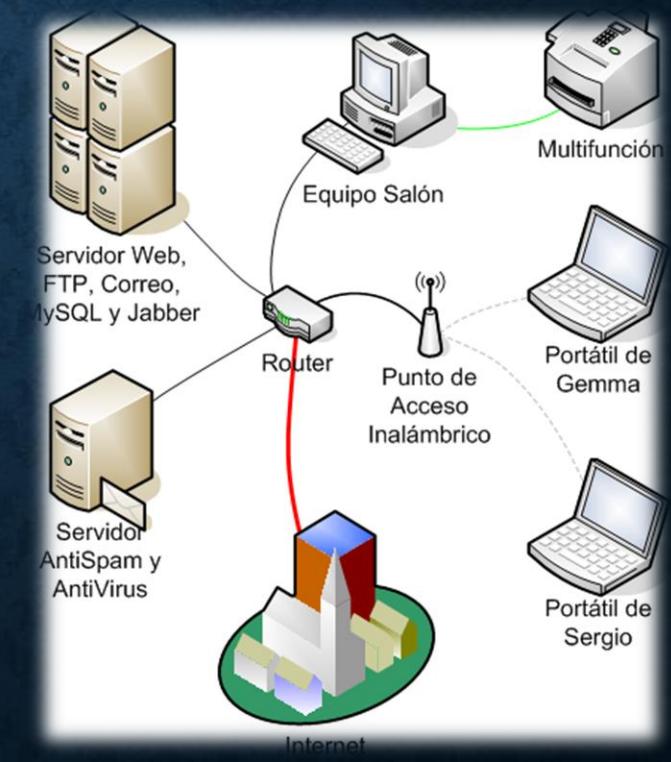


ESCUELA DE FORMACIÓN DE TECNÓLOGOS

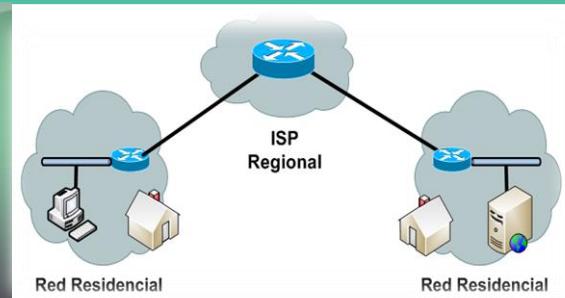


APLICACIONES DISTRIBUIDAS

ING. BYRON LOARTE



INTRODUCCIÓN

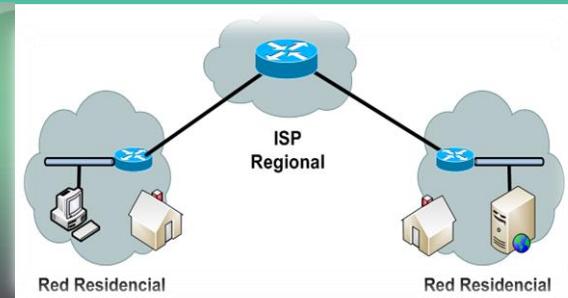


DATOS PERSONALES

- ❖ BYRON LOARTE
- ❖ 0995644186
- ❖ byron.loarteb@epn.edu.ec
- ❖ Oficina 18



INTRODUCCIÓN



PREGUNTAS DE REFUERZO

❖ ¿ Cuáles son los tipos de sistemas que existen actualmente ?

SISTEMA DE LOTES

SISTEMAS CENTRALIZADOS DE TIEMPO COMPARTIDO

SISTEMAS DE TELEPROCESO

SISTEMAS PERSONALES

SISTEMAS EN RED

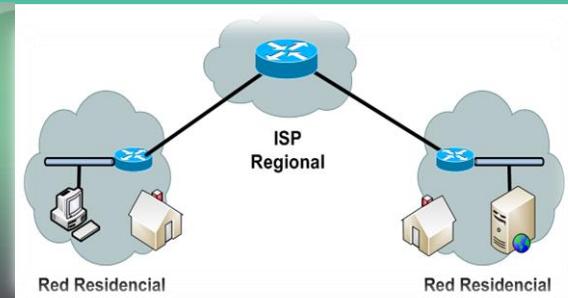
SISTEMAS DISTRIBUIDOS

❖ ¿ Qué es un sistema distribuido ?

Es una colección de computadoras independientes que aparecen ante los usuarios del sistema como una única computadora

Un **SERVICIO** único a los usuario

INTRODUCCIÓN



PREGUNTAS DE REFUERZO

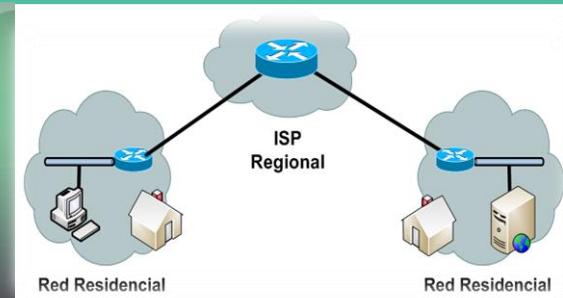
- ❖ ¿ Qué es un aplicación distribuida ?

Es una aplicación con distintos componentes que se ejecutan en entornos separados , normalmente en diferentes plataformas conectadas a través de una red

- ❖ ¿ Cuál es la función de cada capa ?



INTRODUCCIÓN

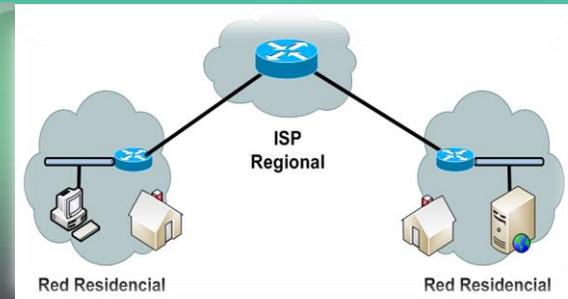


OBJETIVOS DE LA CLASE

- ❖ Determinar claramente la definición de componentes
- ❖ Determinar claramente los diferentes tipos de aplicaciones distribuidas
- ❖ Determinar claramente los patrones de diseño
- ❖ Determinar claramente las temáticas para el desarrollo de aplicaciones distribuidas



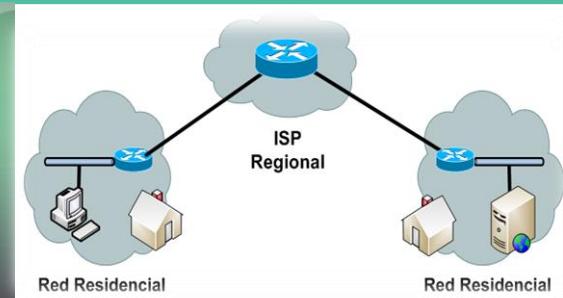
INTRODUCCIÓN



RESUMEN



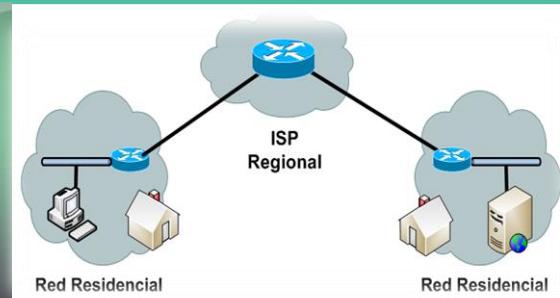
INTRODUCCIÓN



NECESIDADES DE LA APLICACIÓN DISTRIBUIDA

- ❖ Trabajo a distancia
- ❖ Compartir información
- ❖ Accesibilidad
- ❖ Seguridad en la protección de la información (tener la base de datos particionada en dos o mas nodos)
- ❖ Independencia lugares

