

## Company

Company is an **association of people based on legal acts for some common object(s), which could be economic as well as non-economic**. Its internal structure is based on **division of labor** and **gives a framework for business activities**.

## Company goals

You should define goals, so you'll at least try and have a mindset for it.

Company goals should be **SMART**:

- **Specific** – if it's not specific enough, people won't feel failure, if the goal wasn't reached. Thus, they won't try hard enough to reach these goals.
- **Measurable** – there needs to be a specific number which needs to be reached. "We want to increase our profit" → "We want to increase our profit by 10%"
- **Assignable/Achievable** – who will lead this initiative? Will this person have enough resources, including manpower, to make this project a success?
- **Reliable/Realistic** – Is the goal realistic, with the time and resources available?
- **Time-related** – When do we expect the intended result by?

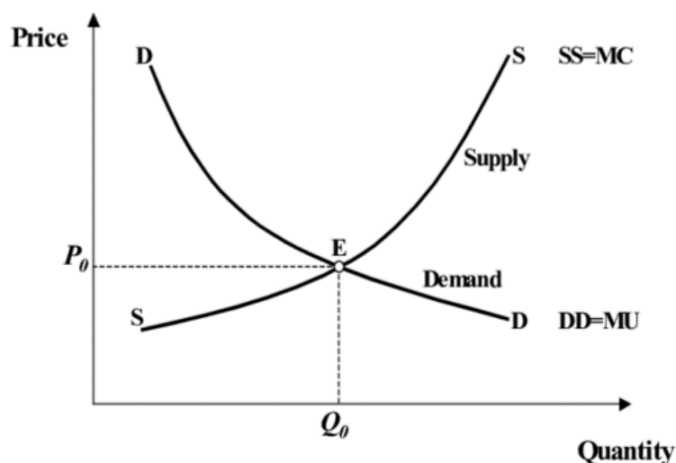
*Example: Company XYZ will save \$100.000 in the year 2025.*

## Company goal categories

- **Profit** - The primary goal is to generate financial gain or profit.
- **Non-profit** - The aim here is not to generate profit but to fulfill a mission or cause. The focus is on community, charity, or social goals rather than financial gain.
- **Growth** - This refers to increasing various aspects of the company, such as income, the number of employees, or production capacity.
- **Market Share** - The goal is to increase the company's percentage of sales within a particular market or industry compared to its competitors.
- **Expansion** - The focus is on expanding into new markets, opening new locations, or diversifying the company's product or service offerings to reach a broader audience.

## Marshall-cross

It illustrates the equilibrium between supply and demand. The point where these the supply and demand curves meet is the **market equilibrium**, where the quantity supplied equals the quantity demanded, and the market clears at a specific price. The cross visually explains price adjustments and how markets reach balance through this intersection.

