香港城市大學
City University of Hong Kong

396EM Airline Operations and Scheduling/ 6075MAA Airline Scheduling and Operations

Lecture 5b
Discrete-event Simulation and Modelling

Developed & Revised :
Dr Yang Dai (Coventry)
Kimman Lui & Rossella Lau (Scope)
Presented: Kinki Leung

Outline of the Lecture

- Discrete-event simulation
- Conceptual Modelling
  - Process and Components
  - Computer simulation
    - Three phase simulation
    - Random samples
    - Time distribution
- A Structural Approach (steps)
- Modelling with Flow Chart
- Queuing Process
- Queue measurement & Reduction

Introduction

- Typically, simulation refers to a specific class of dynamic models involving the detailed observation of a complex system over time.
- *Continuous Simulation models* involve aggregate variables that change more or less continuously with time. (e.g., movement of an airplane, distances between airplanes on a runway).
- *Discrete Event Simulation* is applied whenever individual items are tracked and in which abrupt or non-smooth changes in the timing of events is the norm. (e.g., customers arrived, number of airplanes on a runway).

Definition of Discrete Event Simulation

- Discrete-event simulation concerns the conceptual modelling of a system as it evolves over time by a representation in which the state variables change instantaneously at separate points in time.

- Discrete event simulation models are most commonly used to create detailed operational systems representing demands among activities requiring scarce resources over time.

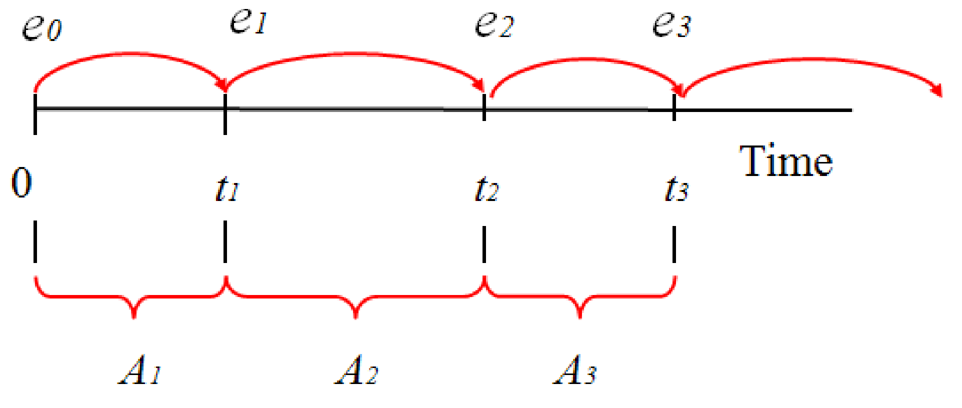
Discrete Event Simulation Approach: Why

- As the operation of any system can be represented as a sequential list of events and each event occurs at an instant in time and marks a change of state in the system.

- Simulation technology can capture such components in terms of computer program and hence the simulation process of any system will become easier and flexible.

Discrete-Event Simulation Process

- State changes at discrete points in time due to the occurrence of asynchronous events.
- The operation of a system is represented as a chronological sequence of events.
- Each event occurs at an instant in time and marks a change of state in the system.



Discrete-Event Simulation Components

- System state: The collection of state variables necessary to describe the system at a particular time
- Simulation clock: A variable giving the current value of simulated time
- Event list: A list containing the next time when each type of event will occur
- Statistical counters: Variables used for storing statistical information about system performance

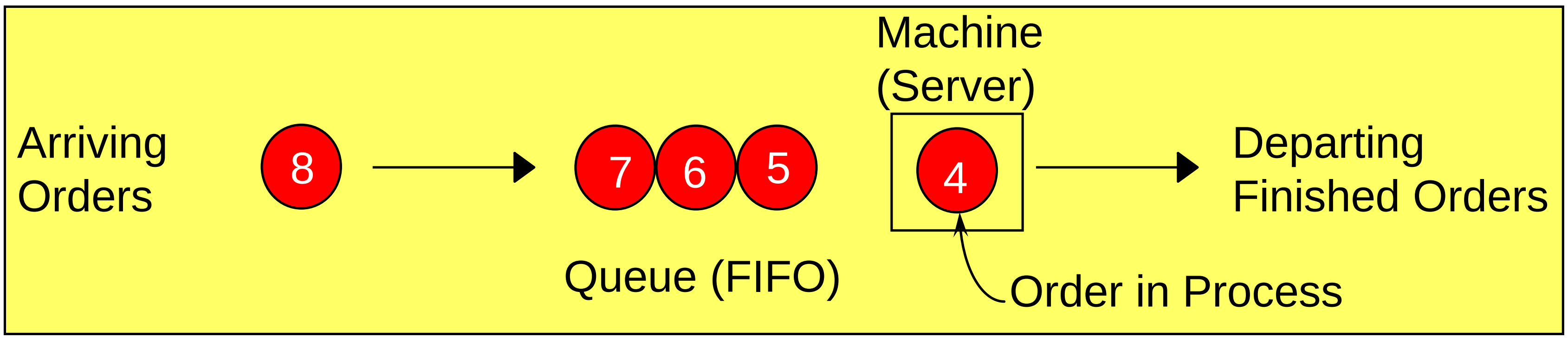
Discrete-Event Simulation Stopping Rules

- Number of events of a certain type reached a pre-defined value
  - Example: stop simulation after the 1000th departure

- Simulation time reaches a certain value; this is usually implemented by scheduling a “end-simulation” event at the desired simulation stop time.

Discrete Event Simulation: Case Study

Modelling of a Manufacturing System



- Simulation Objectives:
  - Identify State, Actions, and other related variables.
  - Real objective: Introduce you to discrete event simulation!

Model Inputs and Specifics

- Initially (time 0) empty and idle
- Base time units: minutes
- Input data (assume given for now …), in minutes:

Order Number	Interarrival Time	Service Time
1	6	4.14
2	12.25	6.25
3	19.10	6.20

The requirements

- Manually track events, actions and other related variables.
- Use given inter-arrival, service times
- Keep track of event calendar
- “Lurch” clock from one event to the next
- Stop after 3 orders.

Identifying System Components

- Actions (Activities) for this system:
  - The starting and completion status of each order.

- Events for this system:
  - The arrival or departure of an order.

Visual Example of Next Event Mechanism: Manufacturing System (Con't)

Generate Orders

TNOW
+
Duration

SIMULATED PROCESS

Order	Arrival	Proc Time	Finish
1	6	4.14	10.14

Order Time in

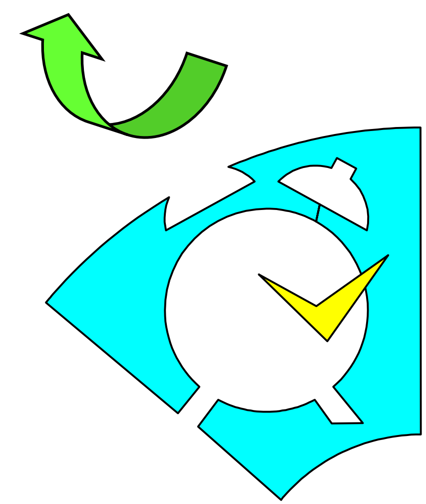
Order Time Out

TNOW=0

Schedule by Simulation

Order	**SIM TIME**

Event	**Action**
O1 Arr	O1 Proc



Visual Example of Next Event Mechanism: Manufacturing System (Con't)

Generate Orders

TNOW
+
Duration

SIMULATED PROCESS

Order	Arrival	Proc Time	Finish
1	6	4.14	10.14
2	12.25	6.25	18.50

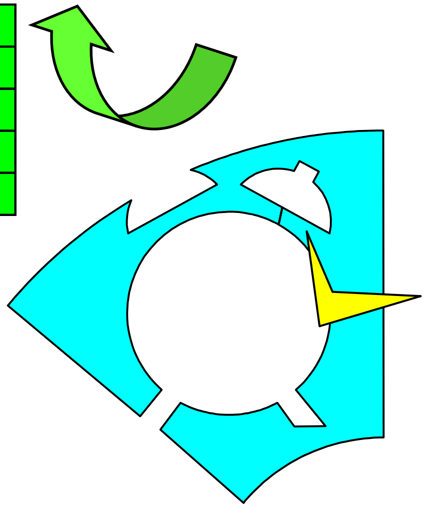
Order Time in

Order Time Out

Schedule by Simulation

Order	**SIM TIME**
1	10.14

Event	**Action**
O1 Arr	O1 Proc
O2 Arr, O1 Dep	O1 Fin, O2 Pro



TNOW=10.14

Visual Example of Next Event Mechanism: Manufacturing System (Con't)

Generate Orders

Schedule by Simulation

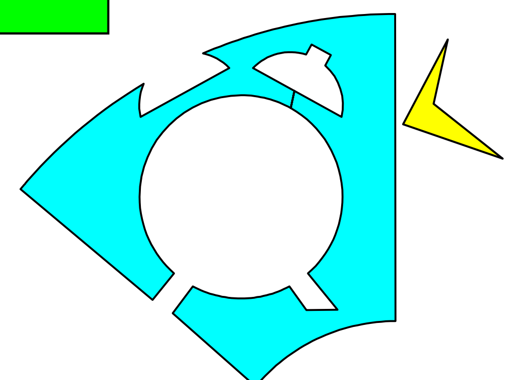
Order	SIM TIME
1	4.14
2	18.5

SIMULATED PROCESS

TNOW + Duration

Order	Arrival	Proc Time	Finish
1	6	4.14	10.14
2	12.25	6.25	18.50
3	19.10	6.20	25.30

Event	Action
O1 Arr	O1 Proc
O2 Arr, O1 Dep	O1 Fin, O2 Pro
O3 Arr, O2 Dep	O12Fin, O3 Pro



Order Time in

Order Time Out

TNOW=18.50

Simulation on Computer

- Computer usually applies a Three-phase simulation approach

- Modelling of arrival / service time
  - Deterministic (predictable, fixed)
  - Probabilistic (unpredictable)
    - Generation of samples using random mechanism according to statistical functions / models
    - Popular probabilistic arrival / service time distribution:
    - Poisson, Exponential, Erlang, Empirical, Triangular, Normal, Uniform
    - More details will be studied with Data / Results analysis

