



Dirección General de Educación Superior Tecnológica

**Tecnológico Nacional de México  
Instituto Tecnológico de Hermosillo**

---

**DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN**

**INGENIERIA EN INFORMÁTICA**



**Reporte Intermedio de Año Sabático  
13 de agosto de 2018 al 12 de enero de 2019**

**MANUAL DE PRACTICAS DE:  
SISTEMAS OPERATIVOS II**

**ELABORADO POR:**

**M.C. DANIEL PEREZ PEREZ  
DICTAMEN: AS-2-106/2018**

**Hermosillo, sonora enero de 2019**

# ÍNDICE

INTRODUCCION .....	1
OBJETIVO GENERAL .....	2
ESQUEMA DE PRACTICAS POR UNIDAD .....	3
UNIDAD 1. SISTEMAS OPERATIVOS EN AMBIENTES DISTRIBUIDOS.	
Práctica # 1.- Sistemas Distribuidos y Sistemas Centralizados, en el mundo real...	5
Práctica # 2.- S.O.'s Distribuidos Vs. Sistemas Operativos convencionales.....	11
Práctica # 3.- S.O.'s Distribuidos a bajo costo.....	14
UNIDAD 2. COMUNICACIÓN EN LOS SISTEMAS OPERATIVOS DISTRIBUIDOS.	
Práctica # 4.- Comunicación Cliente-Servidor.....	19
Práctica # 5.- Llamada a Procedimiento Remoto.....	24
Práctica # 6.- Comunicación en Grupos.....	28
Práctica # 7.- Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.....	34
UNIDAD 5. USOS Y TENDENCIAS DE LOS SISTEMAS OPERATIVOS DISTRIBUIDOS.	
Práctica # 8.- Instalación de Clúster de Alto Rendimiento.....	39
Práctica # 9.- Clúster de Alto rendimiento.....	54
Práctica # 10.- Instalación de Clúster de Alta Disponibilidad.....	58
Práctica # 11.- Clúster de Alta disponibilidad.....	72
Práctica # 12.- Resolución de nombres de dominio.....	82

# MANUAL DE PRÁCTICAS DE SISTEMAS OPERATIVOS II

---

## INTRODUCCIÓN

Esta manual de prácticas aportará al perfil del egresado de la Carrera de Ingeniería en Informática, las competencias necesarias, que le permitan poner en práctica los conocimientos teóricos adquiridos en el aula, en especial lo referente a los Sistemas Operativos en cuanto a su instalación y su manejo. La asignatura Sistemas Operativos II, tiene como requisito la asignatura Sistemas Operativos I que le permite conocer e identificar los fundamentos generales de la materia que abarcan los distintos enfoques del área; Además, esta asignatura dará soporte a otras directamente vinculadas con el desempeño profesional del Ingeniero en Sistemas, como son: Administración de Servidores, Programación en Ambiente Cliente-Servidor.

La asignatura de Sistemas Operativos II contiene 5 unidades, que comprenden temas muy importantes para el alumno; En la primera unidad, el estudiante conocerá los conceptos básicos, características, ventajas y desventajas de los sistemas operativos de Redes, Centralizados y distribuidos. En la segunda unidad, el estudiante conocerá el contexto de la comunicación en los SO considerando la sincronización y la asignación de nombres de recursos en los sistemas operativos distribuidos. En la tercera unidad, conocerá los métodos de asignación de procesos y procesadores en los sistemas operativos distribuidos. En la unidad cuatro, se analizan las diferentes arquitecturas de memoria compartida distribuida y los diferentes modelos de aseguramiento de la consistencia. En la quinta unidad, se abordan los temas de administración de sistemas distribuidos, estándares, tendencias y su aplicación en las diversas áreas de la ciencia.

En las actividades prácticas sugeridas, es conveniente que el docente busque sólo guiar a sus estudiantes para que ellos tomen decisiones relevantes sobre la información que deben utilizar y procesar para llevarlos al resultado deseado.

Hoy en día, los Sistemas Operativos ofrecen tantas opciones y son tan variados que dan a los Ingenieros en Informática la oportunidad de poder seleccionar el más adecuado, de acuerdo a las necesidades que se le presenten y recursos que existan; es por ésta razón que este manual de prácticas adquiere importancia, tanto para los alumnos, como para los maestros.

Nuestros alumnos, deberán estar preparados para llevar a cabo, de la mejor manera, ésta labor.

## Objetivo General

EL objetivo general de este Trabajo es elaborar un manual de prácticas para la materia de Sistemas Operativos II, dirigido a estudiantes de Ingeniería en Informática que permita fomentar en ellos actividades prácticas que promuevan el desarrollo de habilidades tales como: la investigación, auto capacitación, comunicación, trabajo en equipo; así mismo, propicien procesos intelectuales como inducción y deducción y análisis-síntesis. Por esta razón varias de las actividades prácticas se han descrito como actividades previas al tratamiento teórico de los temas, de manera que no sean una mera corroboración de lo visto previamente en clase, sino una oportunidad para conceptualizar a partir de lo observado y los lleve a establecer una propuesta de mejora.

Para el Maestro, que debe actuar como un facilitador, orientador, inductor del aprendizaje, facilitando dicho proceso a través del uso del manual.

El Manual de Prácticas se orienta a desarrollar el aprendizaje significativo en el alumno, a generar un pensamiento crítico y analítico, de abstracción y síntesis, observación y análisis de los problemas propios del campo de estudio, así como, relacionar los contenidos de la asignatura con las demás del plan de estudios a las que da soporte para desarrollar una visión interdisciplinaria en él, que le permita resolver problemas, generar ideas y fomentar la investigación, desarrollando sus competencias profesionales, específicas y genéricas.

El contar con un Manual de prácticas de Sistemas Operativos II, facilitará el seguimiento de esta materia, le permitirá al estudiante concretar los conocimientos teóricos obtenidos, lo vinculará con la problemática en el mercado laboral y estará en condiciones de comparar la teoría con la práctica, lo que lo convertirá en un agente de cambio al utilizar las herramientas adecuadas para la toma de decisiones.

El manual contiene una breve descripción de cada una de las cinco unidades y de los temas que conforman el programa de la materia de Sistemas Operativos II. Al final de cada unidad se incluyen las prácticas programadas para este manual, las cuales se resolverán en forma individual y en trabajo de equipo.

El manual consta de 12 prácticas, cada una de ellas cumple con el formato establecido de 10 puntos: número de práctica, título, objetivo, introducción, correlación con los temas y subtemas del programa de estudio, material y equipo necesario, metodología, sugerencias didácticas, reporte del alumno (resultados) y bibliografía preliminar y requiere un evidente dominio de la competencia a desarrollar por parte del estudiante y la cual el docente facilitador retroalimentará.

## ESQUEMA DE PRÁCTICAS PROPUESTAS POR UNIDAD.

No	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
1	Sistemas Distribuidos y Sistemas Centralizados, en el mundo real.	Realizar visitas a diferentes empresas para conocer el uso de sistemas distribuidos y sistemas centralizados, así como su implementación.	Unidad I. Sistemas Operativos en Ambientes Distribuidos.
2	S.O.'s Distribuidos Vs. Sistemas Operativos Convencionales.	Investigar y comparar los S.O.'s Distribuidos vs S.O.'s Centralizados, S.O.'s para Redes, Modelo Cliente-Servidor y ver las ventajas y desventajas.	Unidad I. Sistemas Operativos en Ambientes Distribuidos.
3	S.O.'s Distribuidos a bajo costo.	Conocer que es un Clúster de computadoras, y tipos que existen.	Unidad I. Sistemas Operativos en Ambientes Distribuidos.
4	Comunicación Cliente-Servidor.	Conocer cómo se da la comunicación cliente-servidor	Unidad II Comunicación en los sistemas operativos distribuidos.
5	Llamada a Procedimiento Remoto.	Conocer cómo se realiza una llamada a procedimiento remoto.	Unidad II Comunicación en los sistemas operativos distribuidos.
6	Comunicación en Grupos.	Conocer las formas en que se puede dar la comunicación en grupos.	Unidad II Comunicación en los sistemas operativos distribuidos.
7	Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.	Conocer las diferentes formas para enviar mensajes en un sistema distribuido.	Unidad II Comunicación en los sistemas operativos distribuidos.
8	Instalación de Clúster de Alto Rendimiento.	Conocer cuáles son los requerimientos y los pasos para instalar un S.O. para implementar un Clúster de Alto Rendimiento.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.

9	Clúster de Alto rendimiento.	Conocer cómo se construye un Clúster de Alto rendimiento.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.
10	Instalación de Clúster de Alta Disponibilidad.	Conocer cuáles son los requerimientos y los pasos para instalar un S.O. para implementar un Clúster de Alta Disponibilidad.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.
11	Clúster de Alta disponibilidad.	Conocer cómo se construye un Clúster de Alta disponibilidad.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.
12	Resolución de nombres de dominio.	Conocer las formas en que trabajan los Servidores de Nombres de Dominio.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.

## **DESARROLLO DE LAS PRÁCTICAS. REQUISITOS.**

No. de la práctica.

Nombre.

Objetivo.

Introducción.

Especificar en cada práctica la correlación con el o los temas y subtemas del programa de estudios vigente.

Material y equipo necesario.

Metodología.

Sugerencias didácticas.

Reporte del alumno (Resultados).

Bibliografía preliminar.

# UNIDAD I. SISTEMAS OPERATIVOS EN AMBIENTES DISTRIBUIDOS. SISTEMAS OPERATIVOS II.

**En el primer tema se abordarán los sistemas operativos en ambientes distribuidos y los centralizados.**

Núm .	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
1	Sistemas Distribuidos y Sistemas centralizados, en el mundo real.	Realizar visita a una empresa para conocer el uso de sistemas distribuidos y sistemas centralizados, así como su implementación.	Unidad I. Sistemas Operativos en ambientes Distribuidos.

## **Las Competencias específicas a desarrollar son:**

Identificar ventajas y desventajas entre los sistemas operativos distribuidos, centralizados y los sistemas operativos de red.

## **Subtemas.**

- 1.1 Conceptos y características de los sistemas operativos de redes y sistemas Operativos centralizados.
- 1.2 Conceptos y características de los sistemas operativos distribuidos.
- 1.3 Sistemas operativos distribuidos: ventajas y desventajas vs. Sistemas Operativos centralizados, sistemas operativos para redes, modelo cliente – servidor, modelo de Ncapas, características del hardware y características del software (homogéneos y heterogéneos), direccionamiento lógico y físico.
- 1.4 Sistemas distribuidos de alto rendimiento a bajo costo (clustering) en sistemas operativos de libre distribución.

# PRÁCTICA No. 1

**NOMBRE DE LA PRÁCTICA:** Sistemas Distribuidos y Sistemas Centralizados, en el mundo real.

**OBJETIVOS:** Hacer una investigación en el ámbito local y/o regional, para identificar empresas donde estén implementados al menos estos dos tipos de sistemas operativos, distribuidos y centralizados; tratar de concertar al menos una visita para observar cómo se han implementado, compararlos e Identificar las ventajas y desventajas de cada uno de ellos.

## INTRODUCCIÓN.

### Sistemas Operativos Centralizados.

Si queremos dar una definición simple de lo que es un sistema operativo centralizado, diremos que se trata de aquel que administra los recursos de una sola computadora, es decir: memoria, CPU, disco y periféricos. Respecto al hardware podemos decir que se suele tratar de un computador caro y de gran potencia, con terminales alfanuméricos directamente conectados. Suele tratarse de una computadora de tipo escritorio, en la que es común encontrar un monitor grande con un teclado y un ratón; además de un armazón para albergar la unidad de procesamiento y los demás componentes. Podemos encontrar este tipo de sistemas operativos en un entorno de empresa, donde puede haber un soporte multiusuario, mismas que utilizan una mainframe potente para dar capacidad de cómputo a muchas terminales, o también se puede encontrar a empresas con muchas minicomputadoras para los empleados que las necesiten en sus actividades. Uno de los primeros modelos de computadoras interconectadas fue el centralizado, donde todo el procesamiento de la organización se llevaba a cabo en una sola computadora, normalmente un Mainframe, y los usuarios empleaban computadoras personales para conectarse a ella; Los problemas de este modelo son que cuando la carga de procesamiento aumentaba se tenía que cambiar el hardware del Mainframe, lo cual es más costoso que añadir más computadores personales clientes o servidores que aumenten las capacidades. El otro problema que surgió son las modernas interfaces gráficas de usuario, las cuales podían conllevar a un gran aumento de tráfico en los medios de comunicación y por consiguiente podían colapsar.

Otro entorno donde se encuentran sistemas operativos de arquitectura centralizada es el científico, en ellos se busca la ejecución eficiente de aplicaciones y el uso de súper computadores, que son computadoras con capacidades de cálculo muy superiores a las comúnmente disponibles de las máquinas de escritorio. Se suele usar este tipo de máquinas para cálculos en los que intervienen una gran cantidad de operaciones complejas y muchos factores.

En un entorno de hogar se puede encontrar también una computadora; Estas tienen un sistema operativo centralizado porque el sistema es único y no necesita trabajar en paralelo con ninguna otra, ya que no se encuentra conectado a ninguna. Normalmente estos sistemas tienen uno o dos procesadores potentes y caros que satisfacen las necesidades de cómputo del usuario y hay un uso ocasional de la red, como para transferir archivos o hacer accesos remotos.

En la actualidad prácticamente todos los sistemas operativos permiten la transferencia de archivos, uno puede conectarse con una computadora en la misma red y acceder a los documentos que ésta esté dispuesta a compartir por orden del usuario o viceversa. Pero no se trata de una transferencia realmente transparente, pues el usuario está consciente de que está accediendo archivos almacenados en un disco diferente, al que forma parte de su computadora. Asimismo, es posible conectarse de forma remota a otra computadora como en el caso de la asistencia remota, pero son más que nada utilidades o funciones agregadas que permite realizar el sistema operativo centralizado, sin llegar a ser lo que buscaba como objetivo principal el sistema al ser diseñado. Son muy conocidos los sistemas centralizados con los que contamos en la actualidad, basta con empezar por los que tenemos instalados en nuestras propias computadoras como: Windows, Linux, Mac OS, Unix, etc.

### **Sistemas Operativos Distribuidos.**

Según el autor Tanenbaum, un sistema distribuido es "una colección de computadoras independientes que aparecen ante los usuarios del sistema como una única computadora". De eso podemos entender que las máquinas son autónomas y los usuarios siempre piensan que el sistema es como una única computadora. Un sistema distribuido se caracteriza por comportarse frente al usuario como una sola máquina; mismo que desconoce sobre qué procesador se están ejecutando sus procesos y dónde residen sus archivos; vamos a dar un ejemplo, primero consideremos una red de estaciones de trabajo en un departamento de una compañía, además de ellas, podría existir una pila de procesadores en el cuarto de máquinas que no estén asignados a usuarios específicos, sino que se utilicen de manera dinámica conforme sea necesario. Tal vez el sistema podría tener un sistema de archivos único con todos los archivos accesibles desde todas las máquinas de la misma forma y con el mismo nombre de ruta de acceso. Además, cuando el usuario escriba un comando el sistema podría buscar el mejor lugar para ejecutarlo, tal vez en la propia estación de trabajo del usuario o en una estación de trabajo inactiva que pertenezca a otra persona o en uno de los procesadores no asignados en el cuarto de máquinas. Si el sistema se ve como un todo y actúa como un sistema de tiempo compartido clásico con un único procesador pondría considerarse como un sistema distribuido.

Otro ejemplo podría ser el de un gran banco con cientos de sucursales por todo el mundo, cada oficina tiene una computadora maestra para guardar las cuentas locales y el manejo de las transacciones locales; además, cada computadora tiene la capacidad de comunicarse con las otras sucursales y con una computadora central en las oficinas centrales. Si las transacciones se pueden realizar sin importar dónde se encuentre el cliente o la cuenta y los usuarios no observan diferencia alguna entre este sistema y el antiguo centralizado que ha reemplazado también se podría considerar como un sistema distribuido.

Los sistemas operativos de red están formados por un software débilmente acoplado en un hardware débilmente acoplado, de no ser por el sistema compartido de archivos, a los usuarios les parecería que el sistema consta de varias computadoras, puede ejecutar su propio sistema operativo y hacer lo que quiera.

El siguiente paso en la evolución es el del software fuertemente acoplado en hardware débilmente acoplado, es donde aparecen los sistemas distribuidos. El objetivo es crear la ilusión en las mentes de los usuarios que toda la red de computadoras es un sistema de tiempo compartido, en vez de máquinas diversas. Debe haber un mecanismo de comunicación global entre los procesos, de forma que cualquiera de ellos pueda comunicarse con cualquier otro; también un sistema global de protección. La administración de procesos debe ser la misma en todas partes, la forma en que se crean, destruyen y detienen los procesos no debe variar de una máquina a otra; de la misma forma el sistema de archivos debe tener la misma apariencia en todas partes. Como consecuencia lógica del hecho de tener una misma interfaz de llamadas al sistema en todas partes, es normal que se ejecuten núcleos idénticos en todos los CPU de sistema, eso facilita la coordinación de las actividades globales. Por ejemplo, cuando se inicia un proceso, todos los núcleos deben cooperar en la búsqueda del mejor lugar para ejecutarlo, los sistemas distribuidos se basan en la utilización de sistemas de transmisión fiables, eficaces, rápidos que permitan integrar sistemas de distintos fabricantes.

## **MATERIAL Y EQUIPO NECESARIO.**

- Computadora.
- Servicio de Internet.
- Bibliografía sugerida.
- Proyector.

## **METODOLOGÍA.**

- a) El alumno deberá de Investigar los conceptos de: Sistemas Distribuidos y Sistemas Centralizados, en al menos tres fuentes de información, tomando en cuenta la bibliografía recomendada, referencias WEB y entrevistas con Maestros y/o profesionistas del ramo.
- b) Los alumnos y el Maestro, llevarán a cabo una investigación y búsqueda de empresas donde puedan estar implementados este tipo de sistemas operativos.
- c) El Maestro solicitará al menos una visita para ir a conocer los sistemas de la empresa que se haya considerado como la mejor para ello, dependiendo de la viabilidad para lograr concertar la visita, se realizará ésta.
- d) Durante la visita los alumnos y el Maestro deberán llevar a cabo un análisis de los sistemas que estén implementados en la empresa visitadas, tomando en consideración una serie de puntos que previamente serán definidos para tal fin.
- e) Los alumnos deberán presentar un informe individual escrito de al menos cuatro cuartillas, incluyendo la portada, como trabajo individual y para discusión en el grupo, donde se identifiquen las ventajas y desventajas de los sistemas operativos vistos en la empresa visitada y algunas anotaciones que se consideren importantes.
- f) Se hará un foro de discusión en el aula, para poder conocer los resultados de las investigaciones; considerando todas las aportaciones, con la ayuda del docente, se redactarán las conclusiones a que se llegue.

### **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

- Explicar que es un Sistema Operativo Centralizado.
- Explicar que es un Sistema Operativo Distribuido.
- Buscar y establecer las diferencias entre ellos.
- Identificar Ventajas y Desventajas de cada uno de ellos.

## SUGERENCIAS DIDACTICAS:

Propiciar actividades de búsqueda, selección y análisis de información en distintas fuentes, para ubicar y localizar las posibles empresas a visitar donde estén implementados estos dos tipos de sistemas.

Llevar a cabo la búsqueda de los conceptos de sistemas operativos centralizados y distribuidos, en al menos tres fuentes; definirlos y diferenciarlos.

Fomentar actividades grupales que propicien la comunicación, el intercambio de ideas con argumentación, la reflexión, la integración y la colaboración entre los estudiantes; así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

## REPORTE DEL ALUMNO (Resultados):

**Presentación de trabajo:** Sistemas Distribuidos y Sistemas Centralizados, en el mundo real.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

## Bibliografía preliminar:

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- [https://es.wikipedia.org/wiki/Computaci%C3%B3n\\_distribuida](https://es.wikipedia.org/wiki/Computaci%C3%B3n_distribuida)
- 6.- [https://es.wikipedia.org/wiki/Computaci%C3%B3n\\_centralizada](https://es.wikipedia.org/wiki/Computaci%C3%B3n_centralizada)
- 7.- [https://es.wikipedia.org/wiki/Sistema\\_operativo\\_distribuido](https://es.wikipedia.org/wiki/Sistema_operativo_distribuido)
- 8.- [http://es.so2.wikia.com/wiki/Sistemas\\_Operativos\\_Centralizados](http://es.so2.wikia.com/wiki/Sistemas_Operativos_Centralizados)

[Sistemas Distribuidos vs Sistemas Centralizados...](#)

# UNIDAD I. SISTEMAS OPERATIVOS EN AMBIENTES DISTRIBUIDOS. SISTEMAS OPERATIVOS II.

**En la segunda práctica se compararán los Sistemas Operativos Distribuidos vs. Sistemas Operativos Centralizados.**

Núm.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
2	S.O.'s Distribuidos Vs. Sistemas Operativos Convencionales.	Investigar y comparar los S.O.'s Distribuidos vs S.O.'s Centralizados, S.O.'s para Redes, Modelo Cliente-Servidor y ver las ventajas y desventajas.	Unidad I. Sistemas Operativos en Ambientes Distribuidos

## **Las Competencias específicas a desarrollar son:**

Identificar ventajas y desventajas entre los sistemas operativos distribuidos, centralizados y los sistemas operativos de red.

