CO457 Business Modelling

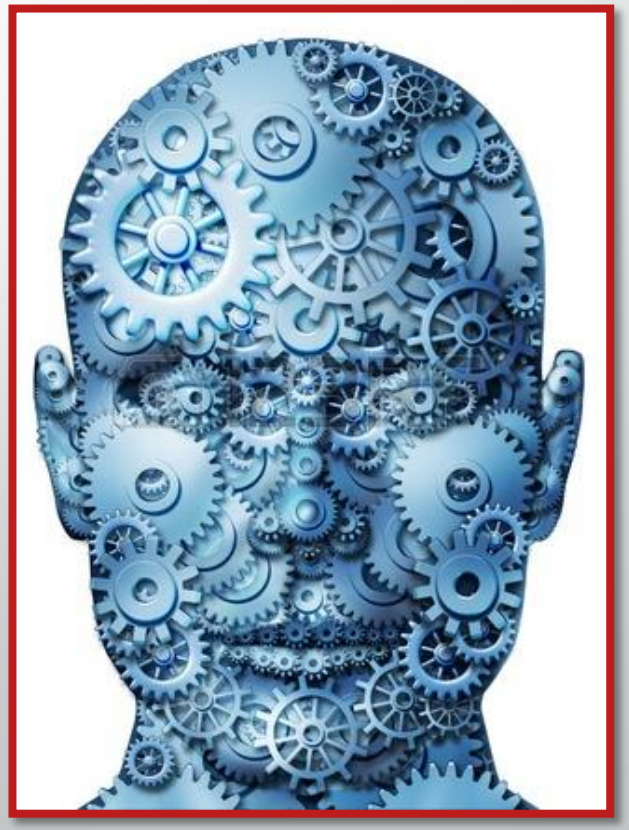
Module Week 4

Scoping Business Functionality

Functional Decomposition

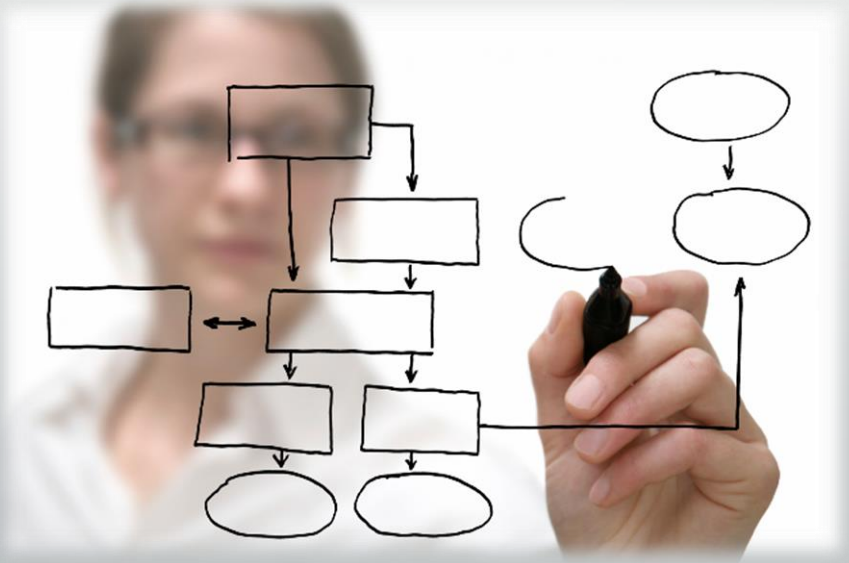
Functional Decomposition

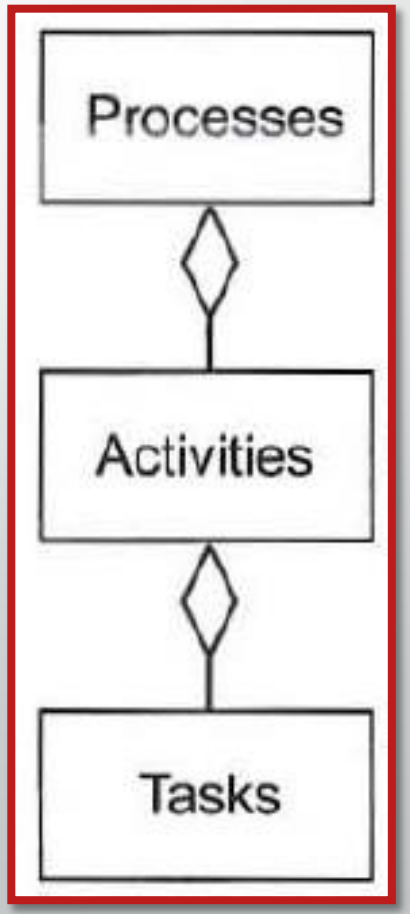
- Business functionality Is modelled at **three levels of detail**
- Functions:
  - Captured in a **functional decomposition**
  - For scoping the **business functions**
  - Spanned by **processes**



Functional Decomposition

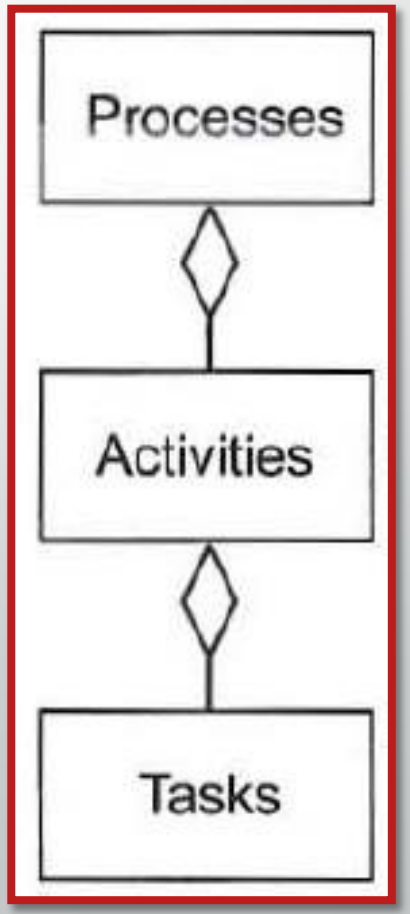
- Business functionality Is modelled at three levels of detail
- **Processes - May cut across several functional areas:**
- Modelled as:
  - **Use cases**
  - **Cross-functional maps**
- **Consist of activities**





