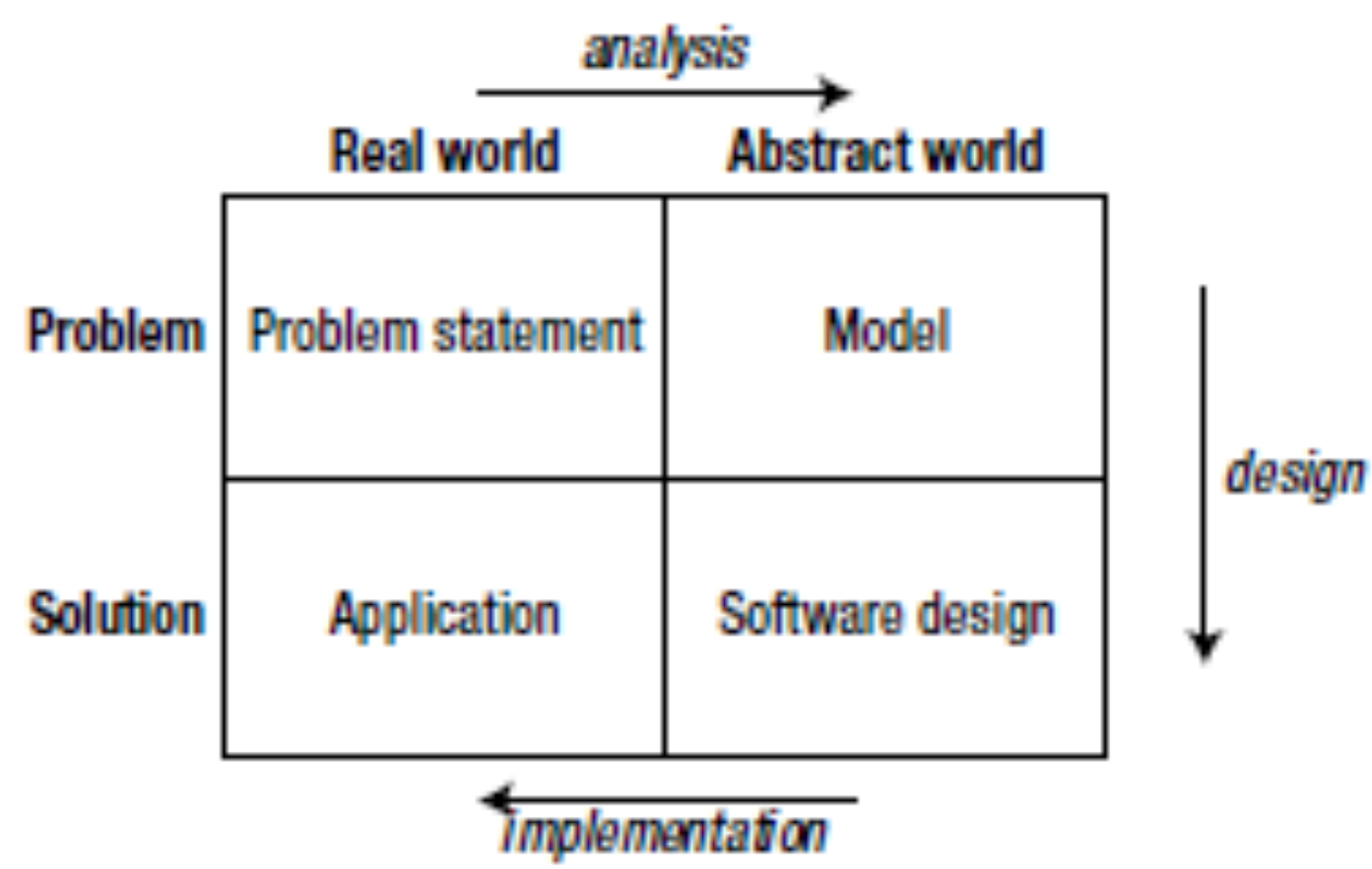
# 3. Development Process

# Schedule\*

0. Seminar Overview
1. Overview of the Relational Data Model
2. Why DB Design?
3. Development Process
4. Requirements
5. Conceptual Data Model
6. Generalization & Specialization
7. Relational Database Design
8. Normalization
9. Keys and Constraints

\*subject to change without notification

- 일반적으로 몇 가지 특정 작업만을 염두에두고 작은 데이터베이스를 설정하는 결정을 내립니다.
- 직접적인 즉각적인 작업에 집중할 뿐만 아니라 해당 작업을 지원할 데이터 및 데이터 독립성이라고 하는 기타 작업도 이해하여야 합니다.
- 데이터의 현재 주된 사용만을 생각하는 것보다 관련 데이터 타입을 반영하도록 데이터베이스를 설계하는 것이 장기적으로 바람직할 것입니다.
- 중소기업이 고객에게 송장과 명세서를 보내고자 할 수 있습니다. 이때 명세서와 진행상황에 대해 생각하기 보다는 사용될 데이터 항목에 대해 생각하는 것이 중요합니다.
- 가능성을 분명히 하기 위해 문제를 추상화합니다.



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

# What will be dealt

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

# Initial Problem Statement

- 문제를 기술하는 한 가지 방법은 UML (Unified Modeling Language)의 일부인 use case를 사용하는 것입니다.
- use case는 (공식적으로는 행위자라고 하는) 여러 유형의 사용자가 시스템과 상호 작용하는 방식에 대한 설명입니다.



