An Introduction to Structured Analysis and Design

Objectives

- To introduce conventional structured analysis and design method
- To introduce the elements of the method
- To set out the data modelling and functional modelling about software application
- To introduce the process of structured analysis and design
- To explain when this method is used and why

Topics covered

- Data dictionary
- Entity relationship diagram
- Data flow diagram
- State transition diagram

The Analysis Model

- Must achieve three primary objectives:
  - To describe what the customer requires
  - To establish a basis for the creation of a software design
  - To define a set of requirements that can be validated once the software is built

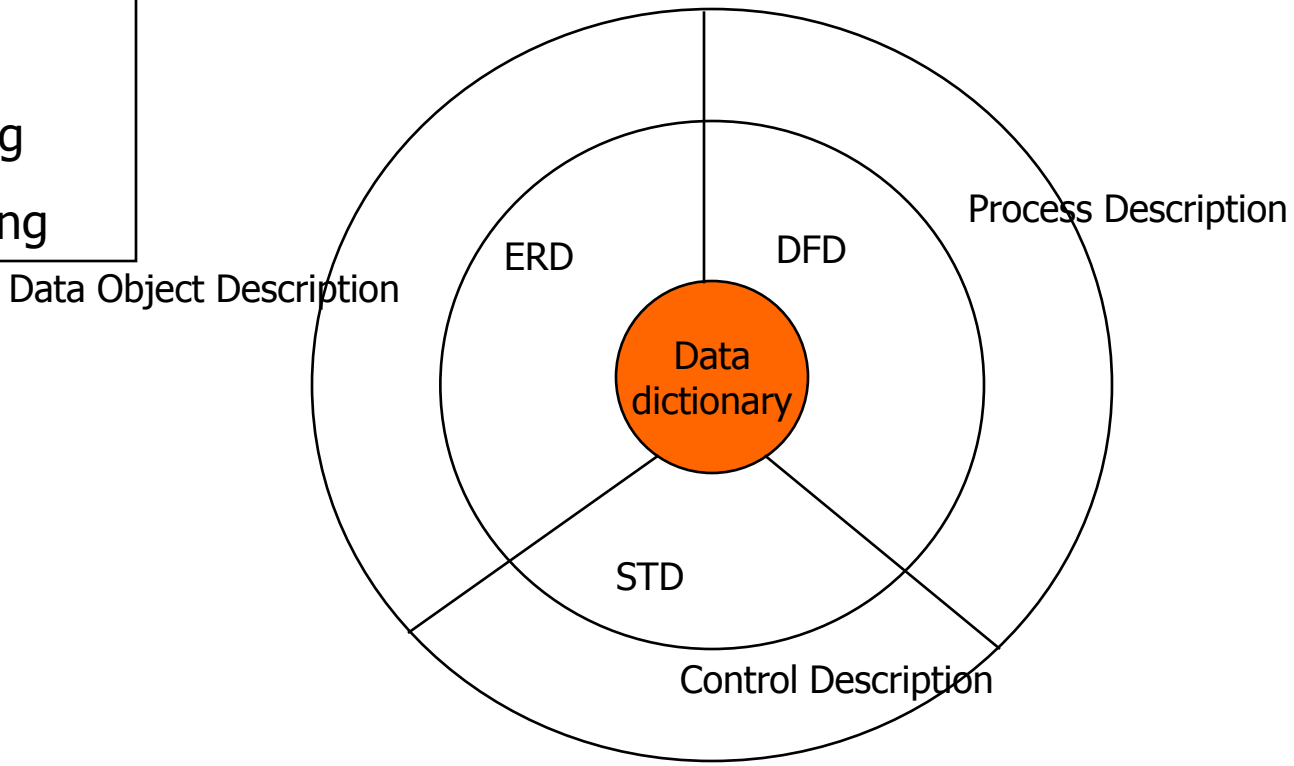
The elements of the Structured Model

Three parts:

1)Data Modelling

2)Process Modelling

3)Behavior Modelling



The elements of the Structured Model

- Data dictionary, at the core of the model, is a repository that contains descriptions of all data objects consumed or produced by the software
- Entity relation diagram(ERD), depicts relation-ships between data objects, is the notation that is used to conduct the data modelling activity
- Data flow diagram(DFD),serves two purposes:
  - To provide an indication of how data are transformed as they move through the system
  - To depict the functions that transform the data flow