Functional Decomposition

- Business functionality Is modelled at three levels of detail
- **Activities**:
  - Described in **process/workflow models**
  - Modelled using BPMN
  - **Consist of tasks**

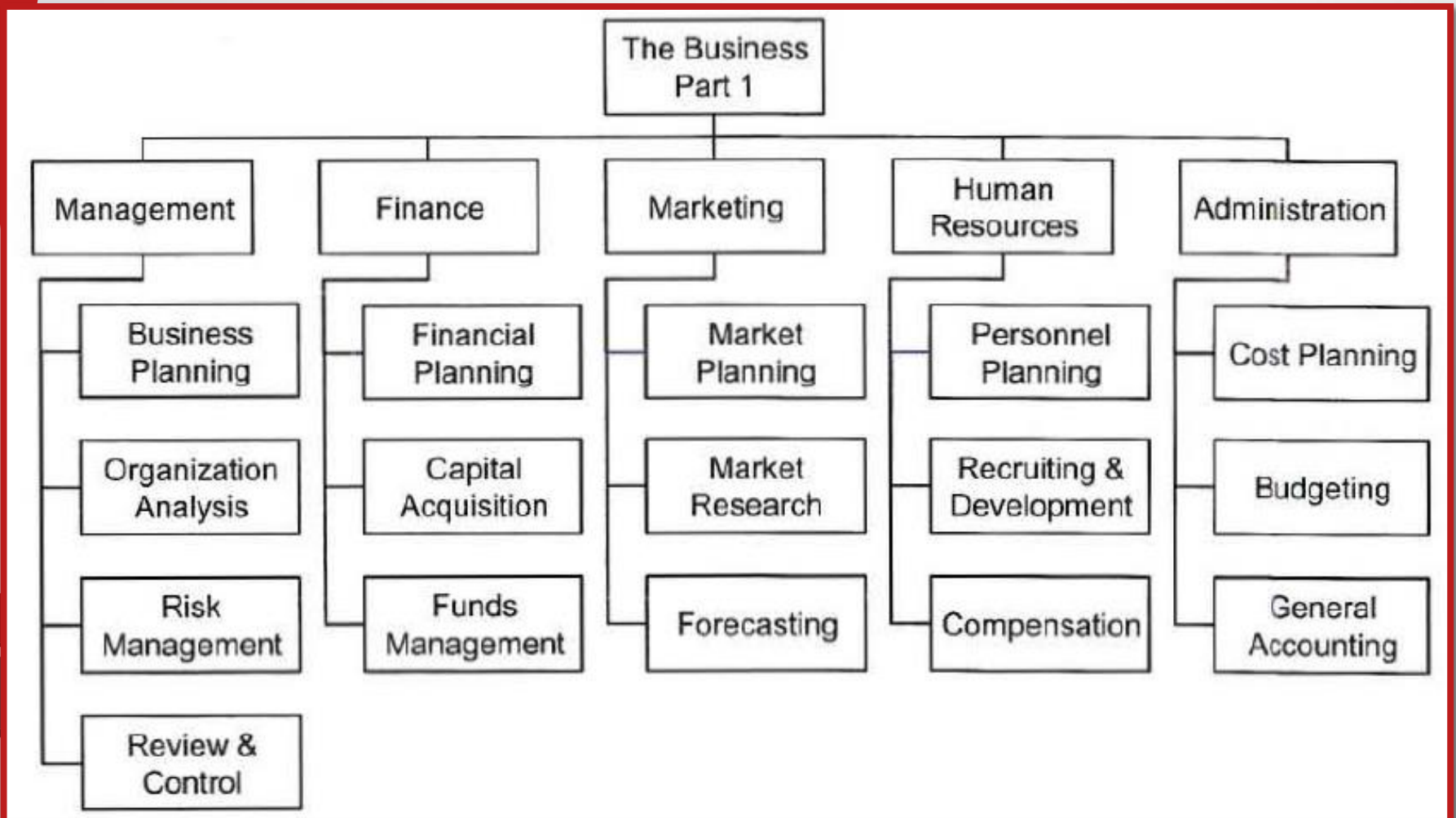


Functional Decomposition

- Functional decomposition breaks down a business into its **functions and sub-functions**
- **Functions**
  - **Describes the types of processes** the business does at the highest level
  - Appear in the **functional/process view** of the business architecture
  - Frequently correspond to a single organisational unit
- **Function names are nouns or noun phrases**
  - For example:
    - Market Research
    - Shipping and Receiving