INTRODUCCIÓN

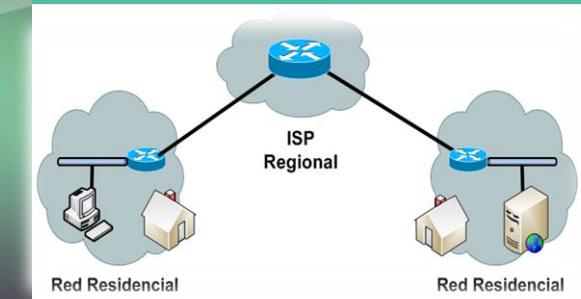


COMPONENTE

- ❖ Un componente es un elemento de software que encapsula una serie de funcionalidades.
- ❖ Un componente es una unidad independiente , que puede ser utilizado en conjunto con otros componentes para formar un sistema más complejo.

Cada componente de un sistema puede verse como un paquete o módulo

INTRODUCCIÓN



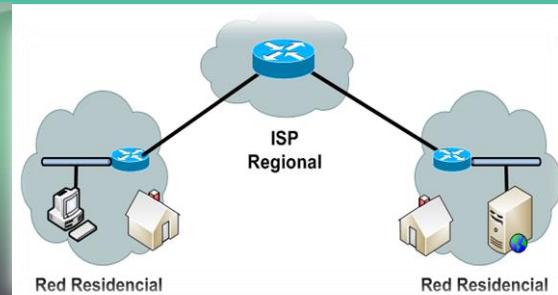
COMPONENTE



Debes conocer

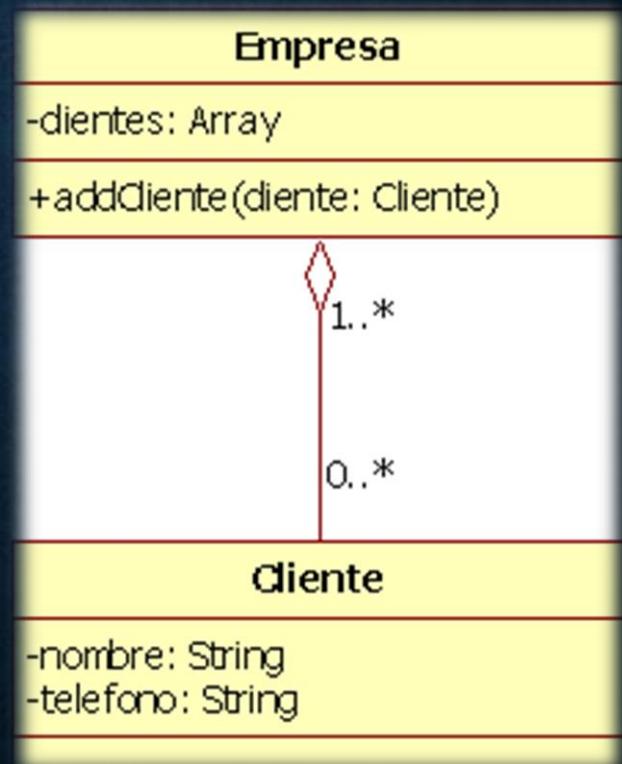
Cada componente de un sistema puede verse como un paquete o módulo

INTRODUCCIÓN

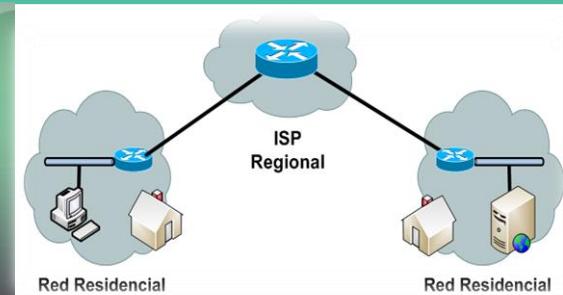


COMPONENTE

- ❖ Un componente está compuesto por elementos que pueden ser **CLASES** y/o **RECURSOS COMPLEMENTARIOS**.
- ❖ Y a su vez esas clases y recursos pueden estar agrupados dentro del componente en subpaquetes , de acuerdo a su naturaleza o necesidades de negocio.

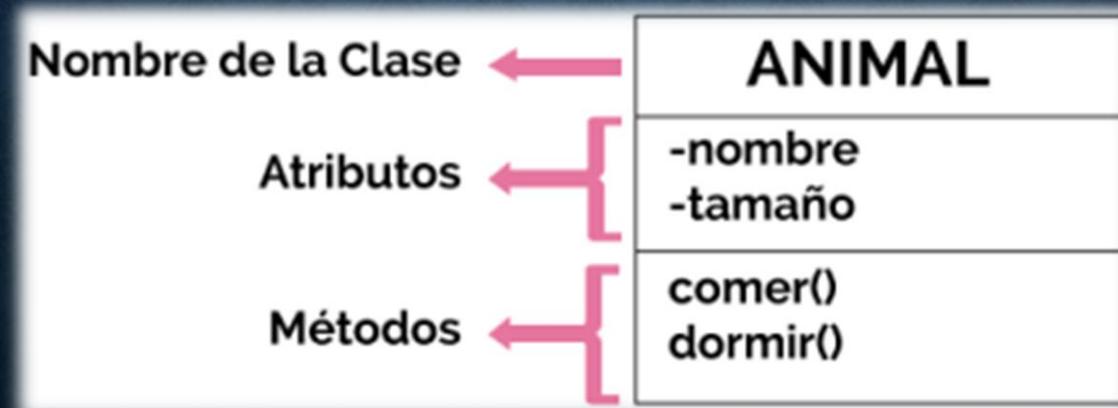


INTRODUCCIÓN

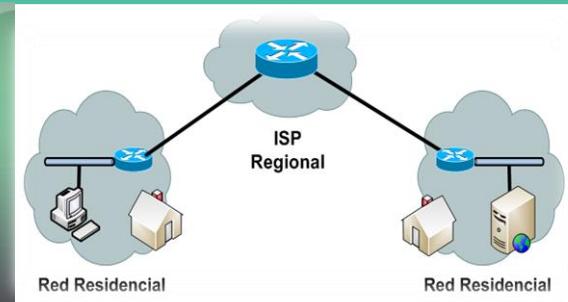


COMPONENTE

- ❖ Hablar de **CLASES** en este punto, hace referencia sobre programación orientada a objetos , ya que la construcción de componentes esta basada es los mismos principios y/o características con las que cumple este conocido **PARADIGMA**.



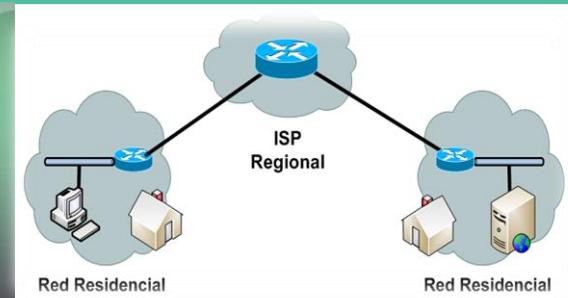
INTRODUCCIÓN



¿Estas listo para desarrollar tu
primer componente?



