

SYSTEMS ANALYSIS

Systems Analysis & Design

Author: Eng. Carlos Andrés Sierra, M.Sc.
cavirguezs@udistrital.edu.co

Full-time Adjunct Professor
Computer Engineering Program
School of Engineering
Universidad Distrital Francisco José de Caldas

2025-III



Outline

1 Basic Concepts

2 Chaos and Dynamic Systems

3 Abstraction and Modularity



Outline

1 Basic Concepts

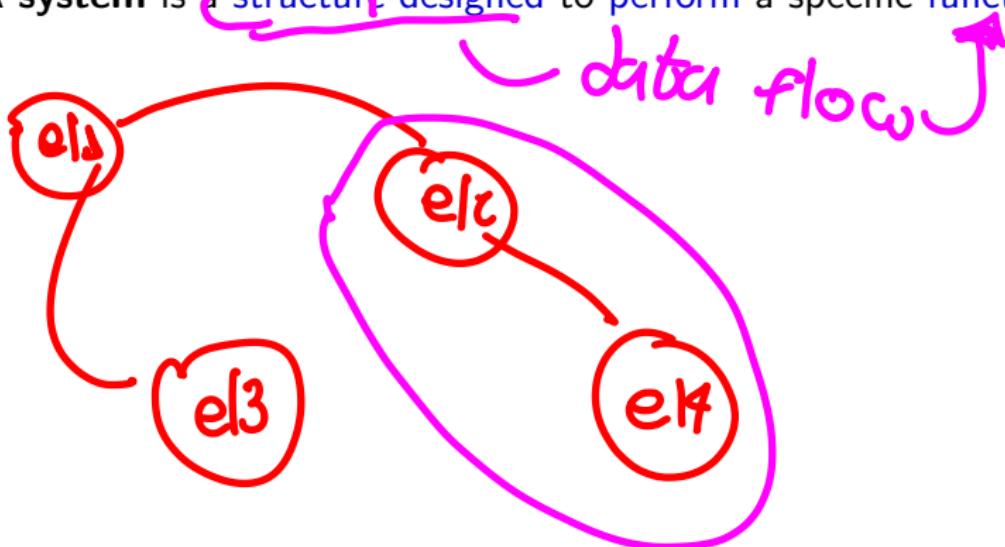
2 Chaos and Dynamic Systems

3 Abstraction and Modularity



What is a System?

- A **system** is a set of interacting components that work together to achieve a common goal.
- A **system** is a collection of elements organized in a specific way.
- A **system** is a structure designed to perform a specific function.



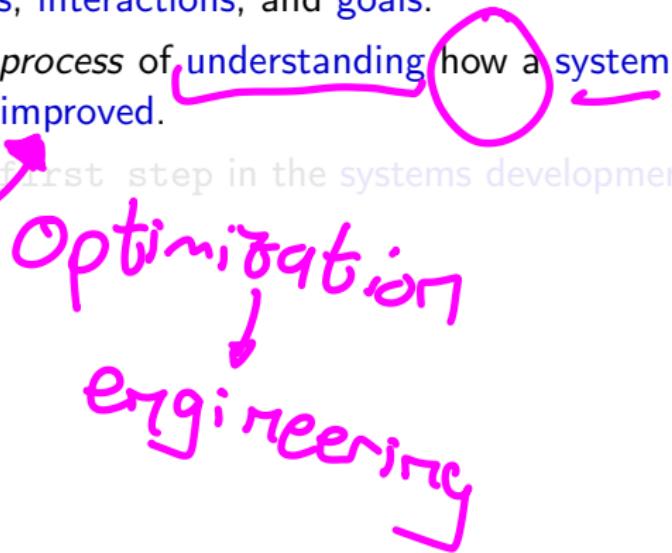
Systems Analysis Process

- **Systems analysis** is the process of studying a system in order to identify its components, interactions, and goals.
- **Systems analysis** is the process of understanding how a system works and how it can be improved.
- **Systems analysis** is the first step in the systems development lifecycle.



Systems Analysis Process

- **Systems analysis** is the *process of studying a system* in order to identify its components, interactions, and goals.
- **Systems analysis** is the *process of understanding how a system works and how it can be improved.*
- **Systems analysis** is the first step in the systems development lifecycle.