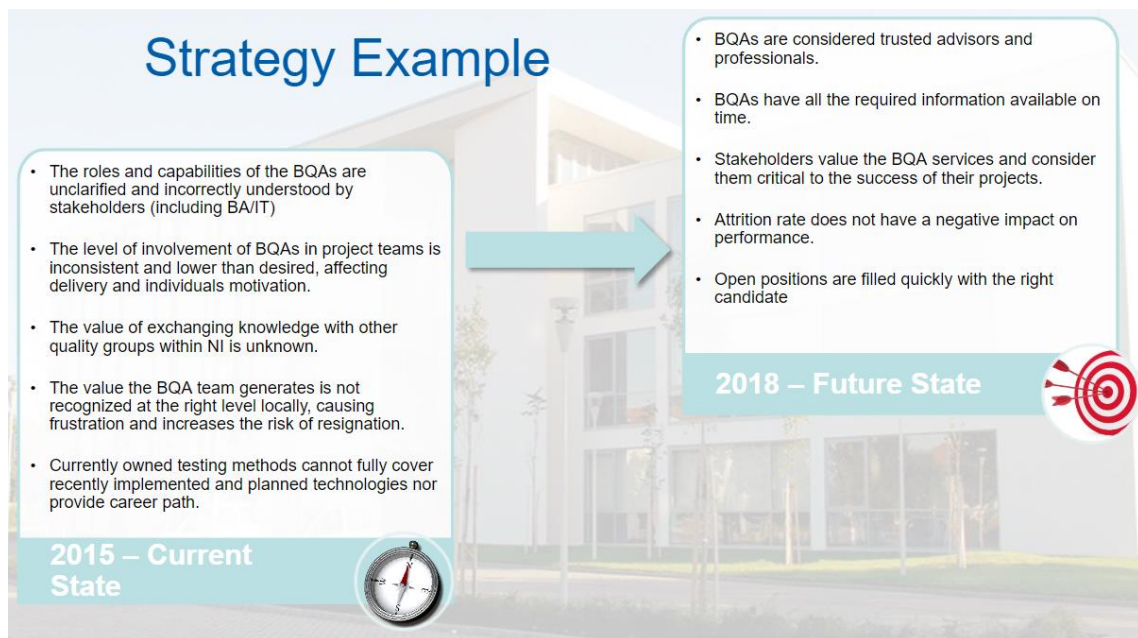
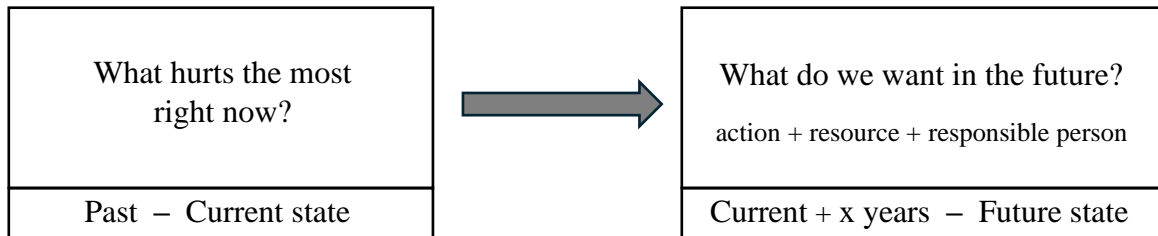


## Strategy

An action plan designed to reach the long-term goals with the given resources.

strategy  $\xrightarrow{\text{leader}}$  knowing exactly what to do

### Creating a strategy – 2 window strategy:



### Micro and macro environment of a company/business





## Why do we need Enterprise Architecture?

- Complex Company Goals
- Complex IT
- IT expenses are increasing
- Return-on-investment of IT is decreasing

## Planning in the company

The 100-Year Plan		
Qualitative  Quantitative	100-Year Plan	Core Values Culture The NI Way
	10-Year Plan	Core Strategic Vision People Learning/Development Platforms
	5-Year Plan	Products/Markets Company Goals Growth Goals Quality Goals
	1-Year Plan	Sales Forecasts Goal Deployment
	Quarterly Plan	Financial Goals Sales and Profits Update Goals

## Architecture:

- **Structure of building items:** the way individual components (building items) of a system or architecture are organized and arranged. It defines how these elements fit together to form the whole, such as the layout of physical or software components in a building or system.
- **Relations between building items:** how the different components or building items interact with each other. It's important to understand these relationships to ensure the components work together efficiently.
- **Principles of evolution:** the guidelines or principles that govern how a system or structure changes over time. It includes factors like scalability, adaptability, and flexibility, which allow the architecture to evolve and grow to meet future needs or changing requirements without requiring a complete redesign.

## MODELS:

### I. Zachman framework

	Why	How	What	Who	Where	When
Contextual	Goal List	Process List	Material List	Organisational Unit & Role List	Geographical Locations List	Event List
Conceptual	Goal Relationship	Process Model	Entity Relationship Model	Organisational Unit & Role Relationship Model	Locations Model	Event Model
Logical	Rules Diagram	Process Diagram	Data Model Diagram	Role Relationship Diagram	Locations Diagram	Event Diagram
Physical	Rules Specification	Process Function Specification	Data Entity Specification	Role Specification	Location Specification	Event Specification
Detailed	Rules Details	Process Details	Data Details	Role Details	Location Details	Event Details

