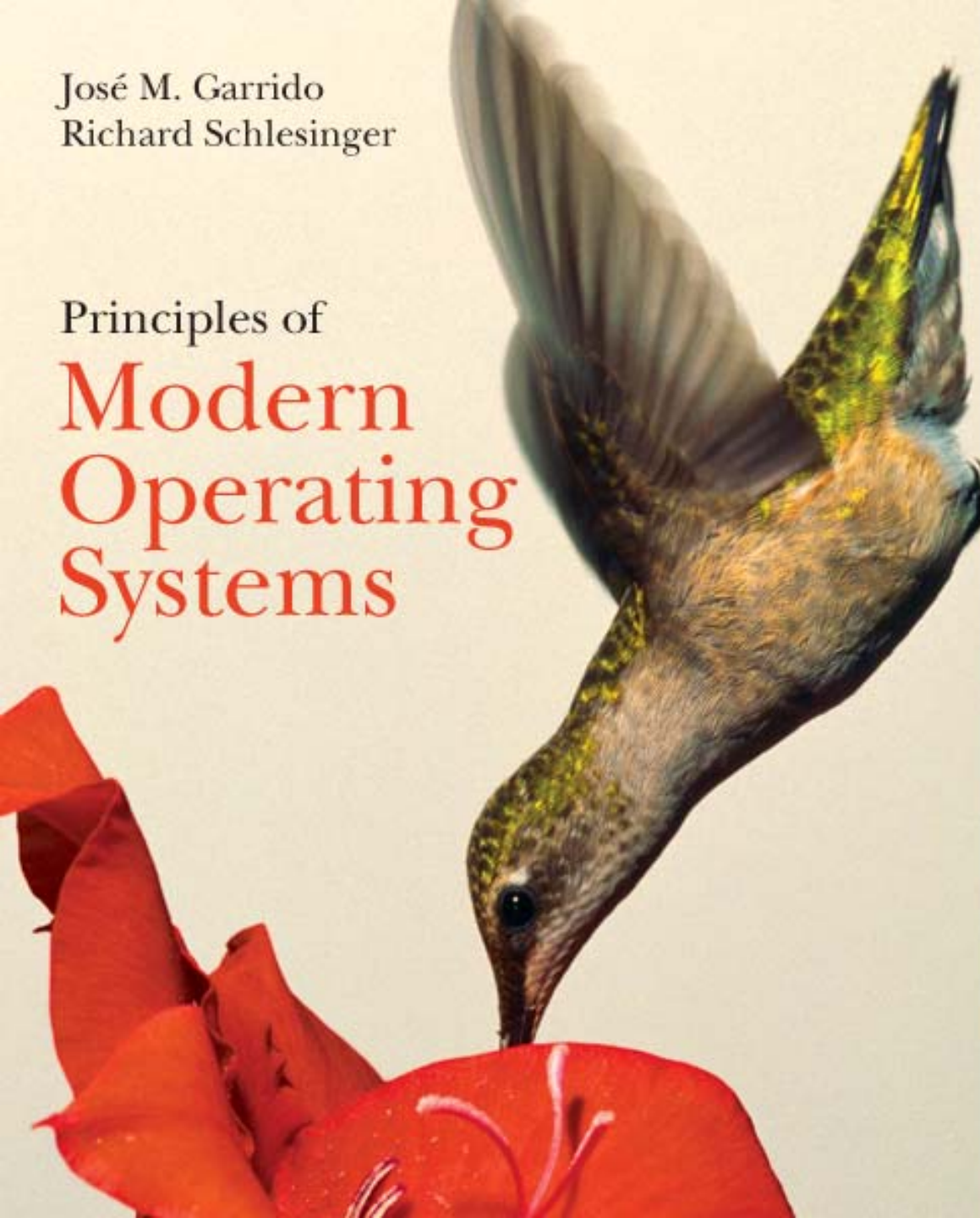


José M. Garrido
Richard Schlesinger

Principles of
**Modern
Operating
Systems**



Chapter 3

System Performance and Models

Systems and Models

- A system is the part of the real world under study. Composed of a set of entities interacting among themselves and with the environment.
- A model is an abstract representation of a system.
- The system behavior is dependent on the input data and actions from the environment.

Abstraction

- The most important concept in analysis and design
- A high-level description of a collection of objects
- Only the relevant properties and features of the objects are included.