Data Modelling

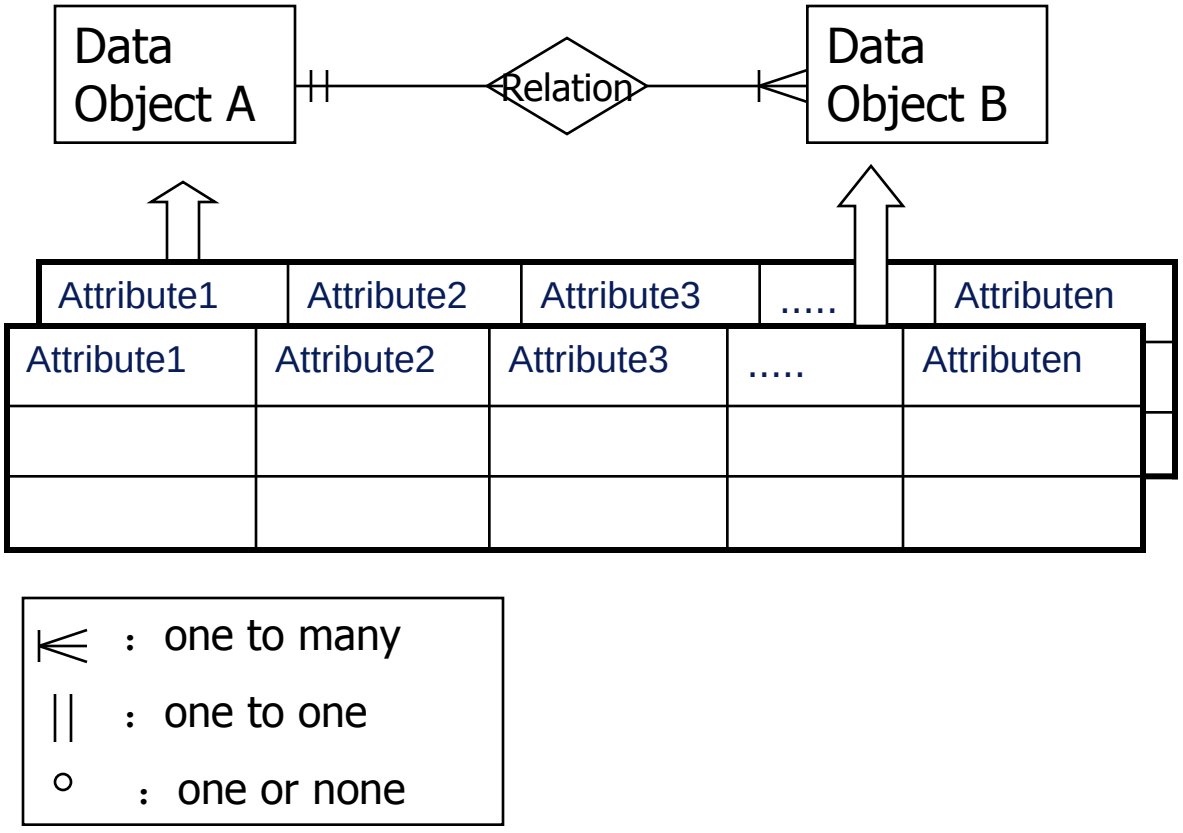
- Answers a set of specific questions that are relevant to any data processing application
  - What are the primary data objects to be process by the system?
  - What is the composition of each data object ?
  - What attributes describe the object?
  - Where do the objects currently reside?
  - What are the relationships between each object and other objects?
  - What are the relationships between the object and the processes that transform them?

Data Modelling

- The data model consist of three interrelated pieces of information:
  - The data object
  - The attributes that describe the data object
  - The relationships that connect data objects to one another

A data object encapsulate data only -- there is no reference within a data object to operations that act on the data.

Data Modelling



Data Modelling

- Data object is a representation of almost any composite information that must be understand by software;
- Has a number of different properties or attributes;
- A data object can be an external entity, a thing, a role, and organizational unit, a place or a structure
- Data object are related to one another, person can own a car
- A data object encapsulates data only, there is no reference within a data object to operations that act on the data.

