Chapter 2

A. Types of Systems

B. Other Classifications of Systems

A. Types of Systems

C. Types of Systems

Systems can be categorized based on

- ✓ different criteria, such as their nature (physical or abstract),
- ✓ purpose, or
- ✓ complexity.

Here are the key types relevant to systems analysis and design:

1. **Information Systems (IS)**
2. **Transaction Processing Systems (TPS)**
3. **Decision Support Systems (DSS)**
4. **Management Information Systems (MIS)**
5. **Executive Information Systems (EIS) or Executive Support Systems (ESS)**
6. **Expert Systems**
7. **Enterprise Resource Planning (ERP) Systems**
8. **Office Automation Systems (OAS)**
9. **Real-Time Systems**
10. **Physical Systems**

1. Information Systems (IS)

Definition: A system that collects, processes, stores, and distributes data to support decision-making, coordination , and control within an organization.

Information systems typically involve hardware, software, data, people, and procedures.

Purpose To provide accurate, timely, and relevant information for organizational tasks.

Components Databases, software applications, user interfaces, and network infrastructure.

Characteristics: Focus on data management, reporting, and supporting business processes.

Examples

- **A Customer Relationship Management (CRM) إدارة علاقات العملاء system to track customer interactions.**
- **A Management Information System (MIS) that generates reports for managers.**

2. Transaction Processing Systems (TPS)

Definition: A type of information system designed to process routine, high-volume transactions efficiently and accurately. TPS handles day-to-day operational data.

Purpose: To record and process transactions (e.g., sales, payments) in real-time or batch mode.

Components Input devices (e.g., POS terminals) نقاط البيع , processing software, and databases.

Characteristics High reliability, speed, and accuracy; supports large numbers of repetitive transactions.

Examples

- **A Point of Sale (POS) system in a retail store for processing customer purchases.**
- **A Banking System for handling deposits الودائع, withdrawals, and transfers.**

3. Decision Support Systems (DSS)

Definition **Systems that provide analytical tools and data to support complex decision-making by managers or analysts.**

Purpose **To assist in solving semi-structured or unstructured problems using data analysis and modeling.**

Characteristics: Interactive, uses data visualization, and often incorporates what-if analysis.

Examples

- A Financial Forecasting التنبؤ System to predict revenue trends ربح .
- A Supply Chain Optimization System to determine optimal inventory المخزون levels.

4. Management Information Systems (MIS)

Definition **Systems that provide managers with summarized reports and data to support routine decision-making and planning.**

Purpose **To generate periodic reports for monitoring and controlling organizational performance.**

Characteristics: Focus on structured data and predefined reports for middle management.

Examples

- **A Sales Performance Dashboard showing monthly sales metrics مقاييس .**
- **An Inventory Management System reporting stock levels.**

5. Executive Information Systems (EIS) or Executive Support Systems (ESS)

Definition Systems designed for senior executives to provide high-level, summarized data for strategic decision-making.

Purpose To deliver key performance indicators (KPIs) and external data for long-term planning.

Characteristics User-friendly interfaces, high-level summaries, and external data integration.

Examples

- A Business Intelligence Dashboard showing market trends اتجاهات السوق and company performance.
- A Strategic Planning System for analyzing competitor data..بيانات المنافسين

6. Expert Systems

Definition Systems that emulate تحاكي the decision-making ability of a human expert by using knowledge bases and inference rules.

Purpose To solve complex problems in specialized domains.

Characteristics Uses artificial intelligence, rule-based logic, and domain-specific knowledge.

Examples

- **A Medical Diagnosis System suggesting treatments based on symptomsأعراض.**
- **A Financial Advisory System recommending investments.**

نظام استشاري مالي يوصي بالاستثمارات.

7. Enterprise Resource موارد المؤسسة Planning (ERP) Systems

Definition **Integrated systems that manage core business processes across an organization, such as finance, HR, and supply chain.**

Purpose **To streamline تبسيط operations and provide a unified view of organizational data.**

Characteristics Cross-functional, scalable, and supports multiple departments.

متعددة الوظائف، وقابلة للتطوير، وتدعم أقسام متعددة

Examples

- SAP (Systems, Applications, and Products in Data Processing) or Oracle ERP (Enterprise Resource Planning) for managing business operations.
- A Manufacturing ERP to coordinate production and inventory.

8. Office Automation Systems (OAS)

Definition **Systems that automate routine office tasks to improve productivity and communication.**

Purpose **To enhance clerical and administrative efficiency.الكفاءة الإدارية والكتابية**

Characteristics Focus on communication, document processing, and scheduling.

