

# INGENIERIA WEB Y DE SERVICIOS.

Ing. Walter Arias Aguirre

# DOCENTE

- WALTER ARIAS AGUIRRE – INGENIERO DE SISTEMAS.
  - Especialista en Telecomunicaciones – Universidad de Manizales.
  - Especialista en Desarrollo de Software – Universidad Católica de Pereira.
  - En la industria de las TICS desde 1998.
  - Instructor SENA desde el 2007.
  - Email: [walter.arias.aguirre@gmail.com](mailto:walter.arias.aguirre@gmail.com)
  - WhatsApp : 3225684123

# ACUERDO DE AULA

- No consumir alimentos en el aula.
- Celular en Silencio.
- Actitud asertiva y constructiva.

# CRONOGRAMA Y CONDICIONES

- **14** SESIONES APROXIMADAMENTE.
- NOTAS:
  - 40 % Notas (Exámenes, test).
  - 60% Mini proyectos
  - Asistencia ?

# PLAN DE TRABAJO

TEMA	ACTIVIDAD
SOA : INTRODUCCIÓN, MODELADO, COMPONENTES E IMPLEMENTACIÓN.	
SERVICIOS WEB (COMPUTACIÓN EN LA NUBE)	
EL PROTOCOLO HTTP A FONDO	TALLER PRÁCTICO
JSON: EL FORMATO DE INTERCAMBIO DE LA WEB	TALLER PRÁCTICO
CONSUMIENDO SERVICIOS WEB (JAVASCRIPT+ HTML)	TALLER PRÁCTICO
CREANDO SERVICIOS (API REST) CON JAVASCRIPT	TALLER PRÁCTICO
MINIPROYECTO DE CLASE No 1	
CREANDO SERVICIOS (API REST) CON PHP (LARAVEL)	TALLER PRÁCTICO
MINIPROYECTO DE CLASE No 2	

# REQUISITOS Y HERRAMIENTAS

- Conocimientos previos en programación básica HTML, CSS, JS, PHP.
- Editor de código : VSCODE o cualquiera.
- XAMPP, WAMP, LARAGON (instalador PHP 7.X, MYSQL, APACHE)
- POSTMAN o PLUGIN REST CLIENT de VSCODE

# ARQUITECTURA DE SOFTWARE

¿Es necesaria?

# ¿Arquitectura?

- La arquitectura de software es el conjunto de decisiones de diseño importantes para organizar el software y promover los atributos de calidad deseados de acuerdo a las restricciones impuestas.

**F.U.R.P.S**





# Otra Definición.

- “La organización fundamental de un sistema representado por sus componentes, sus relaciones entre ellos y el ambiente, y los principios que gobiernan su diseño y evolución”

[IEEE 1471].

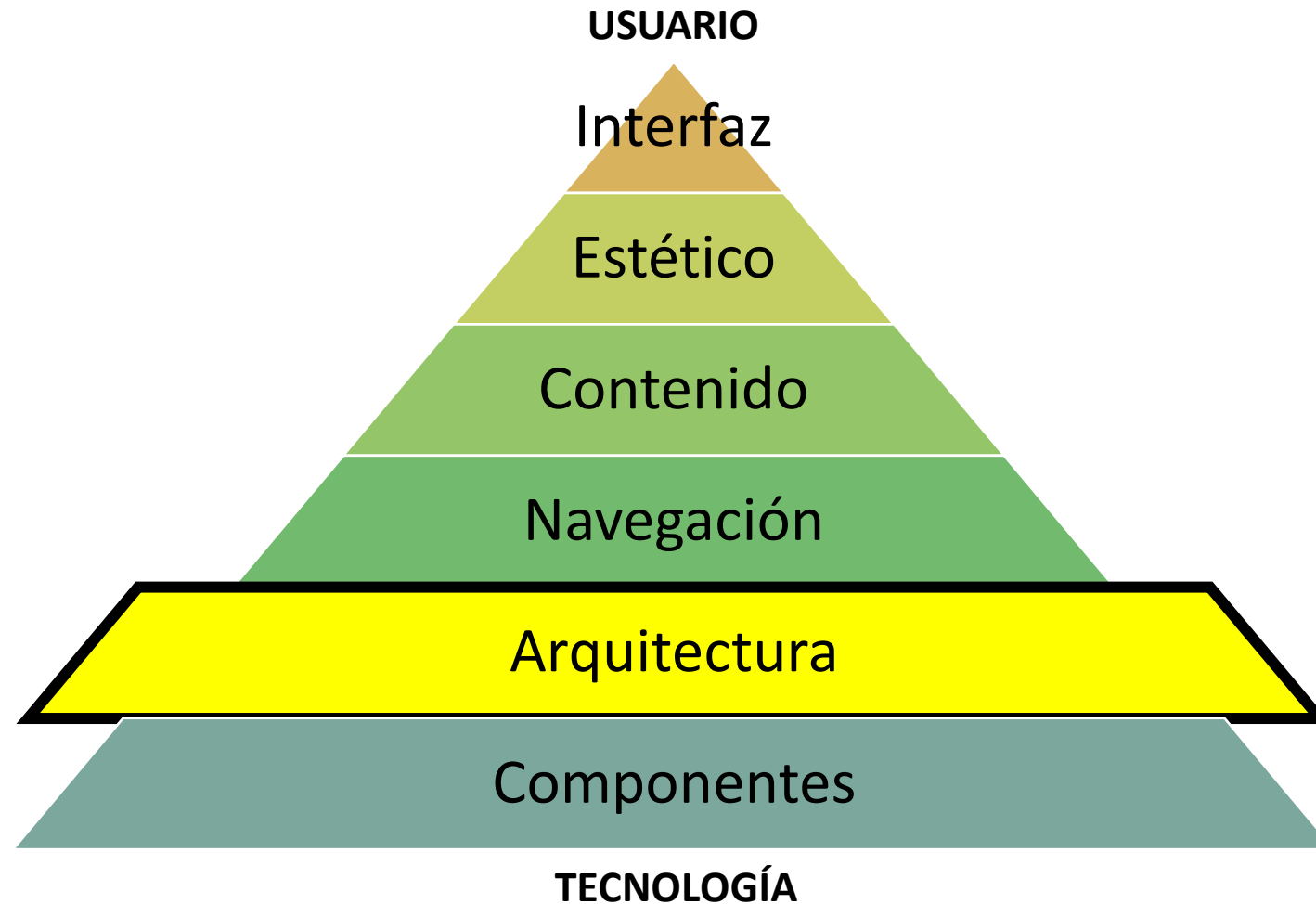
# FURPS Atributos de calidad del software.

ATRIBUTO		DEFINICIÓN
Funcionality	Funcionalidad	La funcionalidad es la capacidad del sistema para realizar el trabajo para el que fue concebido.
Usability	Usabilidad	La usabilidad tiene que ver con la facilidad con la que el usuario realiza una tarea y el tipo de apoyo al usuario que ofrece el sistema.
Reliability	Fiabilidad	La fiabilidad del software es la probabilidad de que un programa informático funcione sin fallos durante un periodo determinado en un entorno específico.
Perfomance	Rendimiento	Cuando se producen eventos -interrupciones, mensajes, peticiones de usuarios u otros sistemas, o eventos de reloj que marcan el paso del tiempo, el sistema o algún elemento del sistema, debe responder a ellos a tiempo.
Supportability	Soporte y mantenimiento	La sustentabilidad del software es la capacidad de soportar un sistema de software durante toda la vida del producto. Esto implica la satisfacción de cualquier necesidad o requisito necesario, pero también el suministro de equipos, infraestructura de apoyo, software adicional, instalaciones, mano de obra o cualquier otro recurso necesario para mantener el software operativo y capaz de satisfacer su función.