## Subtemas.

- 1.1 Conceptos y características de los sistemas operativos de redes y sistemas Operativos centralizados.
- 1.2 Conceptos y características de los sistemas operativos distribuidos.
- 1.3 Sistemas operativos distribuidos: ventajas y desventajas vs. Sistemas Operativos centralizados, sistemas operativos para redes, modelo cliente – servidor, modelo de Ncapas, características del hardware y características del software (homogéneos y heterogéneos), direccionamiento lógico y físico.
- 1.4 Sistemas distribuidos de alto rendimiento a bajo costo (clustering) en sistemas operativos de libre distribución.

# PRÁCTICA No. 2

**NOMBRE DE LA PRÁCTICA:** S.O.'s Distribuidos Vs. Sistemas Operativos Convencionales.

**OBJETIVOS:** Hacer una investigación para identificar los conceptos de los siguientes Sistemas Operativos: Distribuidos, para Redes, Modelo Cliente-Servidor; compararlos e Identificar las ventajas y desventajas de cada uno de ellos.

## INTRODUCCIÓN.

Siempre que sea posible ofrecerles varias opciones a nuestros usuarios o clientes, en relación a diferentes plataformas que se pueden implementar en los equipos de cómputo, será importante conocer las ventajas y desventajas de dichas opciones, así como, los requerimientos para cada uno de ellos. Es por ello que ésta segunda práctica es muy relevante y consiste en llevar a cabo una investigación para identificar los conceptos de los siguientes tipos de Sistemas Operativos: Distribuidos, para Redes, Modelo Cliente-Servidor; compararlos e Identificar las ventajas y desventajas de cada uno de ellos.

## MATERIAL Y EQUIPO NECESARIO.

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## METODOLOGÍA.

- a) El alumno deberá de Investigar los conceptos de: Sistemas Operativos Distribuidos, para Redes y Modelo Cliente-Servidor, en al menos tres fuentes de información, tomando en cuenta la bibliografía recomendada, referencias WEB y entrevistas con Maestros y/o profesionistas del ramo.
- b) El alumno, en forma individual, hará una investigación para conocer, definir y diferenciar estos tipos de sistemas operativos.
- b) El alumno definirá para cada uno de los tipos de sistemas operativos, cuales son la ventajas y desventajas, comparativamente, así como los requisitos que deben de cumplir para poder instalarlos de manera correcta.
- c) El resultado de este trabajo deberá presentarse en forma de un reporte, de al menos cinco cuartillas, incluyendo la portada.
- e) Se hará un foro de discusión en el aula, para poder conocer los resultados de las investigaciones; considerando todas las aportaciones, con la ayuda del docente, se redactarán las conclusiones a que se llegue.

### **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

- Explicar que es un Sistema Operativo de Red.
- Explicar que es un Sistema Operativo de Modelo Cliente-Servidor.
- Explicar que es un Sistema Operativo Centralizado.
- Explicar que es un Sistema Operativo Distribuido.
- Buscar y establecer las diferencias entre ellos.

### **SUGERENCIAS DIDACTICAS:**

Se debe de propiciar y fomentar, el uso de las nuevas tecnologías en el desarrollo de las prácticas, tanto en actividades grupales como individuales, de búsqueda, selección y análisis de información en distintas fuentes, que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración, el uso adecuado de conceptos, terminología científico-tecnológica y la colaboración de y entre los estudiantes; así mismo, el maestro facilitador deberá retroalimentar sobre la participación del alumno y proponerle problemas que le permitan la integración de contenidos de la asignatura en relación con otras, propiciando una visión interdisciplinaria, que les permitan elaborar desarrollar materiales didácticos sobre la materia.

### **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** S.O.'s Distribuidos Vs. Sistemas Operativos Convencionales.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

### **Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- [https://es.wikipedia.org/wiki/Sistema\\_operativo\\_distribuido](https://es.wikipedia.org/wiki/Sistema_operativo_distribuido)
- 6.- [https://es.wikipedia.org/wiki/Computaci%C3%B3n\\_distribuida](https://es.wikipedia.org/wiki/Computaci%C3%B3n_distribuida)
- 7.- [https://es.wikipedia.org/wiki/Sistema\\_operativo\\_de\\_red](https://es.wikipedia.org/wiki/Sistema_operativo_de_red)
- 8.- <https://es.wikipedia.org/wiki/Cliente-servidor>
- 9.- [http://es.so2.wikia.com/wiki/Sistemas\\_Operativos\\_Centralizados](http://es.so2.wikia.com/wiki/Sistemas_Operativos_Centralizados)

[Sistema Operativo Distribuido...](#)

# UNIDAD I. SISTEMAS OPERATIVOS EN AMBIENTES DISTRIBUIDOS. SISTEMAS OPERATIVOS II.

**En esta tercera práctica se abordarán temas sobre los Sistemas Operativos Distribuidos a bajo costo.**

Núm.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
3	S.O.'s Distribuidos a bajo costo.	Conocer que es un Clúster de computadoras, y los tipos que existen.	Unidad I. Sistemas Operativos en Ambientes Distribuidos.

## **Las Competencias específicas a desarrollar son:**

Identificar ventajas y desventajas entre los sistemas operativos distribuidos, y los sistemas operativos de red.

### **Subtemas.**

- 1.1 Conceptos y características de los sistemas operativos de redes y sistemas Operativos centralizados.
- 1.2 Conceptos y características de los sistemas operativos distribuidos.
- 1.3 Sistemas operativos distribuidos: ventajas y desventajas contra S. Operativos centralizados, sistemas operativos para redes, modelo cliente–servidor, modelo de Ncapas, características del hardware y características del software (homogéneos y heterogéneos), direccionamiento lógico y físico.
- 1.4 Sistemas distribuidos de alto rendimiento a bajo costo (clustering) en sistemas operativos de libre distribución.

# PRÁCTICA No. 3

**NOMBRE DE LA PRÁCTICA:** S.O.'s Distribuidos a bajo costo.

**OBJETIVOS:** Conocer que es un Clúster de computadoras, y tipos que existen.

## INTRODUCCIÓN.

En el ámbito de los Sistemas Operativos, es muy importante que las implementaciones de los sistemas sean de bajo costo; es por ello que el hecho de poder agrupar un conjunto de equipos de cómputo, heterogéneos u homogéneos, que puedan ser manipulados como uno solo y a bajo costo, también lo es. Los diferentes tipos de Clúster que existen son: de Alto Rendimiento, de Alta Disponibilidad y de Alta Eficiencia, por lo que es necesario conocerlos y saber cómo se implementan.

Un *clúster* de computadoras, también conocido como cúmulo o granja, lo podemos definir como un sistema de procesamiento paralelo o distribuido. Consta de un conjunto de computadoras independientes, interconectadas entre sí, de tal manera que funcionan como un solo recurso computacional. A cada uno de los elementos del *clúster* se le conoce como nodo. Estos pueden tener uno o varios procesadores, memoria volátil (*RAM, Random Access Memory*), interfaces de red, dispositivos de entrada y salida, y sistema operativo. Los nodos pueden estar contenidos e interconectados en un solo gabinete, o, como en muchos casos, acoplados a través de una red de área local (*LAN, Local Área Network*). Otro componente básico en un *clúster* es la interfaz de la red, la cual es responsable de transmitir y recibir los paquetes de datos, que viajan a través de la red entre los nodos. Finalmente, el lograr que todos estos elementos funcionen como un solo sistema, es la meta a la que se quiere llegar para dar origen a un *clúster*.

Comúnmente, en un *clúster* existe una máquina (con monitor, teclado, ratón, etcétera) que funciona como nodo-maestro y se encarga de administrar, controlar y monitorear todas las aplicaciones y recursos del sistema, en tanto que el resto de los nodos está dedicado al procesamiento de datos o a ejecutar operaciones aritméticas. Se les conoce como nodos-esclavos. Ver. Fig. 1.

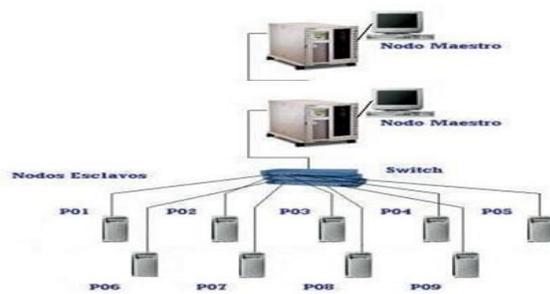


Figura 1. Esquema general de un cluster. En la figura podemos ver la distribución de las partes principales de un cúmulo de computadoras: nodo maestro, nodos esclavos, un switch y una red.

## Cluster.

El concepto clustering se refiere a una técnica que permite combinar múltiples sistemas para que trabajen en paralelo y se comporten como un recurso informático unificado para: servir a un grupo de tareas, proporcionar tolerancia a fallos y tener disponibilidad continua. Por ejemplo, en el caso de usuarios de Internet, el clustering proporciona bases de datos, correo electrónico, archivos u otros servicios de sistema sin interrupciones. Si se presentara una falla dentro de una red de servidores de un clúster, ésta se corregiría inmediatamente sin que los usuarios lo notaran. Dentro de esta técnica existen una serie de conceptos fundamentales que se describen a continuación. Comencemos por explicar el concepto de paralelismo, que consiste en el procesamiento de una serie de instrucciones de un programa, que son ejecutadas por múltiples procesadores que trabajan de manera independiente. El paralelismo puede manejarse en dos niveles: paralelismo de hardware y de software. El primero depende básicamente de la tecnología de cómputo disponible, mientras el segundo se refiere a la habilidad del usuario para encontrar áreas bien definidas del problema por resolver, de tal forma que éste pueda ser dividido en partes autónomas que serán distribuidas entre los nodos del clúster, obteniendo un sistema de alto rendimiento computacional.

Por otro lado, está el concepto de multiprocesamiento, una característica del sistema operativo que controla el hardware. El software asegura la interacción entre los procesadores a nivel de carga y descarga de datos, además de realizar el despacho de trabajos en forma múltiple, independiente y simultánea. Otro concepto fundamental es la programación de hilos (programming threads). Un hilo (thread) es una secuencia de instrucciones ejecutables que pueden ejecutarse independientemente, compartiendo recursos computacionales con otros hilos; Si en un programa existe la posibilidad de ejecutar varios hilos simultáneamente, todos los hilos activos pueden competir por, y compartir los, recursos del sistema, por lo mismo, el programador puede recurrir a la programación multihilos (multithread) que traerá como consecuencia la concurrencia entre procesos y tiene una gran importancia en el cómputo paralelo.

### **Clasificación de los clústers.**

- HPCC (*High Performance Computing Clusters*: clúster de alto rendimiento).
- HA o HACC (*High Availability Computing Clusters*: clúster de alta disponibilidad).
- HT o HTCC (*High Throughput Computing Clusters*: clúster de alta eficiencia).

Alto rendimiento: es un clúster en el cual se ejecutan tareas que requieren de gran capacidad computacional, grandes cantidades de memoria, o ambos a la vez. El llevar a cabo estas tareas puede comprometer los recursos del clúster por largos periodos de tiempo.

Alta disponibilidad: es un clúster cuyo objetivo de diseño es el de proveer disponibilidad y confiabilidad, tratando de brindar la máxima disponibilidad de los servicios que ofrecen; La confiabilidad se provee mediante software que detecta fallos y le permite recuperarse frente a los mismos, mientras que en hardware se evita tener un único punto de fallos.

Alta eficiencia: clúster cuyo objetivo de diseño es el ejecutar la mayor cantidad de tareas en el menor tiempo posible. Existe independencia de datos entre las tareas individuales. El retardo entre los nodos del clúster no es considerado un gran problema.

Componentes de un Clúster:

- nodos.
- almacenamiento.
- sistemas operativos.
- conexiones de red.
- middleware.
- protocolos de comunicación y servicios.
- aplicaciones.
- ambientes de programación paralela.

## **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## **METODOLOGÍA.**

El alumno, en equipos de tres integrantes:

- a) llevará a cabo una investigación, en al menos tres fuentes de información, tomando en cuenta la bibliografía recomendada, referencias WEB y entrevistas con Maestros y/o profesionistas del ramo, que les permita identificar el concepto de Clúster y con ello los diferentes tipos que existen: Homogéneos, Heterogéneos, de Alto Rendimiento, de Alta Disponibilidad y de Alta Eficiencia.
- b) Investigará cuales son los requerimientos de instalación, así como las ventajas y desventajas, para los clústers heterogéneos.
- c) Investigará cuales son los requerimientos de instalación, así como las ventajas y desventajas, para los clústers homogéneos.
- d) Los resultados de estos trabajos, deberán presentarse en forma de un reporte escrito de no más de 6 cuartillas, incluyendo la portada.
- e) Se hará un foro de discusión en el aula, para poder conocer y comparar los resultados de las investigaciones; considerando todas las aportaciones, con la ayuda y conducción del docente, se redactarán las conclusiones a que se llegue.

## **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

- Explicar que es un Clúster.
- Explicar que es un Clúster Homogéneo.
- Explicar que es un Clúster Heterogéneo.
- Explicar que es un Clúster de Alta Eficiencia.
- Explicar que es un Clúster de Alta Disponibilidad.
- Explicar que es un Clúster de Alta Rendimiento.

## **SUGERENCIAS DIDACTICAS:**

Propiciar y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

## **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** S.O.'s Distribuidos a bajo costo.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

## **Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- [https://es.wikipedia.org/wiki/Cl%C3%BAster\\_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Cl%C3%BAster_(inform%C3%A1tica))
- 6.- <http://clusterfie.epn.edu.ec/clusters/Definiciones/definiciones.html>
- 7.- [https://es.wikipedia.org/wiki/Cl%C3%BAster\\_de\\_alta\\_disponibilidad](https://es.wikipedia.org/wiki/Cl%C3%BAster_de_alta_disponibilidad)
- 8.- <http://www.rocksclusters.org/assets/usersguides/spanish/cluster-practical-guide-rel1.pdf>

[Clústeres y Granja de Servidores...](#)

# UNIDAD II. COMUNICACIÓN EN LOS SISTEMAS OPERATIVOS DISTRIBUIDOS.

## SISTEMAS OPERATIVOS II.

En la cuarta práctica se verá lo relacionado con la Comunicación Cliente-Servidor.

Núm.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
4	Comunicación Cliente-Servidor.	Conocer cómo se da la comunicación cliente-servidor.	Unidad II Comunicación en los sistemas operativos distribuidos.

### Las Competencias específicas a desarrollar son:

Establecer la comunicación y la sincronización entre los nodos de un sistema distribuido, conocer y manejar todos los elementos utilizados durante el proceso de resolución y distribución de nombres.

#### Subtemas.

- 2.1 Comunicación: comunicación con cliente –servidor, comunicación con llamada a procedimiento remoto, comunicación en grupo, tolerancia a fallos.
- 2.2 Sincronización: relojes físicos, relojes lógicos, usos de la sincronización.
- 2.3 Nominación: características y estructuras, tipos de nombres, resolución y distribución, servidores y agentes de nombres, mapeo de direcciones, mapeo de rutas, modelo de Terry.
- 2.4 Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.

# PRÁCTICA No. 4

**NOMBRE DE LA PRÁCTICA:** Comunicación Cliente-Servidor.

**OBJETIVOS:** Conocer cómo se da la comunicación cliente-servidor.

## INTRODUCCIÓN.

Uno de los sistemas más conocido y ampliamente adoptado en la actualidad, es el modelo cliente-servidor. En éste modelo, todos los recursos compartidos son tomados y manejados por los procesos servidores. Los procesos clientes realizan peticiones a los servidores cuando necesitan acceder a algún recurso, si la petición es válida, entonces el servidor lleva a cabo la acción requerida y envía una respuesta al proceso cliente. El modelo cliente-servidor nos da un enfoque efectivo y de propósito general para la compartición de datos y recursos en los sistemas distribuidos. El modelo puede ser implementado en una gran variedad de entornos de hardware y software; los sistemas que ejecuten los programas clientes y servidores pueden ser de muchos tipos y no existe la necesidad de distinguir entre ellos, los procesos clientes y servidores pueden, incluso, residir en la misma máquina.

En esta visión simple del modelo cliente-servidor, cada proceso servidor podría ser visto como un proveedor centralizado de los recursos que maneja, aunque no es deseable en los sistemas distribuidos. Es por esta razón por lo que se hace una distinción entre los servicios proporcionados a los clientes y los servidores encargados de proveer dichos servicios. Se considera un servicio como una entidad abstracta que puede ser provista por varios procesos servidores ejecutándose en computadoras separadas y cooperando vía red.

El modelo cliente-servidor se ha extendido y utilizado en los sistemas actuales con servicios, manejando diferentes tipos de recursos compartidos tales como: correo electrónico y mensajes de noticias, archivos, sincronización de relojes, almacenamiento en disco, impresoras, comunicaciones de área extensa, e incluso las interfaces gráficas de usuario. Pero no es posible que todos los recursos que existen en un sistema distribuido sean manejados y compartidos de esta manera; algunos tipos de recursos deben permanecer locales a cada computadora de cara a una mayor eficiencia, por ejemplo: la memoria de acceso aleatoria, el procesador y la interfaz de red local; Estos recursos clave son manejados separadamente por un sistema operativo en cada máquina; solo podrían ser compartidos entre procesos localizados en él mismo.

¿Qué es la arquitectura cliente-servidor?

La arquitectura cliente-servidor es un paradigma de aplicación distribuida en el que un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Cliente y servidor suelen estar en computadoras diferentes, aunque también pueden ejecutarse ambos programas sobre una sola computadora.

Veamos las diferentes características del cliente y el servidor:

-Servidor: es el receptor de la solicitud enviada por el cliente. Al iniciarse quedan a la espera de que lleguen solicitudes de los clientes. Tras recibir una solicitud, la procesan y luego envían la respuesta al cliente. Generalmente aceptan conexiones de un gran número de clientes, aunque el número máximo de peticiones simultáneas puede estar limitado.

-Cliente: es el remitente de una solicitud al servidor, quien inicia la comunicación. Una vez enviada la petición, queda a la espera hasta recibir la respuesta del servidor. Generalmente puede conectarse a múltiples servidores a la vez.

En la arquitectura cliente-servidor los roles de cliente y servidor están bien diferenciados y no son intercambiables, a diferencia de otros modelos como el peer to peer (P2P) en el que los roles no están definidos y se puede ser tanto cliente como servidor, incluso simultáneamente.

Respecto a las ventajas del modelo cliente-servidor, las principales son las siguientes:

-Recursos centralizados: en el servidor se centraliza el control del sistema, los accesos, recursos, actualizaciones y la integridad de los datos, de tal manera que un programa cliente defectuoso no pueda dañar el sistema.

-Seguridad mejorada: las tecnologías diseñadas para el paradigma cliente-servidor aseguran la seguridad en las transacciones, una interfaz amigable y la sencillez de empleo.

-Fácil administración: gracias a la encapsulación de funciones en los servidores, generalmente es posible reemplazar, reparar, actualizar o incluso trasladar un servidor, sin que los clientes se vean afectados por el cambio, o sean afectados mínimamente.

-Escalabilidad: en cualquier momento puede ser mejorado cualquier elemento o se pueden añadir nuevos nodos a la red, tanto clientes como servidores, sin que esto afecte al funcionamiento de la red.

En cuanto a las desventajas de la arquitectura cliente-servidor, podemos citar:

-Costo elevado: dada la complejidad técnica del servidor, tanto a nivel de software como de hardware, el coste se ve incrementado.

-Debilidad: el paradigma cliente-servidor presenta la desventaja de que cuando un servidor está caído, las peticiones de los clientes no pueden ser satisfechas. Para evitar esta situación, se implementan sistemas de servidores redundantes balanceados que puedan asumir la falla de alguno de ellos manteniendo la integridad del sistema.

-Tráfico excesivo: la demanda excesiva por parte de clientes al mismo servidor puede congestionar la red y causar problemas al servidor.

El modelo cliente-servidor es recomendado para redes que necesiten fiabilidad, seguridad, escalabilidad y fácil administración, por lo que se utiliza habitualmente en servidores web, servidores de archivos y servidores de correo electrónico, ya que todos ellos básicamente reciben solicitudes de información y envían los datos requeridos a múltiples clientes.

## **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## **METODOLOGÍA.**

- a) El alumno en forma individual, llevará a cabo una investigación, en al menos tres fuentes de información, tomando en cuenta la bibliografía recomendada, referencias WEB y entrevistas con Maestros y/o profesionistas del ramo, que le permita identificar la forma en que se establecen las Comunicaciones en un sistema con el Modelo Cliente-Servidor y los diferentes tipos de comunicación que existen.
- b) El alumno, en equipos de tres integrantes, investigará cuales son los requerimientos de instalación, así como las ventajas y desventajas que tiene un sistema con el modelo Cliente-Servidor.
- c) Los resultados de estos trabajos, deberán presentarse en forma de un reporte escrito de no más de 4 cuartillas, incluyendo la portada.
- d) Se hará un foro de discusión en el aula, para poder conocer los resultados de las investigaciones; considerando todas las aportaciones, con la ayuda y conducción del maestro, se redactarán las conclusiones a que se llegue.

