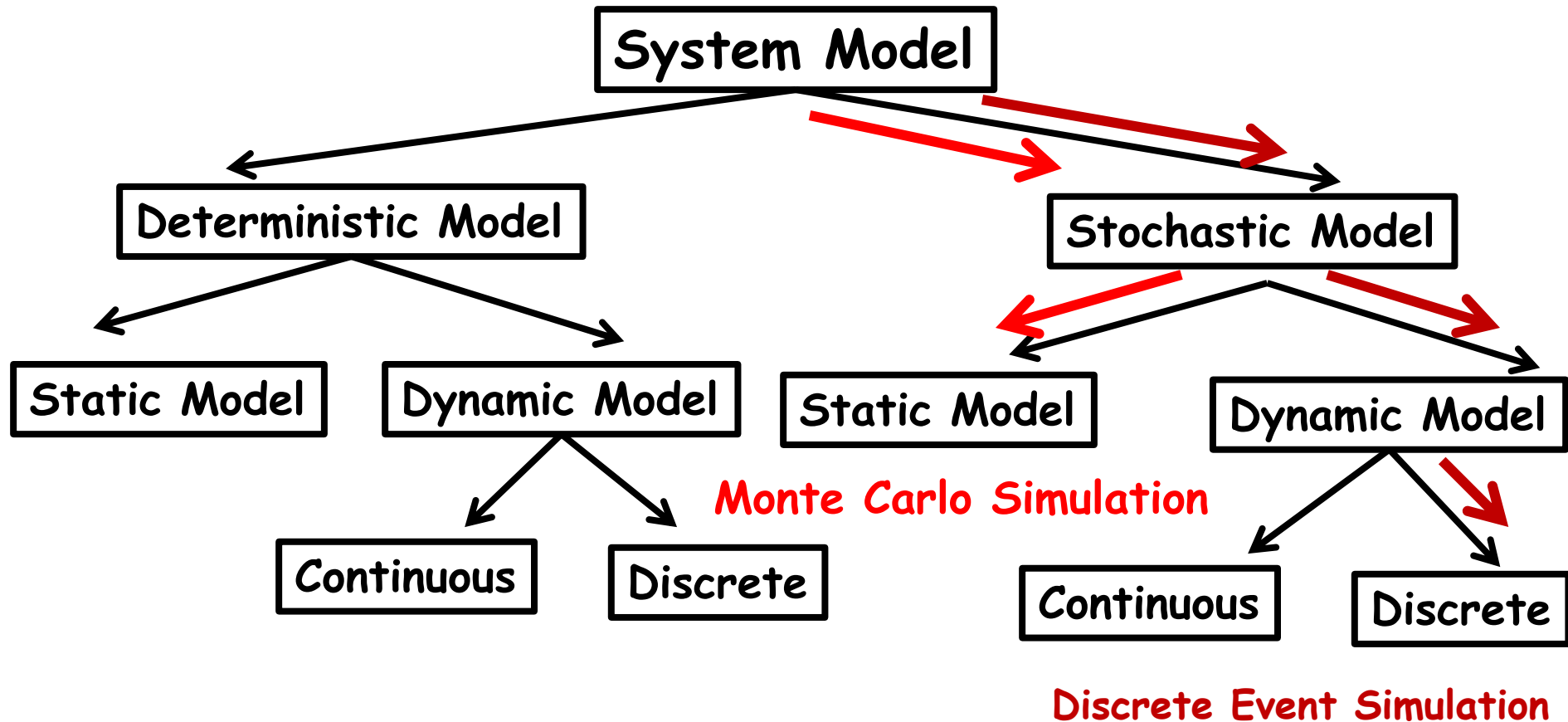# Discrete Event Simulation

Time Advance Mechanism

Event Modeling of discrete dynamic systems

DES

- Introduction:
  - System
  - Models
  - Discrete event simulation and
  - Continuous simulation
- Discrete Event Simulation:
  - Time-Advance Mechanisms
  - Event Modeling of discrete dynamic systems
  - Single-Server Single-Queue Model
  - Event graphics
  - Monte Carlo Simulation

Learn M & S : From B K Sharma

Models Taxonomy

- System Model
  - Deterministic Model
    - Static Model
    - Dynamic Model
      - Continuous
      - Discrete
  - Stochastic Model
    - Static Model
    - Dynamic Model
      - Continuous
      - Discrete

Monte Carlo Simulation

Discrete Event Simulation

Time Advance Mechanism

Advancement of Simulation Time

How to progress Simulation time?

