



Tema 14. Sistemas Distribuidos de Tiempo Real

1. Introducción
2. La arquitectura
3. Cuestiones de diseño
4. La comunicación
5. El DDS

La mayor parte está basada en material original de Alejandro Alonso Muñoz y Juan Antonio de la Puente. DIT (UPM).

La parte de DDS está basada en material de RTI, Inc.



Introducción



Introducción

- Seguimos pensando en sistemas con fuertes requisitos de tiempo, pero ahora,
- trataremos con sistemas en los que la capacidad de proceso está distribuida entre varios ordenadores interconectados
- ¿Por qué sistemas distribuidos?
 - sistemas físicamente distribuidos
 - requisitos de procesamiento
 - fiabilidad y seguridad: tolerancia de fallos
- Los SDTR son sistemas muy complejos



Características

- Nos centraremos en sistemas débilmente acoplados
 - la comunicación se basa en el paso de mensajes
 - el tiempo de comunicación es significativo
- Ejemplos:
 - sistemas multimedia
 - SCADA
 - aviónica
 - fabricación automatizada
 - robótica
- Es necesario proponer soluciones viables:
 - hacer suposiciones realistas
 - tener en consideración el estado de la técnica



La Arquitectura de los Sistemas Distribuidos de Tiempo Real

Tema 14

Sistemas Distribuidos de Tiempo Real

5



La arquitectura de los SDTR

- Consiste en un conjunto de nodos conectados por redes de comunicación
- Aunque nos centramos en sistemas débilmente acoplados
 - cada nodo puede ser un multiprocesador conectado por un *bus*
- Normalmente, son sistemas de una gran complejidad
- Se suelen estructurar jerárquicamente, para tratar:
 - plazos de respuesta de diversas escalas
 - diferentes tipos de información
 - requisitos variados de procesador

Tema 14

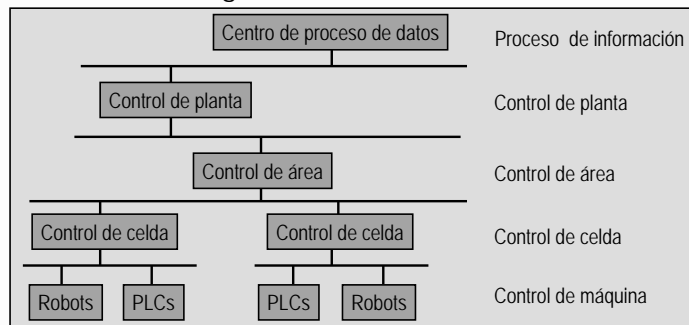
Sistemas Distribuidos de Tiempo Real

6



Sistemas de fabricación

- CIM: sistema integrado de fabricación
 - automatización, integración y flexibilidad
- Requisitos temporales más estrictos cuanto menor es el grado de abstracción



Tema 14

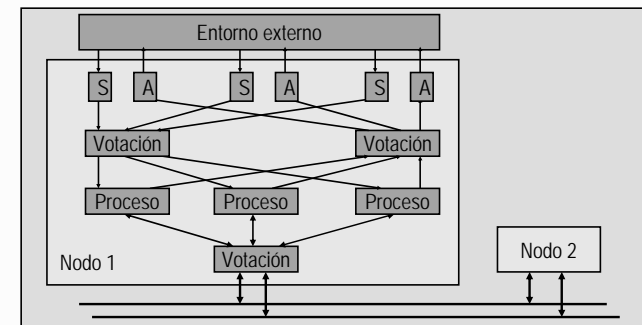
Sistemas Distribuidos de Tiempo Real

7



Sistemas de aviónica

- Tienen requisitos de seguridad críticos
- Necesitan tolerar fallos hardware y software
- Utilizan replicación y métodos de votación



Tema 14

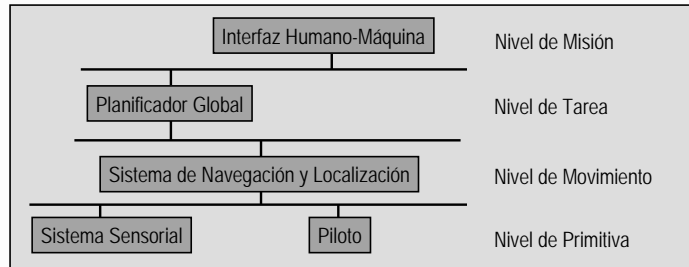
Sistemas Distribuidos de Tiempo Real

8



Sistemas de robótica

- La arquitectura refleja los niveles de control involucrados
- Cada nivel trata información con una granularidad diferente
- Arquitectura del sistema de control de un robot:



Tema 14

Sistemas Distribuidos de Tiempo Real

9



Cuestiones de diseño en Sistemas Distribuidos de Tiempo Real

Tema 14

Sistemas Distribuidos de Tiempo Real

10



Cuestiones de diseño de los SDTR

- Distribución de la carga
- Existencia de plazos de respuesta globales
 - extremo a extremo
- Sincronización de relojes
- Planificación y análisis temporal

Tema 14

Sistemas Distribuidos de Tiempo Real

11



Distribución de la carga

- Asignación de tareas a procesadores:
 - estática: al diseñar el sistema (lo habitual)
 - dinámica: en tiempo de ejecución
- La distribución inadecuada de la carga puede:
 - llevar a infrautilizar los recursos
 - impedir el cumplimiento de los plazos
- Variados criterios de asignación:
 - los procesadores tienen recursos limitados
 - necesidad de replicación de tareas
 - requisitos de utilización de recursos específicos
 - distribución geográfica
- Problema muy complejo:
 - más fácil con asignación estática
 - DMS y EDF son óptimos en cada nodo
 - hay que tener en cuenta el tiempo de transmisión de los mensajes

Tema 14

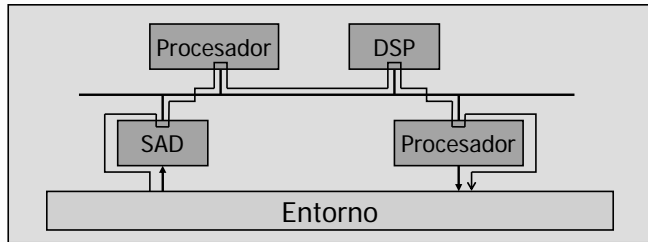
Sistemas Distribuidos de Tiempo Real

12



Plazos de respuesta globales

- Plazo de respuesta: asociado a una transacción
 - Transacción: conjunto de actividades relacionadas que se sincronizan/comunican mediante mensajes
- Tiempo de respuesta:
 - Desde la lectura hasta la actuación



Tema 14

Sistemas Distribuidos de Tiempo Real

13



Sincronización de relojes

- Los procesadores deben tener una visión global del tiempo
 - pero la deriva de los relojes obliga a sincronizarlos periódicamente
- La diferencia entre los valores locales del tiempo de observación de un mismo evento en diferentes procesadores debe estar acotada
 - ¿Con qué precisión? (Granularidad del tiempo)
 - dependiente de cada aplicación
- La sincronización de los relojes no debe degradar el rendimiento del sistema

Tema 14

Sistemas Distribuidos de Tiempo Real

14



Planificación y análisis temporal

- Para poder analizar el tiempo de respuesta global de una transacción hay que conocer:
 - los tiempos de cómputo máximo de cada tarea que interviene en la transacción
 - los tiempos de transmisión máximos de los mensajes
- Garantizar estos plazos \Rightarrow acotar los tiempos de transmisión
 - hay que usar protocolos de comunicación deterministas
 - los mensajes tienen atributos temporales (período, plazo)
 - la planificación de los mensajes también es importante
 - se debe considerar el tiempo de transmisión completo
 - desde que una tarea envía un mensaje hasta que otra lo recibe
 - se deben incluir los tiempos de espera en colas, búferes, etc.

Tema 14

Sistemas Distribuidos de Tiempo Real

15



La comunicación en los Sistemas Distribuidos de Tiempo Real

Tema 14

Sistemas Distribuidos de Tiempo Real

16



Comunicación en SDTR

- CSMA/CD no son adecuados para STR
 - en general, no pueden acotar el tiempo de transmisión de cada mensaje individual
 - Pero se puede superponer un protocolo de acceso determinista (por ejemplo TDMA) sobre una red IEEE 802.3
- Mejor opción: usar un protocolo de comunicación determinista:
 - TDMA (time-division multiple access)
 - Paso de testigo temporizado
 - Protocolos de acceso basados en prioridades

Tema 14

Sistemas Distribuidos de Tiempo Real

17



TDMA : acceso múltiple por división del tiempo

- Equivalente del ejecutivo cíclico para la planificación del medio de comunicación
 - el envío de los mensajes es sincrónico
 - todos los procesadores tienen relojes sincronizados entre sí
 - el acceso al medio se divide en rodajas de tiempo
 - cada procesador tiene asignadas rodajas específicas durante las cuales puede enviar sus mensajes
- La planificación del medio es estática
 - no puede haber colisiones
 - el tiempo máximo de transmisión está acotado
- Este protocolo es determinista \Rightarrow predecible
 - Sin embargo, es poco flexible y difícil de mantener y
 - puede desaprovechar el ancho de banda de la red
- Se usa en TTA (Time-Triggered Architecture)
 - para sistemas de alta integridad con planificación cíclica
 - tolerancia de fallos por redundancia activa

