MAHARISHI UNIVERSITY of MANAGEMENT

Engaging the Managing Intelligence of Nature

Computer Science Department

CS401 Modern Programming
Practices (MPP)
Professor Joe Bruen

Lecture 2: Classes

Analysis and Design of Concepts

Wholeness Statement

In this lecture we will focus on finding the core concepts, the principle entities that underlie a specific problem domain.

By watering the root –understanding the core principles of the problem domain– we can enjoy the fruit – create an easy to use, easy to maintain software solution.

Object-Oriented Model of a System

- OO Model should have a 1:1 relationship with a real world problem domain
- Static Model: classes of objects, including attributes, methods, and structural relationships
- Dynamic Model: details of object collaborations and interactions
- Today we focus on the Static (Class) Model

Abstraction:

Finding the Fundamentals

Object-Oriented Analysis

Problem Description

We have been asked to develop an automated Student Registration System (SRS) for the university. This system will enable students to register online for courses each semester, as well as track their progress toward completion of their degree.

When a student first enrolls at the university, he/she uses the SRS to set forth a plan of study as to which courses he/she plans on taking to satisfy a particular degree program, and chooses a faculty advisor. The SRS will verify whether or not the proposed plan of study satisfies the requirements of the degree that the student is seeking.

Once a plan of study has been established, then, during the registration period preceding each semester, students are able to view the schedule of classes online, and choose whichever classes they wish to attend, indicating the preferred section (day of the week and time of day) if the class is offered by more than one professor. The SRS will verify whether or not the student has satisfied the necessary prerequisites for each requested course by referring to the student's online transcript of courses completed and grades received (the student may review his/her transcript online at any time).

Assuming that (a) the prerequisites for the requested course(s) are satisfied, (b) the course(s) meet(s) one of the student's plan of study requirements, and (c) there is room available in each of the class(es), the student is enrolled in the class(es).

If (a) and (b) are satisfied, but (c) is not, the student is placed on a first-come, first-served wait list. If a class/section that he/she was previously waitlisted for becomes available (either because some other student has dropped the class or because the seating capacity for the class has been increased), the student is automatically enrolled in the waitlisted class, and an email message to that effect is sent to the student. It is the student's responsibility to drop the class if it is no longer desired; otherwise, he/she will be billed for the course.

Students may drop a class up to the end of the first week of the semester in which the class is being taught.

Problem Description – In class Exercise

In your small group create a list of all the noun phrases from our problem description.

Problem Description

We have been asked to develop an automated Student Registration System (SRS) for the university. This system will enable students to register online for courses each semester, as well as track their progress toward completion of their degree.

When a student first enrolls at the university, he/she uses the SRS to set forth a plan of study as to which courses he/she plans on taking to satisfy a particular degree program, and chooses a faculty advisor. The SRS will verify whether or not the proposed plan of study satisfies the requirements of the degree that the student is seeking.

Once a plan of study has been established, then, during the registration period preceding each semester, students are able to view the schedule of classes online, and choose whichever classes they wish to attend, indicating the preferred section (day of the week and time of day) if the class is offered by more than one professor. The SRS will verify whether or not the student has satisfied the necessary prerequisites for each requested course by referring to the student's online transcript of courses completed and grades received (the student may review his/her transcript online at any time).

Assuming that (a) the prerequisites for the requested course(s) are satisfied, (b) the course(s) meet(s) one of the student's plan of study requirements, and (c) there is room available in each of the class(es), the student is enrolled in the class(es).

If (a) and (b) are satisfied, but (c) is not, the student is placed on a first-come, first-served wait list. If a class/section that he/she was previously waitlisted for becomes available (either because some other student has dropped the class or because the seating capacity for the class has been increased), the student is automatically enrolled in the waitlisted class, and an email message to that effect is sent to the student. It is the student's responsibility to drop the class if it is no longer desired; otherwise, he/she will be billed for the course.

Students may drop a class up to the end of the first week of the semester in which the class is being taught.

List of Noun Phrases

system
students
courses
semester
progress
completion
degree
student
university
plan of study
courses
degree program
faculty advisor
plan of study
requirements of degree
student
plan of study
registration period

semester
students
schedule of classes
classes
preferred section
day of the week
time of day
class
professor
student
prerequisites
requested course
student
transcript
courses completed
grades received
student
transcript

student
waitlisted class
email message
student
responsibility
class
course
Students
class
end

Noun Phrases (con't)

prerequisites
requested course(s)
course(s)
student
plan of study requirements
room
class(es)
student
class(es)
student
first-come, firstserved wait list
class/section that he/she was previously wait-listed for student
class
seating capacity
class

Sort and Eliminate Duplicates

class
class/section that he/she was previously wait-listed for student
completion
course
courses completed
day of the week
degree
degree program
email message
end
faculty advisor
first-come, firstserved wait list
grades received
plan of study
plan of study requirements
preferred section
prerequisites

Sorted Nouns (con't)

professor
progress
registration period
requested course
requirements of degree
responsibility
room
schedule of classes
seating capacity
semester
student
system
time of day
transcript
university
waitlisted class

Eliminate Terms

- Eliminate terms that do not seem to be objects, such as: 'completion', 'end', 'progress', 'responsibility', 'registration period' and 'requirements of the degree'.