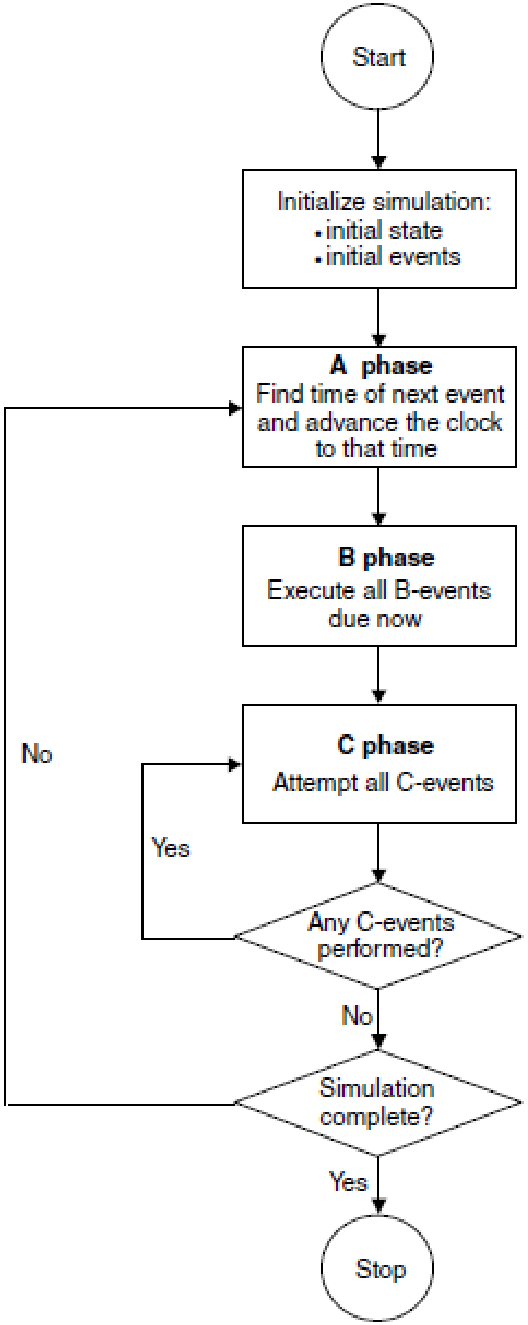
- Contingencies
  - A departing customer could leave behind an empty queue
  - Termination conditions

- Time ties

Three-phrase simulation

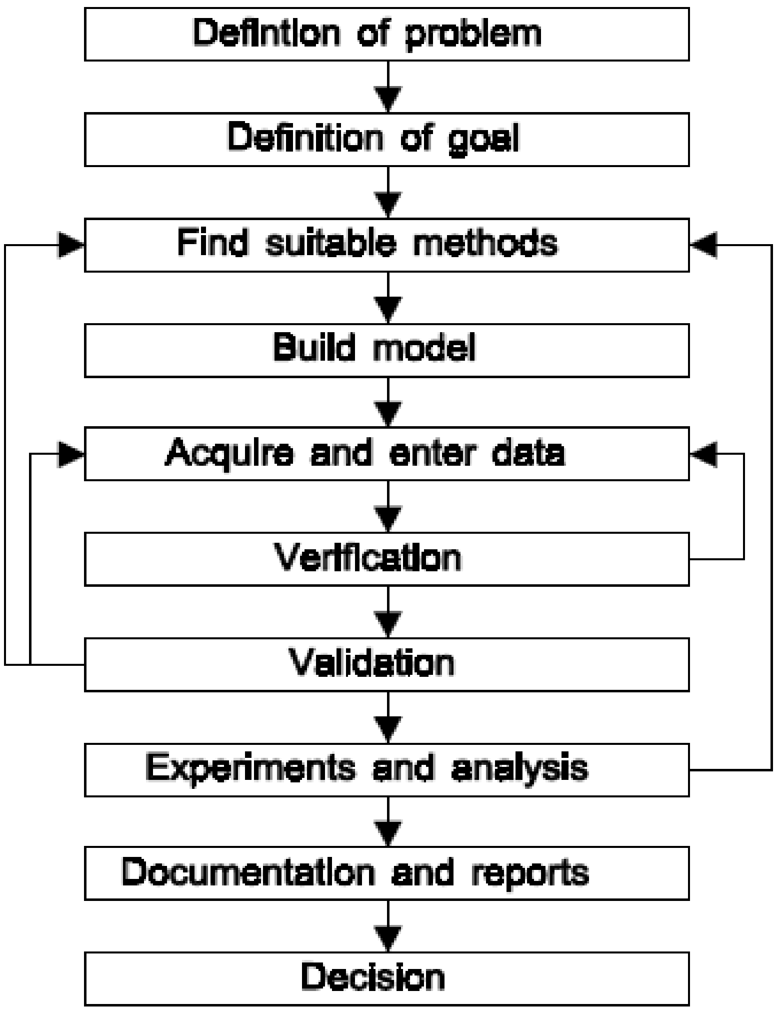
- Phase B – Event based simulation
  - B-Events (Bounded or booked) events
  - State changes that are scheduled to occur at a point of time
  - Relate to arrivals or the completion of an activity
  - E.g., Customer arrivals occur in a predictable or unpredictable manner

- Phase C – Activity based simulation
  - C-Events (conditional) events
  - State changes that are dependent on the conditions in the model
  - Relate to the start of some activity
  - E.g., A service can only start if there is a customer waiting and the operation is not busy

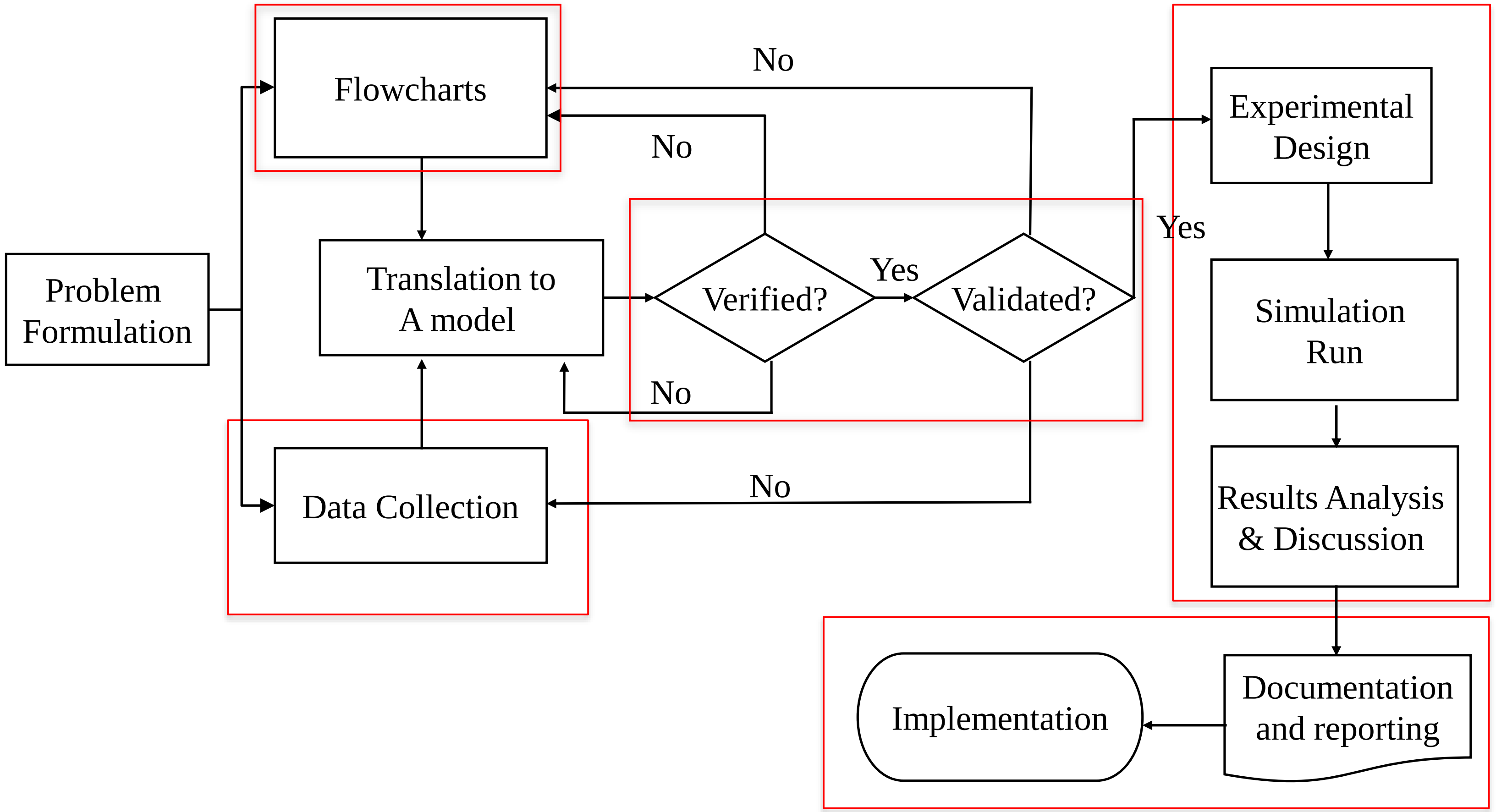


A Systematic Simulation Approach

- A simulation study normally consists of several distinct stages.
- However, not all simulation studies consist of all these stages or follow the order stated here.
- On the other hand, there may even be considerable overlap between some of these stages.



Example: Steps in a Simulation Study



Problem Formulation

- In the problem formulation we:
  - Identify controllable and uncontrollable inputs.
  - Identify constraints on the decision variables.
  - Define system key performance indicators and an objective function.
  - Develop a preliminary model structure to interrelate the inputs and the measure of performance.
  - May be the problem needs reformulation as the study progresses.