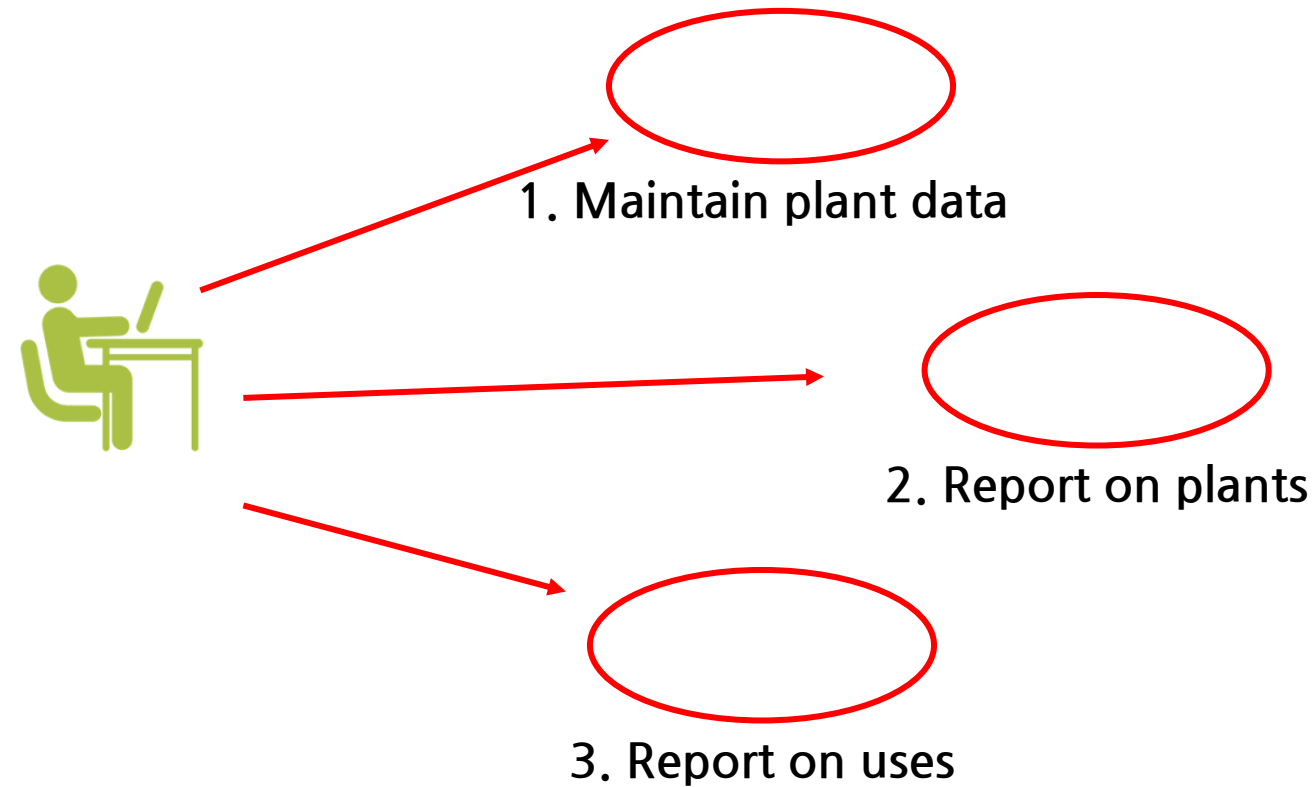
## Example: Initial Use Case for Plant DB

나무에 관한 정보를 기록하는 데이터베이스 테이블의 작은 부분입니다. 개발자는 각 나무의 식물학 및 일반 이름과 함께 용도에 대한 정보를 유지하고자 합니다. 나무가 구매자의 요구 사항에 대한 적합 여부를 결정할 수 있도록 예비 구매자를 돕기 위한 것입니다.

“나무에 관한 모든 정보를 저장합니다.”

“이 나무의 용도는?”

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:** 각 플랜트에 대한 모든 데이터를 입력 (또는 편집). 즉 식물 ID, 속, 종, 일반 이름 및 용도.