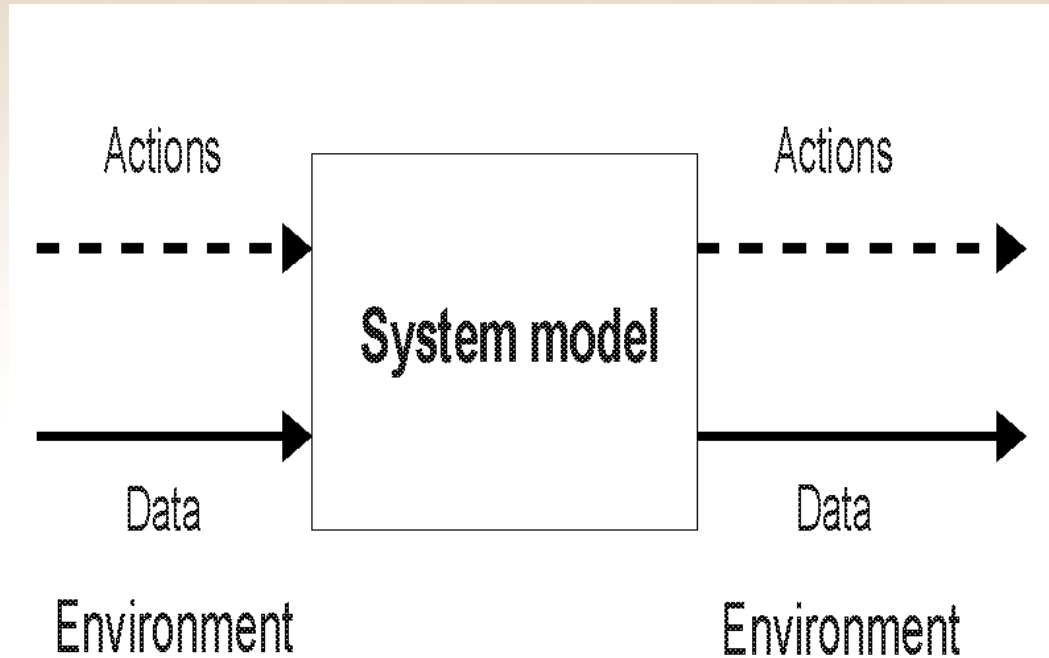
System

A system has:

- Structure
- Behavior

The model of a system is simpler than the real system in its structure and behavior. But it should be equivalent to the system.

A High-Level Model



Using Models

A user can:

- Manipulate the model by supplying it with a set of inputs
- Observe its behavior or output
- Predict the behavior of the real system by analyzing the behavior of the model .

Behavior of a Model

Depends on:

- The passage of time
- Input data
- Events generated by the environment

Types of Model

The most general categories of models are:

- Physical models (scale models)
- Graphical models
- Mathematical models.

Mathematical models are the most flexible ones and are the ones studied here.