Examples

- A Document Management System for storing and retrieving files.
- An Email and Calendar System like Microsoft Outlook.

9. Real-Time Systems

Definition Systems that process data and provide outputs within strict time constraints.

Purpose To handle time-critical operations where immediate response is essential.

Characteristics High reliability موثوقية and responsiveness,استجابة وسرعة often used in safety-critical applications.

Examples

- **An Air Traffic Control System for monitoring aircraft.**
- **A Medical Monitoring System for real-time patient data.**

10.Physical Systems

Definition Tangible ملموسة systems involving physical components and processes, often automated with technology. يتم أتمتتها باستخدام التكنولوجيا.

Purpose To manage physical operations or processes.

Characteristics Involves hardware and physical processes, often integrated with information systems.

Examples

- A Manufacturing Assembly Line controlled by automated systems.
- A Smart Home System controlling lighting and temperature.

D. Other Classifications of Systems

D. Other Classifications of Systems

In addition to the above, systems can be classified based on other perspectives:

1. Open vs. Closed Systems:
2. Abstract vs. Physical Systems:
3. Deterministic vs. Probabilistic Systems

Open Systems

- An open system is a system that interacts with its environment, exchanging information, energy, or resources.
- It continuously receives inputs from the external environment, processes them, and produces outputs that are sent back to the environment.

- **This interaction allows open systems to adapt and evolve based on external changes.**

Example: An e-commerce system is a classic open system. It receives customer orders (input) from users, processes them through inventory checks and payment gateways, and delivers products or services (output) back to the customer.

It also interacts with external systems like payment processors, shipping companies, and customer feedback platforms.

Characteristics:

- **Dynamic and adaptive to environmental changes.**
- **Reliant on external inputs (e.g., data, resources).**
- **Outputs impact the environment (e.g., delivering goods or generating reports).**

Closed Systems

- A closed system operates independently of its external environment, with minimal or no exchange of matter, energy, or information.
- It is self-contained, designed to function in isolation, and its processes are not influenced by external factors.

Example: Closed chemical reactor.

Once the reactants are placed inside and closed, the chemical reactions occur without external interference (no new materials enter, and no products leave during the process). The system operates based on its initial conditions until the reaction is complete.

Characteristics:

- Isolated from external influences.
- Fixed inputs and outputs within the system.
- Often used in controlled environments like scientific experiments or mechanical systems.

Key Differences

Aspect	Open System	Closed System
Interaction	Interacts with the environment	Isolated from the environment
Exchange	Exchanges matter, energy, or information	No or minimal exchange

Aspect	Open System	Closed System
Adaptability	Adapts to external changes	Operates based on internal conditions
Examples	E-commerce, ERP systems like Oracle/SAP	Sealed reactor, self-contained machine

2. Abstract vs. Physical Systems:

Abstract Systems:

Definition: Abstract systems are conceptual models or representations that describe relationships, processes, or structures without physical form. They exist as ideas, theories, or logical constructs.

Characteristics:

- **Non-tangible, existing in the mind or as mathematical/logical frameworks.**
- **Often used to simplify complex phenomena for analysis or design (e.g., algorithms, mathematical models, or software logic).**

- **Focus on rules, patterns, or relationships rather than physical components.**
- **Can be universal and independent of specific physical implementations.**

Examples:

- A flowchart for a computer program.
- Economic models like supply and demand.
- A mathematical equation describing gravity.

Use Cases: Used in computer science (e.g., data structures), theoretical physics, or system design to conceptualize processes before implementation.

Physical Systems:

Definition: Physical systems are tangible entities composed of physical components that interact according to natural laws or engineered designs.

Characteristics:

- Made of material objects (e.g., machines, circuits, or biological organisms.
- Operate in the physical world and are subject to physical laws (e.g., gravity, thermodynamics).

- **Can be observed, measured, and manipulated directly.**
- **Often have real-world constraints like energy, space, or material limitations.**

Examples:

- ✓ A car engine (mechanical system).
- ✓ A computer’s hardware (electronic system).
- ✓ The human circulatory system (biological system).

Use Cases: Studied in engineering, physics, and biology to design, analyze, or optimize real-world systems.