# RESTRICCIONES DE DISEÑO

RESTRICCIÓN	EJEMPLO
Diseño	Solo para Iphone, debe ser responsivo, para discapacitado visual, etc.
Implementación	Escrito en Java, Python, con soporte de lenguaje en Ruso o Chino.
Interfaz	Se conectará a un sistema externo - DIAN, a red ATM de cajeros automáticos.
Físicas	Se ejecutará en una cámara digital, en un aparato médico, en dispositivo de toma de inventarios

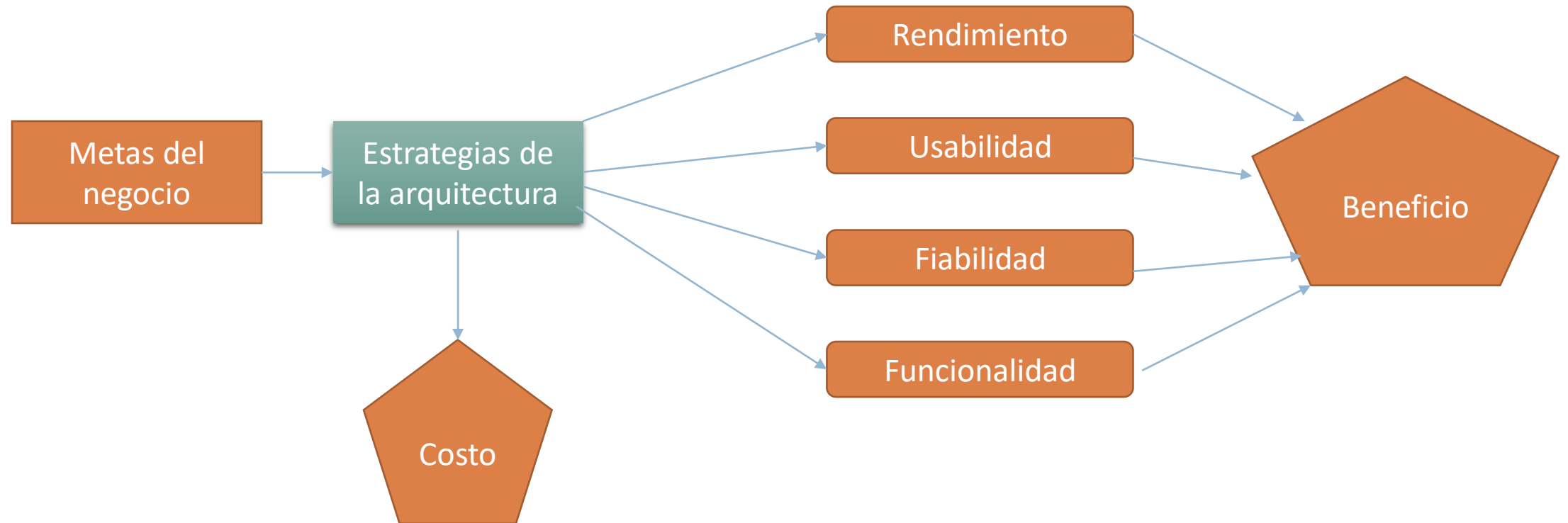
# PIRÁMIDE DE DISEÑO WEBAPPS (Pressman)



# ARQUITECTURA Y METAS DEL NEGOCIO

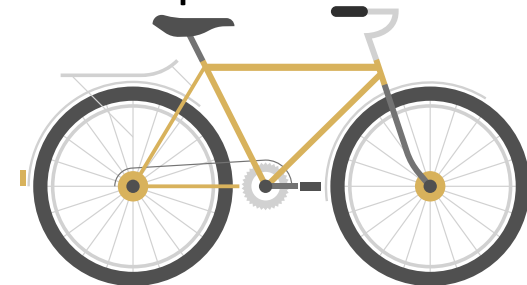
- Tal vez el trabajo más importante de un arquitecto es ser un punto de apoyo donde las decisiones técnicas y de negocio se encuentran e interactúan.
- La arquitectura (el arquitecto) debe traducir continuamente las necesidades y objetivos del negocio en realizaciones técnicas.

# METAS DEL NEGOCIO Y ARQUITECTURA



# PATRONES DE DISEÑO ARQUITECTONICO.

- Los patrones de diseño son unas técnicas para resolver problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.
- Un patrón de diseño resulta ser una solución a un problema de diseño.
- Para que una solución sea considerada un patrón debe poseer ciertas características:
  - haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores.
  - ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias.