**Use case 2:** 지정한 플랜트 (또는 모든 플랜트)에 대한 정보를 찾거나 보고하며, 나무의 용도를 확인합니다.

**Use case 3:** 용도를 지정하고 (또는 모든 용도에 대하여) 적절한 식물을 찾습니다.

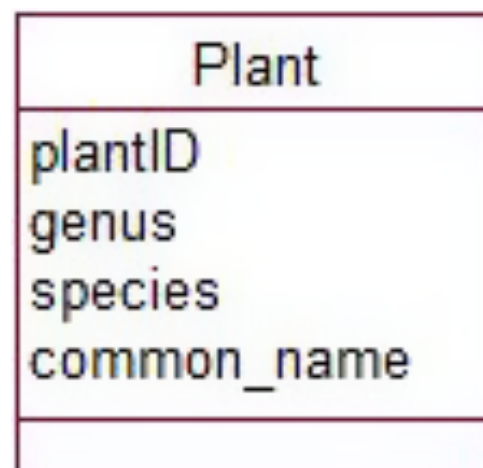


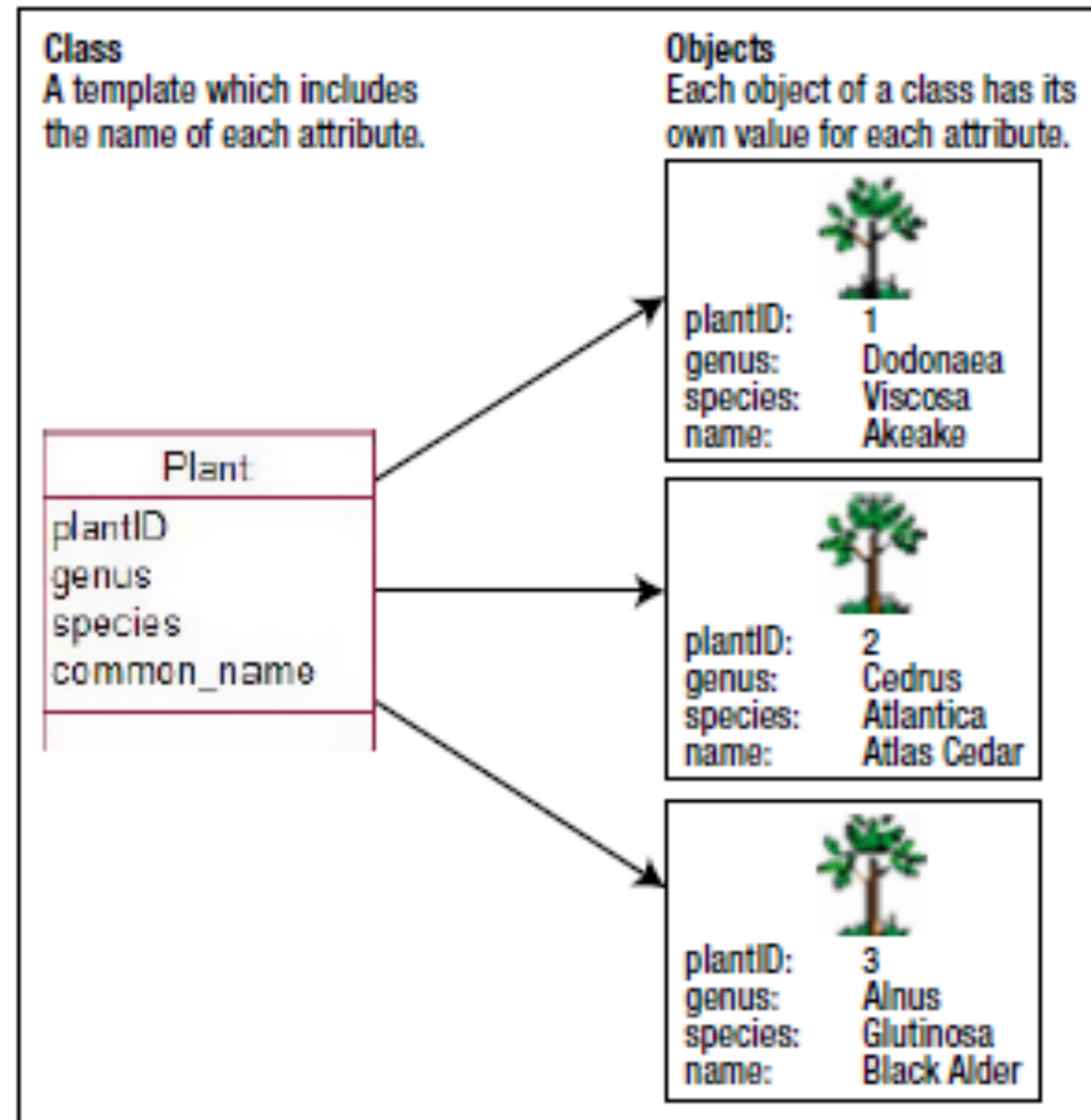
# Analysis and Simple Data Model

- 다른 타입의 데이터가 상호 작용하는 방식을 나타내는 초기 데이터 모델을 스케치하면 데이터와 관련된 사항에 대한 느낌으로 시작합니다.

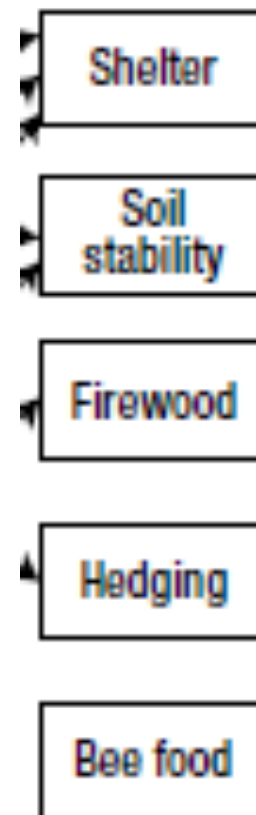
# Classes and Objects

- 클래스는 유사한 것들의 데이터를 저장하기 위한 템플릿으로 생각할 수 있습니다.
- 첫 번째 클래스의 분명한 후보자는 식물일 것입니다. 속 (genus), 종 (species), 일반이름 (common\_name), 아마도 식물 ID 번호 (plantID number)를 갖는 것과 비슷한 방식으로 각 식물을 기술 할 수 있습니다.
- 각 식물에 대하여 저장할 정보의 이러한 부분을 클래스의 **애틀리뷰트** (또는 **속성**)이라고 합니다.



