

UNIVERSIDAD MAYOR DE SAN SIMÓN  
FACULTAD DE CIENCIAS Y TECNOLOGÍA  
DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS

---

PLAN GLOBAL  
OBJETOS DISTRIBUIDOS

**I. DATOS DE IDENTIFICACIÓN**

Nombre de la materia:	Objetos Distribuidos
Código:	2010128
Grupo:	1
Carga horaria:	4 periodos teórico/prácticos
Materias relacionadas	Taller de Ingeniería de Software Taller de Aplicación en un Ambiente de Desarrollo
Docentes:	Leticia Blanco Coca
Teléfono:	4233719
Correo electrónico:	leticia@memi.umss.edu.bo

**II. JUSTIFICACIÓN**

Desarrollar aplicaciones distribuidas se ha convertido en el mayor desafío para los profesionales en ciencias de computación, debido a que se quiere aprovechar de manera eficaz y eficiente la gran red - INTERNET y sus servicios. Para ello se plantean diversas áreas de acción, empezando en niveles bajos como el diseño de algoritmos que permitan efectivizar las operaciones dentro del marco de la red, hasta aspectos sociales como son los CSCW. Transversalmente, se tiene el desarrollo de tecnologías que permiten implementar sistemas distribuidos, considerando paradigmas de programación, como es la OO, dentro de estas tecnologías, está la de Objetos Distribuidos y sus características además de estándares que permiten transparentar el uso de medios de interconexión e interoperabilidad de los distintos componentes de un sistema. Es así, que dentro de la formación de un profesional en computación, el conocer y aplicar tecnologías para el desarrollo de estas aplicaciones es importante.

Considerando las bases de la POO, una tecnología basada en las primeras formas de interoperación es RMI (basada en conceptos de RPC) que es dependiente de un lenguaje de programación, pero importante para entender el concepto de distribución, la otra es CORBA que se constituye en un estándar que permite definir interfaces de interacción independientes de lenguaje. Esta materia, tiene la finalidad de introducir los conceptos básicos de sistemas distribuidos y aplicar tecnologías RMI y CORBA.

**III. OBJETIVOS**

Los objetivos de esta materia son:

- Identificar los elementos fundamentales de los sistemas distribuidos.
- Introducir los elementos básicos de RMI.
- Conseguir habilidades para enfrentar la solución de problemas utilizando la POO.
- Aplicar RMI al desarrollo de pequeñas aplicaciones.
- Introducir conceptos de CORBA estándar.
- Introducir al lenguaje de definición de interfaces - IDL
- Desarrollar pequeñas prácticas aplicando distintos tipos de IDL's.
- Desarrollar aplicación de mediana escala haciendo uso de CORBA y JAVA.

#### **IV. SELECCIÓN Y ORGANIZACIÓN DE CONTENIDOS**

##### **Unidad 1: Elementos fundamentales de los sistemas distribuidos**

###### **Objetivo de la unidad**

Al final de la unidad, el estudiante será capaz de establecer los fundamentos de un sistema distribuido, considerando las características y los modelos mínimos que se deben tomar en cuenta cuando se quiere desarrollar un sistema de esta naturaleza.

###### **Contenido**

- 1.1** Qué es un sistema distribuido?
- 1.2** Arquitecturas de sistemas distribuidos.
- 1.3** Modelos fundamentales de los sistemas distribuidos.
- 1.4** Comunicación en un sistema distribuido.

##### **Unidad 2: RMI**

###### **Objetivo de la unidad**

Al final de la unidad, el estudiante tendrá la capacidad de desarrollar una pequeña aplicación haciendo uso de la invocación a métodos remotos - RMI.

## **Contenido**

- 2.1** Elementos de un RMI.
- 2.2** Definición de interfaces.
- 2.3** Aplicaciones Cliente y Servidor.
- 2.4** Compilación de aplicaciones con RMI.

## **Unidad 3: Conceptos de CORBA estándar**

### **Objetivo de la unidad**

Al final de la unidad, el estudiante será capaz de reconocer los elementos básicos de CORBA y su utilidad.

## **Contenido**

- 3.1** CORBA.
- 3.2** Elementos de CORBA estándar.
- 3.3** Pasos para desarrollar una aplicación basada en CORBA.
- 3.4** Invocación remota.
- 3.5** Interoperabilidad en CORBA estándar.

## **Unidad 4: Introducción al lenguaje de definición de interfaces - IDL**

### **Objetivo de la unidad**

Al final de la unidad, el estudiante será capaz de definir IDL's, haciendo uso correcto de la sintaxis.