Tema 14

Sistemas Distribuidos de Tiempo Real

18



Paso de testigo temporizado: *Timed Token Protocol*

- Se usa con redes rápidas: FDDI
- El ancho de banda se asigna cíclicamente
- Cada nodo puede enviar mensajes durante cierto tiempo cuando tiene el testigo
 - el tiempo de envío debe ser suficiente para enviar los mensajes dentro de su plazo
 - no puede haber colisiones
- El tiempo de rotación del testigo está acotado

Tema 14

Sistemas Distribuidos de Tiempo Real

19



Acceso al medio gobernado por prioridades

- Los mensajes tienen asociada una prioridad
- Cuando varios procesadores intentan transmitir, hay un mecanismo que resuelve la contienda a favor del mensaje de mayor prioridad
- Hace falta un intervalo de prioridades suficiente
- Ejemplo: CAN

Tema 14

Sistemas Distribuidos de Tiempo Real

20



El CAN Bus

- **CAN: Controller Area Network**
 - Creado por Bosch (originalmente para automóviles)
 - Normalizado como ISO 11898
 - cubre los 2 niveles inferiores de OSI
 - medio físico: par trenzado
 - longitud máxima de los mensajes: 64 bits (8 bytes útiles)
- **No hay direccionamiento convencional:**
 - Los mensajes se difunden a todos los nodos
 - cada uno selecciona los que le interesan
 - tienen un *identificador* que, además, indica la *prioridad*
 - 11 bits: 2047 valores de prioridad
 - y permite la resolución dinámica de colisiones
- Es posible aplicar el análisis de planificabilidad de sistemas con prioridades estáticas

Tema 14

Sistemas Distribuidos de Tiempo Real

21



Características del CAN-Bus

- El bus CAN (Controller Area Network) es un protocolo de comunicaciones en serie, de radiado y sensible a portadora
- Orientado a la transmisión y recepción de mensajes pequeños en ambientes ruidosos, con un alto nivel de integridad de los datos
- Velocidad máxima: 1Mbps (redes hasta 40m)
- Distancia máxima: 1km (hasta 50kbps)
- La resolución de la contienda se basa en la prioridad de los mensajes
- El identificador describe el contenido del mensaje y su prioridad
- Consistencia de los datos: todos los nodos reciben los mismos datos

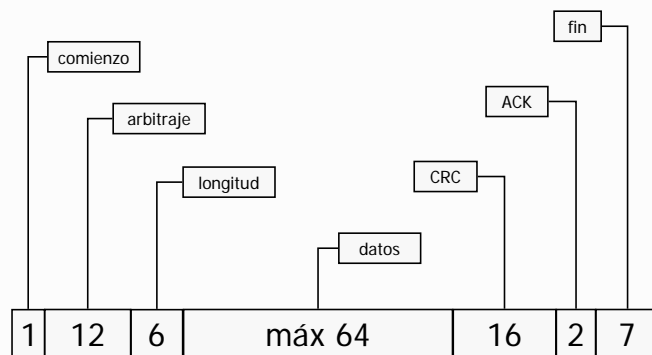
Tema 14

Sistemas Distribuidos de Tiempo Real

22



Estructura de los mensajes



Tema 14

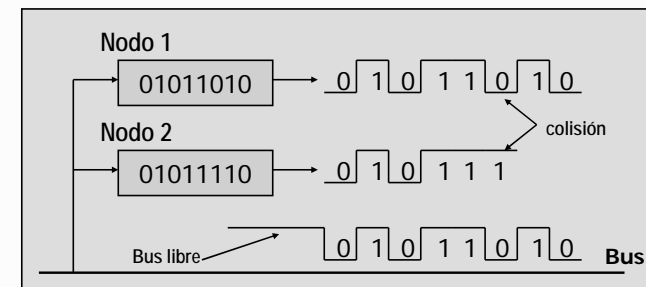
Sistemas Distribuidos de Tiempo Real

23



Arbitraje del Bus

- En la resolución de las colisiones se emplea el campo de arbitraje: identificador y bit de petición remota
- Si un nodo envía un bit recesivo (1) y lee un bit dominante (0), hay un nodo transmitiendo con más prioridad