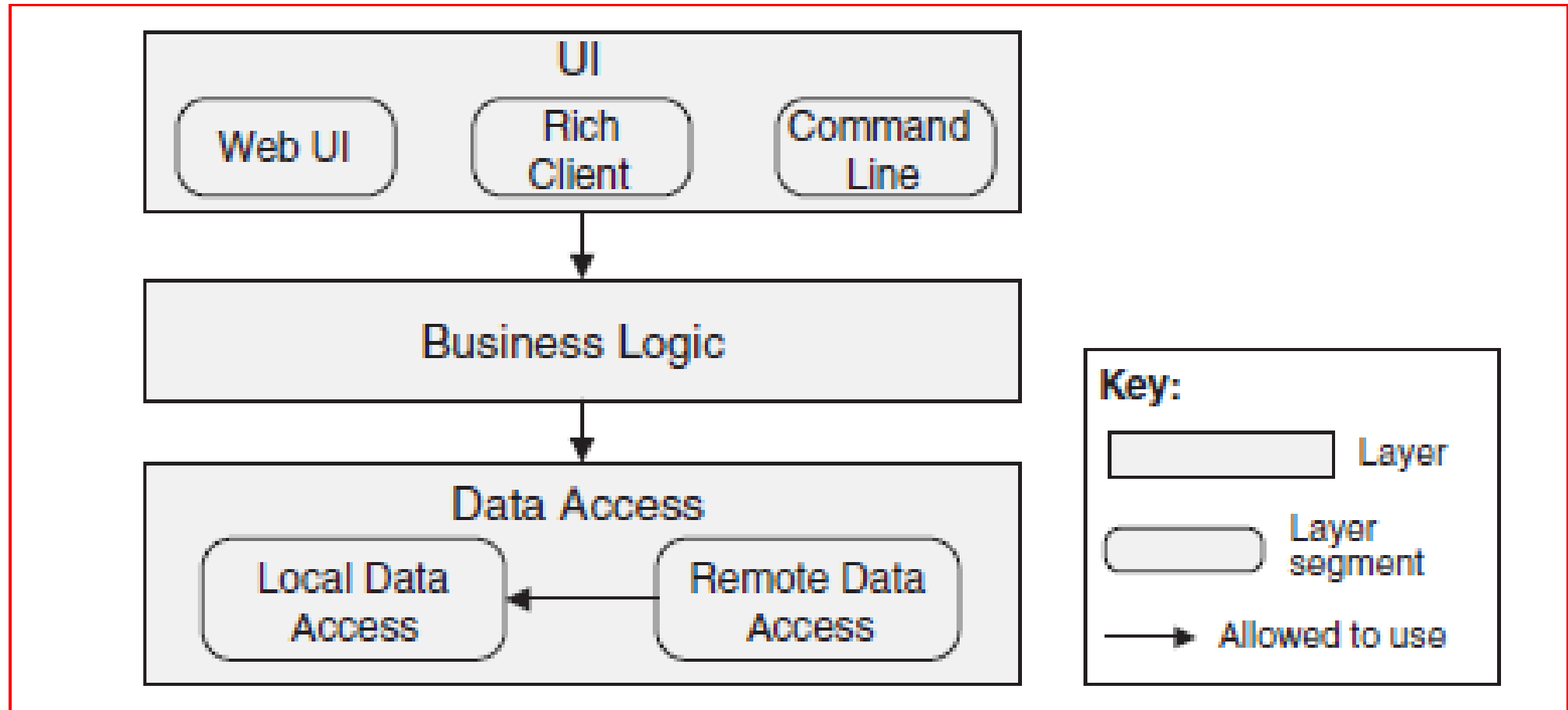


# ALGUNOS PATRONES ARQUITECTÓNICOS

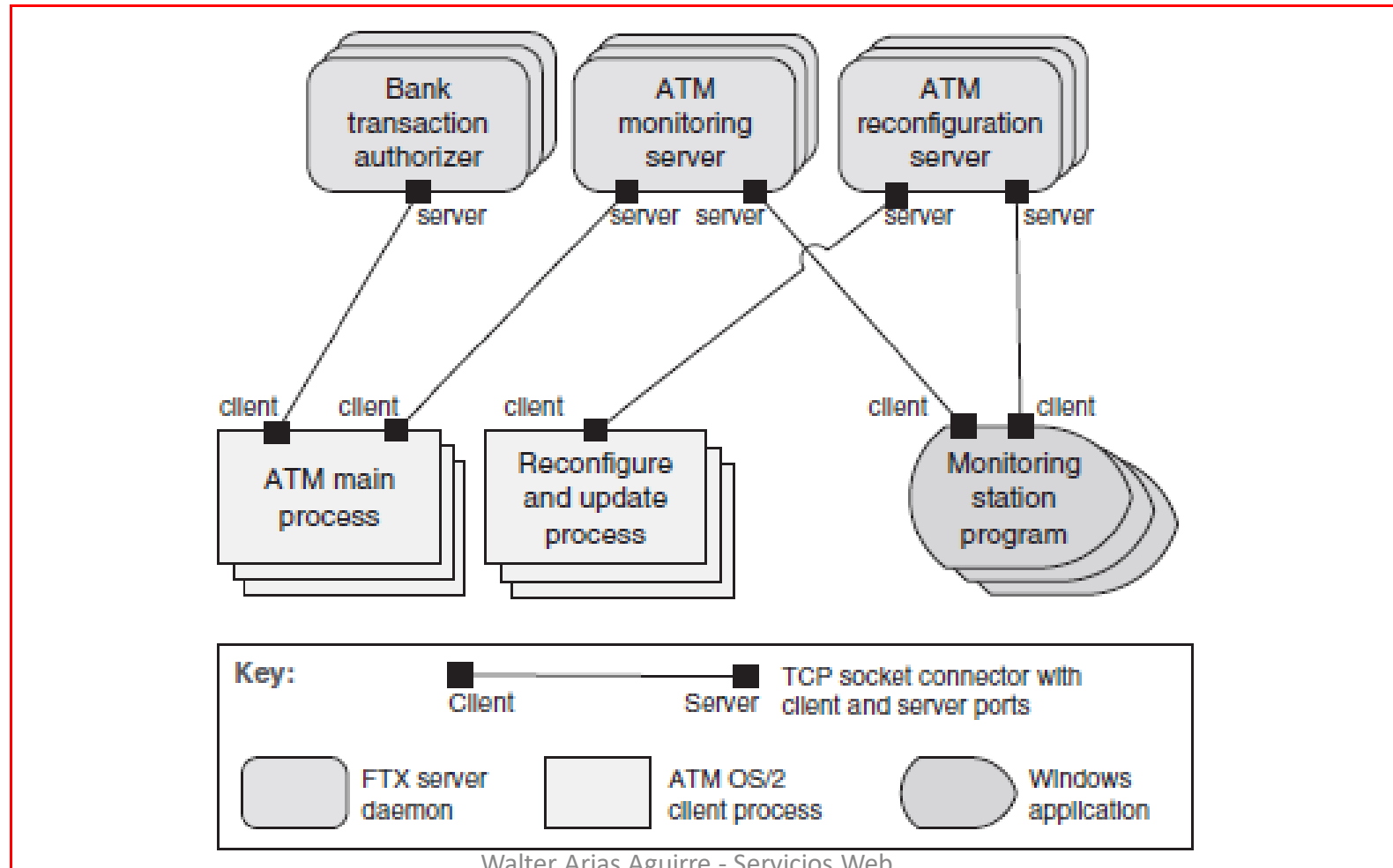
- Por capas (presentación, negocio, persistencia).
- Cliente – Servidor.
- Modelo Vista Controlador.
- SOA – Orientada a Servicios.
  - SOAP
  - REST



