Information Technology

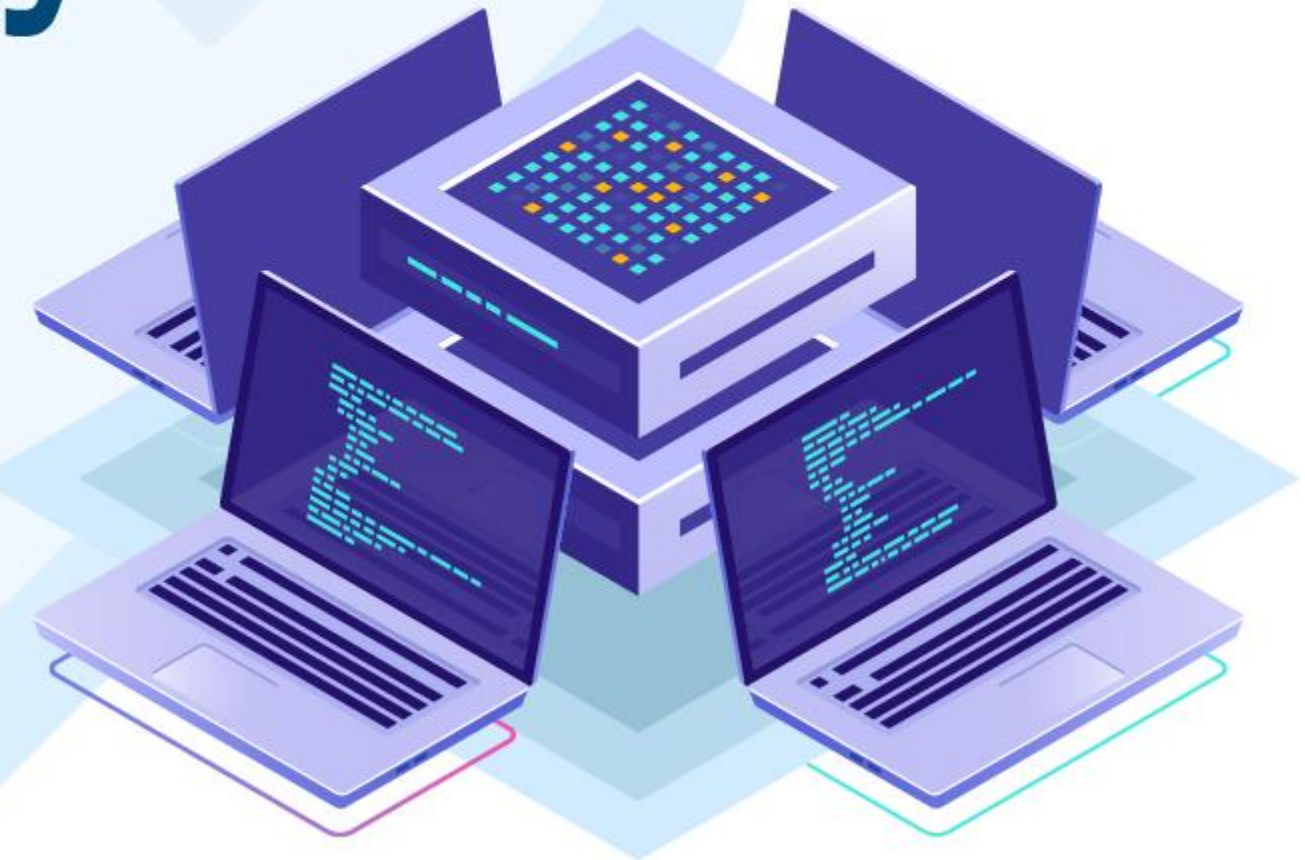
Spring 23-24

Semester 4

- Web Technology

- Dr. Rodaina Abdelsalam

rabdelsalam@eelu.edu.eg



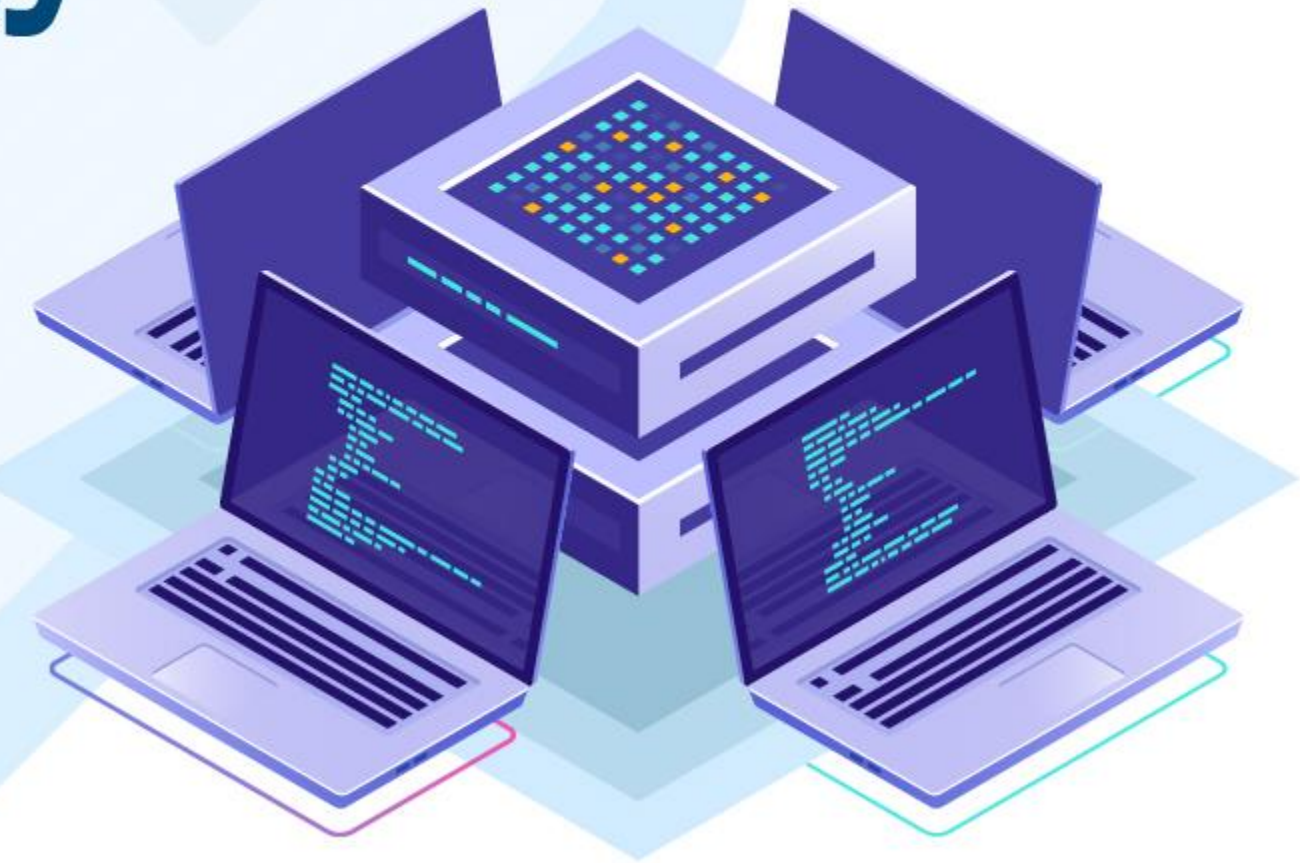
EELU

الجامعة المصرية للتعلم الإلكتروني الأهلية

THE EGYPTIAN E-LEARNING UNIVERSITY

Information Technology

- Introduction to
- internet concepts



EELU

الجامعة المصرية للتعلم الإلكتروني الأهلية

THE EGYPTIAN E-LEARNING UNIVERSITY

How do you think the search engine works?

Searching for Information on the web

- A search engine collects the important terms from each web page on the Web or at least as many pages as it can access.
- Then it creates a master index for the Web that tells which term exists in which pages.
- Finally, it responds to user queries by providing a list of web pages that match the terms he is looking for.
- A search engine ranks the query results delivered to the user based on their popularity not just the presence of the term in them, as we will explain later.
- Thus, the most relevant and popular web pages are displayed first.

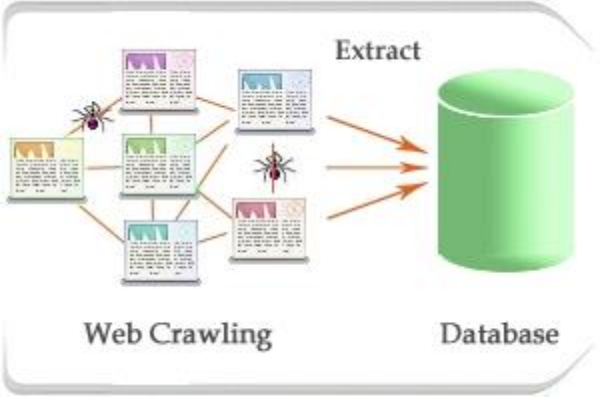
What Search Engines Can Do

- It is not possible for a search engine to go and collect information when the users ask for them.
- Instead, a search engine is continuously working on gathering and indexing information from the Web to be ready to answer queries in a fraction of second.

So, a search engine is...

- So, **search engines** are huge computational tools that operate very sophisticated algorithms on a huge number of powerful computers to provide information in answer to millions of users' queries per minute in numerous languages using the World Wide Web.