## **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

¿Qué es un Sistema Operativo de Modelo Cliente-Servidor?

## **SUGERENCIAS DIDACTICAS:**

Propiciar y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación de los alumnos.

## **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** Comunicación Cliente-Servidor.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

## **Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- <https://es.wikipedia.org/wiki/Cliente-servidor>
- 6.- <http://redestelematicas.com/el-modelo-cliente-servidor-en-las-redes-de-datos/>

[Que es el modelo Cliente/Servidor...?](#)

# UNIDAD II. COMUNICACIÓN EN LOS SISTEMAS OPERATIVOS DISTRIBUIDOS.

## SISTEMAS OPERATIVOS II.

**En el quinto tema se abordarán las llamadas a procedimientos remotos.**

Núm .	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
5	Llamada a Procedimiento Remoto.	Conocer cómo se realiza una llamada a procedimiento remoto.	Unidad II Comunicación en los Sistemas Operativos distribuidos.

### **Las Competencias específicas a desarrollar son:**

Establecer la comunicación y la sincronización entre los nodos de un sistema distribuido, conocer y manejar todos los elementos utilizados durante el proceso de resolución y distribución de nombres.

### **Subtemas.**

- 2.1 Comunicación: comunicación con cliente –servidor, comunicación con llamada a procedimiento remoto, comunicación en grupo, tolerancia a fallos.
- 2.2 Sincronización: relojes físicos, relojes lógicos, usos de la sincronización.
- 2.3 Nominación: características y estructuras, tipos de nombres, resolución y distribución, servidores y agentes de nombres, mapeo de direcciones, mapeo de rutas, modelo de Terry.
- 2.4 Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.

# PRÁCTICA No. 5

**NOMBRE DE LA PRÁCTICA:** Llamada a Procedimiento Remoto.

**OBJETIVOS:** Conocer cómo se realiza una llamada a procedimiento remoto.

## INTRODUCCIÓN.

Un tema primordial en el ámbito de los Sistemas Operativos, es el de la comunicación entre ellos; tener la posibilidad de poder ejecutar un programa o aplicación de manera remota es muy importante, ya que ello conlleva el hecho de no tener que utilizar el equipo local, por no contar con él en ese momento o por no tener disponible la aplicación en nuestro equipo.

El RPC (del inglés Remote Procedure Call, Llamada a Procedimiento Remoto) es un protocolo que permite a un programa de computadora ejecutar código en una máquina remota sin tener que preocuparse por las comunicaciones entre ambas. El protocolo es un gran avance sobre los sockets utilizados hasta el momento. De esta manera el programador no tiene que estar pendiente de las comunicaciones, estando éstas encapsuladas dentro de las RPC.

Las llamadas a procedimiento remoto están implementadas mediante varios tipos de protocolos, muchos de ellos estandarizados como pueden ser:

- RPC de Sun Microsystems nombrado también ONC RPC<sup>1</sup> (Open Network Computing Remote Procedure Call).
- RPC de Open Software Foundation (OSF) denominado DCE/RPC<sup>2</sup>.
- DCOM (Distributed Component Object Model, Modelo de Objetos de Componentes Distribuidos) de Microsoft.

Aunque ninguno de estos es compatible entre sí, la mayoría de ellos utilizan un lenguaje de descripción de interfaz o IDL (Interface Description Language) que define los métodos exportados por el servidor.

Hoy en día se está utilizando el XML<sup>3</sup> como lenguaje para definir el IDL y el HTTP<sup>4</sup> como protocolo de red, dando lugar a lo que se conoce como servicios web. Ejemplos de estos pueden ser SOAP<sup>5</sup> o XML-RPC<sup>6</sup>.

<sup>1</sup> ONC RPC, abreviación del inglés *Open Network Computing Remote Procedure Call*, es un protocolo de llamada a procedimiento remoto desarrollado por el grupo ONC de Sun Microsystems como parte del proyecto de su sistema de archivos de Red NFS, algunas veces se le denomina Sun ONC o Sun RPC. Trabaja sobre los protocolos TCP y UDP. La codificación de datos se realiza utilizando el protocolo XDR (presentación de datos).

<sup>2</sup> DCE Remote Procedure Call o bien DCE RPC es un sistema de llamada a procedimiento remoto del conjunto de software OSF DCE. DCE / RPC, la abreviatura de "Distributed Computing Environment / Remote Procedure Calls", es el sistema de llamada a procedimiento remoto desarrollado para el entorno de la informática distribuida (DCE). Este sistema permite a los programadores escribir software distribuido como si fuera todos los que trabajan en el mismo equipo, sin tener que preocuparse por el código de red subyacente.

<sup>3</sup> XML, siglas en inglés de *eXtensible Markup Language*, traducido como "Lenguaje de Marcado Extensible" o "Lenguaje de Marcas Extensible", es un meta-lenguaje que permite definir lenguajes de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible.

<sup>4</sup> El Protocolo de transferencia de hipertexto (en inglés: Hypertext Transfer Protocol o HTTP) es el protocolo de comunicación que permite las transferencias de información en la World Wide Web.

<sup>5</sup> SOAP (originalmente las siglas de *Simple Object Access Protocol*) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Este protocolo deriva de un protocolo creado por Dave Winer en 1998, llamado XML-RPC. SOAP fue creado por Microsoft IBM y otros. Está actualmente bajo el auspicio de la W3C. Es uno de los protocolos utilizados en los servicios Web.

<sup>6</sup> XML-RPC es un protocolo de llamada a procedimiento remoto que usa XML para codificar los datos y HTTP como protocolo de transmisión de mensajes.

Es un protocolo muy simple ya que solo define unos cuantos tipos de datos y comandos útiles, además de una descripción completa de corta extensión. La simplicidad del XML-RPC contrasta con la mayoría de protocolos RPC que tiene una documentación extensa y requiere considerable soporte de software para su uso.

Fue creado por Dave Winer de la empresa UserLand Software en asociación con Microsoft en el año 1998. Al considerar Microsoft que era muy simple decidió añadirle funcionalidades, tras las cuales, después de varias etapas de desarrollo, el estándar dejó de ser sencillo y se convirtió en lo que es actualmente conocido como SOAP. Una diferencia fundamental es que en los procedimientos en SOAP los parámetros tienen nombre y no interesa su orden, no siendo así en XML-RPC.

## **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## **METODOLOGÍA.**

a) El alumno en forma individual, llevará a cabo una investigación bibliográfica, en al menos tres fuentes de información, tomando en cuenta la bibliografía recomendada, referencias WEB y entrevistas con Maestros y/o profesionistas del ramo, que le permita identificar el concepto de RPC (Remote Procedure Call o Llamada a Procedimiento Remoto).

b) El alumno, en equipos de tres integrantes, investigará cuales son los Sistemas Operativos que permiten llevar a cabo este tipo de llamadas.

c) El alumno, en equipos de tres integrantes, analizará las diferencias entre RPC y comunicación Cliente-Servidor.

c) Los resultados de estos trabajos, deberán presentarse en forma de un reporte escrito de no más de 3 cuartillas, incluyendo la portada.

d) Se llevará a cabo un foro de discusión en el aula, para dar a conocer los resultados de las investigaciones; considerando todas las aportaciones de los alumnos; con la ayuda y conducción del maestro, se redactarán las conclusiones a que se llegue.

## **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

Explicar que es una RPC.

Explicar que es un Sistema Operativo de Modelo Cliente-Servidor.

Buscar y establecer las diferencias entre ellos.

## **SUGERENCIAS DIDACTICAS:**

Propiciar y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

## **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** Llamada a Procedimiento Remoto.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

## **Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- [https://es.wikipedia.org/wiki/Llamada\\_a\\_procedimiento\\_remoto](https://es.wikipedia.org/wiki/Llamada_a_procedimiento_remoto)
- 6.- [https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/RPC\\_notas.pdf](https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/RPC_notas.pdf)
- 7.- <https://www.textoscientificos.com/redes/tcp-ip/servicios-capa-transporte/rpc>

[Llamada a Procedimiento Remoto...](#)

# UNIDAD II. COMUNICACIÓN EN LOS SISTEMAS OPERATIVOS DISTRIBUIDOS.

## SISTEMAS OPERATIVOS II.

**En el sexto tema se abordará la comunicación en grupos.**

No.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
6	Comunicación en Grupos.	Conocer las formas en que se puede dar la comunicación en grupos.	Unidad II Comunicación en los sistemas operativos distribuidos.

### **Las Competencias específicas a desarrollar son:**

Establecer la comunicación y la sincronización entre los nodos de un sistema distribuido, conocer y manejar todos los elementos utilizados durante el proceso de resolución y distribución de nombres.

### **Subtemas.**

- 2.1 Comunicación: comunicación con cliente –servidor, comunicación con llamada a procedimiento remoto, comunicación en grupo, tolerancia a fallos.
- 2.2 Sincronización: relojes físicos, relojes lógicos, usos de la sincronización.
- 2.3 Nominación: características y estructuras, tipos de nombres, resolución y distribución, servidores y agentes de nombres, mapeo de direcciones, mapeo de rutas, modelo de Terry.
- 2.4 Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.

# PRÁCTICA No. 6

**NOMBRE DE LA PRÁCTICA:** Comunicación en Grupos.

**OBJETIVOS:** Conocer las formas en que se puede dar la comunicación en grupos.

## INTRODUCCIÓN.

Un tema muy importante en el ámbito de los Sistemas Operativos, es el de la comunicación entre ellos; tener la posibilidad de poder enviar mensajes al mismo tiempo es muy importante, ya que ello conlleva el hecho de poder sincronizar, entre otras cosas, los equipos con esas señales.

Comunicación.

En un sistema distribuido, a diferencia de uno centralizado no existe la memoria compartida y por ello la naturaleza de su comunicación ocurre lo siguiente: los procesos para comunicarse deben apegarse a reglas conocidas como "protocolos", para los sistemas distribuidos en un área amplia estos protocolos toman frecuentemente la forma de varias capas y cada capa tiene sus propias metas y reglas. Los mensajes se intercambian de diversas formas, existiendo muchas opciones de diseño al respecto, una importante opción es la llamada a un procedimiento remoto, también se consideran las posibilidades de comunicación entre grupos de procesos, no solo entre dos procesos.

Comunicación con RPC.

Es un protocolo que permite a un programa de computadora ejecutar código en otra máquina remota sin tener que preocuparse por las comunicaciones entre ambos. El protocolo es un gran avance sobre los sockets usados hasta el momento. De esta manera el programador no tenía que estar pendiente de las comunicaciones, estando éstas encapsuladas dentro de las RPC. Las RPC son muy utilizadas dentro de la comunicación cliente-servidor. Siendo el cliente el que inicia el proceso solicitando al servidor que ejecute cierto procedimiento o función y enviando éste de vuelta el resultado de dicha operación al cliente.

Comunicación con el cliente-servidor.

Se basa en un protocolo solicitud-respuesta, es sencillo y sin conexión, es sencillo y orientado a la conexión con OSI o TCP/IP, el cliente envía un mensaje de solicitud al servidor pidiendo cierto servicio, el servidor ejecuta el requerimiento, regresa los datos solicitados y un código de error si no pudo ejecutarlo correctamente, no se tiene que establecer una conexión sino hasta que esta se utilice, la pila del protocolo es más corta y por lo tanto más eficiente.

## Comunicación en grupo.

Un grupo es una colección de procesos que actúan juntos en cierto sistema o alguna forma determinada por el usuario, se trata de una comunicación uno a muchos, que se distingue de la comunicación punto a punto; de la misma manera se maneja lo que es la multitransmisión donde es posible crear una dirección especial de red a la que pueden escuchar todas las maquinas, las redes que no soportan la multitransmisión operan con transmisión simple donde se envían los mensajes a otras máquinas y donde cada una debe verificar mediante su software si el paquete va dirigido a ella, en caso negativo se descarta.

Los grupos deben poder direccionarse al igual que los procesos, una forma es darle a cada grupo una dirección única similar a una dirección de proceso, si una red soporta multitransmisión, la dirección del grupo se puede asociar con una dirección de multitransmisión, cada mensaje enviado a la dirección del grupo se podrá multitransmitir, si la red no soporta multitransmisión se tendrá que utilizar transmisión simple, si la red no soporta multitransmisión ni transmisión simple, se tendrá que utilizar unitransmisión, donde la comunicación en grupo se realiza mediante la transmisión, por parte del emisor, de paquetes individuales a cada uno de los miembros del grupo.

## Direccionamiento en grupo.

Con la comunicación en grupo existen en potencia varias respuestas referentes y no resulta aplicable el esquema de RPC, se utilizan llamadas explícitas para el envío y recepción, si se unifican las primitivas puntuales y grupales para Send, uno de los parámetros indica el destino, un segundo parámetro apunta al mensaje por enviar, si se fusionan las primitivas puntuales y grupales para Receive la operación concluye cuando llega un mensaje puntual o un mensaje de grupo.

## Primitivas Send y Receive.

La mayoría de los sistemas de comunicación en grupo están diseñados para que los mensajes enviados al mismo lleguen correctamente a todos los miembros o a ninguno de ellos; esta propiedad de todo o nada en entrega, se llama atomicidad o transmisión atómica, facilita la programación de los sistemas distribuidos y es de gran importancia para garantizar la consistencia de las bases de datos en los archivos distribuidos y duplicados.

## Ordenamiento de mensajes.

Un sistema debe tener una semántica bien definida con respecto al orden de entrega de los mensajes, la mejor garantía es la entrega de todos los mensajes en el orden en que fueron enviados; todos los mensajes dirigidos a un grupo deben entregarse antes de iniciar la entrega siguiente, todos los receptores reciben todos los

mensajes en un mismo orden, el esquema se denomina ordenamiento con respecto al tiempo global.

### **Atomicidad.**

Relojes lógicos.

Las computadoras poseen un circuito para el registro del tiempo conocido como dispositivo de reloj, éste es un cronometro consistente en un cristal de cuarzo de precisión sometido a una tensión eléctrica; teniendo varias computadoras con sus respectivos relojes es imposible garantizar que los cristales de cuarzo en computadoras distintas oscilen con la misma frecuencia, habrá una pérdida de sincronía en los relojes, es decir, que tendrán valores distintos al ser leídos, la diferencia entre los valores del tiempo se llama distorsión de reloj y podría generar fallas en los programas dependientes del tiempo.

Relojes físicos.

En ciertos sistemas se precisan relojes físicos externos, se deben sincronizar con los relojes del mundo real entre sí, la medición del tiempo real con alta precisión no es sencilla, ya que desde tiempos antiguos se mide astronómicamente.

Sincronización.

Este término se define como la forma de forzar una orden parcial o total en cualquier conjunto de eventos y es usado para hacer referencia a tres problemas distintos, pero relacionados entre sí:

- 1.- Sincronización entre el emisor y el receptor.
- 2.- Especificación y control de la actividad común entre procesos cooperativos.
- 3.- Serialización de accesos concurrentes a objetos compartidos de múltiples procesos.

### **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## **METODOLOGÍA.**

a) El alumno, en equipos de tres integrantes, llevará a cabo una investigación bibliográfica, en al menos tres fuentes de información, tomando en cuenta la bibliografía recomendada, referencias WEB y entrevistas con Maestros y/o profesionistas del ramo, que le permita identificar y definir el concepto de Comunicación en grupos.

b) El alumno, en equipos de tres integrantes, investigará cuales son los tres Sistemas Operativos que permiten llevar a cabo este tipo de comunicación.

c) El alumno, en equipos de tres integrantes, analizará las diferencias entre la Comunicación en grupos y la comunicación Cliente-Servidor, de al menos uno de los sistemas operativos investigados.

d) Los resultados de estos trabajos, deberán presentarse en forma de un reporte escrito de no más de 4 cuartillas, incluyendo la portada.

e) Cada equipo deberá elaborar una presentación de su trabajo, para ser presentada en un foro de discusión en el aula.

f) Se hará un foro de discusión en el aula, para poder conocer los resultados de las investigaciones de los equipos; considerando todas las aportaciones, con la ayuda y conducción del docente, se redactarán las conclusiones a que se llegue.

### **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

Explicar, ¿qué es la comunicación en grupos?

Explicar, ¿qué es una RPC?

Explicar, ¿qué es un Sistema Operativo de Modelo Cliente-Servidor?

Buscar y establecer las diferencias entre ellos.

### **SUGERENCIAS DIDACTICAS:**

Proponer y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

## **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** Comunicación en grupo.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

## **Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.

[Comunicación en Grupos....](#)

# UNIDAD II. COMUNICACIÓN EN LOS SISTEMAS OPERATIVOS DISTRIBUIDOS.

## SISTEMAS OPERATIVOS II.

**En el séptimo tema se abordará la comunicación de procesos a través del paso de mensajes en sistemas distribuidos.**

No.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
7	Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.	Conocer las diferentes formas para enviar mensajes en un sistema distribuido.	Unidad II Comunicación en los sistemas operativos distribuidos.

### **Las Competencias específicas a desarrollar son:**

Establecer la comunicación y la sincronización entre los nodos de un sistema distribuido, conocer y manejar todos los elementos utilizados durante el proceso de resolución y distribución de nombres.

### **Subtemas.**

- 2.1 Comunicación: comunicación con cliente –servidor, comunicación con llamada a procedimiento remoto, comunicación en grupo, tolerancia a fallos.
- 2.2 Sincronización: relojes físicos, relojes lógicos, usos de la sincronización.
- 2.3 Nominación: características y estructuras, tipos de nombres, resolución y distribución, servidores y agentes de nombres, mapeo de direcciones, mapeo de rutas, modelo de Terry.
- 2.4 Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.

# PRÁCTICA No. 7

**NOMBRE DE LA PRÁCTICA:** Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.

**OBJETIVOS:** Conocer las diferentes formas para enviar mensajes en un sistema distribuido.

## INTRODUCCIÓN.

Un tema muy importante en el ámbito de los Sistemas Operativos, es el de la comunicación entre ellos; tener la posibilidad de poder enviar mensajes al mismo tiempo es muy importante, ya que ello conlleva el hecho de poder sincronizar los equipos con esas señales.

Los procesos de un S.O. pueden comunicarse entre sí al compartir espacios de memoria, ya sean variables compartidas o buffers, o a través de las herramientas provistas por las rutinas de Comunicación de Interprocesos.

Para comunicar procesos en un ambiente distribuido, además del uso de un sistema de nombres de recursos, se necesita un esquema de comunicación lógico que dé sentido a estas transacciones. El sistema operativo provee mínimamente dos primitivas, enviar y recibir, normalmente llamadas send y receive, pero tendrá que implementar un enlace de comunicación entre los procesos. Este enlace puede ser unidireccional o multidireccional según permita la comunicación en solo uno o en varios sentidos, y dependiendo de la forma en que se dé la comunicación. Pueden existir los siguientes tipos de ésta:

**Comunicación Síncrona.** Quien envía permanece bloqueado esperando a que llegue una respuesta del receptor antes de realizar cualquier otro ejercicio.

**Comunicación Asíncrona.** Quien envía continúa con su ejecución inmediatamente después de enviar el mensaje al receptor.

**Comunicación Persistente.** El receptor no tiene que estar operativo al mismo tiempo que se realiza la comunicación, el mensaje se almacena tanto tiempo como sea necesario para poder ser entregado.

**Comunicación Transitoria.** El mensaje se descarta si el receptor no está operativo al tiempo que se realiza la comunicación. Por lo tanto, no será entregado.

**Comunicación Directa.** Las primitivas enviar y recibir usan directamente el nombre del proceso con el que se comunican. Por ejemplo: enviar (mensaje, A) envía un mensaje al proceso A. Obsérvese que la primitiva sólo debe especificar cuál va a ser el proceso Destino, ya que el proceso fuente viene direccionado en la comunicación.

Las operaciones básicas Send y Receive se definen de la siguiente manera: Send (P, mensaje); envía un mensaje al proceso P. Receive (Q, mensaje); espera la recepción de un *mensaje* por parte del proceso Q.

Comunicación Indirecta. Es aquella donde la comunicación está basada en un gateway, enrutador, puente o switch, ya que el emisor y el receptor están a distancia.

Comunicación Asimétrica. Un proceso puede enviar, los demás procesos solo reciben. También llamada unidireccional o no interactiva. Es el esquema típico de algunos servidores de Internet.

Comunicación con uso de buffers automático. El transmisor se bloquea hasta que el receptor recibe el mensaje completo, pero éste tiene capacidad para recibirlo, aunque no esté listo para procesarlo.