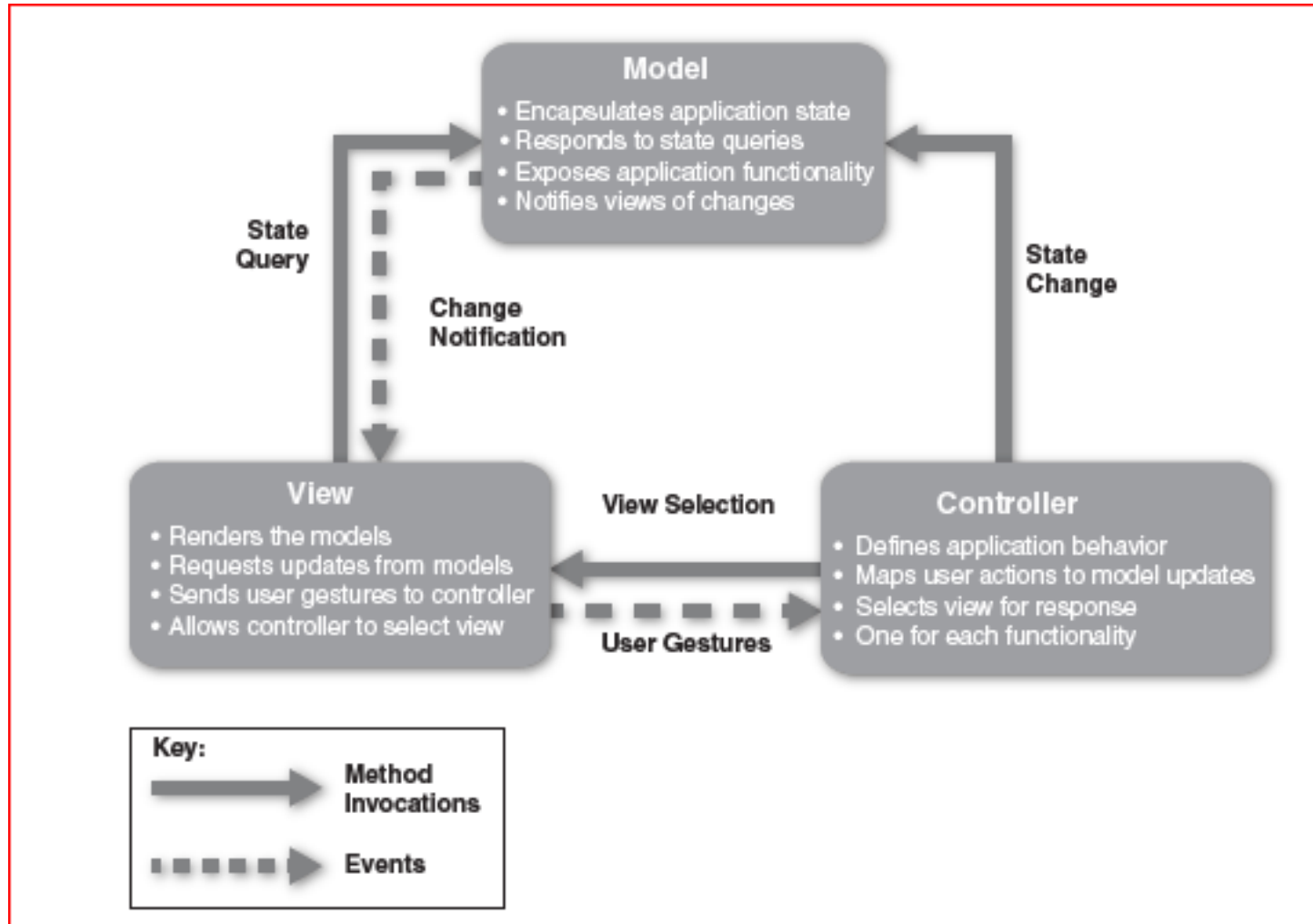
# POR CAPAS



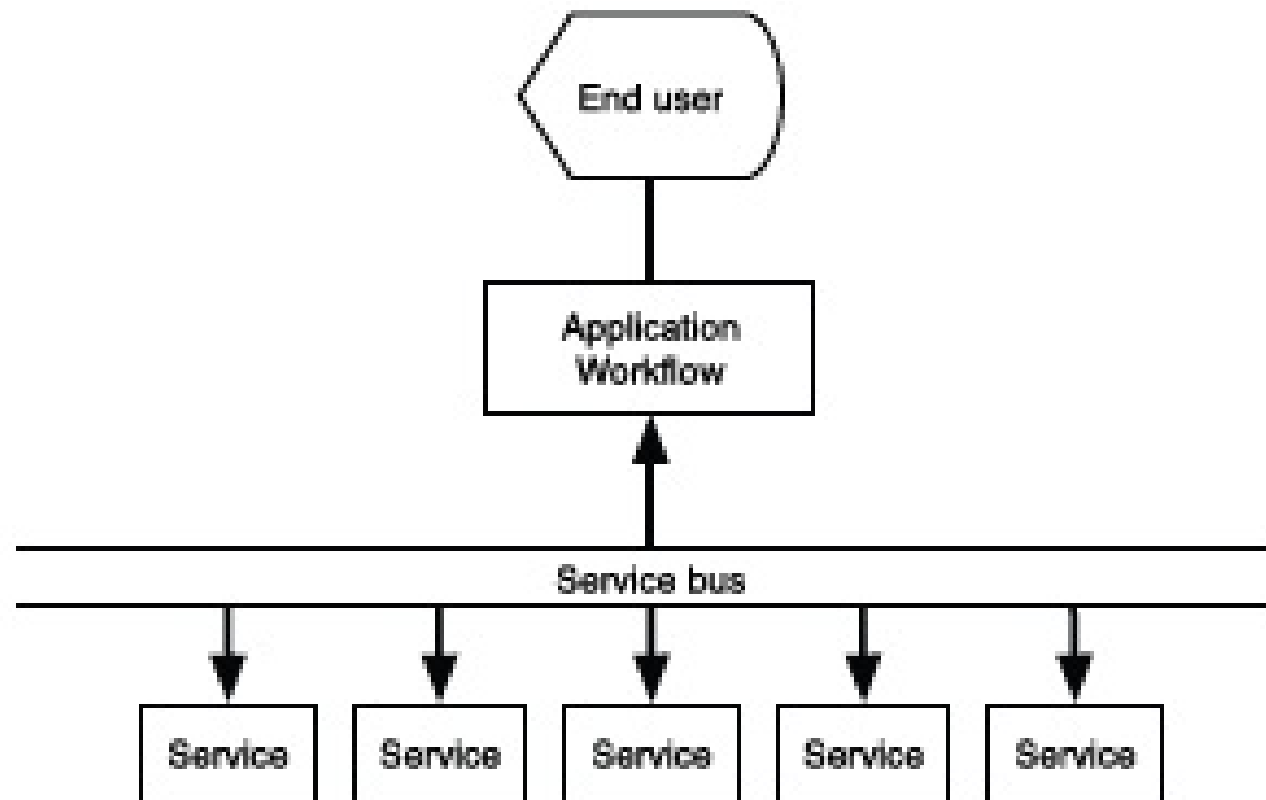
# CLIENTE SERVIDOR



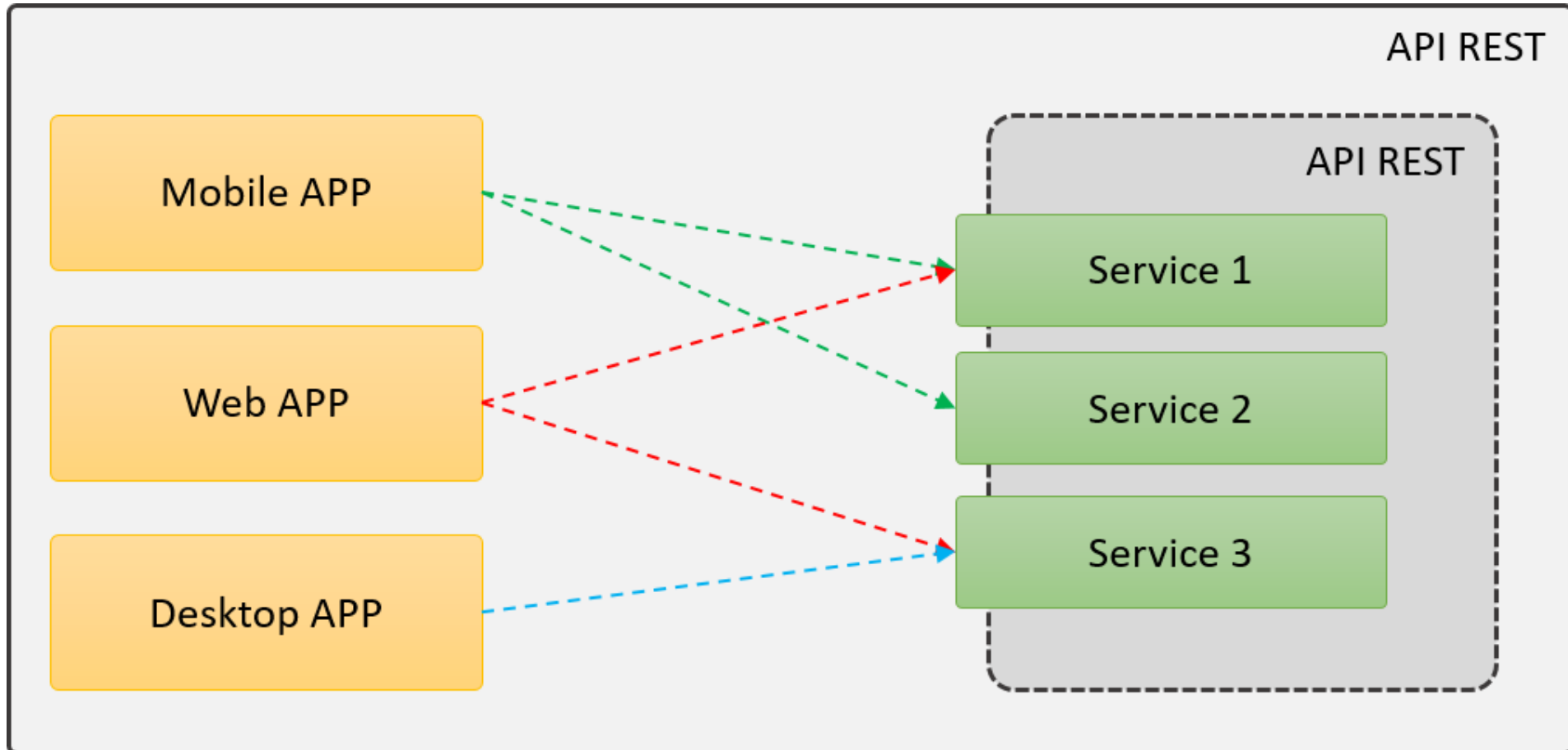
# MODELO VISTA CONTROLADOR



# S.O.A. (TECNOLOGIA S.O.A.P.)



# S.O.A. (TECNOLOGIA R.E.S.T.)



# HISTORIA

# ARQUITECTURA DE LAS APLICACIONES

60'S Y  
70'S



EJEMPLO : IBM S36, IBM AS400, UNIX,  
XENYX