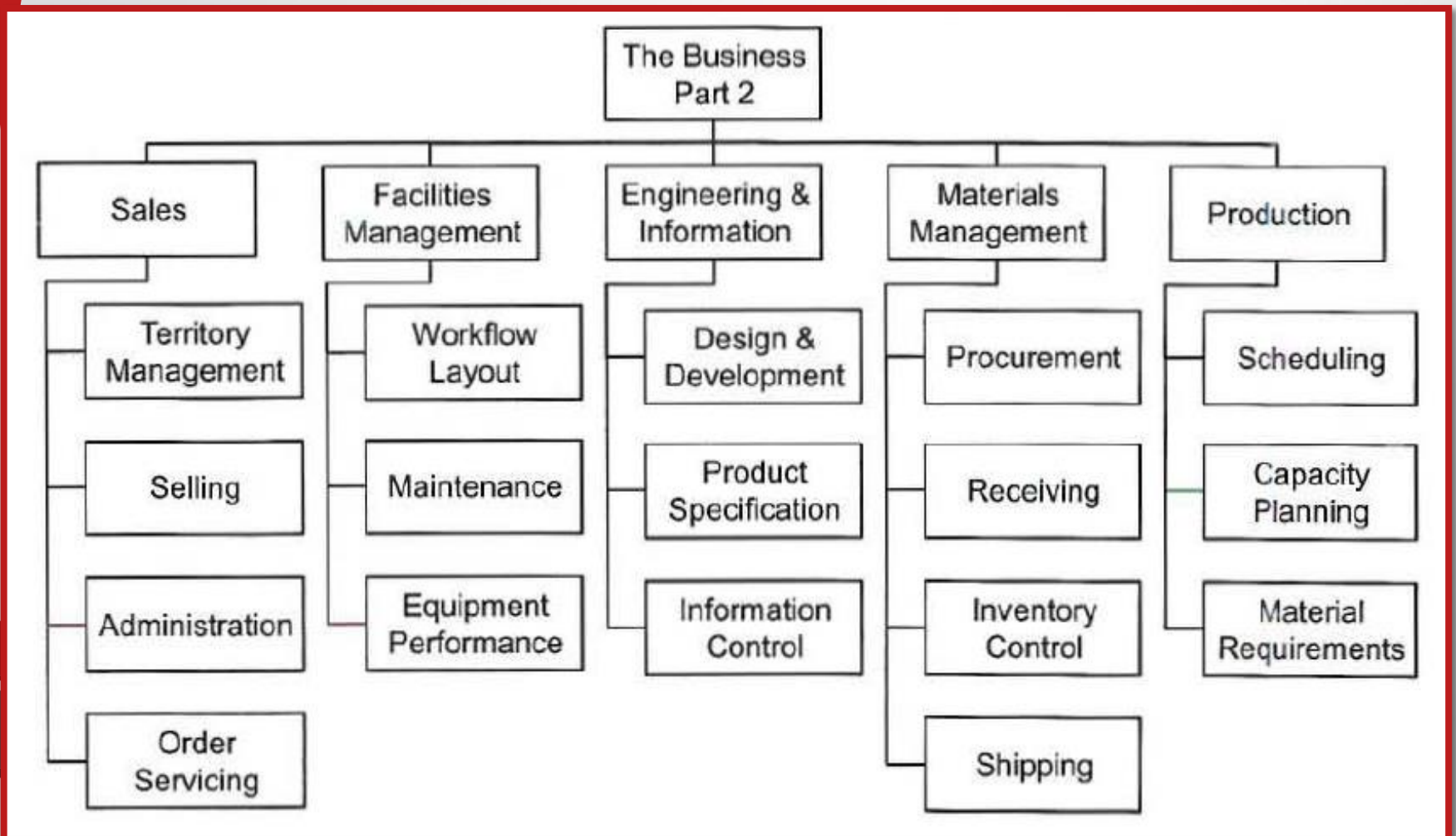
Functional Decomposition Template

- Generic business **functional decomposition** template



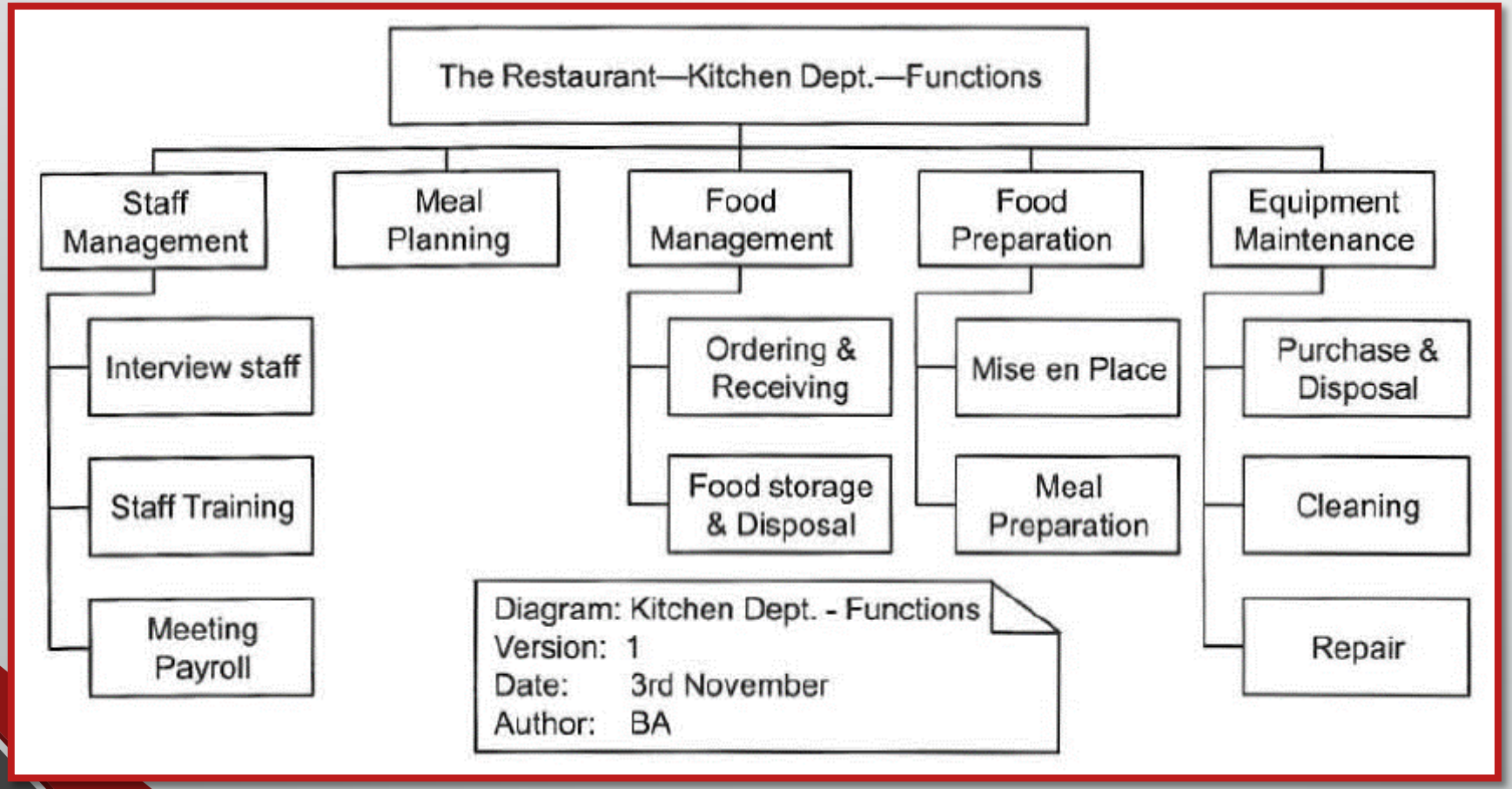
Functional Decomposition Template

- Generic business functional decomposition template



Functional Decomposition: Example

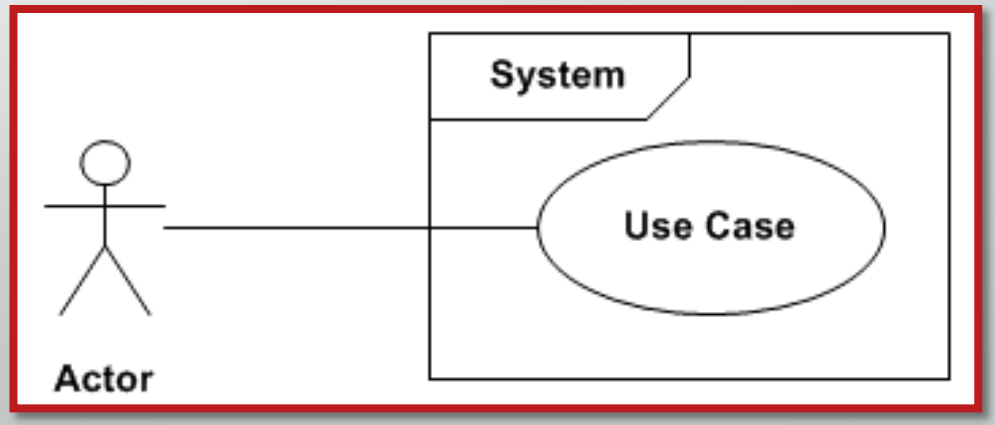
- Kitchen example:



Use-Case Models

Use-Case Models

- A use-case model **describes the processes** performed in a subject
  - **Business function(s)**
- Business use-case models consist of
  - A use-case diagram at two levels of detail
    - **Contextual view**
      - High level
    - **Refined view**
      - Use-case decomposition level