La comunicación y sincronización en S.O.D. es más compleja y se establece en canales lentos y menos confiables que los buses internos de una computadora, lo que incorpora problemas como la pérdida de mensajes, la llegada de datagramas desordenados, la heterogeneidad de los nodos y su diferente rendimiento, etc. La forma natural de comunicar y sincronizar procesos en los sistemas distribuidos es mediante paso de mensajes; los procesos intercambian mensajes mediante las primitivas que además establecen una extensión de los semáforos en la que se transmite más información en un contexto sincronizado.

Una de las ventajas de emplear mecanismos de comunicación y sincronización basados en paso de mensaje es la portabilidad de las soluciones programadas para diferentes arquitecturas de computadoras, incluidos los sistemas con memoria compartida, otra ventaja es que no existe el problema del acceso en exclusión mutua a datos compartidos, ya que no hay contienda por el acceso al recurso, sino una fila en espera. Un sistema operativo o lenguaje de programación podría ofrecer herramientas, algunas basadas en memoria compartida y otras basadas en la comunicación mediante paso de mensajes, por lo que podríamos llegar a tener un mismo proceso o hilo que empleara las dos posibilidades. Los siguientes son aspectos relevantes en el diseño de los sistemas de paso de mensajes:

1. Identificación en el proceso de comunicación.
2. Sincronización.
3. Características del canal (capacidad, flujo de datos, etc.)

## **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## **METODOLOGÍA.**

a) El alumno en forma individual, llevará a cabo una investigación, en al menos tres fuentes de información, tomando en cuenta la bibliografía recomendada, referencias WEB y entrevistas con Maestros y/o profesionistas del ramo, que le permita identificar el concepto de Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.

b) El alumno, en equipos de tres integrantes, investigará cuales son los Sistemas Operativos que permiten llevar a cabo este tipo de llamadas, así como las ventajas y desventajas de ellos.

c) Los resultados de estos trabajos, deberán presentarse en forma de un reporte escrito de no más de 4 cuartillas, incluyendo la portada, que incluyendo un cuadro comparativo de los principales sistemas operativos.

d) Se hará un foro de discusión en el aula, para poder conocer los resultados de las investigaciones; considerando todas las aportaciones, con la ayuda y conducción del docente, se redactarán las conclusiones pertinentes.

### **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

Explicar que es la Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.

### **SUGERENCIAS DIDACTICAS:**

Proponer y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

### **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** Comunicación de procesos a través del paso de mensajes en sistemas distribuidos.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

## Bibliografía preliminar:

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- [https://es.wikipedia.org/wiki/Comunicaci%C3%B3n\\_entre\\_procesos](https://es.wikipedia.org/wiki/Comunicaci%C3%B3n_entre_procesos)

[Comunicación con paso de mensajes...](#)

## UNIDAD V. USOS Y TENDENCIAS DE LOS SISTEMAS OPERATIVOS DISTRIBUIDOS. SISTEMAS OPERATIVOS II.

**En el octavo tema se abordará la Instalación de un Clúster de Alto Rendimiento.**

No.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
8	Instalación de Clúster de Alto Rendimiento.	Conocer cuáles son los requerimientos y los pasos para instalar un S.O. para implementar un Clúster de Alto Rendimiento.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.

**Las Competencias específicas a desarrollar son:**

Comprender las tendencias de investigación de los sistemas operativos distribuidos, su administración y el impacto en la sociedad.

**Subtemas.**

- 5.1 Administración de sistemas operativos distribuidos: instalación, configuración y control.
- 5.2. Instalación de clúster.
- 5.3. Estándares de administración en sistemas operativos distribuidos.
- 5.4. Cómputo de alto rendimiento a bajo costo.
- 5.5. Súper cómputo basado en clustering como solución a la necesidad de alto procesamiento en la nueva era.
- 5.6. Tendencias de la investigación.
- 5.7. Sistemas distribuidos como infraestructura para el soporte de las empresas en las TI.

# PRÁCTICA No. 8

**NOMBRE DE LA PRÁCTICA:** Instalación de Clúster de Alto Rendimiento.

**OBJETIVOS:** Conocer cuáles son los requerimientos y los pasos para instalar un S.O. para implementar un Clúster de Alto Rendimiento.

## INTRODUCCIÓN.

¿Qué es un clúster?

El término clúster se aplica a los conjuntos o conglomerados de computadoras construidos mediante la utilización de componentes de hardware comunes y que se comportan como si fueran un solo sistema. Hoy en día desempeñan un papel importante en la solución de problemas de las ciencias, las ingenierías y del comercio moderno. La tecnología de clúster ha evolucionado en apoyo de actividades que van desde aplicaciones de supercómputo y software de misiones críticas, servidores web y comercio electrónico, hasta bases de datos de alto rendimiento, entre otros usos. El cómputo con clústers surge como resultado de la convergencia de varias tendencias actuales que incluyen la disponibilidad de microprocesadores económicos de alto rendimiento y redes de alta velocidad, el desarrollo de herramientas de software para cómputo distribuido de alto rendimiento, así como la creciente necesidad de potencia computacional para aplicaciones que la requieran. Simplemente, un clúster es un grupo de múltiples sistemas de cómputo unidos mediante una red de alta velocidad, de tal forma que el conjunto es visto como un único sistema, más potente que los comunes de escritorio. Los clústers son usualmente empleados para mejorar el rendimiento y/o la disponibilidad por encima de la que es proporcionada por un solo sistema, prácticamente siendo más económico que computadores individuales de rapidez y disponibilidad comparables. La construcción del clúster es más fácil y económica debido a su flexibilidad: pueden tener la misma configuración de hardware y sistema operativo (clúster homogéneo), diferente rendimiento, pero con arquitecturas y sistemas operativos similares (clúster semihomogéneo), o tener diferente hardware y sistema operativo (clúster heterogéneo), lo que hace su construcción más fácil y económica.

Para que un clúster funcione como tal, no basta sólo con conectar entre sí las computadoras, sino que es necesario proveer un sistema de manejo del clúster, el cual se encargue de interactuar con el usuario y los procesos que corren en él para optimizar el funcionamiento.

Un clúster de alto rendimiento es un conjunto de computadoras que está diseñado para dar altas prestaciones en cuanto a capacidad de cálculo. Los motivos para utilizar un clúster de alto rendimiento son: el tamaño del problema a resolver y el precio de la máquina necesaria para resolverlo. Por medio de un clúster se pueden conseguir capacidades de cálculo superiores a las de una computadora más cara que el costo total de las computadoras del clúster.

## MATERIAL Y EQUIPO NECESARIO.

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## METODOLOGÍA.

Para poder instalar un clúster, se puede hacer utilizando varias máquinas y conectándolas en red o utilizar una aplicación que nos permita crear máquinas virtuales; esta última opción será la que tomaremos, por lo que procederemos a instalar el Software que nos permita hacerlo.

¿Qué es VirtualBox?

Virtualbox es un programa de virtualización capaz de instalar en nuestra máquina cualquier sistema operativo de forma virtual. Esta herramienta es ideal para empezar a conocer nuevos sistemas operativos y probar aplicaciones de software sin alterar nuestro sistema. Podemos descargar de forma totalmente gratuita este software de virtualización desde su página web principal.

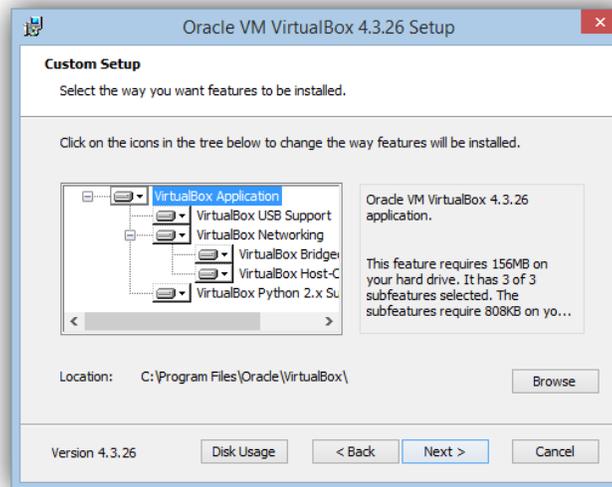
## Instalación.

En primer lugar, instalaremos VirtualBox en nuestro sistema que, en este caso, corresponde a Windows. Ejecutaremos el instalador del programa y se nos abrirá la primera pantalla tal y como aparece a continuación.

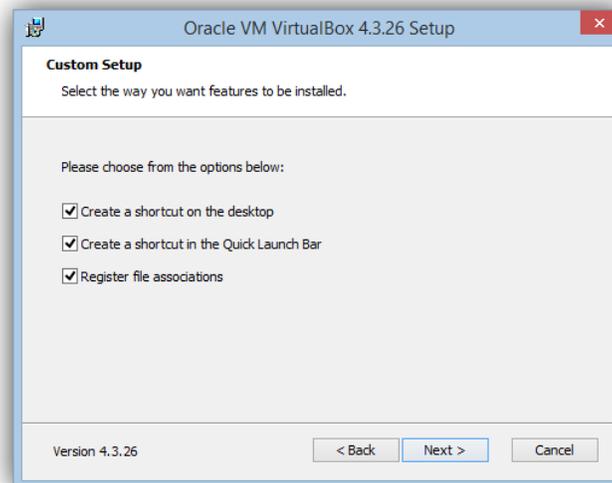


Nos da la bienvenida y nos informa de que si continuamos se instalara Virtualbox en nuestro sistema, pulsamos en “Next”.

En la siguiente pantalla deberemos elegir los componentes y la ruta donde lo instaremos.



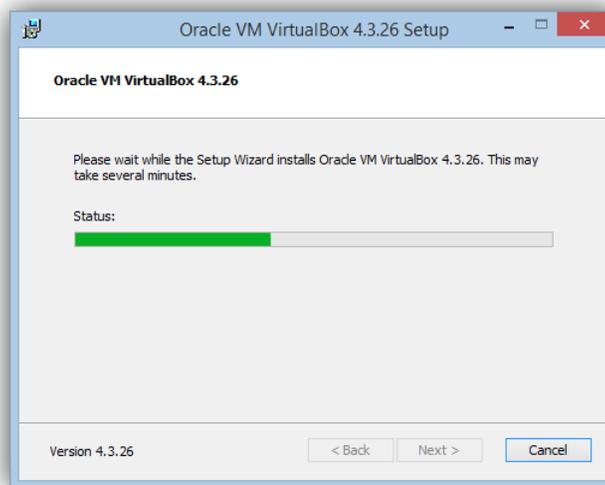
También los accesos directos que queremos configurar.



En esta pantalla se nos advertirá de que se van a copiar los archivos, se reiniciarán las tarjetas de red y se instalará el programa pulsamos en “Yes” y esperaremos el final de la instalación.



Una ventana como esta nos dirá cómo va la instalación así que tendremos que esperar a que finalice.

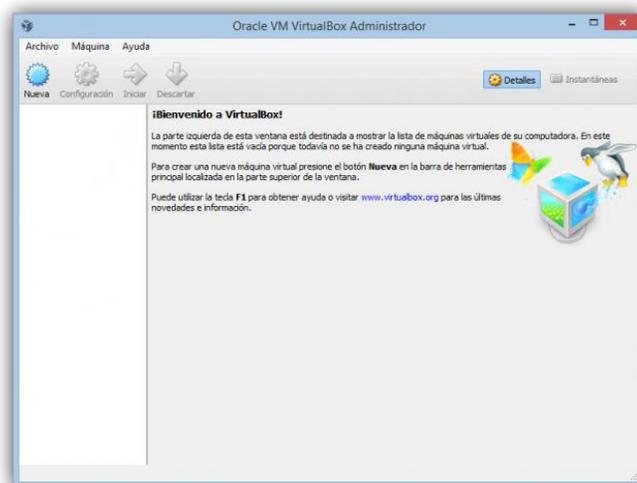


Esta es la ventana que nos indicará cuando el programa se ha instalado correctamente en nuestro sistema y que ya está listo para ser usado, oprimimos la opción “Finish”.

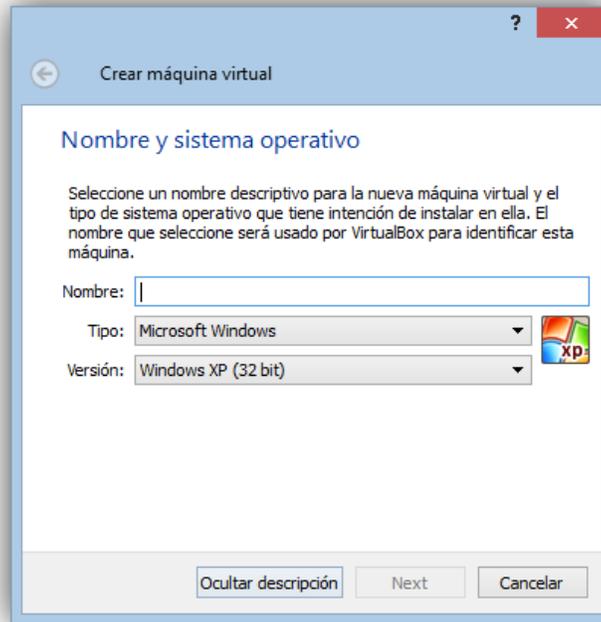


### Instalación de un sistema operativo.

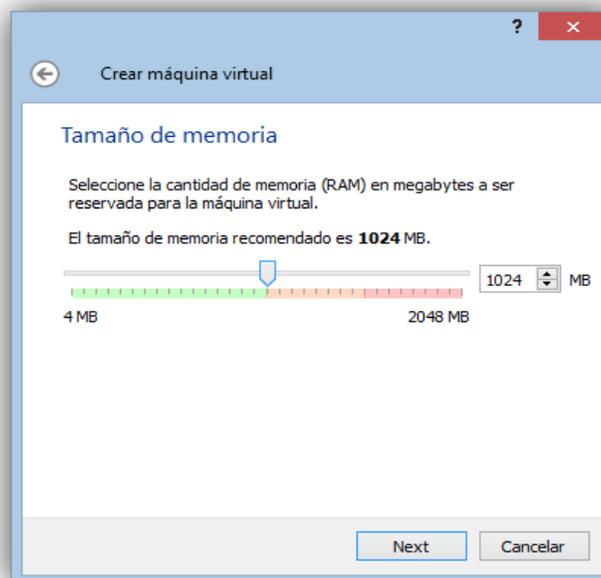
El uso del programa es sencillo ya que tiene un asistente que nos guiará en la instalación de nuestro sistema virtual. Para empezar, pulsaremos en la opción “Nueva” y se nos abrirá la pantalla siguiente que nos guiará en la instalación.



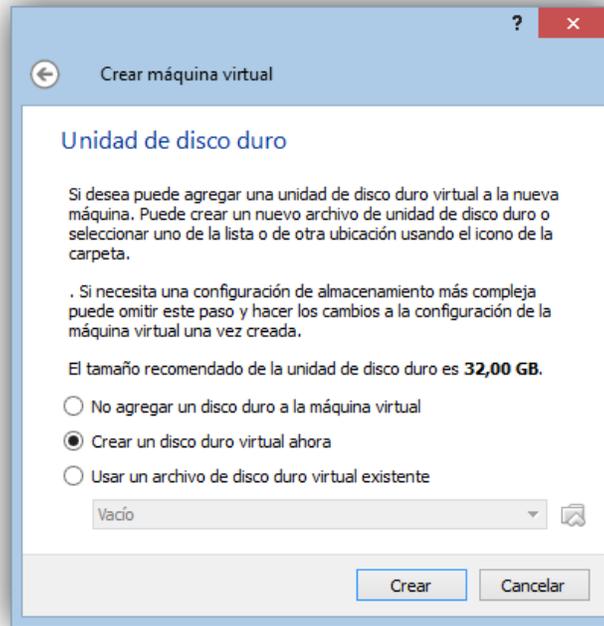
Ahora tendremos que dar un nombre al sistema que vamos a instalar y el sistema operativo que es y pulsamos en siguiente para continuar.



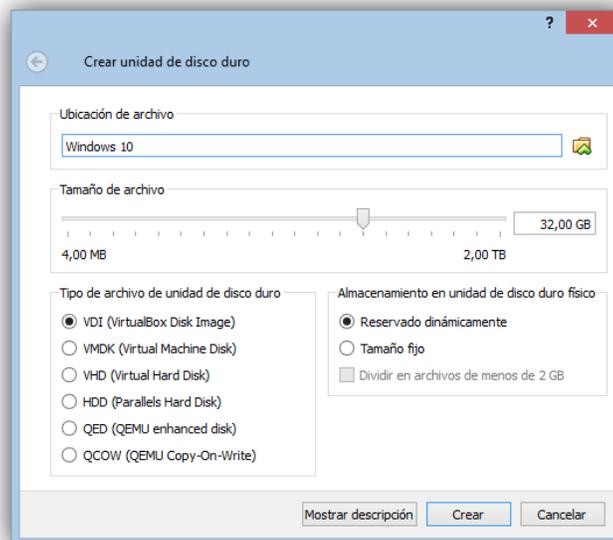
En la siguiente pantalla nos pedirá que especifiquemos la cantidad de memoria RAM que se asignara a la máquina virtual. En nuestro caso hemos dejado el tamaño predefinido, pulsamos la opción "Next".



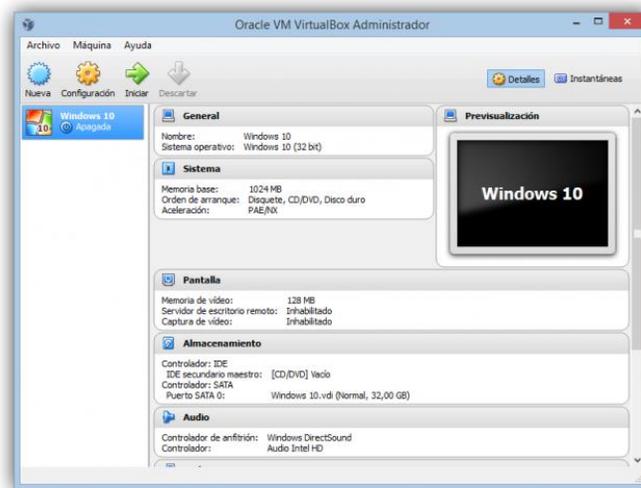
Seguimos con el asistente y ahora debemos crear el disco duro virtual, seleccionamos la opción de “Crear un disco duro virtual ahora” o utilizar uno ya existente. Pulsamos la opción “Crear”.



Al hacerlo entraremos en un asistente que nos dará las opciones como en la imagen que se muestra a continuación, Pulsamos la opción “Crear”, una vez que hayamos seleccionado las opciones adecuadas.

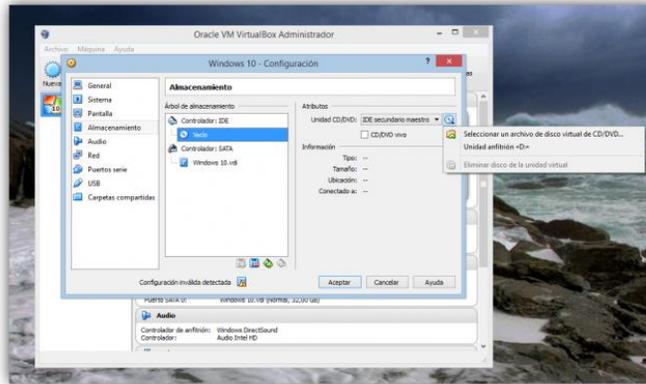


Una vez que le dimos las opciones deseadas al asistente, sólo quedará guardar los cambios para volver a la pantalla anterior. Pulsamos “Crear” para continuar. Así terminamos de crear nuestra máquina virtual y esta será la pantalla que nos mostrará.



## INSTALAR WINDOWS 7/8.1/10 EN NUESTRA MAQUINA VIRTUAL.

Vamos a configurar nuestra unidad de DVD desde donde instalaremos Windows. Para ello abrimos el menú de configuración de la máquina virtual y nos desplazamos hasta el menú que aparece en la pantalla.



En la pantalla que se nos abre seleccionaremos la letra de la unidad donde hemos cargado el disco y/o buscaremos una imagen ISO del mismo. Guardamos los cambios y la máquina virtual estará lista. Ahora pulsamos en “Iniciar” y seguiremos el asistente de instalación del sistema operativo como si se tratara de una computadora real.

Tutorial de VMware workstation.

VMware es un programa de virtualización. Con este programa se pueden instalar varios sistemas operativos de forma virtual en nuestras máquinas, por lo que podremos probar diferentes sistemas operativos, sin modificar nada en nuestro sistema. Con VMware workstation, podemos probar tantos sistemas operativos como queramos, por los que es bueno descargar el software desde su página Web.

## INSTALACION.

Al abrir el programa veremos la pantalla donde tendremos que aceptar el contrato de licencia tal y como muestra la siguiente imagen.