## **Contenido**

- 4.1** Elementos lexicos de IDL.
- 4.2** Tipos de IDL.
- 4.3** Constantes IDL.
- 4.4** Excepciones.
- 4.5** Declaración de interfaces.
- 4.6** Declaración de módulos
- 4.7** Ejercicios.

## **Unidad 5: Mapeo de IDL a Java**

### **Objetivo de la unidad**

Al final de la unidad, el estudiante reconocerá el mapeo de los distintos elementos de un IDL en clases Java. Además de experimentar con ejercicios simples de conversión.

### **Contenido**

- 5.1** Consideraciones generales.
- 5.2** Nombres.
- 5.3** Mapeo para tipos de datos básicos.
- 5.4** Mapeo para tipos de datos especiales.
- 5.5** Mapeo para interfaces.
- 5.6** Mapeo para atributos.
- 5.7** Mapeo para operaciones.
- 5.8** Otras clases de colección más sofisticadas.
- 5.9** Ejercicios.

## **Unidad 6: Elementos importantes del ejecutor ORB**

### **Objetivo de la unidad**

Al final de la unidad, el estudiante será capaz de identificar elementos importantes para poder utilizar CORBA, distinguiendo las distintas interfaces que se utilizan en la ejecución de una aplicación CORBA.

### **Contenido**

- 6.1** Inicializando aplicaciones CORBA.
- 6.2** Pseudo interface CORBA::ORB.
- 6.3** POA.
- 6.4** Pseudo interface CORBA::OBJECT.
- 6.5** Pseudo interface CORBA::TYPECODE.
- 6.6** DII - DSI.
- 6.7** Servant.
- 6.8** Ejercicios.

## **Unidad 7: Prácticas con CORBA**

### Objetivo de la unidad

Al final de la unidad, el estudiante tendrá la capacidad de aplicar el estándar CORBA en el desarrollo de una aplicación de mediana envergadura.

### Contenido

**7.1** Planteamiento de práctica.

**7.2** Generación de objetos remotos.

**7.3** Análisis de alternativas de implementación.

## V. METODOLOGÍAS

Clases magistrales que permitan introducir conceptos y fundamentos.

Clases en laboratorio, para poner en práctica los elementos de RMI y CORBA. El estudiante en clases tiene la obligación de probar y realizar sus propias adaptaciones a elementos de desarrollo y definición de interfaces.

## VI. CRONOGRAMA O DURACIÓN EN PERIODOS ACADÉMICOS POR UNIDAD

El cronograma y avance de materia se registrará por la siguiente secuencia:

UNIDAD	DURACIÓN Hs. ACADÉMICAS	DURACIÓN SEMANAS
Elementos fundamentales de sistemas distribuidos	4 Hrs	1 sem
RMI	6 Hrs	1 1/2 sem
Conceptos de CORBA estándar	8 Hrs	2 sem
Introducción al lenguaje de definición de interfaces	12 Hrs	3 sem
Mapeo de IDL a Java	12 Hrs	3 sem
Elementos importantes del ejecutor ORB	6 Hrs	1 1/2 sem
Prácticas con CORBA	24 Hrs	6 sem

## VII. CRITERIOS DE EVALUACIÓN

- Evaluación diagnóstica. Al principio del semestre es necesario nivelar a los estudiantes respecto a conceptos de POO. Para ello se hace un repaso de los conceptos principales, en base a la participación de los estudiantes respondiendo a preguntas y respuestas que el profesor realiza. Del mismo modo en clases acerca de material ya avanzado.
- Evaluación formativa. Durante el semestre y por cada unidad el estudiante debe entregar trabajos que apliquen los conceptos introducidos.

- Evaluación sumativa. La materia consta de un examen escrito que tienen un ponderado de 20 % en la nota final, así mismo los trabajos que se realizan en el transcurso del semestre son de carácter obligatorio que tendrán un ponderado de 30 % y el desarrollo de la práctica con un ponderado del 50 %.

## VIII. BIBLIOGRAFÍA

- *Implementing Distributed Systems with Java and CORBA*, Markus Aleksy, Axel Korthaus, Martin Schader. Springer. 2005
- *Distributed Systems Concepts and Design*, George Colouris, Jean Dollimore, Tim Kindberg. Addison Wesley. 4ta, Edición. 2005.
- *Programación Avanzada en CORBA con C++*, Michi Henning, Steve Vinoski. Addison Wesley. 2002
- *RMI Documentation*. <http://java.sun.com/products/jdk/rmi/index.html>
- *CORBA: documents and specifications*. <http://www.omg.org/technology/documents/index.html>