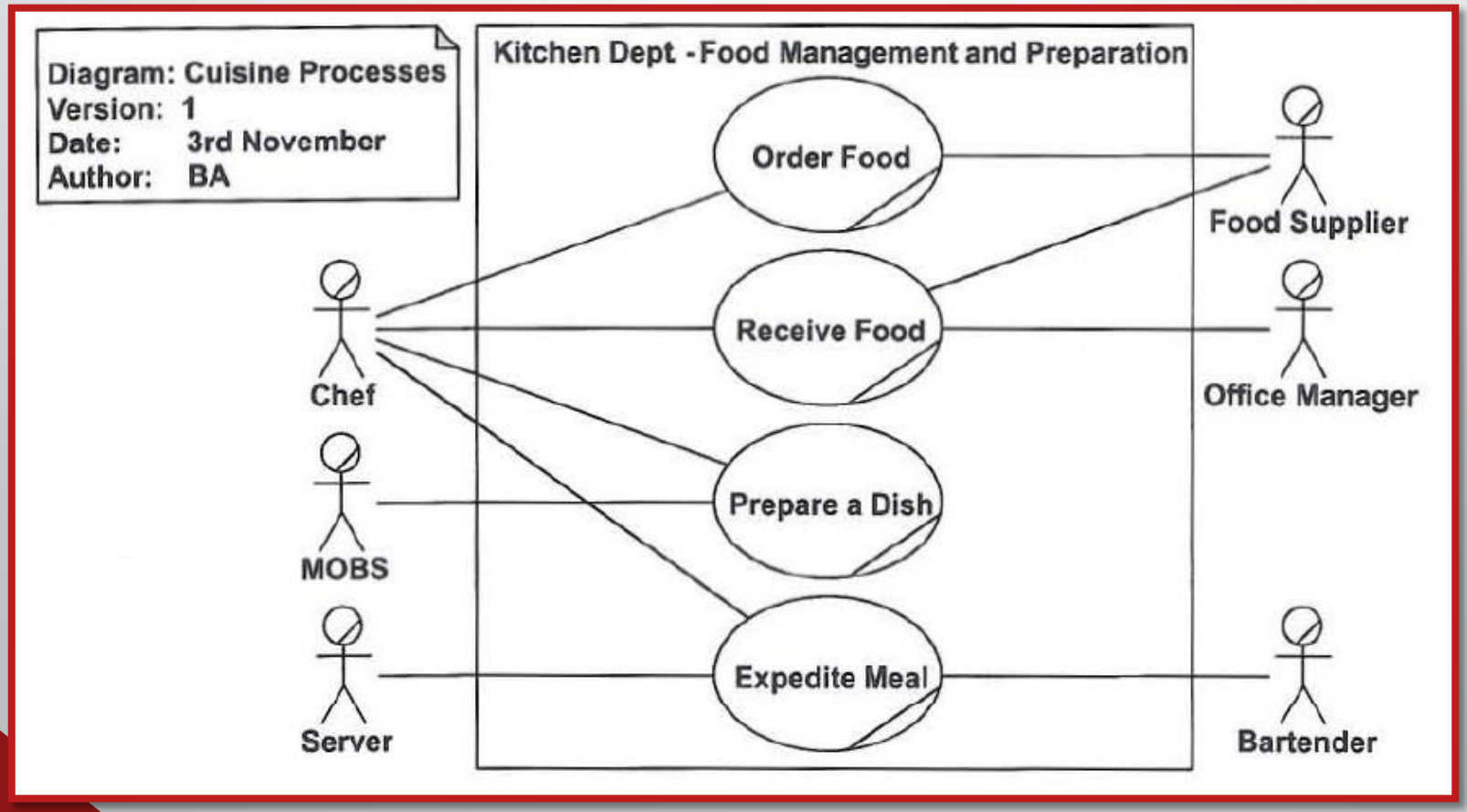


Use-Case Models

- Structured documentation describing the business use case
  - At three levels of detail:
  - **Brief**
  - **Casual**
  - **Fully dressed**

