# 2. MARCO CONCEPTUAL

En el desarrollo del presente trabajo de grado fueron usados conceptos tanto de redes de datos como de tecnologías y técnicas avanzadas para el desarrollo de software. A continuación tenemos una descripción básica de cada uno de estos aspectos, para una mayor profundización sobre estos temas se debe consultar las referencias que se encuentran en la bibliografía.

## 2.1 REDES DE DATOS

Una red datos es un sistema de intercomunicación e intercambio de información entre dispositivos o equipos, las redes se originaron por la necesidad de compartir información de forma dinámica entre computadores, dando la opción que varios usuarios dispusieran de la misma información sencilla y rápida. El compartir información fue indispensable para el eficiente uso de las nuevas tecnologías, los computadores y otros dispositivos usados por el usuario son conectados a la red por medio de una tarjeta de interfaz de red. (NIC), posibilitando el envió y recepción de datos e información. Hoy en día existen muchas topologías y clasificaciones de redes de datos. Estas están organizadas según su tamaño, método de conexión, relación funcional, tipo de transmisión, topología y según la tecnología en las que fueron implementadas. Entre la clasificación según el tamaño y según la tecnología tenemos:

|  |
| --- |
| * **Tipos de red según su tamaño** |
| * + Red de área personal (*PAN*) |
| * + Red de área local (*LAN*) |
| * + Red de área metropolitana (*MAN*) |
| * + Red de área amplia (*WAN*) |

Tabla 1 Clasificación de redes según su tamaño

|  |
| --- |
| * **Tipos de red según su tecnología** |
| Ethernet IEEE 802.3 |
| Token Ring IEE 802.5 |
| Wi-Fi IEEE802.11 |
| Bluetooth IEE 802.15 |

Tabla 2 Clasificación de redes según su tecnología

El objetivo de nuestro proyecto de grado se enfoca en la redes de Datos tipo LAN con tecnología Ethernet, usando como medio de transmisión cable de par trenzado UTP

## 2.2 Dispositivos de Redes de Datos.

Los equipos que son usados en una red de datos son denominados dispositivos y clasificados en dos tipos. El primero son los dispositivos de usuario final, como los computadores, impresoras, escáneres, cámaras y demás dispositivitos que brindan un servicio directamente al usuario final, en el segundo grupo tenemos los dispositivos que son usados para interconectar los dispositivos de la primera clase, aquí encontramos los Hub, Switch, Rauter, con ellos conseguimos una conexión entre los dispositivos que son usados directamente por el usuario, los símbolos de estos dispositivos no son estándar, pero usamos como simbología imágenes de apariencia similar a los dispositivos reales.

|  |  |
| --- | --- |
| Dispositivos de usuario final | Dispositivos de red |
| Computador (PC) | Hub |
| Impresora | Swtich |
| Servidor de archivos | Router |

### 2.1.2 Modelo OSI (Open System Interconnection)

Para el estudio de este tipo de redes usaremos el modelo de referencia OSI (Open System Interconnection), la arquitectura de este tipo de redes se analiza y se comprende claramente al segmentar el proceso de transmisión de datos a través de la red.

Este modelo de referencia está dividido en siete capas, cada una tiene determinada el manejo de un conjunto de Protocolos que son usados para establecer un canal de comunicación entre dispositivos, estas capas se aprecian en la grafica N°1

Cada una de estas capas permite fraccionar el proceso de comunicación en una red de datos.

Dentro de las funciones y características de cada una de estas capas tenemos lo siguiente:

### 2.1.1 Capa física

La capa física es la encargada de las conexiones físicas entre los equipos de la red. En esta se definen las características y los valores nominales (como voltajes, polaridad, tiempo de duración del bit y otras) del medio por el que se transfiriere la información.

### 2.1.2 Capa de enlace

Esta es la capa encargada de recibir los bits provenientes de la capa física y comprobar si durante la transmisión del mensaje hubo alguna variación de este. Esta capa fragmenta los mensajes que se están recibiendo, los analiza y determina si hay algún error en el mensaje recibido, por medio de protocolos con algoritmos que gestionan dichos bits erróneos, perdidos o duplicados. En esta capa también se maneja la dirección física de los dispositivos, esta sirve cómo identificación para cada puerto Ethernet, este número es único e irrepetible por cada uno.

### 2.1.3 Capa de red

Esta capa determina cual es el mejor camino que debe usarse para enrutar los paquetes transmitidos, aquí se incorpora la dirección lógica, origen y destino de la información. Existen diferentes algoritmos tanto dinámicos como estáticos, que determinan cual es la mejor ruta, dependiendo de diferentes variables como son tiempo, longitud del camino y estado del mismo.

### 2.1.4 Capa de transporte

Esta es la primera capa donde hay comunicación directa entre host. El objetivo es transportar eficientemente la información. Aquí se identifica cual fue el proceso o programa para lo que fue enviada la información. Esto es posible, ya que toda aplicación que se ejecuta en el host transfiere su información a través de un puerto identificado y distinguido de forma única, así si el host está ejecutando varias aplicaciones la capa de red sabe a cual pertenece cada paquete de información recibido y enviado.

### 2.1.5 Capa de sesión

De forma complementaria a la capa de transporte, la capa de sesión maneja servicios adicionales para la mejor gestión de la transmisión de la información, Por ejemplo, puede manejar “tokens (objetos abstractos y únicos) para controlar las acciones de participantes o puede hacer checkpoints (puntos de recuerdo) en las transferencias de datos”.

### 2.1.6 Capa de presentación

En esta capa se gestionan protocolos y funciones comunes para muchas aplicaciones tales como traducciones entre caracteres, códigos de números, etc.

### 2.1.7 Capa de aplicación

Finalmente en esta capa son gestionados los protocolos usados por las aplicaciones individuales.

**Capa física**: Señal y transmisión binaria

**Capa enlace de datos**: Direccionamiento físico

**Capa de red**: Direccionamiento Lógico

**Capa Transporte**: Conexión entre dispositivos

**Capa sesión**: Comunicación entre dispositivos

**Capa de representación**: Representación de Datos

**Capa de aplicación**: Aplicaciones de usuarios

**Ilustración 1 Modelo OSI**

### 2.1.2. Trasmisión de Datos

### 

En el proceso de comunicación entre dispositivos la información viaja a través de la red, pero antes de esto cada paquete de información es preparado y debe cumplir ciertas reglas y patrones para que llegue a su destino y pueda ser entendido por el dispositivo que lo recibe.

Cada fragmento de información que deseamos transmitir recibe el nombre de PDU (Protocol Data Unit), esta unidad de datos viaja a traves de las capas del modelo OSI hasta la capa física, En cada capa del modelo OSI a la PDU se le adiciona información en el encabezado o en la fin. Con esta información la trasmisión de la PDU por la red podrá ser gestionado por los equipos de red. En seguida se describe el nombre que adopta el PDU atreves de su viaje en cada capa del modelo OSI.

Aplicación

Presentación

Transporte

Sesión

Red

Enlace de datos

Física

Aplicación

Presentación

Transporte

Sesión

Red

Enlace de datos

Física

Datos

Datos

Datos

Segmentos

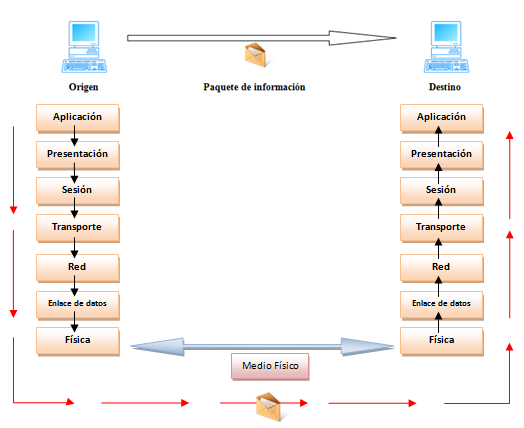
Paquetes

Tramas

Bits

En primer lugar cuando los datos viajan desde el origen hasta su destino cada capa en el origen debe comunicarse con su par en el lugar destino, Entendiendo los datos de igual manera y procesados de forma lógica. Las capas de los dispositivos excepto la capa física NUNCA se comunican directamente, la única capa que se conecta directamente es la capa física, cada PDU que se quiere transmitir siempre debe ser procesada desde la capa en el que fue originada hasta la capa física. Una vez es transmitida por la capa física, el dispositivo destino recibe la trama de datos y la procesa por cada capa del modelo OSI hasta al mismo nivel donde su originada. En este punto la información es entendida y procesada y se dice que hay comunicación de par en par.