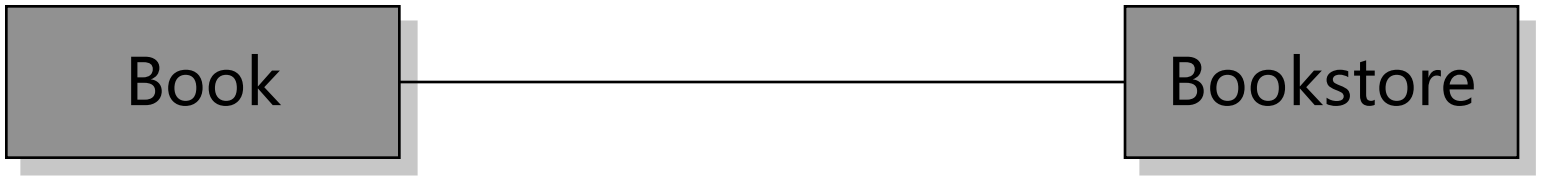
Data Modelling

- Attributes define the properties of a data object and take on one of three different characteristics, they can be used to
  - name an instance of the data object;
  - Describe the instance;
  - Make reference to another instance in another table;
- “Key” is the identifier attribute that is used to find an instance of the data object

Data Modelling

- Relationships is the connection that is established to one another.
- What are the relationships of two objects, we must understand the role of the two objects within the context of the software to be built, e.g., book and bookstore:
  - A bookstore orders books;
  - A bookstore displays books;
  - A bookstore stocks books
  - A bookstore sells books;
  - A bookstore returns books.

Data Modelling



(a) A basic connection between objects

(b) Relationships between objects

Data Modelling

- cardinality.
  - The data model must be capable of representing the number of occurences objects in a given relation-ships;
  - Cardinality is the specification of the number of occurrences of one object that can be related to the number of occurrences of another object.
  - Is usually expressed as simply ‘one’ or ‘many’

Data Modelling

- cardinality types:
  - One-to-one (1:1)- An occurrence of object A can relate to one and only one occurrence of object B, and an occurrence of B can relate to only one occurrence of A;
  - One-to-many (1:N)-one occurrence of A can relate to one or many occurrence of B, but an occurrence of B can relate to only one occurrence of A;
  - Many-to-many (M:N)-an occurrence of A can relate to one or more occurrence of B, while an occurrence of B can relate to one or more occurrence of A;

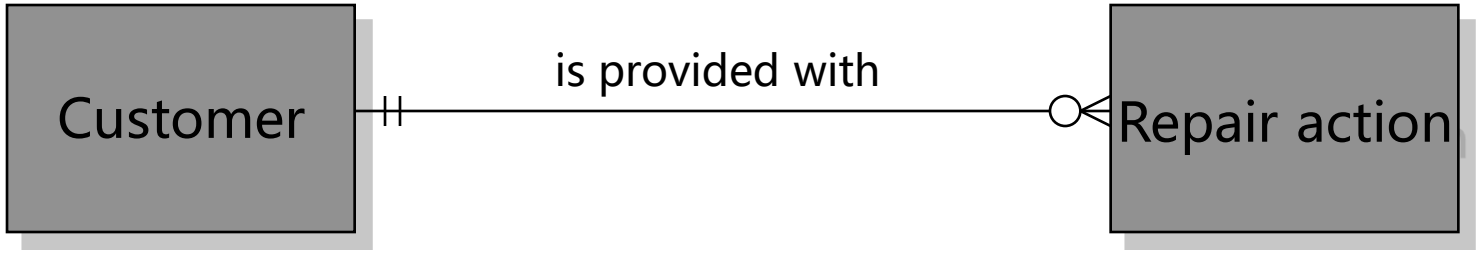
Data Modelling

- Modality.
  - Is 0 if there is no explicit need for the relationship to occur or the relationship is optional;
  - Is 1 if an occurrence of the relationship is mandatory.