Solutions to Mathematical Models

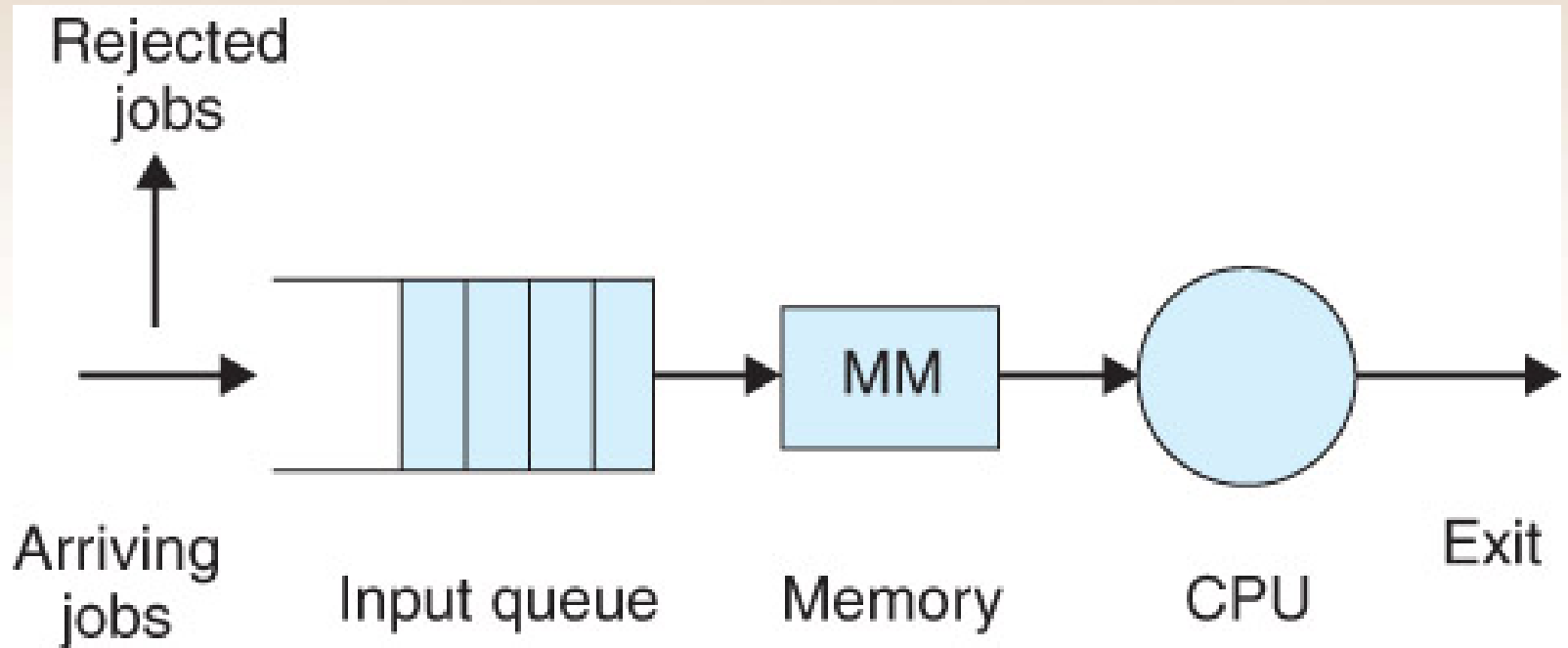
- Analytic, the solution is a set of expressions that define the behavior of the model.
- Numeric, mathematical techniques are used to derive values for the model behavior within given intervals.
- Deterministic and stochastic models are solved with numerical techniques.

Models with Uncertain Behavior

Models are further categorized as:

- Deterministic - models that display a completely predictable behavior (with 100% certainty)
- Stochastic - models that display some level of uncertainty in their behavior. This random behavior is implemented with random variables.

Model of a Simple Batch System



Stochastic Models

- A stochastic model is one which includes some uncertainty in its behavior.
- One or more attributes (implemented as random variables) change value according to some probability distribution.
- Example of random variables in the simple batch OS model:
 - Inter-arrival time of jobs
 - CPU service time of jobs

Random Variables

- The values of most workload parameters change randomly, and are represented by a probability distribution.
- For example, in a model of a computer system, the inter-arrival intervals of jobs usually follow an exponential distribution. Other probability distributions that are used in the models discussed in this book are: normal, uniform, and Poisson.
- In the simulation models considered, most workload parameters are implemented as random variables.

Simulation Models

- A simulation model is a mathematical model implemented with a general-purpose programming language, or a simulation language.
- A simulation run is an experiment carried out for some observation period and with the simulation model to study the behavior of the model.

Continuous and Discrete Models

- Continuous models change their state continuously with time. Mathematical notation is used to completely define the behavior. For example, the free-falling object.
- Discrete models only change their state at discrete instants. For example, arrival of a job in the Simple batch OS model.

Discrete Simulation

- Event approach
- Activity approach
- Process interaction approach
 - Psim3 and PsimJ uses the full power of Object orientation and process interaction simulation
 - GPSS has limited support
 - Simscript has limited support
 - Modsim III has good support

Simulation Results

For every simulation run there are two types of output:

- Trace - sequence of events that occur during the simulation period
- Performance measures - summary statistics about the simulation.

The Process Interaction Approach

- A simulation model using this approach consists of a description of how different active entities are going to interact among themselves, as time passes.
- A simulation run consists of creating and starting the processes interacting among themselves, synchronizing and using resources.

Advantages of the Process Style of Simulation

- Compatible with the Object-Oriented approach to modeling and programming, every process is an active object.
- Suitable for modeling large and complex systems.
- C++, Java and/or a higher-level object-oriented simulation language are used to implement the models.

Studying Operating Systems

- An operating system is a large and complex software system
- To study the structure and behavior of an OS is a very complicated task
- Modeling and simulation are used to study the different components and various aspects of an operating system.