INTRODUCCIÓN

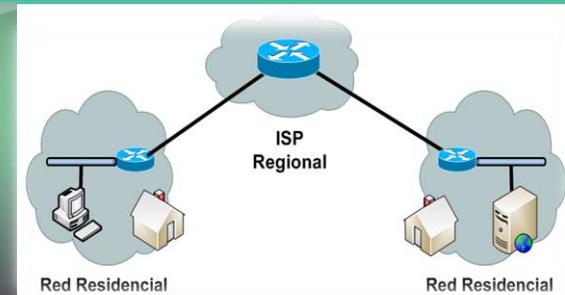


APLICACIONES DISTRIBUIDAS ... IMPORTANTE -

- ❖ Una aplicación puede estar distribuida físicamente, pero por malas prácticas de programación hagan que su presentación, reglas del negocio y acceso a datos, no esté lógicamente distribuidas de forma correcta.



INTRODUCCIÓN

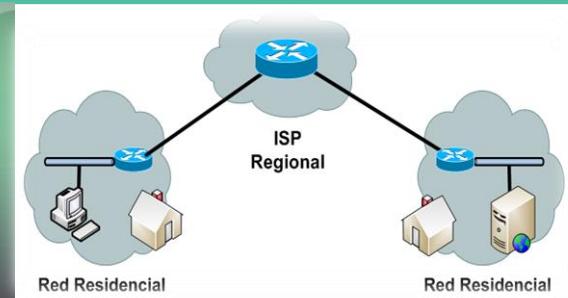


APLICACIONES DISTRIBUIDAS ... IMPORTANTE -

- ❖ Los problemas que conllevan la mala separación de la lógica de programación , no son inmediatos, sus efectos se verán muy seguramente en el transcurso de la evolución de las aplicaciones.



INTRODUCCIÓN

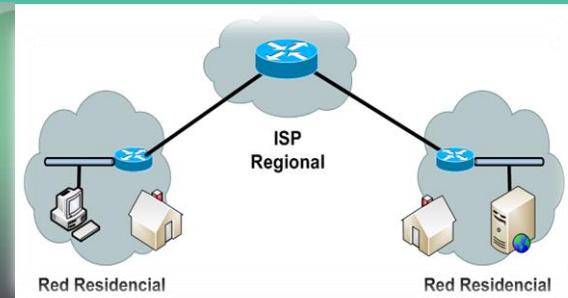


PREGUNTA

- ❖ ¿ Como podrías solucionar estos problemas ?



INTRODUCCIÓN

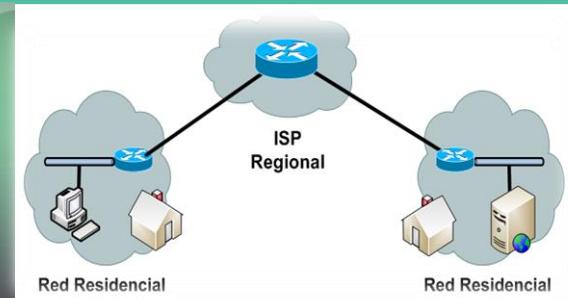


APLICACIONES DISTRIBUIDAS ... PATRON DE DISEÑO ...

- ❖ Los patrones de diseño (design patterns) son la base para la **BÚSQUEDA DE SOLUCIONES A PROBLEMAS** comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.
- ❖ Un patrón de diseño es una solución a un problema de diseño.



INTRODUCCIÓN



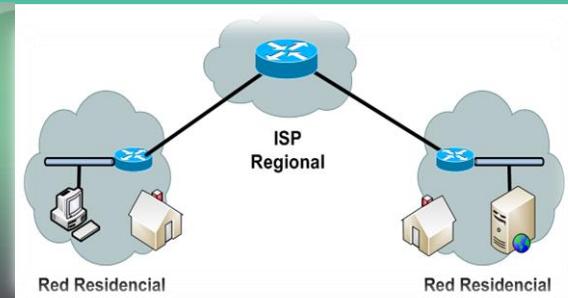
APLICACIONES DISTRIBUIDAS ... PATRON DE DISEÑO ...

- ❖ Para que una solución sea considerada un patrón debe poseer ciertas características.

EFFECTIVO

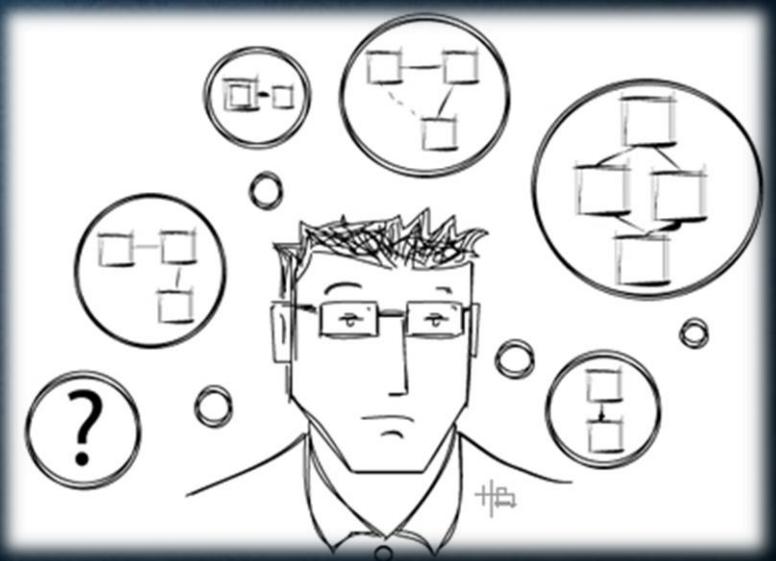
REUTILIZABLE

INTRODUCCIÓN

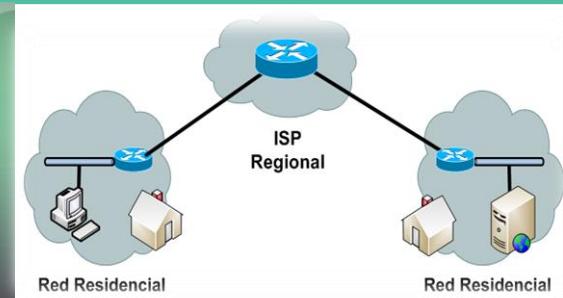


APLICACIONES DISTRIBUIDAS ... PATRON DE DISEÑO-

- ❖ Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores.
- ❖ Aplicable a diferentes problemas de diseño en distintas circunstancias.



INTRODUCCIÓN

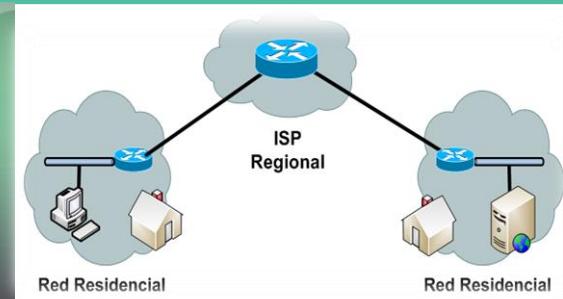


APLICACIONES DISTRIBUIDAS ... PATRON DE DISEÑO ...

Los patrones pretenden:

- ❖ Proporcionar catálogos de elementos reusables en el diseño de sistemas software.
- ❖ Evitar la reiteración en la búsqueda de soluciones a problemas ya conocidos y solucionados anteriormente.