Systems Analysis Process

- **Systems analysis** is the *process* of **studying** a **system** in order to identify its **components**, **interactions**, and **goals**.
- **Systems analysis** is the *process* of **understanding** how a **system** works and how it can be **improved**.
- **Systems analysis** is the first step in the **systems development lifecycle**.

SDLC

requirements



Systems Development Lifecycle

- The **systems development lifecycle** is a process that *guides* the development of a system.
 - It includes **planning, analysis, design, implementation, and maintenance** phases.
 - It is a **structured approach** to developing a system that ensures it meets the needs of its users.
- step by step*



Systems Development Lifecycle

- The **systems development lifecycle** is a *process* that *guides* the development of a **system**.
- It includes **planning**, **analysis**, **design**, **implementation**, and **maintenance** phases.
- It is a **structured approach** to *developing* a **system** that ensures it meets the **needs** of its **users**.



Systems Analysis Techniques

- Systems analysis uses a variety of techniques to study a system.
- These include interviews, surveys, observations, and document analysis.
- It also includes data modeling, process modeling, and requirements analysis.

Final user

(Fix)

client + user

information gathering

Block Breaker

quickly

questions

web (1997)

cost
3 years

warehouse



Systems Analysis Techniques

- Systems analysis uses a variety of techniques to study a system.
- These include interviews, surveys, observations, and document analysis.
- It also includes data modeling, process modeling, and requirements analysis.

sources? format? periodicity?

↓
impact
↓
implementation
trade-offs

step by step
↓
optimization



Systems Analysis Tools

- Systems analysis uses a variety of tools to study a system.
- These include **diagrams**, charts, flowcharts, and **data models**.
- It also *includes* software tools such as spreadsheets, databases, and simulation software.

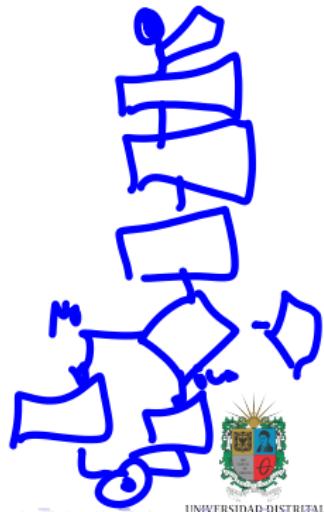
DB
Pipeline
ETL



Descriptive Analysis

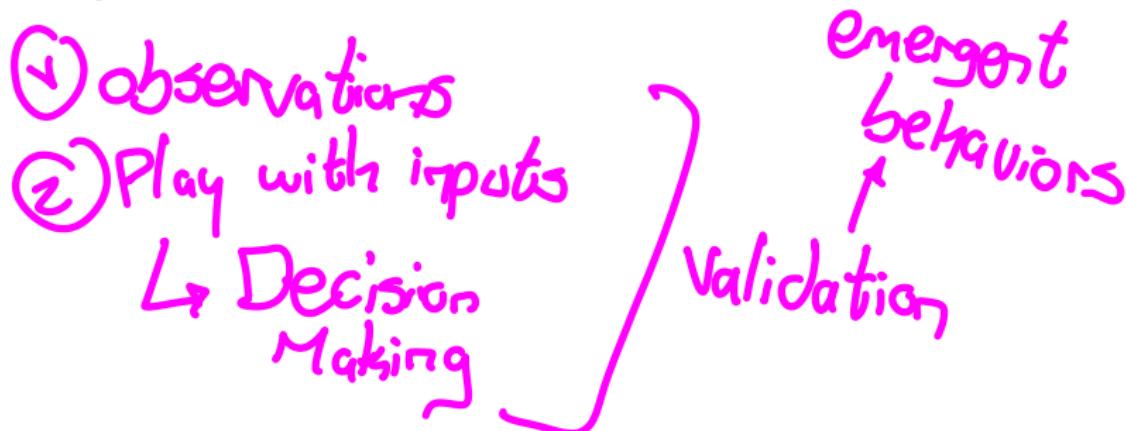


Process



Systems Analysis Tools

- Systems analysis uses a variety of tools to study a system.
- These include diagrams, charts, flowcharts, and data models.
- It also includes software tools such as spreadsheets, databases, and simulation software.