Data Collection

- Regardless of the method used to collect the data, the decision of how much to collect is crucial.
- Data amount changes with the degree of complexity of the model.
- Data should be collected for the validation as well.

Translation to a Computer Model

- Model requires great deal of information and computation.

- Needs to be translated into computer recognizable format using either special purpose or general purpose languages.

- Focus of this course is by using SIMUL8 for model building.

Verification vs. Validation Definition

- Verification:

  "Are we building the model right?"

  – simulation model should conform to reality

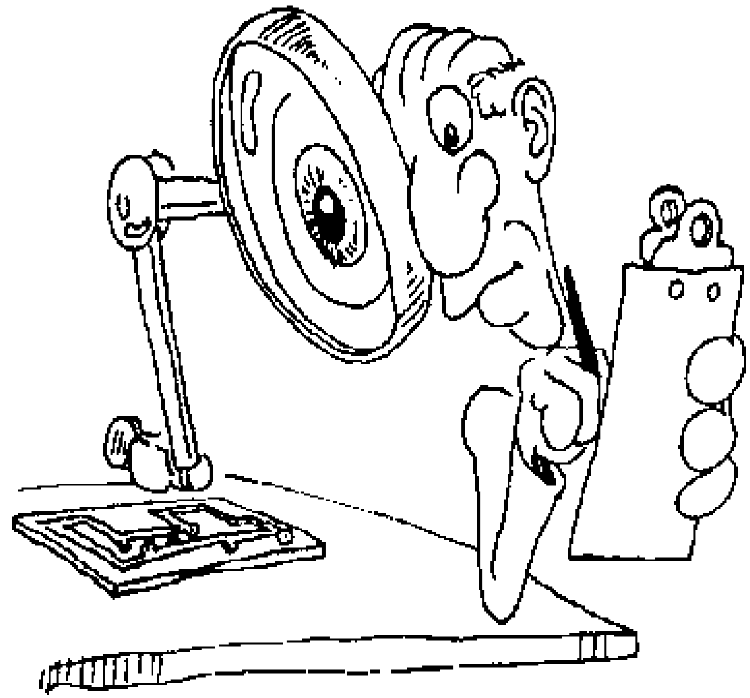
- Validation:

  "Are we building the right model?"

  – Simulation model should do what industry *really* requires.

Experimentation and Analysis of Results

- Experimentation: The execution of the simulation model to obtain output values
- Analysis of Results: The process of analysing the simulation outputs to draw inferences and make



recommendations for problem resolution

Implementation and Documentation

- The process of implementing decisions resulting from the simulation and documenting the model and its use.

- Success depends how well previous steps were followed.

- Preparation of progress reports includes:
  - Alternative scenarios
  - Performance measures or criteria used
  - Results of experiments
  - Recommendations

Modelling Languages

Table 5.1 Component List for Single Server Queue.

Component	Detail
Customers	Time between arrivals (distribution)
Queue	Capacity
Service desk	Service time (distribution)

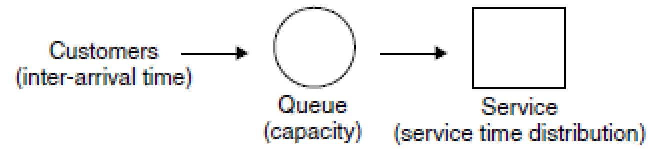


Figure 5.2 Process Flow Diagram for Single Server Queue.

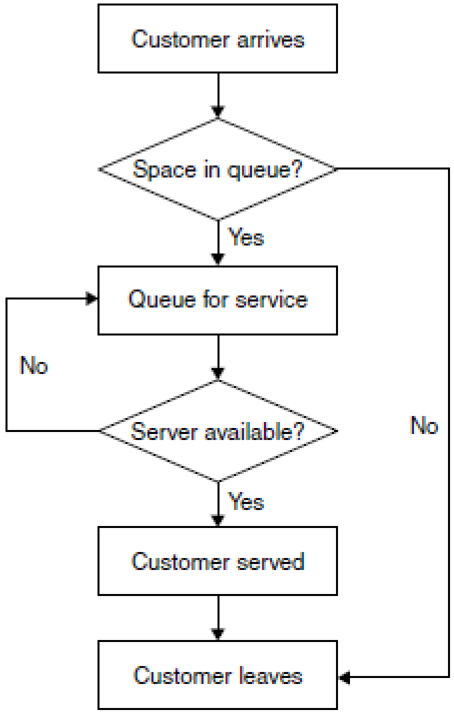


Figure 5.3 Logic Flow Diagram for Single Server Queue.

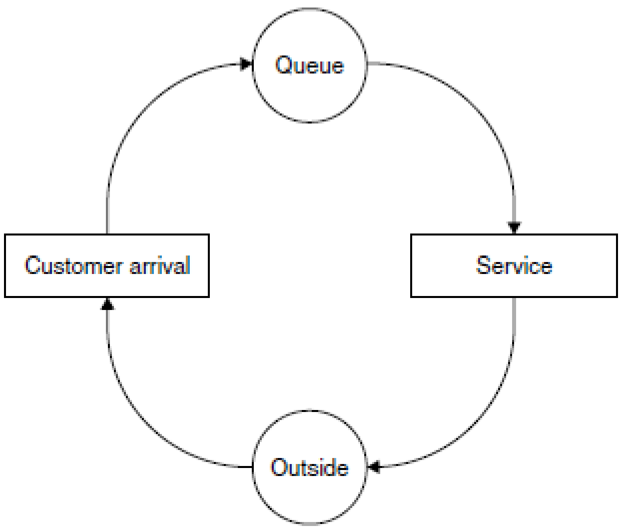
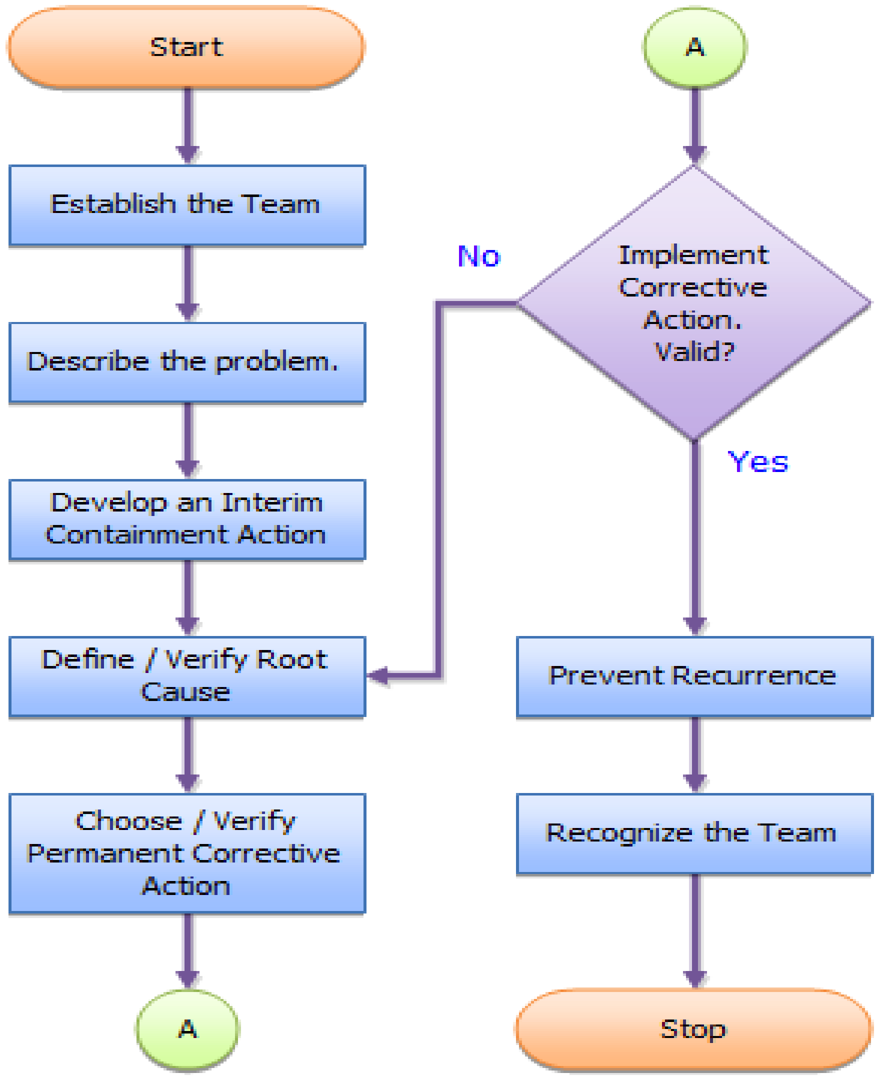


Figure 5.4 Activity Cycle Diagram for Single Server Queue.

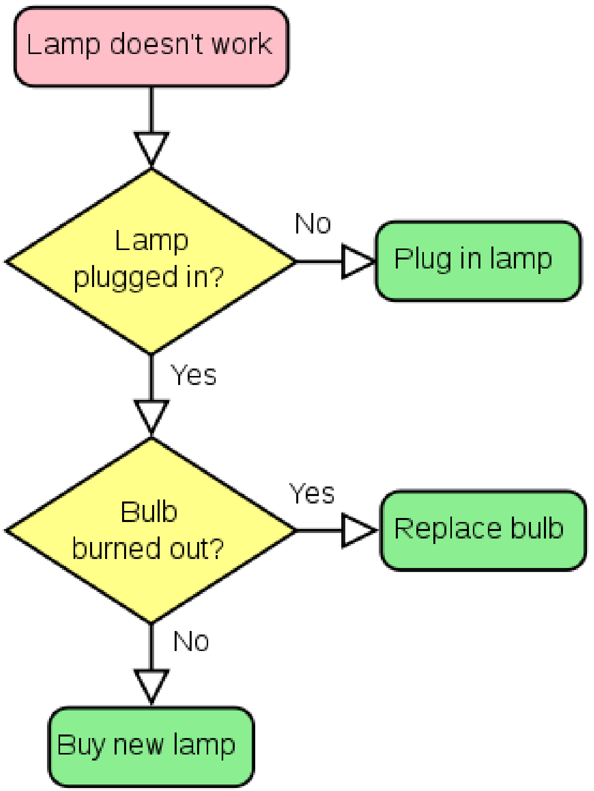
Flow chart


Start
Establish the Team
Describe the problem.
Develop an Interim Containment Action
Define / Verify Root Cause
Choose / Verify Permanent Corrective Action
A
A
Implement Corrective Action. Valid?
No
Yes
Prevent Recurrence
Recognize the Team
Stop

Flowcharts Definition

- A visual modelling technique shows the logical sequence of set of processes in order to give an idea about how a system with processes works.