- the rows represent different elements (people) in the plan
- the columns are different questions about the plan that needs to be answered to each person, changing the amount of details and your language according to the person you're talking to (e.g. „Why do we make this application?“ – Answer to the founder/owner of the app: „To make great profit“. Answer to the programmer: „To challenge yourself“)
- if we can fill out each and every cell, it is a good plan
- we can't reuse the content of a cell in another one

#### Zachman layers:







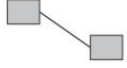


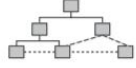


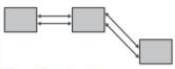

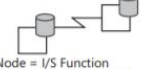
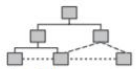


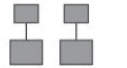


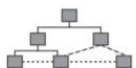








Contextual – Executive perspective  
Conceptual – Business management perspective  
Logical – Architect perspective  
Physical – Engineer perspective  
Detailed – Technician perspective

#### Columns:

Why – Motivation Intentions  
How – Process Flows  
What - Inventory Set  
Who – Responsibility Assignment  
Where – Distribution Networks  
When – Timing Cycles



## Example:

	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)	List of Things Important to the Business 	List of Processes the Business Performs 	List of Locations in which the Business Operates 	List of Organizations Important to the Business 	List of Events/Cycles Significant to the Business 	List of Business Goals/Strategies 	SCOPE (CONTEXTUAL)
<i>Planner</i>	ENTITY = Class of Business Thing	Process = Class of Business Process	Node = Major Business Location	People = Major Organization Unit	Time = Major Business Event/Cycle	Ends/Mean = Major Business Goal/Strategy	<i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)	e.g. Semantic Model 	e.g. Business Process Model 	e.g. Business Logistics System 	e.g. Work Flow Model 	e.g. Master Schedule 	e.g. Business Plan 	BUSINESS MODEL (CONCEPTUAL)
<i>Owner</i>	Ent = Business Entity Rein = Business Relationship	Proc. = Business Process I/O = Business Resources	Node = Business Location Link = Business Linkage	People = Organization Unit Work = Work Product	Time = Business Event Cycle = Business Cycle	End = Business Objective Means = Business Strategy	<i>Owner</i>
SYSTEM MODEL (LOGICAL)	e.g. Logical Data Model 	e.g. Application Architecture 	e.g. Distributed System Architecture 	e.g. Human Interface Architecture 	e.g. Processing Structure 	e.g. Business Rule Model 	SYSTEM MODEL (LOGICAL)
<i>Designer</i>	Ent = Data Entity Rein = Data Relationship	Proc. = Application Function I/O = User Views	Node = I/S Function (Processor, Storage, etc) Link = Line Characteristics	People = Role Work = Deliverable	Time = System Event Cycle = Processing Cycle	End = Structural Assertion Means = Action Assertion	<i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)	e.g. Physical Data Model 	e.g. System Design 	e.g. Technology Architecture 	e.g. Presentation Architecture 	e.g. Control Structure 	e.g. Rule Design 	TECHNOLOGY MODEL (PHYSICAL)
<i>Builder</i>	Ent = Segment/Table/etc. Rein = Pointer/Key/etc.	Proc. = Computer Function I/O = Data Elements/Sets	Node = Hardware/Systems Link = Line Specifications	People = User Work = Screen Format	Time = Execute Cycle = Component Cycle	End = Condition Means = Action	<i>Builder</i>
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)	e.g. Data Definition 	e.g. Program 	e.g. Network Architecture 	e.g. Security Architecture 	e.g. Timing Definition 	e.g. Rule Specification 	DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)
<i>Sub-Contractor</i>	Ent = Field Rein = Address	Proc. = Language Statement I/O = Control Block	Node = Address Link = Protocol	People = Identity Work = Job	Time = Interrupt Cycle = Machine Cycle	End = Sub-condition Means = Step	<i>Sub-Contractor</i>
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE

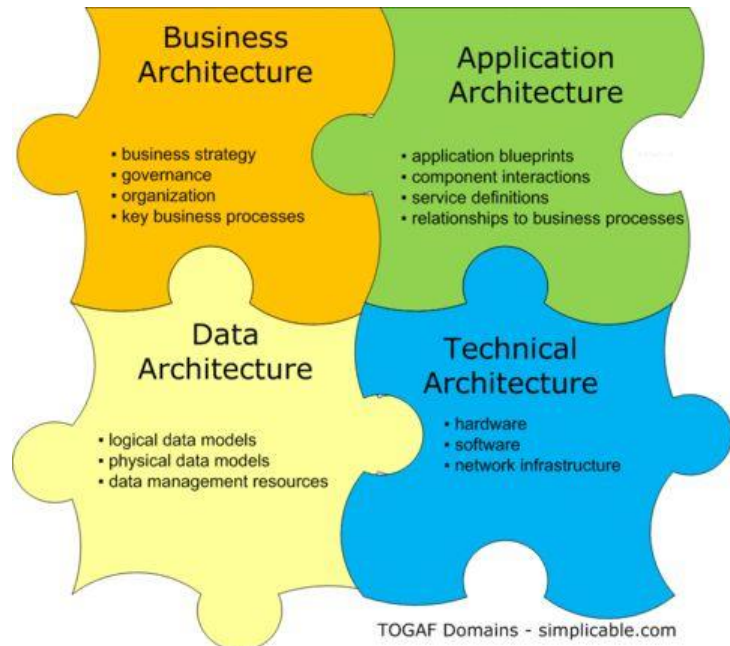
Advantages	Disadvantages
<ul style="list-style-type: none"> <li>– <b>All stakeholders are considered</b> (All relevant parties, or stakeholders, involved in a system's development are accounted for. This comprehensive approach reduces miscommunication and ensures that everyone's requirements are reflected in the final system.)</li> <li>– <b>Concentrates on documents (artifact)</b> (This focus on documentation ensures clarity and traceability at every stage of the system's lifecycle. Well-structured artifacts make it easier for teams to understand the system, maintain it, and implement changes when needed.)</li> <li>– <b>Unnecessary functions are avoided</b> (It promotes efficiency by ensuring that every component of the architecture serves a defined purpose. This minimizes wasted effort on features or processes that do not contribute directly to the organization's goals)</li> <li>– <b>IOT is involved in the planning</b> (It supports futureproofing and ensures that the architecture can adapt to emerging tech trends)</li> </ul>	<ul style="list-style-type: none"> <li>– <b>No proper process for architecture development</b> (The Zachman Framework is primarily a taxonomy or classification scheme rather than a methodology. It doesn't provide a step-by-step process or methodology for developing an architecture. While it organizes the architectural components across different perspectives, it doesn't guide how to implement or develop these components in practice.)</li> <li>– <b>No measurement of effectiveness of the architecture</b> (The framework doesn't include metrics or tools to evaluate the success or effectiveness of the developed architecture. Without built-in mechanisms to measure how well the architecture meets business goals or how efficiently it operates.)</li> </ul>

## II. TOGAF – The Open Group Architecture Framework

While Zachman categorizes the artifacts in a framework, TOGAF gives a **process** to create the artifacts.

It divides into 4 domains/architectures:

1. **Business architecture** - Describes the processes the business uses to meet its goals
2. **Application architecture** - Describes how specific applications are designed and how they interact with each other
3. **Data architecture** - Describes how the enterprise datastores are organized and accessed
4. **Technical architecture** - Describes the hardware and software infrastructure that supports applications and their interactions



### The ADM (Architecture Development Method) approach

It goes from *every possible architecture* to your *exact architecture solution* for your company.