Lateral Thinking

- **Lateral thinking** is a creative problem-solving technique that involves thinking outside the box.
- It is a non-linear approach to problem-solving that encourages innovation and creativity.
- It is a useful technique for generating new ideas and solving complex problems.
- Examples:
 - How can you improve the design of a product?
 - What are the benefits of failure?
 - Why is ignorance important?
 - When is failure better than success?

tech \Rightarrow startups

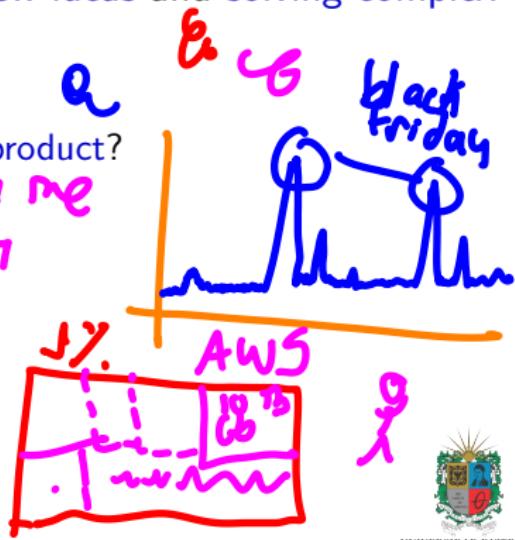


Lateral Thinking

- Lateral thinking is a creative problem-solving technique that involves thinking outside the box.
- It is a non-linear approach to problem-solving that encourages innovation and creativity.
- It is a useful technique for generating new ideas and solving complex problems.
- Examples:

- How can you improve the design of a product?
- What are the benefits of failure? ↗ me
- Why is ignorance important? ↗ learn
- When is failure better than success?

↳ business opportunity



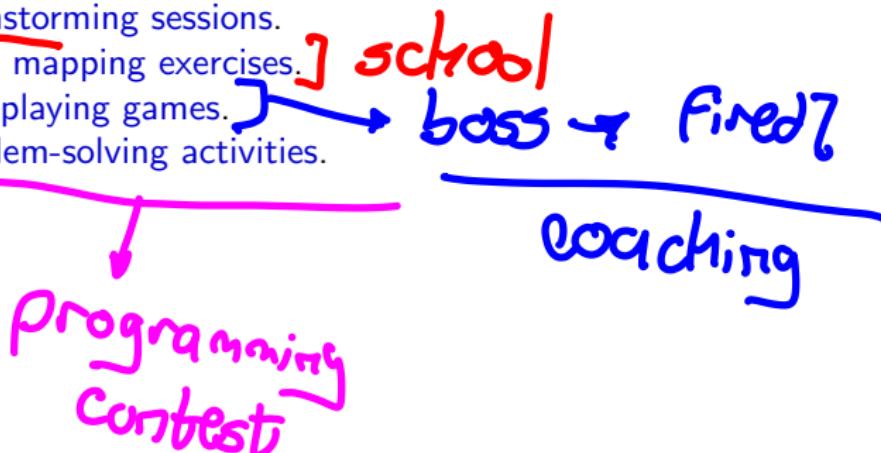
Lateral Thinking Training

- **Lateral thinking** is a *skill* that can be learned and developed through training and practice.
- It involves **exercises**, **games**, and **activities** that encourage creative thinking.
- Examples of lateral thinking exercises:
 - Brainstorming sessions
 - Mind mapping exercises.
 - Role-playing games.
 - Problem-solving activities.