Tema 14

Sistemas Distribuidos de Tiempo Real

24



Análisis de la planificabilidad. 1

- Se puede aplicar el análisis de tiempo de respuesta para planificación con prioridades fijas
 - pero se planifica el uso del bus, no del procesador
 - no hay expulsión durante la transmisión
 - se modela como un bloqueo de duración igual al tiempo de transmisión del mensaje más largo: $B_{BUS} = 13$ ms.
- Se supone que:
 - el controlador del bus intenta transmitir siempre que haya mensajes pendientes
 - los mensajes se generan al principio del período y
 - no hay errores de transmisión

	Tareas	Mensajes
T	Período	Período
C	Tiempo de cómputo	Tiempo de transmisión
D	Plazo de respuesta	Plazo de respuesta
B	Bloqueo por región crítica ocupada	Bloqueo por Bus ocupado
R	Tiempo de respuesta de caso peor	Tiempo de envío de caso peor

Tema 14

Sistemas Distribuidos de Tiempo Real

25



Análisis de la planificabilidad. 2

- Tiempo de respuesta de caso peor de un mensaje:

$$R_i = C_i + B_{BUS} + \sum_{j \in hp(i)} \left\lceil \frac{R_j}{T_j} \right\rceil \cdot C_j$$

- La ecuación se resuelve por iteración simple
- Si $R_i \leq D_i$
 \Rightarrow el mensaje M_i es planificable

Tema 14

Sistemas Distribuidos de Tiempo Real

26



El Bus CAN y la industria

- Desde 1989 se dispone de integrados de bajo coste que implementan el nivel de enlace de datos
 - hay más de 50 controladores de 15 fabricantes
- Se usa en muchos automóviles de fabricantes europeos
- Intensa actividad de normalización de usos y extensiones a los sectores médico, industrial y marítimo
- En la primavera de 1999: más de 150 millones de nodos CAN instalados

Tema 14

Sistemas Distribuidos de Tiempo Real

27



Planificación holística

- Se trata de integrar el análisis de tiempo de respuesta de los procesadores y de la red de comunicaciones:
 - las tareas se planifican con prioridades fijas en cada procesador
 - la red se planifica con prioridades (CAN) o con otro método determinista (TDMA, paso de testigo)
- Se considera un *modelo lineal* suceso-respuesta
 - un suceso da lugar a una respuesta formada por una secuencia de *sucesos internos* y *acciones* (tareas o mensajes)
 - el conjunto constituye una *transacción*

Tema 14

Sistemas Distribuidos de Tiempo Real

28



El DDS

Tema 14

Sistemas Distribuidos de Tiempo Real

29



DDS. Introducción

- El *OMG Data Distribution Service* es una especificación emergente
 - para sistemas de distribución de datos según el modelo de publicación-suscripción
 - conecta a productores de información (publicadores)
 - con consumidores (suscriptores)
 - todos ellos anónimos y
 - desacoplados en tiempo, espacio y flujo
 - proporciona un modelo independiente de la plataforma
 - orientado a sistemas distribuidos de tiempo real
 - configuración mediante calidades de servicio (*QoS*)

Tema 14

Sistemas Distribuidos de Tiempo Real

30



DDS. Modelo en capas

- *DCPS* (Publicación/Suscripción Data-Céntrica)
 - Distribuye los datos
 - Modela el servicio por medio de estas entidades:
 - Participante (*DomainParticipant*)
 - Tema (*Topic*)
 - Escritor (*DataWriter*)
 - Publicador (*Publisher*)
 - Lector (*DataReader*)
 - Suscriptor (*Subscriber*)
 - Todas las entidades se pueden configurar por medio de *QoS*
 - El consumo de datos puede ser:
 - sincrónico: basado en espera (*Wait*)
 - asíncrónico: basado en escucha (*Listener*)
- *DLRL* (Capa de Reconstrucción Local de Datos)
 - construye una cache de objetos locales que permite a las aplicaciones acceder a los datos "como si fueran locales"

