



Planificaciones

7574 - Sistemas Distribuidos I

Docente responsable: ROCA PABLO DANIEL

OBJETIVOS

Introducir a los alumnos en los paradigmas de los sistemas distribuidos, sus modelos clásicos y fundamentos del área. Incorporar habilidades de diseño y arquitectura para plataformas de middleware focalizando en aspectos de transparencia y tolerancia a fallos.

CONTENIDOS MÍNIMOS

-

PROGRAMA SINTÉTICO

Características de sistemas concurrentes y distribuidos. Paradigmas de nombres, serialización, sincronismo y asincronismo, relojes, orden, coordinación, consistencia. Modelos de sistemas distribuidos. Middlewares orientado a mensajes transientes y persistentes, sincrónicos y asincrónicos. Definición de patrones de comunicación para sistemas distribuidos (request-reply, publisher-subscriber, pipelines, memoria distribuida). Estudio de catálogo usual de aplicaciones distribuidas (distributed file systems, distributed shared memory, name services, time services). Estrategias de diseño para sistemas tolerantes a fallos, escalables e intercomunicables. Sistemas de tiempo real.

PROGRAMA ANALÍTICO

Introducción:

Definición de sistemas distribuidos. Objetivos, arquitecturas y tipos de sistemas distribuidos.

Repaso de programación paralela y concurrente. Repaso de multithreading y multiprocessing. Repaso de comunicaciones entre procesos (IPC). Definición de multicomputing. Concepto de Middleware.

Unidad 1: Herramientas de diseño

Herramientas para diseño y documentación de sistemas distribuidos. Diagramas UML. Diagramas de Comunicación. Diagramas de Arquitectura. Concepto de objeto activo y pasivo. Concepto de interfaces abiertas. Contratos.

Unidad 2: Comunicaciones

Comunicaciones en sistemas distribuidos. Servicios TCP y UDP. Definición de protocolos. Mensajes de longitud fija y variables. Mensajes de texto y binarios. Definición de arquitecturas con capas.

Definición de interfaces de comunicación. Primitivas de servicios con y sin conexión. Análisis de tráfico, distorsión y demora. Canales de comunicación y sincronismo en flujos de datos.

Unidad 3: Paradigmas

Paradigmas de sistemas distribuidos. Nombres y direccionamiento. Pasaje de mensajes. Comunicación de mensajes transientes y persistentes. Comunicaciones de grupos. Tiempo y relojes. Sincronismo. Orden. Estado y cortes del sistema. Coordinación. Consistencia.

Unidad 4: Modelos

Modelo Cliente-Servidor. Modelo Peer-to-Peer. Concepto de binding dinámico. Modelo RPC. Modelo de Distributed Object Communication. Middleware orientado a mensajes (MOM).

Casos de estudio: distributed file systems, distributed shared memory, name services, time services.

Unidad 5: Patrones

Patrones para el diseño de sistemas distribuidos. Modelos asincrónicos y sincrónicos. Memoria distribuida. Request-Reply. Publisher-Subscriber. Pipelines secuenciales y paralelos. Load balancing and routing. Distribución y coordinación de tareas.

Unidad 6: Diseño y Modelado

Modelización de cómputo distribuido. Modelización de procesamiento de datos distribuido. Diseño de arquitecturas distribuidas actuales. Sistemas elásticos. Sistemas de alta disponibilidad.

Unidad 7: Tolerancia a fallos y Tiempo real

Tolerancia a fallos. Concepto de confiabilidad (dependability). Detección de fallas. Paradigmas de tolerancia a fallos. Replicación. Consenso y líderes. Introducción a sistemas de tiempo real.

BIBLIOGRAFÍA

- [1] P. Verissimo, L. Rodriguez: Distributed Systems for Systems Architects, Kluwer Academic Publishers, 2001.
- [2] A. Tanenbaum, M. Van Steen: Distributed Systems. Principles and Paradigms, 2nd Edition. Prentice Hall, 2007.
- [3] G. Coulouris, J. Dollimore, t. Kindberg, G. Blair: Distributed Systems. Concepts and Design, 5th Edition, Addison Wesley, 2012.
- [4] Gomaa, Hassan: Software Modeling & Design. UML, Use cases, pattern & software architectures. Cambridge, 2011.
- [5] Comer, D., Stevens, D.: Internetworking with TCP/IP, Volume 3, Client-Server Programming and Applications, Linux /Posix Sockets Version, Prentice Hall, 2001.
- [6] Ben-Ari, M. Principles of Concurrent and Distributed Programming, 2nd. Ed. Addison Wesley, 2006.
- [7] Garg, V., Elements of Distributed Computing, 1st. Ed. Wiley IEEE Press, 2002.
- [8] Lynch, Nancy: Distributed Algorithms, Morgan Kaufmann, 1996.