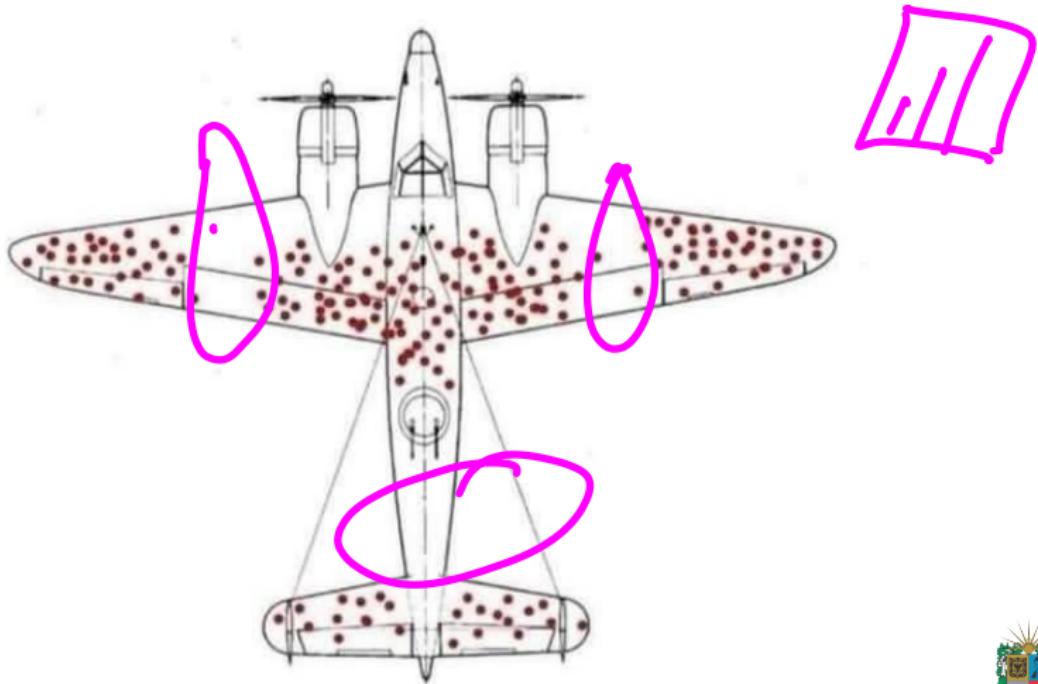
Lateral Thinking Training

- Lateral thinking is a *skill* that can be learned and developed through training and practice.
- It involves exercises, games, and activities that encourage creative thinking.
- Examples of lateral thinking exercises:
 - Brainstorming sessions.
 - Mind mapping exercises.
 - Role-playing games.
 - Problem-solving activities.



Lateral Thinking Exercise

World War II, the perspective of Abraham Wald:



Uncertainty and Risk

- **Uncertainty** is the lack of knowledge about the future outcome of a decision or event.

- Risk is the probability of a negative outcome or loss associated with a decision or event.

• Uncertainty and risk are *inherent* in complex systems and decisions.

- They can be managed through planning, analysis, and mitigation strategies.

→ **XSimulation** → **Probability**

+ information →

+ knowledge



Uncertainty and Risk

- **Uncertainty** is the lack of knowledge about the future outcome of a decision or event.
 - **Risk** is the probability of a negative outcome or loss associated with a decision or event.
- Uncertainty and risk are inherent in complex systems and decisions.
- They can be managed through planning, analysis, and mitigation strategies.
- casos!



Uncertainty and Risk

- **Uncertainty** is the lack of knowledge about the future outcome of a decision or event.
- **Risk** is the probability of a negative outcome or loss associated with a decision or event.
- **Uncertainty and risk** are *inherent* in complex systems and decisions.
- They can be managed through planning, analysis, and mitigation strategies.



Uncertainty and Risk

- **Uncertainty** is the lack of knowledge about the future outcome of a decision or event.
- **Risk** is the probability of a negative outcome or loss associated with a decision or event.
- **Uncertainty** and **risk** are *inherent* in complex systems and decisions.
- They can be managed through planning, analysis, and mitigation strategies.