- Today's search engine can provide high quality answers in a few seconds or even a fraction of a second and rank the results according to their importance. At the same time, they filter out low-value and automatically produced content.
- Besides searching HTML pages, they can search all sorts of file formats and extensions including pdf, Microsoft Word, Microsoft PowerPoint, postscript and others.
- They can also search for images or videos.

Search Engine Architecture



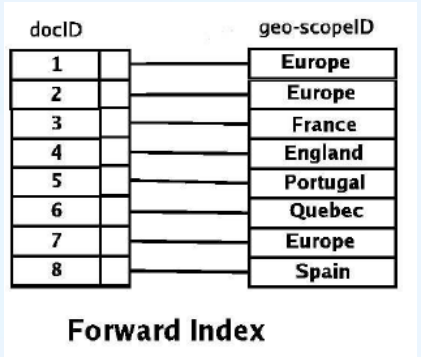
1. **How does the search engine know about everything?**
   - Through **Web crawling,** which is automatically fetching web pages and storing local copies of them.
   - Web crawlers also called **spiders** are special programs that search engines employed to continuously browse the Web in an automated manner.
   - They usually start from a seed list of heavily used servers and very popular pages. When the crawler fetches the page, it scans the contents for links to other URLs and adds each previously unseen URL to the list. Finally, the crawler saves the page content for indexing.
   - web pages vary in their importance and change rate. So, web crawlers maintain a **priority queue** with the most important pages to crawl at the top. So pages' importance is determined by change frequency, incoming link count, click frequency, and so on.
   - Once a URL is crawled, its priority is reassessed and the URL is reinserted in the queue accordingly.