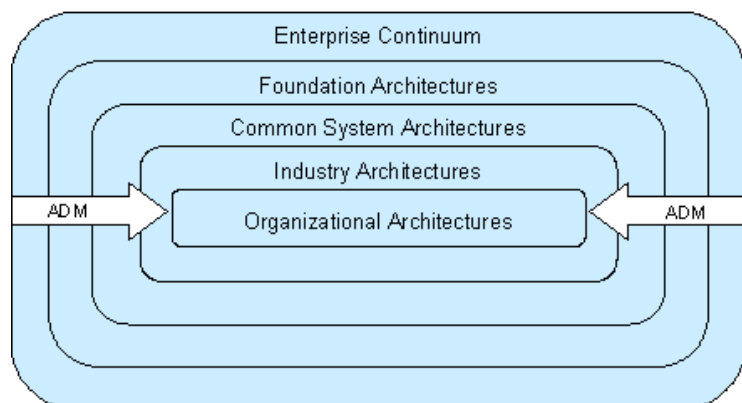
**Enterprise Continuum:** every possible architecture in the world

**Foundation Architectures:** These are architectural principles that can, **theoretically**, be used by any **IT organization**

**Common Systems Architectures:** These are principles that one would expect to see in many but, perhaps, **not all types of enterprises**

**Industry Architectures:** These are principles that are specific across many enterprises that are part of **the same domain**

**Organizational Architectures:** These are the architectures that are **specific to a given enterprise**



### Critics of TOGAF:

- Lack of specificity in documentation
- Provides only high-level guidelines rather, which lead to inconsistent interpretations
- No guarantee for successful enterprise architecture

**Preliminary phase:** establishes the parameters for a successful iteration of an ADM. Main objectives of the phase is to determine and establish the *Architecture Capability* desired by the organization. What needs to be done, and how will it be carried out? We need to get a *‘buy-in’* (everyone agrees with the project) from the highest (possible) level of the organisation, so we can have as many resources as possible.

**A. Architecture Vision:** establishes a high-level vision for enterprise architecture, *defining scopes and constraints* (like budget or personnel), engaging stakeholders, validating the business context, and obtaining approvals for the *Statement of Architecture Work*.

**B. Business Architecture:** develops a *Target Business Architecture* that describes the processes the business uses to meet its goals and to align with *phase A*. We need to make a detailed *as-is* (overview of the *current state* of the organization's process and capabilities) and *to-be* (overview of the *future state* of the organization).

By creating these diagrams, organizations can analyze and identify gaps (*gap analysis*) between the Baseline and the Target state.

**C. Information Systems Architectures:** develops the *Target Data Architecture* (describes how the enterprise datastores are organized and accessed) and the *Target Application Architecture* (describes how specific applications are designed and how they interact with each other) to support *phase A and B*, addressing data management and identifying roadmap components based on gaps between the *Baseline* and the *Target Architecture*.

**D. Technology Architecture:** develops the *Target Technology/Technical Architecture* to support *phase C*, and to deliver business, data, and application components through *technology*. We need to define what *hardware and software* we'll use in the architecture to support the applications, as well as their criteria and measurements like performance, maintainability, physical location, latency and availability.

**E. Opportunities and Solutions:** the phase is about finding *opportunities* for delivering the Target architecture by implementing specific *solutions*. It concentrates on *how* to deliver the architecture. You can use the *“low hanging fruit”* to achieve your solutions. It is fast and cheap, but not as tailored to you own specific needs, as a solution would, which you made for yourself from zero.

**F. Migration Planning:** a detailed plan to move from the *Baseline* to the *Target Architecture* (made in *phases B to D*). Efforts are needed to be *prioritized* using different criteria like return-on-investment, business value and measure of effectiveness.

**G. Implementation Governance:** provides an architectural overview of implementation. The *Target Architecture* is developed as a series of *Transition Architectures* from the *Baseline*, to realize business value and benefits *as soon as possible*, and to *reduce risk* during the transformation.

**H. Architecture Change Management:** here we make sure that the architecture achieves its intended or target business value. It requires *continual monitoring* of the governance requests, new technologies and the changes in the business environment. It supports the implemented enterprise architecture as a dynamic environment that has the *flexibility to evolve rapidly in response to the required changes*. A change request could start either a *simple architectural update* in *phase H*, or *start a whole new cycle* in the ADM. Monitoring business growth and decline is critical in this phase.