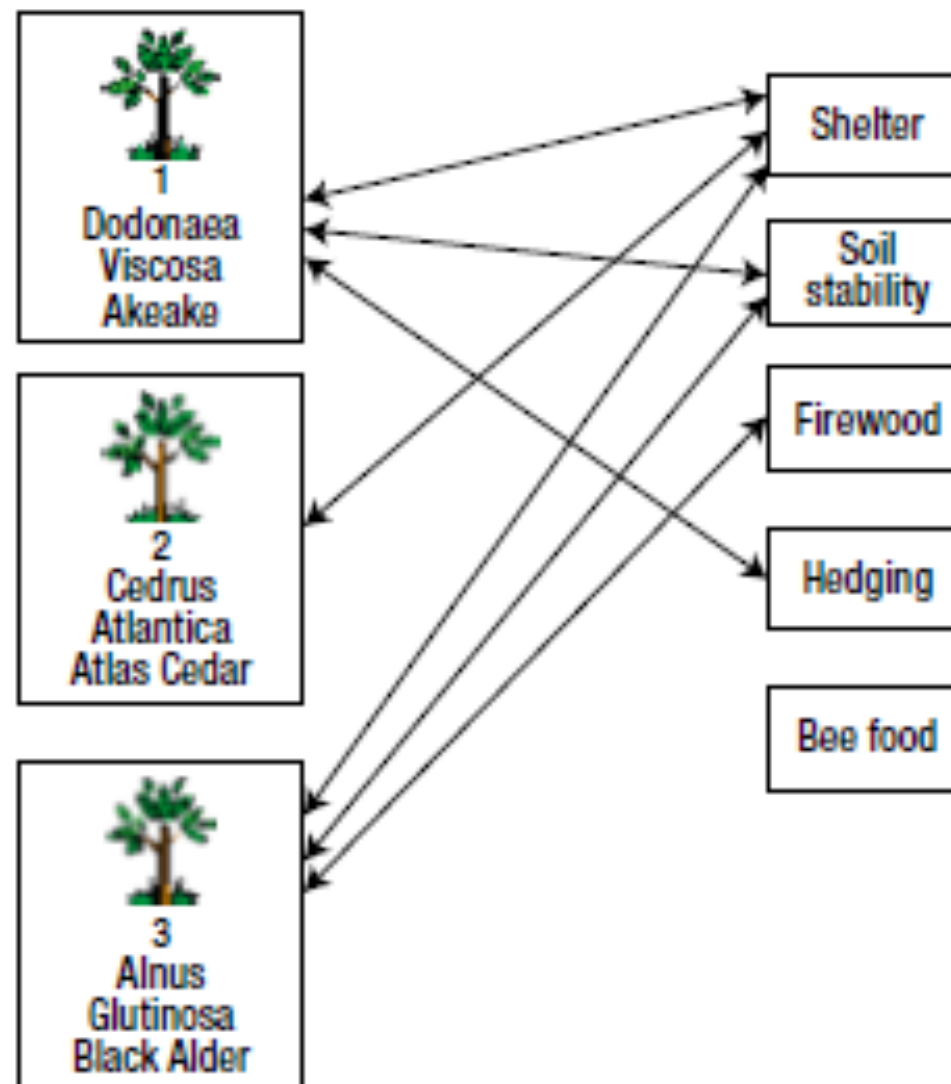


클래스의 다른 후보로는 **Use**를 들 수 있습니다. Use 클래스의 각 객체는 "hedging", "shelter" 또는 "bird food"와 같은 값을 갖습니다.



# Relationships

- Akeake는 토양 안정성, 헛지 및 심터로 사용될 수 있습니다. 우리는 이것을 Plant 클래스의 특정 객체와 Use 클래스의 객체 사이의 relationship(또는 association)라고 생각할 수 있습니다.



- 일반적으로 각 클래스를 테이블로 만들고 각 객체에 대한 정보를 해당 테이블에 행으로 기록합니다.
- 특정 relationship에 대한 정보도 테이블에 기록합니다.

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

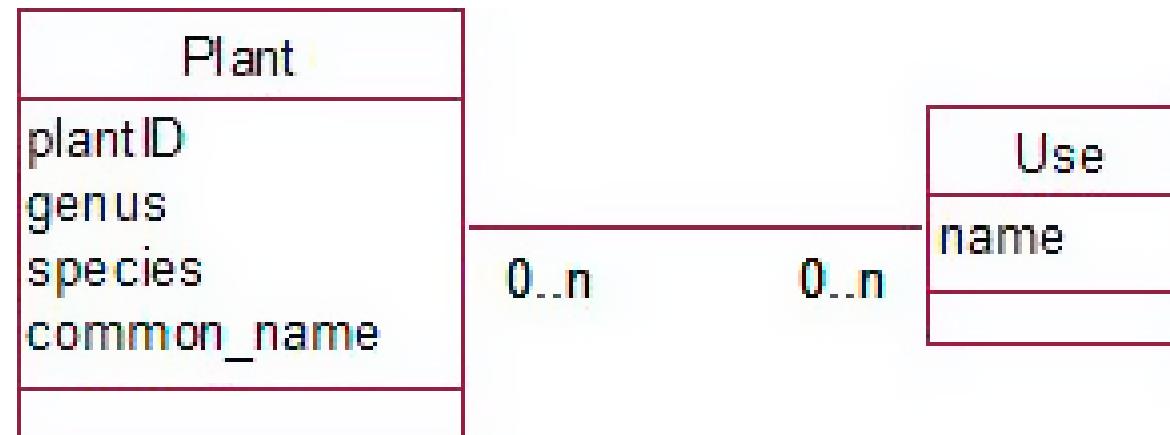
- 두 개의 클래스 사각형 사이의 선으로 relationship을 표시합니다. relationship를 명확하게 하기 위해 그 선을 명명할 수 있습니다.
- 선의 양 끝에있는 숫자 쌍은 한 클래스의 객체 수를 상대 클래스의 특정 객체와 연관되어 있음을 나타냅니다.
- 첫 번째 숫자는 최소 숫자이며 대개 0 또는 1입니다. 따라서 **선택성(optionality)**이라 합니다.
- 두 번째 숫자는 관련 개체의 최대 수입니다. 일반적으로 1 또는 다수 (n으로 표시)하지만 다른 숫자도 가능합니다. 이 수는 **카디널리티(cardinality)** 또는 관계의 **다양성(multiplicity)**이라고 합니다.



	Left to Right	Right to Left
 <pre> graph LR     Plant[Plant] --- 0..n  Use[Use]     </pre>	<p>One particular plant may have no uses or it could have any number</p>	<p>One particular use may have no plants associated with it, or it may have many plants</p>
 <pre> graph LR     Person[Person] --- 1..n  Interest[Interest]     </pre>	<p>One person may have lots of interests or may have none</p>	<p>Each interest has at least one person associated with it and maybe several</p>
 <pre> graph LR     Customer[Customer] --- 1..1  Transaction[Transaction]     </pre>	<p>One customer may have several transactions but might not have any</p>	<p>Each transaction is associated with exactly one customer</p>
 <pre> graph LR     Visit[Visit] --- 1..1  Sample[Sample]     </pre>	<p>A visit has at least one sample associated with it and maybe many</p>	<p>Each sample comes from a single visit</p>



# Further Analysis: Revisiting the Use Cases



- 모델이 세 가지 use case의 요구 사항을 충족시킬 수 있는지 여부를 확인합니다.

**Use case 1:** 나무 정보 관리.

각 나무에 대한 객체를 생성하고 현재 또는 미래에 필요할 수 있는 애트리뷰트를 기록 할 수 있습니다. Use 객체를 생성 할 수 있으며, 특정 Plant 객체와 Use 객체간에 관계 인스턴스를 지정할 수 있습니다.

**Use case 2:** 나무에 대한 검색.

특정 plant 객체 (또는 차례로 하나씩)의 해당 애트리뷰트 값을 찾습니다. 그런 다음 해당 plant 객체와 관련된 모든 use 개체를 찾습니다.

**Use case 3:** 사용에 대한 검색.

특정 use 객체와 관련된 모든 plant 객체를 찾습니다.

## Use case 1: 나무 정보 관리.

- 두 세 가지 개별 작업
- 데이터베이스의 실제 실행 방법을 고려할 때, 용도 (hedging, shelter 등)는 프로젝트 시작시 바로 입력되고 수시로 업데이트 될 것으로 예상됩니다.
- use에 정보를 입력하는 것은 사용자가 특정 plant 정보에 대한 작업과 독립적으로 수행하고자 하는 작업입니다. 나중에 동일한 사용자 또는 다른 사람이 플랜트의 세부 정보를 입력하고 이미 저장된 use와 연결시킬 수 있습니다.