- This diagrammatic representation can give a step-by-step solution to a given problem.



Why We Use Flowcharts

- Create visual maps of a process.
- Help with planning a project.
- Quality improvement tool.
- Identify processes that need improvement.
- Identify unnecessary/ problem steps in a process.
- Good communication tool.

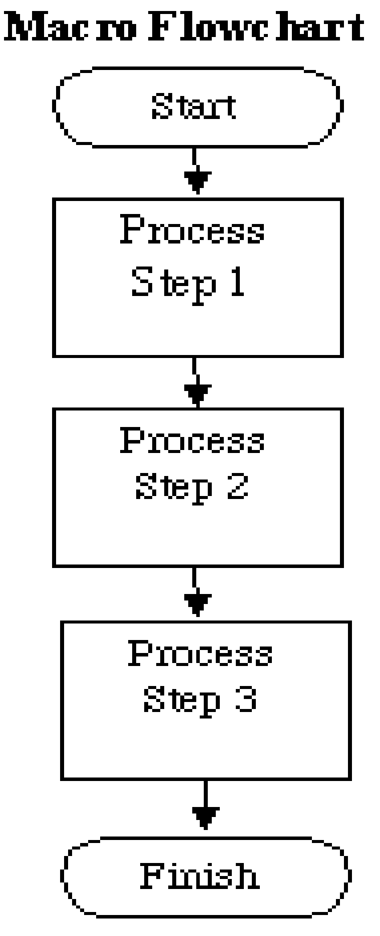
Flowchart Field of Applications

- **Airlines**
  - Used to identify logic of schedules
- **Airports**
  - Help improve airport operations
- **Warehouses**
  - Identify logic of goods in and out operations
- **Education**
  - Curriculum flowcharts
  - Student flow through process
- **Hospitals**
  - Patient flow
  - Medical processes

Types of Flowcharts

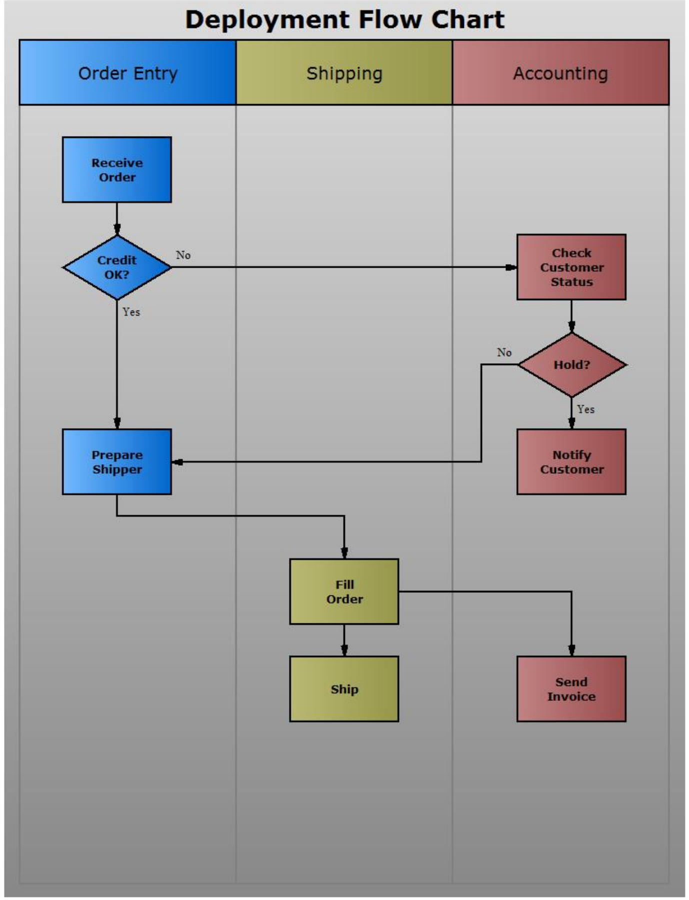
Macro Flowcharts

- High level perspective
- Steps in a process connected by a flow line
- Identifies problems in the process

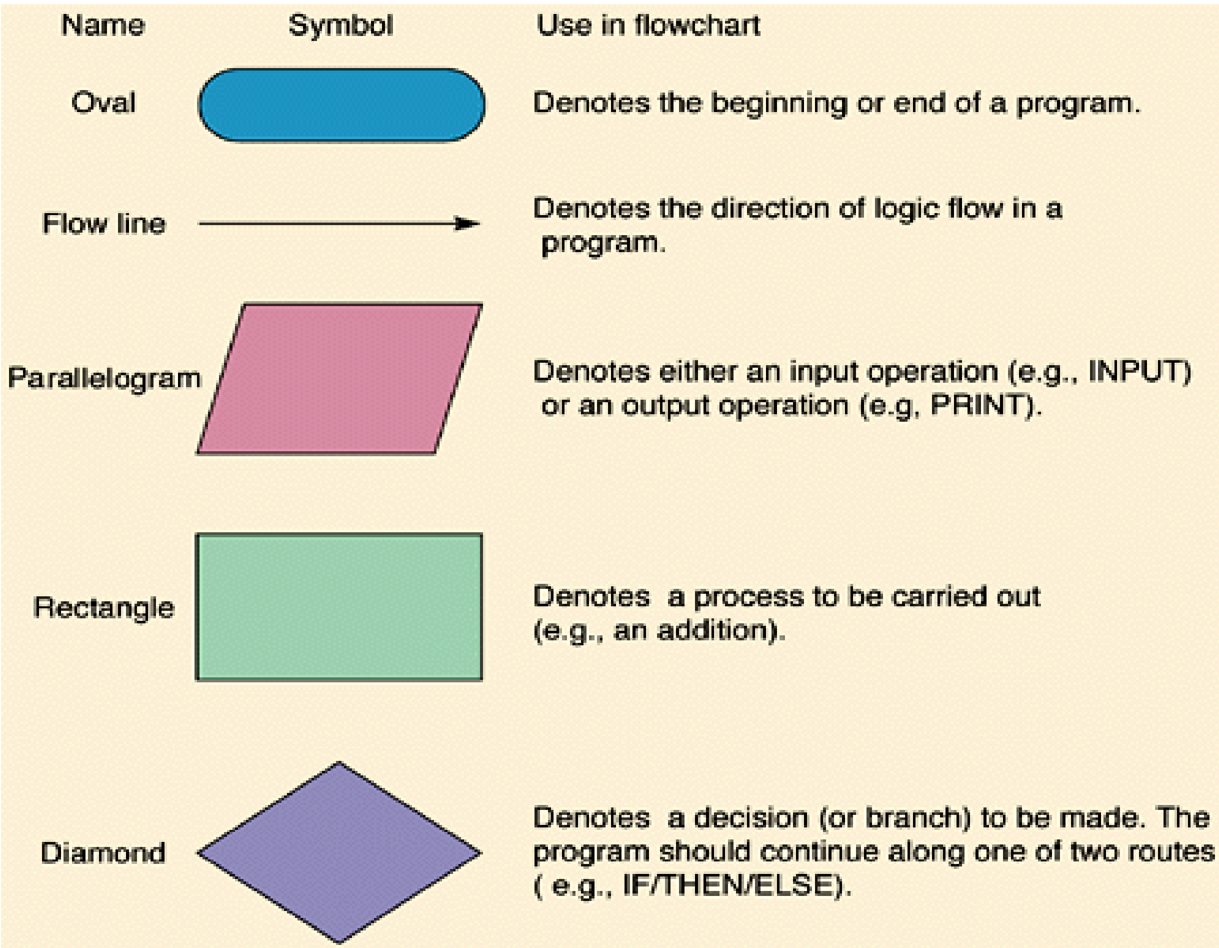


Types of Flowcharts (Con't)

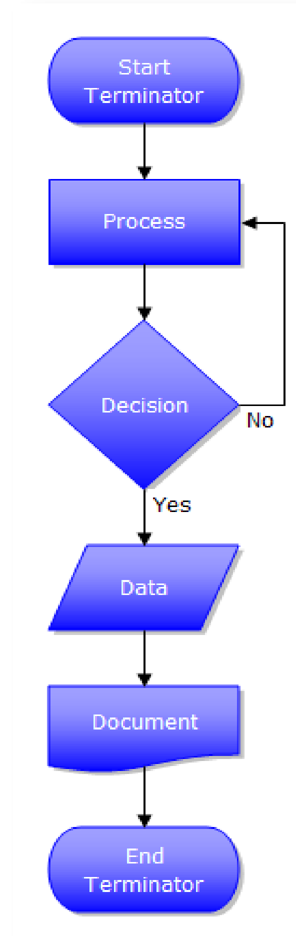
- Deployments Flowcharts
  - Much more detailed
  - Assign the steps to the person who performed them
  - Shows how members of the project team are deployed, or used throughout the project.



Popular Flowchart Symbols

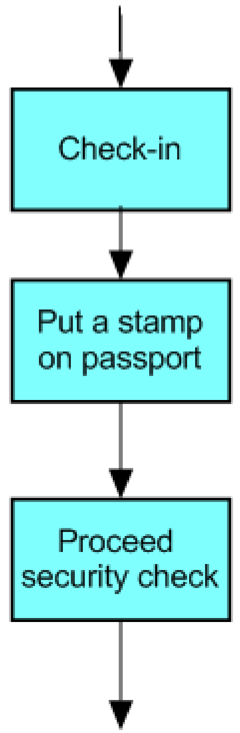


Name	Symbol	Use in flowchart
Oval		Denotes the beginning or end of a program.
Flow line		Denotes the direction of logic flow in a program.
Parallelogram		Denotes either an input operation (e.g., INPUT) or an output operation (e.g, PRINT).
Rectangle		Denotes a process to be carried out (e.g., an addition).
Diamond		Denotes a decision (or branch) to be made. The program should continue along one of two routes (e.g., IF/THEN/ELSE).