Cada capa del origen se comunica y se entiende únicamente con la misma capa del dispositivo destino, de modo que si tenemos un segmento de información en la capa de transporte del dispositivo origen y será enviada a la capa de transporte del dispositivo destino, el PDU debe ser pasado por las capas de transporte, capa de red, capa de enlace de datos, y finalmente a la capa física, en este punto la Trama de Datos es enviada por el medio transmisión y será recibida por la capa física del dispositivo destino, una vez recibida esta es pasada a la capa de enlace de datos , luego capa de red y finalmente a la capa de trasporte. En este punto se culmina el envío de información del origen al destino. Este proceso lo podemos observar en la siguiente figura



### Proceso de empaquetamiento y desempacamiento de datos

**Flujo de datos**

Aplicación

Presentación

Transporte

Sesión

Red

Enlace de datos

Física

Aplicación

Presentación

Transporte

Sesión

Red

Enlace de datos

Física

**Flujo de datos**

**Flujo de datos**

Datos

Datos

Encabezado TCP

Encabezado TCP

Encabezado IP

Datos

FCS

Datos

Encabezado MAC

Encabezado IP

Encabezado TCP

11110101010101010111000100011110101010000101110011

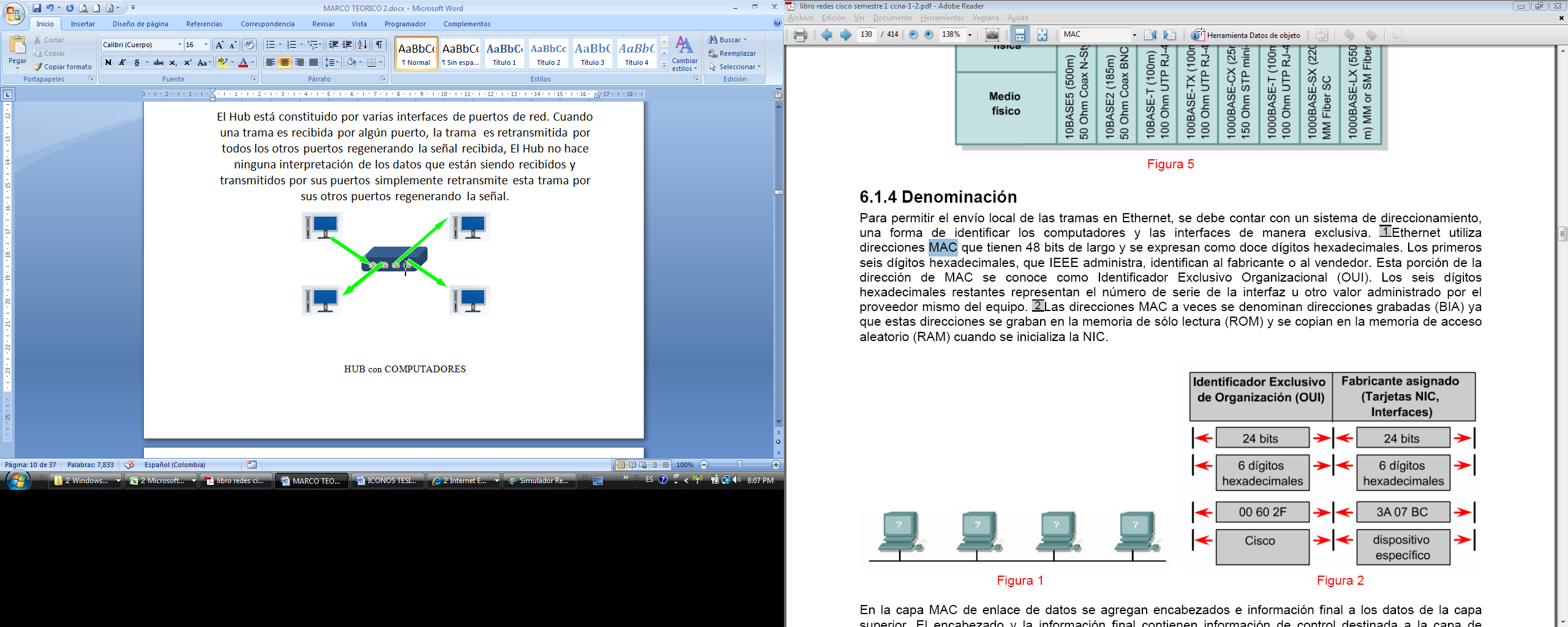
Empaquetamiento de la PDU

En el proceso de transmisión de datos, la PDU es procesada en cada capa adicionando información en la misma. La información adicionada corresponde a direcciones físicas y lógicas del origen y destino de la información, además existen otros campos en la trama de información que son usados para el control de la información y gestión de la trama durante la transmisión.

## Dispositivos de capa física y dirección física (MAC).

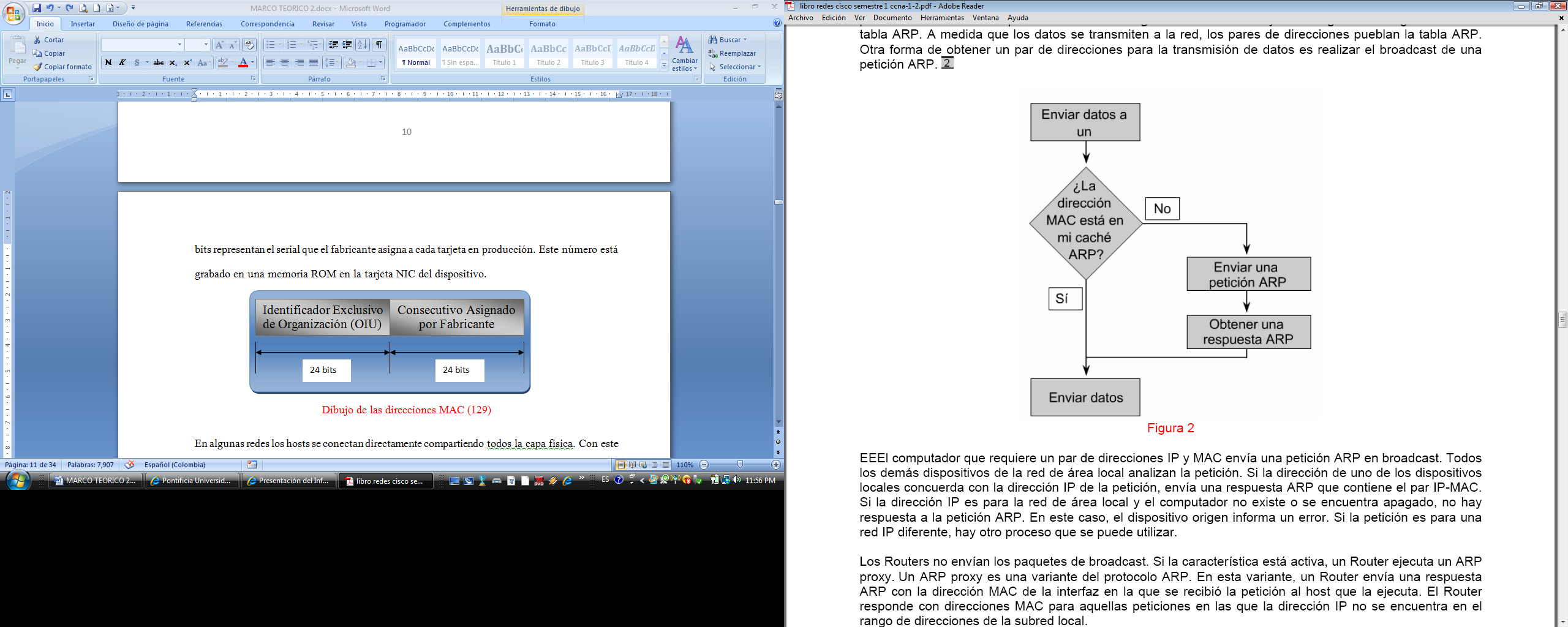
Cada equipo con una tarjeta de interfaz de red que esté conectado a la red, se denomina host, para tener comunicación entre dos Host dentro de una red, debemos tener una conexión física entre sus tarjetas de interfaz de red. La información intercambiada entre los equipos se transmite a través de señales eléctricas, electromagnéticas u ópticas que mediante una modulación y demodulación adecuada transmiten bits de información correspondientes a la trama de información. Usamos el dispositivo de red llamado Hub cuando la señal es degradada o pierde potencia, el Hub regenera y retransmite nuevamente la señal.