Una vez aceptemos el contrato de licencia nos mostrara la ventana principal del programa, la cual nos mostrara tres opciones principales tal y como muestra la siguiente imagen, que son:

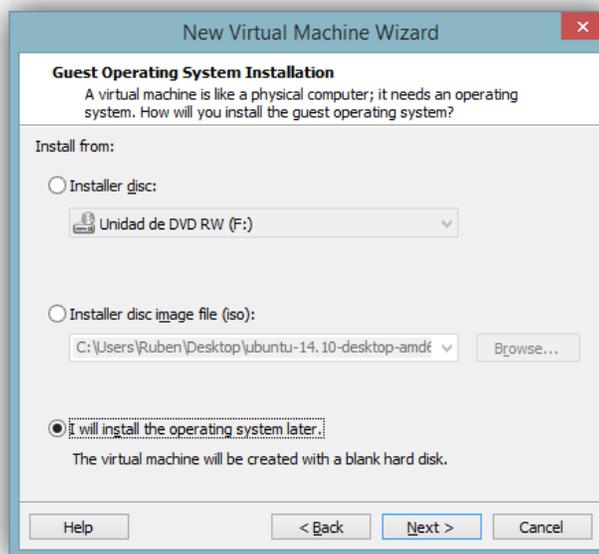
- **NEW VIRTUAL MACHINE:** esta será la opción que utilizaremos en este manual ya que es la encargada de crear nuestra máquina virtual.
- **NEW TEAM:** con esta opción agregamos máquinas virtuales a nuestra red de área privada.
- **OPEN EXISTING VM OR TEAM:** abre equipos virtuales existentes.

## CREAR UNA MAQUINA VIRTUAL.

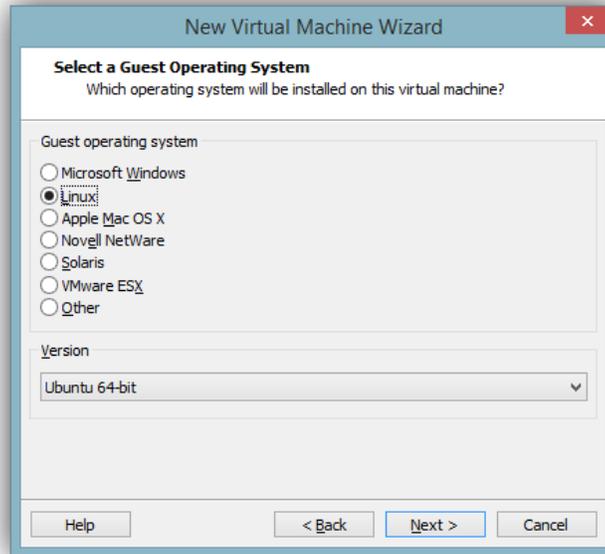
Lo primero que haremos será crear una nueva máquina virtual, para ello pulsaremos en la opción NEW VIRTUAL MACHINE y se abrirá un asistente. Tan solo tendremos que pulsar Next para continuar.



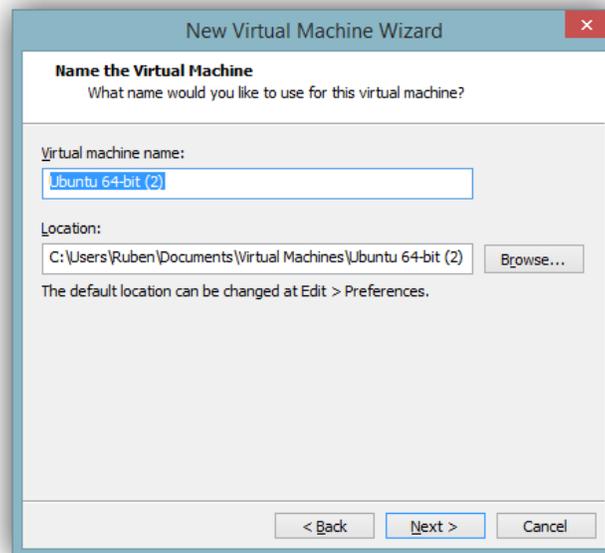
En este paso nos pregunta si queremos crear nuestra maquina con unas opciones preestablecidas (Typical) o si por el contrario somos nosotros quienes configuremos esas opciones (Custom). Escogemos la opción (Typical) para que sea VMware quien establezca la configuración y pulsaremos en siguiente para continuar. El siguiente paso es indicar desde qué medio vamos a instalar el sistema operativo.



La siguiente ventana nos pregunta cuál es el sistema operativo que vamos a cargar en nuestra máquina virtual, hemos seleccionado Ubuntu, una vez elegido pulsamos en “Next” para continuar.

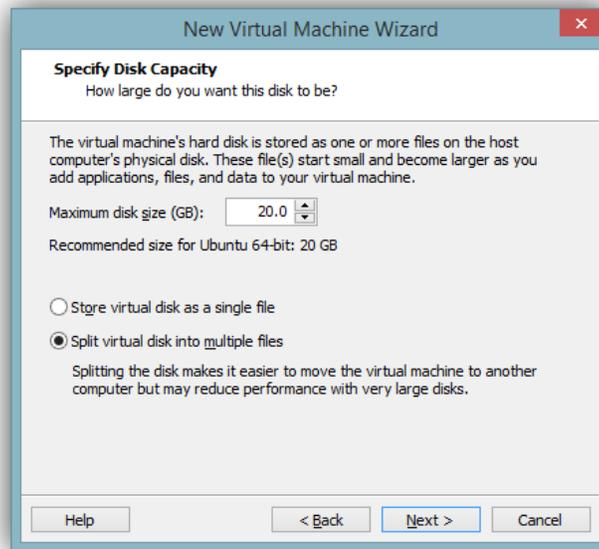


Ahora elegiremos el nombre y la ubicación donde se creará la máquina virtual.

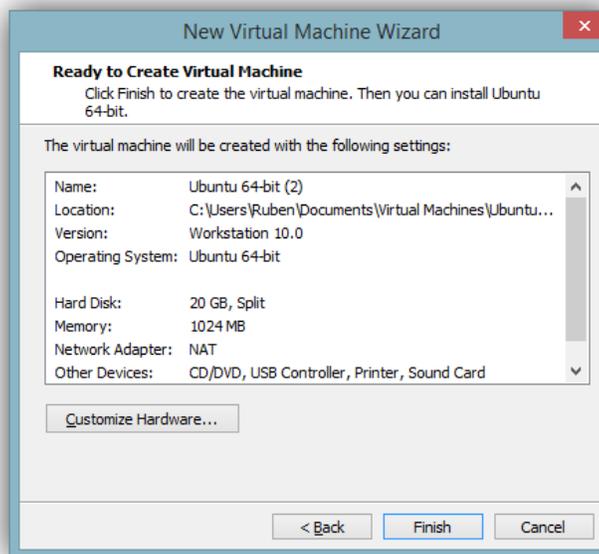


Pulsando en el botón “Browse...” se nos abrirá una ventana donde navegaremos por el disco duro para buscar la carpeta en la que guardaremos nuestra máquina virtual, una vez elegida, pulsamos en “Next” para continuar. Después, tendremos que elegir la capacidad del disco para la nueva máquina virtual.

- Storage virtual disk as a single file – destinar la totalidad de espacio en disco ahora.
- Split Virtual disk into multiple files 2 GB files – le asignamos 2 GB de espacio.

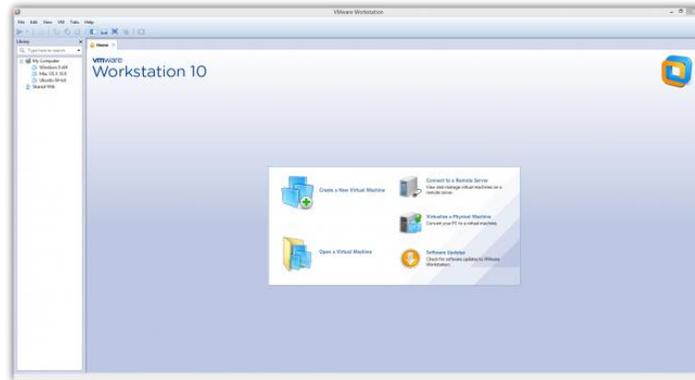


Seleccionamos la opción deseada y pulsamos en finalizar. Si todo ha salido bien nos mostrara la siguiente ventana indicando las características de la máquina virtual que ha sido creada.



## INSTALAR UN SISTEMA VIRTUAL.

Esta es la ventana principal de VMware Workstation.



Ahora vamos a instalar el sistema operativo; tan solo tendremos que cargar el CD con el sistema operativo en la unidad DVD y pulsar en “Start this virtual machine”, ahora instalará Ubuntu de forma virtual. Como podemos ver, en la parte izquierda nos aparecen los sistemas operativos ya instalados.

### Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).

Explicar que es un Clúster.  
Tipos de Clúster que existen.

### SUGERENCIAS DIDACTICAS:

Propiciar y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

### REPORTE DEL ALUMNO (Resultados):

**Presentación de trabajo:** Instalación de Clúster de Alto Rendimiento.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

**Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addis Wesley.
- 5.-file:///C:/Users/ITH-LAB-CC/Downloads/CuerpoAcademico/1748-5959-1-PB.pdf

[Cluster de Alto Rendimiento...](#)

# UNIDAD V. USOS Y TENDENCIAS DE LOS SISTEMAS OPERATIVOS DISTRIBUIDOS.

## SISTEMAS OPERATIVOS II.

En el noveno tema se abordará el cómo se construye un de Clúster de Alto Rendimiento.

No.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
9	Clúster de Alto rendimiento.	Conocer cómo se construye un Clúster de Alto rendimiento.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.

### Las Competencias específicas a desarrollar son:

Comprender las tendencias de investigación de los sistemas operativos distribuidos, su administración y el impacto en la sociedad.

### Subtemas.

- 5.1 Administración de sistemas operativos distribuidos: instalación, configuración y control.
- 5.2. Instalación de clúster.
- 5.3. Estándares de administración en sistemas operativos distribuidos.
- 5.4. Cómputo de alto rendimiento a bajo costo.
- 5.5. Súper cómputo basado en clustering como solución a la necesidad de alto procesamiento en la nueva era.
- 5.6. Tendencias de la investigación.
- 5.7. Sistemas distribuidos como infraestructura para el soporte de las empresas en las TI.

# PRÁCTICA No. 9

**NOMBRE DE LA PRÁCTICA:** Clúster de Alto Rendimiento.

**OBJETIVOS:** Conocer cómo se construye un Clúster de Alto rendimiento.

## INTRODUCCIÓN.

Como en muchos ámbitos, en el de los Sistemas Operativos, es muy importante saber hacia dónde van los avances en éstas áreas y cuál podría ser el impacto de ese avance en la sociedad.

Un clúster de alto rendimiento es un conjunto de computadoras utilizadas como un recurso unificado de procesamiento que comparte una administración común. El primer clúster de tipo Beowulf fue desarrollado en la NASA en el año de 1994 por Thomas Sterling y Donald Becker. Este clúster estaba formado por 16 procesadores DX4 interconectados mediante una red Ethernet de 10Mbps. Los principales componentes de un clúster Beowulf son: procesador, memoria principal, red de intercomunicación entre nodos, un sistema de almacenamiento secundario y software que permita la comunicación entre los procesos de diferentes nodos (bibliotecas de envío de mensajes). Se le llama clúster de alto rendimiento ya que se construye con la intención de optimizar los procesos ejecutados tratando de lograr el mayor número de operaciones de punto flotante en el menor tiempo posible. Los procesos que se ejecutan en un clúster corren bajo un entorno de procesamiento paralelo. El procesamiento paralelo consiste en dividir grandes tareas en pequeñas subtareas que son procesadas paralelamente. El cómputo paralelo ha sido un punto clave en el desarrollo de nuevos algoritmos para optimizar las capacidades de procesamiento y poder desarrollar cómputo de alto rendimiento. En el año de 1966 Michael J. Flynn introdujo un esquema de clasificación de computadoras que se basaba en la multiplicidad de instrucciones y el flujo de datos que podía manejar cada tipo de computadoras: SISD (Single Instruction Single Data, Una instrucción un flujo de datos), MISD (Multiple Instruction Single Data, Muchas instrucciones un flujo de datos), SIMD (Single Instruction Multiple Data, Una instrucción muchos flujos de datos), MIMD (Multiple Instruction Multiple Data, Muchas instrucciones Muchos datos). Los clústers se encuentran clasificados en el último tipo. El uso del cómputo de alto rendimiento en el área científica se ha propagado de manera tal que se ha generado una rama dedicada al cómputo científico y con esto han surgido nuevas metodologías: experimental, teórica y computacional.

El diseño de un Clúster de Alto Rendimiento comienza con la obtención y recopilación de información acerca de la utilidad que nos va proporcionar este tipo de arquitectura y se concluye en el armado de un clúster tipo Beowulf, incluye el resumen y estudio acerca del estado del arte del “clustering”, lo que nos proporciona los conocimientos necesarios para lograr la configuración y más tarde la implementación ya con un objetivo general, el cómputo de alto rendimiento en el área científica. El siguiente paso es la obtención del equipo, aquí fue necesario la gestión para la adquisición del equipo necesario para la implementación del clúster, en donde entran

las computadoras, concentrador (hub) y una adecuada área de trabajo, además de buscar el software (librerías, programas y utilerías) que se necesitaran en tiempo de ejecución de tareas para el clúster.

## **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## **METODOLOGÍA.**

El desarrollo del proyecto contempla las siguientes fases:  
Instalación de las computadoras.

- a) Instalación de los sistemas operativos, así como todos los controladores necesarios para la puesta a punto de las computadoras implicadas en el proyecto.
- b) Instalación de software específico.
- c) Instalación de todas las aplicaciones necesarias para la administración de los recursos. Aplicaciones necesarias para montar el servidor web, las bases de datos, configuración del firewall (funcionamiento de la red). Así como todo el software extra para el funcionamiento del clúster.
- d) Configuración de las computadoras. Configurar las diferentes aplicaciones de cada máquina.
- e) Configuración de la red. Configurar la red, asignación de rangos de ip's comunicación entre la red interna y externa, configurar la DMZ (zona desmilitarizada).
- f) El alumno en forma individual, llevará a cabo una investigación de campo, en al menos tres fuentes, que le permita identificar que es un clúster de alto rendimiento y cuáles son los requerimientos que tienen que cumplir para su implementación; así como sus ventajas y desventajas. Los resultados de estos trabajos de investigación, deberán presentarse en forma de un reporte individual, de al menos dos cuartillas. Se hará un foro de discusión en el aula, para poder conocer los resultados de las investigaciones; considerando todas las aportaciones para redactar conclusiones.

## **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

Explicar que es un Clúster de Alto Rendimiento.

### **SUGERENCIAS DIDACTICAS:**

Propiciar y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

### **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** Instalación de Clúster de Alto Rendimiento.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

### **Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.

[Video Clúster...](#)

# UNIDAD V. USOS Y TENDENCIA DE LOS SISTEMAS OPERATIVOS DISTRIBUIDOS. SISTEMAS OPERATIVOS II.

**En el décimo tema se abordará el cómo se lleva a cabo la Instalación de Clúster de Alta Disponibilidad.**

No.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
10	Instalación de Clúster de Alta Disponibilidad.	Conocer cuáles son los requerimientos y los pasos para instalar un S.O. para implementar un Clúster de Alta Disponibilidad.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.

## **Las Competencias específicas a desarrollar son:**

Comprender las tendencias de investigación de los sistemas operativos distribuidos, su administración y el impacto en la sociedad.

## **Subtemas.**

- 5.1 Administración de sistemas operativos distribuidos: instalación, configuración y control.
- 5.2. Instalación de clúster.
- 5.3. Estándares de administración en sistemas operativos distribuidos.
- 5.4. Cómputo de alto rendimiento a bajo costo.
- 5.5. Súper cómputo basado en clustering como solución a la necesidad de alto procesamiento en la nueva era.
- 5.6. Tendencias de la investigación.
- 5.7. Sistemas distribuidos como infraestructura para el soporte de las empresas en las TI.

# PRÁCTICA No. 10

**NOMBRE DE LA PRÁCTICA:** Instalación de Clúster de Alta Disponibilidad.

**OBJETIVOS:** Conocer cuáles son los requerimientos y los pasos para instalar un S.O. para implementar un Clúster de Alta Disponibilidad.

## INTRODUCCIÓN.

Para poder conocer cómo trabajan este tipo de sistemas operativos, es necesario saber instalarlos, así como manejarlos.

## CLUSTER DE ALTA DISPONIBILIDAD.

Clúster de alta disponibilidad (HA).

En la actualidad las organizaciones dependen cada vez más de sus sistemas de información, y como es obvio se desea que estos sean seguros y permanezcan disponibles el mayor tiempo posible. Para cualquier institución, una interrupción de sus sistemas de información supone un serio problema.

Efectos por la interrupción de un sistema de información:

- Costos directos asociados a la reparación del sistema de información (piezas a reparar o sustituir, servicios técnicos, etc.).
- Horas de trabajo adicionales para el departamento de sistemas que tiene que reparar la avería.
- Perdidas de productividad o incluso horas de trabajo perdidas por los empleados que dependen del sistema.
- Pérdida de ingresos, por las ventas o servicios que se han dejado de realizar.
- Costos indirectos: satisfacción de los clientes, pérdida de reputación, mala publicidad, desconfianza de los empleados, etc.

La disponibilidad es una medida relativa a la preparación para su utilización de un sistema informático, mientras que la fiabilidad es una medida relativa a su capacidad para mantenerse operativo en el tiempo sin ningún tipo de fallo. Los fallos potenciales de un sistema son los errores de componentes de hardware, los errores o bloqueos del sistema operativo, los errores de las aplicaciones.

Un Clúster de alta disponibilidad es un conjunto de dos o más máquinas que se caracterizan por mantener una serie de servicios compartidos y por estar constantemente monitoreándose entre sí.

## Tipos de Clúster.

**Alta disponibilidad de infraestructura:** Si se produce un fallo de hardware en alguna de las máquinas del clúster, el software de alta disponibilidad es capaz de arrancar automáticamente los servicios en cualquiera de las otras máquinas del clúster (failover). Y cuando la máquina que ha fallado se recupera, los servicios son nuevamente migrados a la máquina original (failback). Esta capacidad de recuperación automática de servicios nos garantiza la alta disponibilidad de los servicios ofrecidos por el clúster, minimizando así la percepción del fallo por parte de los usuarios.

**Alta disponibilidad de aplicación:** Si se produce un fallo del hardware o de las aplicaciones de alguna de las máquinas del clúster, el software de alta disponibilidad es capaz de arrancar automáticamente los servicios que han fallado en cualquiera de las otras máquinas del clúster. Y cuando la máquina que ha fallado se recupera, los servicios son nuevamente migrados a la máquina original. Esta capacidad de recuperación automática de servicios nos garantiza la integridad de la información, ya que no hay pérdida de datos, y además evita molestias a los usuarios, que no tienen por qué notar que se ha producido un problema.

**Disponibilidad:** La disponibilidad es el grado en que una aplicación o servicio está disponible cuándo y cómo los usuarios esperan. La disponibilidad se mide por la percepción de una aplicación del usuario final. Los usuarios finales experimentan frustración cuando sus datos no están disponibles, y ellos no entienden o son capaces de diferenciar los complejos componentes de una solución global. Fiabilidad, valorización, continuas operaciones y detección de errores son características de una solución de alta disponibilidad.

**Fiabilidad:** Los componentes hardware fiables de una solución de HA, el software fiable, incluida la base de datos, servidores web y aplicaciones, es la parte crítica de una implementación de una solución de alta disponibilidad.

**Recuperación:** Puede haber muchas opciones para recuperarse de un fracaso si ocurre alguno. Es importante determinar qué tipo de fallos pueden ocurrir en su entorno de alta disponibilidad y la forma de recuperarse de estos fallos en el tiempo que satisface las necesidades comerciales. Por ejemplo, si una tabla importante es eliminada de la base de datos, ¿qué medidas adoptarías para recuperarla? ¿Su arquitectura ofrece la capacidad de recuperarse en el tiempo especificado en un acuerdo de nivel de servicio (SLA)?

**Detección de errores:** Si un componente en su arquitectura falla, entonces la rápida detección, de dicho componente es esencial en la recuperación de un posible fracaso inesperado. Si bien es posible que pueda recuperarse rápidamente de un corte de luz, si se lleva a otros 90 minutos para descubrir el problema, entonces usted no puede satisfacer su SLA. La monitorización del estado del entorno de trabajo requiere un software fiable, para ver de forma rápida y notificar al administrador de bases de datos (DBA) un problema.

Continuas operaciones: El continuo acceso a sus datos es esencial, por muy pequeño o inexistente que sea el tiempo de caída del sistema, para llevar a cabo las tareas de mantenimiento. Actividades como mover una tabla de un lado a otro dentro de la base de datos, o incluso añadir nuevas CPU's a su hardware debe ser transparente para el usuario final en una arquitectura HA.

Cálculo de la disponibilidad.

En un sistema real, si falla uno de los componentes, es reparado o sustituido por un nuevo componente. Si este nuevo componente falla, es sustituido por otro, y así sucesivamente. El componente fijo se considera en el mismo estado que un nuevo componente. Durante su vida útil, uno de los componentes puede ser considerado en uno de estos estados: Funcionando o en Reparación; El estado funcionando indica que el componente está operacional y en Reparación significa que ha fallado y todavía no ha sido sustituido por un nuevo componente.

