

TECNOLOGIA WEB

CONCEPTOS BASICOS

INTERNET

- Internet
 - Conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP.
 - Un poco de Historia
 - Nació a inicio de los 60's
 - Advanced Research Projects Agency (ARPA)
 - ARPAnet - Fines investigativos.
 - 1972 – Nace InterNetwork Group
 - 1982 – Se define TCP/IP y palabra “Internet”
 - 1983 – Primer servidor de nombre de sitios
 - 1984 – 1000 Computadoras conectadas
 - 1990 - Desaparece ARPANet
 - 1991 – Se anuncia públicamente “World Wide Web”
 -
 - 2013 – Miles de millones de computadoras conectadas

SERVICIOS Y PROTOCOLOS

- Servicios y protocolos en Internet
 - SMTP (Simple Mail Transfer Protocol): Envío de correos electrónicos.
 - FTP, TFTP, P2P, FTPS, SFTP: Transferencia de archivos
 - Mensajería Instantánea (IRC)
 - Telefonía (VoIP)
 - Acceso remoto: SSH y Telnet
 - Televisión: Internet Protocol Television (IPTV)

SERVICIOS Y PROTOCOLOS

- Servicios y protocolos en Internet (Cont.)
 - Network News Transport Protocol (NNTP)
 - DNS (Domain Name System)
 - HTTP (Hypertext Transfer Protocol)
 - HTTPS (Hypertext Transfer Protocol Secure)

World Wide Web

- WWW
 - Word Wide Web: Red informática mundial
 - Sistema de distribución de información basado en hipertexto o hipermedios enlazados y accesibles a través de Internet.
 - La Web fue creada alrededor de 1989 por el inglés Tim Berners-Lee con la ayuda del belga Robert Cailliau mientras trabajaban en el CERN en Ginebra, Suiza, y publicado en 1992.
 - Con un navegador web, un usuario visualiza sitios web compuestos de páginas web que pueden contener texto, imágenes, vídeos u otros contenidos multimedia, y navega a través de ellas usando hiperenlaces.

CARACTERISTICAS DE LA WWW

- HIPERTEXTO

- Texto que en la pantalla de un dispositivo electrónico, permite conducir a otros textos relacionados

- HIPERMEDIA

- Conjunto procedimientos para escribir, diseñar o componer contenidos que integren: texto, imagen, video, audio, mapas y otros, de tal modo que se tenga la posibilidad de interactuar con los usuarios

URI, URL, URN

- URI (Uniform Resource Identifier)
 - Cadena de caracteres corta que identifica inequívocamente un recurso
- URN (Uniform Resource Name)
 - Cadenas de caracteres que se usan para nombrar recursos en Internet para su identificación. Se usan para nombrar, no para identificar
- URL (Uniform Resource Locator)
 - Cadenas de texto que se usan para nombrar recursos en Internet para su localización
- $URI = URL + URN$

FORMATO URL

- URL (Formato)
 - esquema://usuario:contraseña@máquina:puerto/
directorio/archivo
 - Ej:
 - <http://www.uagrm.edu.bo/index.php>

ESQUEMA URL

- Un URL se clasifica por su esquema.
- El esquema de una URL generalmente identifica el protocolo que se usa para recuperar la información del recurso identificado.
- Esquemas registrados: cid, data, dav, fax, ftp, file, gopher, http, https, imap, ldap, mailto, mid, news, nfs, nntp, pop, pres, sip, sips, snmp, tel, telnet, urn, wais, xmpp.

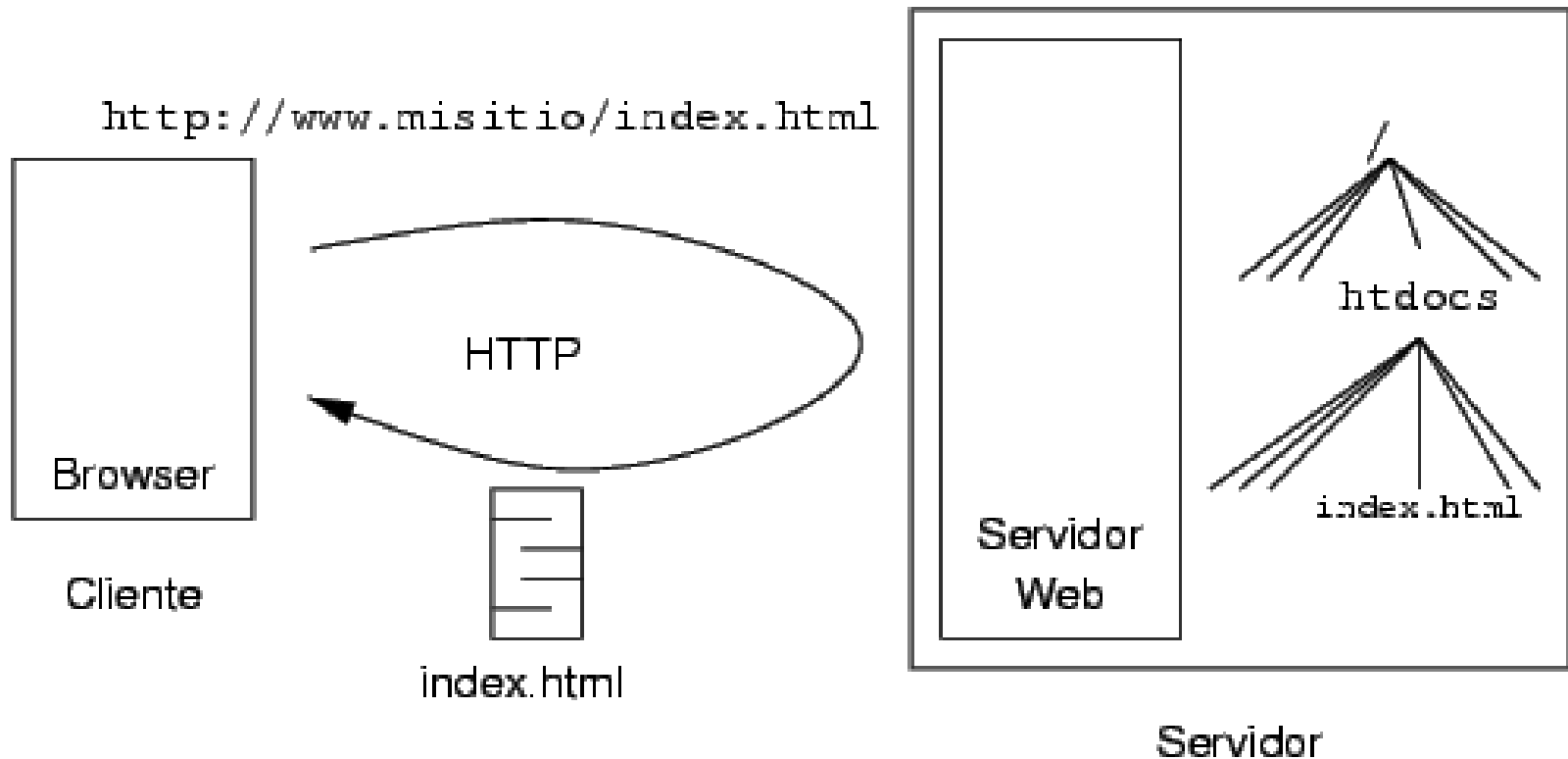
ESQUEMA URL

- Ejemplos de esquema URL:
 - http - recursos HTTP
 - https - HTTP sobre SSL
 - ftp - File Transfer Protocol
 - mailto - direcciones de correo electrónico
 - file - recursos disponibles en el sistema local, o en una red local
 - news - grupos de noticias Usenet (newsgroup) (nntp)
 - telnet - el protocolo telnet
 - data - el esquema para insertar pequeños trozos de contenido en los documentos Data: URL

Diferencias URI Y URL

- URI es un identificador más completo y por eso es recomendado su uso en lugar de la expresión URL.
- Un URI se diferencia de un URL en que permite incluir en la dirección una subdirección, determinada por el “fragmento”
- Ej:
 - <http://www.uagrm.edu.bo/index.php#inicio>