El Hub está constituido por varias interfaces de puertos de red. Cuando una trama es recibida por algún puerto, la trama es retransmitida por todos los otros puertos regenerando la señal recibida, El Hub no hace ninguna interpretación de los datos que están siendo recibidos y transmitidos por sus puertos simplemente retransmite esta trama por sus otros puertos regenerando la señal.



HUB con COMPUTADORES

Para la identificación de cada host la tarjeta de interfaz de red está identificada con un numero único, este es la dirección MAC o dirección Física, este número identifica de forma única los equipos y dispositivos que están conectados a la red. Esta identificación se compone de 48 bits, los primeros 24 bits identifica el fabricante o vendedor de la tarjeta de interfaz, los restantes 24 bits representan el serial que el fabricante asigna a cada tarjeta en producción. Este número está grabado en una memoria ROM en la tarjeta NIC del dispositivo.



Dibujo de las direcciones MAC (129)

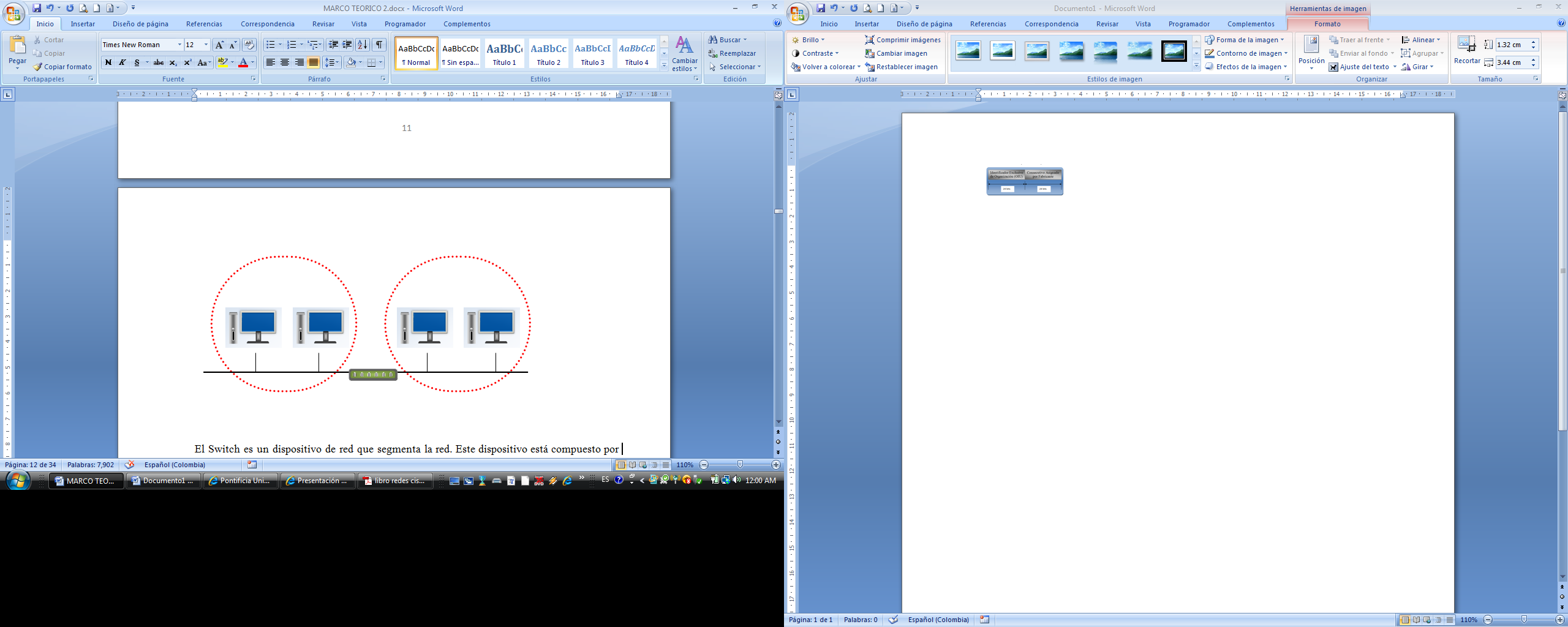
En algunas redes los hosts se conectan directamente compartiendo todos la misma capa física. Al no haber direccionamiento de tráfico de información en el medio de transmisión compartido podemos tener colisiones de los mensajes transmitidos ocasionando incoherencia y errores en la información, el segmento de red donde los host usan un mismo medio de transmisión directamente se denomina *Dominio de colisiones*. Entre mayor sea el numero de host conectados en un mismo segmento de red, mayor será el dominio de colisiones haciendo el transporte de los datos más demorado y el ancho de banda del medio mas limitado. La capa de enlace de datos nos provee de protocolos y dispositivos de red para el uso más eficiente del medio,

Podemos segmentada la red, de modo que las tramas de datos que son transferidas por un host no sean replicadas a todos los rincones de la topología de red, sino que estas tramas sean filtradas y direccionadas a su destino, logramos tener dividido el dominio de colisiones en segmentos. Haciendo más eficiente el ancho de banda de la red.

El Switch es un dispositivo de red que segmenta la red. Este dispositivo está compuesto por varios puertos de red. En cada puerto de red se tiene una interfaz de datos a la red donde se envían y reciben bits de información, el Switch basándose en las dirección MAC que tiene relacionadas en cada una de sus interfaces filtra la información y dirige a cada trama únicamente los el puerto por donde se encuentra conectado el Host con la dirección MAC destino, enseguida se describe los pasos de operación y funcionamiento del Switch.

|  |
| --- |
| * Al encender el switch, las tablas se encuentran vacías |
| * AL detectar en alguna interfaz del switch una trama, verifica si ya tiene guardada en sus tablas la MAC del origen del mensaje que está recibiendo, si no lo está la guarda si ya la tiene guardada verifica lo siguiente |
| * Verifica si tiene en sus tablas la dirección MAC destino, si la tiene relacionada envía la trama de datos por el puerto correspondiente, si no la tiene relacionada en sus tablas lo envía por todos por puertos excepto por que recibió del mensaje. De esta forma se garantiza que el mensaje llegue a su destino. |

Al utilizar el Switch tenemos una segmentación de la red, el ancho de banda de la red aumenta al disminuir el número de host que tenemos en el dominio de colisiones para cada segmento. El switch es un dispositivo de capa dos, es de uso difundido en las redes de comunicaciones al proveer de gran numero de conexiones y segmentaciones de red, haciendo la red más eficiente.



Con el Switch también tenemos la opción de crear la LAN virtuales (VLANS), con este opción podemos crear redes independientes dentro de una misma red física, varias VLANs pueden coexistir en una misma red física, con el uso de VLANs reducimos es tamaño del dominio de colisiones de la red, segmentando la red con lo que conseguimos un mejor uso del medio de transmisión.

Una Vlan se crea mediante el dispositivo de red Switch, en este se configuran sus puertos de tal forma, que aunque los host estén conectados al mismo switch las Vlans que fueron creadas sean un Dominio de Colisiones distinto.

