
INTRODUCCIÓN A ARQUITECTURA

Autor: Luciano Straccia

Versión 2023.04



Temario

- Concepto de arquitectura
- Niveles de arquitectura
- Concepto de arquitectura de software
- Vistas y modelos
- Infraestructura

Arquitectura: conceptos introductorios



Definición de arquitectura

- El conjunto de estructuras necesarias para razonar sobre el sistema, que comprende elementos de software, relaciones entre ellos, y las propiedades de ambos (Documenting Software Architectures: Views and Beyond, 2nd Edition, Clements et al, AddisonWesley, 2010).
- La práctica recomendada define la arquitectura como la organización fundamental de un sistema, plasmada en sus componentes, sus relaciones entre sí y con el entorno y los principios que rigen su diseño y evolución (ANSI/IEEE Std 1471-2000, Recommended Practice for Architectural Description of Software-Intensive Systems).
- Una arquitectura genérica se define como una organización topológica de un conjunto de módulos parametrizados, junto con las relaciones intermodulares (Bhansali 92).



Arquitectura de software

- Origen de la noción de arquitectura de software: Dijkstra 1968.
- Wirth 1971: “stepwise refinement” (refinamiento gradual) el software debe desarrollarse teniendo en cuenta la descomposición de la tarea en subtareas y de los datos en estructuras de datos.
- DeRemer and Kron 1975: "programming-in-the-large" diseñar un sistema como composición de partes más pequeñas
- Arquitectura más a alto nivel: Mary Shaw (1989) y el término comienza a utilizarse por Perry and Wolf (1992)



Definición de arquitectura

- ¿Por qué usamos acá la definición de arquitectura en forma genérica? ¿Cuál es la diferencia con arquitectura de software? ¿Existen otras arquitecturas dentro de SI&TI?
- Si, dentro de SI&TI y el mundo de las organizaciones existen diferentes “niveles” de arquitectura.

Niveles de la arquitectura



Niveles de arquitectura

- ENTERPRISE – Define la estrategia tecnológica y de negocio de la organización para el desarrollo de sus Sistemas.
- SISTEMA – Arquitectura de Software e Infraestructura.
- SOFTWARE – Arquitectura de Software para una Aplicación o Subsistema.



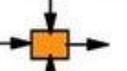
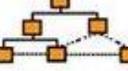
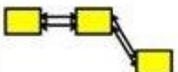
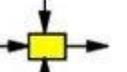
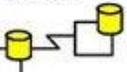
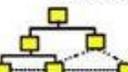
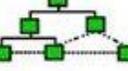
Arquitectura Empresarial

- Estructuras y comportamiento de la empresa
- Diferentes vistas empresariales
- Frameworks: TOFAG, Zachman, TAFIM, Gartner, etc.
- A continuación se presentan algunas ideas de la estructura de Zachman y TOGAF



Zachman

ENTERPRISE ARCHITECTURE - A FRAMEWORK TM

	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 Significant to the Business 	List of Business Goals/Strat 	SCOPE (CONTEXTUAL)
<i>Planner</i>	ENTITY = Class of Business Thing	Function = Class of Business Process	Node = Major Business Location	People = Major Organizations	Time = Major Business Event	Ends/Mean=Major Bus. Goal/ Critical Success Factor	<i>Planner</i>
ENTERPRISE 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 	ENTERPRISE 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/System Software 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 Strit I/O = Control Block	Node = Addresses Link = Protocols	People = Identify 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



TOGAF

- **Arquitectura de negocios:** define la estrategia de negocio, la estructura organizacional y los procesos clave de la organización. Se realiza el detalle de la estrategia de la organización, considerando el estado actual, pero también identificando posibles escenarios para el futuro.
- **Arquitectura de información:** describe la estructura de los datos físicos y lógicos de la organización y sus modelos de gestión. Se encarga del estudio, análisis, organización, disposición y estructuración de la información en espacios de información y de la selección y presentación de los datos en los sistemas de información interactivos y no interactivos.
- **Arquitectura tecnológica:** describe la estructura de hardware, software y comunicaciones requerida para dar soporte a la implantación de los sistemas de información. Esta arquitectura se vincula a los componentes y unidades tecnológicos que mantienen los sistemas de información y los medios de comunicación.
- **Arquitectura de aplicaciones:** conceptos fundamentales o propiedades de un sistema en un entorno definido, encarnado en elementos, las relaciones que existen entre ellos; y los principios que guían su diseño y evolución (definición IEEE)



Niveles de arquitectura

- Nosotros enfocaremos en estos niveles:
 - SISTEMA – Arquitectura de Software e Infraestructura.
 - SOFTWARE – Arquitectura de Software para una Aplicación o Subsistema.
- Sin embargo, el modelo TOGAF nos puede dar una buena estructura para pensar la arquitectura en términos de sistemas y software



Concepto de vista

- ¿La arquitectura se refiere a aspectos técnicos y de software y hardware?
- Veamos el concepto de vista y algunos modelos...
 - Una vista es una presentación de un modelo. Es una descripción completa de un sistema desde una perspectiva particular (Kruchten).
 - Algunos modelos de vistas propuestos: Vista 4+1 (Kruchten) - Vistas del SEI – UML
 - 4+1: Vista lógica + Vista de procesos + Vista de despliegue + Vista física
- Mas adelante hablaremos más de esto, por ahora... suficiente para ver que el concepto de arquitectura es muy amplio

Arquitectura de software



Definiciones clásicas

- La arquitectura de software es el estudio de la estructura a gran escala y el rendimiento de los sistemas de software [Lane 90].
- La arquitectura de software se ocupa del diseño e implementación de la estructura de alto nivel del software. Es el resultado de ensamblar un cierto número de elementos arquitectónicos en algunas formas para satisfacer los principales requisitos de funcionalidad y rendimiento funcionalidad y rendimiento, como la escalabilidad y la disponibilidad. Software La arquitectura del software se ocupa de la abstracción, la descomposición y composición, con el estilo y la estética [Kruchten 94].