③

software
proyectos



Complexity and Emergence

- **Complexity** is the degree to which a system is **difficult** to understand.
 - Emergence is the appearance of unexpected properties in a system that arise from the interactions of its components.
 - Complexity and emergence are common in dynamic systems that are non-linear and chaotic.
 - They can be studied and understood through systems analysis and modeling.
- (+ elements
+ relations)

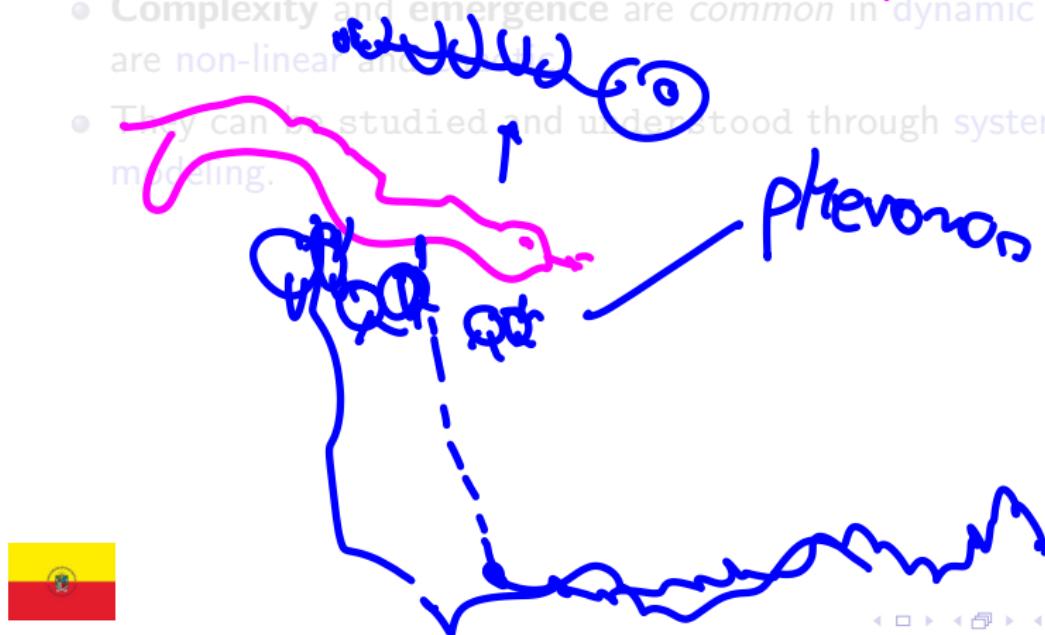


Planet Nine



Complexity and Emergence

- **Complexity** is the degree to which a system is **difficult** to understand.
- **Emergence** is the appearance of **unexpected properties** in a **system** that **arise** from the interactions of its **components**.
- Complexity and emergence are *common* in dynamic systems that are *non-linear* and *chaotic*.
- They can be studied and understood through systems analysis and modeling.



Complexity and Emergence

- **Complexity** is the degree to which a system is **difficult** to understand.
- **Emergence** is the appearance of **unexpected properties** in a **system** that **arise** from the interactions of its **components**.
- **Complexity and emergence** are *common* in dynamic **systems** that are **non-linear** and **chaotic**.
- They can be studied and understood through **systems analysis and modeling**.



Complexity and Emergence

- **Complexity** is the degree to which a system is **difficult** to understand.
- **Emergence** is the appearance of **unexpected properties** in a **system** that **arise** from the interactions of its **components**.
- **Complexity** and **emergence** are *common* in dynamic systems that are **non-linear** and **chaotic**.
- They can be **studied** and understood through **systems analysis** and **modeling**.

→ Exploratory Data Analysis

↳ data-profiling

→ correlation matrix
↳ kaggle



Outline

1 Basic Concepts

2 Chaos and Dynamic Systems

3 Abstraction and Modularity



What is Chaos?

- **Chaos** is a *branch* of **mathematics** that *studies* the **sensitivity** of dynamical systems to **initial conditions**.
- **Chaos** is a **non-linear** behavior that is **highly sensitive** to **initial conditions**.
- **Chaos** is a deterministic behavior that is **not predictable** in the **long term**.
- **Chaos** is a complex behavior that is **hard to understand**.