# Direccionamiento IP

Dentro de una red grande con muchos Host conectados el direccionamiento físico no es suficiente para dirigir el tráfico de información por la red, esta situación sería similar a una ciudad donde el único medio para repartir correspondencia fuera el nombre de la persona destinataria. Este método podría servir en un lugar pequeños donde la mayoría de personas conocieran el resto, pero en un lugar más grande este método de distribución de correspondencia seria obsoleto. El método de reparto de correspondencia en una ciudad se basa en el uso de direcciones con calles y carreras acompañadas del número de domicilio, este direccionamiento es consecuente pues los números de las calles y carreras con consecutivos. De esta misma forma el direccionamiento de información en una red ampliada se hace por medio de un direccionamiento lógico. En el nivel físico la dirección MAC del equipo es equivalente al nombre del equipo, el direccionamiento lógico nos ayuda a buscar un equipo del mismo modo que una dirección de un inmueble nos ayuda para la ubicación de una persona en una ciudad.

Para el direccionamiento lógico cada interfaz de red está identificada por una dirección única llamada dirección IP, la dirección IP está constituida por cuatro octetos de bits (32 Bits). La primera parte representa a que red pertenece el equipo, la segunda parte de la dirección IP representa el ID del host. En el siguiente cuadro observamos esta división.

RED

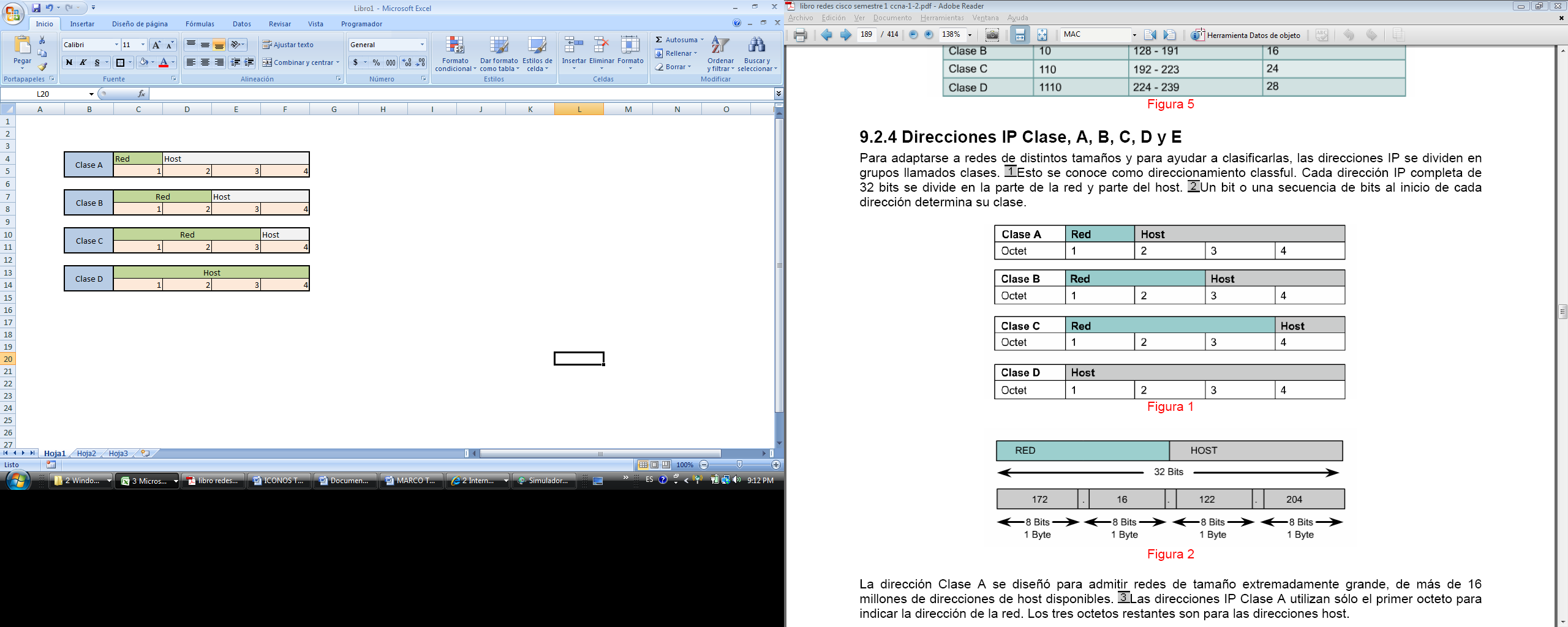
HOST

32 Bist

Las direcciones IP se clasifican según el número de redes y host que pueden albergar, en la siguiente forma

* Clase A 1.0.0.1 -- 126.255.255.255
* Clase B 128.0.0.1 – 191.255.255.255
* Clase C 192.0.0.1 – 223.255.255.255

Según sus octetos los segmentos de red y de ID de host serian los siguientes



Con esta organización, el direccionamiento lógico alcanza mayor eficiencia, con el gran crecimiento de las redes de comunicaciones el número de direcciones IP podría ser agotado en su totalidad. Para la mejor utilización de las direcciones IP y para que estas no se agoten, se creó una nueva segmentación de estas redes llamada subredes, cada segmento de red puede ser dividida en varias subredes, de esta forma una dirección IP se puede usar varias veces, pero al estar en diferente subred, nunca se confundirán las direcciones, estas subredes al salir a la nube de internet salen como solo una IP. Es de vital importancia las direcciones MAC, pues varios computadores son representados por una misma dirección IP, pero nunca podemos tener direcciones MAC repetidas, es importante aclarar que la dirección IP es dinámica, es decir puede ser cambiada según el punto de red donde estemos conectando el equipo, pero la dirección MAC es siempre la misma (del mismo modo que nosotros al trastearnos cambia la dirección pero nuca podemos cambiar nuestro nombre)

El dispositivo de red llamado Router es un dispositivo de capa 3 utilizado para gestionar el direccionamiento de información en la red por medio de las direcciones lógicas IP, el Router al igual que el switch, tiene tablas de almacenamiento. Una de las funciones del Router es almacenar las direcciones IP de las redes a las que están conectados los puertos del Router. De modo que cuando una trama de datos es recibida por un puerto del Router, este analiza la dirección IP a la que necesitamos enviar el mensaje, si la tiene almacenada en sus tablas de enrutamiento, direccionara esta trama por el puerto de salida. Si no la tiene almacenada en sus tablas, el Router reenvía el mensaje por todos los puertos esperando respuesta, si el mensaje pasa por otros Routers y no llega a su destino el paquete está configurado para que a cierto número de salto sin tener respuesta sea destruido automáticamente

### Subredes y mascaras de red

Como ya se había mencionado la clasificación de redes en los tipos ABCD, es un patrón que se usa internacionalmente. pero no es una estructura estática a la que nos debemos regir, si nuestras necesidades son otras. La división de las direcciones IP en las clases mencionadas se hace mediante mascaras de red, este es un numero de la misma longitud de una dirección IP, con la parte que representa la red sustituida por unos y la dígitos que representan el host son remplazados por ceros, esto se muestra enseguida.

**Mascaras de Red Predeterminadas**

* Clase A: 255.0.0.0 11111111.00000000.00000000.00000000
* Clase b: 255.255.0.0 11111111.11111111.00000000.00000000
* Clase C: 255.255.255.0 11111111.11111111.11111111.00000000

Cada una de estas clases se puede dividir en subclases al aumentar el número de unos de la mascara

**ID HOST**

**ID RED**

SUBRED ID

De esta manera el número de host que podemos tener por red disminuye pero aumenta el número de redes que podamos tener. Un ejemplo de lo que sucede con el uso las subredes es el siguiente:

* Tenemos un host con la siguiente direccion IP

Esta corresponde a una IP clase C, pero las mascara de red es diferente.