Definiciones clásicas

- Estructura del sistema que consiste en módulos activos, un mecanismo para permitir la interacción entre estos módulos, y un conjunto de reglas que gobiernan la interacción [Boasson 95]
- La estructura de los componentes de un programa/sistema, sus interrelaciones, y los principios y directrices que rigen su diseño y evolución en el tiempo [Garlan 95]
- La arquitectura de un sistema de software complejo sistema de software complejo es su "estilo y método de diseño y construcción"[BHayes-Roth 95]



Definiciones clásicas

- Una arquitectura de software se representa utilizando los siguientes conceptos [Moriconi 94]:
 1. Componente: Un objeto con existencia independiente, por ejemplo, un módulo, proceso procedimiento o variable.
 2. Interfaz: Un objeto tipificado que es un punto lógico de interacción entre un componente y su entorno.
 3. Conector: Un objeto tipificado que relaciona puntos de interfaz, componentes o ambos.
 4. Configuración: Una colección de restricciones que conectan los objetos en una arquitectura específica.
 5. Mapeo: Una relación entre los vocabularios y las fórmulas de una arquitectura abstracta y una arquitectura concreta



Definición de arquitectura de software

- La arquitectura de software de un programa o sistema informático es la estructura o estructuras del sistema, que comprenden elementos de software, las propiedades visibles externamente propiedades de esos elementos, y las relaciones entre ellos (Software Architecture in Practice, 2nd edition, Bass, Clements, Kazman; AddisonWesley 2003).



Beneficios de la arquitectura

- La calidad y la longevidad de un sistema está determinado en gran medida por su arquitectura.
- Beneficios del diseño arquitectónico:
 - Fuerza la articulación entre objetivos del negocio y los atributos de calidad
 - Priorización entre objetivos/atributos conflictivos
 - Fuerza la definición clara del enfoque arquitectónico proporcionando una guía para el equipo de desarrollo
 - Mejora la Calidad del Producto (La calidad no se puede agregar al Final)



Características de la arquitectura

- Debe ser correctamente comunicada y entendida por cada stakeholder según sus propias necesidades
- Debe ser capaz de evolucionar a lo largo del proyecto de la mano de nuevos requerimientos.
- Debe ser la arquitectura más simple posible que cumpla con los requerimientos del Sistema. (GOOD ENOUGH)
- Debe permitir el análisis de medidas cuantitativas y de evaluar el cumplimiento de los atributos cualitativos (ATAM)



Entradas de la arquitectura

- Requerimientos Funcionales
- Atributos de Calidad y Requerimientos No Funcionales
- Restricciones de Negocio, técnicas, etc.
- Futuros Requerimientos
- Experiencia del Arquitecto



Atributos de calidad





Atributos de calidad

- Los atributos de calidad pueden entrar en conflicto
- Ejemplos de conflictos típicos:
 - Seguridad vs. Usabilidad
 - Performance vs. Modificabilidad
 - Portabilidad vs. Performance
- Estrategias de resolución:
 - Priorización / Jerarquización
 - Balanceo / Good Enough

Infraestructura



Infraestructura

- La infraestructura es el conjunto de componentes, nodos y configuraciones de red que permiten realizar el despliegue de un software
 - Componentes: software, sistemas operativos, middleware, máquinas virtuales
 - Nodos: físicos, servers, procesadores, memoria, almacenamiento
 - Networking: routers, firewalls, swichs



Decisiones de Infraestructura

- On Premise vs Cloud Computing
- On Premise: “in situ”, “instalaciones propias”
- Cloud Computing: red de servicios que ofrece una serie de herramientas que permiten administrar los recursos en distintos niveles: hardware, servicios de plataformas de desarrollo y aplicaciones para el usuario final.
- IAAS: nivel de Cloud Computing asociado a la infraestructura



Niveles de servicio de Cloud Computing





Variables de decisión

- Sensibilidad de la información
- Capacidad de almacenamiento y procesamiento
- Costos
- Elasticidad bajo demanda
- Conectividad



Dimensionamiento y escalabilidad

- En la infraestructura es importante tener en cuenta el dimensionamiento (Hardware Sizing)
- Se entiende por escalabilidad a la capacidad de adaptación y respuesta de un sistema con respecto al rendimiento del mismo a medida que aumentan de forma significativa el número de usuarios del mismo.
- Escalabilidad vertical: más recursos sobre los nodos existentes.
- Escalabilidad horizontal: incorporación de nodos. Estrategias de clusterización (conjunto de computadoras que se comportan como una única computadora) que permite balanceo de carga



Comunicación de la infraestructura

- La relación entre componentes de software e infraestructura se presenta en el Diagrama de Despliegue

INTRODUCCIÓN A ARQUITECTURA

Autor: Luciano Straccia