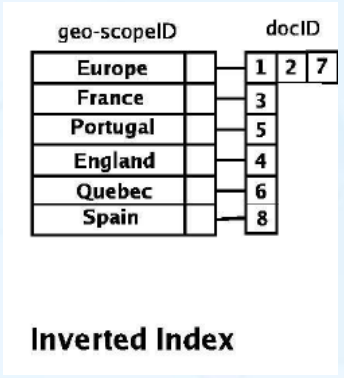
EELU

Search Engine Architecture

2. How does it decide what is relevant?

- Through **indexing,** which is creating a huge searchable index of all terms that were encountered in the fetched web pages.
- This index is the reference for which terms are found where in which documents and with what frequency.
- The first step is **preprocessing, stop words** are identified and removed from the documents. These are words like “a”, “the”, “and”, etc. , also stemming is applied, which is the process of normalizing words by reducing them to the stem or root form, e.g. “technologies” and “technologists” will be reduced to “technology”.
- The second step is **scanning** and creating forward index.
- The third step is **inversion,** in which the forward index is inverted to create an **inverted file index**.





Search Engine Architecture

3. How does it serve the results?

- Through **query processing,** which is responding to users' queries with a list of the most relevant pages ranked from the most important to the least.
- Search engines use hundreds of factors to rank pages like the number of query term occurrences, document length, the number of incoming links, and others.
- The exact ranking factors are business secrets for search engines

Search Engine Architecture

EELU

Outline

- Modeling Web Applications
  1. Content Modeling
  2. Hypertext Modeling
  3. Presentation Modeling

Modeling Web Applications

Levels of Web Application Model

- To model Web applications, there are three levels:
  1. **Content**, i.e., the information and application logics underneath the Web application.
  2. **Hypertext**, i.e., the structuring of the content into nodes and links between these nodes.
  3. **Presentation**, i.e., the user interface or page layout.

Modeling Web Applications

- A clear separation of these three levels allows reuse and helps to reduce complexity.
- For example, we could specify a number of different hypertext structures that will do justice to the specific requirements of different user groups and use devices for a given content.

Modeling Web Applications

Content Model:

- The aim of a content model is the explicit definition of the information structure.
- Comparable to a database schema in data modeling; it eliminates redundancies. This means that the structure of the information will remain unchanged, even if the information itself changes frequently.
- Due to the separation of concerns, content is just modeled once in the content model.

Modeling Web Applications

Hypertext Structure Model:

- To design efficient navigation, content may be offered redundantly on several nodes on the hypertext level.
- So the hypertext structure model just references the corresponding content.
- In this way, users can find this information over several access paths.
- To prevent users from getting lost while navigating and to keep the cognitive stress on users as low as possible, hypertext modeling should rely on recurring navigation patterns.

Modeling Web Applications

Presentation Model:

- In turn, when modeling the presentation level, the focus is on a uniform presentation structure for the pages to achieve a brand recognition effect for the Web application among its users.
- Although the visual appearance of a Web application is of importance, artistic aspects are not within the major focus of modeling.

Modeling Web Applications

- Despite a separation of concerns and the different objectives at the three levels, we would like to map the levels to one another.
- To achieve this mapping between levels, level inter-dependencies have to be captured explicitly. For example, different personalized hypertext access paths could be mapped onto one single content model.
- A comprehensive model of a Web application includes all three levels discussed here, however, the emphasis can vary depending on the type of Web application.
  - Web applications that provide a purely hypertext-oriented user interface to a large data set will probably require the modeling focus to be on content and hypertext structure.
  - In contrast, presentation-oriented Web applications, e.g., corporate portals or online shopping malls, will most likely have larger demands on presentation modeling.

Levels of Modeling

- In any case, the sequence of steps to model the levels should be decided by the modeler.
- Depending on the type of Web application, it should be possible to follow either
  - An information driven approach, that is, starting with content modeling,

  Or
  - A presentation-driven approach, that is, starting with modeling of the application's presentation aspects.

Outline

- Modeling Web Applications
  1. Content Modeling
  2. Hypertext Modeling
  3. Presentation Modeling

Content Modeling

- Modeling the content in the sense of pure data modeling is normally sufficient for static Web applications.
- Complex Web applications additionally require the modeling of behavioral aspects. This means that content modeling includes the creation of the problem domain model, consisting of static and dynamic aspects, as known from traditional Software Engineering.

Content Modeling

- In addition, the following Web application characteristics have to be taken into account:
  - **Document-centric character and multimedia:**
    - It is necessary to take all kinds of different media formats into account when modeling the content, including the structures on which the information is based.
  - **Integration of existing data and software:**
    - Many Web applications are built on existing data repositories and software components, which were not created for Web applications originally.
    - Content modeling has to satisfy two potentially contradicting objectives, i.e., it should cover the content requirements of the Web application to the best possible extent, and it should include existing data structures and software components.