Analizandola en binario tenemos:

Dirección IP: 192. 168. 0. 5

Mascara: 255. 255. 255. 224

Dirección IP: 1100 0000 . 1010 1000 . 0000 0000 . 0000 0101

Mascara: 1111 1111 . 1111 1111 . 1111 1111 . 1110 0000

Dirección de subred

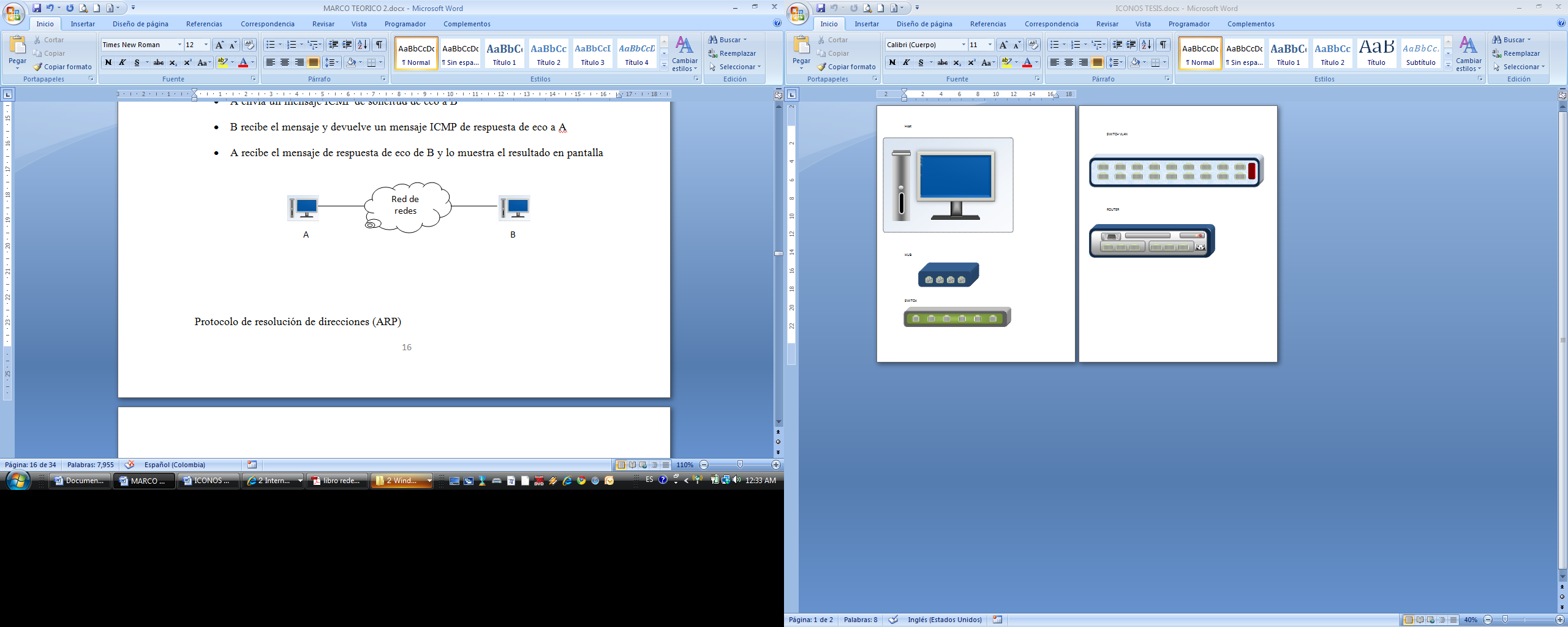
Dirección de red

El tener esta configuración de dirección IP nos da la posibilidad de tener:

* + 23= 8 subredes
  + 25= 32 host en la red

Como método para probar la conectividad básica entre redes, existe las pruebas de eco, donde un paquete es enviado a un host y se le solicita como respuesta un eco confirmando que si recibió el primer paquete. El comando Ping cumple con esta función. El envía un paquete al host destino y luego espero un paquete de respuesta de ese host, este es un mecanismo de prueba básico para verificar conectividad. El Ping es un comando del protocolo ICMP (protocolo de control y error de mensajes) su funcionamiento en caso de no producirse problemas en el camino es el siguiente:

* A envía un mensaje ICMP de solicitud de eco a B
* B recibe el mensaje y devuelve un mensaje ICMP de respuesta de eco a A
* A recibe el mensaje de respuesta de eco de B y lo muestra el resultado en pantalla



## Protocolo de resolución de direcciones (ARP)

Las tramas de información que son transmitidas deben tener tanto la dirección MAC origen como la dirección Mac destino. Si el paquete no tiene algunas de estas dos dirección no es procesado por las capas superiores a la capa 3, las direcciones MAC y IP actúan como controles y son vitales para el enrutamiento de la información. Por medio del protocolo de resolución de direcciones una vez que sepamos la dirección IP destino, podemos adicionar la dirección Mac del dispositivo de red al que necesitemos alcanzar. En Algunos dispositivos de red como los host se tiene en su memoria tablas que contienen las relaciones de las direcciones pares IP – MAC de los dispositivos que están relacionados con su red, estas tablas reciben el nombre de tablas ARP.

Para la recolección de los pares de direcciones los dispositivos usan dos formas, la primera es monitorear la información que llega a su tarjeta de interfaz de red. La tarjeta de interfaz de red siempre lee la dirección IP y Mac de todas las tramas que llegan a sus terminales para determinar si el mensaje es dirigido a ella, en este momento al descartar el mensaje, guarda los pares de direcciones IP- MAC, en el momento que el host desee enviar una trama a cierta dirección IP, primero busca su par dirección MAC en su tabla ARP, si la tiene relacionada simplemente la adiciona a la trama, pero si no la tiene guardada en su memoria hace un broadcast de petición de ARP, este método corresponde a la segunda forma de recolección de pares de direcciones IP-MAC, su funcionamiento es de la siguiente forma.

Transferir información a un IP host

Enviar un broadcast solicitando ARP

La dirección MAC esta mi tabla ARP

Recibir respuesta ARP

Transferir información

* El dispositivo de red que requiera saber la dirección MAC, correspondiente a la IP destino envía un broadcast a la red preguntando si algunos de los dispositivos locales concuerda con la dirección IP de la petición
* Si el equipo con la dirección IP está en la red, envía una respuesta ARP que contiene el par IP-MAC

Si la dirección IP que estamos buscando esta fuera de nuestra red el Router por medio de este protocolo se encarga de buscar la dirección MAC en redes vecinas. Al no tener ninguna respuesta el equipo que está solicitando la dirección MAC informa un error.

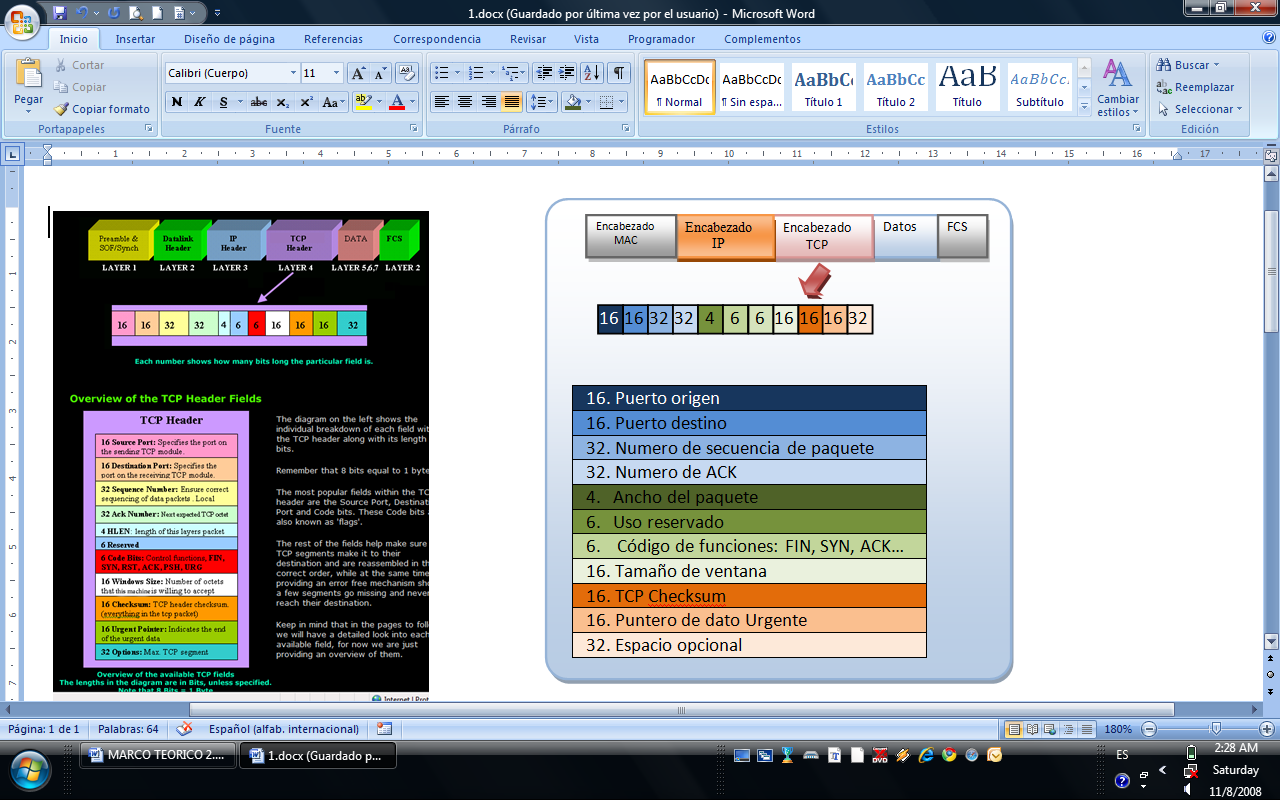
### Protocolo TCP (protocolo de transporte)

