

# Data Modeling

- Data modeling is a process used to define and analyze data requirements needed to support the business processes within the scope of corresponding information systems in organizations.
  - Needs business experts
  - Professional Data Analysts
  - Other stakeholder

# Data Models

- Data Stores in a DFD display “data at rest”
  - Does not show the “natural structure of data”
  - Does not show the definition, structure, and relationships within the data
- Data Models are used to develop these descriptions

# Type of Models

- During Early Analysis Stage:
  - Conceptual Data Model
    - A set of technology independent specifications about data
    - Used to discuss initial data requirements with the business stakeholders
- During Early Design Stage:
  - Logical Data Model
    - Translation of the conceptual data model to logical model
    - Documents the structure of the data that can be implemented in database
- During Late Design Stage
  - Physical Data Model
    - Translation of logical model to database table

# What is a Conceptual Data Model?

- It is an abstract graphical presentation of data entities within a system, serving as tool.
  - to diagram the major concepts and relationships.
  - to analysis the systems database and data management **requirements**
  - It may assist the database designers, but :
    - may not include all physical table
    - may refer to some entities to facilitate the communication with customers and other stakeholders
    - my refer to some relationships that not necessarily needs to be implemented.
    - Should be independent of any particular commercial database product.

# Conceptual Data Modeling Approaches

- Typically constructed based on DFD diagrams.
- Other Approaches
  - Top-Down Approach
    - derive business rules for a data model from an intimate understanding of the nature of the business
  - Bottom-Up Approach
    - gather information for data modeling by reviewing specific business documents
      - i.e. reports, business forms, etc.

# **What Is ERD During The Analysis?**

# Why ERD During Analysis Phase?

1. Give the analyst a clear, high-level static view of data.
2. In conjunction with data flow diagrams, gives the analyst an alternative logical perspective of the system.
3. It can be even a good starting point for modeling the system, when some of the stakeholder have better understanding of the data rather than system's processes.

# What is ERD?

- Data Models use two main constructs:
  - Data Entities
  - Relationships
- These models known as Entity Relationship Diagrams (ERD) are used as a means of quickly obtaining, with minimum effort, a good sense of the structure of system's database.



# What is an Entity

- Entities are abstract concepts within the data model. Each representing one or more instances.
- Each entity is represented by a box within the ERD.
- Can be visualized as an equivalent to tables in a relational database:
  - each row of the table representing an instance of that entity.



Student

A rectangular box with a blue border and a light gray shadow, containing the word "Student" in black text.

Course

A rectangular box with a blue border and a light gray shadow, containing the word "Course" in black text.

# Characteristics of Data Entities

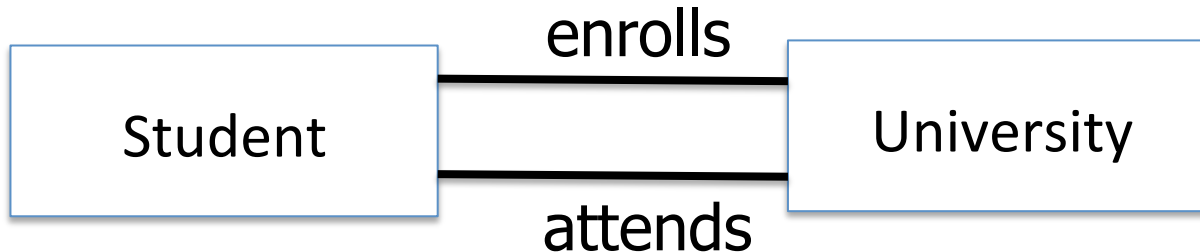
- Names of entities are singular, since the name represents a class or set
- A simple noun is used to name an entity, since an entity is an object
- Names of entities are capitalized
- A rectangle represents an entity on an ERD