Data Modelling

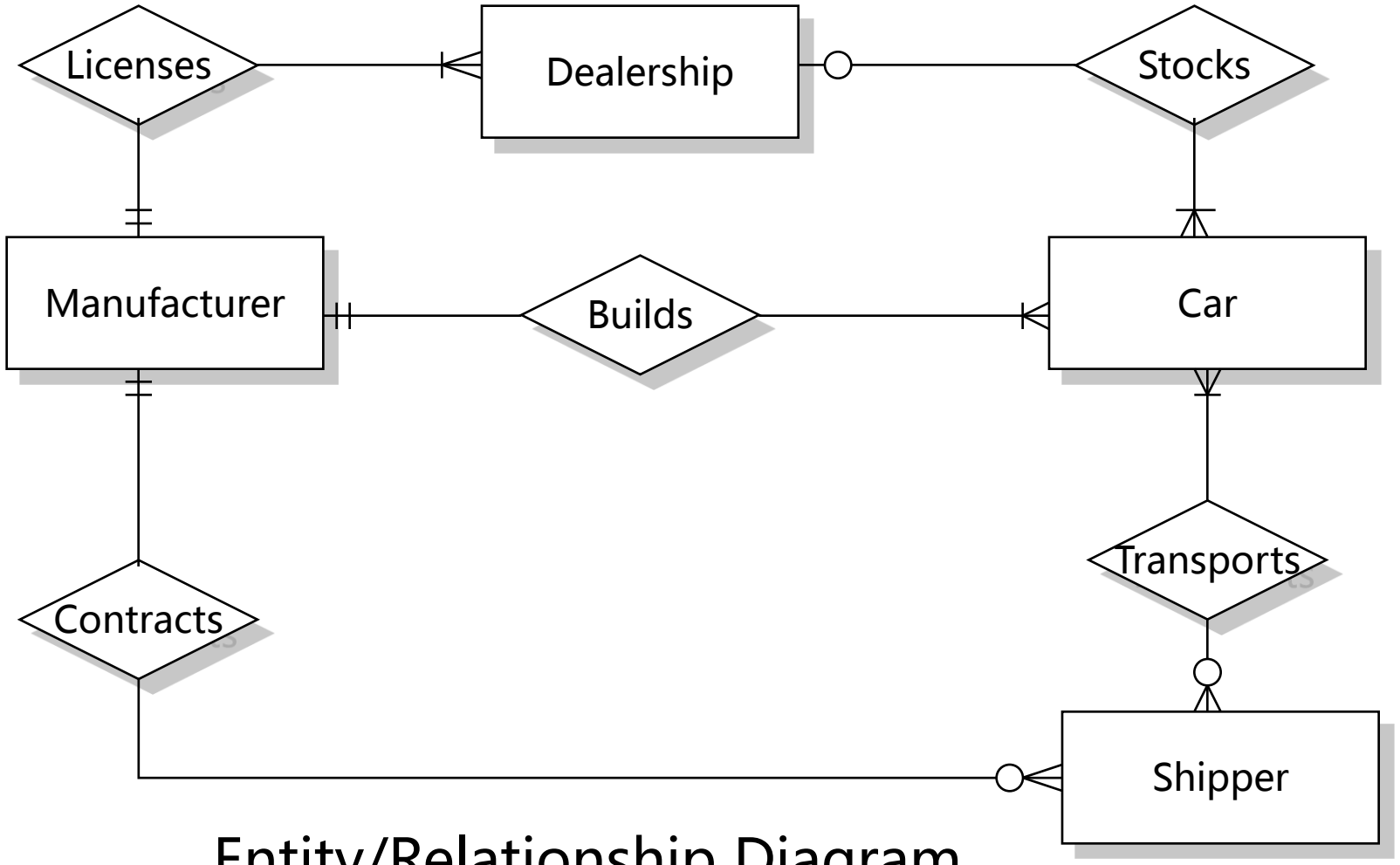
- Entity/Relationship Diagram is used to graphically represent the object/relationship pair which is the cornerstone of the data model
- A set of primary components are identified for the ERD
  - data objects;
  - Attributes
  - Relationships
  - Various type indicators

Data Modelling



Cardinality and modality

Data Modelling



Entity/Relationship Diagram

Data Modelling

- The analysis model encompasses representations of data objects, functions, and control.
- It is necessary to provide an organized approach for representating the characteristics of each data object and control item;
- ‘data dictionary is an organized listing of all data elements that are pertinent to the system, with precise, rigorous definitions so that both user and system analyst will have a common understanding of input, output, components of stores and intermediate calculations’