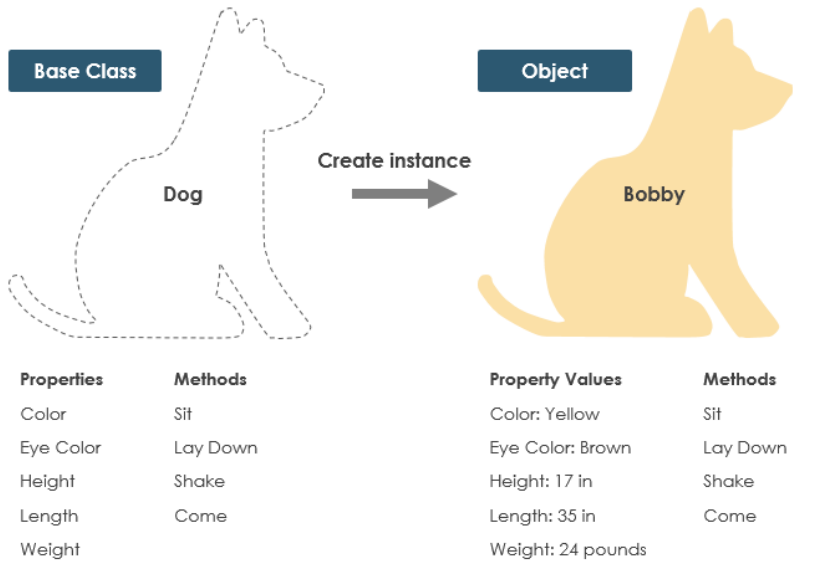
Content Modeling

- Content modeling is aimed at transferring the information and functional requirements determined by requirements engineering to a model.
- Content modeling produces a model that comprises both:
  - the structural aspects of the content, e.g., in the form of a class diagram, and,
  - depending on the type of Web application, the behavioral aspects, e.g., in the form of state and interaction diagrams.

Content Modeling

UML class diagram

- The **UML Class diagram** is a graphical notation used to construct and visualize object oriented systems.
- A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's:
  - Classes; a class describes what an object will be, but it isn't the object itself.
  - their attributes,
  - operations (or methods),
  - and the relationships among objects.