Customer

Payment

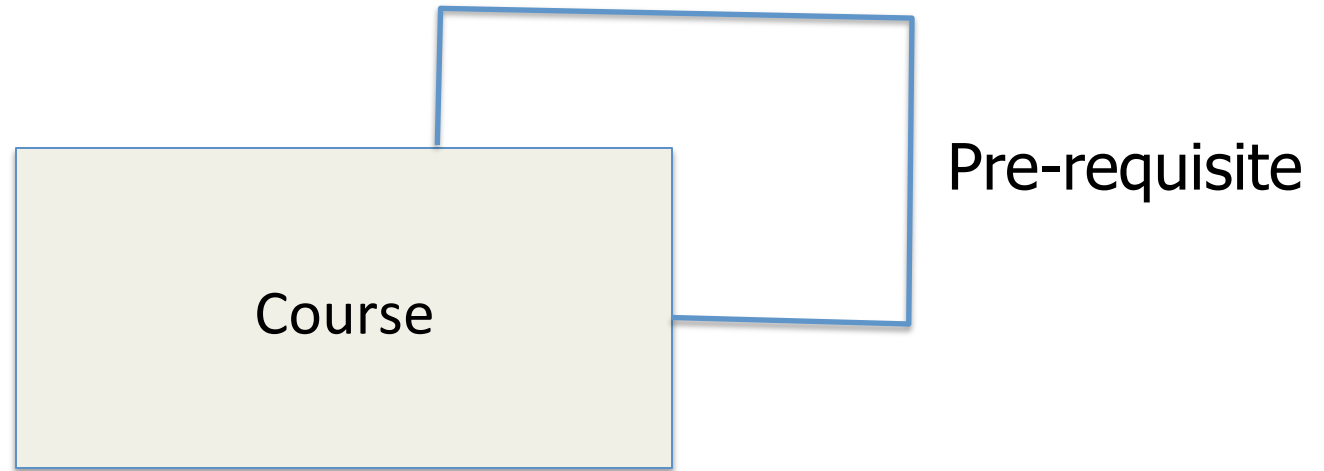
# Relationships

- Each instance of an entity may have a relationship with one or more instances of another entity, and vice versa.
- If necessary, a relationship line may be labeled.



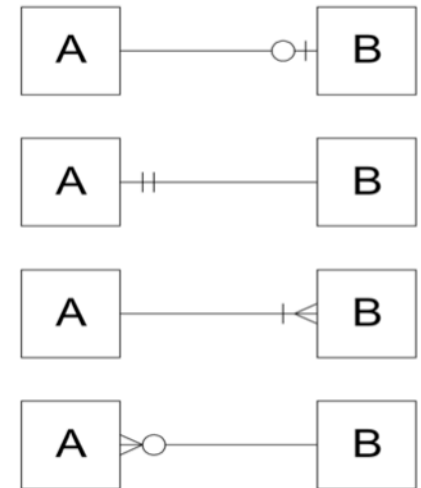
# Recursive Relationships

- an entity can have relationships with itself...

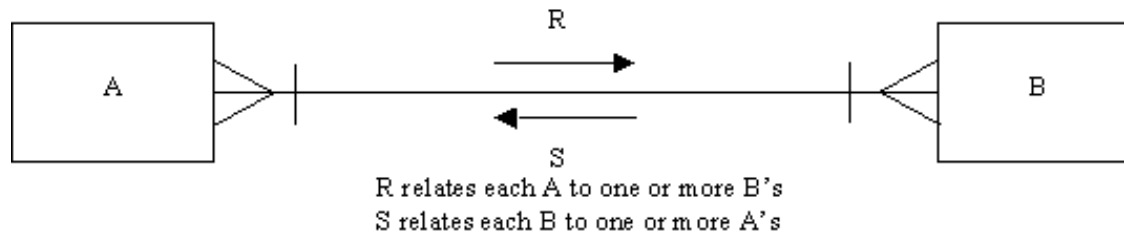
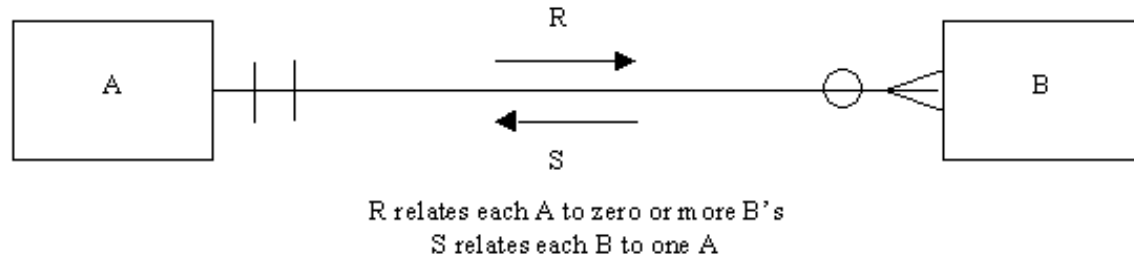


# Optionality and Cardinality

- Symbols at the ends of the relationship lines indicate the optionality and the cardinality of each relationship.
  - “Optionality” expresses whether the relationship is optional or mandatory.
  - “Cardinality” expresses the maximum number of relationships.
  - A circle ( ○ ) indicates optionality of zero.
  - A stroke ( | ) indicate mandatory—the minimum of one.
- A “crows-foot”, indicates a many relationship.