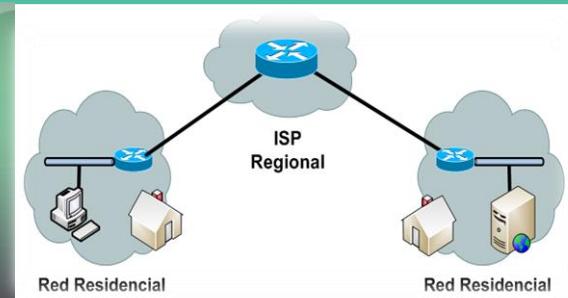
INTRODUCCIÓN



APLICACIONES DISTRIBUIDAS ... PATRON DE DISEÑO ...

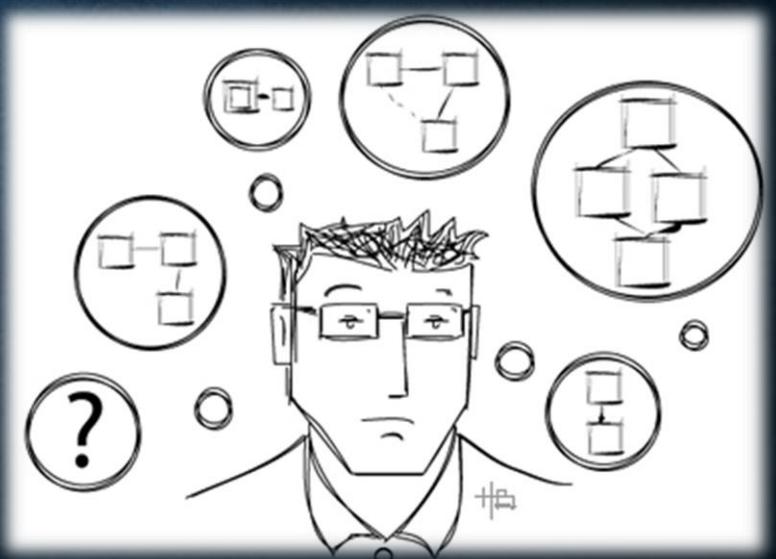
- ❖ Formalizar un vocabulario común entre diseñadores.
- ❖ Estandarizar el modo en que se realiza el diseño.
- ❖ Facilitar el aprendizaje de las nuevas generaciones de diseñadores condensando conocimiento existente.

INTRODUCCIÓN

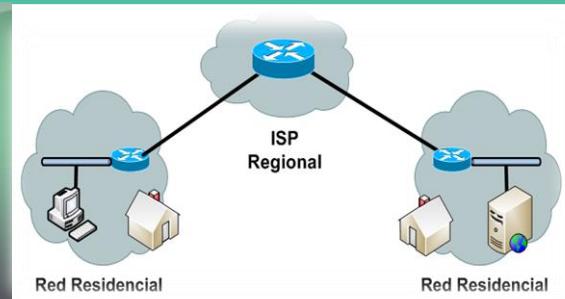


APLICACIONES DISTRIBUIDAS ... PATRON DE DISEÑO-

- ❖ Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores.
- ❖ Aplicable a diferentes problemas de diseño en distintas circunstancias.

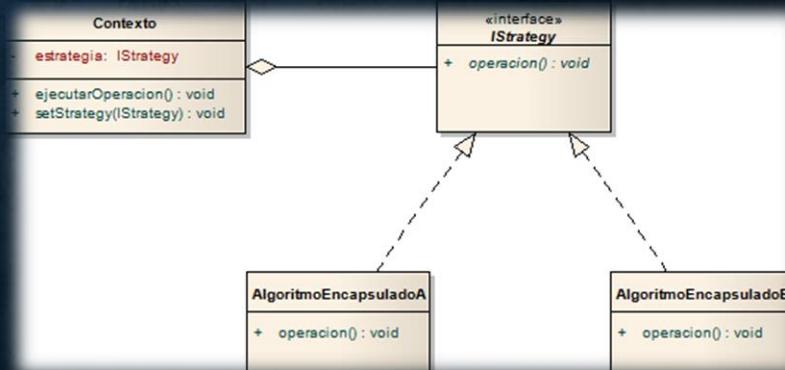
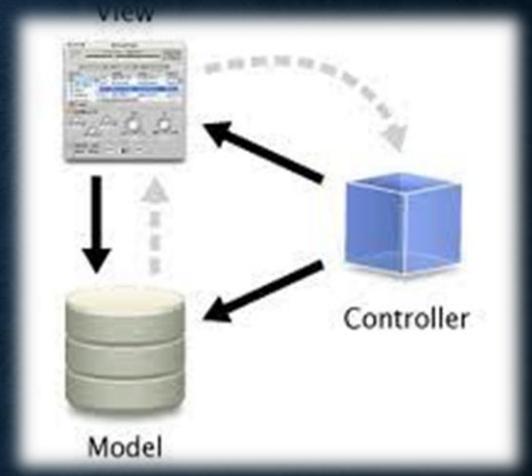


INTRODUCCIÓN

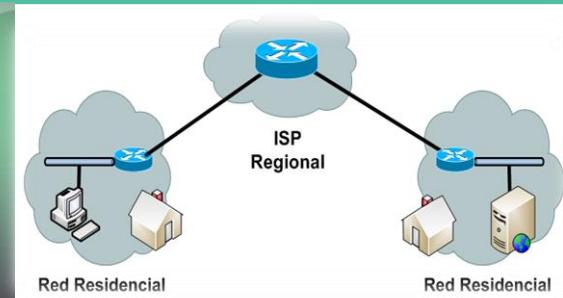


APLICACIONES DISTRIBUIDAS ... TIPOS ...

- ❖ **PATRONES DE ARQUITECTURA:** Aquellos que expresan un esquema organizativo estructural fundamental para sistemas de software.
- ❖ **PATRONES DE DISEÑO:** Aquellos que expresan esquemas para definir estructuras de diseño (o sus relaciones) con las que construir sistemas de software.



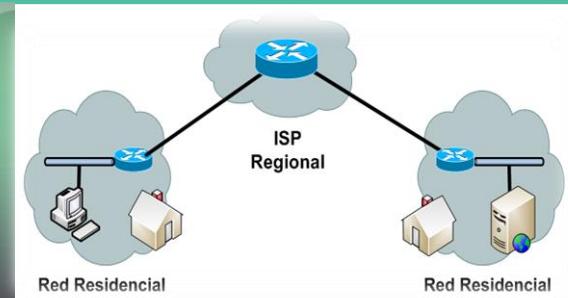
INTRODUCCIÓN



APLICACIONES DISTRIBUIDAS ... MVC ...

- ❖ El modelo–vista–controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de la interfaz de usuario.
- ❖ Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario..