Data Modelling

Data dictionary

usuage: represent the characteristics of data items and control items

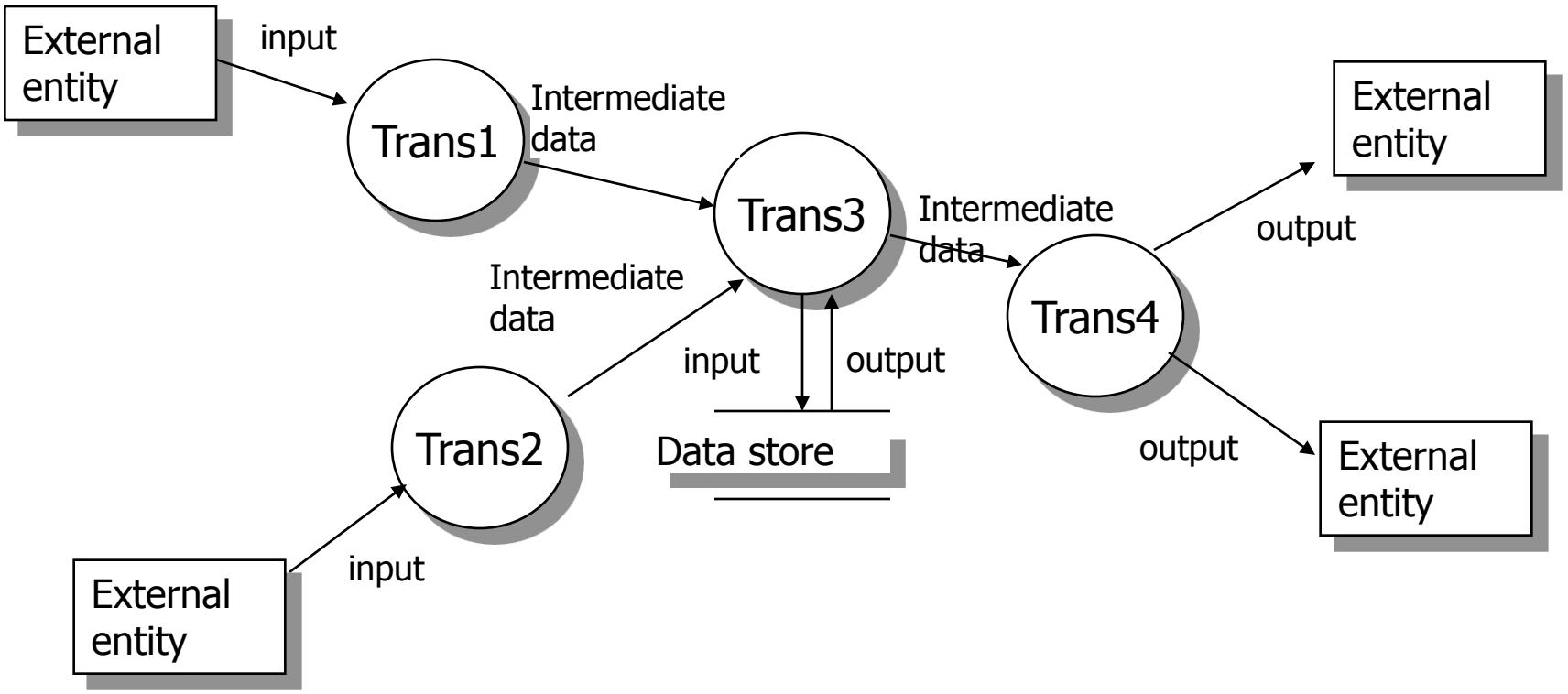
Data dictionary content and style：

name：_____ alias：_____
Where to use/How to use：
Content description：
supplement：

Functional Modelling

- Information is transformed as it flow through a computer-based system
- Transforms may comprise a single logical comparison, a complex numerical algorithm or a rule-inference approach of an expert system
- Structured analysis began as an information transform as shown in next figure.