실제로 어떻게 수행되니까? 다른 사람들이 참여하여야 합니까? 다른 시점에 데이터를 입력할 예정입니까?

데이터 입력 또는 편집의 경우, 다른 사람이 수행하는 작업이나 다른 시간에 수행 하는 작업을 각각 use case로 분리하십시오.

## Use case 2: 나무에 대한 검색.

다음 대화에 대하여 살펴봅시다.

**You:** 폴더에 넣거나 사람들에게 보낼 모든 plant 목록 인쇄가 필요합니까?

**User:** 그러면 좋습니다.

**You:** 어떤 순서로 인쇄할까요?

**User:** By their genus, I guess. Alphabetical? 속순으로, 가나다 순으로?

**You:** 속순? 예를 들면 모든 Eucalyptus 속에 속한 나무들 모두 함께요?

**User:** 네, 좋을 것 같아요.

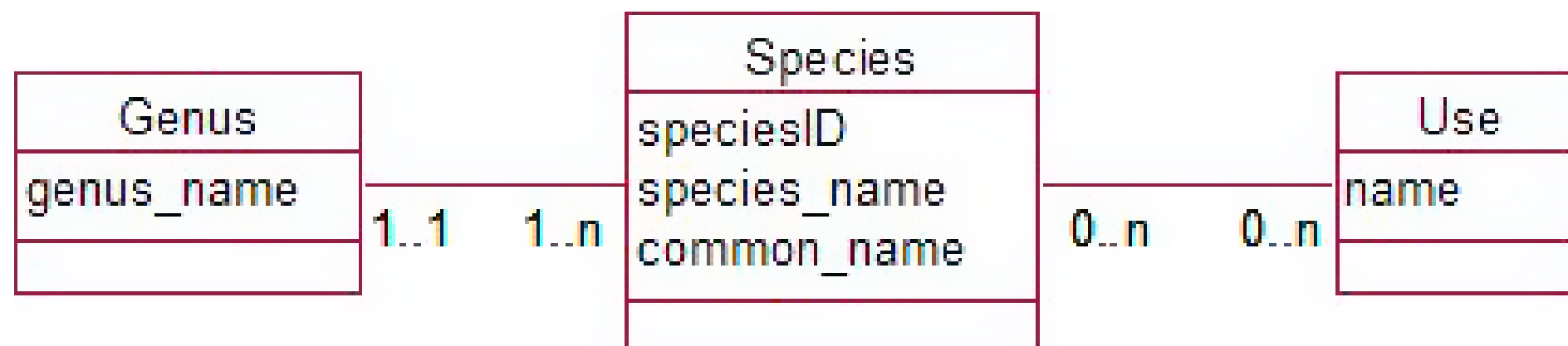
또 다른 문제가 있습니다. 원래 테이블의 데이터를 주의 깊게 살펴보면, 속은 여러 종으로 나누어져 있으며, 종 각각은 여러 용도가 있음을 알 수 있습니다.

**You:** 각 종은 하나의 속에만 속하나요?

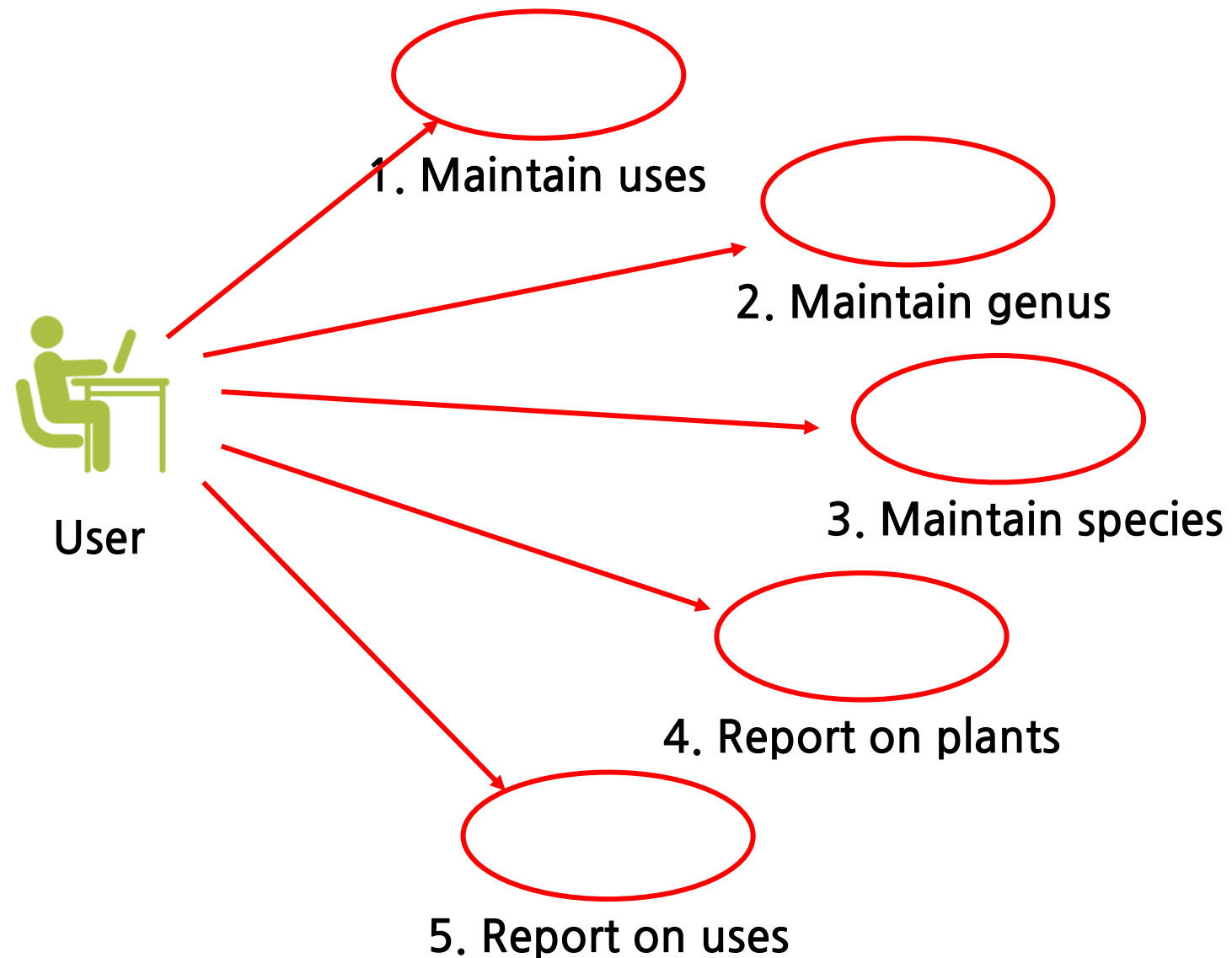
**User:** 맞습니다.