TCP es un protocolo que opera en la capa de transporte del modelo OSI, este protocolo tiene como objetivo trasferir archivos de un host a otro de forma confiable y segura, es de vital importancia ya que con él las aplicaciones y protocolos más avanzados de las capas superiores pueden transferir su información. Las características del protocolo de transferencia de información TCP son las siguientes:

* Transporte fiable
* Orientado a conexión
* Control de flujo de información

### TRANSPORTE FIABLE

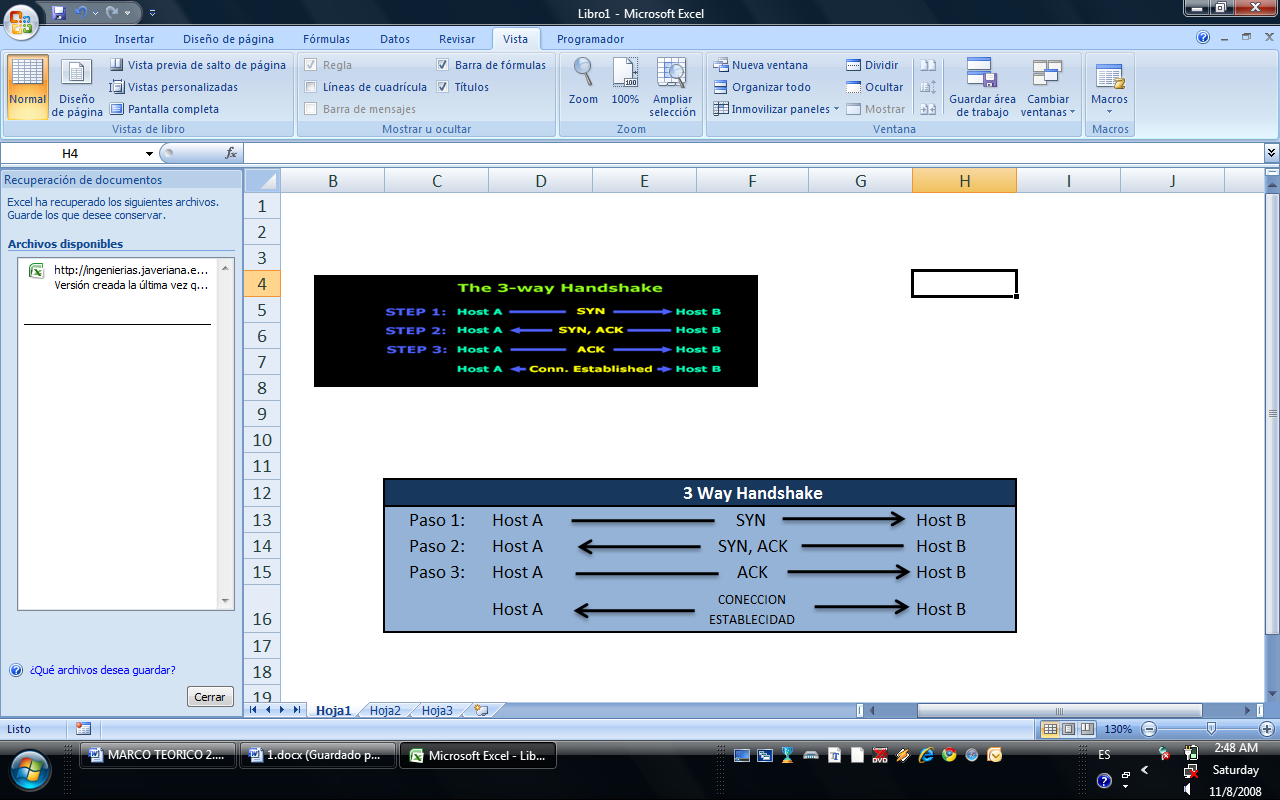
TCP es un protocolo de transporte fiable ya en su modo de operación utiliza diferentes técnicas para garantizar que los datos recibidos estén libres de errores. Dentro del segmento de información el protocolo TCP inserta su encabezado en el PDU, la información que tiene el encabezado TCP se aprecia en la siguiente figura.



Estos campos en la cabecera TCP son usados para ejercer un control de la información que se está transmitiendo.

### Orientado a conexión

El ser orientado a conexión, implica que TCP establece una conexión entre los dos equipos antes de que los datos sean transferidos. El método de conexión entre los host se denomina 3 way handshake, el siguiente diagrama muestra el procedimiento de conexión 3 way handshake.