- [9] M. Van Steen, A. Tanenbaum: Distributed Systems. 3rd Edition. Pearson Education, 2017.

RÉGIMEN DE CURSADA

Metodología de enseñanza

Exposición teórica de conceptos fundamentales. Resolución metódica de problemas tipo y ensayos sobre objetivos. Análisis, diseño e implementación de soluciones distribuidas por parte de los alumnos bajo dirección y control docentes. Modelización de algoritmos y arquitecturas según las unidades temáticas del programa mediante requerimientos abiertos que motiven la consulta a los docentes y la profundización del conocimiento a través de la bibliografía. Desarrollo de un trabajo práctico final que integre los conceptos aprendidos.

Modalidad de Evaluación Parcial

De manejo de conceptos, aplicación de conocimientos y dominio de técnicas, mediante la respuesta a preguntas y el desarrollo de trabajos prácticos escritos y en computadora. Las instancias de evaluación corresponden a los contenidos de las distintas unidades temáticas. Su evaluación es por presentación en tiempo y forma, método de desarrollo, cumplimiento de estándares de calidad y corrección del resultado.

CALENDARIO DE CLASES

Semana	Temas de teoría	Resolución de problemas	Laboratorio	Otro tipo	Fecha entrega Informe TP	Bibliografía básica
<1> 09/03 al 14/03	Presentación de la materia. Definición de sistemas distribuidos	Repaso de multithreading y multiprocessing				[1, 2, 3, 6]
<2> 16/03 al 21/03	Multicomputing. Concepto de middleware	Práctica de multicomputing en entornos virtualizados				[2, 3]
<3> 23/03 al 28/03	Herramientas de diseño y documentación	Definición de interfaces				[4]
<4> 30/03 al 04/04	Comunicación y protocolos. Arquitecturas por capas	Práctica de comunicación y diseño de arquitecturas				[5, 9]
<5> 06/04 al 11/04	Paradigmas de sistemas distribuidos	Algoritmos de sincronización de procesos				[7, 8]
<6> 13/04 al 18/04	Modelos. Cliente-Servidor, Peer-to-Peer, RPC, Distributed Objects	Definición de MOM				[1, 2, 3]
<7> 20/04 al 25/04	Casos de estudio de sistemas distribuidos típicos	Empleo de MOM en la resolución de un sistema distribuido				[2, 3]
<8> 27/04 al 02/05	Patrones. Request-Reply, Publisher-Subscriber, Pipelines	Practica de distribución y coordinación de tareas mediante MOM y MPI				[1, 9]
<9> 04/05 al 09/05	Diseño y modelado de cómputo y procesamiento de datos distribuidos	Técnicas de diseño para sistemas elásticos y de alta disponibilidad				[1]
<10> 11/05 al 16/05	Diseño de arquitectura distribuidas en la actualidad	Practica de diseño de arquitecturas distribuidas				Apuntes de la cátedra
<11> 18/05 al 23/05	Tolerancia a fallos. Confiabilidad. Replicación. Consenso.	Algoritmos de consenso, elección de lider, fallas bizantinas				[1, 2]
<12> 25/05 al 30/05	Introducción a sistemas de tiempo real	Definición de grupos y proyecto final. Control de los proyectos y consultas				[1]
<13> 01/06 al 06/06		Control de los proyectos y consultas				
<14> 08/06 al 13/06		Control de los proyectos y consultas				
<15> 15/06 al 20/06		Control de los proyectos y consultas				
<16> 22/06 al 27/06		Control de los proyectos y consultas				

CALENDARIO DE EVALUACIONES

Evaluación Parcial

Oportunidad	Semana	Fecha	Hora	Aula
1º				
2º				
3º				
4º				
Observaciones sobre el Temario de la Evaluación Parcial				
Las evaluaciones parciales se establecerán de acuerdo al desarrollo de trabajos prácticos correspondiente a las distintas unidades temáticas.				