ARQUITECTURA WEB



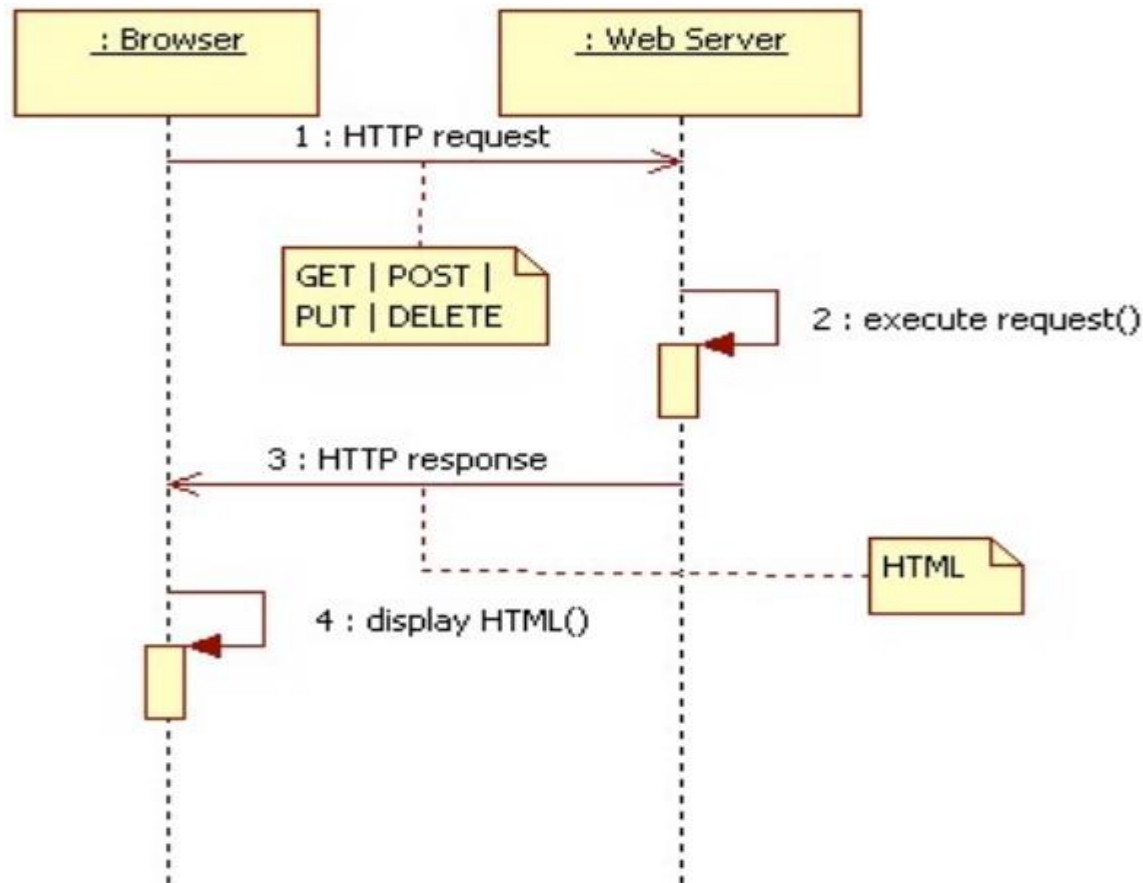
FUNCIONAMIENTO DE LA WEB

- El cliente solicita un recurso ingresando una URI por medio de alguna interfaz de usuario.
- La interfaz de usuario inicia un proceso de ubicación, solicitud y desplegado del recurso.
- Este proceso inicia con la traducción de la parte del nombre del servidor de la URL a una IP usando DNS
- Luego envía la solicitud al servidor solicitando el recurso.
- El recurso solicitado por el aplicativo cliente, es enviado por el servidor y recibido por el aplicativo cliente. Este ultimo se encarga de guardar y genera la vista del recurso recibido.

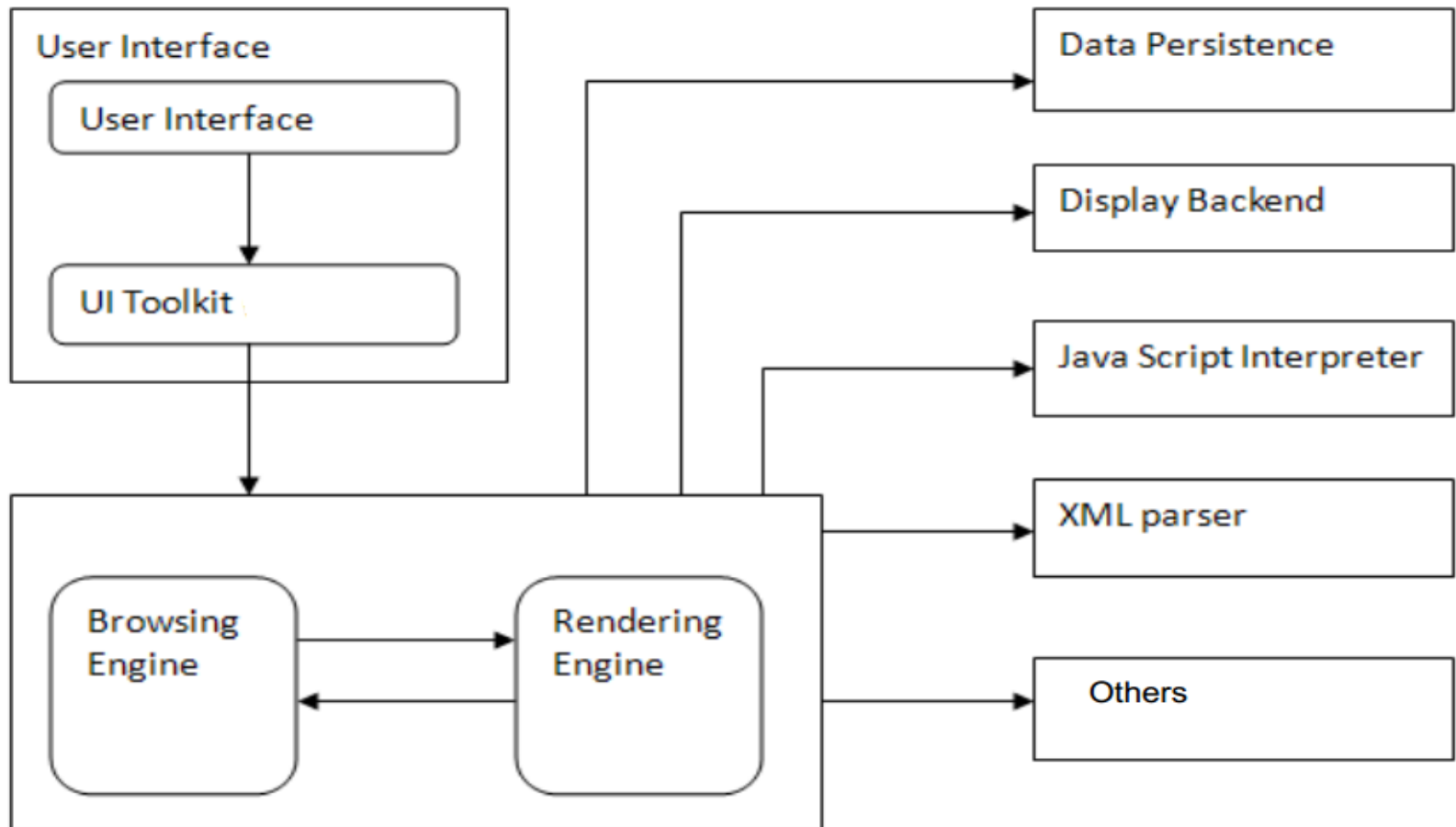
El Navegador Web, Browser

- Interfaz de usuario universal
- Es una aplicación de software que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML
- FUNCIONES:
 - Petición de las páginas Web
 - Representación adecuada de sus contenidos
 - Gestión de los posibles errores que se puedan producir