- Optional - eliminate 'university' since there is only one in this case. *If you keep it as a class, what attributes would it have? Is it useful to organize your design?*

- Retain list of eliminated terms, so you can use them later if need be.

Problem Description – In class Exercise – cont.

In your small group create your sorted and reduced list of noun phrases.

Final List of Noun Phrases

class
class/section that he/she was
    previously wait-listed for
course
courses completed
day of the week
degree
degree program
email message
faculty advisor
first-come, firstserved wait list
grades received
plan of study
plan of study requirements

preferred section
prerequisites
professor
requested course
room
schedule of classes
seating capacity
section
semester
student
system
time of day
transcript
waitlisted class

Main Point 1

The OO paradigm is based on the assumption that there is order and intelligence in real world systems, which can be abstracted into a model of objects that collaborate with each other by sending messages back and forth.

Problem Description – In class Exercise –cont.

Create groups of class name synonyms and pick a common class name for each group of synonyms.

Group Synonyms

class
course
waitlisted class
class/section that he/she was
    previously wait-listed for
preferred section
requested course
section
prerequisites

day of the week

degree
degree program

email message

faculty advisor
professor

first-come, firstserved wait list

plan of study
plan of study requirements

room
schedule of classes
seating capacity
semester
student
system
time of day

courses completed
grades received
transcript

Choose Class Name

class
course
waitlisted class
class/section that he/she was
previously wait-listed for
preferred section
requested course
section
prerequisites

Avoid choosing nouns that imply roles between objects. For example, "prerequisite" is a role in an association between two courses. "Waitlisted class" is a role in an association between a student and a course. "Preferred section" is a role in an association between a student and a course.

Choose Class Name (con't)

first-come, firstserved wait list

plan of study
plan of study requirements

room
schedule of classes
seating capacity
semester
student
system
time of day

courses completed
grades received
transcript

day of the week
degree
degree program
email message
faculty advisor
professor

Choose Class Name (con't)

courses completed
grades received
transcript

The notion of "transcript" includes "courses completed" and "grades received" although they are not actually synonyms.

Tests for a Class

- Is the class well defined?
- Are there any attributes for this class?
- Are there any services that would be expected of objects in this class?
- Can this item simply be included as an attribute of another class?

Types of Classes

- Domain Classes: abstractions that the end user will recognize and that represent real-world entities.
- Implementation Classes: introduced solely behind the scenes to hold the application together (example: a dictionary to look up students based on ID number).
- Retain only Domain Classes.

Final List of Classes

course

plan of study

professor

section

student

transcript

Problem Description – In class Exercise – cont.

In your small group create a definition for each of the final list of classes:

- course

- plan of study

- professor

- section

- student

- transcript

Data Dictionary of Classes

- **Course**: a semester-long series of lectures, assignments, exams, etc. that all relate to a particular subject area, and which are typically associated with a particular number of credit hours; a unit of study toward a degree. For example, 'Software Engineering' is a required **course** for the Master of Science Degree in Computer Science.

- **Plan of Study**: a list of the **courses** that a student intends to take to fulfill the **course** requirements for a particular degree.

Data Dictionary (con't)

- **Professor**: a member of the faculty who teaches **sections** and/or advises **students**.

- **Section**: the offering of a particular **course** during a particular semester on a particular day of the week and at a particular time of day (for example, **course** 'Software Engineering' is taught in the Spring 2012 semester on Mondays from 1:00 – 3:00 PM).

- **Student**: a person who is currently enrolled at the university and who is eligible to register for one or more **sections**.

Data Dictionary (con't)

- **Transcript**: a record of all of the **courses** taken to date by a particular **student** at this university, including which semester each **course** was taken in, the grade received, and the credits granted for the **course**, as well as reflecting an overall total number of credits earned and the **student's** grade point average (GPA).

Main Point 2

Software is by its nature complex, and the only way to manage this complexity is through abstraction. Every person can manage all complexity by transcending to the most abstract field, which manages all complexity in nature.