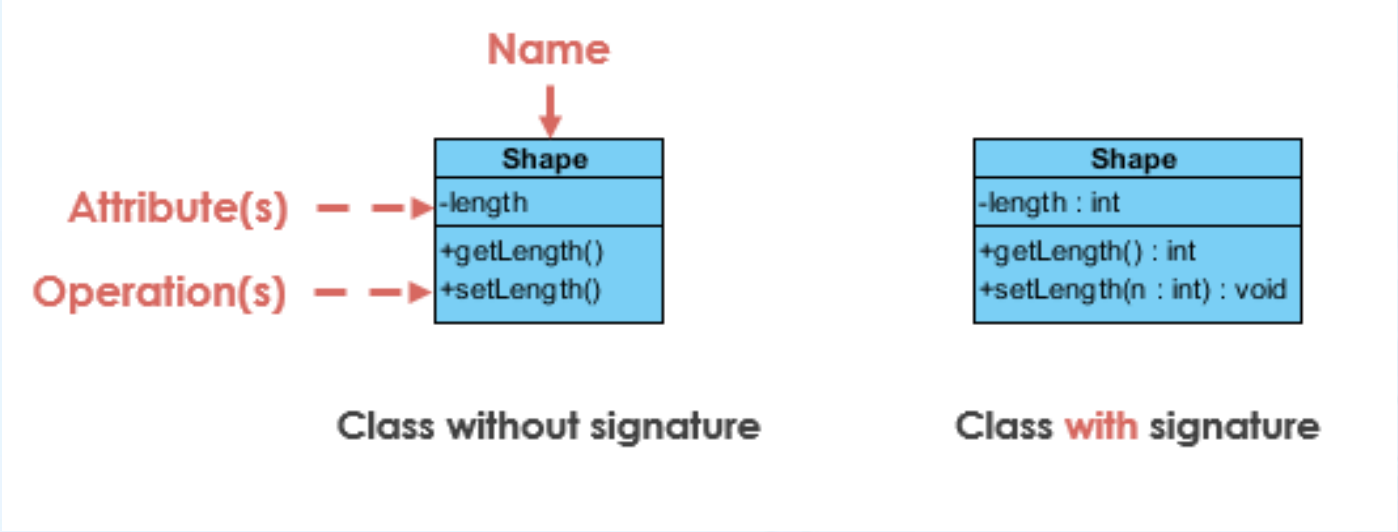


UML =Unified Modeling Language

Content Modeling

UML class notation

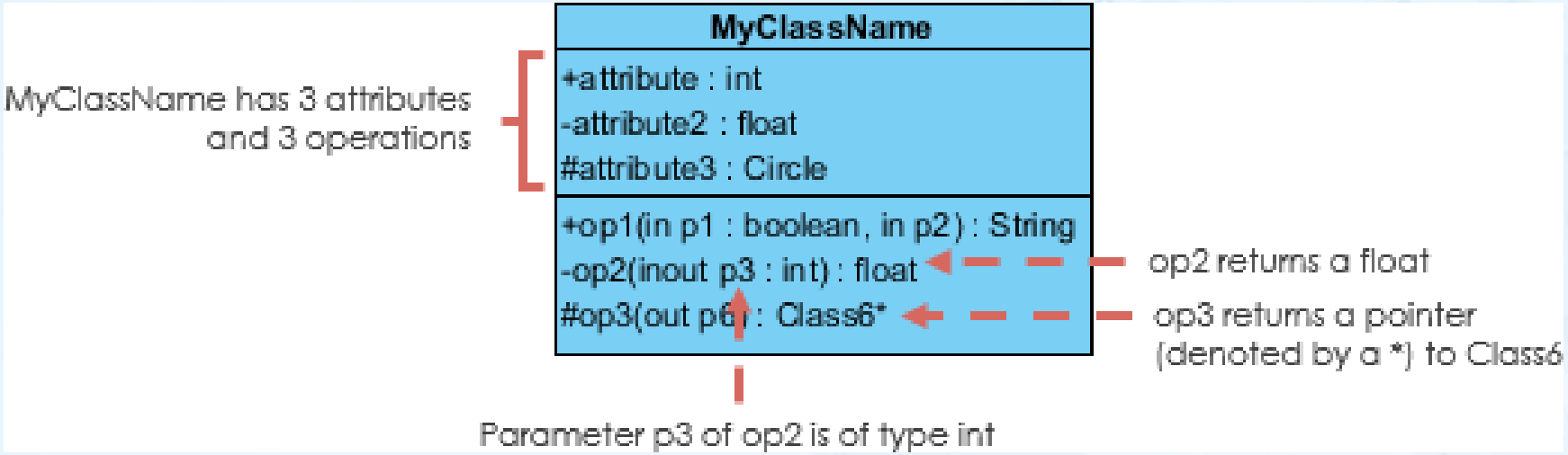
- A class represent a concept which encapsulates state (**attributes**) and behavior (**operations**). Each attribute has a type. Each **operation** has a **signature**. *The class name is the **only mandatory information***.
- **Class Name:** The name of the class appears in the first partition.
- **Class Attributes:** Attributes are shown in the second partition. The attribute type is shown after the colon.



Content Modeling

UML class notation

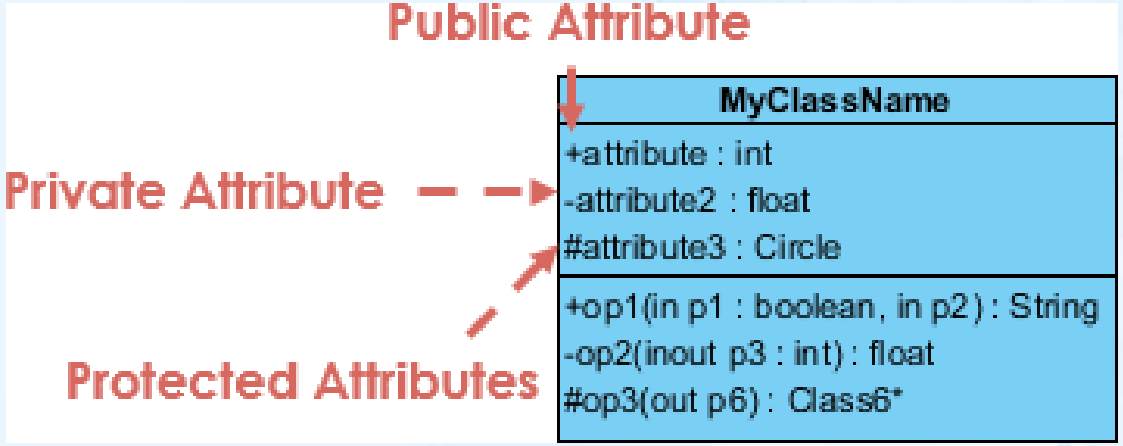
- **Class Operations (Methods):** Operations are shown in the third partition. They are services the class provides.
- The return type of a method is shown after the colon at the end of the method signature.
- The return type of method parameters are shown after the colon following the parameter name.



Content Modeling

UML class notation

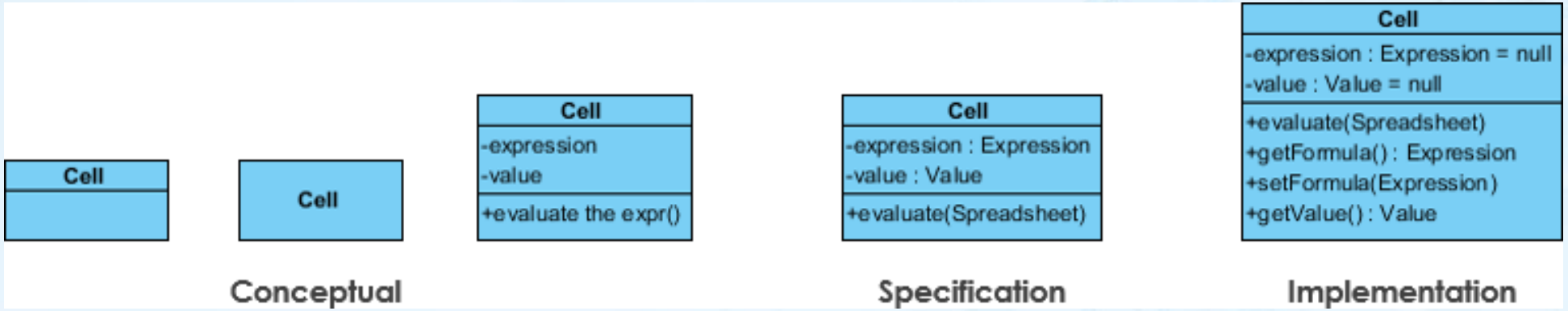
- **Class Visibility**
  - The +, - and # symbols before an attribute and operation name in a class denote the visibility of the attribute and operation.
    - + denotes public attributes or operations
    - - denotes private attributes or operations
    - # denotes protected attributes or operations