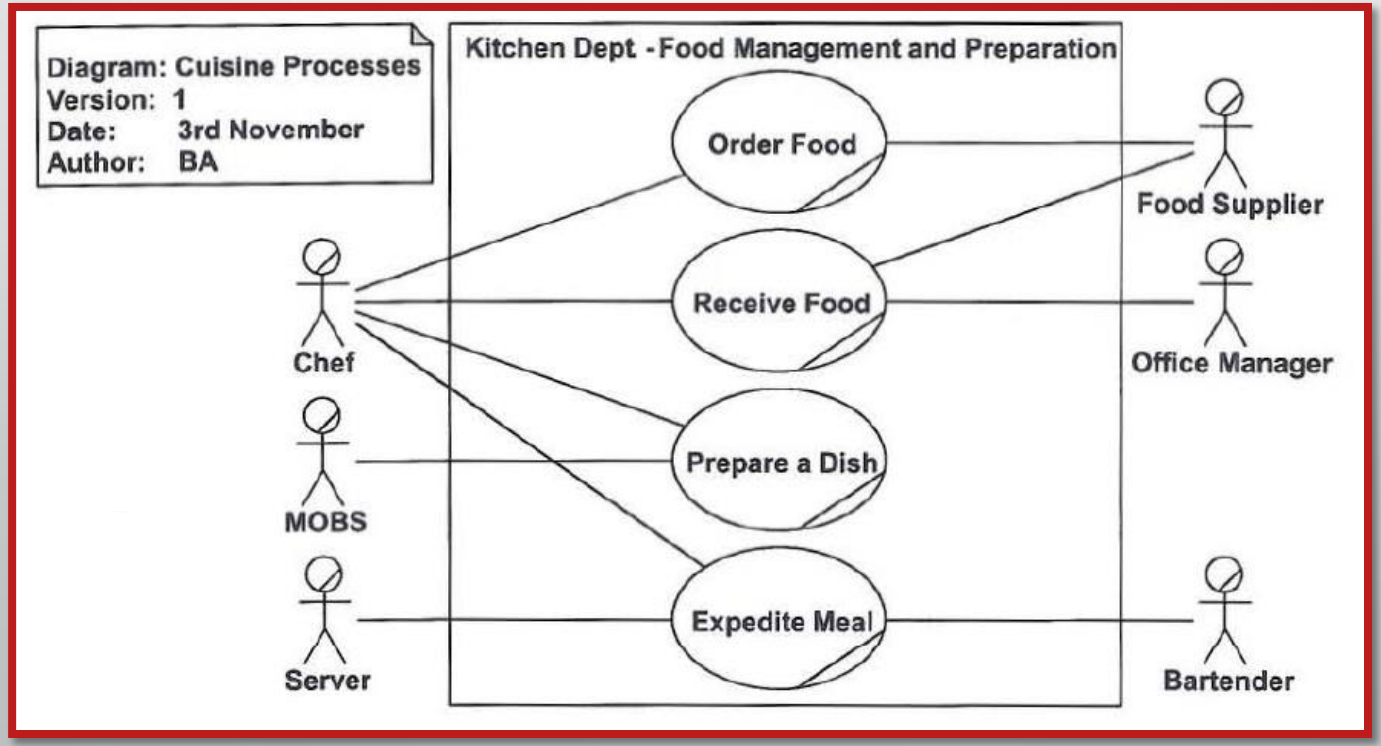
Use-Case Models: Example

- Example: **Context level use-case diagram** for kitchen functionality



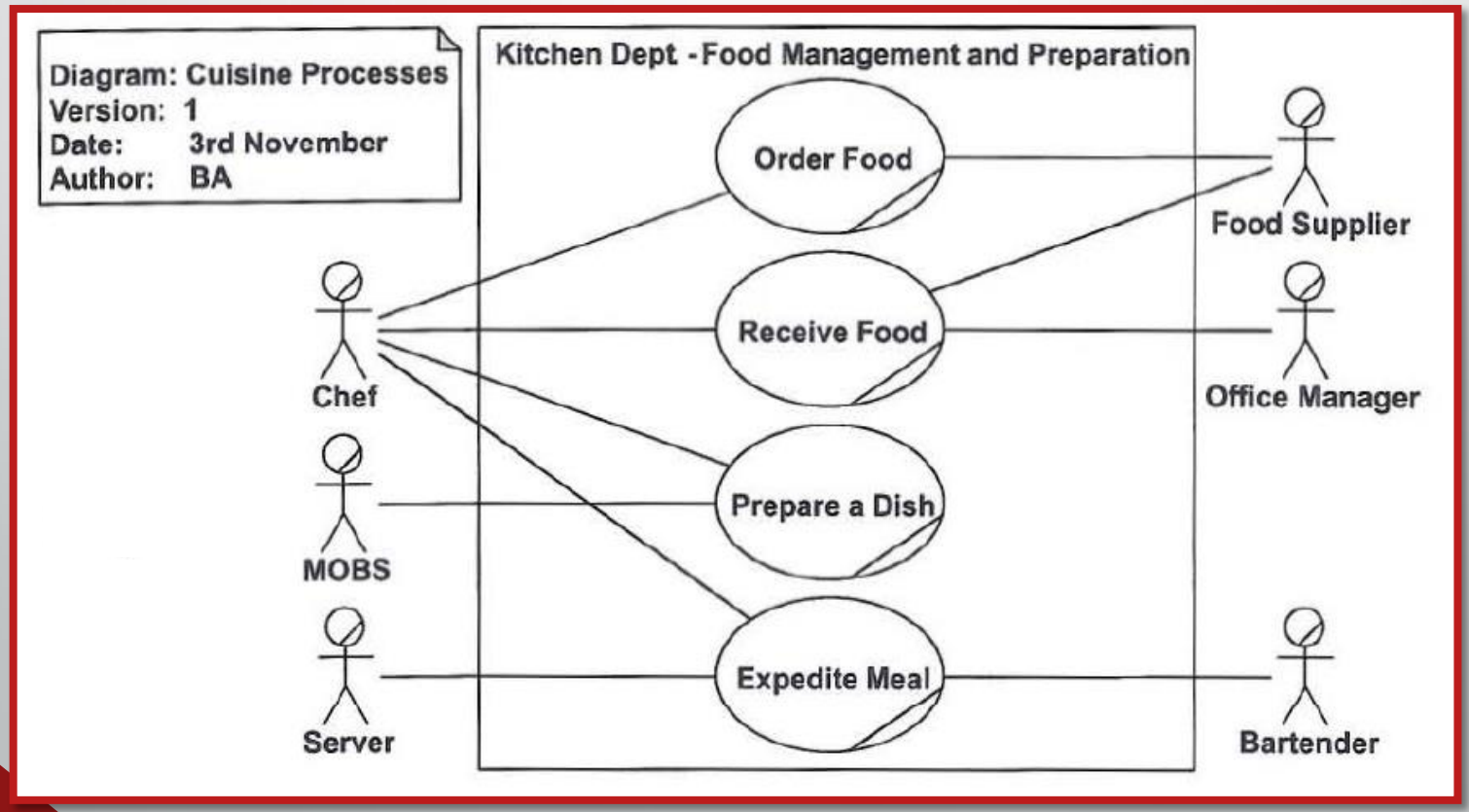
Boundary, Use Cases, and Actors

- The scope being modelled is illustrated with a **subject boundary**