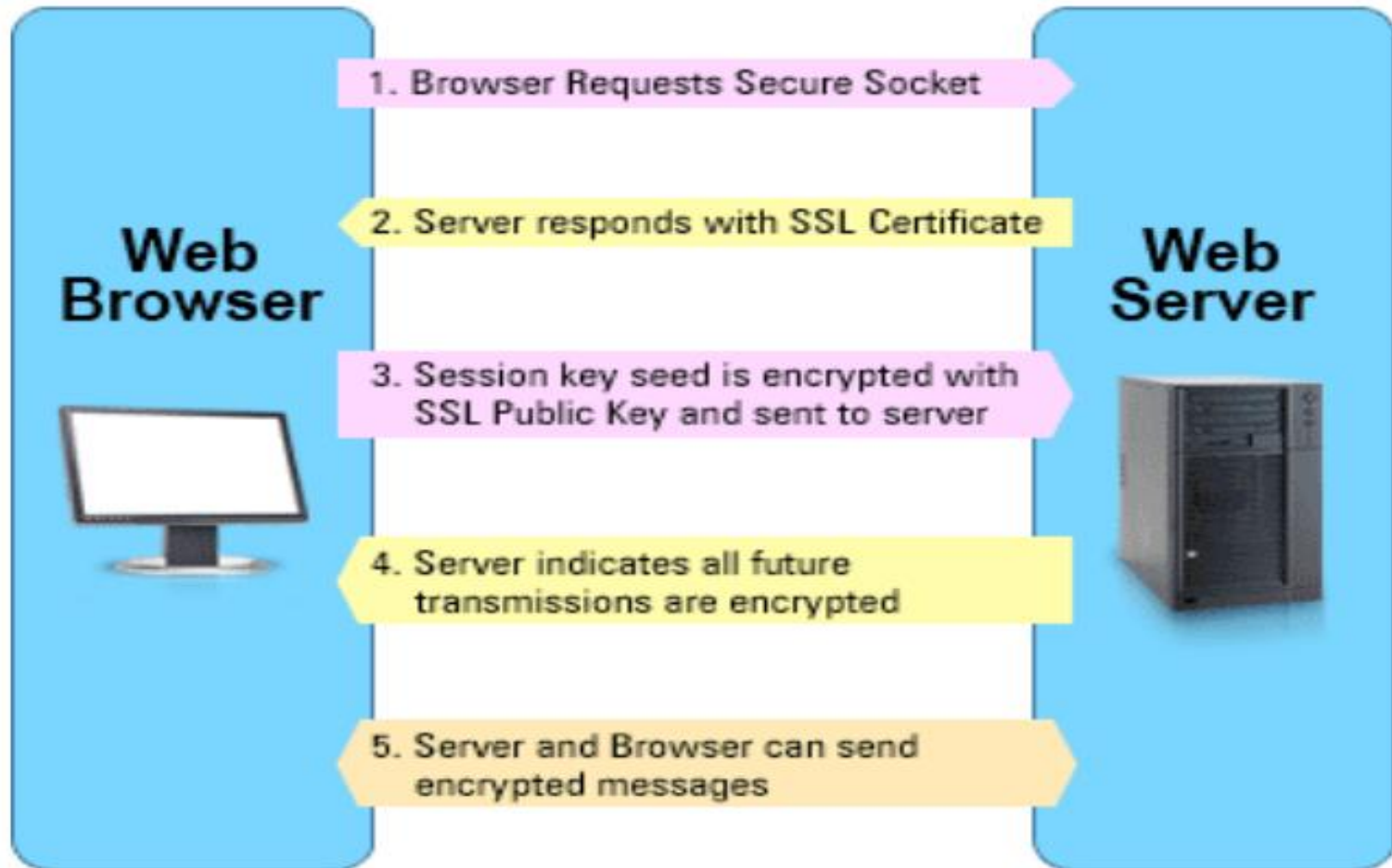
El Navegador Web. Como funciona?



El Navegador Web. Arquitectura



El Navegador Web. Soporte conexiones seguras



El Navegador Web

- POSIBILIDADES

- Ejecución de programas de tipo *script*(*Javascript*, *Jscript*, *VBScript*):

- Proporcionan las soluciones llamadas del lado del cliente (Client side)
 - Permiten realizar validaciones de datos recogidos en las páginas antes de enviarlos al servidor
 - Proporcionan un alto grado de interacción con el usuario dentro del documento

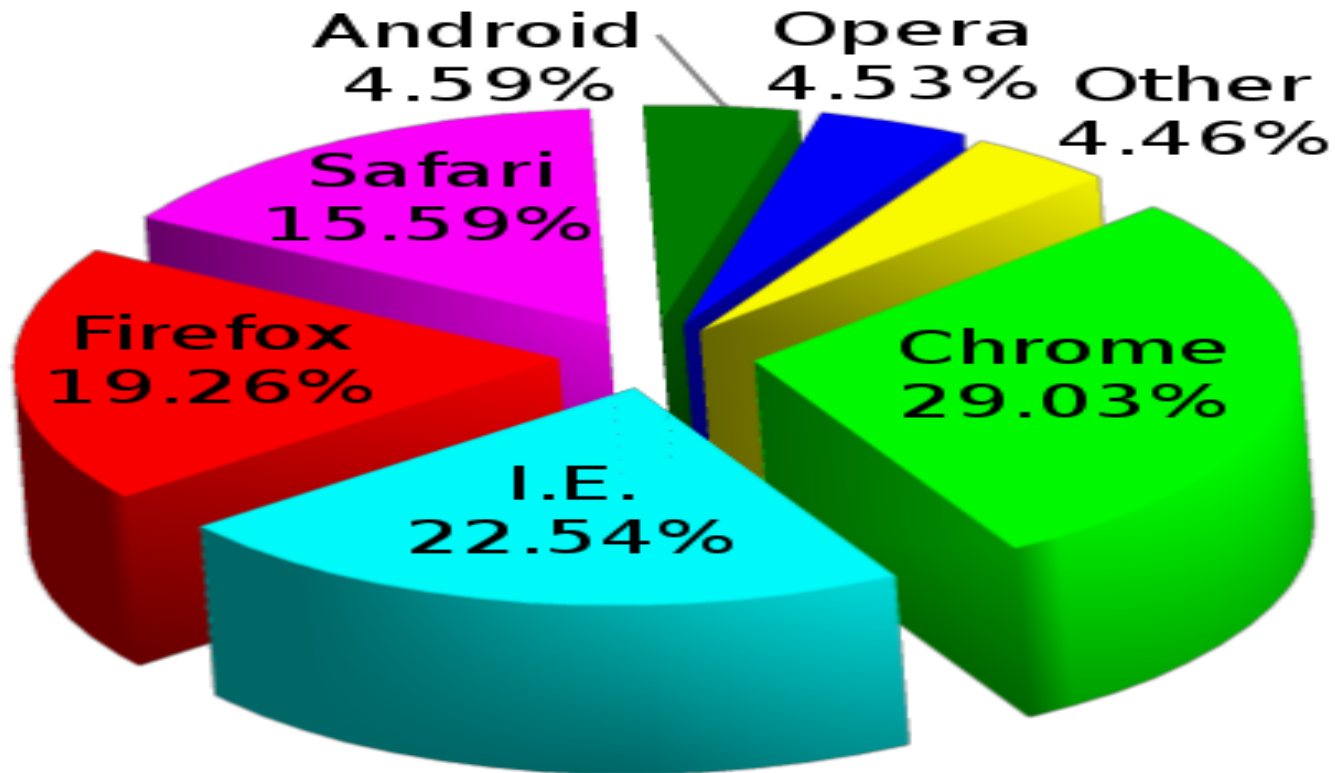
El Navegador Web

- POSIBILIDADES (Continuación)
 - Gestión del llamado HTML dinámico (*Dinamic HTML*, DHTML), el cual esta compuesto por:
 - Hojas de Estilo en Cascada(CSS)
 - Modelo de objetos (DOM)
 - Script de programación
 - Visualizacion de documentos XML (eXtensible Markup Language)
 - Ejecucion de ActiveX y Applets

El Navegador Web. Navegadores Comunes

- Internet Explorer, Mozilla Firefox, Safari, Opera, Avant Browser, Konqueror, Lynx, Google Chrome, Maxthon, Flock, Arachne, Epiphany, K-Meleon, AOL Explorer.

El Navegador Web. Guerra de Navegadores



Browser usage September 2012

Componentes de un navegador web

- Layout Engine (Rendering Engine)
 - Toma contenido marcado (HTML, XML, archivos de imágenes, etc.) e información de formato (CSS, XSL, etc.) y lo despliega en la pantalla en el área de una ventana.

Componentes de un navegador web

- Características Internas
 - Bookmark managing
 - Download managing
 - Password managing
 - Form managing
 - Spell checking
 - Search engine toolbar
 - Persite security configuration
 - Privacy mode
 - Auto-updater
 - Image Grabber

Componentes de un navegador web

- Gestor de Plug-in's
 - Componente externo que se agrega/desagrega a un navegador según la funcionalidad deseada por el usuario
 - Común para funciones como búsqueda, anotación, traducción, visualización, etc.
 - El navegador expone una API que permite integrar el plug-in desarrollado