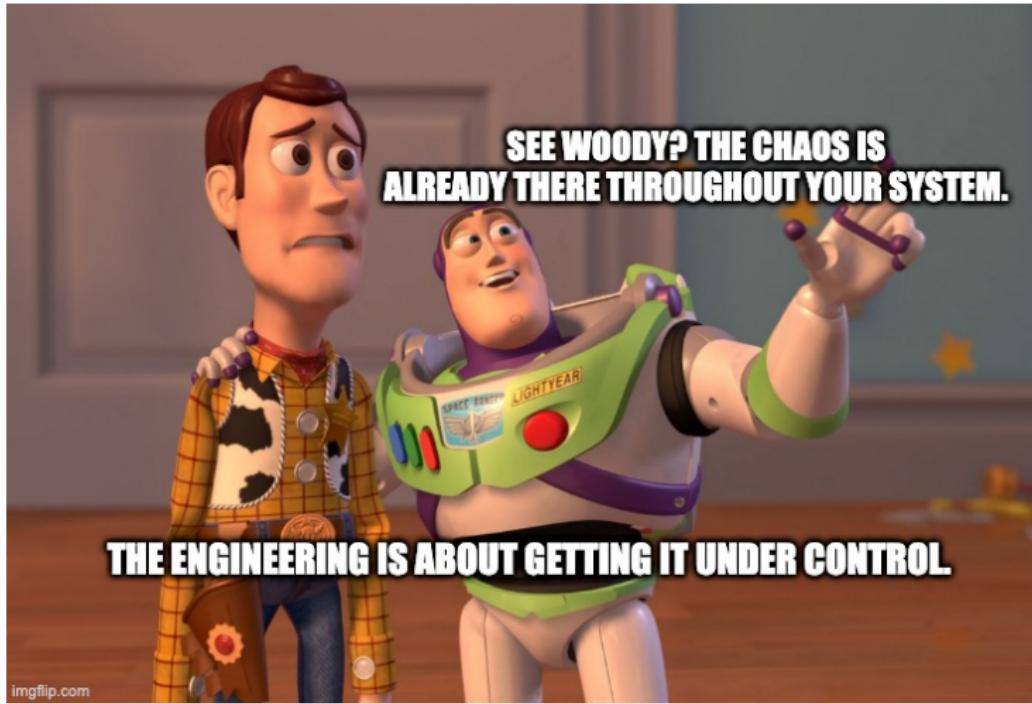


What is Chaos?

- **Chaos** is a *branch* of **mathematics** that *studies* the **sensitivity** of dynamical systems to **initial conditions**.
- **Chaos** is a **non-linear** behavior that is **highly sensitive** to **initial conditions**.
- **Chaos** is a **deterministic** behavior that is **not predictable** in the **long term**.
- **Chaos** is a **complex** behavior that is **hard** to understand.



Chaos is Everywhere!



What is a Dynamic System?

- A **dynamic system** is a system that **changes** over time.
- A **dynamic system** is **sensitive** to **initial conditions**.
- A **dynamic system** is **non-linear**.
- A **dynamic system** can be **chaotic**.



What is a Dynamic System?

- A **dynamic system** is a system that **changes** over time.
- A **dynamic system** is **sensitive** to **initial conditions**.
- A **dynamic system** is **non-linear**.
- A **dynamic system** can be **chaotic**.



Chaotic Attractors

A **chaotic attractor** is a **set** of **points** in a phase space that **attracts** the trajectories of a **dynamical system**.



Fractals in Nature



Watch this video: <https://www.youtube.com/watch?v=kkGeOWYOFoA>



Swarm Intelligence I

- Swarm intelligence is the collective behavior of *decentralized, self-organized systems*, natural or artificial.
- The concept is employed in work on *artificial intelligence*.
- The expression was introduced by *Gerardo Beni* and *Jing Wang* in 1989, in the context of *cellular robotic systems*. For example, watch this [video](#).



Swarm Intelligence I

- Swarm intelligence is the collective behavior of *decentralized, self-organized systems*, natural or artificial.
- The concept is employed in work on *artificial intelligence*.
- The expression was introduced by *Gerardo Beni* and *Jing Wang* in *1989*, in the context of *cellular robotic systems*. For example, watch this [video](#).



Swarm Intelligence II

- The **idea** is: if you see an **individual**, it may seem random; **however**, several **individuals interacting** with each other and the environment show **smart behaviors**.
- **Yu Takeuchi** said: one colombian is more intelligent than one japanese, but two japanese are smarter than two colombians.
- There are interesting **population behaviors** in nature, specially in insects: bees, ants, termites, among others.
- There are also many examples in nature: schools of fish, birds, wolves.