  - Appears as a **rectangle on the diagram**
  - **Kitchen Department - Food Management and Preparation**



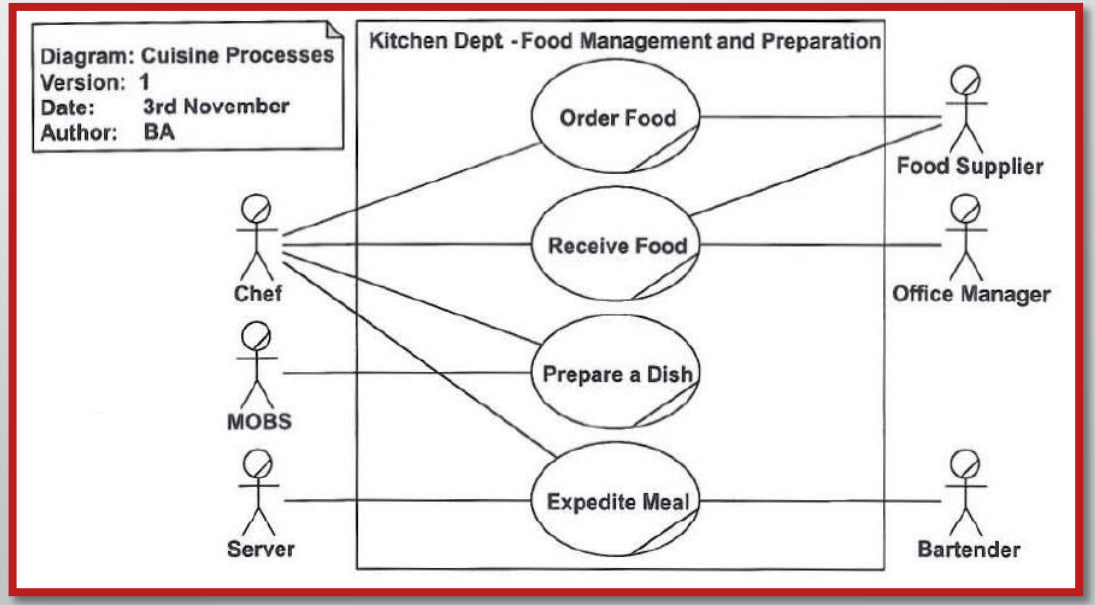
Boundary, Use Cases, and Actors

- Business use cases are the **processes performed in the business function**
  - Appear as **ovals inside the subject boundary**