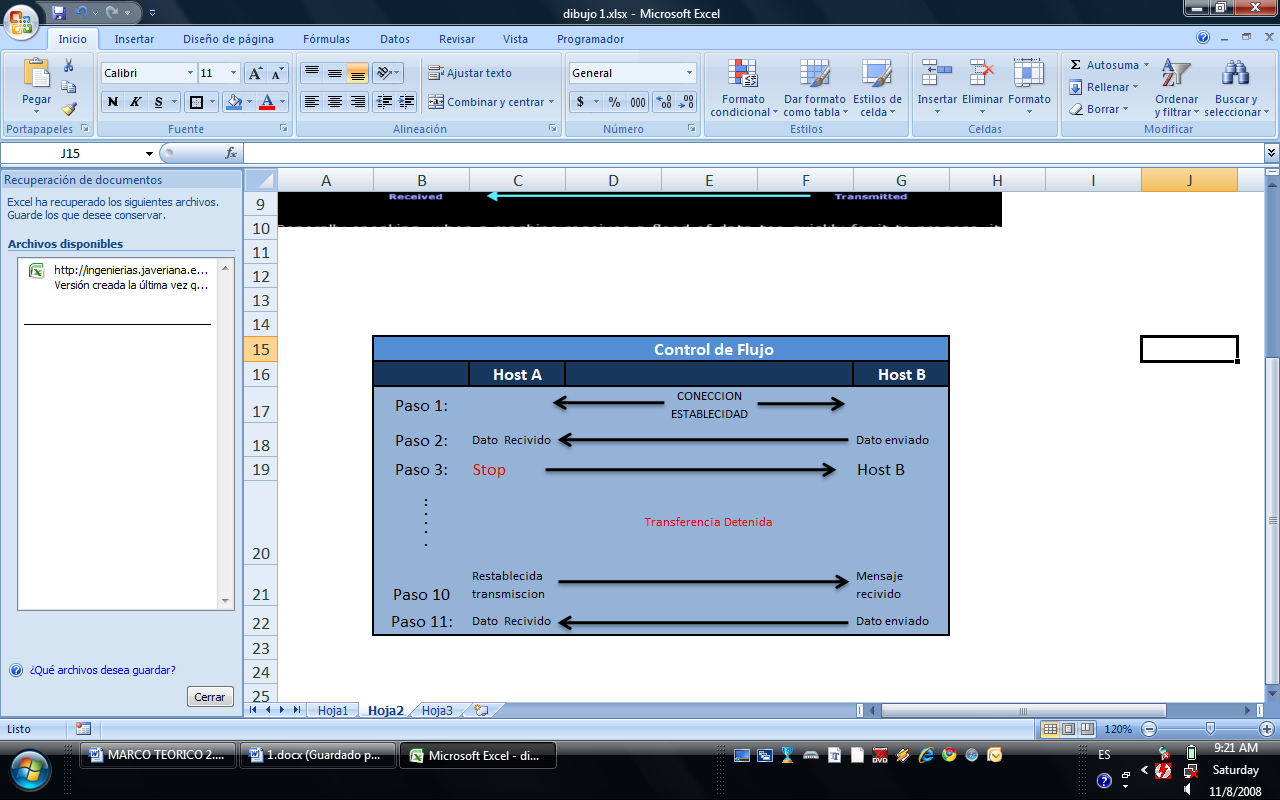


* Paso 1: El host A envía un SYN al host B. El host B entiende que A esta tratando de establecer una conexión con él
* Paso 2: si el host B tiene las condiciones adecuadas para establecer una conexión, contesta con un SYN, ACK, lo que se entiende por host A, que reconoce y acepta su solicitud de conexión.
* Paso 3. El host a envía al B un ACK, que implica que reconoce y acepta su solicitud anterior.

Una vez que el 3 way handshake se haya completado, se establece la conexión y la transferencia de datos comienza.

### Control de flujo

El control de Flujo se utiliza para controlar el flujo de datos entre la conexión de los host, si existe algúna razón por la que algunos de los host no sea capaz de sostener la tranferencia de informaqcion , este envía al otro un aviso solicitando la detección de transferencia, una vez sea solucionada la razón por la que se paro la tranferencia, este solicita la reanudación del envio de información. Esto se aprecia de en la siguiente figura:



## ENRUTAMIENTO

El enrutamiento es el proceso mediante el que se determina cuales la mejor ruta de se debe escoger para transferir un archivo de un lugar origen a su logar destino. Es posible que en nuestra topología de red existan varios caminos de conexión entre dos host, con los protocolos de enrutamiento se busca la ruta más eficiente para la transferencia de información.

Entre los parámetros y las condiciones para escoger la mejor red tenemos:

* Menor Tiempo medio de retardo
* Menor número de saltos entre dispositivos de red.

Es más sencillo determinar la mejor ruta, eligiendo la ruta en la que halla el menor número de saltos en la red. Realmente en las redes se usa un hibrido entre varios parámetros para escoger la mejor ruta

Para tener acceso a redes que estén fuera del alcance de la red local, los dispositivos de red emisora de mensajes tienen configurado una dirección de enlace predeterminada que apunta a la puerta de un Router, este enlace se llama Gateway, toda la comunicación que un host quiera establecer con otro host que este fuera del alcance local, este se realiza predeterminadamente por el puerto de salida Gateway.

La tarea de escoger la mejor ruta la realiza el dispositivo de red llamado Router, para un Router ser capaz de enrrutar un paquete de datos el usa una tabla de información donde registra todas las posibles rutas que están conectadas a sus interfaces, en esta tabla también se registra cuantos saltos hay que hacer para llegara a su red destino. La tabla de enrutamiento que tiene cada puerto del Router se puede alimentar de dos formas, Dinámica o estáticamente

* En rutas estáticas son las determinadas manualmente por el usuario, la única forma de variar estas rutas es manualmente.
* En las rutas estáticas las tablas de enrutamiento son intercambiadas entre los Routers vecinos cada cierto tiempo, con este intercambio de información, los Routers pueden determinar cual es la ruta con las mejores características para enviar la información.

RIP, es un algoritmos de de enrutamiento, con el se determina cual es la mejor ruta analizando en menor número de salto que haya que hacer para llegar al destino. Las tablas de enrutamiento de los Routers vecinos son intercambiadas cada 30 segundos. El Rip es un protocolo de enrutamiento sencillo y de uso difundido en los principios de las redes de comunicación, hoy en días existen algoritmos de enrutamiento mucho más robustos y eficientes que analizan muchas más variables para determinar cuál es la mejor ruta, pero el funcionamiento básico de enrutamiento sigue siendo válido.

Una vez visto la teoría de las principales características de una red LAN con tecnología Ethernet, Ahora mostraremos brevemente las principales características de la programación orientada a objetos y las especificaciones de la plataforma de desarrollo .Net. Con el estudio de estos aspectos es un buen principio para emprender con la práctica de nuestro software de simulación redes Lan para la tecnología Ethernet, con el uso y practica de nuestro software estos principios serán mas claros y evidentes las teorías expuestas. Enseguida seguiremos exponiendo sobre las características con las que fueron desarrolladas el software. Una vez visto esto seguiremos con las características y espcificaciones del software seguido de los desarrollos alnsados.

# Metodologías para desarrollo de software.

## Programación orientada a objetos.

### 

El estudio de las características y la teoría sobre las diversas aspectos que implican el funcionamiento de una red tipo Lan y componentes software anteriormente expuestos y las necesidades particulares de la materia de integración de redes, definimos y delimitamos las características de software del simulador que desarrollaremos en nuestro proyecto.

El diseño de la arquitectura del software, se desarrollo en base en la programación orientada a objetos. Con la Programación orientada a objetaos el desarrollo del software es mucho más eficiente y sostenible. Entre los conceptos fundamentales para describir este tipo de programación tenemos los siguientes:

* Clase: describe las propiedades y comportamiento que tiene el objeto en particular, la instanciación es la asimilación de estas propiedades y la creación de un objeto a partir de la lectura de las propiedades y comportamientos de la clase.
* Objeto: La Clase instanciada forma un objeto, además este equipado con un comportamiento o método.
* Metodo. Son los comportamiento o algoritmos que se desencadenan al llegar algún mensaje al Objeto. Estos comportamientos pueden estas orientados a cambios y operaciones del objeto como etidad.

Con este tipo de programación, logramos que la practica de programar sea mas dinámica al pensar mas en soluciones de tipo sistemático, donde cada sistema es un objeto que interactúa con otro, pero el funcionamiento interno es transparente. Únicamente vemos cuales son las variables de salida. Además de estas aspectos, la programación orienta a objetos logra las siguientes características respecto al desarrollo de software.

* Abstracción: La idea de abstracción al estar referida a programación, nos evoca a que en mas sencillo el solucionar partes pequeñas de nuestro desarrollo de software, la abstracción de software lo que implica es hacer una partición del problema y solucionarlo por partes.
* Encapsulación: una vez tratado la parte del problema, lo que importa al mundo exterior de la parte son las variables con las que interactúa el sistema con su mundo externo. De forma que cuando tengamos que interactuar con un objeto para, mi sea tranparente los que pasa interiormente, simplemente me importa las variables que puedo manejar el Objeto y el método en que se desenvuelven.
* Herencia: esta carateristica no permite el ahorro de esfuerzo al escribir líneas de código al desarrollar un software. Una vez descritas las características de las clases, una vez instanciadas esta pueden cambiar su aspecto, pero las características con las que fueron creadas se conservan.
* Polimorfismo: esta característica nos determina la forma especial de los objetos de responde a los mensajes o eventos en función de los parámetros utilizados durante su invocación.

## DESARROLLO DE LA APLICACIÓN EN CAPAS:

Una característica de nuestro software es que sea extensible sencillamente, ya que servirá como base para implementaciones especificas, o para la actualización de nuevas tecnologías. Cuando sea necesario implementar un nuevo módulo, no se tendrá que volver a escribir todo el código; el diseño de esta aplicación debe permitir la reutilización del código, y que sólo sea necesario codificar nuevas implementaciones, permitiendo que la aplicación sea extensible de una forma rápida para futuros proyectos que requieran ser complementados.

Se podría hacer funcionar la aplicación en forma rápida, haciendo caso omiso de ningún patrón de desarrollo de software por capas, dejando todos los módulos que la compongan altamente acoplados entre ellos, esto conllevaría a que sería imposible la extensión de esta aplicación, o que fuera necesaria volver a escribir gran parte de código, por lo que se podría correr el riesgo que la aplicación quede limitada a las funcionalidades que se van a realizar con este proyecto.