Content Modeling

Perspectives of Class Diagram

- A diagram can be interpreted from various perspectives:
  - **Conceptual**: represents the concepts in the domain
  - **Specification**: focus is on the interfaces of Abstract Data Type (ADTs) in the software
  - **Implementation**: describes how classes will implement their interfaces
- As we mentioned before, the class name is the only mandatory information.



Content Modeling

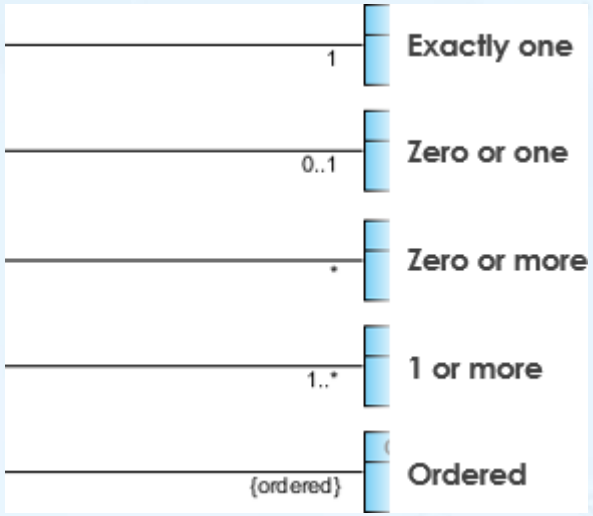
Perspectives of Class Diagram

- The perspective affects the amount of detail to be supplied and the kind of relationships worth presenting.
- The choice of perspective depends on how far along you are in the development process.
  - During the formulation of a **domain model**, for example, you would seldom move past the **conceptual perspective**.
  - **Analysis models** will typically feature a mix of **conceptual and specification perspectives**.
  - **Design model** development will typically start with heavy emphasis on the **specification perspective**, and evolve into the **implementation perspective**.

Content Modeling

Relationships between classes

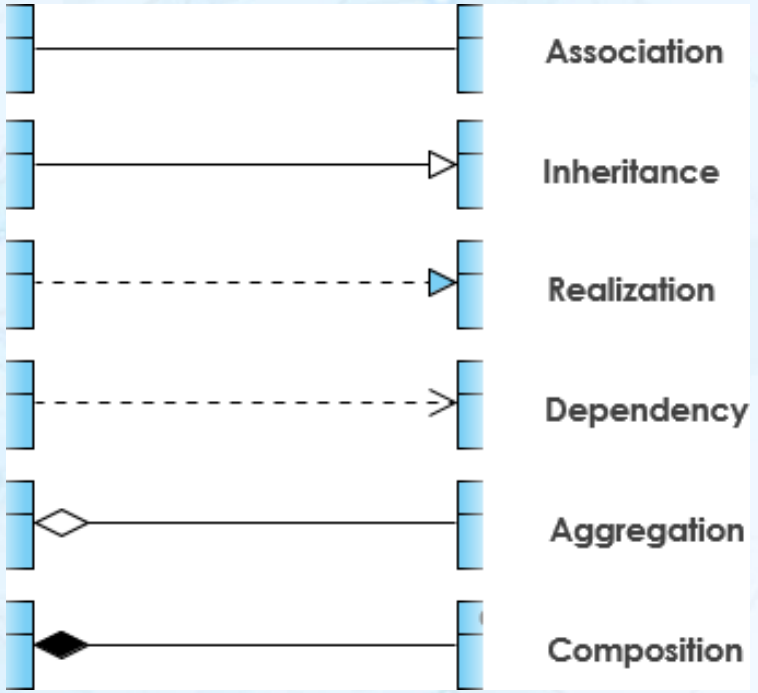
- UML precisely conveys how code should be implemented from diagrams. If precisely interpreted, the implemented code will correctly reflect the intent of the designer.