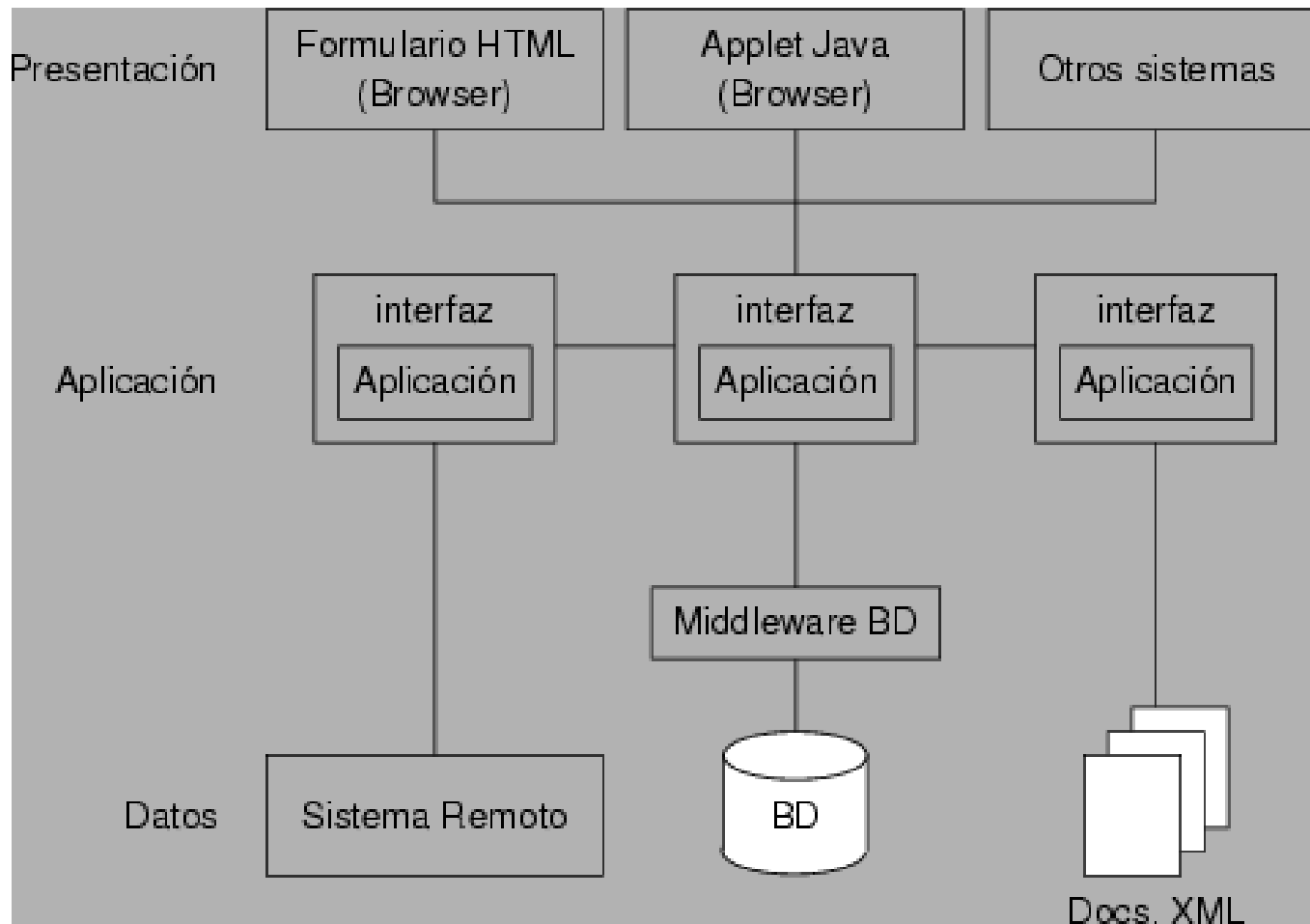
Componentes de un navegador web

- JavaScript Engine – DOM
 - Interpreta y ejecuta JavaScript
 - JavaScript
 - Lenguaje de programación interpretado. Es un lenguaje orientado a objetos, imperativo, débilmente tipificado y dinámico.
 - Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

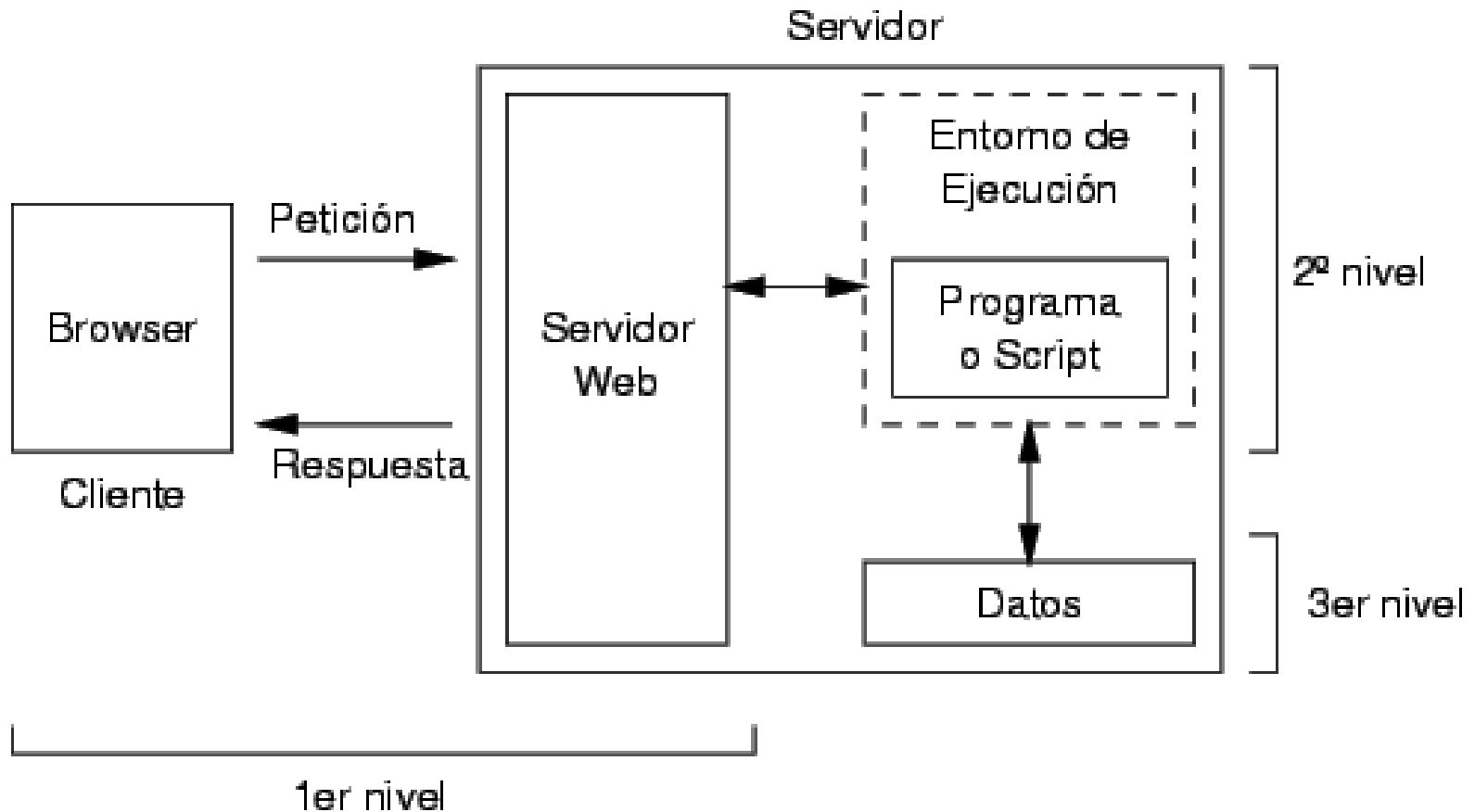
SERVIDOR WEB

- Programa que corre sobre el servidor que escucha las peticiones HTTP que le llegan y las satisface
- Dependiendo del tipo de petición, buscara una pagina o un programa.
- Devuelve un documento HTML al cliente
- Fundamental en el desarrollo de las aplicaciones del lado del servidor (Server side applications)

APLIACIONES MULTINIVEL



APLICACIONES MULTINIVEL



APLICACIÓN WEB

- Herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.
- Multicapas.
- Tienen un alto grado de interacción con el usuario, por lo cual el diseño de su interfaz debe ser claro, simple y debe estar estructurado de tal manera que sea orientativo para cada tipo de usuario.

TIPO DE APLICACIONES WEB

- Un criterio bastante practico seria clasificarlas de acuerdo a la intención de la aplicación web:
 - Informacionales
 - Orientado a la descarga de datos
 - Interactivas
 - Orientadas al servicio
 - Transaccionales
 - Flujo de datos
 - Entorno de trabajo colaborativo
 - Comunidades On-line
 - Portales Web
 - Orientados al análisis de datos

VENTAJA DE APLICACIONES WEB

- Ahorra tiempo
- No hay problemas de compatibilidad
- Actualizaciones inmediatas
- Consumo de recursos bajo
- Multiplataforma
- Portables
- Alta disponibilidad .
- Los virus no dañan los datos (Almacenados en el serv.).
- Colaboración(Compartición de datos)
- Los navegadores ofrecen cada vez más y mejores funcionalidades para crear aplicaciones web ricas

DESVENTAJA DE APLICACIONES WEB

- Ofrecen menos funcionalidades que las aplicaciones de escritorio.
- La disponibilidad depende de un tercero.

REQUISITOS PARA LAS APLICACIONES WEB

- Portabilidad
- Rapidez de implantación
- Creación de contenido (Incluirlo en la fase de ingeniería de la aplicación)
- Integración.
- Capacidad de evolucionar.
- Seguridad.
- Calidad
- Velocidad
- Interfaces intuitivas
- Capacidad de personalización

SITIO WEB

- Colección de páginas web relacionadas y comunes a un dominio de Internet o subdominio en la World Wide Web en Internet
- Son una conjunción muy compleja de distintos sistemas integrados entre sí (Bases de datos, servidores, redes, componentes de backup y seguridad, etc...)