En caso de defectos, el sistema va de Funcionando en modo Reparación, y cuando se hace la sustitución volverá al estado Funcionando. Por lo tanto, podemos decir que el sistema tiene durante su vida, una media de tiempo para presentar fallas (MTTF) y un tiempo medio de reparación (MTTR). Su tiempo de vida es una sucesión de MTTFs y MTTRs, a medida que este va fallando y siendo reparado. El tiempo de vida útil del sistema es la suma de MTTFs en ciclos MTTF + MTTR ya vividos.

En forma simplificada, se dice que la disponibilidad de un sistema es la relación entre la duración de la vida útil de este sistema y de su tiempo total de vida. Esto puede ser representado por la fórmula de abajo:

Disponibilidad =  $MTTF / (MTTF + MTTR)$

Porcentaje de disponibilidad	Tiempo de Interrupción Anual	Tiempo de Interrupción Semanal
98 %	7,3 días	3,3 horas
99 %	3,6 días	1,7 horas
99,9 %	8,8 horas	10 minutos
99,99 %	52,5 minutos	1 minuto
99,999 %	5,3 minutos	6 segundos
99,9999 %	31,5 segundos	0,6 segundos

En la evaluación de una solución de Alta Disponibilidad, es importante tener en cuenta si en la medición de MTTF son vistas como fallas las posibles paradas planificadas. En la actualidad, eligiendo correctamente el hardware y software adecuados, es relativamente sencillo diseñar un sistema con una disponibilidad del 98% del tiempo. Pero el paso del 98% al 99% y de aquí al 99,9999% es una tarea compleja y a la par supone un aumento exponencial del coste total del sistema. En la práctica se alcanza un compromiso entre la disponibilidad pretendida y el costo abordable.

Las razones para implementar un clúster de alta disponibilidad son:

- Aumentar la disponibilidad.
- Mejorar el rendimiento.
- Escalabilidad.
- Tolerancia a fallos.
- Recuperación ante fallos en tiempo aceptable
- Reducir costos.
- Consolidar servidores.
- Consolidar el almacenamiento.
- Configuraciones de alta disponibilidad.

Las configuraciones más comunes en entornos de clúster de alta disponibilidad son la configuración activo/activo y la configuración activo/pasivo.

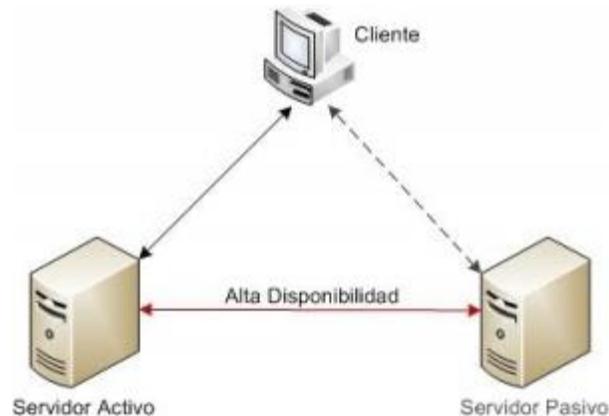


### **Configuración Activo/Activo.**

En una configuración activo/activo, todos los servidores del clúster pueden ejecutar los mismos recursos simultáneamente. Es decir, los servidores poseen los mismos recursos y pueden acceder a estos independientemente de los otros servidores del clúster. Si un nodo del sistema falla y deja de estar disponible, sus recursos siguen estando accesibles a través de los otros servidores del clúster.

La ventaja principal de esta configuración es que los servidores en el clúster son más eficientes ya que pueden trabajar todos a la vez. Sin embargo, cuando uno de los servidores deja de estar accesible, su carga de trabajo pasa a los nodos restantes, lo que produce una degradación del nivel global de servicio ofrecido a los usuarios.

En la siguiente figura se muestra como ambos servidores están activos, proporcionando un mismo servicio a los diferentes usuarios. Los clientes acceden al servicio o recursos de forma transparente y no tienen conocimiento de la existencia de varios servidores formando un clúster.



### **Configuración Activo/Pasivo.**

Un clúster de alta disponibilidad, en una configuración activo/pasivo, consiste en un servidor que posee los recursos del clúster y otros servidores que son capaces de acceder a esos recursos, pero no los activan hasta que el propietario de los recursos ya no esté disponible.

Las ventajas de la configuración activo/pasivo son que no hay degradación de servicio y que los servicios solo se reinician cuando el servidor activo deja de responder. Sin embargo, una desventaja de esta configuración es que los servidores pasivos no proporcionan ningún tipo de recurso mientras están en espera, haciendo que la solución sea menos eficiente que el clúster de tipo activo/activo. Otra desventaja es que los sistemas tardan un tiempo en migrar los recursos (failover) al nodo en espera.

Funcionamiento de un clúster de alta disponibilidad.

En un clúster de alta disponibilidad, el software de clúster realiza dos funciones fundamentales. Por un lado, intercomunica entre sí todos los nodos, monitorizando continuamente su estado y detectando fallos. Y por otro lado administra los servicios ofrecidos por el clúster, teniendo la capacidad de migrar dichos servicios entre diferentes servidores físicos como respuesta a un fallo.

### **Elementos y conceptos básicos en el funcionamiento del clúster.**

Recursos y Grupos de Recursos: Tradicionalmente se entiende como servicio a un conjunto de procesos que se ejecutan en un momento dado sobre un servidor y sistema operativo. Este último provee a los procesos de los recursos necesarios para realizar su tarea: sistema de archivos, interfaces de red, tiempo de procesador, memoria, etc.

En un clúster de alta disponibilidad, el software del mismo, abstrae e independiza a los servicios de un host concreto, posibilitando que estos se desplacen entre diferentes servidores de forma transparente para la aplicación o los usuarios.

El software de clúster permite definir grupos de recursos, que son todos aquellos recursos necesarios por dar un servicio. Estos recursos serán los scripts de arranque del servicio, un sistema de archivos, una dirección IP, etc.

Intercomunicación: El software de clúster gestiona servicios y recursos en los nodos; Pero, además, tiene que mantener continuamente entre estos una visión global de la configuración y estado del clúster. De esta forma, ante el fallo de un nodo, el resto conoce que servicios se deben restablecer. Ya que la comunicación entre los nodos del clúster es crucial para el funcionamiento de este, es habitual utilizar un canal específico como una red IP independiente o una conexión en serie, que no se pueda ver afectada por problemas de seguridad o rendimiento.

Heartbeat: El software de clúster conoce en todo momento la disponibilidad de los equipos físicos, gracias a la técnica de heartbeat. El funcionamiento es sencillo, cada nodo informa periódicamente de su existencia enviando al resto una "señal de vida".

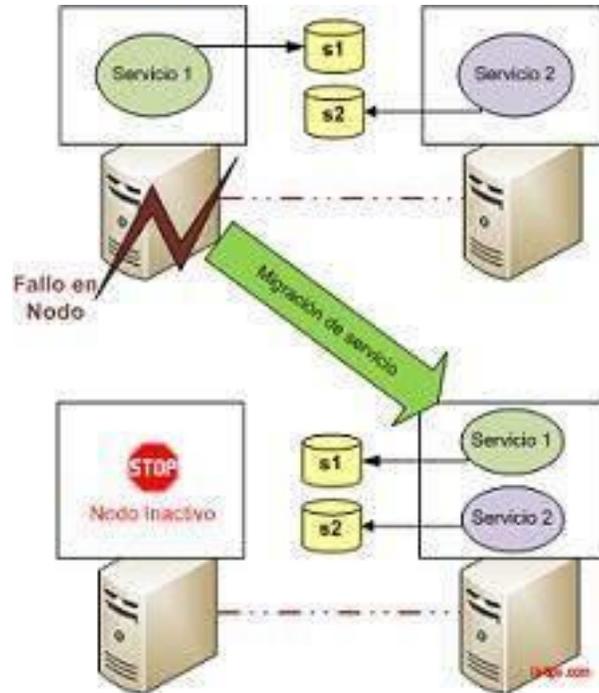
Escenario Split-Brain: En un escenario Split-brain, más de un servidor o aplicación pertenecientes a un mismo clúster intentan acceder a los mismos recursos, lo que puede causar daños a dichos recursos. Este escenario ocurre cuando cada servidor en el clúster cree que los otros servidores han fallado e intenta activar y utilizar dichos recursos.

Monitorización de Recursos (Resource Monitoring): Ciertas soluciones de clustering HA permiten no solo monitorear si un host físico está disponible; también pueden realizar seguimientos a nivel de recursos o servicios y detectar el fallo de estos. El administrador puede configurar la periodicidad de estos monitores, así como las acciones a llevar a cabo en caso de fallo.

Reiniciar Recursos: Cuando un recurso falla, la primera medida que toman las soluciones de clúster es intentar reiniciar dicho recurso en el mismo nodo. Lo que supone detener una aplicación o liberar un recurso y posteriormente volverlo a activar. Algunas implementaciones no permiten reiniciar un único recurso, y lo que realizan es un reinicio completo de todo un grupo de recursos (servicio). Esto puede llegar a demorar bastante para servicios como las bases de datos.

Migración de Recursos (Failover): Cuando un nodo ya no está disponible, o cuando un recurso fallido no se puede reiniciar satisfactoriamente en un nodo, el software de clúster reacciona migrando el recurso o grupo de recursos a otro nodo disponible en el clúster; de este modo el tiempo de inactividad por el posible fallo es mínimo, y el clúster seguirá proporcionando el correspondiente servicio.

Dependencia entre recursos: Habitualmente para que el clúster proporcione un servicio, son necesarios no solo un recurso sino varios (ip virtual, sistema de archivos, procesos), lo que se conoce como grupo de recursos. Cuando se arranca o detiene un servicio, sus recursos tienen que activarse en el orden apropiado ya que unos dependen de otros. El software de clúster tiene que permitir definir estas dependencias entre recursos, así como entre grupos.



**Dependencia entre recursos del Clúster.**

## Tutorial de VMware workstation.

VMware es un programa de virtualización. Con este programa se pueden instalar varios sistemas operativos de forma virtual en nuestra máquina, por lo que podremos probar diferentes sistemas operativos, sin modificar nada en nuestro sistema, podemos descargar de forma gratuita este software de desde su página web principal.

## INSTALACION.

Al abrir el programa veremos la pantalla donde tendremos que aceptar el contrato de licencia tal; Una vez aceptado el contrato de licencia nos mostrara la ventana principal del programa, la cual nos mostrara tres opciones que son:

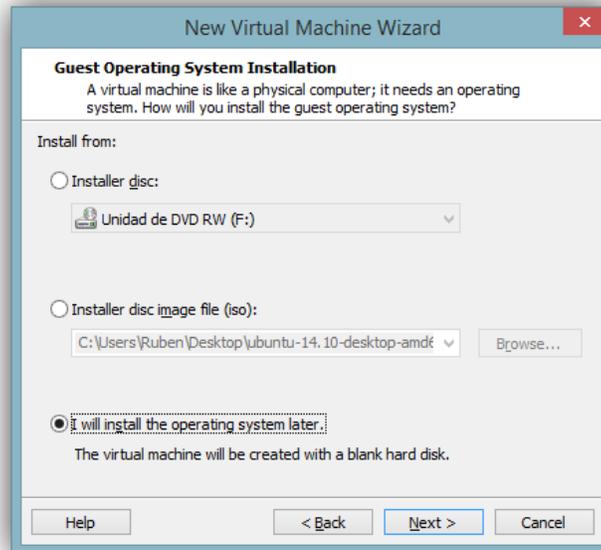
- **NEW VIRTUAL MACHINE:** esta será la opción que utilizaremos en este manual ya que es la encargada de crear nuestra máquina virtual.
- **NEW TEAM:** con esta opción agregamos máquinas virtuales a nuestra red de área privada.
- **OPEN EXISTING VM OR TEAM:** abre equipos virtuales existentes.

## CREAR UNA MAQUINA VIRTUAL.

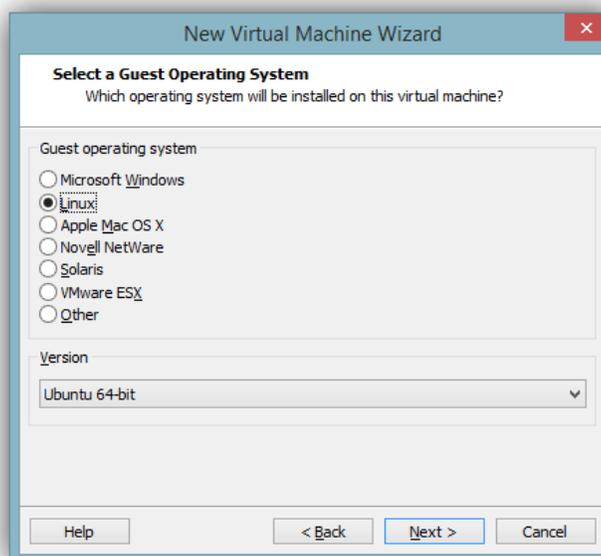
Lo primero que haremos será crear una nueva máquina virtual, para ello pulsaremos en la opción **NEW VIRTUAL MACHINE** y se abrirá un asistente. Tan solo tendremos que pulsar "Next" para continuar. En este paso nos pregunta si queremos crear nuestra maquina con unas opciones preestablecidas (Typical) o si por el contrario somos nosotros quienes configuremos esas opciones (Custom). Seleccionamos la opción (Typical) para que sea VMware quien establezca la configuración y pulsaremos en "Next" para continuar.



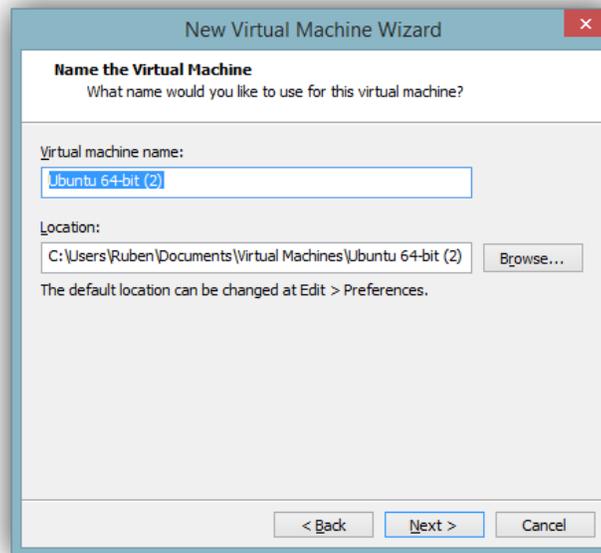
El siguiente paso es indicar desde qué medio vamos a instalar el sistema operativo o si lo instalaremos más tarde (tomamos esta última opción) y pulsamos “Next”.



La siguiente ventana nos pregunta cuál es el sistema operativo que vamos a cargar en nuestra máquina virtual, hemos seleccionado Linux(Ubuntu), una vez elegido pulsamos en “Next” para continuar.



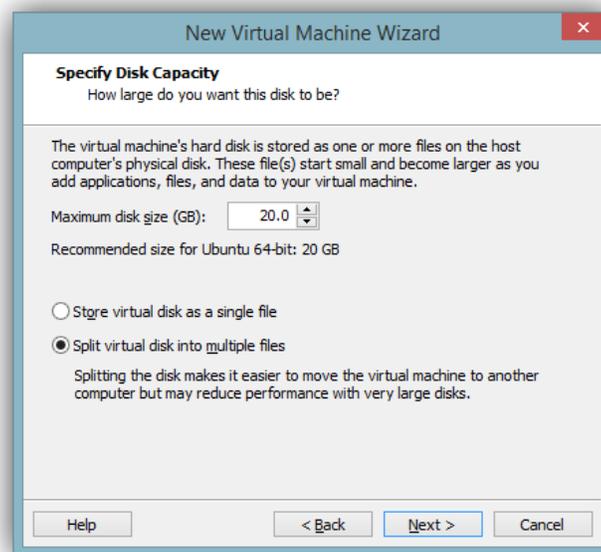
Ahora elegiremos el nombre y la ubicación donde se instalará la máquina virtual. Pulsamos el botón “Browse...” y se nos abrirá una ventana donde navegaremos por el disco duro para buscar la carpeta en la que guardaremos nuestra máquina virtual, una vez seleccionada, pulsamos “Next” para continuar.



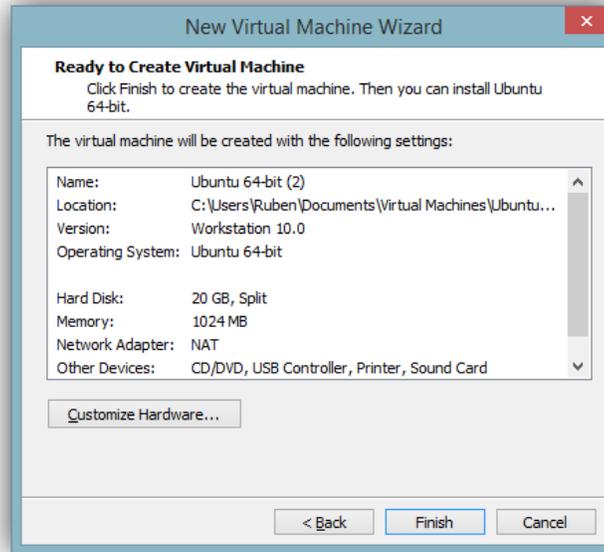
Ahora tendremos que elegir la capacidad de nuestro disco en la máquina virtual. Nos muestra dos opciones:

- Storage single disk as a single file – destinar la totalidad de espacio en disco ahora.
- Split virtual disk into multiples files – Dividir disco virtual en muchos archivos.

Seleccionamos ésta última y pulsámos “Next”.



Aparece la pantalla donde nos indica las características de la máquina virtual que crearemos, una vez hecho esto pulsamos en “Finish”, si todo ha salido bien nos mostrara una ventana indicando que la máquina virtual ha sido creada.



## INSTALAR UN SISTEMA VIRTUAL

Esta es la ventana principal de VMware Workstation



Ahora vamos a instalar el sistema operativo, tan solo tendremos que cargar el CD con el sistema operativo en la unidad DVD y pulsar en “Start this virtual machine”, ahora instalará Ubuntu. Como podemos ver, en la parte izquierda nos aparecen los sistemas operativos ya instalados.

## **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## **METODOLOGÍA.**

El desarrollo del proyecto contempla las siguientes fases:

- a) Instalación de los sistemas operativos, así como todos los controladores necesarios para la puesta a punto de las computadoras implicadas en el proyecto.
- b) Instalación de software específico.
- c) Instalación de todas las aplicaciones necesarias para la administración de los recursos. Aplicaciones necesarias para montar el servidor web, las bases de datos, configuración del firewall (funcionamiento de la red). Así como todo el software extra para el funcionamiento del clúster.
- d) Configuración de las computadoras. Configurar las diferentes aplicaciones de cada máquina.
- e) Configuración de la red. Configurar la red, asignación de rangos de ip's comunicación entre la red interna y externa, configurar la DMZ (zona desmilitarizada).
- f) El alumno en forma individual, llevará a cabo una investigación de campo, en al menos tres fuentes, que le permita identificar que es un clúster de alta disponibilidad y cuáles son los requerimientos que tienen que cumplir para su implementación; así como sus ventajas y desventajas.
- g) Los resultados de estos trabajos de investigación, deberán presentarse en forma de un reporte individual (cuatro cuartillas), incluyendo la portada.
- h) Se hará un foro de discusión en el aula, para conocer los resultados de las investigaciones; considerando todas las aportaciones para redactar conclusiones.

### **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

Explicar cómo se instala un Clúster de Alto Rendimiento y cuáles son los requerimientos para ello.

### **SUGERENCIAS DIDACTICAS:**

Propiciar y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

### **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** Instalación de Clúster de Alta Disponibilidad.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

### **Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- [https://es.wikipedia.org/wiki/Cl%C3%BAster\\_de\\_alta\\_disponibilidad](https://es.wikipedia.org/wiki/Cl%C3%BAster_de_alta_disponibilidad)
- 6.- [https://www.ecured.cu/Cluster\\_de\\_alta\\_disponibilidad](https://www.ecured.cu/Cluster_de_alta_disponibilidad)

[Clúster de Alta Disponibilidad....](#)

# UNIDAD V. USOS Y TENDENCIA DE LOS SISTEMAS OPERATIVOS DISTRIBUIDOS. SISTEMAS OPERATIVOS II.

En el undécimo tema se abordará el cómo se construye un Clúster de Alta Disponibilidad.

No.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
11	Clúster de Alta disponibilidad.	Conocer cómo se construye un Clúster de Alta disponibilidad.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.

## Las Competencias específicas a desarrollar son:

Comprender las tendencias de investigación de los sistemas operativos distribuidos, su administración y el impacto en la sociedad.