At which points in time (in the simulated system)

do we need to compute the system state?

Two approaches:

Event-driven Approach

a. k. a

Irregular time increments (when the state changes)
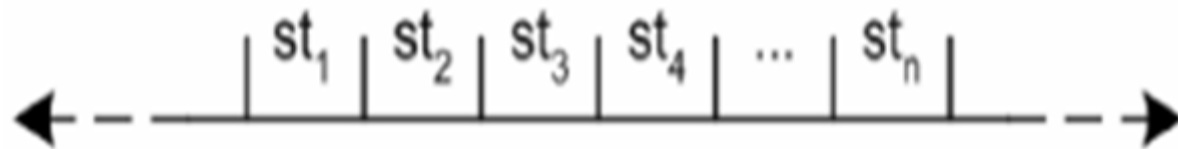
and

Fixed Time Increment Approach

a. k. a

(Time stepped Approach)

Time Advance Mechanism

Event-driven Approach

Generally used

Fixed Time Increment Approach

Seldom used

Time Advance Mechanism

Problems With Time Step Approach

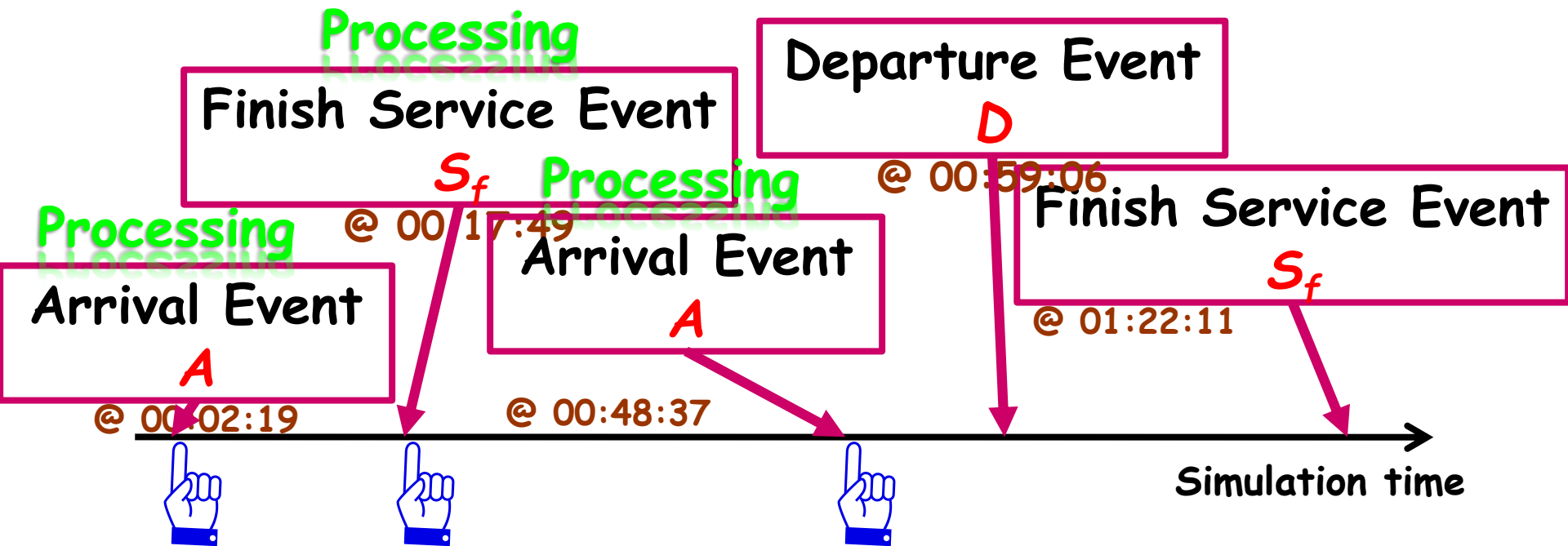**State changes may occur between time steps.**