TIPOS DE SITIOS WEB

- Personales
- Currículum Vitae
- Temáticos
- Acontecimientos
- Promoción
- Pequeño Negocio
- De Descargas....

INGREDIENTES DE UN SITIO WEB

- Páginas web
- Espacio web
- Nombre de dominio
- Herramientas
- Hipervínculos
- Extras indispensables

HERRAMIENTAS

- LENGUAJES DE PROGRAMACIÓN
- Definición:
 - Es un idioma artificial diseñado para expresar procesos que pueden ser llevadas a cabo por máquinas como las computadoras
- Utilidad
 - Se usan para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana

FAMILIAS DE LENGUAJES

- Imperativos
 - Fortran, cobol, basic, pascal, ada, C.
- Funcionales
 - Lisp, scheme, Haskell
- Orientados a Objetos
 - C++, Java, C#.
- Programación lógica
 - Prolog, OZ

Lenguajes (Cont.)

- Lenguajes de Dominio específico
 - Lenguajes cuyo objetivo es un dominio de problema concreto. Ej. AutoCAD, Word, Excel, etc.
- Maquinas abstractas
 - Procedimiento que permite ejecutar un conjunto de instrucciones en un lenguaje formal

Lenguajes para Internet

- De propósito general: C#, Python, Perl
- De computacion en el cliente: Etiquetas de HTML, Javascript, Jscript, VBScript, HaskelScript. HTML Dinamico (Dependencia de posibilidades con el aplicativo cliente)
- De Computación en el servidor: CGI, php. Lenguajes de proposito general empotrados en HTML (jsp (java), asp (Basic))

Lenguajes de marcas

- SGML (Standar Generalized Markup Language):
 - Usado para procesamiento de texto
- HTML:
 - Predeterminado para elaboracion de paginas web
- XML (eXtensible Markup Language):
 - Permite definir la gramática de lenguajes específicos.
 - Estandar de de intercambio de informacion a traves de la web

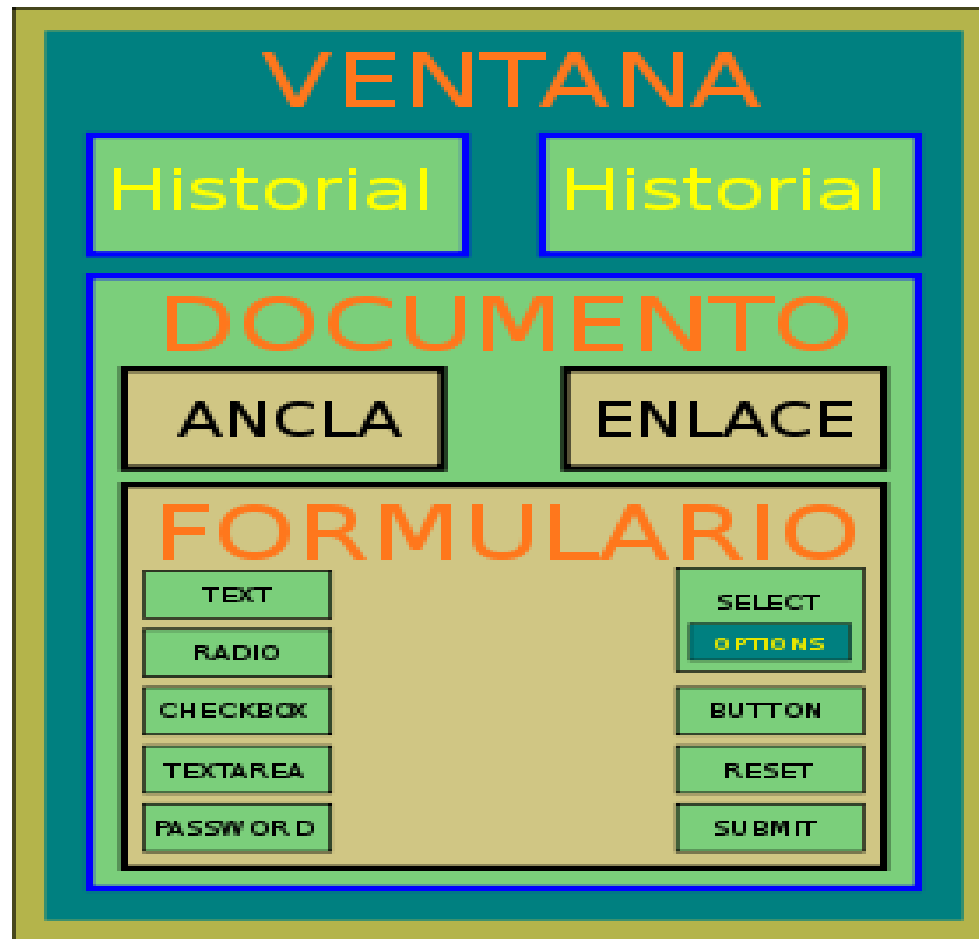
Lenguajes de marcas

- HTML Dinamico o DHTML
 - Técnicas que permiten crear sitios web interactivos utilizando una combinación de:
 - Lenguaje HTML estático,
 - Un lenguaje interpretado en el lado del cliente (como JavaScript),
 - El lenguaje de hojas de estilo en cascada (CSS) y
 - La jerarquía de objetos de un Document Object Model (DOM)
 - HTML Dinámico es cualquier página web en la que los scripts en el lado del cliente cambian el HTML del documento

DOM

- Proporciona un conjunto estándar de objetos para representar documentos HTML y XML.
- A través del DOM, los programas pueden acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML
- Problemas de compatibilidad de navegadores.

ESTRUCTURA DOM



FRAMEWORKS

- Es una estructura conceptual y tecnológica de soporte definido.
- Contiene artefactos o módulos de software concretos, con base a la cual otro proyecto de software puede ser más fácilmente organizado y desarrollado.
- Puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

FRAMEWORKS

- php: Zend, Yii, Symfony, QPHP
- java: Struts 2, JSF, Spring MVC, Wicket
- Haskell: SNAP, Yesod
- Groovy: Grails, Groovestry, GroovyFlow, GroovyWorks
- Perl: Catalyst, Jifty
- Python: Django, Pylons, web2py

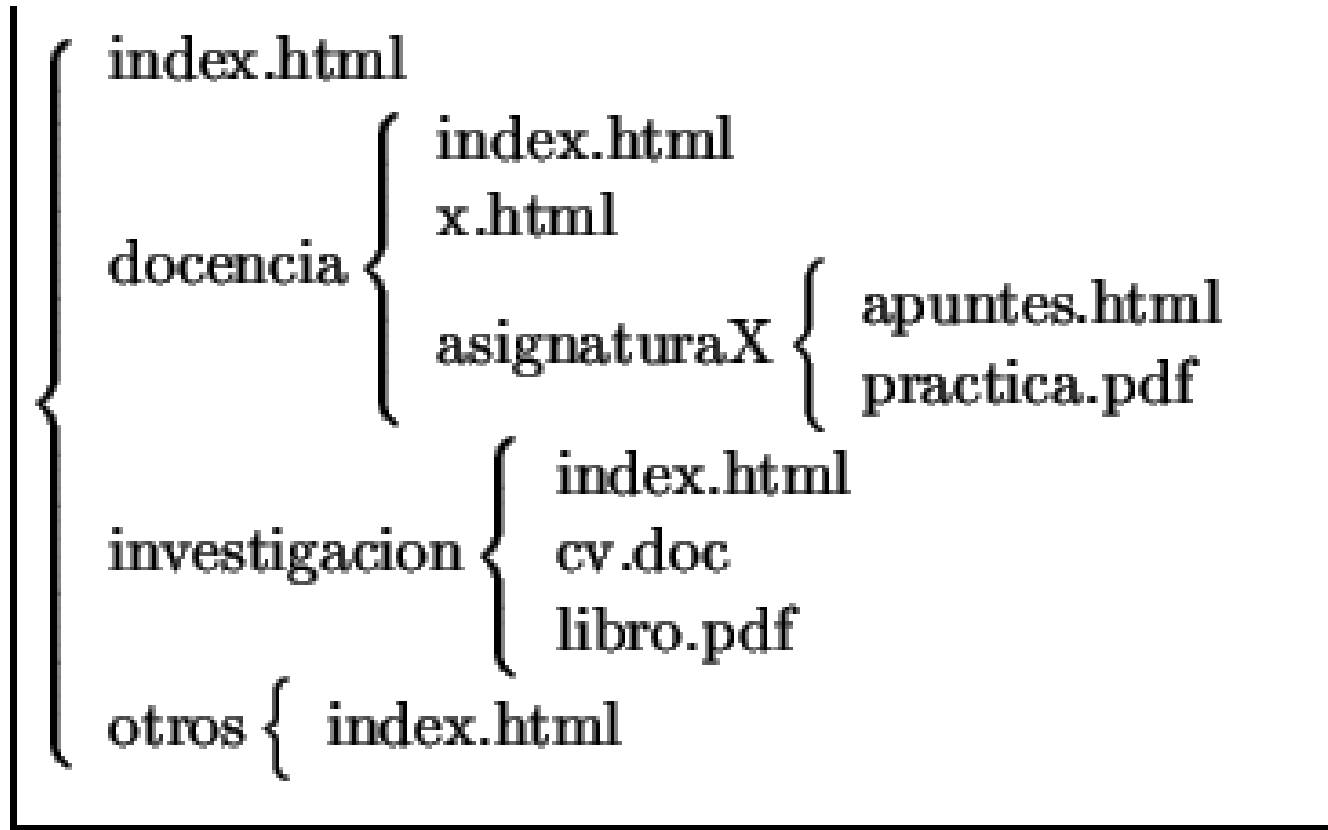
MVC

- Modelo Vista Controlador
- Patrón de desarrollo de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos

MVC

- Modelo
 - Es la representación específica de la información con la cual el sistema opera.
- Vista
 - Presenta el modelo en un formato adecuado para interactuar
 - Usualmente la interfaz de usuario
- Controlador
 - Responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente, a la vista

ESTRUCTURA ARCHIVOS Y DIRECTORIOS



ESTRUCTURA DE UN SITIO WEB

- **Estructura de árbol o jerárquica**
 - A partir de una página de bienvenida o portal (raíz) se abren unas secciones (ramas) que a su vez contienen múltiples páginas web (hojas)
- **Estructura lineal**
 - A partir de una página de bienvenida o portal se suceden las siguientes páginas una tras otra como si se tratara de un libro.

ESTRUCTURA DE UN SITIO WEB

- **Estructura en red**

- Las páginas que forman el sitio web se enlazan unas con otras según sus contenidos en una especie de red en la que no se aprecia ningún tipo de jerarquía

METODOLOGIAS DE DESARROLLO WEB

- Problemas actuales en el desarrollo web:
 - Los principales problemas que nos encontramos es la falta de fiabilidad, seguridad, escalabilidad, mantenimiento, integración y la alta dependencia para su desarrollo e implantación junto con la falta de estándares.

METODOLOGIAS DE DESARROLLO WEB

- ¿Por qué necesitamos una metodología?
 - Para controlar el caos que han provocado en el pasado procesos creativos de desarrollo con el fin de proporcionar un proceso sistemático orientado a la mejora de la calidad de la aplicación final.
 - Para contar con un mecanismo que nos permita estructurar, comunicar, entender, simplificar y formalizar tanto el dominio como las decisiones de diseño, así como disponer de documentación detallada para posibles cambios del software

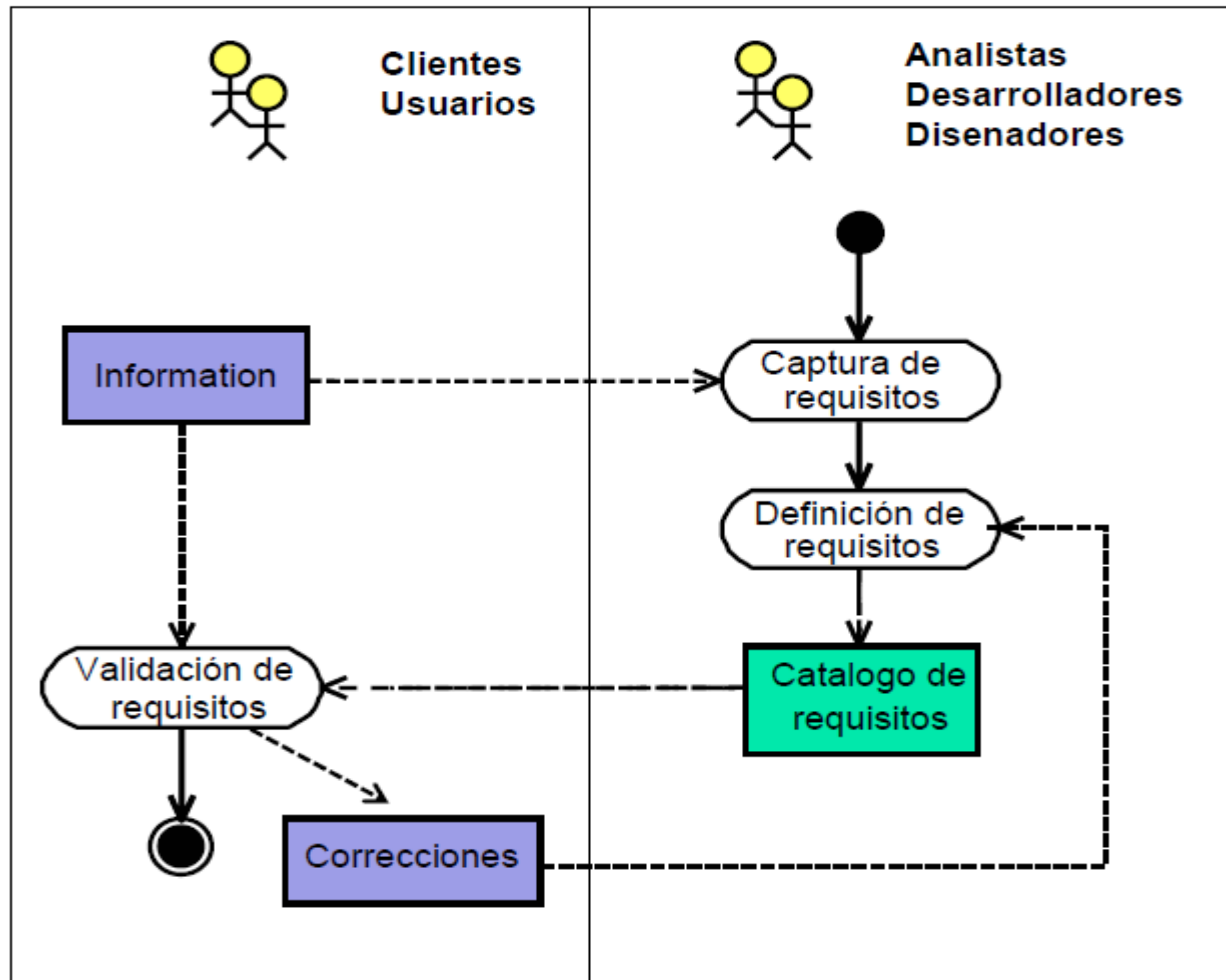
EVOLUCION DE LAS METODOLOGIAS WEB

- Las distintas metodologías se pueden dividir en tres generaciones en base a su sofisticación, estas son:
 - Primera Generación:(Principios de los 90) Se sientan las bases de la ingeniería Web, en los que se incluyen conceptos como construcción de navegación, separación entre estructuras y el contenido durante el ciclo de desarrollo.
 - Segunda Generación: (Segunda mitad de los 90) Se refinan los primeros modelos y se añaden los soportes de funcionalidad básica y se llevan a cabo los primeros esbozos de proceso donde se delimitan los modelos conceptual, lógico y físico.
 - Tercera generación: (A partir del 2000): Se lleva a cabo la profundización en el soporte para la funcionalidad, enfatización de la figura del usuario en los métodos, y se avanza hacia la estandarización de notaciones, procesos y lenguajes de especificación.

LA INGENIERIA DE REQUISITOS

- Proceso mediante el cual se especifican y validan los servicios que debe proporcionar el sistema así como las restricciones sobre las que se deberá operar.
- Consiste en un proceso iterativo y cooperativo de análisis del problema, documentando los resultados en una variedad de formatos y probando la exactitud del conocimiento adquirido.