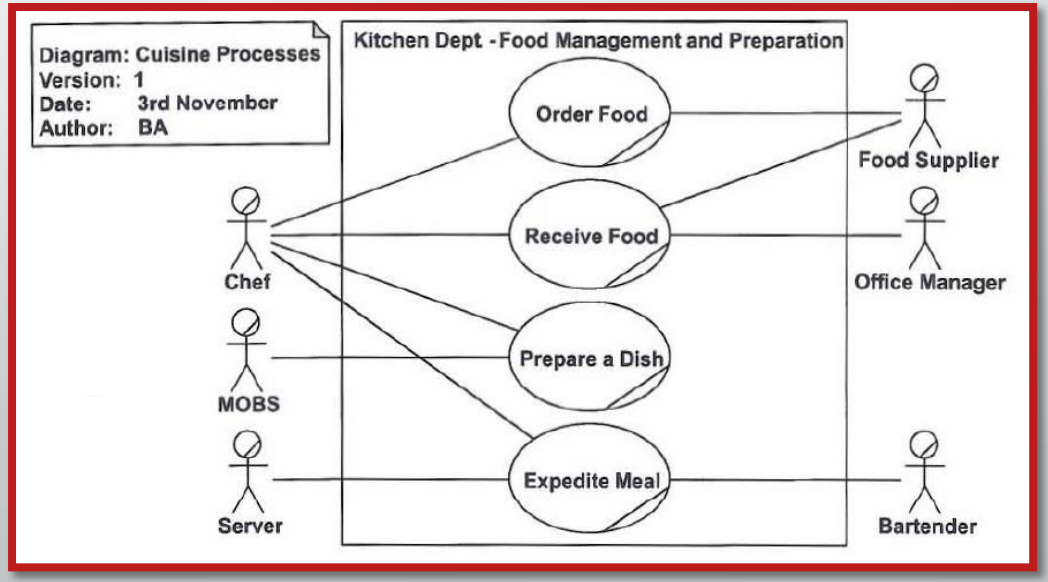
Boundary, Use Cases, and Actors

- Business **actors** are usually types of people or IT systems
  - Appear as **stick figures** on the diagram
    - Outside the boundary because **they are not inside the function**
  - They have a **goal they need to achieve**
    - Goals do not appear on the diagram



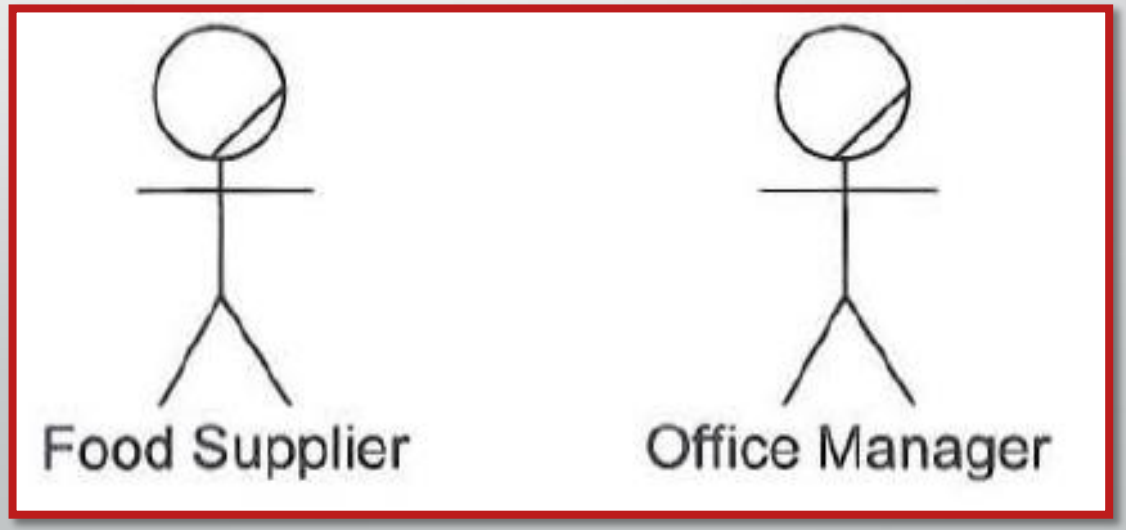
Boundary, Use Cases, and Actors

- **Associations connect actors to the processes** in which they are involved
  - Appear as straight lines connecting the actors and use cases
- Inanimate equipment and documents used to perform the use cases
  - Do not appear on the diagram