### Subtemas.

- 5.1 Administración de sistemas operativos distribuidos: instalación, configuración y control.
- 5.2. Instalación de clúster.
- 5.3. Estándares de administración en sistemas operativos distribuidos.
- 5.4. Cómputo de alto rendimiento a bajo costo.
- 5.5. Súper cómputo basado en clustering como solución a la necesidad de alto procesamiento en la nueva era.
- 5.6. Tendencias de la investigación.
- 5.7. Sistemas distribuidos como infraestructura para el soporte de las empresas en las TI.

# PRÁCTICA No. 11

---

**NOMBRE DE LA PRÁCTICA:** Clúster de Alta disponibilidad.

**OBJETIVOS:** Conocer cómo se construye un Clúster de Alta disponibilidad.

## INTRODUCCIÓN.

### CLUSTER DE ALTA DISPONIBILIDAD.

#### Clúster de alta disponibilidad (High Availability Clúster, HA).

En la actualidad las organizaciones dependen cada vez más de sus sistemas de información, y como es obvio se desea que estos sean seguros y permanezcan disponibles el mayor tiempo posible. Para cualquier institución, una interrupción de sus sistemas de información supone un serio problema.

Efectos por la interrupción de un sistema de información:

- Costos directos asociados a la reparación del sistema de información (piezas a reparar o sustituir, servicios técnicos, etc.).
- Horas de trabajo adicionales para el departamento de sistemas que tiene que reparar la avería.
- Perdidas de productividad o incluso horas de trabajo perdidas por los empleados que dependen del sistema.
- Pérdida de ingresos, por las ventas o servicios que se han dejado de realizar.
- Costos indirectos: satisfacción de los clientes, pérdida de reputación, mala publicidad, desconfianza de los empleados, etc.

La disponibilidad es una medida relativa a la preparación para su utilización de un sistema informático, mientras que la fiabilidad es una medida relativa a su capacidad para mantenerse operativo en el tiempo sin ningún tipo de fallo.

Los fallos potenciales de un sistema son los errores de componentes de hardware, los errores o bloqueos del sistema operativo, los errores de las aplicaciones.

Un Clúster de alta disponibilidad es un conjunto de dos o más máquinas que se caracterizan por mantener una serie de servicios compartidos y por estar constantemente monitoreándose entre sí.

## Tipos de Clúster.

Alta disponibilidad de infraestructura: Si se produce un fallo de hardware en alguna de las máquinas del clúster, el software de alta disponibilidad es capaz de arrancar automáticamente los servicios en cualquiera de las otras máquinas del clúster (failover). Y cuando la máquina que ha fallado se recupera, los servicios son nuevamente migrados a la máquina original (failback). Esta capacidad de recuperación automática de servicios nos garantiza la alta disponibilidad de los servicios ofrecidos por el clúster, minimizando así la percepción del fallo por parte de los usuarios.

Alta disponibilidad de aplicación: Si se produce un fallo del hardware o de las aplicaciones de alguna de las máquinas del clúster, el software de alta disponibilidad es capaz de arrancar automáticamente los servicios que han fallado en cualquiera de las otras máquinas del clúster. Y cuando la máquina que ha fallado se recupera, los servicios son nuevamente migrados a la máquina original. Esta capacidad de recuperación automática de servicios nos garantiza la integridad de la información, ya que no hay pérdida de datos, y además evita molestias a los usuarios, que no tienen por qué notar que se ha producido un problema.

## Disponibilidad.

La disponibilidad es el grado en que una aplicación o servicio está disponible cuándo y cómo los usuarios esperan. La disponibilidad se mide por la percepción de una aplicación del usuario final. Los usuarios finales experimentan frustración cuando sus datos no están disponibles, y ellos no entienden o son capaces de diferenciar los complejos componentes de una solución global. Fiabilidad, valorización, continuas operaciones y detección de errores son características de una solución de alta disponibilidad.

Fiabilidad: Los componentes hardware fiables de una solución de HA, el software fiable, incluida la base de datos, servidores web y aplicaciones, es la parte crítica de una implementación de una solución de alta disponibilidad.

Recuperación: Puede haber muchas opciones para recuperarse de un fracaso si ocurre alguno. Es importante determinar qué tipo de fallos pueden ocurrir en su entorno de alta disponibilidad y la forma de recuperarse de estos fallos en el tiempo que satisface las necesidades comerciales. Por ejemplo, si una tabla importante es eliminada de la base de datos, ¿qué medidas adoptarías para recuperarla? ¿Su arquitectura ofrece la capacidad de recuperarse en el tiempo especificado en un acuerdo de nivel de servicio (SLA)?

Detección de errores: Si un componente en su arquitectura falla, entonces la rápida detección, de dicho componente es esencial en la recuperación de un posible fracaso inesperado. Si bien es posible que pueda recuperarse rápidamente de un corte de luz, si se lleva a otros 90 minutos para descubrir el problema, entonces usted no puede satisfacer su SLA. La monitorización del estado del entorno de trabajo requiere un software fiable, para ver de forma rápida y notificar al administrador de bases de datos (DBA) un problema.

Continuas operaciones: El continuo acceso a sus datos es esencial, por muy pequeño o inexistente que sea el tiempo de caída del sistema, para llevar a cabo las tareas de mantenimiento. Actividades como mover una tabla de un lado a otro dentro de la base de datos, o incluso añadir nuevas CPU's a su hardware debe ser transparente para el usuario final en una arquitectura HA.

Cálculo de la disponibilidad: En un sistema real, si falla uno de los componentes, es reparado o sustituido por un nuevo componente. Si este nuevo componente falla, es sustituido por otro, y así sucesivamente. El componente fijo se considera en el mismo estado que un nuevo componente. Durante su vida útil, uno de los componentes puede ser considerado en uno de estos estados: Funcionando o en Reparación; El estado funcionando indica que el componente está operacional y en Reparación significa que ha fallado y todavía no ha sido sustituido por un nuevo componente.

En caso de defectos, el sistema va de Funcionando en modo Reparación, y cuando se hace la sustitución volverá al estado Funcionando. Por lo tanto, podemos decir que el sistema tiene durante su vida, una media de tiempo para presentar fallas (MTTF) y un tiempo medio de reparación (MTTR). Su tiempo de vida es una sucesión de MTTFs y MTTRs, a medida que este va fallando y siendo reparado. El tiempo de vida útil del sistema es la suma de MTTFs en ciclos MTTF + MTTR ya vividos.

En forma simplificada, se dice que la disponibilidad de un sistema es la relación entre la duración de la vida útil de este sistema y de su tiempo total de vida. Esto puede ser representado por la fórmula de abajo:

$$\text{Disponibilidad} = \text{MTTF} / (\text{MTTF} + \text{MTTR})$$

Porcentaje de disponibilidad	Tiempo de Interrupción Anual	Tiempo de Interrupción Semanal
98 %	7,3 días	3,3 horas
99 %	3,6 días	1,7 horas
99,9 %	8,8 horas	10 minutos
99,99 %	52,5 minutos	1 minuto
99,999 %	5,3 minutos	6 segundos
99,9999 %	31,5 segundos	0,6 segundos

En la evaluación de una solución de Alta Disponibilidad, es importante tener en cuenta si en la medición de MTTF son vistos como fallas las posibles paradas planificadas.

En la actualidad, eligiendo correctamente el hardware y software adecuados, es relativamente sencillo diseñar un sistema con una disponibilidad del 98% del tiempo. Pero el paso del 98% al 99% y de aquí al 99,9999% es una tarea compleja y a la par supone un aumento exponencial del coste total del sistema. En la práctica se alcanza un compromiso entre la disponibilidad pretendida y el costo abordable.

Las razones para implementar un clúster de alta disponibilidad son:

- Aumentar la disponibilidad.
- Mejorar el rendimiento.
- Escalabilidad.
- Tolerancia a fallos.
- Recuperación ante fallos en tiempo aceptable
- Reducir costos.
- Consolidar servidores.
- Consolidar el almacenamiento.
- Configuraciones de alta disponibilidad.

Las configuraciones más comunes en entornos de clúster de alta disponibilidad son la configuración activo/activo y la configuración activo/pasivo.

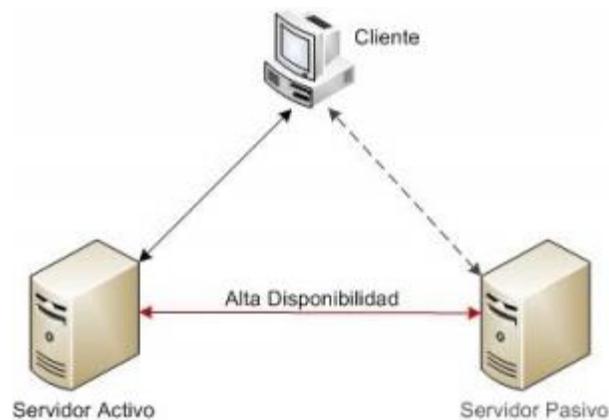


**Ejemplo Alta Disponibilidad  
Configuración Activo/Activo.**

En una configuración activo/activo, todos los servidores del clúster pueden ejecutar los mismos recursos simultáneamente. Es decir, los servidores poseen los mismos recursos y pueden acceder a estos independientemente de los otros servidores del clúster. Si un nodo del sistema falla y deja de estar disponible, sus recursos siguen estando accesibles a través de los otros servidores del clúster.

La ventaja principal de esta configuración es que los servidores en el clúster son más eficientes ya que pueden trabajar todos a la vez. Sin embargo, cuando uno de los servidores deja de estar accesible, su carga de trabajo pasa a los nodos restantes, lo que produce una degradación del nivel global de servicio ofrecido a los usuarios.

En la siguiente figura se muestra como ambos servidores están activos, proporcionando un mismo servicio a los diferentes usuarios. Los clientes acceden al servicio o recursos de forma transparente y no tienen conocimiento de la existencia de varios servidores formando un clúster.



### **Ejemplo Alta Disponibilidad Configuración Activo/Pasivo**

Un clúster de alta disponibilidad, en una configuración activo/pasivo, consiste en un servidor que posee los recursos del clúster y otros servidores que son capaces de acceder a esos recursos, pero no los activan hasta que el propietario de los recursos ya no esté disponible.

Las ventajas de la configuración activo/pasivo son que no hay degradación de servicio y que los servicios solo se reinician cuando el servidor activo deja de responder. Sin embargo, una desventaja de esta configuración es que los servidores pasivos no proporcionan ningún tipo de recurso mientras están en espera, haciendo que la solución sea menos eficiente que el clúster de tipo activo/activo. Otra desventaja es que los sistemas tardan un tiempo en migrar los recursos (failover) al nodo en espera.

Funcionamiento de un clúster de alta disponibilidad.

En un clúster de alta disponibilidad, el software de clúster realiza dos funciones fundamentales. Por un lado, intercomunica entre sí todos los nodos, monitorizando continuamente su estado y detectando fallos. Y por otro lado administra los servicios ofrecidos por el clúster, teniendo la capacidad de migrar dichos servicios entre diferentes servidores físicos como respuesta a un fallo.

Elementos y conceptos básicos en el funcionamiento del clúster.

Recurso y Grupos de Recursos: Tradicionalmente se entiende como servicio a un conjunto de procesos que se ejecutan en un momento dado sobre un servidor y sistema operativo. Este último provee a los procesos de los recursos necesarios para realizar su tarea: sistema de archivos, interfaces de red, tiempo de procesador, memoria, etc.

En un clúster de alta disponibilidad, el software del mismo, abstrae e independiza a los servicios de un host concreto, posibilitando que estos se desplacen entre diferentes servidores de forma transparente para la aplicación o los usuarios.

El software de clúster permite definir grupos de recursos, que son todos aquellos recursos necesarios por dar un servicio. Estos recursos serán los scripts de arranque del servicio, un sistema de archivos, una dirección IP, etc.

Intercomunicación: El software de clúster gestiona servicios y recursos en los nodos; Pero, además, tiene que mantener continuamente entre estos una visión global de la configuración y estado del clúster. De esta forma, ante el fallo de un nodo, el resto conoce que servicios se deben restablecer. Ya que la comunicación entre los nodos del clúster es crucial para el funcionamiento de este, es habitual utilizar un canal específico como una red IP independiente o una conexión en serie, que no se pueda ver afectada por problemas de seguridad o rendimiento.

Heartbeat: El software de clúster conoce en todo momento la disponibilidad de los equipos físicos, gracias a la técnica de heartbeat. El funcionamiento es sencillo, cada nodo informa periódicamente de su existencia enviando al resto una "señal de vida".

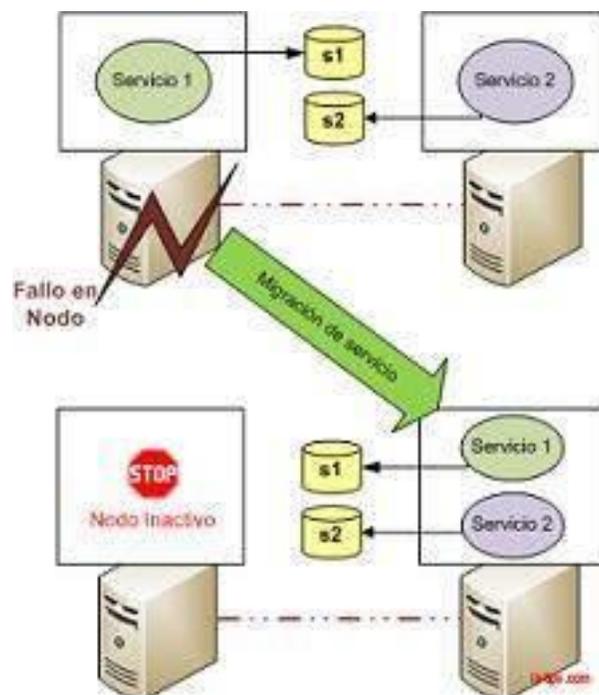
Escenario Split-Brain: En un escenario split-brain, más de un servidor o aplicación pertenecientes a un mismo clúster intentan acceder a los mismos recursos, lo que puede causar daños a dichos recursos. Este escenario ocurre cuando cada servidor en el clúster cree que los otros servidores han fallado e intenta activar y utilizar dichos recursos.

Monitorización de Recursos (Resource Monitoring): Ciertas soluciones de clustering HA permiten no solo monitorear si un host físico está disponible; también pueden realizar seguimientos a nivel de recursos o servicios y detectar el fallo de estos. El administrador puede configurar la periodicidad de estos monitores, así como las acciones a llevar a cabo en caso de fallo.

**Reiniciar Recursos:** Cuando un recurso falla, la primera medida que toman las soluciones de clúster es intentar reiniciar dicho recurso en el mismo nodo. Lo que supone detener una aplicación o liberar un recurso y posteriormente volverlo a activar. Algunas implementaciones no permiten reiniciar un único recurso, y lo que realizan es un reinicio completo de todo un grupo de recursos (servicio). Esto puede llegar a demorar bastante para servicios como las bases de datos.

**Migración de Recursos (Failover):** Cuando un nodo ya no está disponible, o cuando un recurso fallido no se puede reiniciar satisfactoriamente en un nodo, el software de clúster reacciona migrando el recurso o grupo de recursos a otro nodo disponible en el clúster; de este modo el tiempo de inactividad por el posible fallo es mínimo, y el clúster seguirá proporcionando el correspondiente servicio.

**Dependencia entre recursos:** Habitualmente para que el clúster proporcione un servicio, son necesarios no solo un recurso sino varios (ip virtual, sistema de archivos, procesos), lo que se conoce como grupo de recursos. Cuando se arranca o detiene un servicio, sus recursos tienen que activarse en el orden apropiado ya que unos dependen de otros. El software de clúster tiene que permitir definir estas dependencias entre recursos, así como entre grupos.



**Ejemplo de failover en Clúster de Alta Disponibilidad  
Dependencia entre recursos.**

## **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
Servicio de Internet.  
Bibliografía sugerida.  
Proyector.

## **METODOLOGÍA.**

- a) El desarrollo del proyecto contempla las siguientes fases:
- b) Instalación de las computadoras.
- c) Instalación de los sistemas operativos, así como todos los controladores necesarios para la puesta a punto de las computadoras implicadas en el proyecto.
- d) Instalación de software específico.
- e) Instalación de todas las aplicaciones necesarias para la administración de los recursos. Aplicaciones necesarias para montar el servidor web, las bases de datos, configuración del firewall (funcionamiento de la red). Así como todo el software extra para el funcionamiento del clúster.
- f) Configuración de las computadoras. Configurar las diferentes aplicaciones de cada máquina.
- g) Configuración de la red. Configurar la red, asignación de rangos de ip's comunicación entre la red interna y externa, configurar la DMZ (zona desmilitarizada).
- h) El alumno en forma individual, llevará a cabo una investigación de campo, en al menos tres fuentes, que le permita identificar que es un clúster de alta disponibilidad y cuáles son los requerimientos que tienen que cumplir para su implementación; así como sus ventajas y desventajas.
- i) Los resultados de estos trabajos de investigación, deberán presentarse en forma de un reporte individual, de al menos cuatro cuartillas, incluyendo la portada.
- j) Se hará un foro de discusión en el aula, para poder conocer los resultados de las investigaciones; considerando todas las aportaciones para redactar conclusiones.

### **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

Explicar cómo se instala un Clúster de Alta Disponibilidad y cuáles son los requerimientos para ello.

### **SUGERENCIAS DIDACTICAS:**

Propiciar y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

### **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** Instalación de Clúster de Alta Disponibilidad.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

### **Bibliografía preliminar:**

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
4. Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- [https://es.wikipedia.org/wiki/Cl%C3%BAster\\_de\\_alta\\_disponibilidad](https://es.wikipedia.org/wiki/Cl%C3%BAster_de_alta_disponibilidad)
6. <https://www.recerca.cat/bitstream/handle/2072/219063/GutierrezSanmigueIAlfred-ETISa2011-12.pdf?sequence=1>

[Instalación de Clúster de Alta Disponibilidad...](#)

# UNIDAD V. USOS Y TENDENCIA DE LOS SISTEMAS OPERATIVOS DISTRIBUIDOS. SISTEMAS OPERATIVOS II.

**En el duodécimo tema se abordará la Resolución de nombres de dominio.**

No.	Nombre de la Práctica.	Objetivo Específico.	Temas con los que se relacionan.
12	Resolución de nombres de dominio.	Conocer las formas en que trabajan los Servidores de Nombres de Dominio.	Unidad V Usos y tendencias de los Sistemas Operativos Distribuidos.

## **Las Competencias específicas a desarrollar son:**

Comprender las tendencias de investigación de los sistemas operativos distribuidos, su administración y el impacto en la sociedad.

### Subtemas.

- 5.1 Administración de sistemas operativos distribuidos: instalación, configuración y control.
- 5.2. Instalación de clúster.
- 5.3. Estándares de administración en sistemas operativos distribuidos.
- 5.4. Cómputo de alto rendimiento a bajo costo.
- 5.5. Súper cómputo basado en clustering como solución a la necesidad de alto procesamiento en la nueva era.
- 5.6. Tendencias de la investigación.
- 5.7. Sistemas distribuidos como infraestructura para el soporte de las empresas en las TI.

# PRÁCTICA No. 12

**NOMBRE DE LA PRÁCTICA:** Resolución de nombres de dominio.

**OBJETIVOS:** Conocer las formas en que trabajan los Servidores de Nombres de Dominio.

## INTRODUCCIÓN.

¿Qué significa DNS?

Cada equipo conectado directamente a Internet tiene al menos una dirección IP específica. Sin embargo, los usuarios no desean trabajar con direcciones numéricas, como por ejemplo 194.153.205.26, sino con un nombre de dominio o más específicamente, con direcciones (llamadas direcciones FQDN, Fully Qualified Domain Name. Nombre de Dominio Totalmente Calificado) como por ejemplo [/es.ccm.net].

Es posible asociar nombres en lenguaje normal con direcciones numéricas gracias a un sistema llamado DNS (*Domain Name System*, Sistema de Nombres de Dominio). Esta correlación entre las direcciones IP y el nombre de dominio asociado se llama resolución de nombres de dominio (o resolución de direcciones).

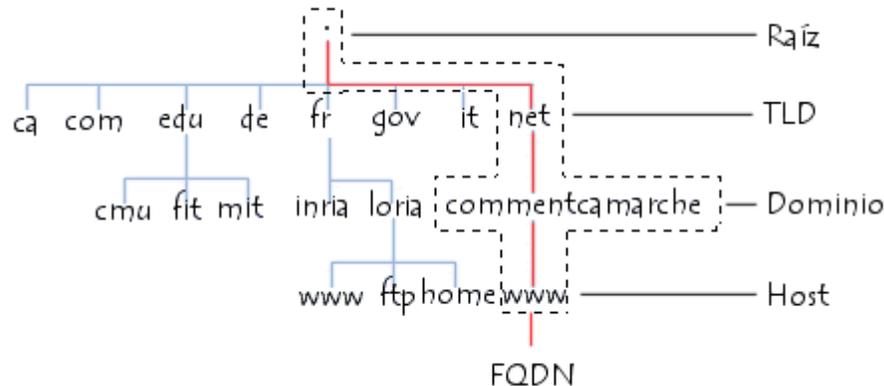
Nombre de "host".