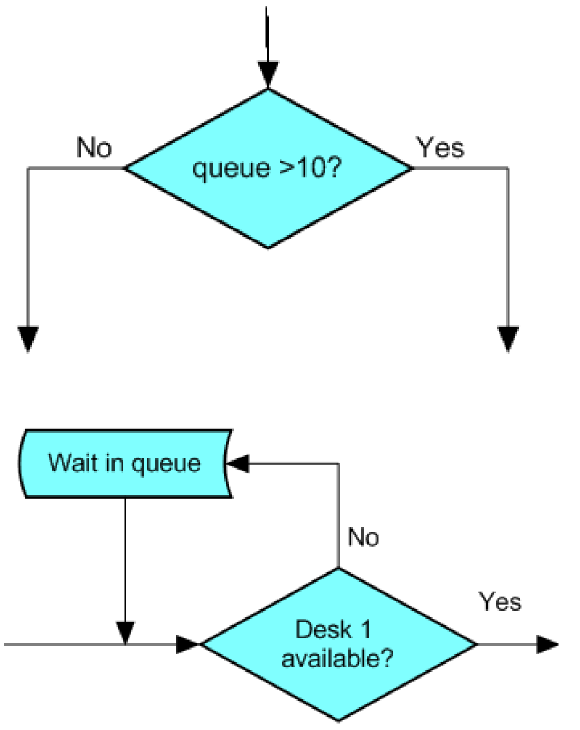


Basic Structures in Flowchart

1. Sequence

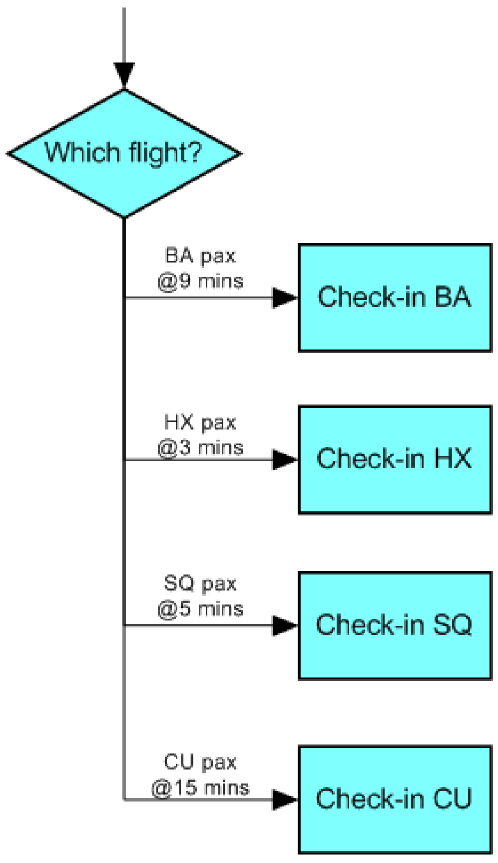


2. Selection

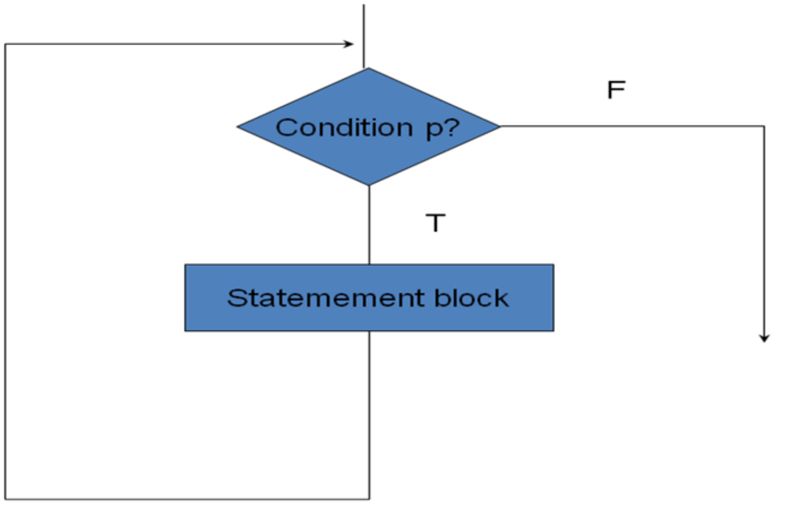


Basic Structures in Flowchart (Con't)