Functional Modelling

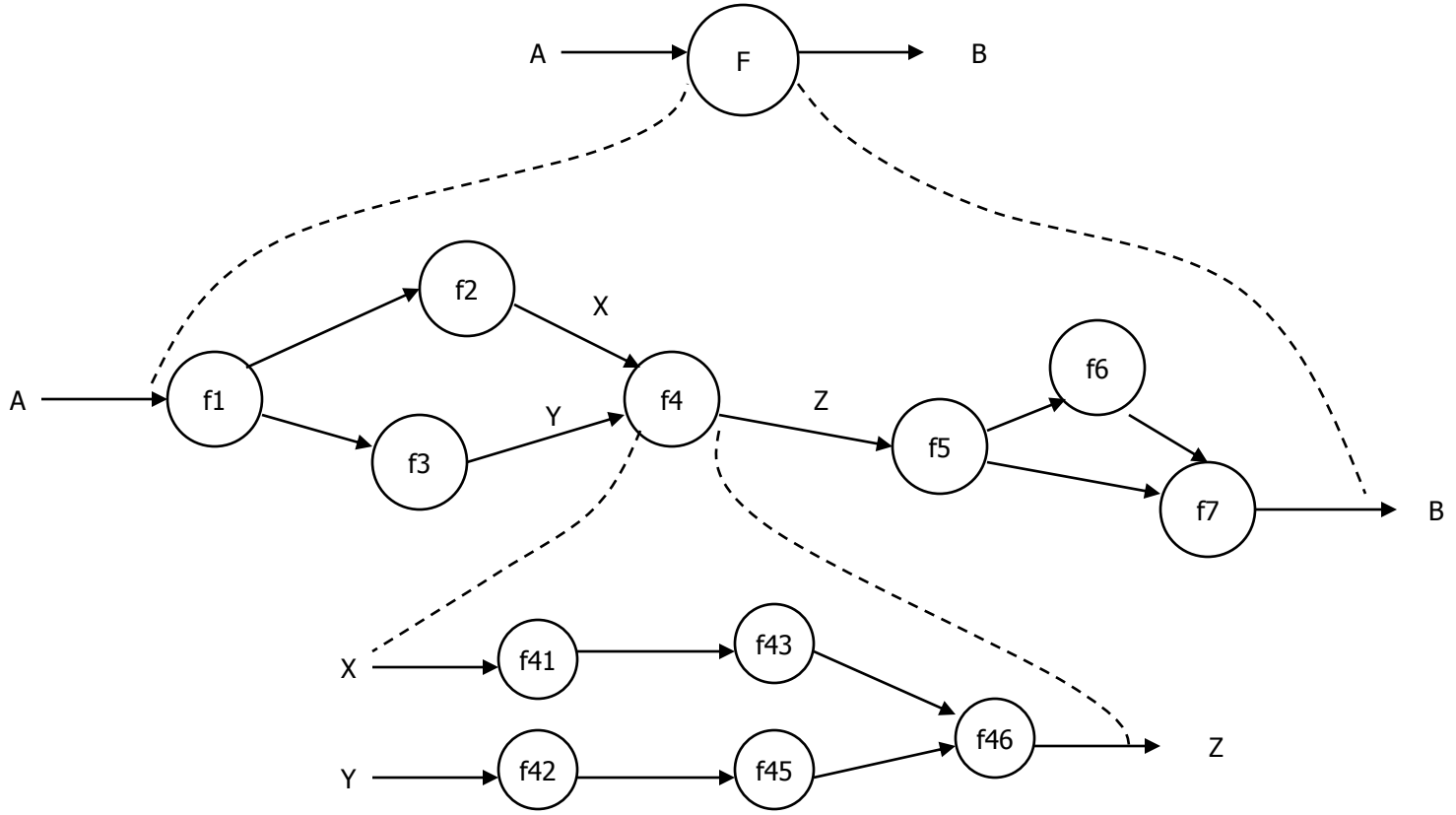


Information flow model

Functional Modelling

- As information moves through software, it is modified by a series of transformations.
- Data flow diagram is a graphical representation that depicts information flow and the transforms that are applied as data move from input to output.
- DFD is partitioned into levels that represent increasing information flow and functional detail.
- DFD provides a mechanism for functional modelling as well as information flow modeling.

Functional Modelling



Information flow refinement

Functional Modelling

- Level 0 DFD also called fundamental system model or context model, represents the entire software elements as a single bubble with input and output data indicated by incoming and outgoing arrows respectively;
- Partition it to reveal more detail ;
- Level 1 DFD might contain five or six bubbles with interconnecting arrows;
- The subfunctions are represented by the processes in level 1 DFD
- Each of the bubbles may be refined or layered to depict more detail.

