

## **Carta al estudiante**

### **Introducción**

Un sistema distribuido es *un conjunto de componentes (hardware y software), localizados en computadoras independientes, que se presenta a sus usuarios como un solo sistema coherente* [1, 2].

Dos aspectos fundamentales sobresalen de esta definición:

1. Un sistema distribuido consiste de componentes (i.e., computadoras o aplicaciones) que son autónomos.
2. Sus usuarios (sean éstos personas o programas) piensan que están interactuando con un único sistema, es decir, el aspecto *distribuido* del sistema es transparente para el usuario.

De lo anterior se deduce que, de una forma u otra, los componentes autónomos necesitan *comunicarse y colaborar* entre ellos a través del *intercambio de mensajes*. Cómo establecer esta colaboración es uno de los aspectos básicos del desarrollo de sistemas distribuidos. Nótese que no se hicieron suposiciones sobre el tipo de computadoras que lo componen; podría tratarse de una colección de supercomputadoras de alto rendimiento, de un grupo de servidores e innumerables máquinas clientes o dispositivos móviles, o de una red de sensores. Tampoco se hicieron suposiciones sobre la forma en que estos componentes están interconectados.

Ejemplos de aplicaciones distribuidas modernas son motores de búsqueda en la Web, sistemas de juegos multiusuarios en línea, redes sociales virtuales y sistemas de intercambio financieros en línea. Entre las principales características de estos sistemas se destacan la concurrencia de sus componentes, la falta de un reloj global y las fallas independientes de sus componentes.

Entre los principales desafíos del desarrollo de sistemas distribuidos podemos mencionar la heterogeneidad de sus componentes, la capacidad de agregar o reemplazar componentes, la seguridad, la escalabilidad, el manejo de fallas, la concurrencia y la transparencia.

### **Objetivo general**

En este curso el alumno logrará conocer las características fundamentales, las tendencias actuales, los modelos básicos, y las principales ventajas y desafíos de los sistemas distribuidos modernos.

## **Objetivos específicos**

Al finalizar el curso el estudiante será capaz de :

1. Comprender los conceptos generales del área de Sistemas Distribuidos.
2. Entender los modelos distribuidos físicos y lógicos fundamentales.
3. Comprender diversos métodos y protocolos para la ejecución remota de procesos.
4. Comprender las diferentes estrategias de coordinación, acuerdo, exclusión mutua y resolución de bloqueos en procesos distribuidos.
5. Implementar algoritmos básicos de comunicación y sincronización de procesos distribuidos.
6. Reconocer los principales problemas de seguridad de los sistemas distribuidos.
7. Conocer las diversas áreas de aplicación de los sistemas distribuidos.
8. Comparar diferentes sistemas de archivos y componentes distribuidos.
9. Conocer las tendencias actuales de investigación en el área de Sistemas Distribuidos.

## **Contenidos**

- Introducción
  - Sistemas distribuidos: definición, objetivos y tipos
- Modelos
  - Físicos y arquitectónicos
  - Modelos fundamentales
- Redes
  - Tipos
  - Principios
  - Protocolos
- Comunicación entre procesos
  - APIs
  - Representación de datos externa y *marshalling*
  - Comunicación *multicast*
  - Virtualización
- Invocación remota
  - Protocolos
  - Remote procedure call
  - Remote method invocation (Java RMI)
- Comunicación indirecta
  - Comunicación en grupo
  - Sistemas de publicación y suscripción
  - Colas de mensajes
  - Enfoques de memoria compartida
- Sincronización: tiempo global y estados globales
  - Relojes, eventos y estados de procesos
  - Sincronización de relojes físicos

- Tiempo lógico y relojes lógicos
- Estados globales
- Debugging distribuido
- Coordinación y acuerdo
  - Exclusión mutua distribuida
  - Comunicación y acuerdo en comunicación de grupo
  - Consenso y problemas relacionados
- Transacciones, control de concurrencia y transacciones distribuidas
  - Transacciones
  - Transacciones anidadas
  - Bloqueos
  - Control de concurrencia optimista
  - Ordenado con marca de tiempo
  - Comparación de métodos de control de concurrencia
  - Transacciones distribuidas planas y anidadas
  - Control de concurrencia en transacciones distribuidas
  - Deadlocks distribuidos
  - Recuperación de transacciones
- Consistencia, replicación y tolerancia a fallas
- Soporte del sistema operativo
  - Protección
  - Procesos e hilos
  - Comunicación e invocación
  - Arquitectura
  - Virtualización al nivel del sistema operativo
- Sistemas de archivos distribuidos
  - Arquitectura de servicios de archivos
- Seguridad
  - Técnicas
  - Algoritmos criptográficos
  - Firmas digitales
- Objetos y componentes distribuidos
  - Objetos distribuidos → CORBA
  - Componentes distribuidos → JavaBeans
- Servicios Web
  - Descripciones y directorios
  - Seguridad XML
  - Coordinación
  - Aplicaciones
- Sistemas *Peer-to-peer*
- Sistemas multiagentes
- Computación ubicua y sistemas móviles
- Sistemas multimedia distribuidos
- Bases de datos distribuidas
- Casos de estudio

## **Evaluación**

- Exámenes: 25%.
- Tareas cortas y quices: 25%.
- Investigación de un tema actual: 25%.
- Proyectos programados: 25%.

## **Bibliografía**

1. Coulouris, G. et al. *Distributed Systems: Concepts and Design*. Fifth Edition. Addison-Wesley, 2011.
2. Tanenbaum, A. & Van Steen, M. *Distributed Systems: Principles and Paradigms*. Second Edition. Pearson - Prentice Hall, 2006.
3. Lynch, N. *Distributed Algorithms*. Morgan Kaufmann, 1996.
4. Jia, W. & Zhou, W. *Distributed Network Systems: From Concepts to Implementations*. Springer, 2005.
5. Ghosh, S. *Distributed Systems: An Algorithmic Approach*. Chapman & Hall, 2007.
6. Kshemkalyani, A. & Singhal, M. *Distributed Computing: Principles, Algorithms, and Systems*. Cambridge University Press, 2008.
7. De Decker, B. et al., Editors. *Advances in Network and Distributed Systems Security*. Kluwer, 2002.
8. Lang, U. & Schreiner, R. *Developing Secure Distributed Systems with CORBA*. Artech House, 2002.
9. Juhasz, Z. et al., Editors. *Parallel Systems: Cluster and Grid Computing*. Springer, 2005.
10. Kacksuk, P. et al., Editors. *Parallel Systems: Cluster and Grid Computing*. Springer, 2007.
11. Puder, A. et al. *Distributed Systems Architecture: A Middleware Approach*. Elsevier, 2006.
12. Shamma, J., Editor. *Cooperative Control of Distributed Multi-Agent Systems*. Wiley, 2007.
13. Muhl, G. et al. *Distributed Event-Based Systems*. Springer, 2006.
14. Vidyarthi, D. et al. *Scheduling in Distributed Computing Systems : Analysis, Design & Models*. Springer, 2009.
15. Kopetz, Hermann. *Real-Time Systems : Design Principles for Distributed Embedded Applications*. Second Edition. Springer, 2011.
16. Tari, Z. & Bukhres, O. *Fundamentals of Distributed Object Systems: The CORBA Perspective*. Wiley, 2001.
17. Van Roy, P. & Haridi, S. *Concepts, Techniques, and Models of Computer Programming -with Practical Applications in Distributed Computing and Intelligent Agents*. 2002.
18. Bacon, J. & Harris, T. *Operating Systems: Concurrent and Distributed Software Design*. Addison-Wesley, 2003.