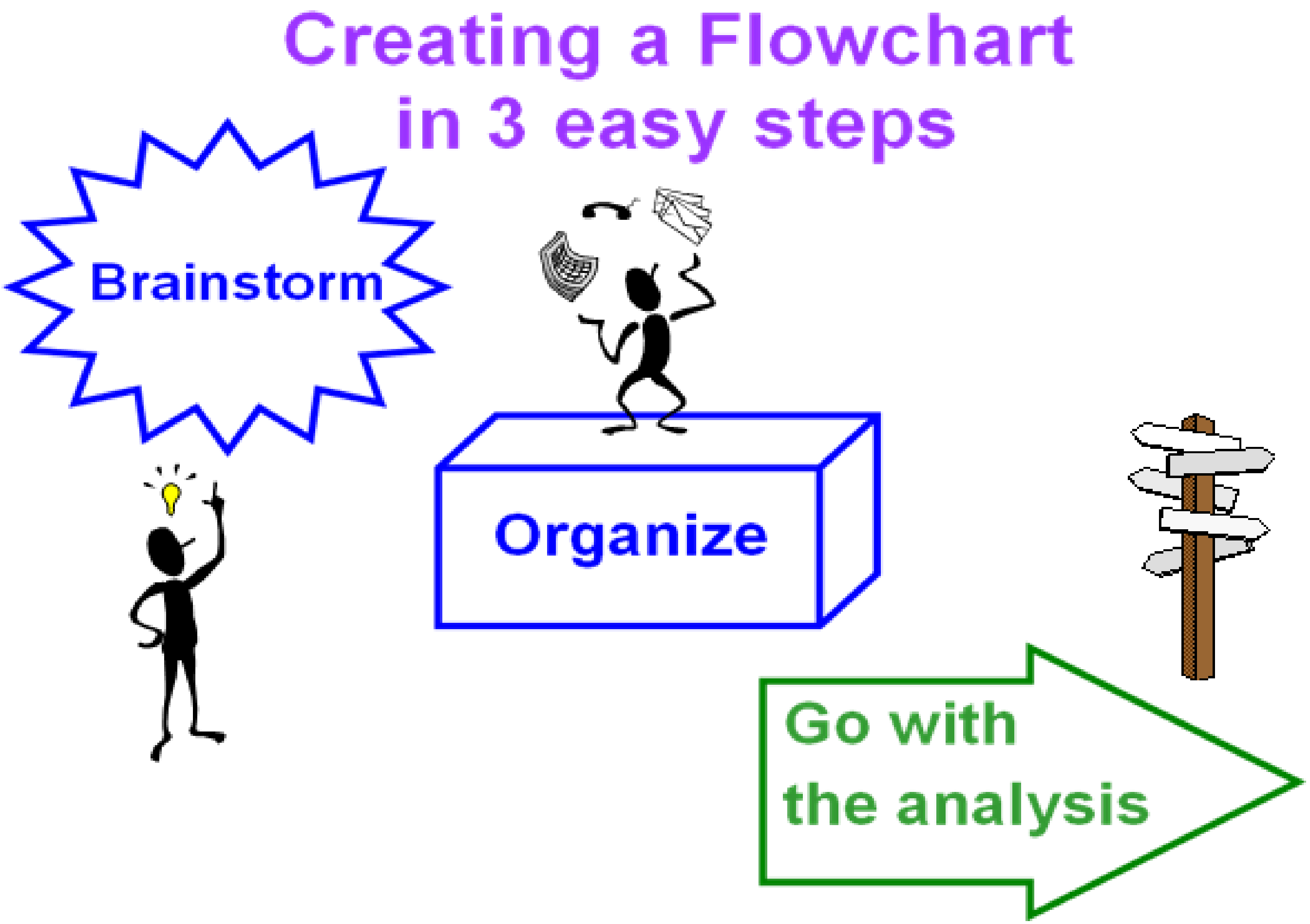
3. Case Selection



4. Repetition



Creating Flowcharts

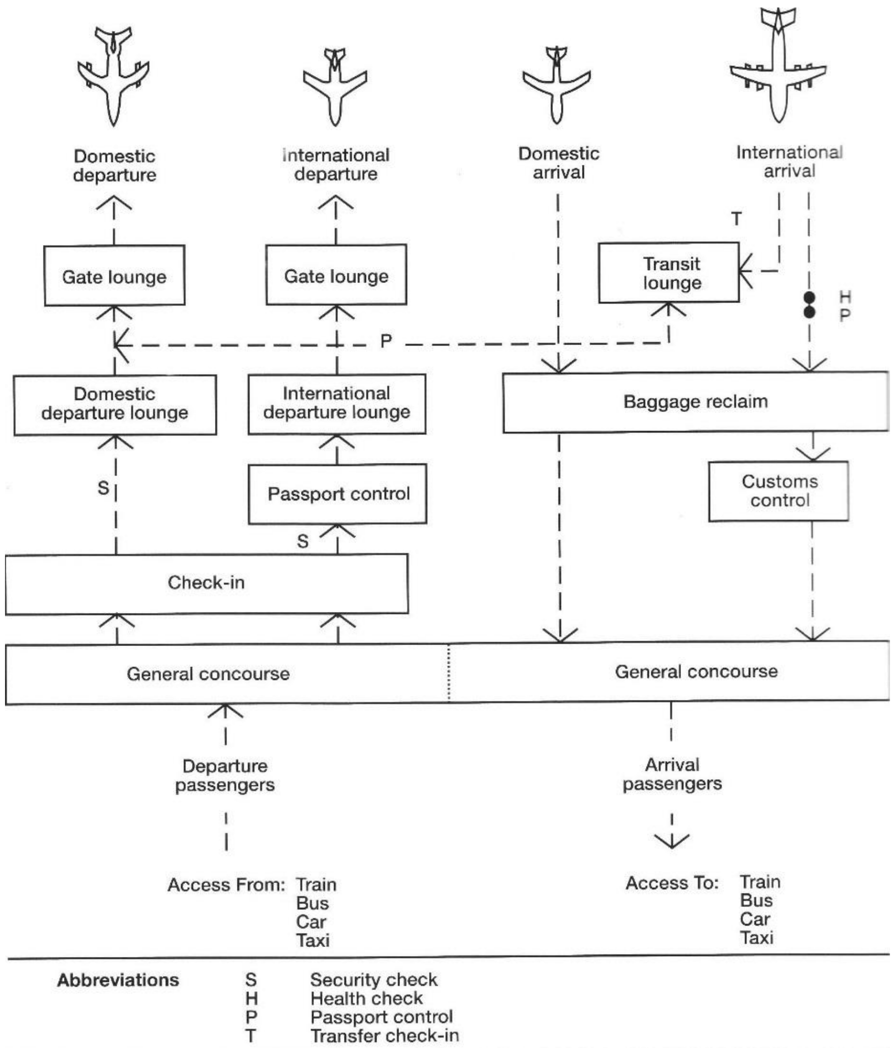


Flowchart Software

- Some tools offer special support for flowchart drawing such as Microsoft Visio, Draw.io, ClickChart.

- Many software packages exist that can create flowcharts automatically, either directly from source code, or from a flowchart description language. On-line Web-based versions of such programs are available

Simplified Flowcharts of How International Airports Generally Function

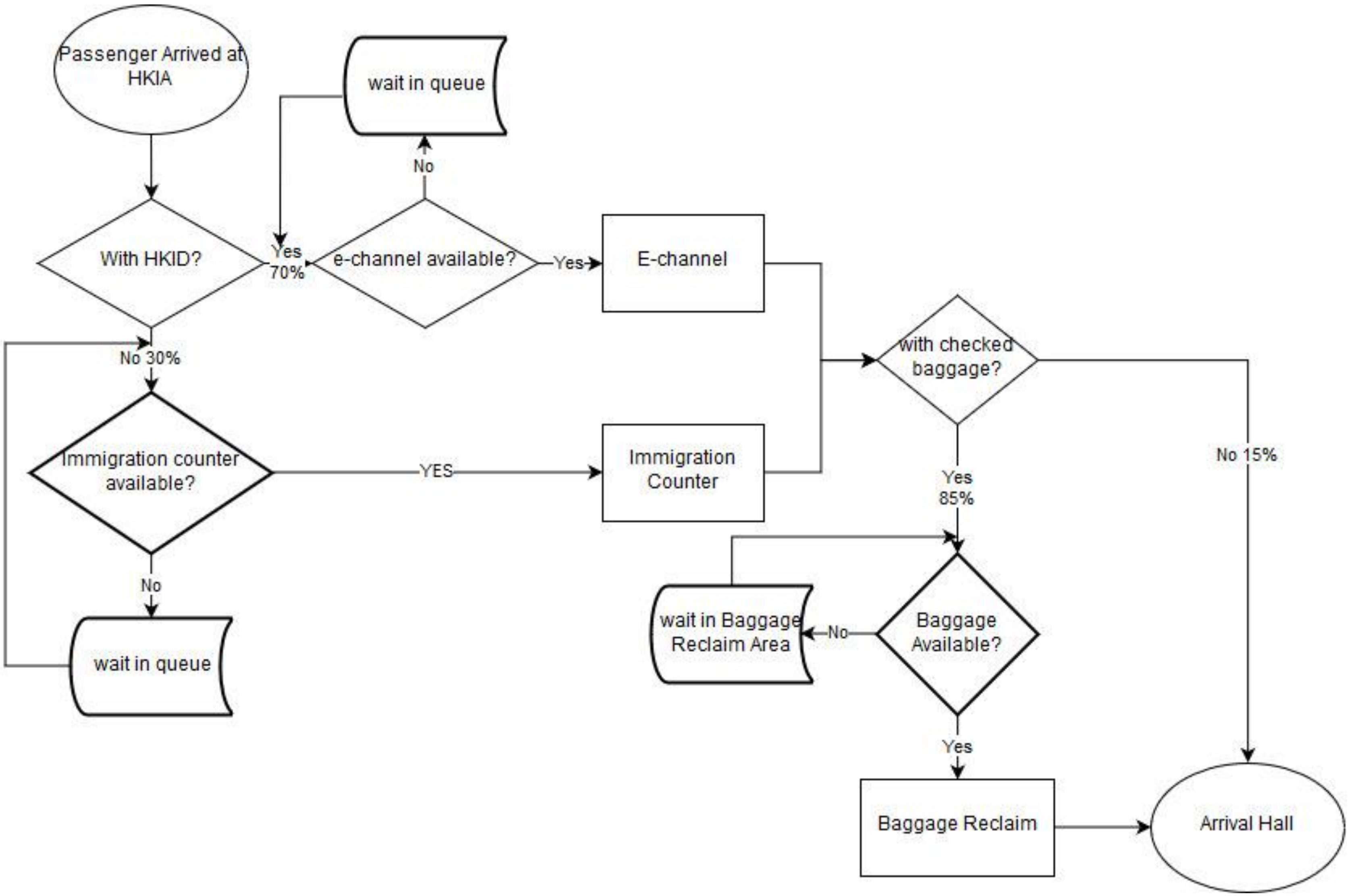


http://tonyhwijaya.wordpress.com/2011/10/03/simplified-flowcharts-of-how-international-airports-generally-function/

Case study1: Flow chat for Arrival PAX

- Passengers arrived at HKIA.
- Passengers with HKID can pass through e-channel for immigration.
- Passengers without HKID should provide a valid passport to pass through immigration counter.
- 70% of passengers are holding HKID.
- Passengers with checked baggage need to collect their baggage at Baggage Reclaim area.
- Passengers without checked baggage can pass through the above area.
- 85% of all passengers have checked baggage.
- All passengers will then proceed directly to arrival hall.