Swarm Intelligence II

- The **idea** is: if you see an **individual**, it may seem random; **however**, several **individuals interacting** with each other and the environment show **smart behaviors**.
- **Yu Takeuchi** said: **one colombian is more intelligent than one japanese**, but **two japanese are smarter than two colombians**.
- There are interesting **population behaviors** in nature, specially in insects: bees, ants, termites, among others.
- There are also many examples in nature: schools of fish, birds, wolves.



Swarm Intelligence II

- The **idea** is: if you see an **individual**, it may seem random; **however**, several **individuals interacting** with each other and the environment show **smart behaviors**.
- **Yu Takeuchi** said: **one colombian** is **more intelligent than one japanese**, but **two japanese** are **smarter than two colombians**.
- There are interesting **population behaviors** in nature, specially in **insects**: bees, ants, termites, among others.
- There are also many **examples in nature**: schools of fish, birds, wolves.

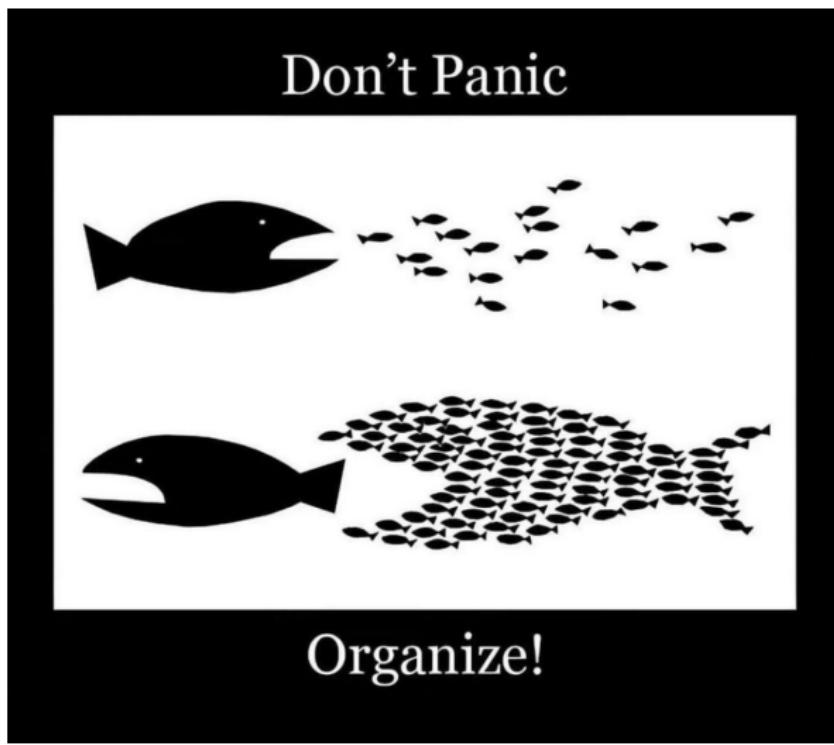


Emergent Behaviors

- Emergent behavior is the **appearance** of **complex patterns** and behaviors from **many** simple interactions.
- Emergent behavior **results** from the **collective** behavior of the **individuals** in the system.
- Emergent behavior is **not planned** or **designed** by any individual, but **arises** from their **interactions**.
- Emergent behavior is **not** the **sum** of the **individual** behaviors, but **something more**: **synergy**.
- **Swarm intelligence** refers to interesting **emergent behaviors**.



School Fish Algorithm



School Fish Algorithm

- **School fish** are interesting. When a predator attacks, they become confused by the large number of individuals and their **diverse movements**.
- The idea is simple: “*Don't touch me, don't come too close, but stay somewhat close.*”
- This behavior is a **chain of action and reaction**. It confuses predators and helps the school move uniformly.
- Do you remember **Nemo**? The fish with a sword snout, the pirates, or Marlin's imitation of talking-all are somewhat similar. Watch [here](#).
- The **school fish algorithm** is a **multi-agent system** that simulates the behavior of a **school of fish**.