INTRODUCCIÓN

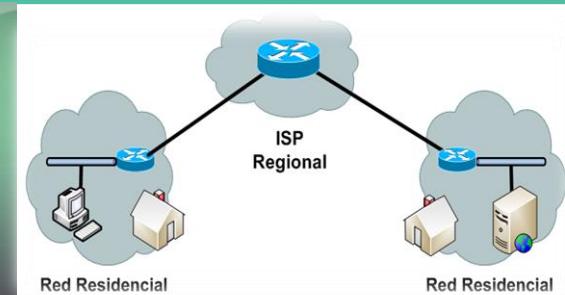


APLICACIONES DISTRIBUIDAS ... TIPOS –

- ❖ Aplicaciones Monolíticas
- ❖ Aplicaciones Cliente/Servidor
- ❖ Aplicaciones de 3 Capas
- ❖ Aplicaciones de N Capas

**VEAMOS UNAS
DEFINICIONES
BÁSICA SOBRE ESTAS**

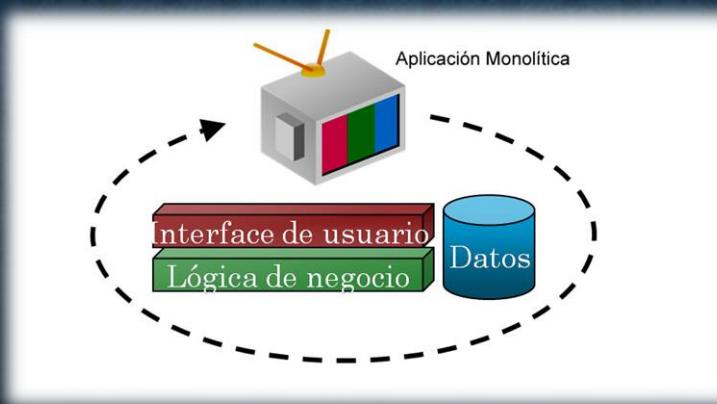
INTRODUCCIÓN



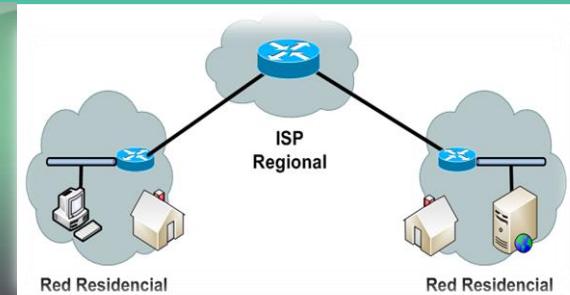
APLICACIONES DISTRIBUIDAS ... TIPOS –

APLICACIÓN MONOLÍTICA O DE UNA CAPA

- ❖ Es aquella cuya interfaz, lógica de negocio y acceso a datos se encuentran mezclados o altamente acoplados, esto dificulta que pueda hacerse una separación lógica y física donde alguna de las partes pueda ser reutilizable.



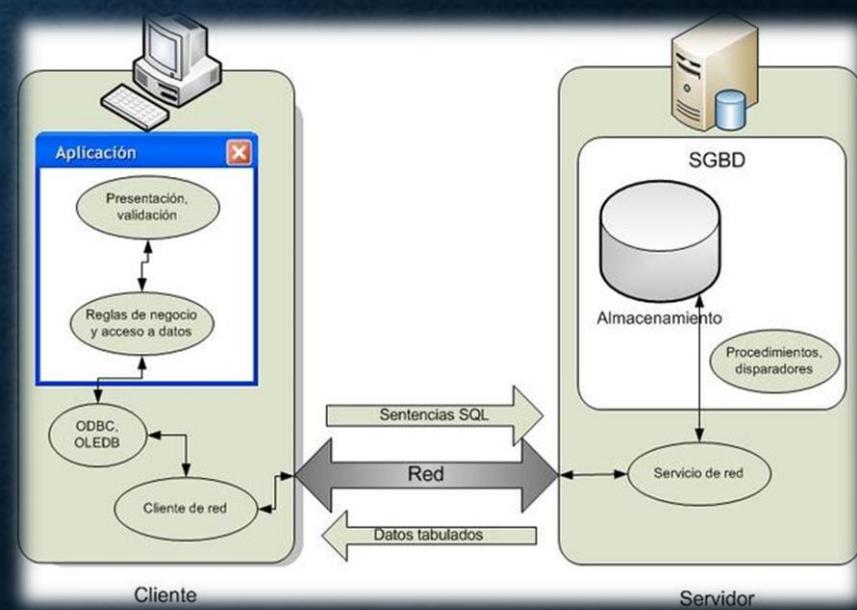
INTRODUCCIÓN



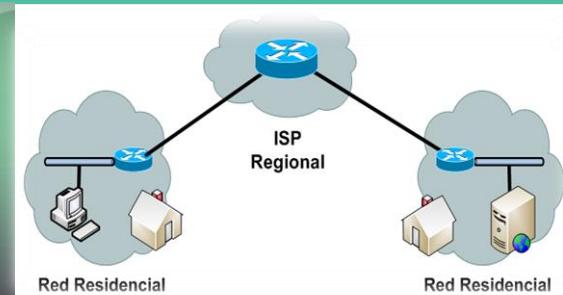
APLICACIONES DISTRIBUIDAS ... TIPOS –

APLICACIÓN CLIENTE / SERVIDOR

- ❖ Una aplicación Cliente/Servidor o aplicación de dos capas es aquella donde los datos y la lógica de negocio se encuentran separados de la interfaz, este tipo de aplicación también es denominada, cliente liviano .



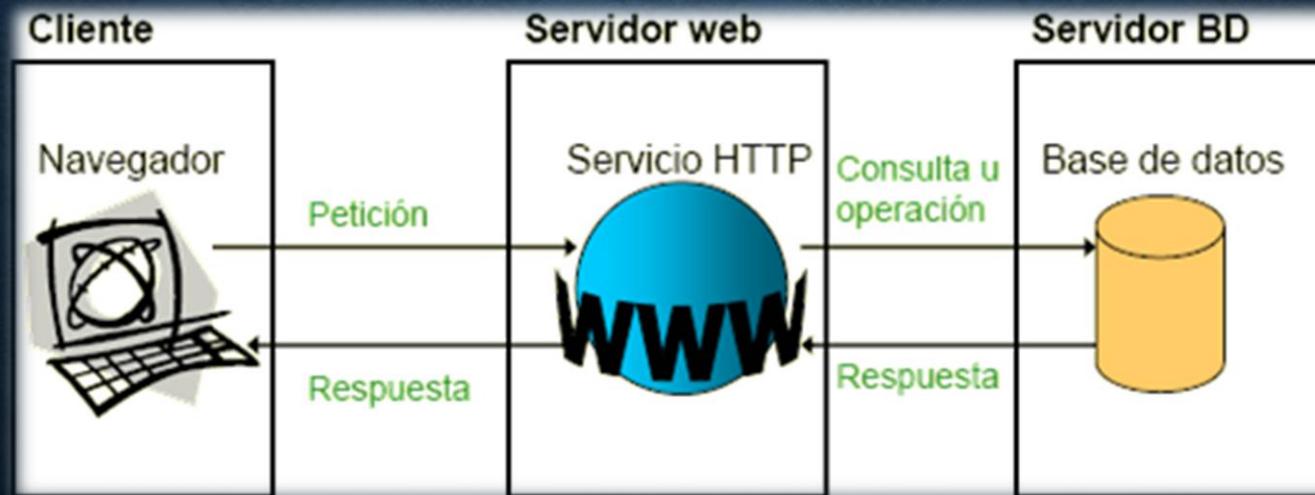
INTRODUCCIÓN



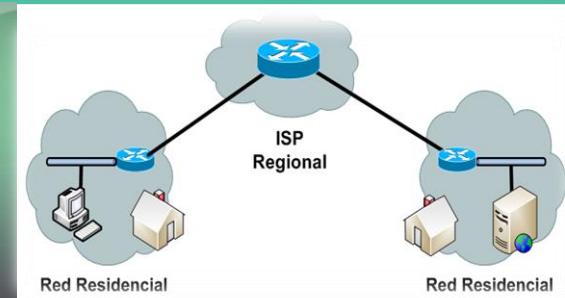
APLICACIONES DISTRIBUIDAS ... TIPOS –

APLICACIÓN DE 3 CAPAS

- ❖ Una aplicación 3 capas es aquella donde la interfaz, la lógica de negocio, el acceso a datos y los datos se encuentran separados.