데이터에 대한 검색 또는 보고 작업의 경우, 데이터를 정렬, 그룹화 또는 선택하는 데 사용할 수 있는 속성에 대해 사용자에게 질문하십시오. 이러한 속성은 추가 클래스의 후보가 될 수 있습니다.

- 이제 데이터 모델에 추가 할 새로운 클래스 인 Genus가 있음을 알게되었습니다. 이 새로운 틀래스를 포함하는 것이 중요한 이유는 무엇일까요?
- 사용자가 속 (genus)으로 그룹핑하는 것에 대한 필요성 언급은 적절히 저장된 속(genus) 데이터 검색이 중요하다는 것을 의미합니다.



## Example: Revised Use Cases for the PLANT DB



**Use case 1:** Maintain uses. Use 객체를 만들거나 업데이트합니다. 이름을 입력 (또는 갱신)합니다.

**Use case 2:** Maintain genus. Genus 객체를 생성하거나 업데이트합니다. 이름을 입력합니다.

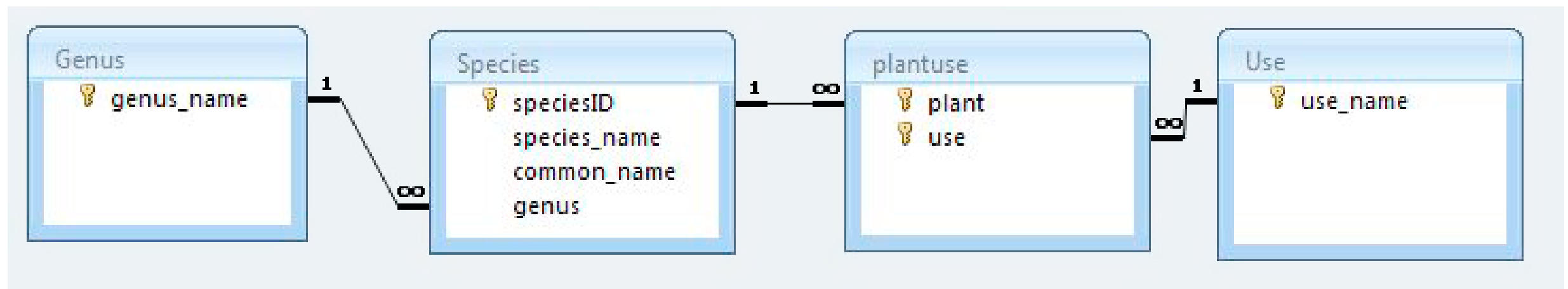
**Use case 3:** Maintain species. Species 객체를 만듭니다. 고유 한 ID를 생성하고 종과 일반 이름을 입력합니다. 새로운 Species 객체를 기존의 Genus 객체 중 하나와 연결시키고 선택적으로 임의의 수의 기존 Use 객체와 연결시킵니다.

**Use case 4:** Report plant information. 각 Genus 객체에 대해 이름으로 모든 관련 Species 객체를 검색합니다. 각 Species 객체에 대해 종과 공통 이름을 입력하여, 관련된 모든 용도를 찾아 이름을 검색합니다.

**Use case 5:** Report use information. 각 Use 객체에 대해 사용 이름에 대해 모든 관련 Species 객체를 찾아 각각의 연관된 속 이름과 종 및 공통 이름을 검색합니다.

# Design

- 어떤 종류의 소프트웨어가 프로젝트 구현에 적합할 지 생각합니다.



# 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 Species

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

- Species 정보를 유지하는 use case는 조금 까다롭습니다. Species (각 종에 관한 데이터)와 PlantUse (각 종에 연결되는 사용을 지정해야하기 때문에) 두 테이블을 업데이트해야 합니다.

The screenshot shows a web application window titled "SpeciesForm". Inside, there's a "Species" section with a blue header. Below it, there are input fields for "speciesID" (value: 1), "species name" (value: viscosa), "common name" (value: ake-ake), and "genus" (value: Dodonaea). A "PlantUse" section contains a subform titled "use" with a list of options: "shelter", "hedging", "soil stability", and a row with an asterisk (\*). The "hedging" option is selected. At the bottom of the form, there's a record indicator showing "Record: 1 of 105" and a "Search" button.

Subform to choose multiple Uses which will End up in Plant Use table

Drop down list to choose genus

Drop down list to choose use

# Reports for Output Use Cases

- 데이터베이스의 보고서 작성과 조회 기능은 단순히 정보를 추출합니다.