School Fish Algorithm

- **School fish** are interesting. When a predator attacks, they become confused by the large number of individuals and their **diverse movements**.
- The **idea** is simple: “*Don't touch me, don't come too close, but stay somewhat close.*”
- This behavior is a **chain of action and reaction**. It **confuses predators** and helps the school move uniformly.
- Do you remember **Nemo**? The fish with a sword snout, the pirates, or Marlin's imitation of talking-all are somewhat similar. Watch **here**.
- The **school fish algorithm** is a **multi-agent system** that **simulates** the behavior of a **school of fish**.



School Fish Algorithm

- **School fish** are interesting. When a predator attacks, they become confused by the large number of individuals and their **diverse movements**.
- The **idea** is simple: “*Don't touch me, don't come too close, but stay somewhat close.*”
- This behavior is a **chain of action and reaction**. It **confuses predators** and helps the school move uniformly.
- Do you remember **Nemo**? The fish with a sword snout, the pirates, or Marlin's imitation of talking-all are somewhat similar. Watch **here**.
- The **school fish algorithm** is a **multi-agent system** that **simulates** the behavior of a **school of fish**.



Ant Colony Algorithm

- Ant colony algorithm is a **multi-agent system** that **simulates** the behavior of an **ant colony**.
- Ant colony algorithm is based on the **social behavior** of **ants** and the use of **pheromones**. Watch **here**.
- Ant colony algorithm is used to **solve optimization problems**. Watch **here**.



Ant Colony Algorithm

- Ant colony algorithm is a **multi-agent system** that **simulates** the behavior of an **ant colony**.
- Ant colony algorithm is based on the **social behavior** of **ants** and the use of **pheromones**. Watch [here](#).
- Ant colony algorithm is used to **solve optimization problems**. Watch [here](#).



Outline

1 Basic Concepts

2 Chaos and Dynamic Systems

3 Abstraction and Modularity



What is Abstraction?

- **Abstraction** is the *process* of ignoring minor details in order to focus on the important aspects of a system.
- **Abstraction** is the *process* of simplifying a complex system to understand it.
- **Abstraction** is the *process* of generalizing a specific system to apply it to other systems.



What is Abstraction?

- **Abstraction** is the *process* of ignoring minor details in order to focus on the important aspects of a system.
- **Abstraction** is the *process* of simplifying a complex system to understand it.
- **Abstraction** is the *process* of generalizing a specific system to apply it to other systems.



What is Abstraction?

- **Abstraction** is the *process of ignoring minor details* in order to focus on the **important aspects** of a **system**.
- **Abstraction** is the *process of simplifying a complex system* to understand it.
- **Abstraction** is the *process of generalizing a specific system* to apply it to **other systems**.



What is Modularity?

- **Modularity** is the *process* of dividing a **system** into **smaller parts** called **modules**.
- **Modularity** is the *process* of organizing a system into **independent units** that can be developed and maintained separately.
- **Modularity** is the *process* of reducing the complexity of a system by breaking it into **smaller parts**.



What is Modularity?

- **Modularity** is the *process* of dividing a **system** into smaller parts called **modules**.
- **Modularity** is the *process* of organizing a **system** into **independent units** that can be **developed** and **maintained separately**.
- **Modularity** is the *process* of reducing the **complexity** of a system by breaking it into smaller parts.



What is Modularity?

- **Modularity** is the *process* of dividing a **system** into smaller parts called **modules**.
- **Modularity** is the *process* of organizing a **system** into independent units that can be **developed** and **maintained** separately.
- **Modularity** is the *process* of reducing the **complexity** of a **system** by breaking it into **smaller parts**.



Abstraction and Modularity

- **Abstraction** and **modularity** are *two important concepts* in **systems analysis**.
- They help **reduce** the **complexity** of a system by **ignoring details** and **dividing** it into **smaller parts**.
- They help **improve** the **understanding**, **development**, and **maintenance** of a system.



Abstraction and Modularity

- **Abstraction** and **modularity** are *two important concepts* in **systems analysis**.
- They help **reduce** the **complexity** of a system by **ignoring details** and **dividing** it into **smaller parts**.
- They help **improve** the **understanding**, **development**, and **maintenance** of a **system**.



Outline

1 Basic Concepts

2 Chaos and Dynamic Systems

3 Abstraction and Modularity



Thanks!

Questions?



Repo: <https://github.com/EngAndres/ud-public/tree/main/courses/systems-analysis>