Performance Measures

- Measures that indicate how well (or bad) is the system being studied carrying out its functions, with respect to some aspect
- In studying a system, usually several performance measures are necessary.

Approaches for Studying Performance

- Measurements on the real system
- Simulation models
- Analytical models

Performance: External Goals

- Goals that can be measured without looking at the internals of a system
- Examples
 - Maximize Work Performed (Throughput)
 - Minimize Response Time
 - Fairness
 - Scheduling processes
- Goals often conflict

Performance: Internal Goals

- Performance Goals for sub-systems that are internal to the computer(s)
- Examples
 - Maximize CPU Utilization
 - % time CPU is busy
 - Maximize Disk Utilization
 - % time Disk is busy
 - Minimize Disk Access Time
 - Time it takes to perform a disk request

Performance Study

- Define a set of relevant objectives and goals
- Decide on the following:
 - The performance metrics
 - The system parameters
 - The system factors
 - The system workloads

Workload on a System

- The performance measures depend on the current workload of the system
- The workload for a system can be characterized by another series of measures, which are made on the input to the system
- Errors in characterizing the workload may have serious consequences.

Workload Parameters

- Inter-arrival time
- Task size
- I/O request rate
- I/O service rate
- Memory request

System Parameters

- System memory
- Processor speed
- Number and type of processors
- Degree of multi-programming
- Length of time slice
- Number and type of I/O ports

Arrival Interval & Rate

- Arrival Interval is the time between 2 resource requests arriving
- Arrival Rate = $1 / \text{Arrival Interval}$
 - How often requests for a resource arrive
 - Denoted by λ

Service Time & Rate

- Service Time is the time to actually perform a request
 - e.g. 500 Msec / request
- Service Rate = $1 / \text{Service Time}$
 - Denoted by μ
 - e.g. 2 requests / second

Arrival Rate vs Service Rate

- Arrival Rate $<$ Service Rate
 - Resource can handle requests
 - Queuing Time will be small
 - If Arrival Rate significantly less
 - Resource may be under-utilized

Arrival Rate vs Service Rate

- Arrival Rate = Service Rate
 - Resource fully utilized
 - Resource can generally handle requests
 - But...Requests will often be queued

Arrival Rate vs Service Rate

- Arrival Rate $>$ Service Rate
 - Resource unable to handle all requests
 - Queuing Time may be large
 - If difference is large
 - Resource will be over-utilized

Arrival Rate vs Service Rate

- Both rates will change from one instant to another
- Queuing system & resource scheduling handle momentary imbalances
- Long-term imbalances indicate a system with a major resource utilization problem

Service Time

- Is the Service Time for a given request always the same?

Service Time

- Service Time for a given request can vary
 - A request to read data from a certain part of a disk will not always take the same amount of time.
- *For some resources, a request's Service Time may be affected by the previous requests that have been executed*

Relevant Performance Measures

- Throughput
- Capacity
- Response time
- Utilization
- Reliability
- Speedup
- Backlog

Meaning of Performance Measures

- The average number of jobs in the system
- The average number of jobs in the queue(s) (i.e., that are waiting)
- The average time that a job spends in the system
- The average time that a job spends in the queue(s)
- The CPU utilization
- Throughput - the total number of jobs serviced.

System Capacity

- The capacity of a system is determined by its maximum performance
- The nominal capacity of a system is given by the maximum achievable throughput under ideal conditions
- The usable capacity is the maximum throughput achievable under specified constraints.

Bottleneck

- The computer system reaches capacity when one or more of its servers or resources reach a utilization close to 100%.
- The bottleneck of the system will be localized in the server or resource with a utilization close to 100%, while the other servers and resources each have utilization significantly below 100%.

Modeling Bottleneck

The bottleneck of the computer system described here can be localized at the processor, the queue, or the memory.

- The queue may become full (reaches capacity) very often as the processor utilization increases.
- The memory capacity may also be used at capacity (100%).
- Thus, in any of the three cases, the processor, the queue, or the memory will need to be replaced or increased in capacity.

Efficiency and Reliability

- The efficiency of a system is the ratio usable capacity to nominal capacity
- The reliability of a system is measured by the probability of errors. Also defined as the mean time between errors.
- Availability is the fraction of time that the system is available for user requests, also called the system uptime.

A More Complete Model of a Computer System