# ARQUITECTURA DE LAS APLICACIONES

80 ' S



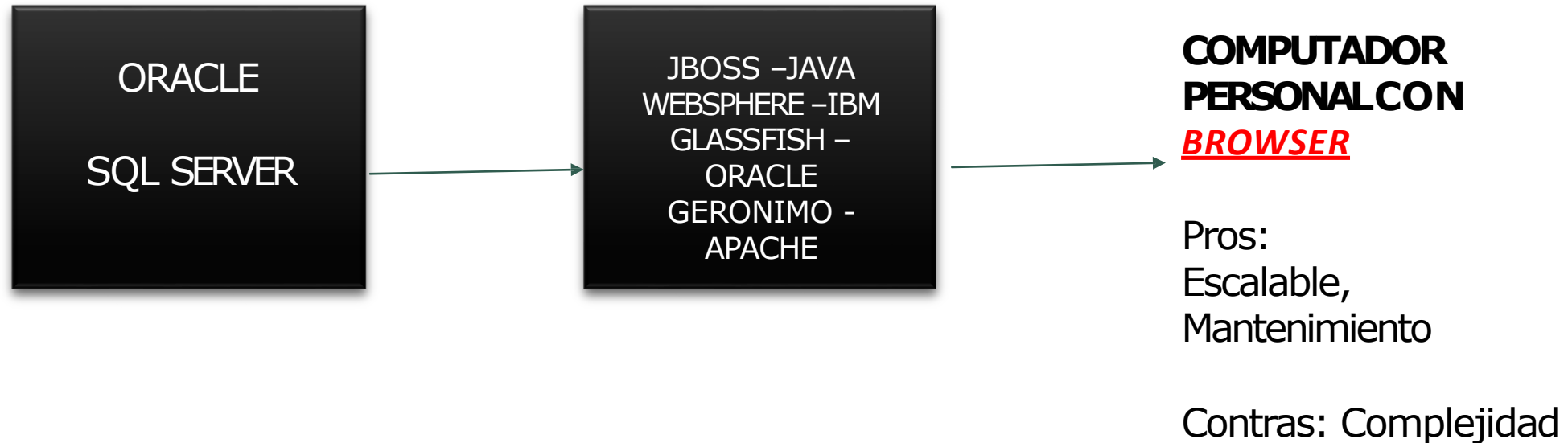


# ARQUITECTURA DE LAS APLICACIONES FIN 90 'S 2000



TREE THIER – TRES  
CAPAS

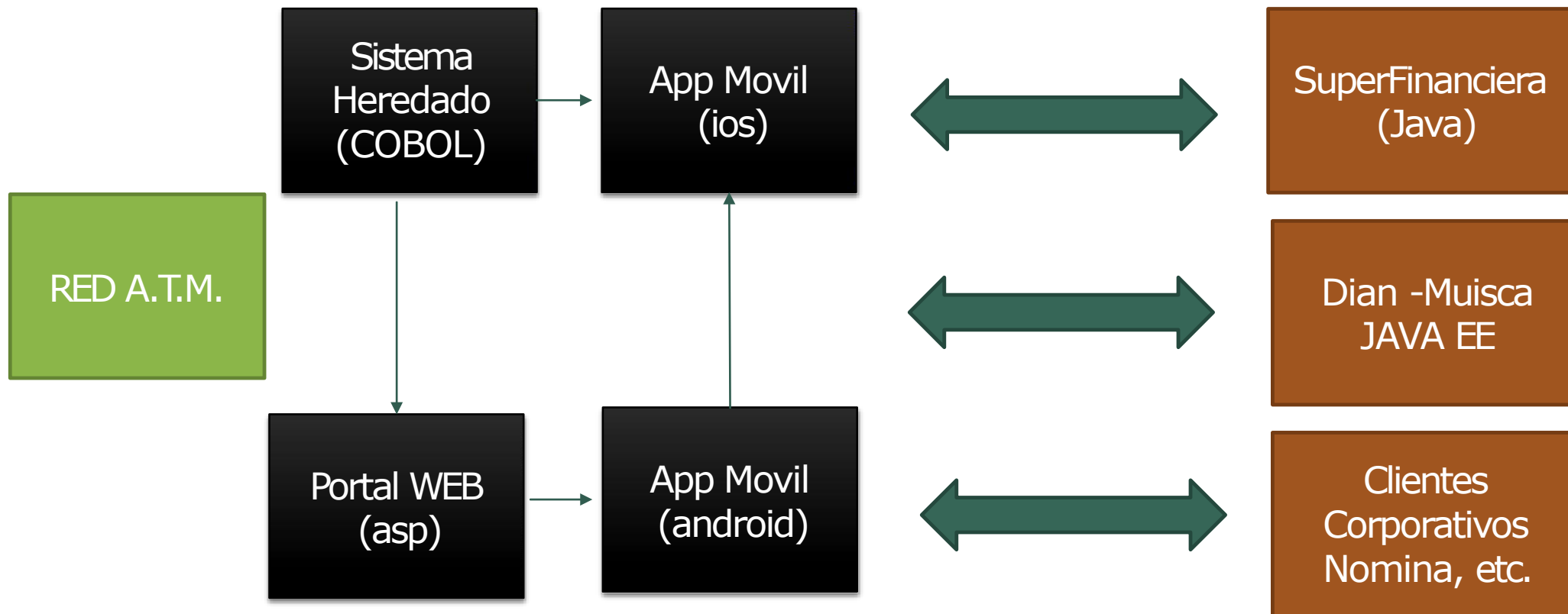
# ARQUITECTURA DE LAS APLICACIONES FIN 90 'S 2000



TREE THIER – TRES  
CAPAS

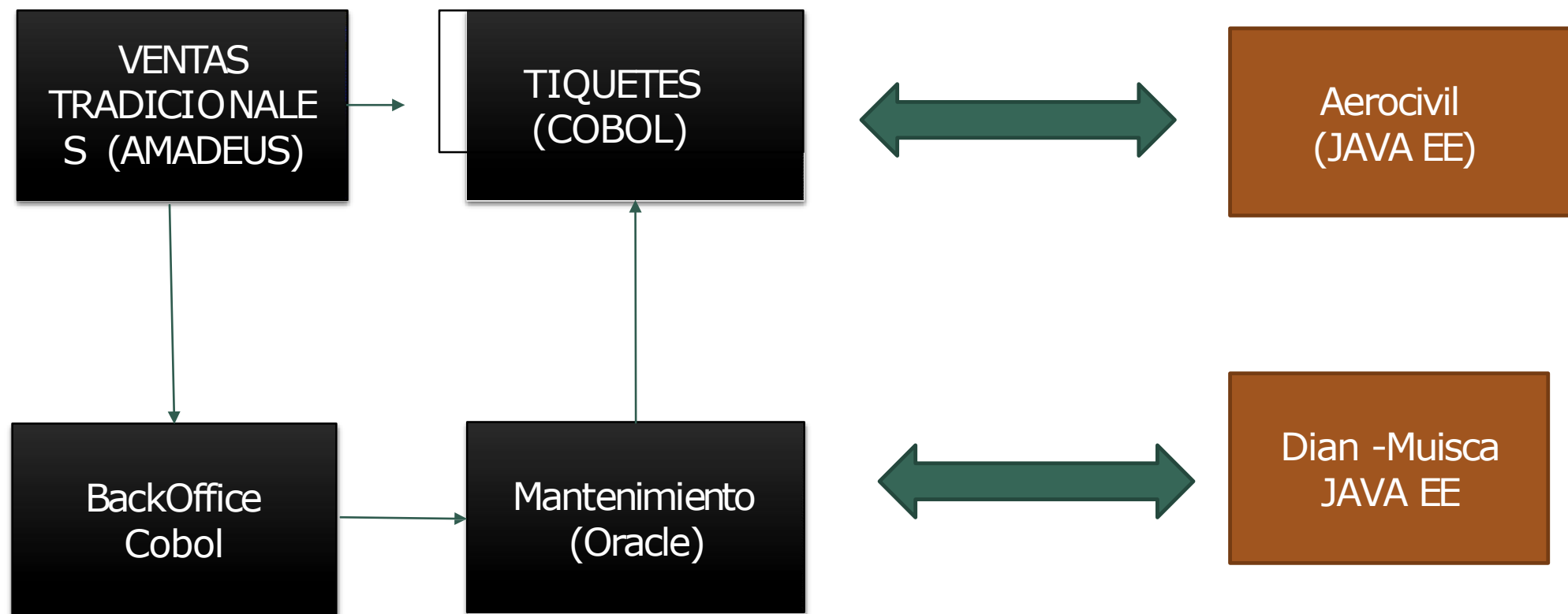
## -UN BANCO ACTUAL (DAVIVIENDA) T.I.