(Arrival, 10:00)   (Landed,10:15)   (Departure, 10:30)

**Too Much Fog, Fast Wind, Or
Any Technical Problem at 10:20**

Time Advance Mechanism

Event-driven Approach

Processing

Finish Service Event

S_f

@ 00:17:49

Processing

Departure Event

D

@ 00:59:06

Processing

Arrival Event

A

Finish Service Event

S_f

@ 01:22:11

Processing

Arrival Event

A

@ 00:02:19

@ 00:48:37

Simulation time

t = 00:48:37

Terminology:

Event Notice:

A record of an event to occur at the current or some future time, along with any associated data necessary to execute the event.

Event list:

A list of event notices for future events, ordered by time of occurrence; known as the future event list (FEL) or future event set (FES).

Event-driven Time Advance Approach

1   Initialize simulation clock to 0

2   Determine times of occurrence of future events – *event list*

3   Clock advances to next event, which is executed

4   Continue until stopping rule is satisfied (must be explicitly stated)

5   Clock "jumps" from one event time to the next, and doesn't "exist" for times between successive events … periods of inactivity are ignored

Old System Snapshot at time t

Clock	State	...	Future event list
t	(5,1,6)		$(3, t_1)$ – Type 3 event to occur at t_1
			$(1, t_2)$ – Type 1 event to occur at t_2
			$(1, t_3)$ – Type 1 event to occur at t_3
			...
			$(2, t_n)$ – Type 2 event to occur at t_n

New System Snapshot at time t_1

Clock	State	...	Future event list
t_1	(5,1,5)		$(1, t_2)$ – Type 1 event to occur at t_2
			$(4, t^*)$ – Type 4 event to occur at t^*
			$(1, t_3)$ – Type 1 event to occur at t_3
			...
			$(2, t_n)$ – Type 2 event to occur at t_n

Terminology:

Event Notice:

A record of an event to occur at the current or some future time, along with any associated data necessary to execute the event.

Event list:

A list of event notices for future events, ordered by time of occurrence; known as the future event list (FEL) or future event set (FES).

Event Scheduling/Time Advance Algorithm

Step 1: Remove the event notice for the imminent event from FEL

e.g. Event (3, t1) in the example

Step 2: Advance Clock to imminent event time,

e.g. Set clock = t_1

Step 3: Execute imminent event

Step 4: Generate future events and place their event notices on FEL

e.g. Event (4, t*)

Step 5: Update statistics and counters

Event Scheduling/Time Advance Algorithm

1. set simulation clock = 0

2. set cumulative statistics to 0

3. define initial system state (queue empty, server idle)

4. generate the occurrence time of the first arrival and advance simulation clock to the time of this event

5. Process this event

6. select the next event on event list (arrival or departure or landed event)

7. advance simulation clock to time of next event

8. process this event (execute the corresponding event routine)

9. if not end-of-simulation, goto step 6

Event Modeling of Discrete Dynamic Systems

What is discrete event dynamic system?

A discrete event dynamic system (DEDS)

is a

dynamic,

asynchronous system,

where the state transitions

are initiated by events

that occur at

discrete instants of time.

What is discrete event dynamic system?

Typical examples of DEDS are:

flexible manufacturing systems,

telecommunication networks,

traffic control systems,

multiprocessor operating systems,

. . . .