En los inicios del internet (protocolo TCP/IP), como las redes no eran muy extensas, o en otras palabras el número de equipos conectados a la misma red era bajo, los administradores de red crearon archivos llamados tablas de conversión manual. Estas tablas de conversión eran archivos secuenciales, por lo general llamados hosts o hosts.txt, y asociaban en cada línea la dirección IP del equipo con el nombre literal relacionado, denominado nombre de host.

Introducción al sistema de nombres de dominio, el antiguo sistema de tablas de conversión exigía una actualización manual de las tablas para la totalidad de los equipos en caso de incluir o modificar el nombre de una máquina. Con el aumento en tamaño de las redes y sus interconexiones, fue necesario implementar un sistema de administración para los nombres que fuera jerárquico y fácil de administrar. El sistema llamado sistema de nombres de dominio (Domain Name System, DNS) fue desarrollado en noviembre de 1983 por Paul Mockapetris.

Este sistema ofrece un espacio de nombre jerárquico que permite garantizar la singularidad de un nombre en una estructura de árbol, tal como se hace en el sistema de archivos en Unix; un sistema de **servidores de distribución** que permite que el espacio de nombre esté disponible; un sistema de **cliente** que permite "resolver" nombres de dominio, es decir, interrogar a los servidores para encontrar la dirección IP que corresponde a un nombre.

Espacio de nombre, la estructura del sistema DNS se basa en una estructura de árbol en donde se definen los dominios de nivel superior (llamados TLD, Top-Level Domain, dominios de nivel superior); esta estructura está conectada a un nodo raíz representado por un punto.



### Estructura de Árbol de dominios de nivel superior.

Cada nodo del árbol se llama nombre de dominio y tiene una etiqueta con una longitud máxima de 63 caracteres. Por lo tanto, todos los nombres de dominio conforman una estructura de árbol inversa en donde cada nodo está separado del siguiente nodo por un punto (".").

El extremo de la bifurcación se denomina *host*, y corresponde a un equipo o entidad en la red. El nombre de la computadora o dispositivo que se provee debe ser único en el dominio respectivo, o de ser necesario, en el subdominio. Por ejemplo, el dominio del servidor web por lo general lleva el nombre *www*.

La palabra dominio corresponde formalmente al sufijo de un nombre de dominio, es decir, la recopilación de las etiquetas de nodo de la estructura de árbol, con excepción de la computadora.

El nombre absoluto está relacionado con todas las etiquetas de nodo de una estructura de árbol, separadas por puntos y que termina con un punto final que se denomina la dirección FQDN (nombre de dominio totalmente calificado). La profundidad máxima de una estructura arbórea es 127 niveles y la longitud máxima para un nombre FQDN es 255 caracteres. La dirección FQDN permite ubicar de manera única un equipo en la red de redes. Por lo tanto, *es.ccm.net* es una dirección FQDN.

Servidores de nombre de dominio, los equipos llamados servidores de nombres de dominio permiten establecer la relación entre los nombres de dominio y las direcciones IP de los equipos de una red. Cada dominio cuenta con un servidor de nombre de dominio, llamado servidor de nombre de dominio principal, así como

también un servidor de nombre de dominio secundario, que puede encargarse del servidor de nombre de dominio principal en caso de falta de disponibilidad.

Cada servidor de nombre de dominio está especificado en el servidor de nombre de dominio en el nivel superior inmediato, lo que significa que la autoridad sobre los dominios puede delegarse implícitamente. El sistema de nombre es una arquitectura distribuida, en donde cada entidad es responsable de la administración de su nombre de dominio. Por lo tanto, no existe organización alguna que sea responsable de la administración de todos los nombres de dominio.

Los servidores relacionados con los dominios de nivel superior (TLD) se llaman servidores de dominio de nivel superior. Son 13, están distribuidos por todo el mundo y sus nombres van desde *a.root-servers.net* hasta *m.root-servers.net*.

El servidor de nombre de dominio define una zona, es decir, una recopilación de dominios sobre la cual tiene autoridad. Si bien el sistema de nombres de dominio es transparente para el usuario, se deben tener en cuenta algunos aspectos. Cada equipo debe configurarse con la dirección de un equipo que sea capaz de transformar cualquier nombre en una dirección IP. Este equipo se llama servidor de nombres de dominio. Cuando te conectes a Internet, el proveedor de servicios automáticamente modificará los parámetros de tu red para hacer que estos servidores de nombres de dominio estén disponibles. La dirección IP de un segundo servidor de nombres de dominio (servidor de nombres de dominio secundario) debe también definirse: el servidor de nombres de dominio secundario puede encargarse del servidor de nombres de dominio principal en caso de fallas en el sistema.

El servidor que se utiliza con más frecuencia se llama BIND (Berkeley Internet Name Domain). Es un *software* gratuito para sistemas Unix, fue desarrollado inicialmente por la Universidad de Berkeley en California y en la actualidad está mantenido por ISC (Internet Systems Consortium).

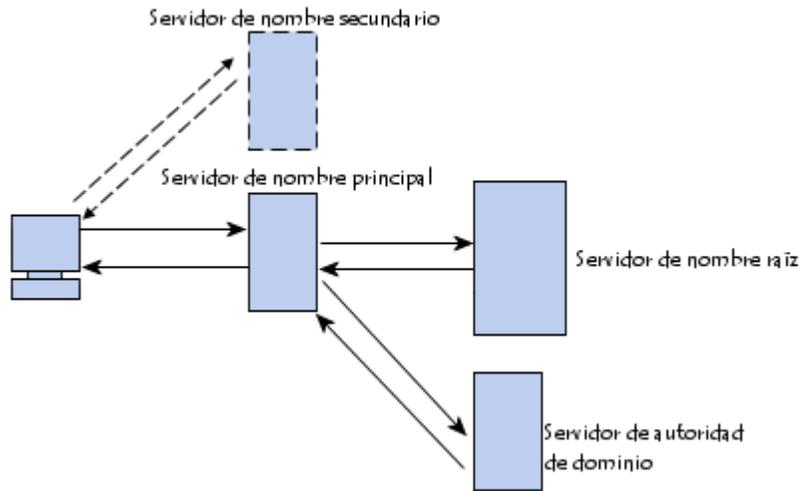
Resolución de nombres de dominio, el mecanismo que consiste en encontrar la dirección IP relacionada al nombre de un equipo se conoce como resolución del nombre de dominio. La aplicación que permite realizar esta operación (por lo general, integrada en el sistema operativo) se llama resolución.

Cuando una aplicación desea conectarse con un *host* conocido a través de su nombre de dominio (por ejemplo, *es.ccm.net*), esta interroga al servidor de nombre de dominio definido en la configuración de su red. De hecho, todos los equipos conectados a la red tienen en su configuración las direcciones IP de ambos servidores de nombre de dominio del proveedor de servicios.

Entonces se envía una solicitud al primer servidor de nombre de dominio (llamado el servidor de nombre de dominio principal). Si este servidor de nombre de dominio tiene el registro en su caché, lo envía a la aplicación; de lo contrario, interroga a un servidor de nivel superior (en nuestro caso un servidor relacionado con el TLD *.net*). El servidor de nombre de nivel superior envía una lista de servidores de nombres de dominio con autoridad sobre el dominio (en este caso, las direcciones IP

de los servidores de nombres de dominio principal y secundario para *es.ccm.net*).

Entonces el servidor de nombres de dominio principal con autoridad sobre el dominio será interrogado y devolverá el registro correspondiente al dominio del servidor (en nuestro caso *www*).



### Servidor de nombres de dominio.

#### Tipos de registro.

Un DNS es una base de datos distribuida que contiene registros que se conocen como RR (Registros de Recursos), relacionados con nombres de dominio. La siguiente información solo es útil para las personas responsables de la administración de un dominio, dado que el funcionamiento de los servidores de nombre de dominio es completamente transparente para los usuarios.

Ya que el sistema de memoria caché permite que el sistema DNS sea distribuido, los registros para cada dominio tienen una duración de vida que se conoce como TTL (Time To Live, tiempo de vida). Esto permite que los servidores intermediarios conozcan la fecha de caducidad de la información y por lo tanto que sepan si es necesario verificarla o no.

Por lo general, un registro de DNS contiene la siguiente información:

Nombre de dominio (FQDN)	TTL	Tipo	Clase	RData
es.ccm.net	3600	A	IN	163.5.255.85

Nombre de dominio: el nombre de dominio debe ser un nombre FQDN, es decir, debe terminar con un punto. En caso de que falte el punto, el nombre de dominio es relativo, es decir, el nombre de dominio principal incluirá un sufijo en el dominio introducido.

Tipo: un valor sobre 16 bits que define el tipo de recurso descrito por el registro. El tipo de recurso puede ser uno de los siguientes:

A: este es un tipo de base que hace coincidir el nombre canónico con la dirección IP. Además, pueden existir varios registros A relacionados con diferentes equipos de la red (servidores).

CNAME (nombre canónico): permite definir un alias para el nombre canónico. Es particularmente útil para suministrar nombres alternativos relacionados con diferentes servicios en el mismo equipo.

HINFO: este es un campo solamente descriptivo que permite la descripción en particular del *hardware* de la computadora(CPU) y del sistema operativo. Generalmente se recomienda no completarlo para evitar suministrar información que pueda ser útil a piratas informáticos.

MX (*Mail eXchange*): es el servidor de correo electrónico. Cuando un usuario envía un correo electrónico a una dirección (user@domain), el servidor de correo saliente interroga al servidor de nombre de dominio con autoridad sobre el dominio para obtener el registro MX. Pueden existir varios registros MX por dominio, para así suministrar una repetición en caso de fallas en el servidor principal de correo electrónico. De este modo, el registro MX permite definir una prioridad con un valor entre 0 y 65.535. NS es el servidor de nombres de dominio con autoridad sobre el dominio. PTR es un puntero hacia otra parte del espacio de nombres del dominio. SOA (*Start Of Authority*, inicio de autoridad en español). El campo SOA permite la descripción del servidor de nombre de dominio con autoridad en la zona, así como la dirección de correo electrónico del contacto técnico (en donde el carácter @ es reemplazado por un punto).

Clase: la clase puede ser IN (relacionada a protocolos de Internet y, por lo tanto, es el sistema que utilizaremos en nuestro caso), o CH (para el sistema caótico).

RDATA: estos son los datos relacionados con el registro. Aquí se encuentra la información esperada según el tipo de registro:

- A: la dirección IP de 32 bits;
- CNAME: el nombre de dominio;
- MX: la prioridad de 16 bits, seguida del nombre del ordenador;
- NS: el nombre del ordenador;
- PTR: el nombre de dominio;
- SOA: varios campos.

Dominios de nivel superior.

Existen dos categorías de TLD (dominios de nivel superior): los dominios que se conocen como "genéricos" y los que se conocen como "nacionales".

Los dominios genéricos, llamados también gTLD (TLD genérico) son nombres de dominio de nivel superior que ofrecen una clasificación de acuerdo con el sector de la actividad. Cada gTLD tiene sus propias reglas de acceso.

Historial de gTLD:

.arpa: relacionado con equipos pertenecientes a la red original.

.com: inicialmente relacionado con empresas con fines comerciales. Sin embargo, este TLD se convirtió en el TLD predeterminado y hasta personas reales pueden adquirir dominios con esta extensión.

.edu: relacionado con las organizaciones educativas.

.gov: relacionado con las organizaciones gubernamentales.

.int: relacionado con las organizaciones internacionales.

.mil: relacionado con las organizaciones militares.

.net: inicialmente relacionado con las organizaciones que administran redes. Con el transcurso de los años este TLD se ha hecho de uso común, de tal forma que, hasta personas reales pueden adquirir dominios con esta extensión.

.org: está normalmente relacionado con organizaciones sin fines de lucro.

Nuevos gTLD presentado en noviembre de 2000 por ICANN:

.aero: relacionado con la industria aeronáutica.

.biz (negocios): relacionado con empresas comerciales.

.museum: relacionado con los museos.

.name: relacionado con el nombre de personas reales o imaginarias.

.info: relacionado con organizaciones que manejan información.

.coop relacionado con cooperativas.

.pro relacionado con profesiones liberales.

gTLD especial:

.arpa: relacionado con las infraestructuras para la administración de redes, también sirve para la resolución inversa de equipos en red y permite hallar el nombre relacionado con una dirección IP.

Los dominios que se conocen como "nacionales", se llaman ccTLD (código de país TLD). El ccTLD está relacionado con los diferentes países y sus nombres refieren a las abreviaturas del nombre del país definidas en la norma ISO 3166. La tabla a continuación resume la lista de ccTLD, para algunos países.

<b>Código</b>	<b>País</b>
AC	Islas Ascensión.
AD	Andorra
AE	Emiratos Árabes Unidos
AF	Afganistán
AG	Antigua y Barbuda
AI	Anguila
AL	Albania
AM	Armenia
AN	Antillas Neerlandesas
AO	Angola
AQ	Antártida
AR	Argentina
AS	Samoa Americana
AT	Austria
AU	Australia
AW	Aruba
AZ	Azerbaiyán
BA	Bosnia y Herzegovina
BB	Barbados
BD	Bangladesh
BE	Bélgica
BF	Burkina Faso
BG	Bulgaria
BH	Bahrein
BI	Burundi
BJ	Benin
BM	Bermudas
BN	Brunei

BO	Bolivia
BR	Brasil
BS	Bahamas
BT	Bhután
BV	Isla Bouvet
BW	Botswana
BY	Bielorrusia
BZ	Belice
CA	Canadá
CC	Islas Cocos
CD	República Democrática del Congo
CCM	República Centroafricana
CG	Congo
CH	Suiza
CI	Costa de Marfil
CK	Islas Cook
CL	Chile
CM	Camerún
CN	China
CO	Colombia
COM	Organización comercial
CR	Costa Rica
CU	Cuba
CV	Cabo Verde
CX	Islas Christmas
CY	Chipre
CZ	República Checa
DE	Alemania
DJ	Djibouti

DK	Dinamarca
DM	Dominica
DO	República Dominicana
DZ	Argelia
EC	Ecuador
EDU	Organización con enlaces relacionados con la educación
EE	Estonia
EG	Egipto
EH	Sahara Occidental
ER	Eritrea
ES	España
ET	Etiopía
EU	Europa
FI	Finlandia
FJ	Fiji
FK	Islas Falkland (Malvinas)
FM	Micronesia
FO	Islas Feroe
FR	Francia
FX	Francia (Territorio EEEE europeo)
GA	Gabón
GB	Gran Bretaña
GD	Granada
GE	Georgia
GF	Guayana Francesa
GG	Guernsey
GH	Ghana
GI	Gibraltar
GL	Groenlandia

GM	Gambia
GN	Guinea
GOV	Organización gubernamental
GP	Guadalupe
GQ	Guinea Ecuatorial
GR	Grecia
GS	Georgia del Sur
GT	Guatemala
GU	Guam (USA)
GW	Guinea Bissau
GY	Guyana
HK	Hong Kong
HM	Islas Heard y McDonald
HN	Honduras
HR	Croacia
HT	Haití
HU	Hungría
ID	Indonesia
IE	Irlanda
IL	Israel
IM	Isla de Man
IN	India
IO	Territorio Británico del Océano Índico
IQ	Iraq
IR	Irán
IS	Islandia
IT	Italia
JM	Jamaica
JO	Jordania

JP	Japón
KE	Kenya
KG	Kirguistán
KH	Camboya
KI	Kiribati
KM	Comoras
KN	Saint Kitts y Nevis
KP	Corea del Norte
KR	Corea del Sur
KW	Kuwait
KY	Islas Caimán
KZ	Kazajstán
LA	Laos
LB	Líbano
LC	Santa Lucía
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lituania
LU	Luxemburgo
LV	Letonia
LY	Libia
MA	Marruecos
MC	Mónaco
MD	Moldova
MG	Madagascar
MH	Islas Marshall
MK	Macedonia

ML	Malí
MIL	Organización militar
MM	Myanmar
MN	Mongolia
MO	Macao
MP	Islas Marianas del Norte
MQ	Martinica
MR	Mauritania
MS	Montserrat
MU	Isla Mauricio
MV	Maldivas
MW	Malawi
MX	México
MY	Malasia
MZ	Mozambique
NA	Namibia
NC	Nueva Caledonia
NE	Níger
NET	Organización con enlaces relacionados con Internet
NF	Isla Norfolk
NG	Nigeria
NI	Nicaragua
NL	Países Bajos
NO	Noruega
NP	Nepal
NR	Nauru
NT	Zona Neutral
NU	Isla Niue
NZ	Nueva Zelanda

OM	Omán
ORG	Organización no específica
PA	Panamá
PE	Perú
PF	Polinesia Francesa
PG	Papua Nueva Guinea
PH	Filipinas
PK	Pakistán
PL	Polonia
PM	San Pedro y Miquelón
PN	Isla Pitcairn
PR	Puerto Rico (USA)
PS	Territorios Palestinos
PT	Portugal
PY	Paraguay
PW	Palau
QA	Qatar
RE	Reunión
RO	Rumania
RU	Federación de Rusia
RW	Rwanda
SA	Arabia Saudita
SB	Islas Solomón
SC	Seychelles
SD	Sudán
SE	Suecia
SG	Singapur
SH	Santa Elena
SI	Eslovenia

SJ	Islas Svalbard y Jan Mayen
SK	República Eslovaca
SL	Sierra Leona
SM	San Marino
SN	Senegal
SO	Somalia
SR	Suriname
ST	Santo Tomé y Príncipe
SU	Unión Soviética
SV	El Salvador
SY	Siria
SZ	Swazilandia
TC	Islas Turcas y Caicos
TD	Chad
TF	Territorios Australes Franceses
TG	Togo
TH	Tailandia
TJ	Tayikistán
TK	Tokelau
TM	Turkmenistán
TN	Túnez
TO	Tonga
TP	Timor Oriental
TR	Turquía
TT	Trinidad y Tobago
TV	Tuvalu
TW	Taiwán
TZ	Tanzania
UA	Ucrania

UG	Uganda
UK	Reino Unido
UM	Islas Periféricas Menores de los Estados Unidos
US	Estados Unidos
UY	Uruguay
UZ	Uzbekistán
VA	Ciudad del Vaticano
VC	San Vicente y las Granadinas
VE	Venezuela
VG	Islas Vírgenes Británicas
VI	Islas Vírgenes de los Estados Unidos
VN	Vietnam
VU	Vanuatu
WF	Islas Wallis y Futuna
WS	Samoa Occidental
YE	Yemen
YT	Mayotte
YU	Yugoslavia
ZA	Sudáfrica
ZM	Zambia
ZR	Zaire
ZW	Zimbabwe

### **MATERIAL Y EQUIPO NECESARIO.**

Computadora.  
 Servicio de Internet.  
 Bibliografía sugerida.  
 Proyector.

## **METODOLOGÍA.**

- a) Los alumnos formarán equipos de hasta cuatro integrantes, para investigar que es un DNS, como funciona, cuáles son los requerimientos para poder llevar a cabo su instalación y cuáles son los pasos para hacerlo.
- b) Tomando en cuenta los requerimientos que se tienen y las limitantes en cuanto a equipo, se procederá a llevar a cabo su instalación, se harán anotaciones de los pasos que se vayan dando y comentarios de los puntos que consideren importantes y al concluir la instalación deberán al menos probar el servidor con datos válidos y reales.
- c) Los resultados de estos trabajos de investigación e instalación del DNS, deberán presentarse en forma de un reporte.
- d) Se hará un foro de discusión en el aula, para poder conocer los resultados de las instalaciones y problemáticas que se les presentaron y como fueron resueltas; se considerarán todas las aportaciones para redactar conclusiones.

### **Preguntas de análisis en el aula: (trabajo individual, equipo y plenaria).**

Explicar que es un DNS y cuáles son los requerimientos para instalarlo y cómo hacerlo.

### **SUGERENCIAS DIDACTICAS:**

Propiciar y fomentar actividades de búsqueda, selección y análisis de información en distintas fuentes, así como actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes, así mismo el facilitador deberá retroalimentar sobre la participación del alumno.

### **REPORTE DEL ALUMNO (Resultados):**

**Presentación de trabajo:** Resolución de nombres de dominio.

**Datos de la portada:** Nombre del Instituto, Nombre del alumno, Carrera, Materia, Horario, Nombre del Maestro y fecha de entrega del trabajo.

## Bibliografía preliminar:

- 1.- Andrew S., Tanenbaum. *Sistemas Operativos Distribuidos*. Ed. Pearson.
- 2.- Goscinsky, *Distributed Operative Systems*.
- 3.- Andrew S., Tanenbaum. *Sistemas Operativos Modernos*. Ed. Pearson.
- 4.- Silberschatz, Abraham y James L., Paterson. *Sistemas Operativos*. Ed. Addison Wesley.
- 5.- [https://es.wikipedia.org/wiki/Sistema\\_de\\_nombres\\_de\\_dominio](https://es.wikipedia.org/wiki/Sistema_de_nombres_de_dominio).

[Resolución de Nombres de Dominio...](#)