INTRODUCCIÓN

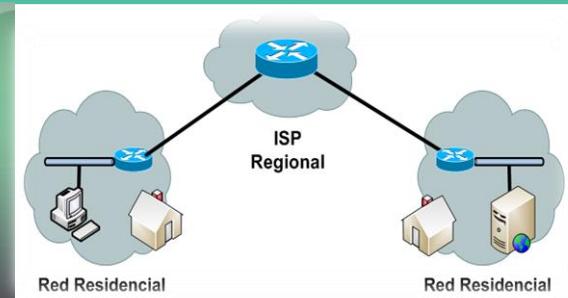


APLICACIONES DISTRIBUIDAS ... TIPOS –

APLICACIÓN N CAPAS

- ❖ En una aplicación distribuida en n-capas los diferentes procesos están distribuidos en diferentes capas no sólo lógicas, sino también físicas.
- ❖ Los procesos se ejecutan en diferentes equipos, que pueden incluso residir en plataformas o sistemas operativos completamente distintos.

INTRODUCCIÓN

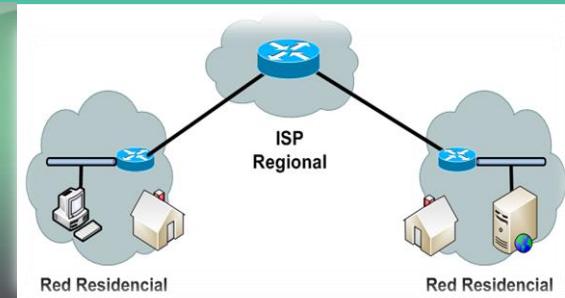


PREGUNTA

- ❖ ¿ Que razones más crees que deberían ser para implementar desarrollo de aplicaciones en N capas?



INTRODUCCIÓN

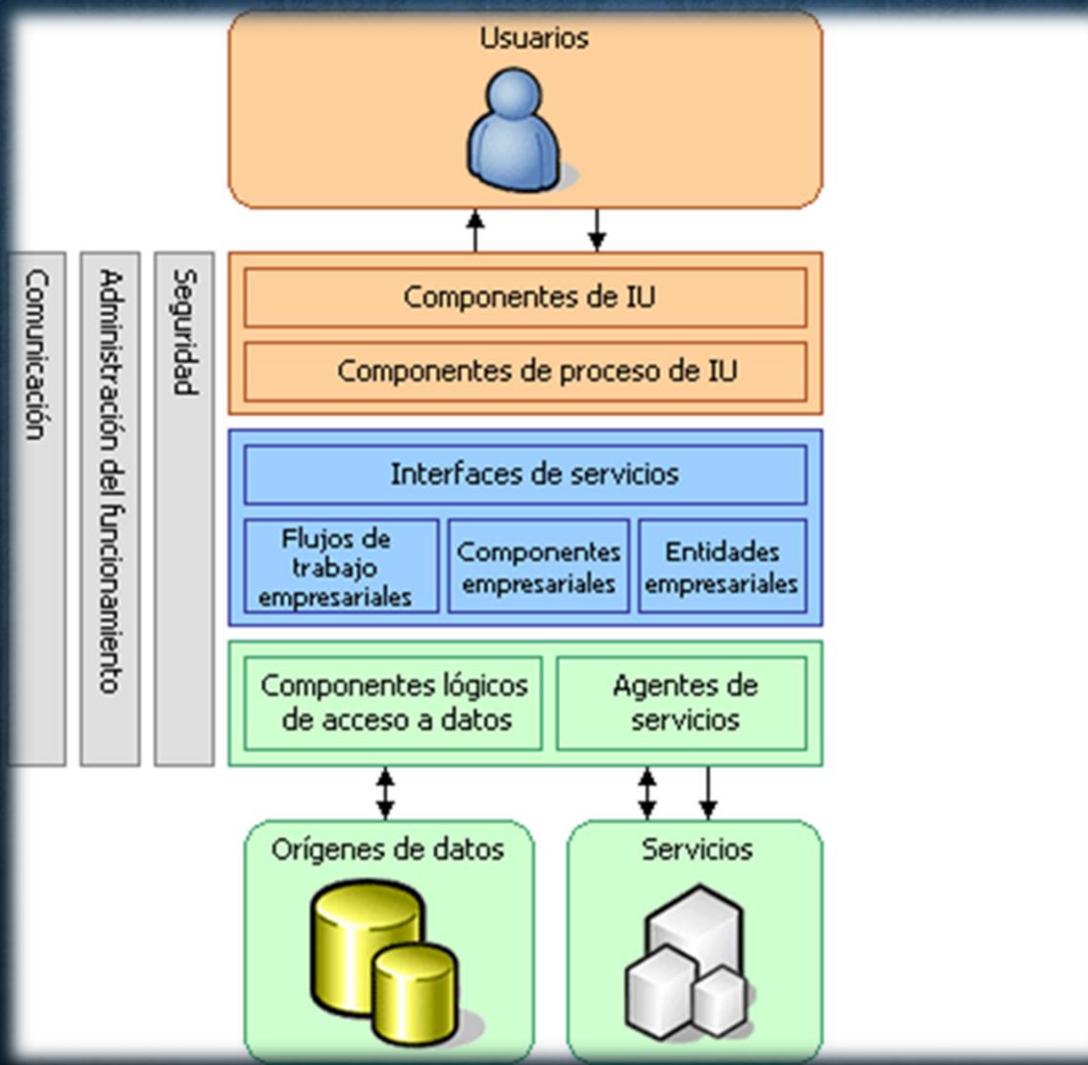
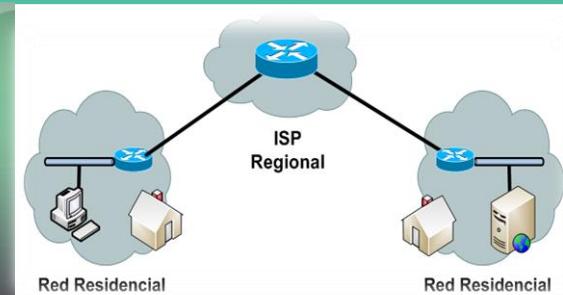


APLICACIONES DISTRIBUIDAS ... TIPOS –

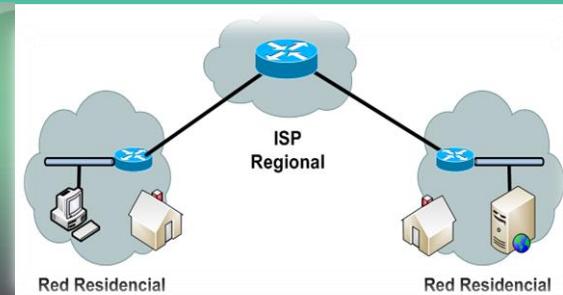
APLICACIÓN N CAPAS

- ❖ Otra de las razones importantes por las que surge este concepto, es debido a que en la evolución del desarrollo de software, se ha identificado la necesidad de crear nuevas capas, especializadas en **FUNCIONES ESPECÍFICAS**, diferentes a las 3 capas tradicionales.
- ❖ Tal es el caso de la seguridad, el control de datos entre capas, la generación de trazas de errores , entre otros.