Key Differences:

Aspect	Abstract Systems	Physical Systems
Nature	Conceptual, non-tangible	Tangible, material-based
Existence	In thought, models, or simulations	In the physical world

Key Differences:

Aspect	Abstract Systems	Physical Systems
Examples	Algorithms, theories, diagrams	Machines, organisms, circuits
Constraints	Logical or theoretical limits	Physical laws, resources, space

Key Differences:

Aspect	Abstract Systems	Physical Systems
Purpose	To design, simulate, or understand	To function, operate, or interact

3. Deterministic vs. Probabilistic Systems:

Deterministic Systems:

Definition: A deterministic system is one where the output or behavior is entirely predictable based on the input and the system's rules. Given the same initial conditions, the system will always produce the same result.

نظام تكون فيه المخرجات أو السلوك متوقعة تمامًا بناءً على المدخلات

Characteristics:

- No randomness or uncertainty is involved..لا يوجد أي عشوائية أو عدم يقين
- The system's behavior is governed by fixed rules or equations.
- Outputs are precise and repeatable.
- Often modeled using mathematical equations or algorithms.

غالبًا ما يتم نمذجتها باستخدام المعادلات أو الخوارزميات الرياضية.

Examples:

- **A mechanical clock: Given the time and mechanism, the next tick is predictable.**

إن الساعة التالية يمكن التنبؤ بها.

- **A computer program with fixed inputs (e.g., calculating 2 + 2 always yields 4).**
- **Newton’s laws applied to planetary motion: Orbits can be predicted exactly.**

قوانين نيوتن المطبقة على حركة الكواكب: يمكن التنبؤ بالمدارات بشكل دقيق.

Use Cases: Used in engineering (e.g., designing bridges), physics (e.g., classical mechanics), and computer algorithms where predictability is essential.

تُستخدم في الهندسة (تصميم الجسور)، والفيزياء (الميكانيكا الكلاسيكية)، وخوارزميات الكمبيوتر حيث تكون القدرة على التنبؤ ضرورية.

Probabilistic Systems:

Definition: A probabilistic system incorporates randomness or uncertainty, where the outcome cannot be predicted with certainty, only with probabilities.

يتضمن النظام الاحتمالي العشوائية أو عدم اليقين، حيث لا يمكن التنبؤ بالنتيجة بشكل مؤكد، بل بالاحتمالات فقط.

Characteristics:

- Involves chance or random variables.
- Outcomes are described by probability distributions..وصف النتائج من خلال توزيعات الاحتمالات
- The same input may lead to different outcomes due to inherent uncertainty. عدم اليقين المتأصل
- Often modeled using statistical methods !or stochastic processes..لعمليات العشوائية

Examples:

- Rolling a die رمي النرد: The outcome is one of six numbers, each with a 1/6 probability.
- Weather forecasting: Predictions are based on probabilities due to complex variables.
- Quantum mechanics: Particle behavior is described by probability waves.

Use Cases: Used in statistics, machine learning (e.g., predictive models), finance (e.g., stock market analysis), and natural systems like genetics or weather.

تُستخدم في الإحصاء، والتعلم الآلي (النماذج التنبؤية)، والتمويل (تحليل سوق الأوراق المالية)، والأنظمة الطبيعية مثل علم الوراثة أو الطقس.

Key Differences:

Aspect	Deterministic Systems	Probabilistic Systems
Nature	Predictable, certain outcomes	Uncertain, probabilistic outcomes
Behavior	Fixed rules, no randomness	Incorporates randomness or chance

Key Differences:

Aspect	Deterministic Systems	Probabilistic Systems
Examples	Clock, calculator, planetary motion	Dice roll, weather, quantum systems

Key Differences:

Aspect	Deterministic Systems	Probabilistic Systems
Modeling	Exact equations or algorithms	Probability distributions, statistics
Applications	Engineering, classical physics	Statistics, AI, complex systems

4.Static vs. Dynamic Systems

In systems theory (and fields like engineering, physics, computer science, and mathematics), static and dynamic refer to how a system behaves over time, particularly in response to inputs.

In computer systems, the terms static and dynamic refer to when decisions, allocations, or behaviors are determined — at compile-time (before execution) vs. at runtime (during execution).

Aspect	Static System / Mechanism	Dynamic System / Mechanism
Definition	Determined before program runs (compile-time or link-time).	Determined during program execution (runtime).
Timing	Fixed at build/deploy. No changes without recompilation.	Adapts based on current conditions, inputs, or state.

Aspect	Static System / Mechanism	Dynamic System / Mechanism
Memory Allocation	Static allocation: Memory size known at compile-time.	Dynamic allocation: Size decided at runtime.
Binding	Static binding: Function/variable resolved at compile-time. Function overloading (C++), early binding.	Dynamic binding: Resolved at runtime. Virtual functions (C++/Java), late binding.