2000'S



+ 6 Millones clientes, 743 oficinas, 2000 cajeros atm

# 2000'S - ? AVIANCA T.I.



Holding 10 aerolíneas en suramerica

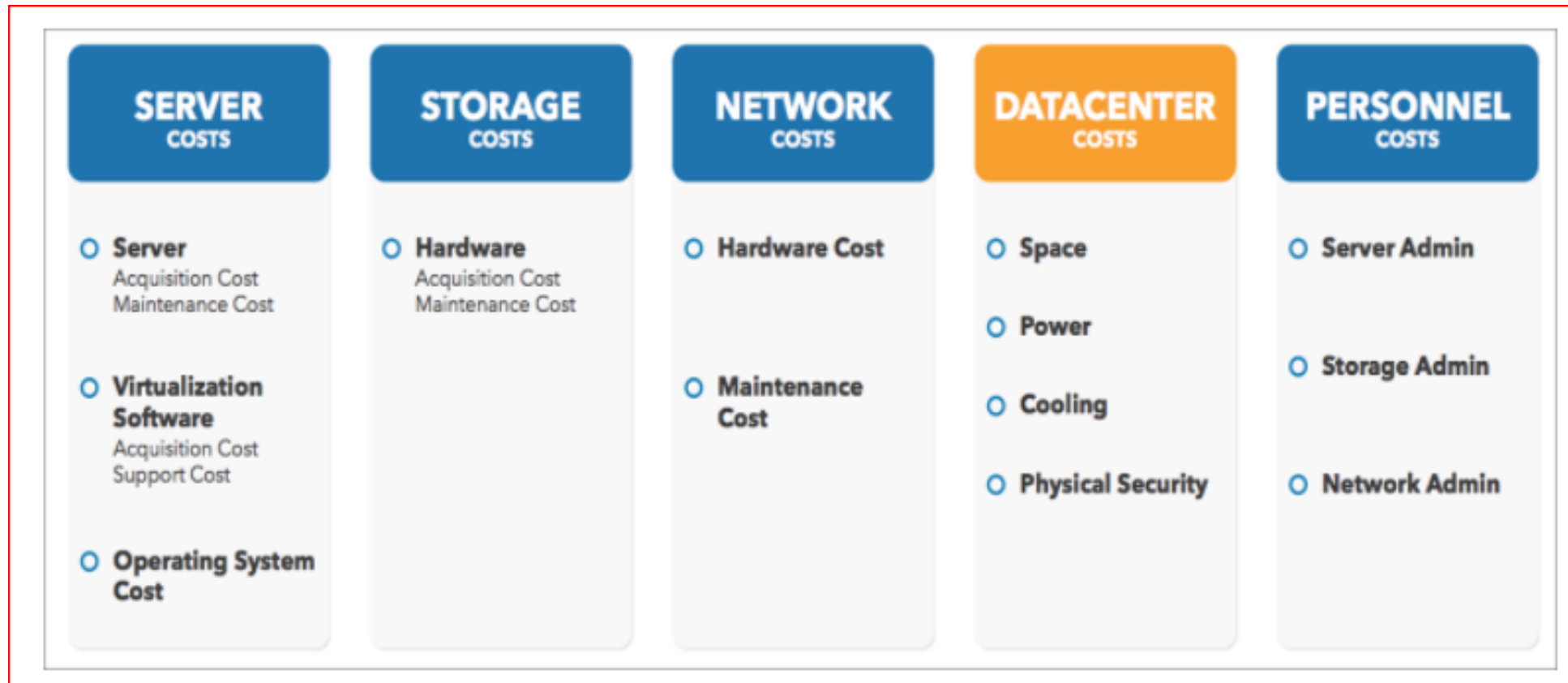
# ACTUALIDAD

# RETOS DIARIOS PARA DEPARTAMENTOS DE T.I.

- TIEMPO REAL
- ONLINE, “TODO CONECTADO”
- CUALQUIER DISPOSITIVO PUEDE CONSUMIR
- SERVICIO 7/24/365
- T.C.O([https://es.wikipedia.org/wiki/Coste\\_total\\_de\\_propiedad](https://es.wikipedia.org/wiki/Coste_total_de_propiedad))
- MANTENER SISTEMAS HEREDADOS.



# Pilares para calculo de T.C.O. Datacenter



# MAS INFO DE T.C.O.

- <https://estudyando.com/costo-total-de-propiedad-definicion-componentes-y-ejemplos/>
- <https://www.the-eshow.mx/post/que-es-tco-y-como-se-calcula>
- <https://azure.microsoft.com/es-es/pricing/tco/calculator/>



# DETALLES TÉCNICOS

- DIFERENTES LENGUAJES DE PROGRAMACIÓN.
- DIFERENTES SISTEMAS OPERATIVOS.
- DIFERENTES ARQUITECTURAS.
- DIFERENTES ESTÁNDARES DE ALMACENAMIENTO DE DATOS (PLANOS, XLS, BASE DE DATOS, DOC, JPG, XML, JSON.....etc)

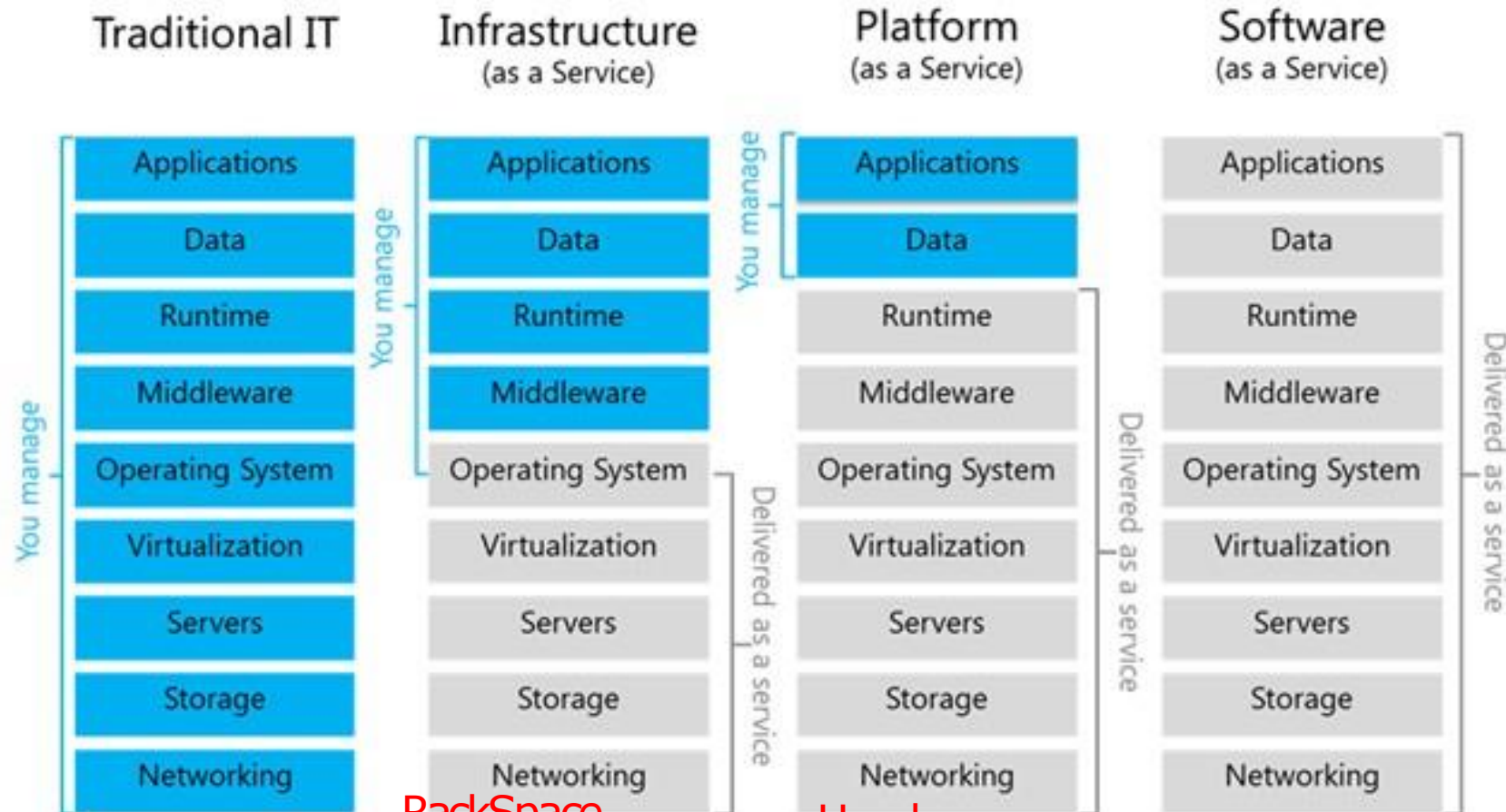


# TENDENCIAS EN INFRAESTRUCTURA DE TECNOLOGIA EMPRESARIAL

CLOUD COMPUTING

# SERVICIOS : IAAS, PAAS, SAAS

(CLOUD  
COMPUTING)