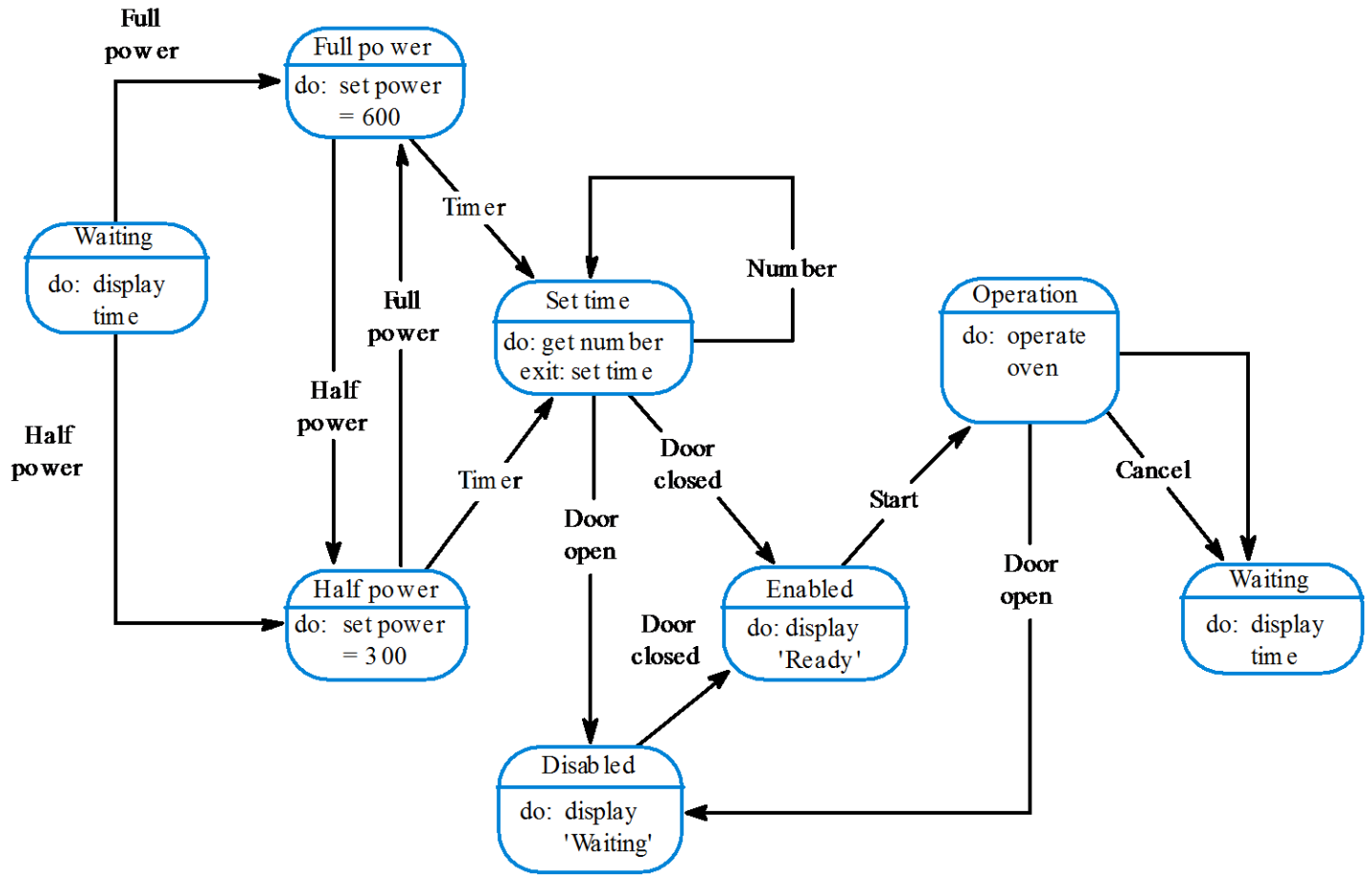
Functional Modelling

- Information flow continuity must be maintained, that is, input and output to each refinement must remain the same;
-

Behavior Modelling

- Some system is not mainly data processing, but the event driven system;
- To describe this kind of system, we use behavior modelling method
- The state transition diagram represents the behavior of a system by depicting its states and the events that cause the system to change state;
- The STD indicates what actions are taken as a consequence of a particular event.

Behavior Modelling



Behavior Modelling

- These model the behaviour of the system in response to external and internal events.
- They show the system's responses to stimuli so are often used for modelling real-time systems.
- State machine models show system states as nodes and events as arcs between these nodes. When an event occurs, the system moves from one state to another.
- Statecharts are an integral part of the UML and are used to represent state machine models.

Statecharts

- Allow the decomposition of a model into sub-models (see following slide).
- A brief description of the actions is included following the ‘do’ in each state.
- Can be complemented by tables describing the states and the stimuli.