## 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	1	Dodonaea	viscosa	ake-ake

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 3-1

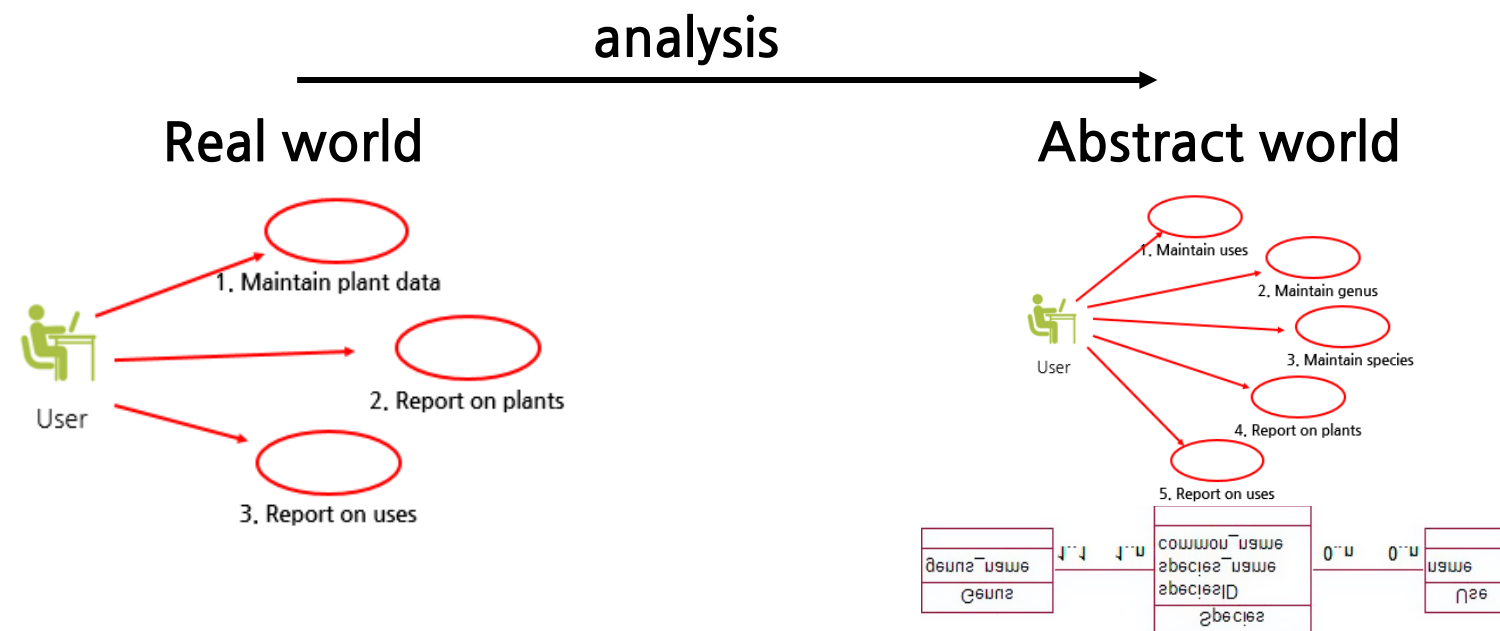
학적 서비스는 교무 행정을 위하여 다양한 통계와 분석 자료, 개인별 학적 변동 이력을 제공하기 위한 것으로 아래와 같은 사용자 요구 사항을 만족하여야 한다.

```
Students ( Person_ID,
           Student_Num,
           LastName, FirstName,
           MobilePhoneNumber, PhoneNumber, Address,
           EnterDate, GraduateDate, ExpulsionDate,
           HighSchool, College4BS, CollegeforMS,
           LeaveStartDate, LeaveEndDate, LeaveReason)
```

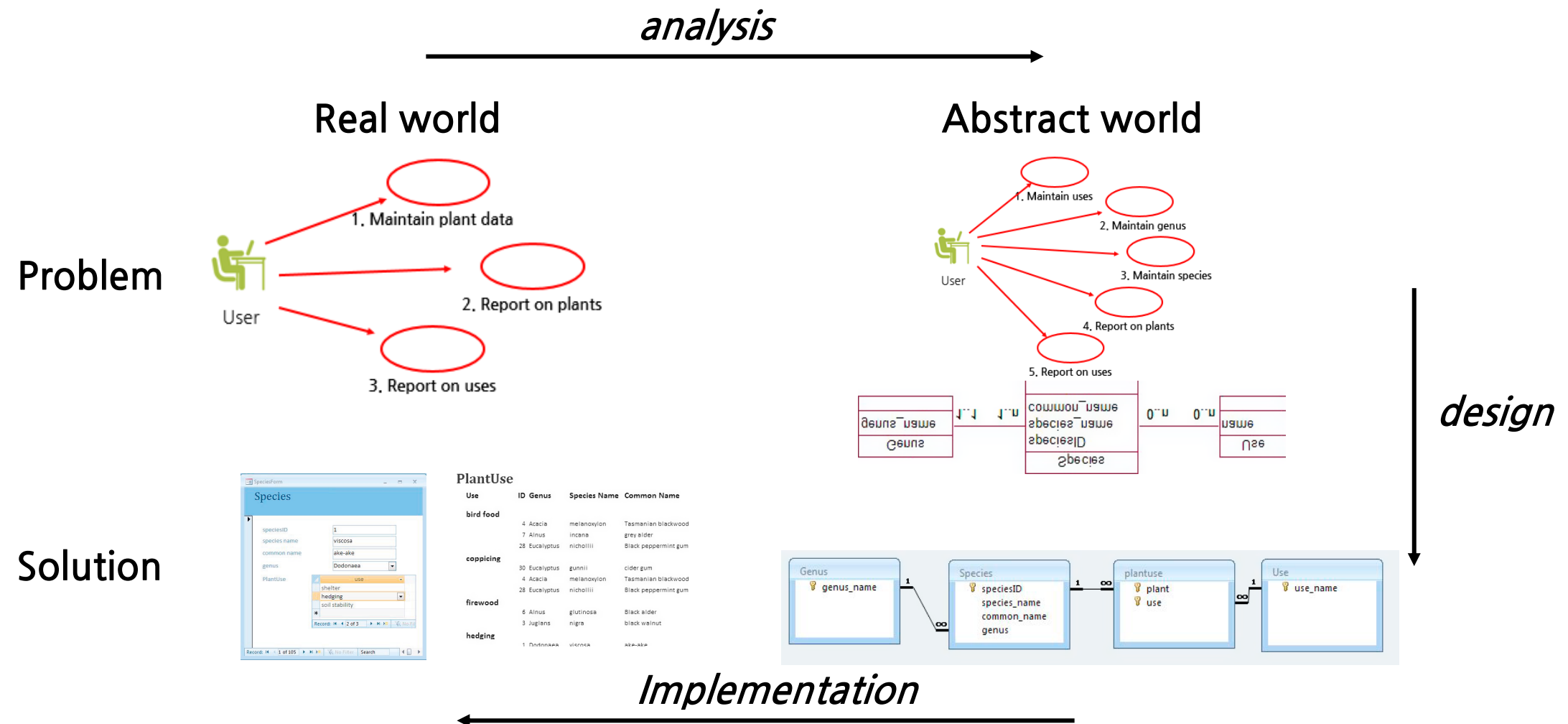
- a. 합격자(입학예정자)의 정보로부터 등록한 학생들을 시스템에 일괄 등록 (입력)
- b. 휴학/휴학연장: 학생이 질병, 사고, 병역, 출산 및 육아, 기타 부득이한 사유로 수강할 수 없을 때에 휴학(연장)을 신청하고 관련 부서 확인을 통해 최종 휴학을 승인
- c. 복학: 휴학기간 종료 및 휴학 사유 소멸 시 다음 학기 등록기간 (2월, 8월) 중 학업을 계속하기 위하여 신청하고 승인
- d. 자퇴: 학생 스스로 학업을 중도 포기할 경우 자퇴를 신청하고 승인
- e. 학적기재 사항 변경관리: 국문성명, 영문성명, 주민등록번호(?), 주소 등 학적 개인정보를 변경하기 위해 신청하는 절차
- f. 졸업(?): 신청 프로그램을 수료하여 학업을 종료할 경우 졸업을 신청하고 승인