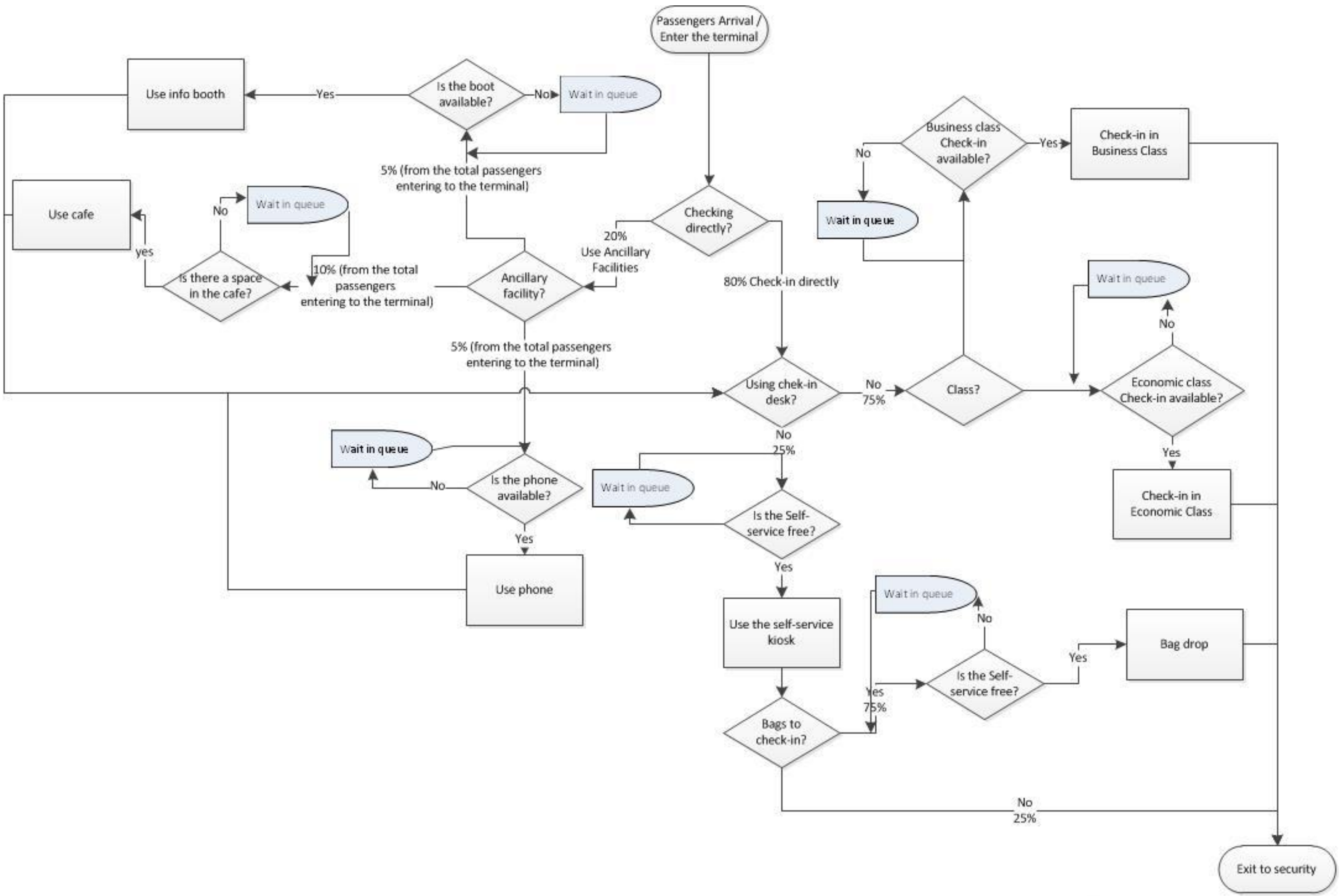
Flow Chart for Arrival PAX



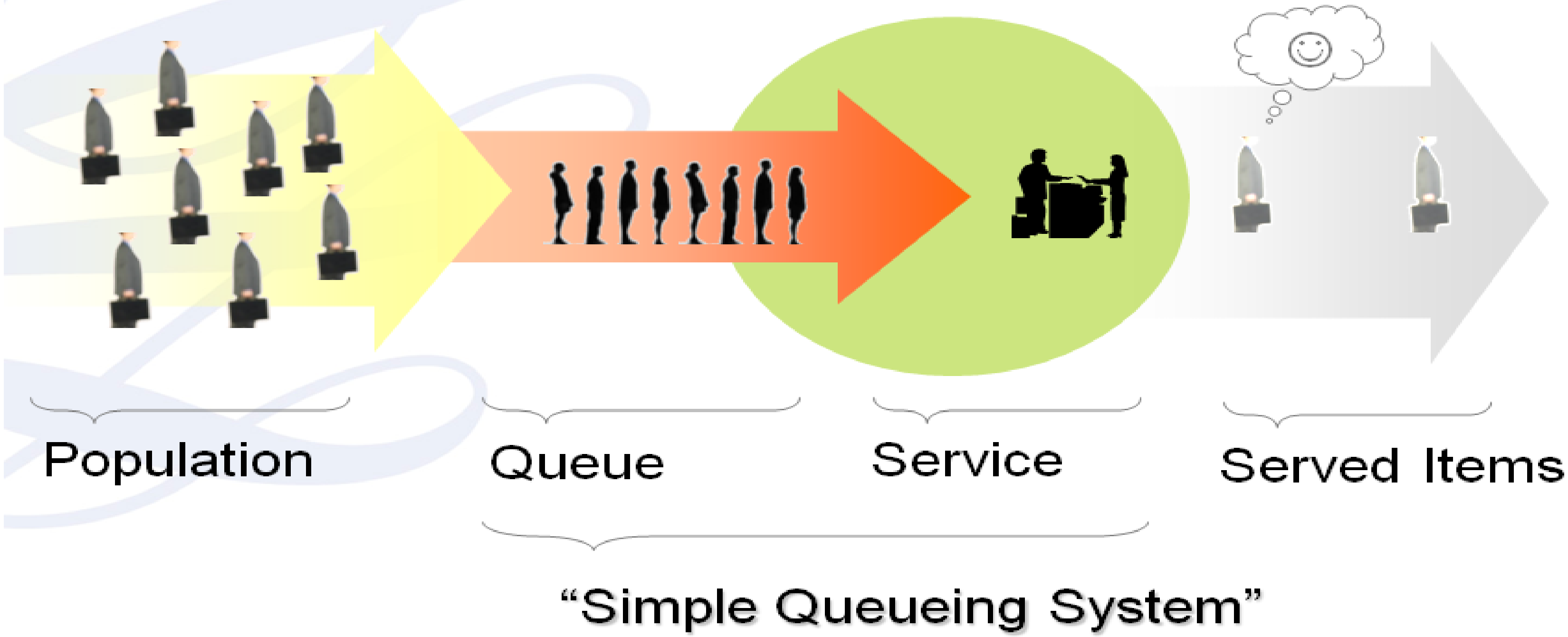
Case study2: A check-in process

- Passengers enter the terminal (only one terminal is considered)
- All passengers have the same flight.
- All passengers need to check-in in the terminal.
- Passengers arrive at the terminal up to 3 hours before the flight is scheduled to board.
- As soon as the passengers enter the terminal, they are able to either use the ancillary facilities or proceed directly to check-in; 80% of passengers will proceed directly to check-in, whilst other passengers will use the phone (in 5% of cases). All remaining passengers will use the cafe (for passengers who are "hungry" and have sufficient time to board; in 10% of cases) or information booth (for those who are "inexperienced" at this particular airport and are willing to ask for assistance; in 5% of cases).
- All passengers (including those who have already used any of the ancillary facilities) will then either choose to enter a check-in desk queue; (in 75% of cases), or use the self-service kiosk.
- Those willing to use the self-service kiosk (in 25% of cases) will then need to proceed to the bag drop counter if they still have bags to check-in (75% they do have bags).
- Of those willing to join the check-in queue, passengers will choose the appropriate check-in queue based on their class of travel:
  - There is one queue for each of the check-in areas, with single service counter for business class, two counters for economy class and three counters for bag drop, and two self-service check-in kiosks.
- All checked in passengers will then proceed directly to security.

Flow Chart for the above Check-in

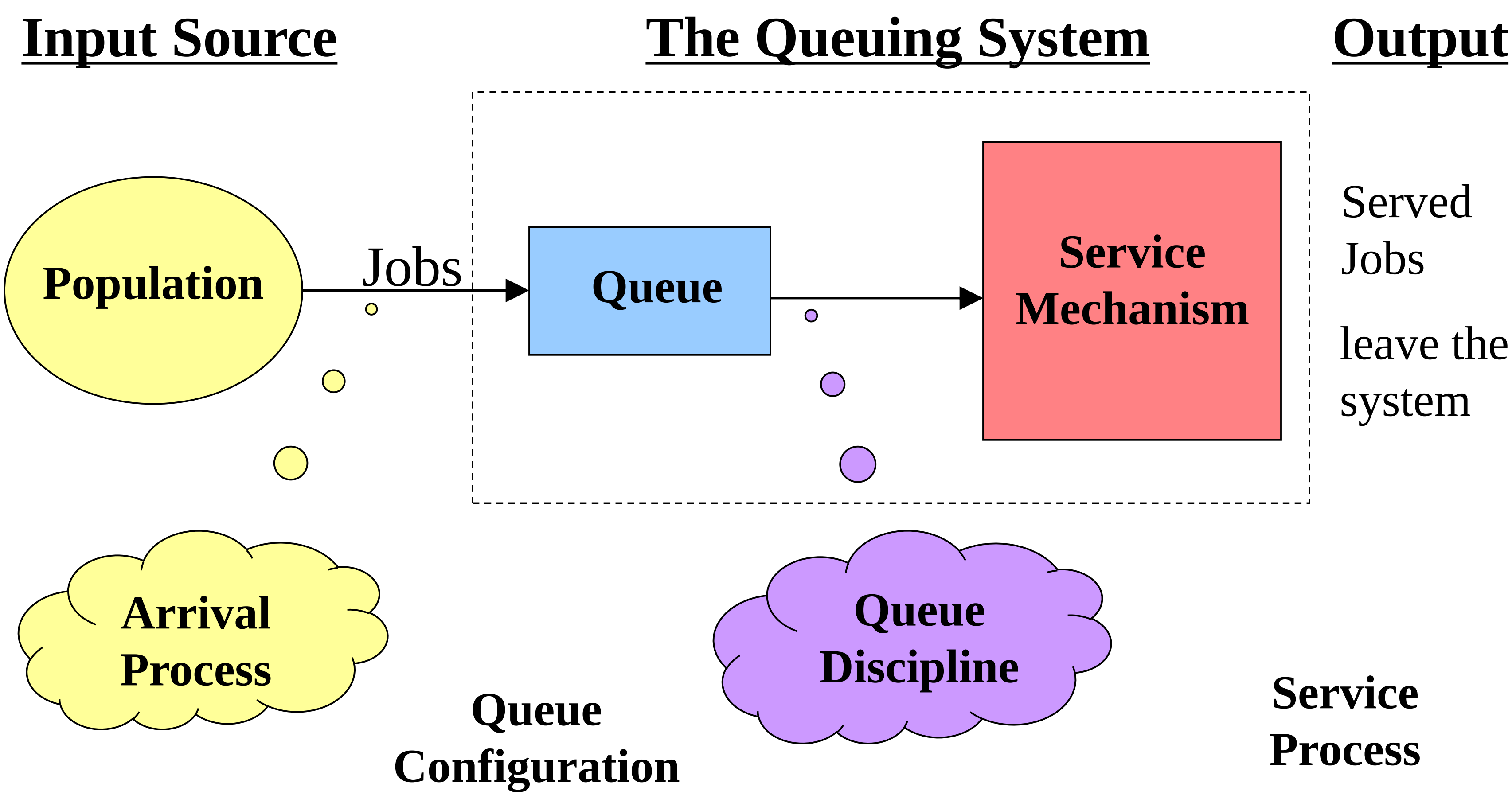


Basic Queueing Process



- Customers are generated from a source
- On arrival at a service facility, they can
  - start service immediately or
  - wait in a queue if the facility is busy
- When a facility completes a service,
  - it automatically "pulls" a waiting customer, if any, from the queue
  - it becomes idle, if the queue is empty

Components of a Basic Queueing Process

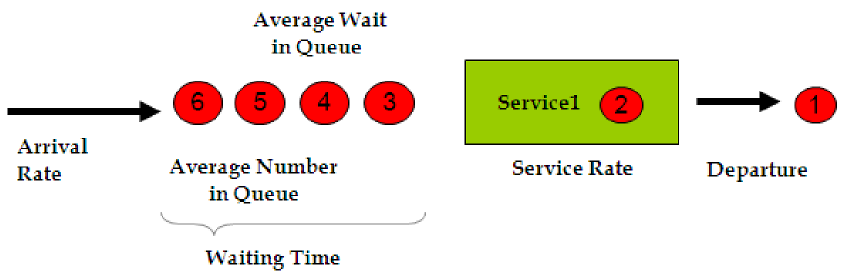


Characteristics of Queueing Models

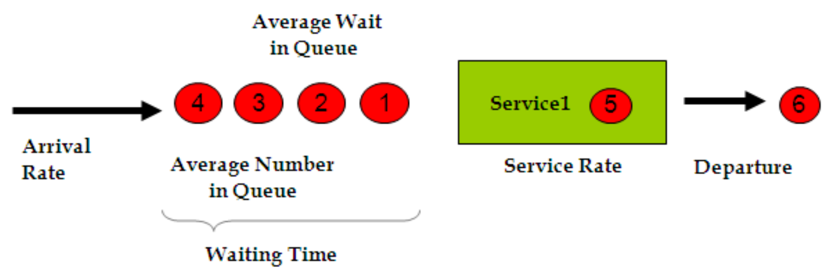
- Input process
  - the arrival of a request for services, often represented by the inter-arrival time between successive requests.
- Service mechanism
  - number of servers and service time
- Inter-arrival and service time can be probabilistic or deterministic
- Queue size
  - may have finite size or infinite (mail order facilities)
- Queue discipline
  - the order in which customers are selected (FCFS, LCFS, SIRO, Priority)
- The source from which requests of services are generated may be
  - finite (e.g., machines requesting the service of a repairperson) or
  - infinite (e.g., calls arriving at a telephone exchange)