Content Modeling

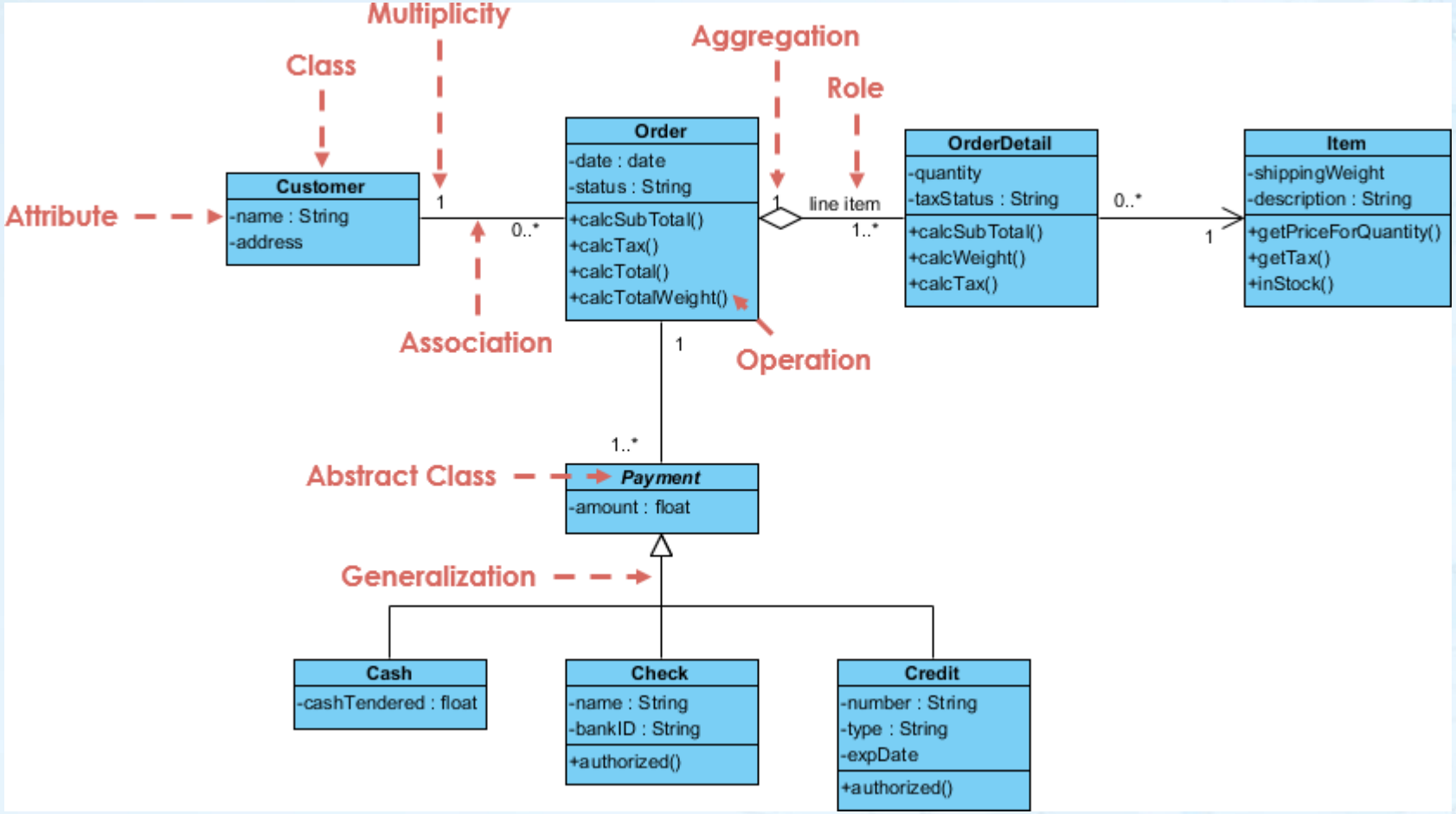
Relationships between classes

- A class may be involved in one or more relationships with other classes. A relationship can be one of the following types:
  - **Association:** A structural link between two peer classes.
  - **Inheritance (or Generalization):** is a taxonomic relationship between a more general classifier and a more specific classifier which inherits the features of the more general classifier. Represents an "is-a" relationship.
  - **Realization:** the relationship between the interface and the implementing class.
  - **Dependency:** A special type of association. Exists between two classes if changes to the definition of one may cause changes to the other (but not the other way around).
  - **Aggregation:** A special type of association. It represents a "part of" relationship.
  - **Composition:** A special type of aggregation where parts are destroyed when the whole is destroyed.