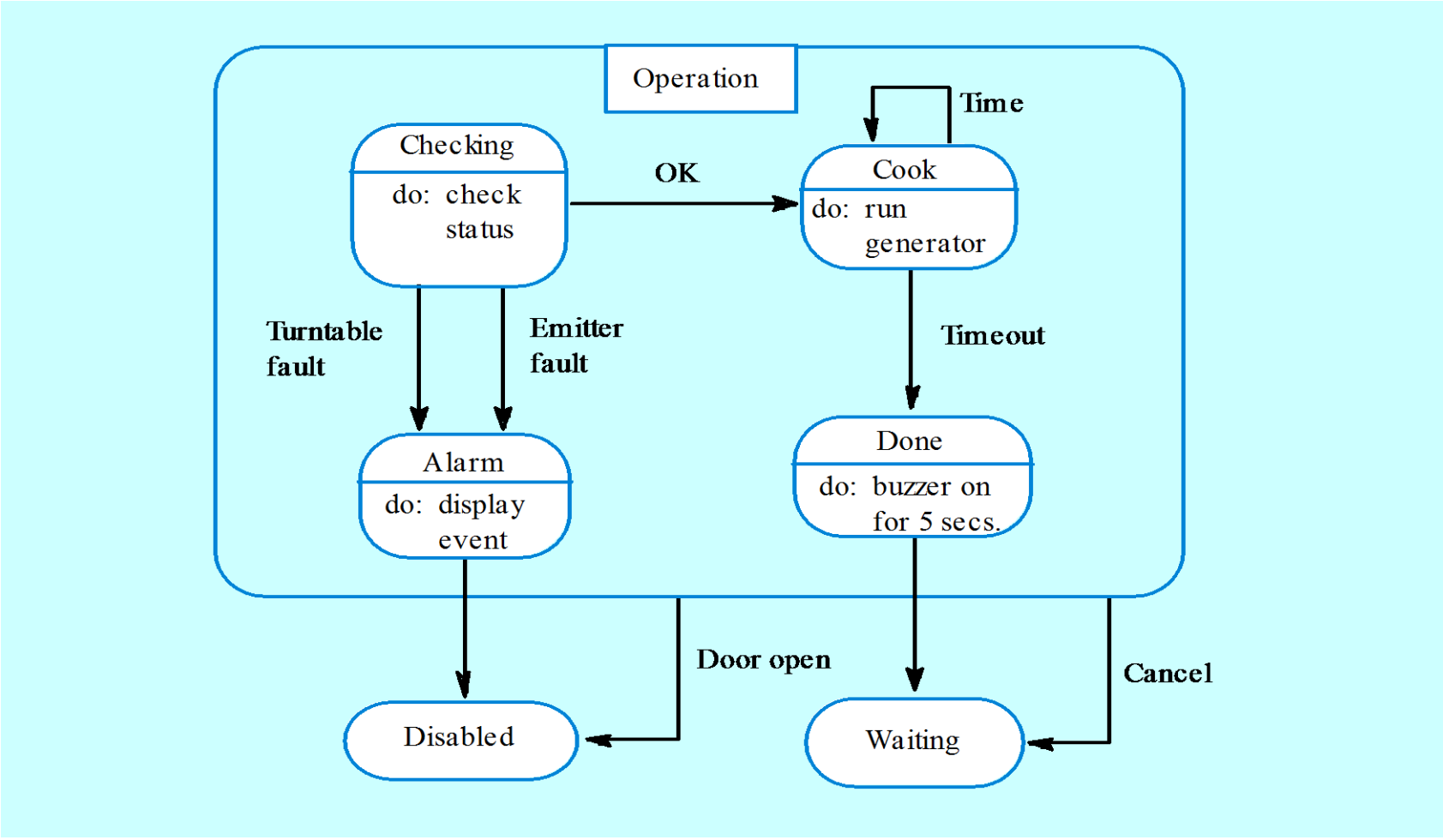
Microwave oven state description

State	Description
Waiting	The oven is waiting for input. The display shows the current time.
Half power	The oven power is set to 300 watts. The display shows ÕHalf powerÕ
Full power	The oven power is set to 600 watts. The display shows ÕFull powerÕ
Set time	The cooking time is set to the userÕs input value. The display shows the cooking time selected and is updated as the time is set.
Disabled	Oven operation is disabled for safety. Interior oven light is on. Display shows ÕNot readyÕ
Enabled	Oven operation is enabled. Interior oven light is off. Display shows ÕReady to cookÕ
Operation	Oven in operation. Interior oven light is on. Display shows the timer countdown. On completion of cooking, the buzzer is sounded for 5 seconds. Oven light is on. Display shows ÕCooking completeÕ while buzzer is sounding.

Microwave oven stimuli

Stimulus	Description
Half power	The user has pressed the half power button
Full power	The user has pressed the full power button
Timer	The user has pressed one of the timer buttons
Number	The user has pressed a numeric key
Door open	The oven door switch is not closed
Door closed	The oven door switch is closed
Start	The user has pressed the start button
Cancel	The user has pressed the cancel button

Microwave oven operation



Key Points

- Conventional structured analysis and design method is widely used for information processing system and event driven system;
- This method relies on data modelling, flow modelling;
- Using entity/relationship diagram the software engineers creates a representation of all data objects that are important for the system;

Key Points

- Data and control flow diagrams are used as a basis for representing the transformation of data and control;
- Data dictionary is the core of these system model;