Business Actors

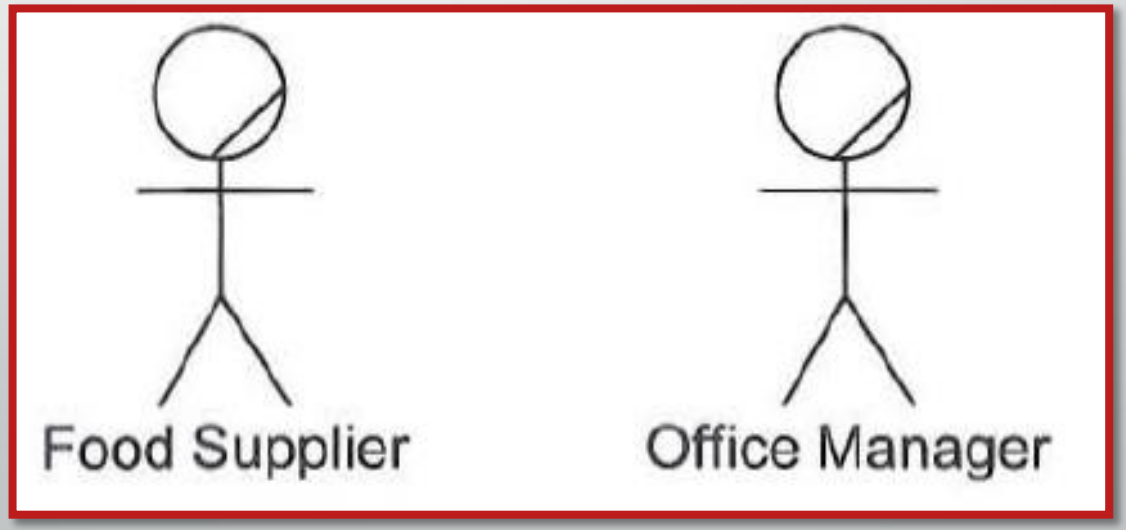
- Business actors exhibit behaviour outside the subject being modeled
  - **People identified by role, not by name and not by title**
    - For example, customer, supplier, inspector
  - **IT systems**
    - For example, banking system



Business Actors

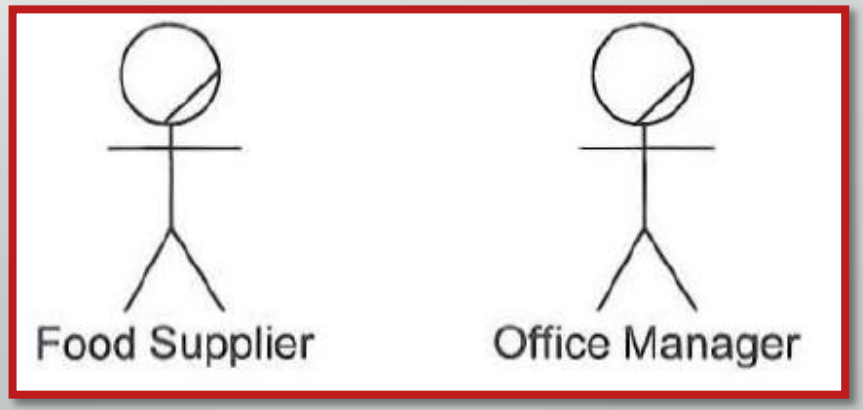
- **Organisations/units**
  - Other businesses, agencies, departments
- **Time**

- Appear as **stick figures with a slash**
  - To **distinguish them from actors on a system use-case diagram**



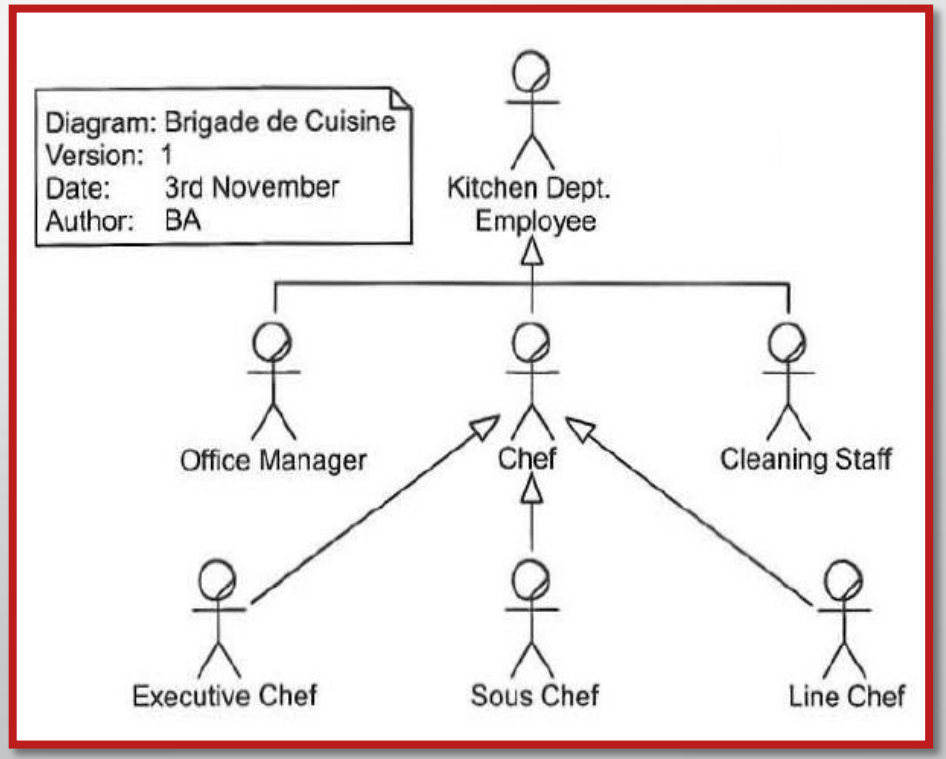
Business Actors

- Two types of actors
  - **Primary**
  - **Supporting**
- Primary actors **obtain something of value** from the process
  - There is **only one primary actor per use case**
- Supporting actors **provide services needed** to reach the primary actor's goal



Generalisation Relationship

- Generalisation is a **relationship between a thing and its type**
  - Called the **'is a kind of', 'is a type of' or 'is a' relationship**
- Terms often used:
  - **Super type** for the more general things
  - **Subtype** for the more specialised things



Generalization Relationship