LA INGENIERIA DE REQUISITOS



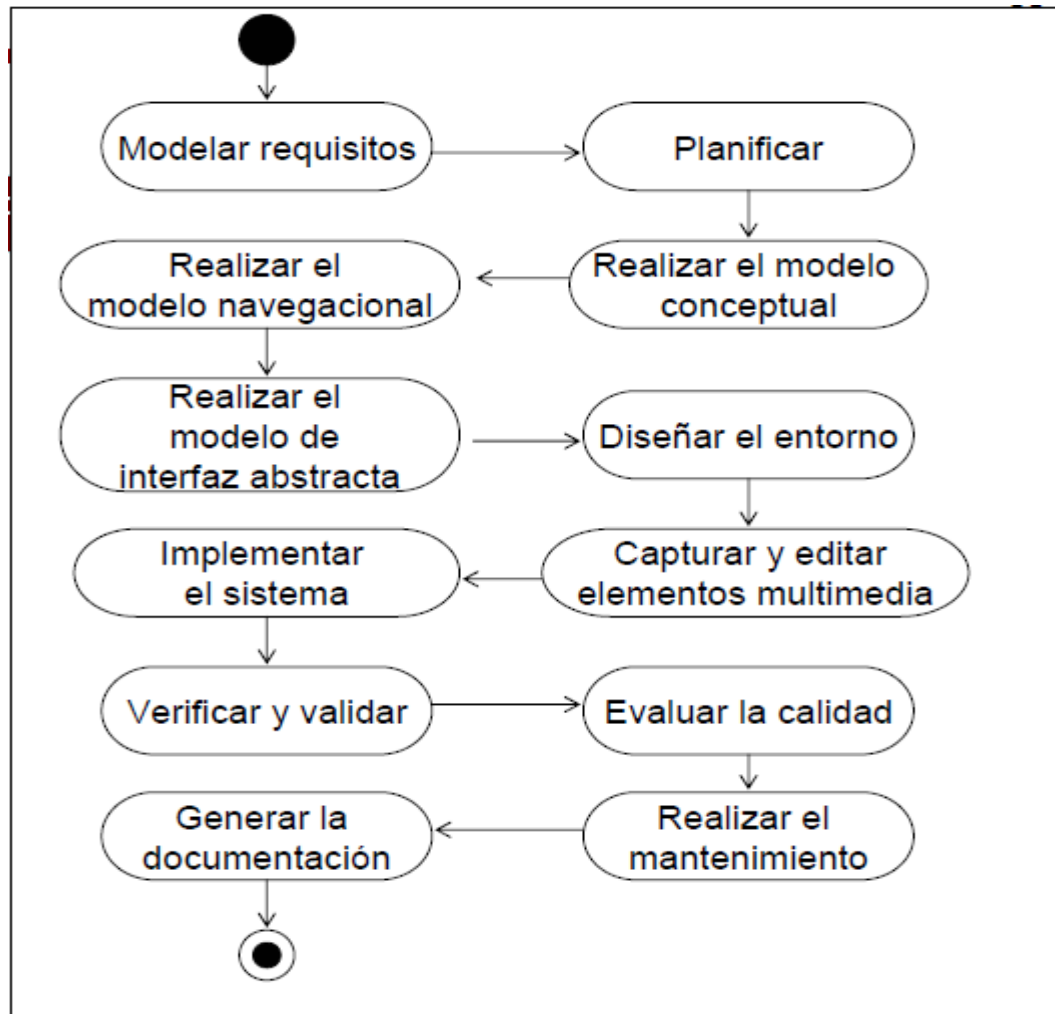
METODOLOGIAS Y LA INGENIERIA DE REQUISITOS

- Muchas metodologías se han centrado en las etapas del diseño e implementación y han dado poca importancia al tratamiento de los requisitos.
- Es aconsejable utilizar alguna metodología que de importancia a la ingeniería de requisitos.

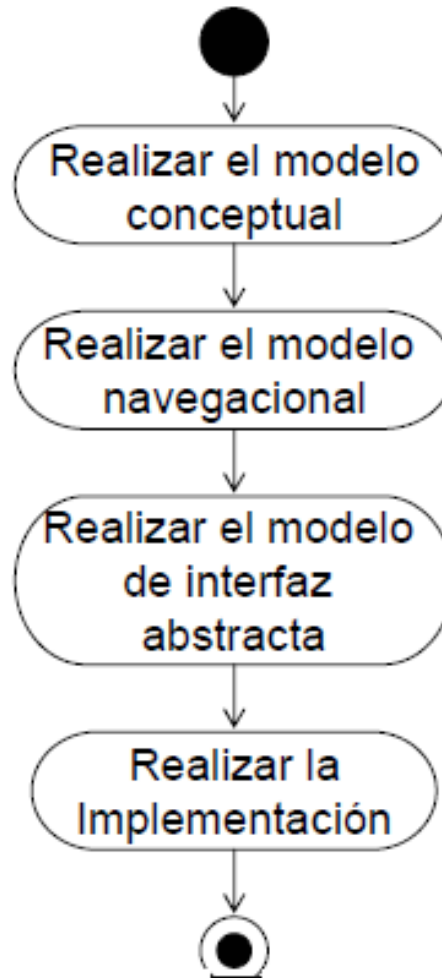
METODOLOGIAS

- Hypermedia Flexible Process Modeling (HFPM)
- Object Oriented Hypermedia Design Model (OOHDM)
- UML-Based Web Engineering (UWE)
- Navigational Development Techniques (NDT)
- Proceso Unificado de Desarrollo (PUD)