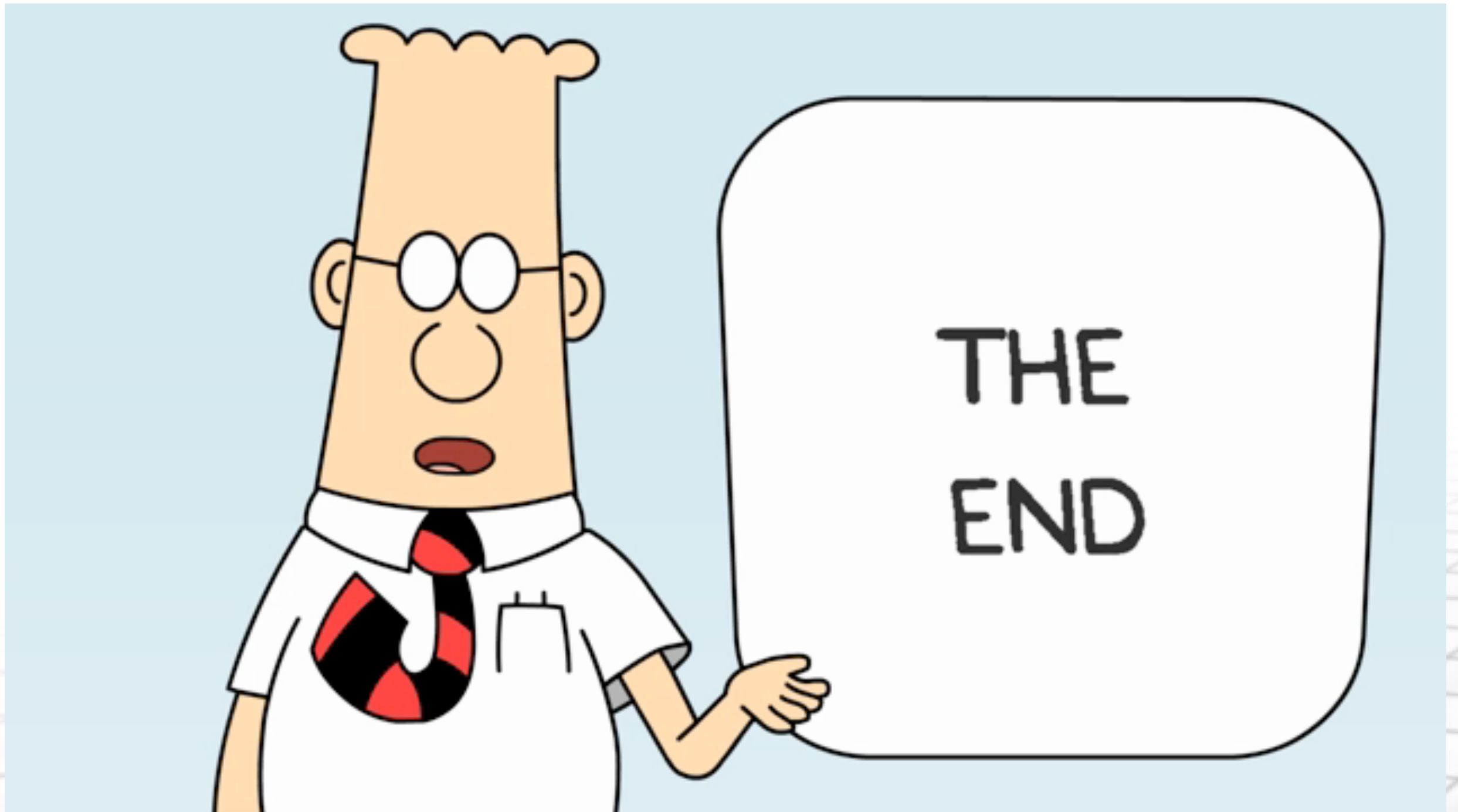
# Summary

1. **사용자 관점**에서 문제를 표현합니다. 데이터베이스 문제의 경우, 일반적으로 **저장**될 데이터와 **검색**해야 하는 정보의 관점에서 이루어집니다.
2. 정보의 가능한 다른 용도와 데이터의 유용한 정렬 순서 또는 그룹화 방법에 대해 고려하여야 합니다. 만족할 때까지 데이터 모델 및 use case를 재검토하는 분석 프로세스를 **반복** 수행합니다.



3. 데이터를 관리할 제품을 선택하고, 적절한 디자인을 작성하십시오. 관계형 데이터베이스의 경우 테이블, 키 및 외래 키를 지정해야 합니다.
4. 응용프로그램을 설계합니다. 관계형 데이터베이스의 경우 테이블 설정 및 use case를 충족시키기 위한 양식 및 보고서 개발을 포함합니다.





출처: metachannels.com

Thank you!