Otro inconveniente que se encuentra al no utilizar el diseño por capas, sería cuando la tecnología de .NET este obsoleta, o sean creados o actualizados nuevos módulos, sería imposible estar a la altura de estos cambios, esto conllevaría a lo que ocurre con muchas aplicaciones, que después de un tiempo serían obsoletas porque no pueden ser utilizadas debido a un cambio de tecnología( como el cambio de Windows XP a Windows Vista), o si se desea utilizar nuevas tecnologías es imposible acondicionarlas, debido a que todos los componentes de la aplicación están altamente acoplados, y un cambio en cualquiera de ellas involucra un cambio en todos los componentes de la aplicación.

El otro inconveniente que encontramos al no diseñar por capas, es la dificultad para realizar pruebas unitarias a los módulos que componen la aplicación, porque estos, entre si están altamente acoplados.

Por estas razones se usará un modelo en capas en donde esté altamente diferenciado, la visualización, el modelo de negociación, y el acceso a los datos.

La visualización es la capa encargada de la interfaz con el usuario, y su única responsabilidad será la de recibir cualquier evento que sea originado por el usuario, o desplegar la información necesaria al usuario. En este caso se utilizará para la capa de visualización, ventanas de Windows para la representación de la información.

La capa de negociación es la encargada del manejo de cálculos y procesos de los datos, esta capa no conoce la existencia de una visualización, o una capa de acceso a datos, su única función es la de realizar toda la lógica de negociación correspondiente. Esta capa será el núcleo de la aplicación en donde verdaderamente estará ocurriendo toda la lógica de la simulación.

En la capa de acceso a datos, está la abstracción de la fuente de datos, que va hacer usada por la capa de Negociación, esta capa obtendrá los datos de la topología de la red, desde un archivo, o una base datos o usando .NET Remoting, para obtener los datos de red de la capa de datos de una aplicación remota del simulador.

Con esta división por capas, es sencilla la extensión de la aplicación debido a que están altamente desacoplados los módulos principales, de esta manera si en un futuro se requiere presentar este simulador en un ambiente WEB, lo único necesario será implementar la capa de visualización en un entorno WEB, la capa de negociación y el acceso a datos quedarán intactas.

Si en un futuro la información que maneja el simulador es demasiado grande, en donde el modelo de acceso a datos de la configuración de la red, a través de archivos no pueda ser mantenida, y sea necesaria la utilización de una base de datos relacional, tan solo sería necesario cambiar la capa de acceso a datos, para que soportara este nuevo formato, las restantes capas quedan intactas, y la aplicación continuará funcionando normalmente.

Lo mismo ocurre al nivel de la capa de negociación, cuando se requiera hacer la implementación de un nuevo protocolo, o cambiar la lógica de algún dispositivo, esto no debe afectar las demás capas.

### COMUNICACIÓN ENTRE LAS DIFERENTES CAPAS

Para la comunicación entre las 3 capas se va a usar la metodología Modelo Vista Presentación. Esta metodología, es muy recomendable para desacoplar las capas de visualización de negociación, como también para que se puedan realizar pruebas unitarias tanto a la visualización como al modelo de negociación.

El modelo es el objeto donde residen los datos de la aplicación y provee métodos para acceder a ellos, en este también esta toda la lógica de negociación, el modelo no debe tener ninguna asociación con la interfaz con el usuario, no contiene una referencia a la vista, el modelo envía eventos cuando sus datos cambian, en el caso del simulador, van a existir diferentes modelos, cada uno para los diferentes dispositivos que existan y diferentes protocolos a implementar.

El nivel de vista es el encargado de mostrar los datos del modelo al usuario, en el caso del simulador la vista será una ventana de Windows, en donde estará el mapa de la topología de red, y la información de cada dispositivo, la vista notificará mediante eventos, cuando intervino el usuario.

La presentación es la encargada de sincronizar y es la mediadora entre la vista y el modelo, la presentación conoce tanto al modelo como a la visualización y continuamente está escuchando los eventos de acción del usuario; una vez la presentación detecte este evento leerá los datos de la visualización y los transportará al modelo, la presentación también escucha los eventos de cambio del modelo, para que cuando ocurran, la visualización sea refrescada con los cambios que ocurrieron en el modelo.

En la aplicación se usará esta metodología por cada componente que exista, por lo que tendrán varios MVPs, y existirá un modelo global MVP, el cual contendrá varios MVPs , que en este caso será el mapa de la red.

Ilustración 2 Metodología MVP (Modelo Vista Presentación)

Para la implementación remota entre diferentes sesiones del simulador se usara .NET remoting junto con el modelo MVP, en donde una sesión del simulador se convertirá en el servidor, y en este estará todo el modelo de red de todas las sesiones las cuales serán los clientes, en el servidor también se encontrará el presentador, y una vista, cada uno las demás sesiones que estén conectadas entre sí, solo serán visibles.

Presentación

Vista

Modelo

Presentación

Vista

Modelo

TCP

HTTP

TCP

HTTP

Host

Host

Ilustración 3. Comunicación Entre Elementos MVP

Mediante este modelo estableceremos la conexión entre diferentes sesiones del simulador, en donde será posible la comunicación entre diferentes componentes de red de las sesiones que estarán contenidas en un mismo modelo de red la cual se encontrará en el servidor.

# Plataforma de desarrollo .Net

La plataforma de desarrollo en la cual se implemento nuestro software es .Net, diseñada por Microsoft. Este ambiente de diseño se escogió por contar con un “Entorno Común de Ejecución para Lenguajes” o CLR (Common Language Runtime), tal como se describe en breve.

## 4.3 .NET FRAMEWORK Y EL ENTORNO COMÚN DE EJECUCIÓN PARA LENGUAJES (CLR)

Uno de los ideales de la ciencia de la computación a través de los últimos años, ha sido el concepto de que un programa de computadora puede ser escrito para que sea ejecutado sin importar el Hardware con el que este compuesto.

Hasta cierto punto este ideal se ha logrado al poder abstraer el acceso al Hardware utilizando un sistema operativo, desde este punto de vista la función básica de un sistema operativo es encapsular la capa de Hardware de un sistema, para que un programador no tenga la necesidad de preocuparse por registros, memoria, y otros problemas que ocurren a nivel de Hardware.

En los últimos años, este problema volvió a surgir, pero esta vez ya no a nivel de Hardware, sino a nivel de un sistema operativo, en el mercado actual se encuentran diferentes sistemas operativos Win32, Solaris, OSs, inclusive de un mismo Proveedor como por ejemplo Microsoft contiene una serie de sistemas operativos diferentes (Win98, WinXp, WinVista). El problema radica ahora en darle al programador, otro nivel de abstracción en donde esté encapsulado el sistema operativo cuando se está desarrollando una aplicación. En muchos sentidos esto es exactamente lo que realiza Microsoft con .NET.

El Entorno Común de Ejecución para Lenguajes o CLR (Common Language Runtime) por sus siglas en ingles, el cual es uno de los pilares del Framework o marco de trabajo de .Net , forma un ambiente que abstrae el sistema operativo.

MARCO DE TRABAJO .NET

SISTEMA OPERATIVO

SISTEMA OPERATIVO

HADWARE

HADWARE

HADWARE

Ilustración 4 Ambiente de Abstracción del Sistema Operativo.

.NET abstrae los sistemas operativos, los cuales abstraen el acceso a Hardware, esto significa que al desarrollar software, se está escribiendo código encaminado al CLR y no directamente al sistema operativo, por lo que este software puede correr sobre diferentes sistemas operativos o cualquier plataforma que implemente el CLR.

Como su nombre lo implica el CLR está diseñado para soportar diferentes lenguajes de programación de forma en común entre ellos. Esto es muy interesante ya que los desarrolladores de software no tienen que aprender un nuevo lenguaje de programación, para trabajar sobre .NET.

Microsoft creó los siguientes lenguajes de programación encaminados para trabajar sobre .NET, los principales son: VB,C#,C++, JScript. Otras firmas están trabajando para desarrollar lenguajes de programación encaminados a .NET como COBOL, Phyton, y posiblemente JAVA.

# 