INTRODUCCIÓN



INTRODUCCIÓN



APLICACIONES DISTRIBUIDAS ... METODOLOGÍA ...

- ❖ Áreas temáticas de formación para desarrollar aplicaciones distribuidas.
- ❖ Hay cinco grandes áreas temáticas que hay que referenciar para situarse en el entorno de desarrollo y administración de sistemas distribuidos:

Formación Básica

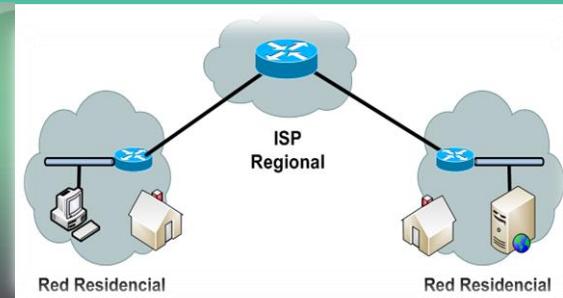
Diseño de servicios

Implementación e Integración

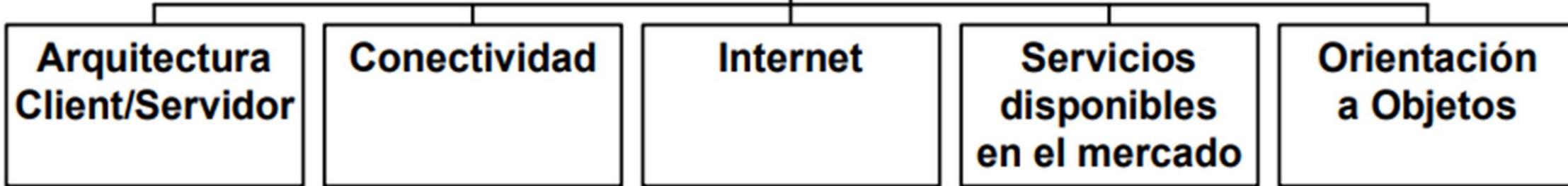
Programación

Administración del Sistema

INTRODUCCIÓN

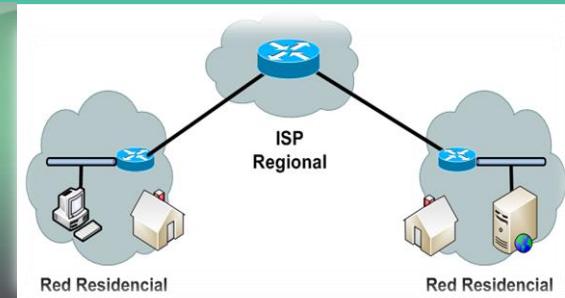


FORMACIÓN BÁSICA



Cualquier profesional que desee diseñar aplicaciones distribuidas deberá disponer de conocimientos en cinco áreas:

INTRODUCCIÓN

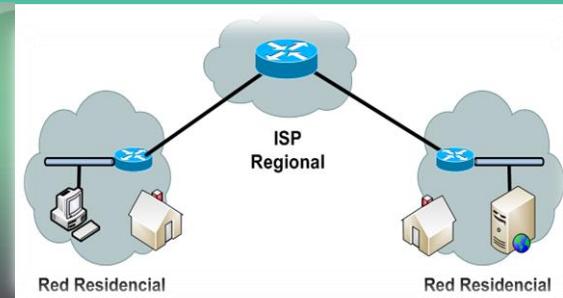


APLICACIONES DISTRIBUIDAS --- FORMACIÓN BÁSICA ---

ARQUITECTURA CLIENTE/SERVIDOR

- ❖ Los conocimientos en esta área deberán ser, evidentemente, completos.
 - ❖ Ha de saber como cambia el diseño y la administración del sistema. Y que ventajas e inconvenientes tienen las aplicaciones distribuidas por C/S sobre otras.
 - ❖ Todo ello le permitirá tomar las decisiones de viabilidad, estrategia, diseño y administración que convengan en cada caso.

INTRODUCCIÓN

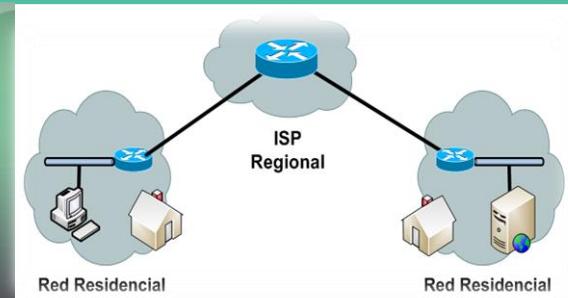


APLICACIONES DISTRIBUIDAS ... FORMACIÓN BÁSICA ...

CONECTIVIDAD

- ❖ En esta área se deben conocer las posibilidades de conectividad de la plataforma actual y de la oferta de mercado.
- ❖ La aplicación puede necesitar en el momento del diseño distribuido modificar o ampliar la plataforma y por esa razón hay que estar al día.
- ❖ Debe conocer solo que servicios le reportan para desarrollo y administración.

INTRODUCCIÓN

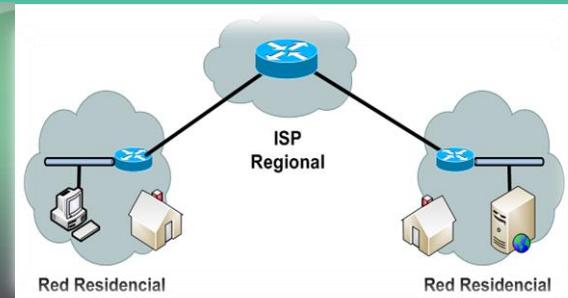


APLICACIONES DISTRIBUIDAS ... FORMACIÓN BÁSICA ...

INTERNET

- ❖ Un buen diseñador de aplicaciones distribuidas debe conocer las posibilidades de las aplicaciones Internet ya que gran parte de las aplicaciones distribuidas pueden y deben implementarse hoy día en Internet.

INTRODUCCIÓN

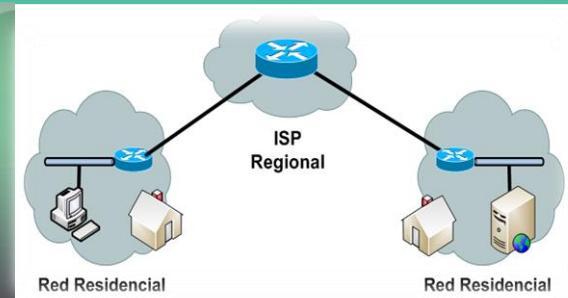


APLICACIONES DISTRIBUIDAS ... FORMACIÓN BÁSICA ...

SERVICIOS DISPONIBLES EN EL MERCADO

- ❖ Cada vez más se hace necesario estar al día de los servicios disponibles en el mercado y que pueden usarse con solo contratarlos.
- ❖ En este tema este muy atento a la evolución del mercado de Servicios WEB.