Popular Service Patterns

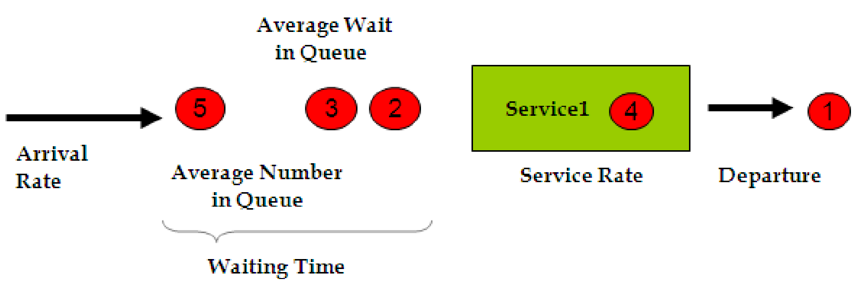
FCFS



LCFS

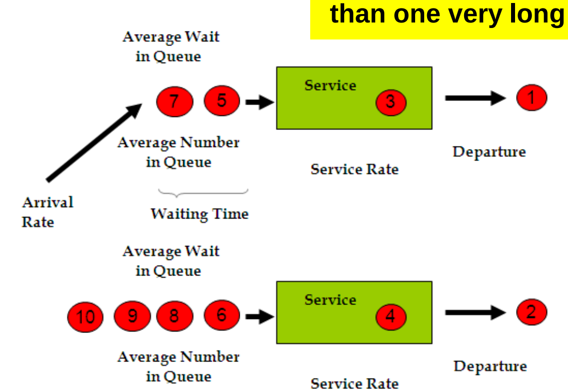


Priority

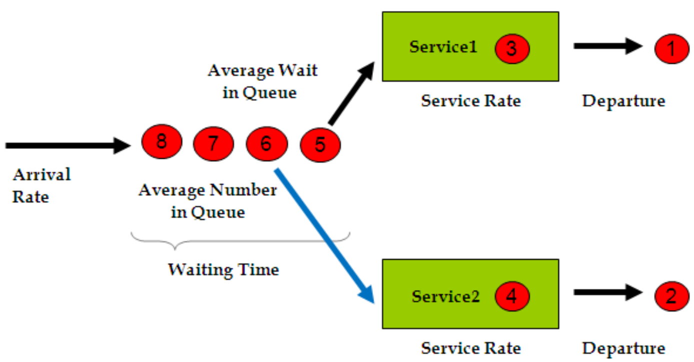


Shortest Queue

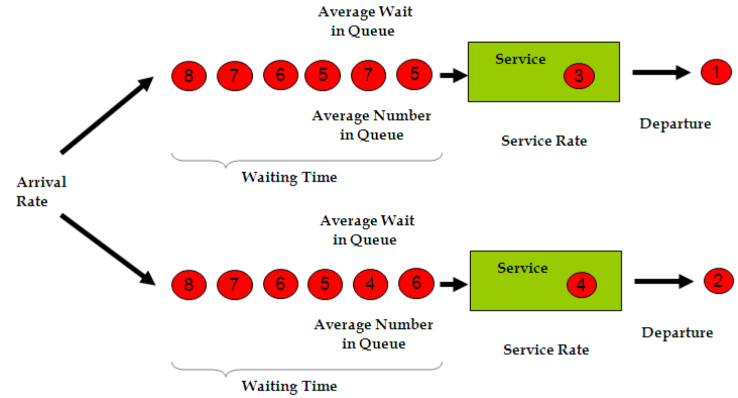
Several medium-length lines are less intimidating than one very long line



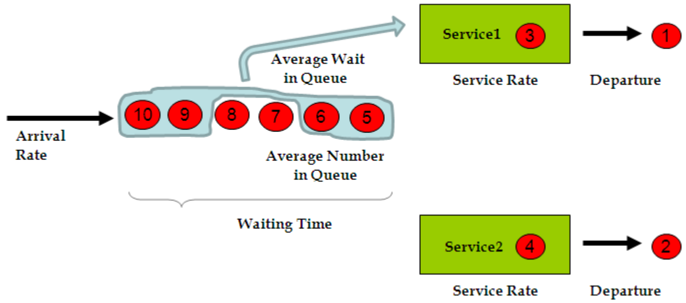
Circulate



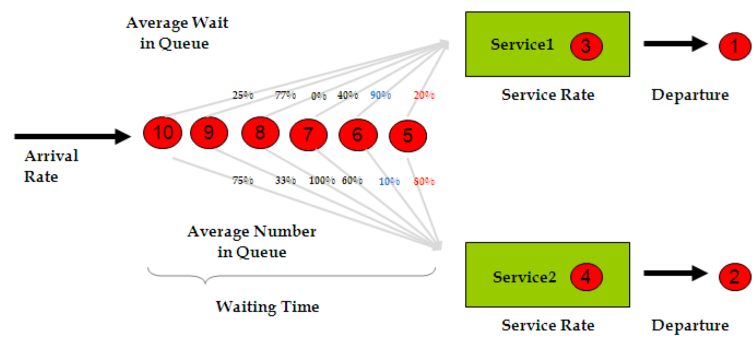
Uniform



Label



Percent

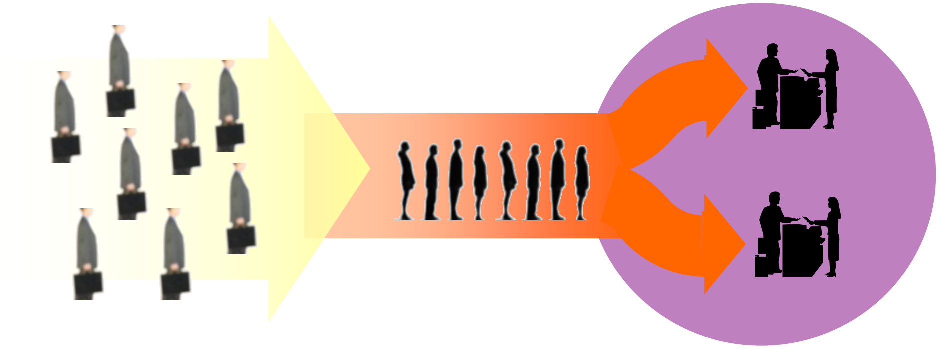


Single vs. Multiple Queues

- Would you ever want to keep separate queues for separate servers? What is the difference?

Single queue

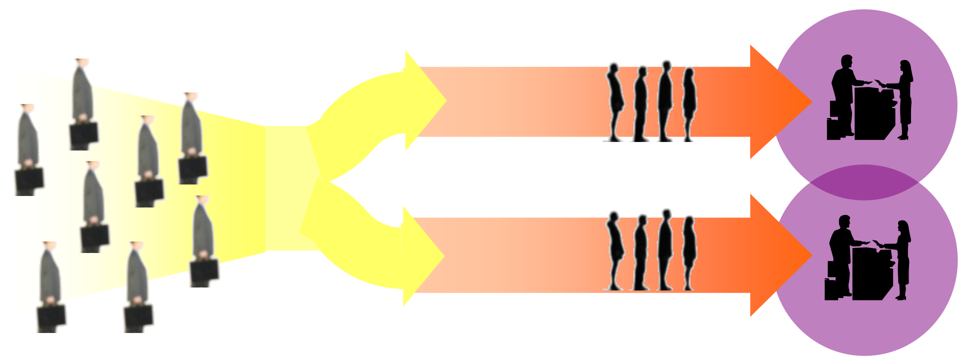
(e.g. two cashiers at local Sainsbury with one queue)



vs.

Multiple queues

(e.g. multiple queues at multiple gas pumps)



Single vs. Multiple Queue Configuration

Single Line Advantages	Multiple Line Advantages
1. **Guarantees fairness** • FCFS applied to all arrivals 2. **No customer anxiety regarding choice of queue** 3. **Avoids "cutting in" problems** 4. **The most efficient set up for minimising time in the queue** 5. **Jockeying (line switching) is avoided.**	1. **The service provided can be differentiated** • Ex. Supermarket express lanes 2. **Labor specialisation possible** 3. **Customer has more flexibility** 4. **Balking behaviour may be deterred** • Several medium-length lines are less intimidating than one very long line

Queue Measurements

- Typical measures
  - Server utilisation
  - Length of waiting lines
  - Delays of customers

- Applications
  - Determine the minimum number of servers needed at a service center
  - Detection of performance bottleneck or congestion
  - Evaluate alternative system designs

How to Reduce Queues

- Add more resources for the existing service in order to reduce the service time of the current service(s).
- Adopt more than one service with multiple queues if required.
- Manipulate the queue organisation (circulate, percent, uniform, label, etc)
- Change the service pattern (FCFS, LCFS, Priority, etc).

Proof of a Reduced Queue

- Less length of waiting lines
- Minimal delays of customers
- Reduced server idle time
- Optimised server utilisation
- Optimal number of both service centre(s) and server(s) of each centre.

References

- Robinson, S. (2014) Simulation: The Practice of Model Development and Use. 2nd edn. Palgrave, Macmillan
- Greasley, A (2004) Simulation Modelling for Business. Ashgate
- Shalliker, J, (2014) An Introduction to Simulation in the Service Industry using SIMUL8 2014 (release 21). SIMUL8 SIM8Serveices.pdf (inside folder Coursework)
- Taha, Hamdy A. (2010) Operations Research: An Introduction. 9th edn. Prentice Hall