Modeling

Object Oriented Design

Concepts modeled as classes

Unified Modeling Language (UML) Class Diagram

| |
|---|
| Class name goes here |
| Attributes compartment: a list of attribute definitions goes here |
| Operators compartment: a list of operation definitions goes here |

Modeling– In class Exercise

Create a class diagram for Student.

Look back at our problem description and your group's definition of a student.

What attributes do we need?

What operations do we need?

Student Class Diagram

Student
name ssn birthdate gpa
registerforCourse( ) dropCourse( ) chooseMajor( )
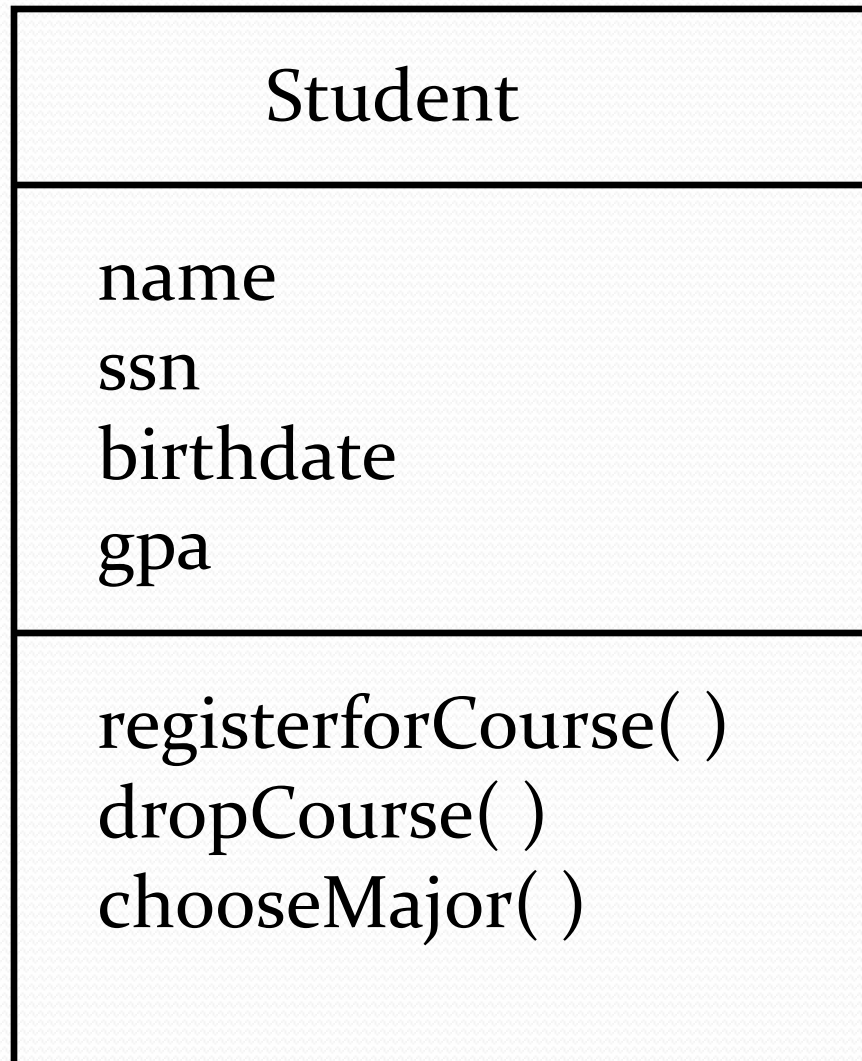
Identifying Attributes

- Use requirements to find attributes of domain classes
- Use your prior knowledge of the domain to help find attributes (e.g. each student has an ID number)
- <u>Talk to the domain expert</u> (often you're not the expert)
- Examine old SRS system already in use to find attributes

Making Items Attributes

- The following items can all be included as attributes of the "Section" class: day of the week, semester, time of the day, room, seating capacity.
- The item "major" could be another attribute of Student class.

Identifying Operations

- Most operations are added during design when looking at implementing specific functionality

- Operations that are added earlier are often just computed attributes. E.g. getAge() when a class has a birthdate attribute

Summary

Today we looked at the analysis and design of concepts as classes

- We saw that with OO we try to create a model that is an abstraction of the real world

- Through abstraction we can manage the complexity of large software systems

- Concept are modeled as Classes, class diagrams have name, attribute and operation comportments.

Connecting the Parts of Knowledge With the Wholeness of Knowledge

1. Class diagrams display concepts and their relationships

2. Class diagrams model an (abstract) representation of a specific real word problem domain.

3. **Transcendental Consciousness** is the simplest state of awareness, where the mind goes beyond thoughts and concepts to the most abstract level of awareness.

4. **Wholeness moving within itself**: in Unity Consciousness one experiences that all objects in the universe arise from consciousness and are ultimately nothing but consciousness.