Aspect	Static System / Mechanism	Dynamic System / Mechanism
Typing	Static typing: Types checked at compile-time. C, C++, Java, Rust	Dynamic typing: Types checked at runtime. Python, JavaScript, Ruby
Loading / Linking	Static linking: Libraries bundled into executable at build. Larger binary, no external deps.	Dynamic linking: Libraries loaded at runtime (DLLs, .so). Smaller binary, shared libs.

Aspect	Static System / Mechanism	Dynamic System / Mechanism
Configuration	Static config: Hardcoded or config files read once at start.	Dynamic config: Reloadable, hot-swappable, environment-aware.
Performance	Faster execution (no runtime overhead). Predictable.	Slower (runtime checks, lookups). More flexible.

Aspect	Static System / Mechanism	Dynamic System / Mechanism
Flexibility	Low — changes require rebuild/redeploy.	High — adapts without restart.
Error Detection	Compile-time errors (safer, earlier feedback).	Runtime errors (may crash in production).

Real-World Computing Examples

Scenario	Static Approach	Dynamic Approach
Array size	char buffer[256];	char* buffer = malloc(size);
Function call	Direct call (known at compile)	Virtual method → vtable lookup at runtime
Web server config	Config baked into binary	Config file reloaded without restart

Scenario	Static Approach	Dynamic Approach
Language	C (mostly static)	Python (fully dynamic)
Language	C (mostly static)	Python (fully dynamic)

Key Trade-offs

Factor	Favors Static	Favors Dynamic
Speed	✅ Faster runtime	❌ Slower (overhead)
Safety	✅ Fail-fast at compile	❌ Errors surface in production
Flexibility	❌ Rigid	✅ Adaptable, extensible
Binary Size	❌ Larger (static linking)	✅ Smaller, shared libs

Key Trade-offs

Factor	Favors Static	Favors Dynamic
Speed	✅ Faster runtime	❌ Slower (overhead)
Safety	✅ Fail-fast at compile	❌ Errors surface in production
Flexibility	❌ Rigid	✅ Adaptable, extensible
Binary Size	❌ Larger (static linking)	✅ Smaller, shared libs

Factor	Favors Static	Favors Dynamic
Deployment	❌ Rebuild + redeploy	✅ Hot reloads, plugins

4.Simple vs. Complex Systems

In **Systems Analysis and Design (SAD)**, **systems are classified based on their complexity — that is, the number of components, relationships, and interactions they contain. They can generally be Simple or Complex systems**.

Simple Systems

Definition:

A **simple system is one that has a small number of components** and **few interactions between them**.

Its **structure and behavior are easy to understand, predict, and control**.

Characteristics:

- **Few and well-defined components.**
- **Linear or straightforward relationships.**
- **Easy to analyze, design, and modify.**
- **Minimal feedback or control loops.**
- **Behavior is predictable.**
- **Usually deterministic (outputs can be accurately predicted from inputs).**

Examples:

- A basic calculator system.
- Manual attendance record system.
- Simple payroll calculation (without automation).
- Thermostat controlling room temperature.

Complex Systems

Definition:

A **complex system is one that has many interrelated components interacting in multiple ways.**

Its behavior is dynamic, adaptive, and sometimes unpredictable.

Characteristics:

- **Many interdependent subsystems and components.**
- **Non-linear relationships between elements.**
- **Harder to understand and model**.
- **Contains feedback loops** and **control mechanisms**.
- **Often involves human and technological interaction**.

- **Behavior may change due to internal or external factors.**

Examples:

- **Airline reservation system.**
- **Hospital management system.**
- **Banking or e-commerce systems.**
- **Traffic management system.**
- **Weather forecasting systems.**

Simple and Complex Systems

Feature	Simple System	Complex System
Number of Components	Few	Many
Interactions	Simple and direct	Multiple and interdependent
Behavior	Predictable	Unpredictable

Feature	Simple System	Complex System
Behavior	Predictable	Unpredictable
Analysis	Easy	Difficult
Control Mechanism	Minimal	Multiple with feedback
Design & Maintenance	Straightforward	Complicated

Feature	Simple System	Complex System
Example	Calculator	Banking System

Summary

- Simple systems are easy to design and understand because they involve fewer parts and predictable outcomes.
- Complex systems involve many interconnected parts, making analysis and control more challenging.

- In SAD, understanding whether a system is simple or complex helps determine the tools, methods, and models needed for analysis and design.

- In SAD, understanding whether a system is simple or complex helps determine the tools, methods, and models needed for analysis and design.

End