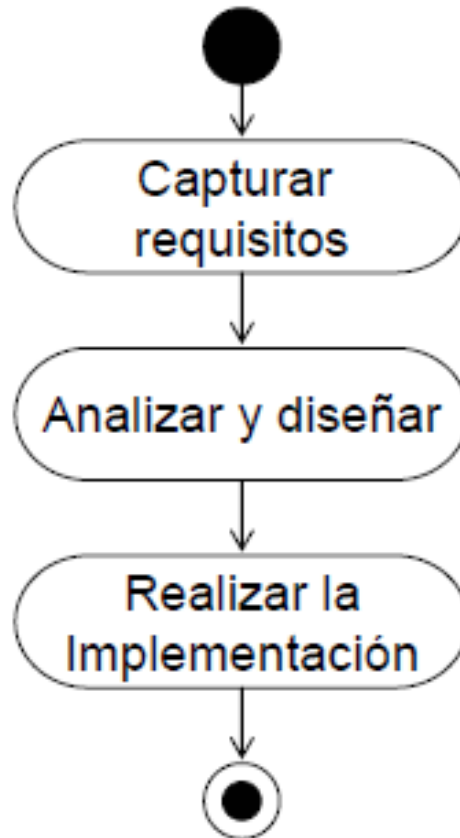
HFPM



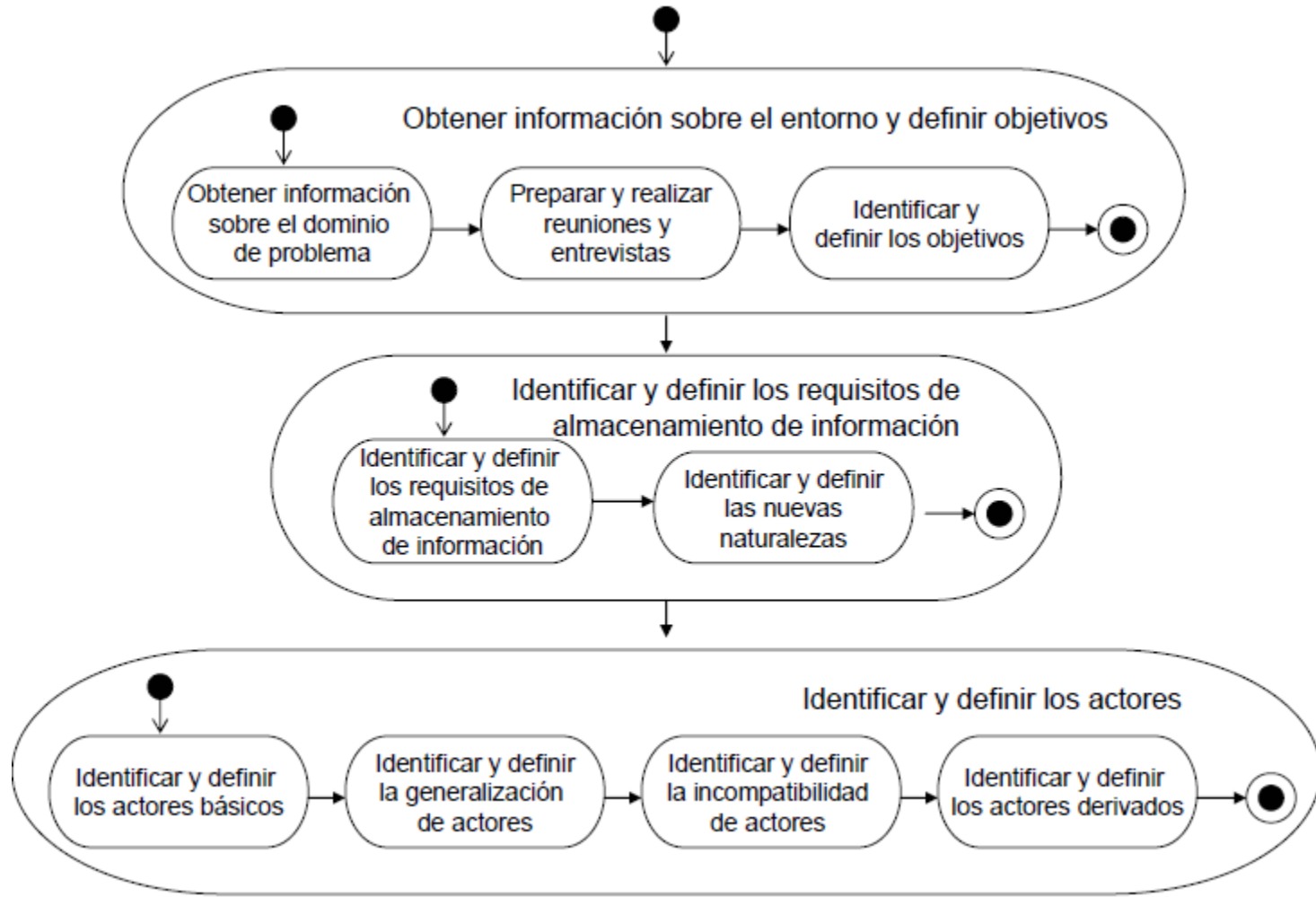
OOHDM



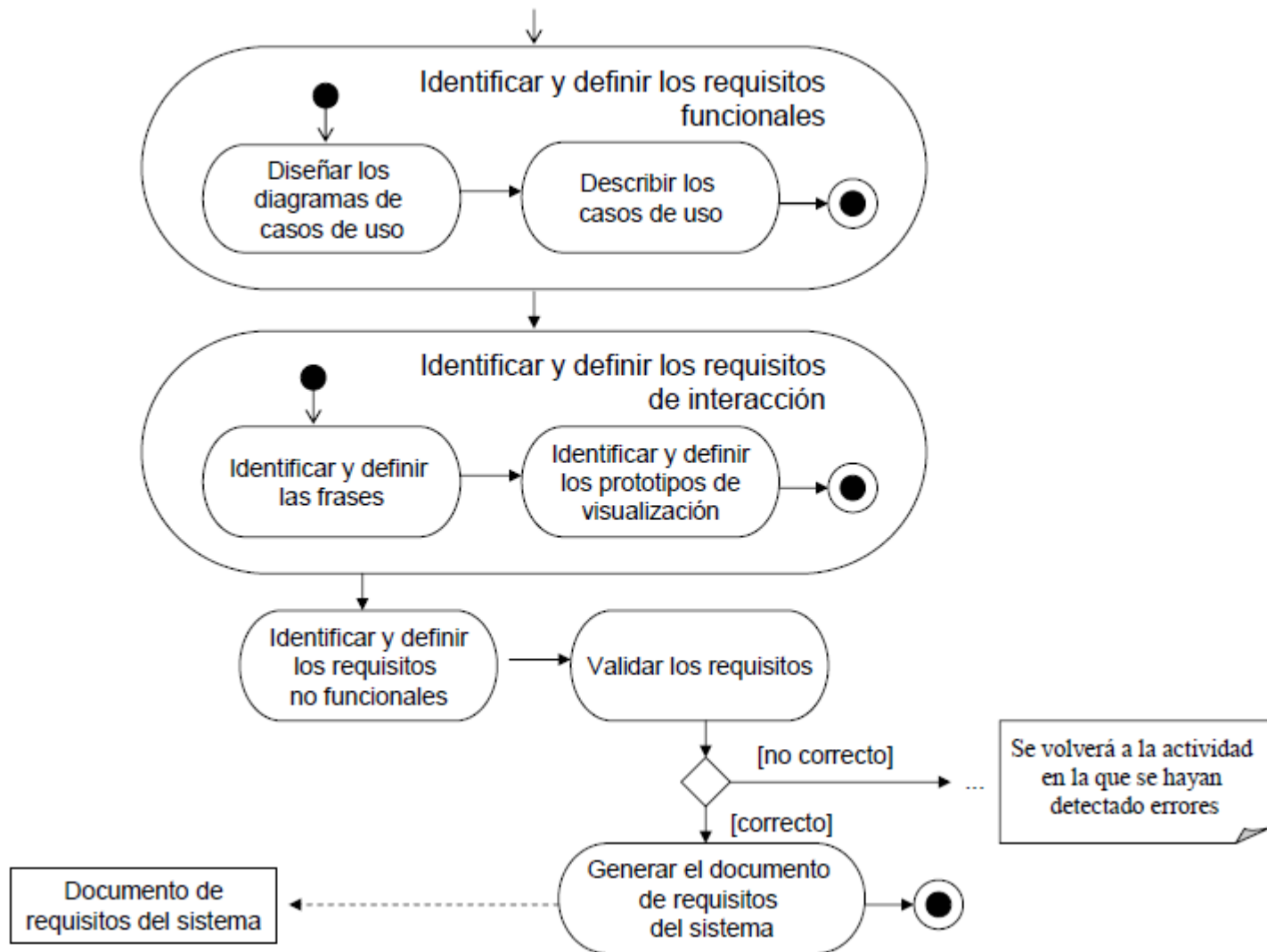
UWE



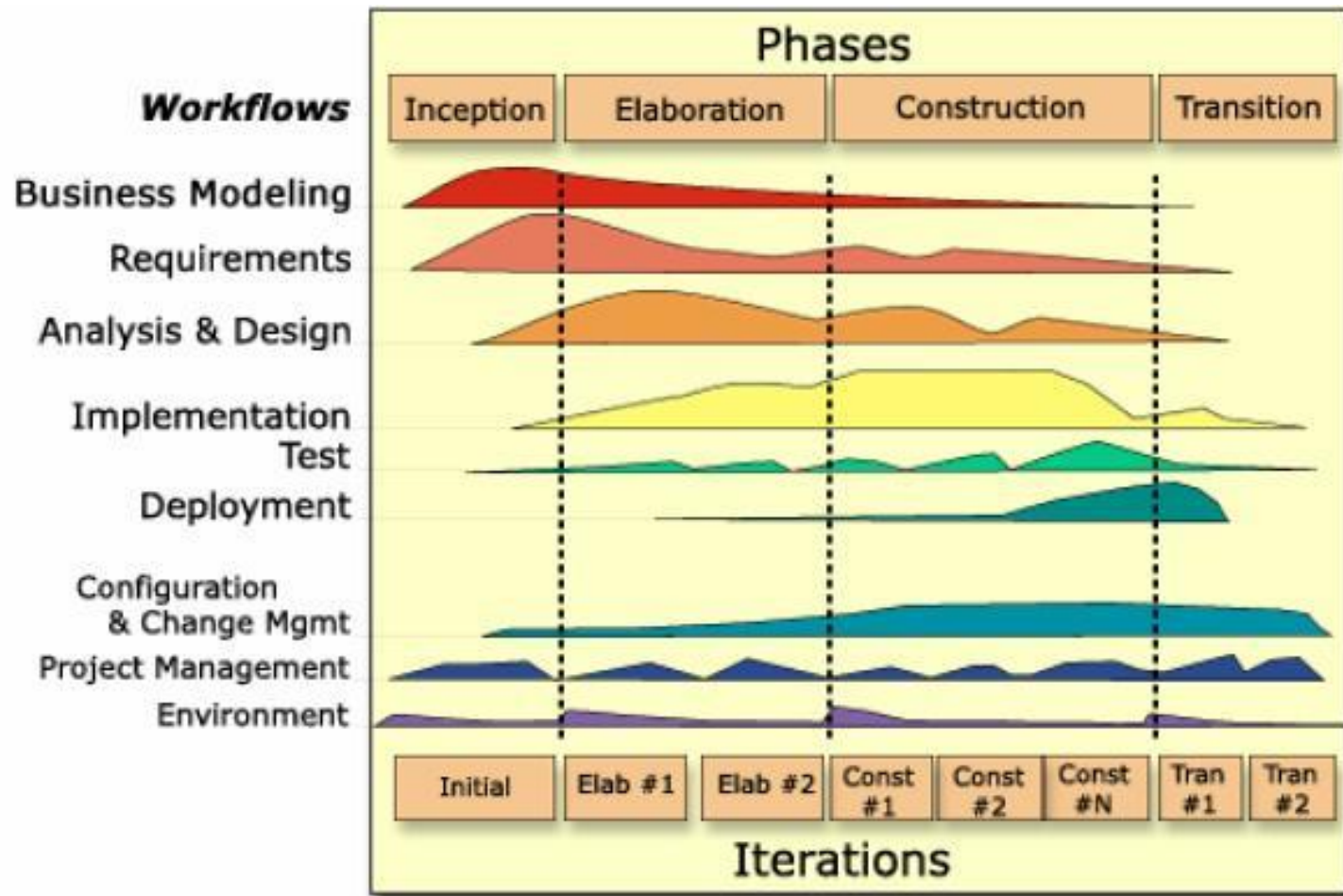
NDT



NDT (Continuación)



PUD



PUD

- HITOS

Fases	Hitos
Conceptualización - Inicio	Objetivos (visión)
Elaboración	Arquitectura
Construcción	Funcionalidad operativa
Transición	Release del sistema

- Dirigido por casos de uso, centrado en la arquitectura, Iterativo e Incremental.
- Como muchas usa UML.