RackSpace

Vcloud

Amazon Web Service

Heroku

Google App Engine

Firebase

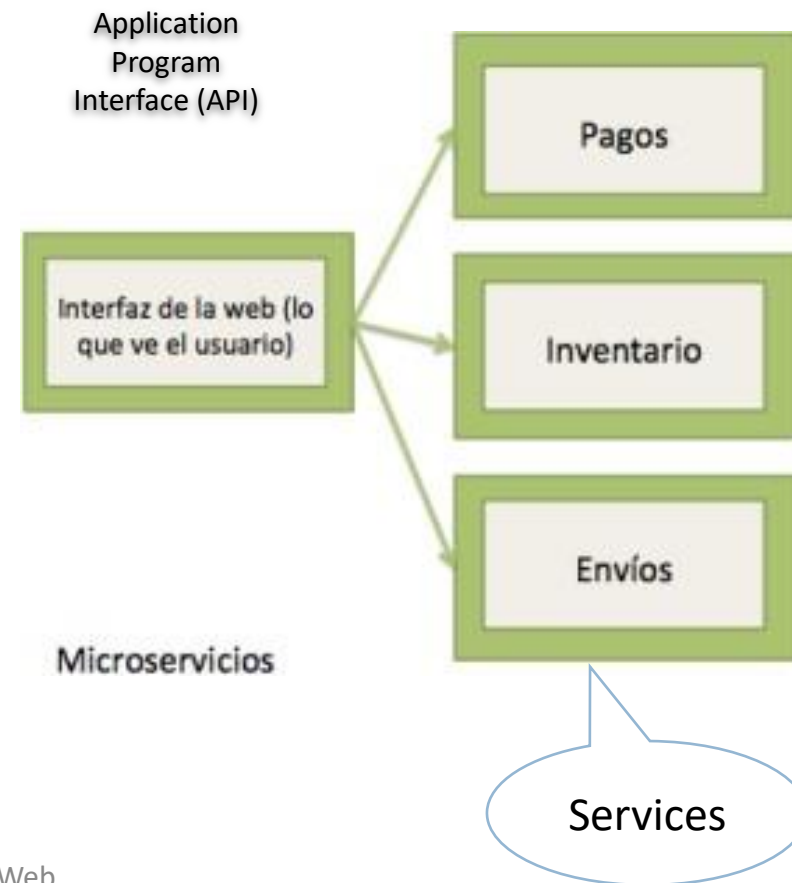
gmail  
dropbox

# FUTURO: LOS MICROSERVICIOS(TIENDA VIRTUAL)

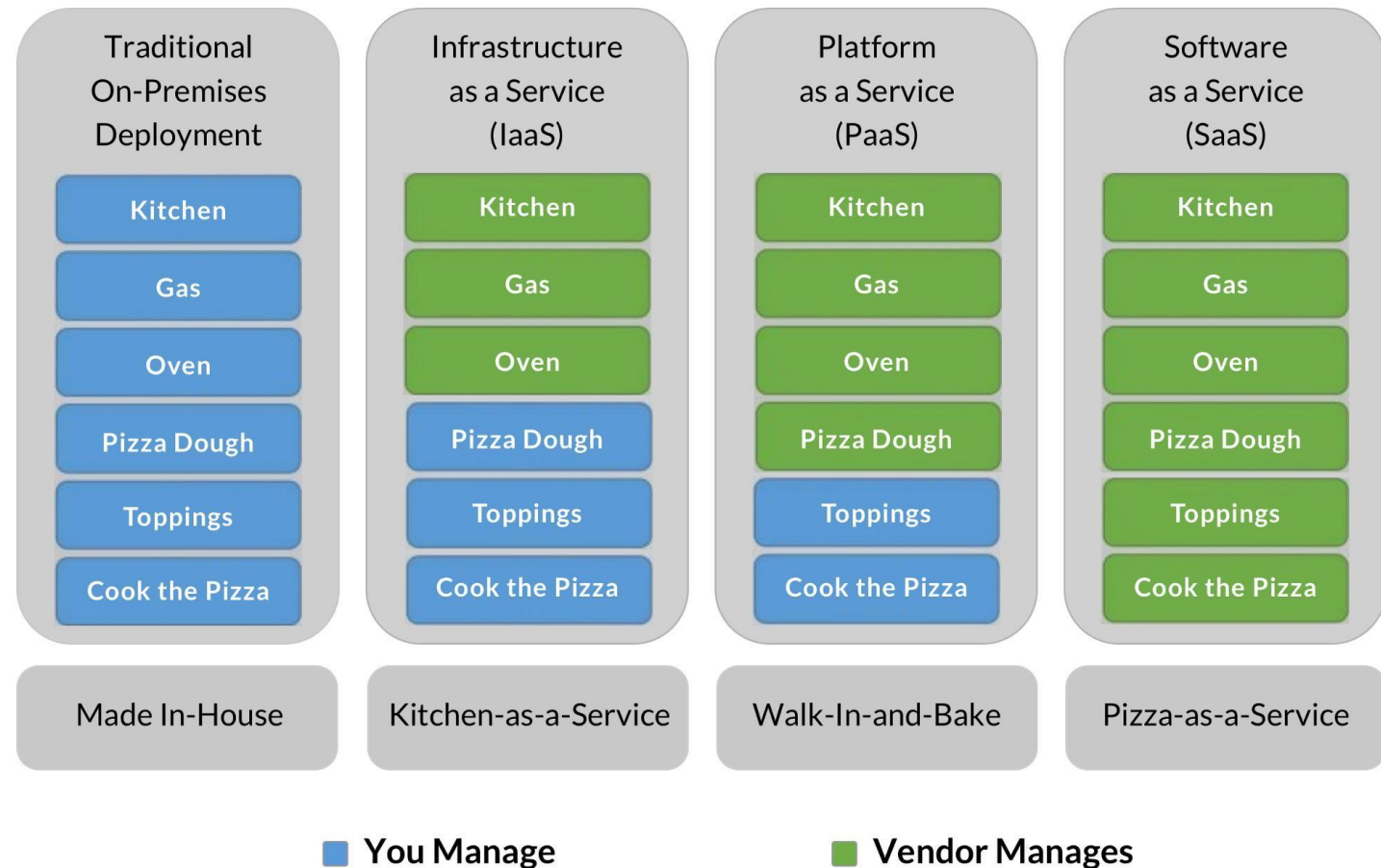
## MONOLITICO



## CON MICROSERVICIOS



## New Pizza as a Service





# ¿SOLUCIÓN ?



# ARQUITECTURA ORIENTADA A SERVICIOS – S.O.A.

# Service Oriented Architecture

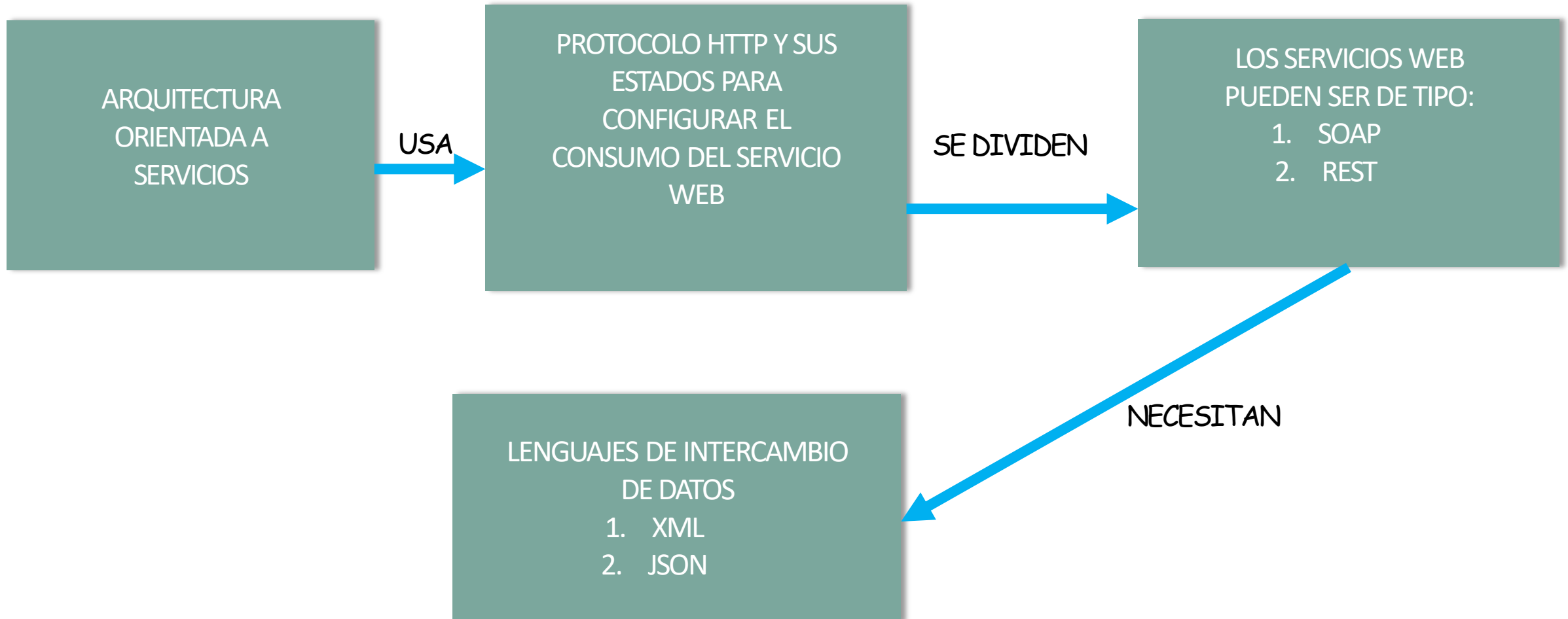
- SOA (Arquitectura orientada a servicios) es un PATRON DE DISEÑO ARQUITECTONICO para la integración de aplicaciones.
- Permite a las organizaciones unir los objetivos de negocio, en cuanto a flexibilidad de integración con sistemas legados (Heredados) y alineación directa a los procesos de negocio, con la infraestructura de TI.