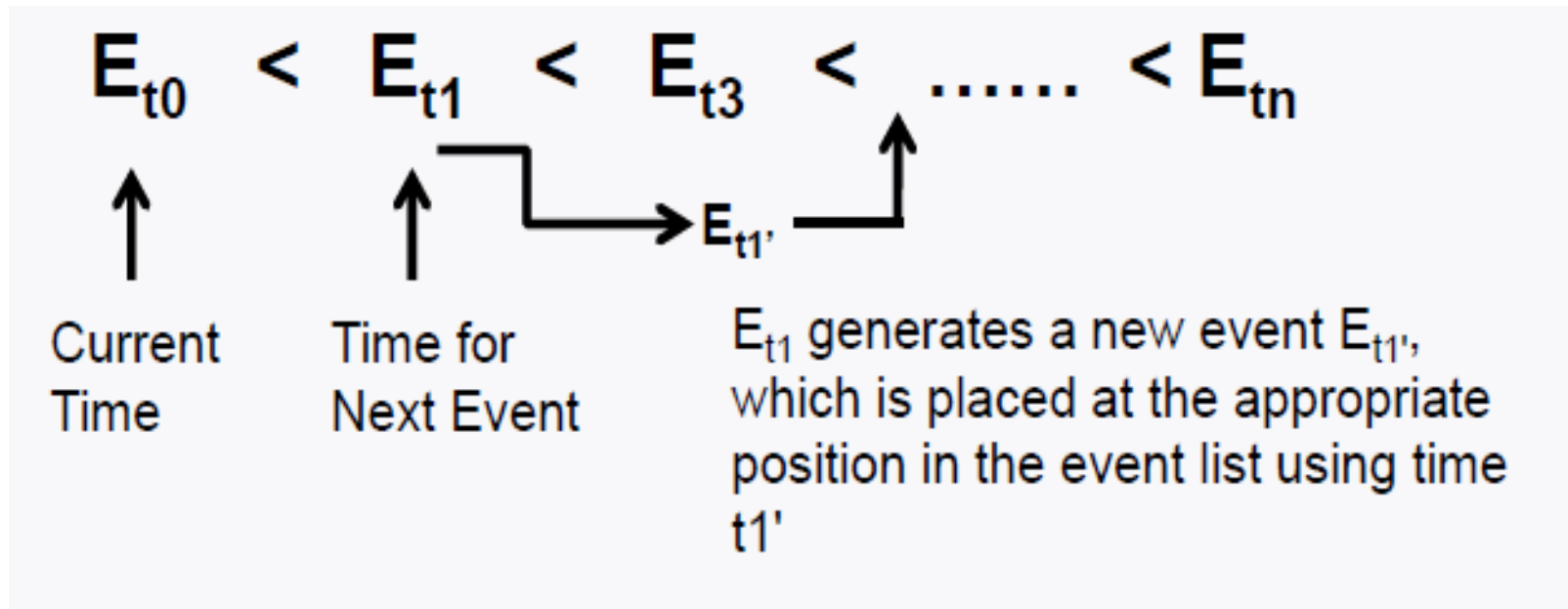# Modeling and Analysis Techniques for DEDS

There are many modeling and analysis techniques

for DEDS, such as

Automata theory

queuing theory,

state machines

computer simulation

Petri nets

generalized semi-Markov processes

....

Design of Events

Events are chronologically ordered in time.

(Arrival, 10:00) (Landed, 10:15)   Departure, 10:30)

$$E_{t0} \quad < \quad E_{t1} \quad < \quad E_{t3} \quad < \quad ...... \quad < \quad E_{tn}$$

$E_{t1'}$

Current Time

Time for Next Event

E_{t1} generates a new event $E_{t1'}$, which is placed at the appropriate position in the event list using time t1'

Too Much Fog, Fast Wind, Or Any Technical Problem at 10:20

Sample Design of Arrival Event:

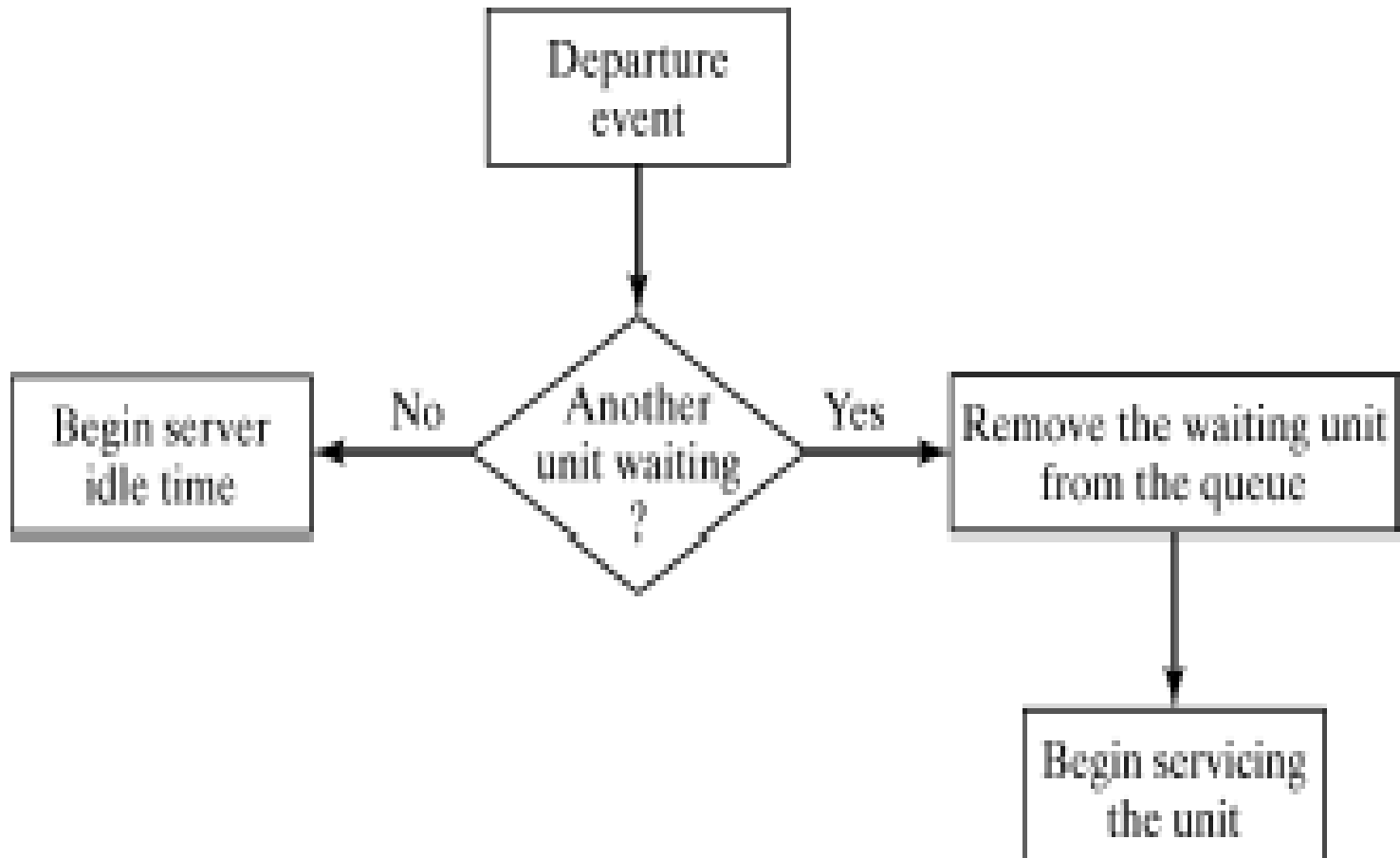**If is server idle, unit gets service, otherwise unit enters queue.**
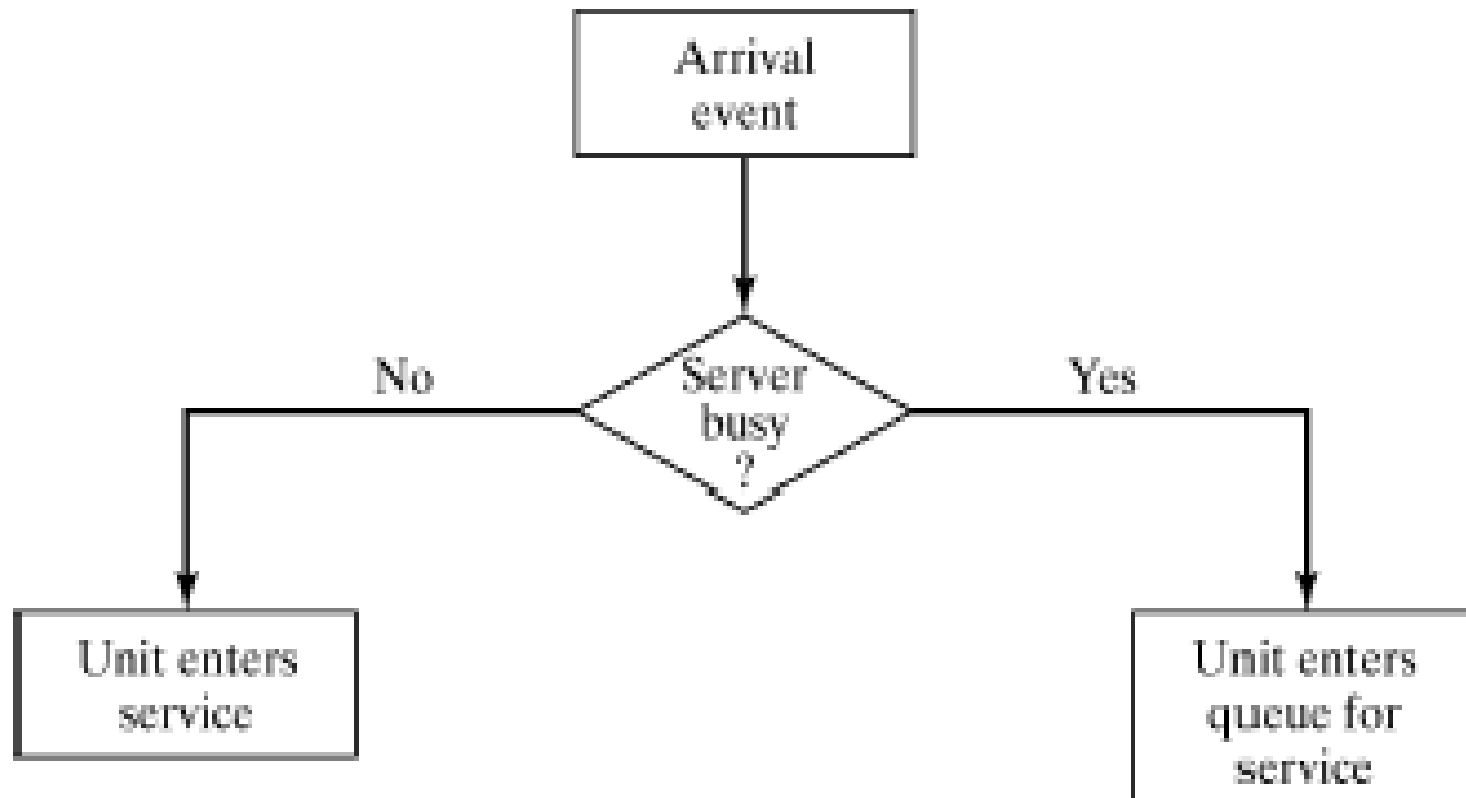