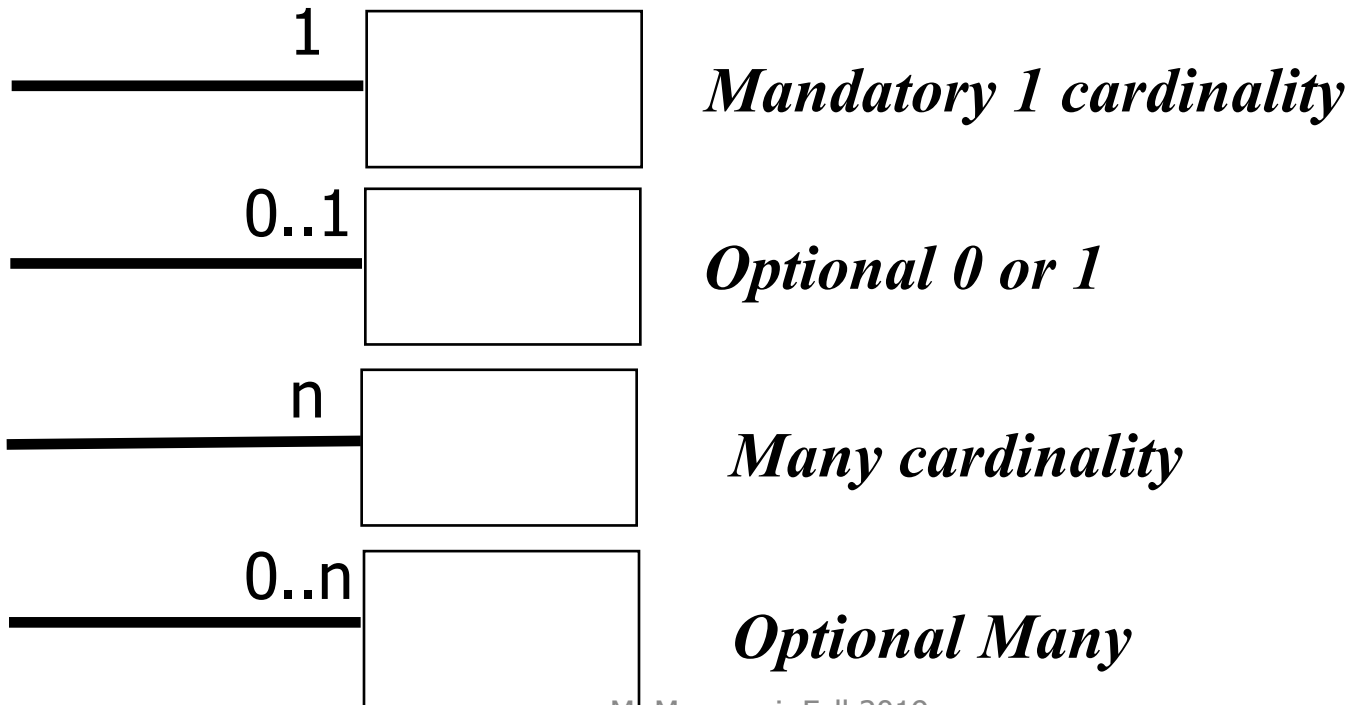


# Optionality Example (using Martin notation)



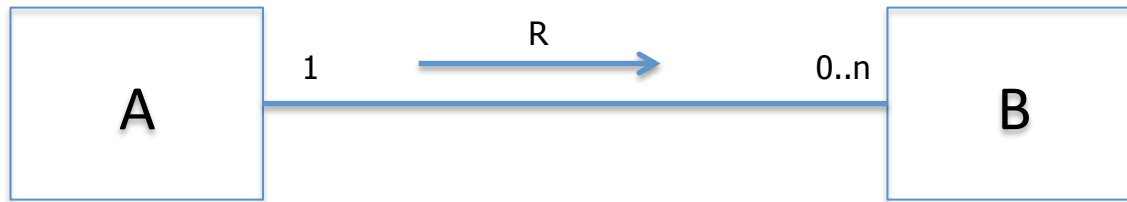
# Cardinality – UML Notation

- Relationship Cardinality Notations Using UML Notation:

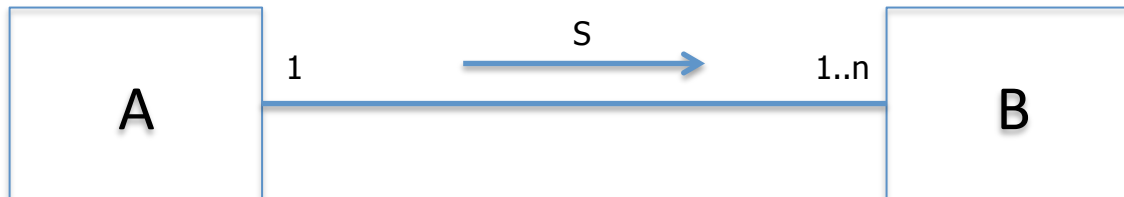


# Optionality Example (UML)

- Optional:
  - R relates each A to zero or more B

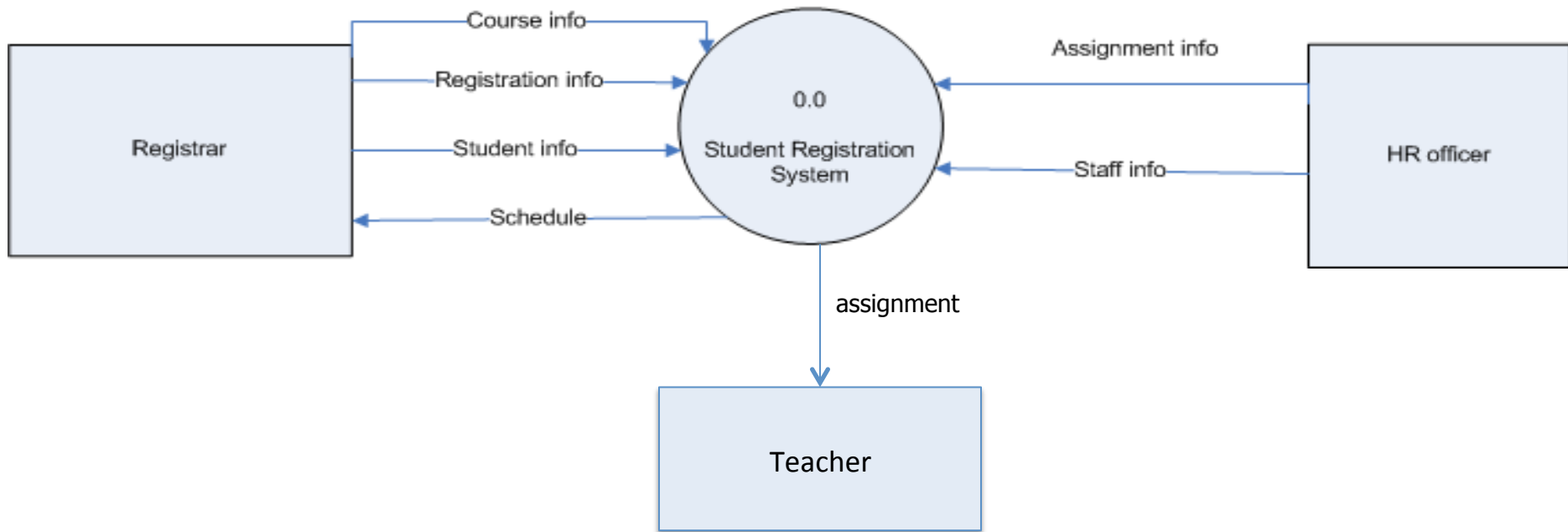


- Mandatory:
  - S relates each A to 1 or more B

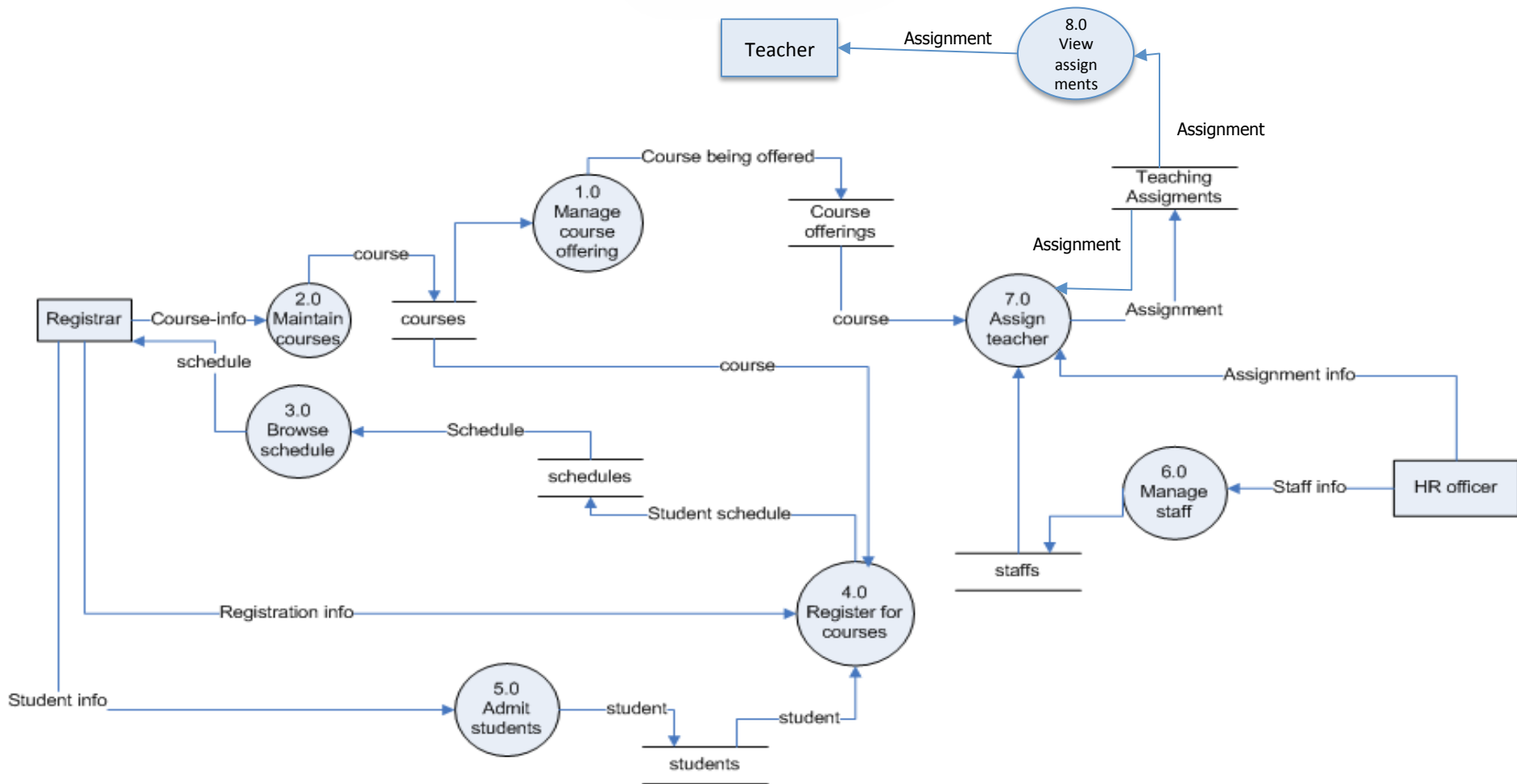




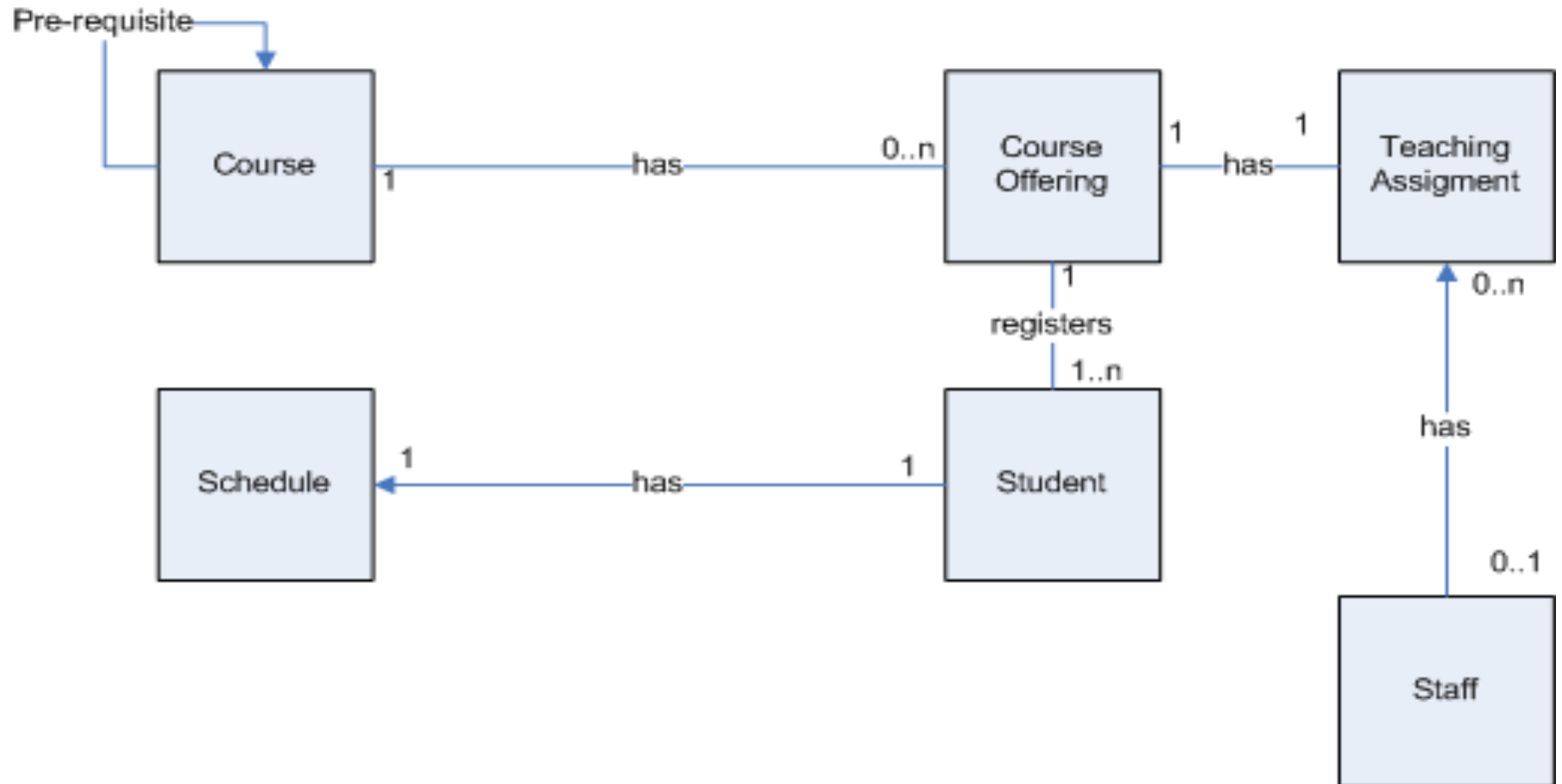
# Context Diagram



# DFD – LEVEL 1



# ERD



# **Data Dictionary**

# The Data Dictionary

- A data dictionary is a collection of definitions for data flows and data stores, in text format.
- A data repository is the same as a data dictionary, with the addition of descriptions for terminators and process specifications.

# Data Dictionary

- A listing of all of the data elements, organized alphabetically.
- Defines each data element in a precise, rigorous manner.
- Describes the:
  - Meaning of all data flows and data stores.
  - Composition of data flows.
  - Composition of stored data.

## Data Dictionary (continued)

- Specifies the values and units of data
- Helps define the relationship between data stores (in conjunction with the ERD).
- Understandable to both client and systems analyst.

# Building a Data Dictionary

Name:	the primary name of the composite data item
Aliases:	other names for the data item
Where used:	data transforms (processes) that use the composite data item
How used:	the role of the data item (input, output, temporary storage, etc.
Description:	a notation for representing content (presented on next slide)
Format:	specific information about data types, pre-set values (if known)



# Data Dictionary Notation

## Notation

## Meaning

=

is composed of

+

and

[ | ]

either-or

{ }<sup>n</sup>

n repetitions of

( ... )

optional data

\* ... text ... \*

delimits a comment

# Template to Use for ENSF 613

- Data Dictionary for Staff

Field	Type	Length	PK	FK	Description
Name	string	30	No	No	Name of staff
Staff_id	string	12	Yes	No	Staff identification number
password	string	14	No	No	Must be combination character and number