# Service Oriented Architecture

- No es una tecnología, ni un producto que se pueda comprar.
- Se basa en services (servicios)
- Services: Representación lógica de una actividad del negocio (Crear factura, Imprimir ventas)
- Para interoperar **independientemente** se usan los SERVICIOS WEB (protocolo HTTP)

# S.O.A.

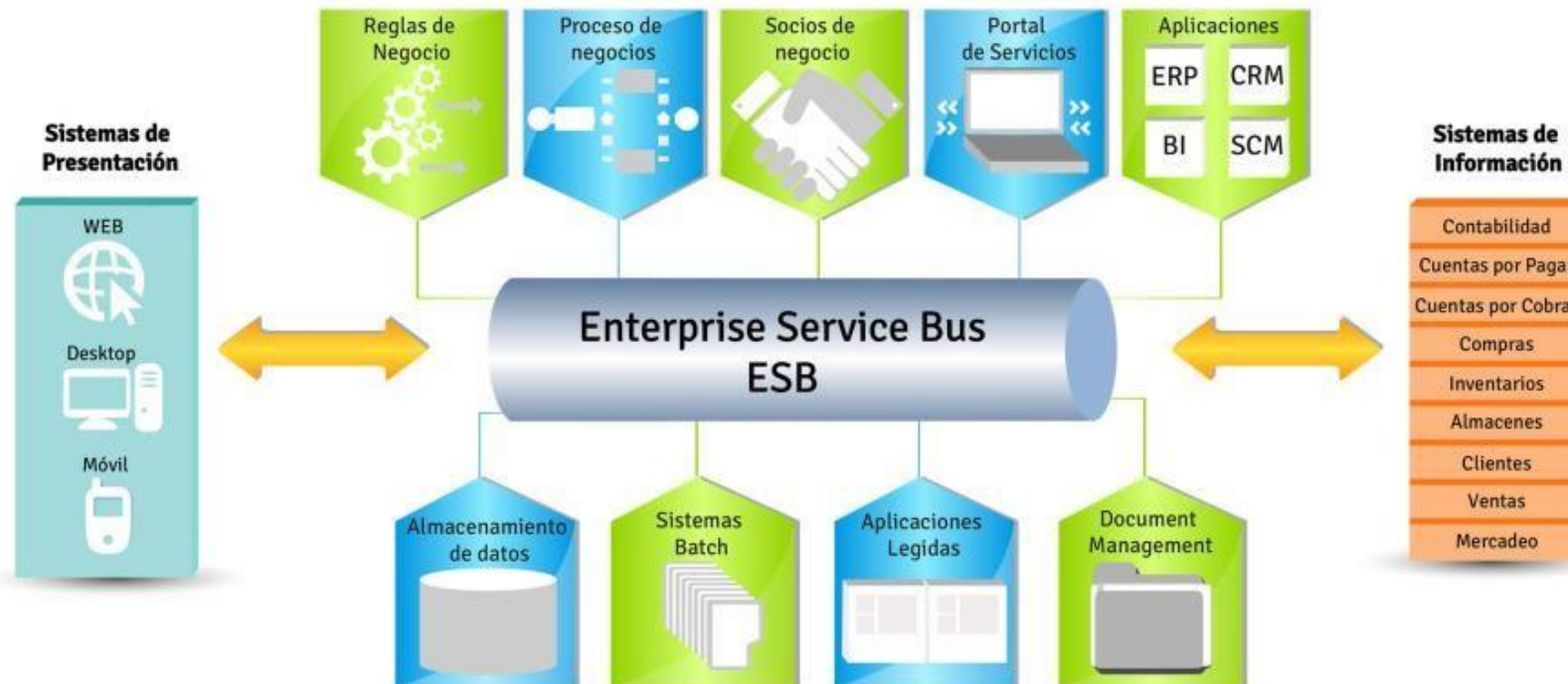


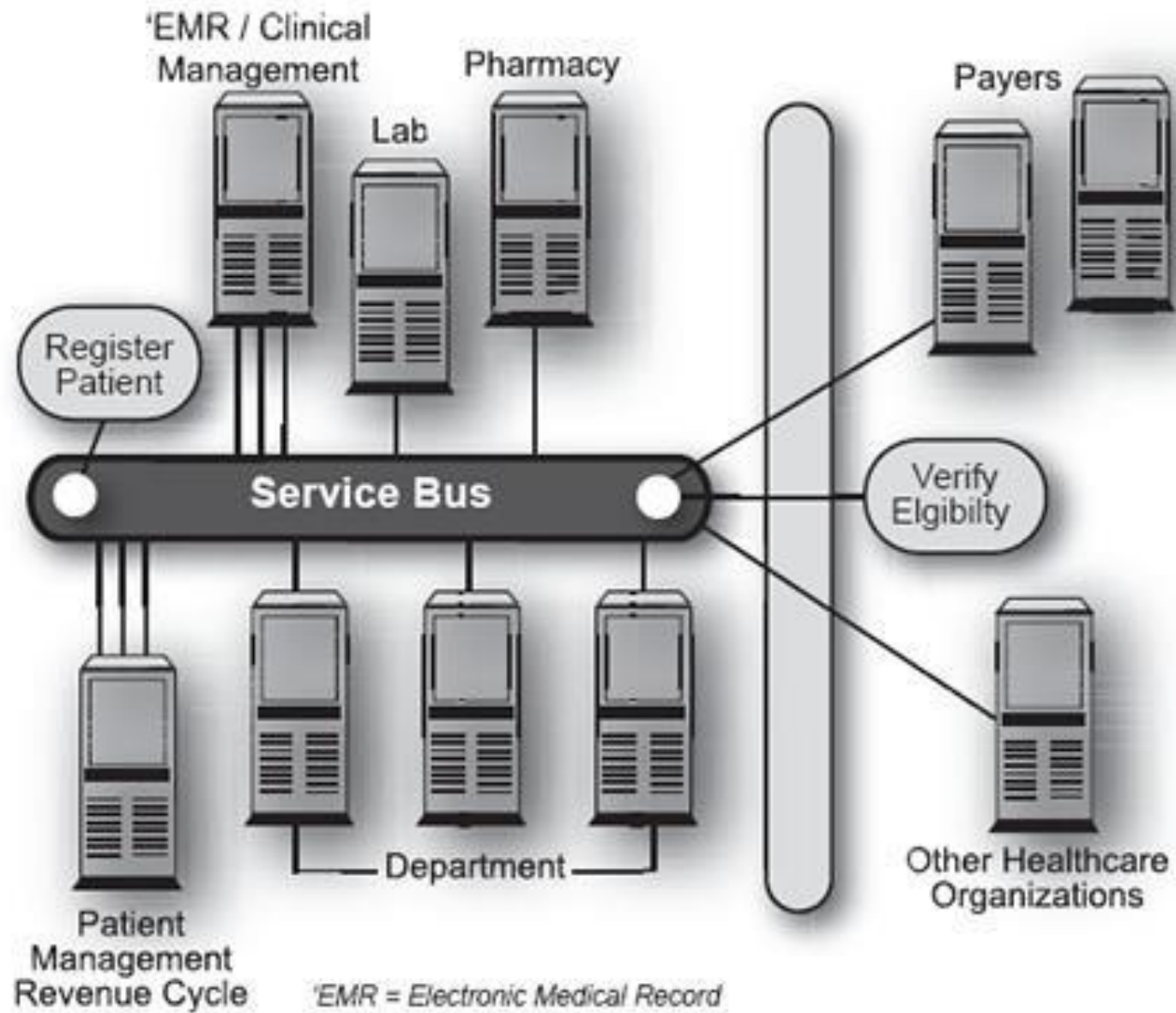
# UN SERVICIO WEB - WEB SERVICE

- Se basa en OPEN INTERNET PROTOCOLS
- Es un componente de aplicación programable accesible a través de OIP

Le puedes preguntar a un sitio web o dispositivo por la descripción de sus SERVICIOS WEB o usando servicio de directorios UDDI	UDDI : Universal Description and integration.
Los servicios web se definen en términos de formatos y ordenamiento de los mensajes	WSDL: WEB SERVICES DESCRIPTION LANGUAGE.
Los consumidores de servicios web pueden enviar y recibir mensajes usando SOAP o REST	SOAP: SIMPLE OBJECT ACCESS PROTOCOL REST: REPRESENTATIONAL STATE TRANSFER
Todo es posible gracias a los protocolos abiertos de internet	Xml, json, http,smtp

# Arquitectura Orientada a Servicios (SOA)





# BIBLIOGRAFIA

- Len Bass, Paul Clements (2015). Software Architecture in practice, Addison Wesley.
- Pressman, Roger(2017). Ingeniería del software. un enfoque practico, McGraw-Hill.