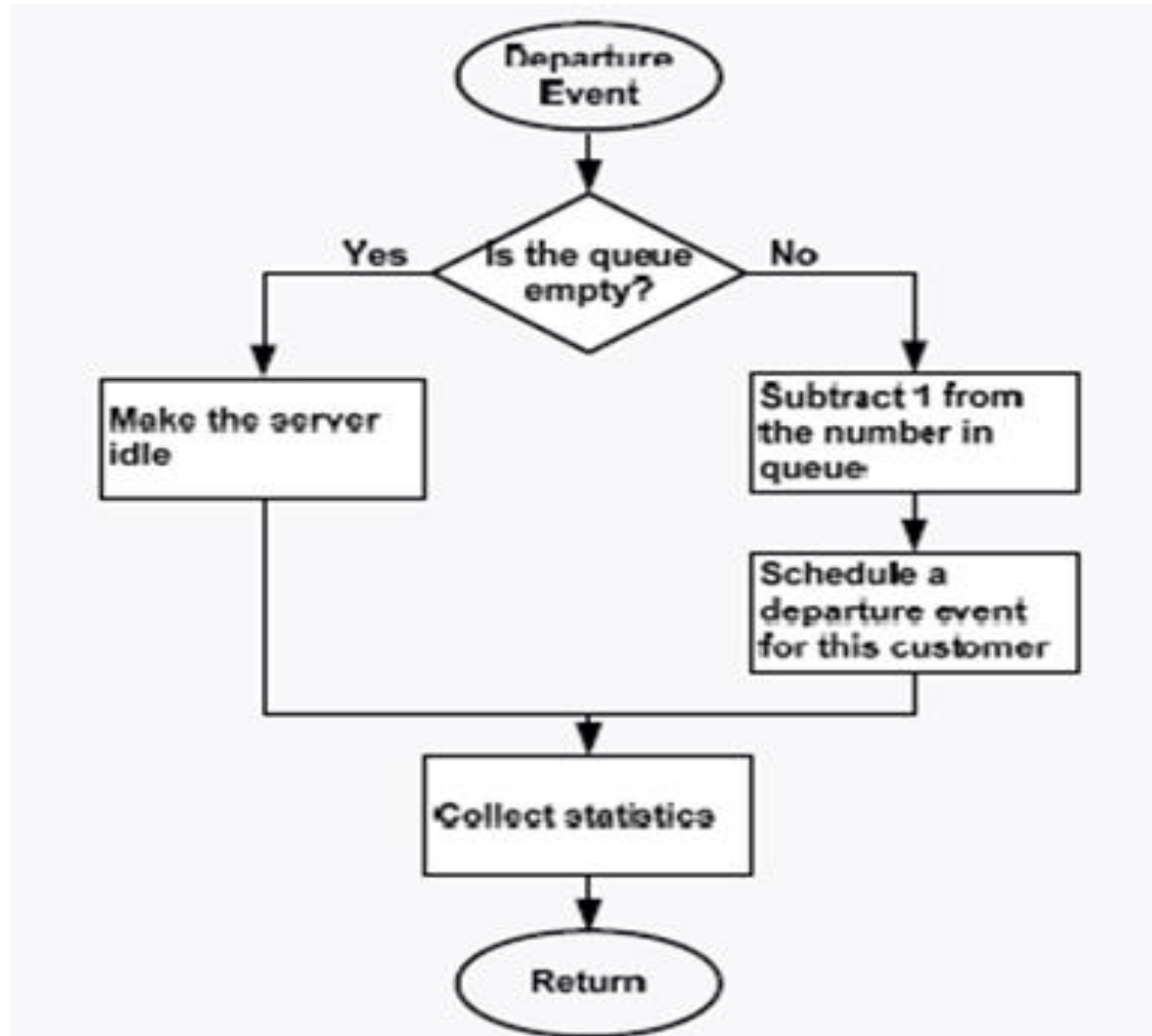## Sample Design of Arrival Event:
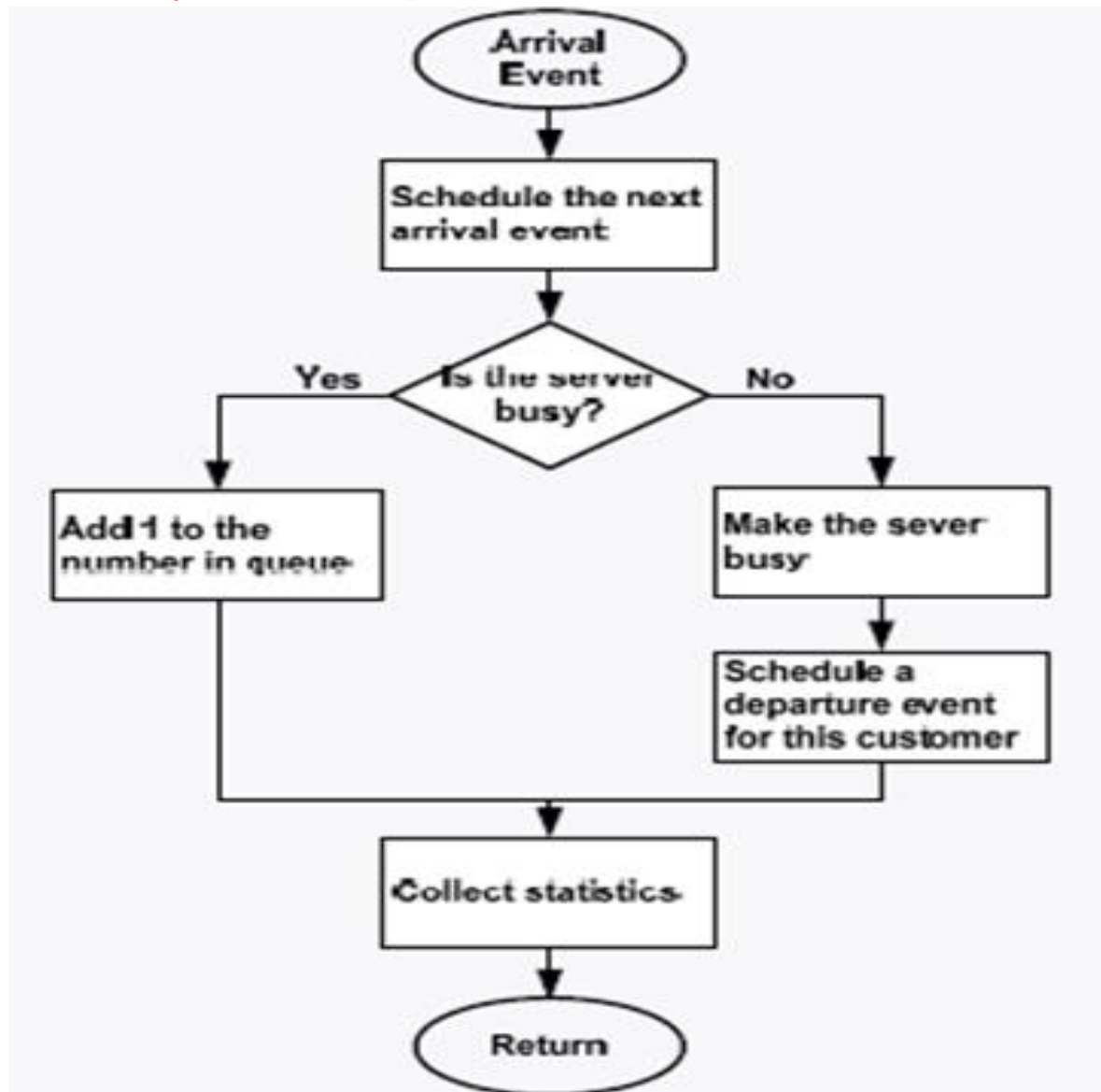
Sample Design of Departure Event:

If queue is not empty, begin servicing next unit, otherwise server will be idle.

Sample Design of Departure Event:

Questions

Explain fixed increment time advance mechanism.

Explain event modeling of discrete dynamic system with object diagram.

What are event graphs? Give an event graph for queuing model.