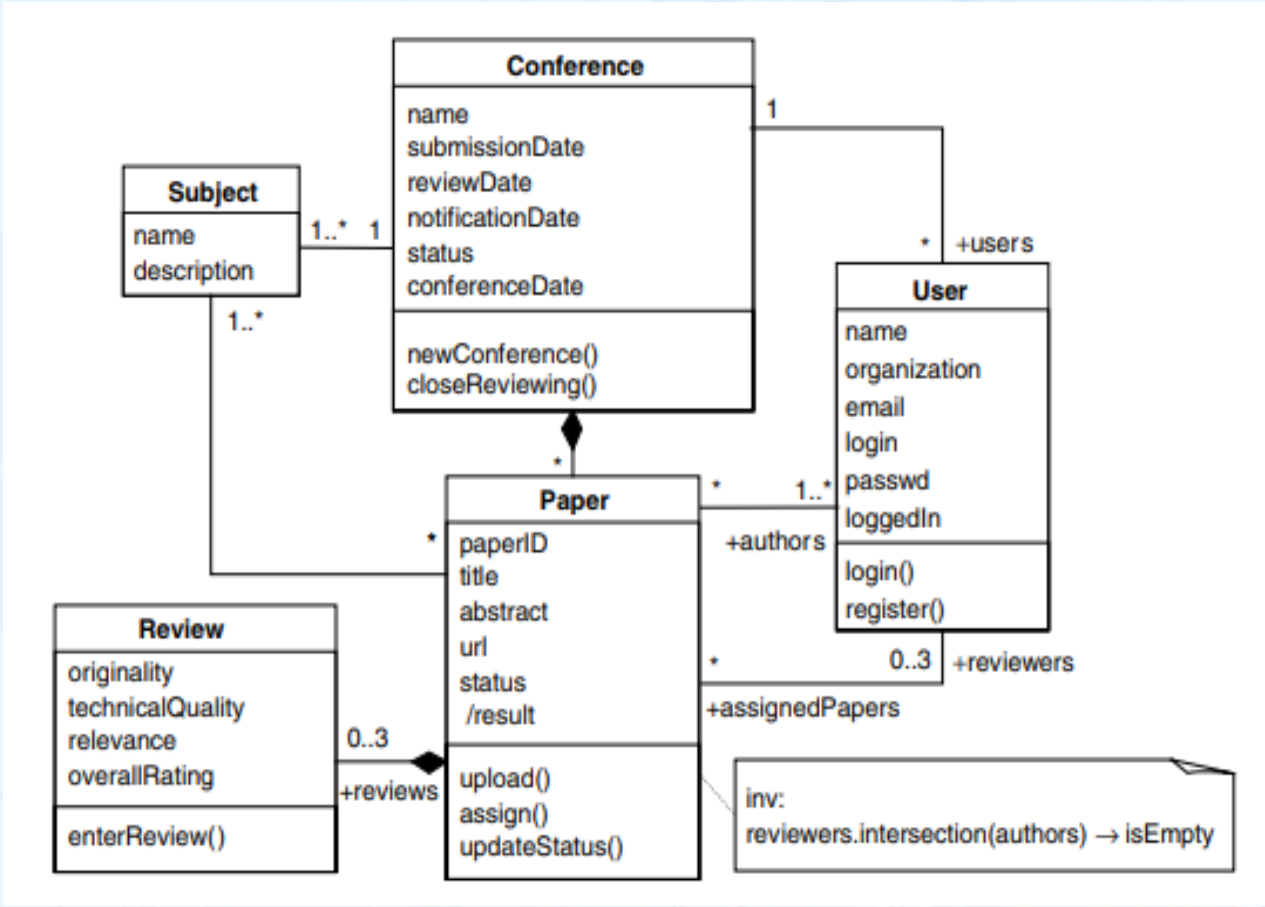
Content Modeling

Class Diagram Example: Order System



Content Modeling :
Simple Example of UML class diagram for the reviewing system.

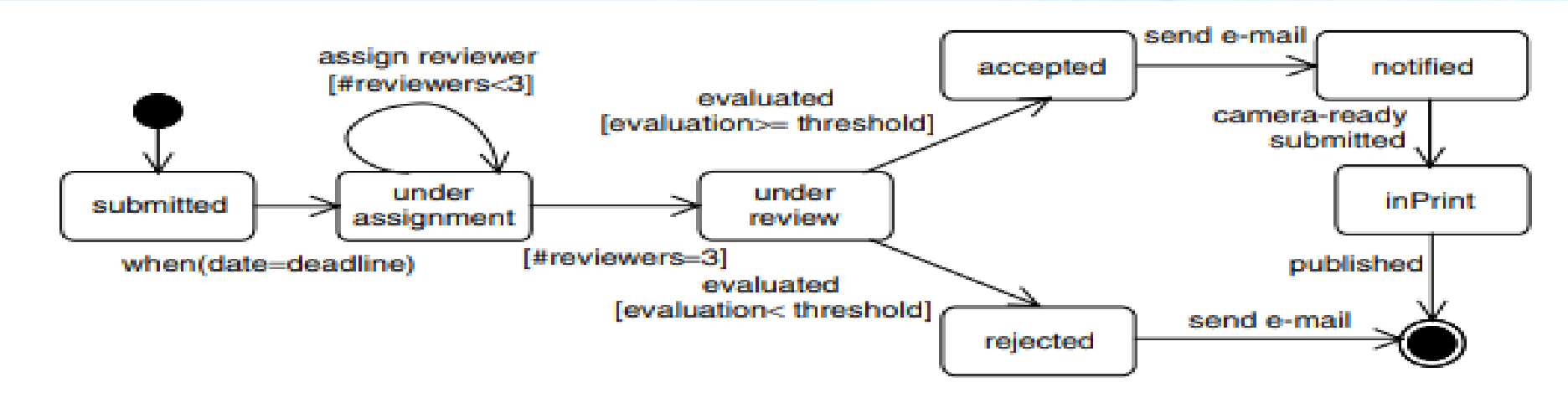
- The diagram models a conference to be held on a number of topics, and users who can sign in to the conference and submit their papers.
- A paper is subject to a review by three reviewers.
- Notice the invariant attached to the class ‘‘Paper’’: it ensures that authors won’t be able to review their own papers.



UML =Unified Modeling Language

Content Modeling :
Example State machine diagram to model the various states of a paper in the reviewing system

- The figure shows that a submitted paper will be assigned to three reviewers for review after the submission deadline has expired.
- If a pre-set threshold value is reached, the paper is accepted; otherwise, it is rejected. In both cases the authors are notified via e-mail about the outcome of the review.
- Finally, an accepted paper will be printed once the final version has been submitted.



EELU

الجامعة المصرية للتعلم الإلكتروني الأهلية

THE EGYPTIAN E-LEARNING UNIVERSITY

THANK YOU FOR WATCHING

QUESTIONS?