INTRODUCCIÓN

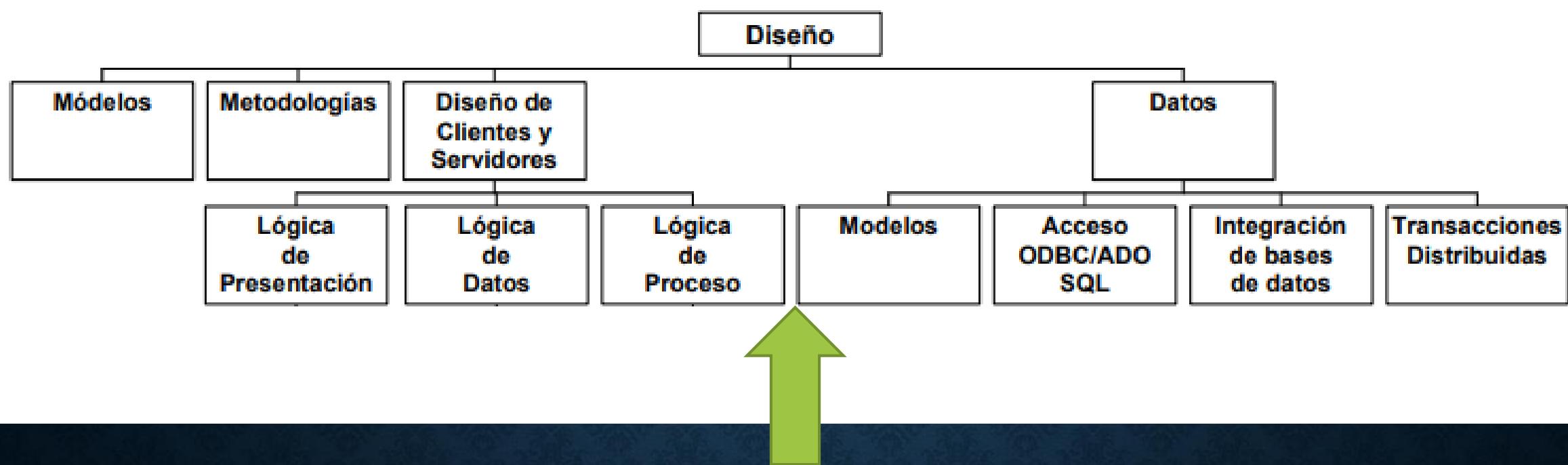
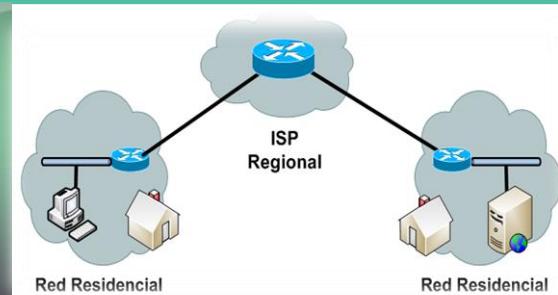


APLICACIONES DISTRIBUIDAS ... FORMACIÓN BÁSICA ...

ORIENTACIÓN A OBJETOS

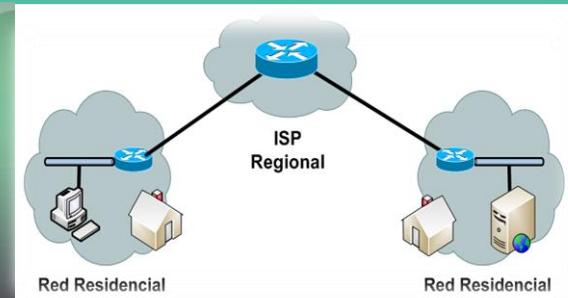
- ❖ Se puede hacer C/S e Internet sin saber OO. Sin embargo, conocer la tecnología ayuda muchísimo. Más adelante plantearemos una reflexión sobre las dos tecnologías.

INTRODUCCIÓN



Cualquier profesional que desee diseñar aplicaciones distribuidas deberá disponer de conocimientos en cuatro áreas:

INTRODUCCIÓN

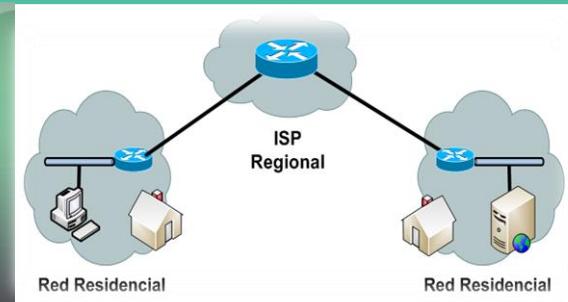


APLICACIONES DISTRIBUIDAS ... FORMACIÓN BÁSICA ...

MODELOS

- ❖ Como en todas las actividades de la Ingeniería, en el diseño informático en general, y en el C/S en particular, disponer de modelos para analizar los problemas, asociarlos a uno de esos modelos y aplicarles el esquema de solución asociado al modelo es fundamental.

INTRODUCCIÓN



APLICACIONES DISTRIBUIDAS ... FORMACIÓN BÁSICA ...

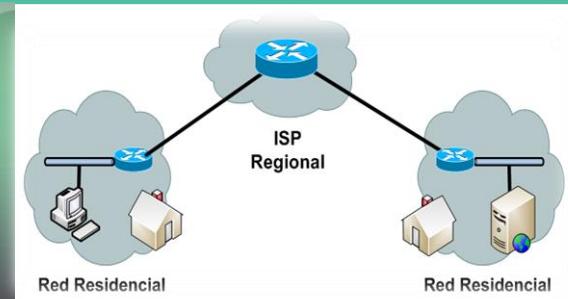
METODOLOGÍAS

- ❖ Las metodologías nos darán la pauta para resolver los problemas.

- ❖ APLICACIONES AVANZADAS

- ❖ APLICACIONES DE DESARROLLO RÁPIDO

INTRODUCCIÓN

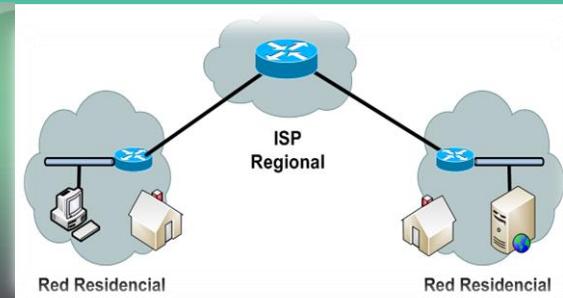


APLICACIONES DISTRIBUIDAS ... FORMACIÓN BÁSICA ...

CLIENTE/ SERVIDOR

- ❖ Existe una lógica de presentación, de datos y del negocio que hay que repartir entre clientes y servidores.

INTRODUCCIÓN

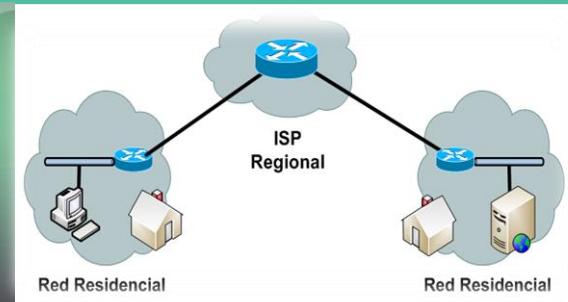


APLICACIONES DISTRIBUIDAS ... FORMACIÓN BÁSICA ...

DATOS

- ❖ La organización, integración y gestión de datos es uno de los temas más duros de esta tecnología y uno de sus más graves inconvenientes si no se pueden centralizar los datos ya que es obvio que los datos centralizados se pueden gestionar infinitamente mejor que los distribuidos.

INTRODUCCIÓN



Muchas
GRACIAS