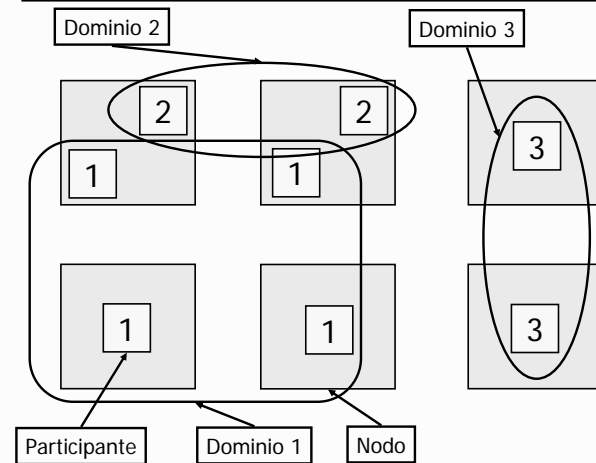
Tema 14

Sistemas Distribuidos de Tiempo Real

31



Dominios y participantes



Tema 14

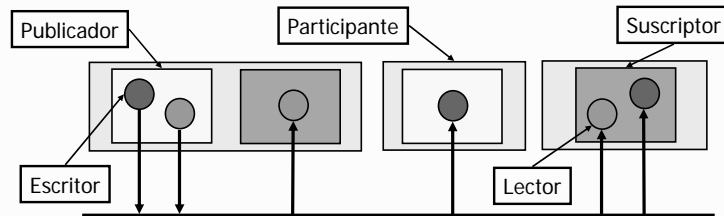
Sistemas Distribuidos de Tiempo Real

32



Entidades del DCPS

- Participante (*DomainParticipant*):
 - representa la participación de la aplicación en el colectivo de comunicación
- Escritor (*DataWriter*):
 - Acceso para escribir datos tipados sobre un Tema particular
- Publicador (Publisher):
 - Agregación de Escritores. Responsable de diseminar la información
- Lector (*DataReader*):
 - Acceso para leer datos tipados relativos a un Tema específico
- Suscriptor (*Subscriber*):
 - Agregación de Lectores. Responsable de recibir información



Tema 14

Sistemas Distribuidos de Tiempo Real

33



DDS. Temas

- Los Temas (*Topics*) se usan para asociar los Lectores a los Escritores:
 - Descrito por una cadena. Ej: "Presion"
 - Enlazado a un tipo de datos. Ej: `struct SensorAnalogico { ... };`
 - Representa una colección de objetos de datos distinguibles por medio de una clave definida por la aplicación. Ej:

```
struct SensorAnalogico {
    string id_sensor; // clave
    float valor; //datos del sensor
}
```

```
id_sensor = "Horno#32"
valor = 211
```

```
id_sensor = "Horno#17"
valor = 312
```

- Los Escritores se enlazan (al crearlos) a un único Tema
- Los Lectores se enlazan (al crearlos) a uno o más Temas
 - Topic, ContentFilteredTopic, MultiTopic
 - Los 2 últimos proporcionan los medios para suscripciones basadas en contenido

Tema 14

Sistemas Distribuidos de Tiempo Real

34



El DDS y la Calidad de Servicio

- QoS a la medida de la distribución de datos en los sistemas de tiempo real:
 - Fiabilidad (Reliability: Best_Effort, Reliable)
 - permite el uso de transportes eficientes para datos repetitivos
 - Plazo (Deadline)
 - establece contratos acerca de la tasa a la que se refrescan los datos repetitivos
 - Presupuesto de latencia (Latency_Budget)
 - establece las guías para obtener retardos aceptables
 - Límites a los recursos (Resource_Limits)
 - controla los recursos usados por el servicio
 - Filtro basado en tiempo (Time_Based_Filter)
 - desacopla a productores rápidos de consumidores lentos
 - Historia (History: Keep_Last, Keep_All)
 - soporta las semánticas de los datos de estado
 - Durabilidad (Durability: Volatile, Transient, Persistent)
- La compatibilidad entre lo solicitado y lo ofrecido es comprobada por el Servicio.

Tema 14

Sistemas Distribuidos de Tiempo Real

35



El DDS y CORBA

- Buscan ajustar la lógica de mediación al problema

CORBA

- *Objetos* Distribuidos
 - Cliente/servidor
 - Invocación de métodos remotos
 - Transporte fiable
- Mejor para:
 - proceso de ordenes remotas
 - transferencia de ficheros
 - transacciones síncronas

DDS

- *Datos* distribuidos
 - Publicación/suscripción
 - Envío múltiple de datos
 - QoS configurable
- Mejor para:
 - rápida diseminación a/de múltiples nodos
 - redes dinámicas
 - requisitos de entrega flexibles

Tema 14

Sistemas Distribuidos de Tiempo Real

36



El DDS y la industria

- **Televisión digital**
 - múltiples fuentes de vídeo en tiempo real
- **Buque LPD-17 San Antonio**
 - (LPD = baja probabilidad de detección)
 - múltiples fuentes de datos (algunas transitorias)
- **Imagen por Resonancia Magnética**
 - sustituir al bus multiprocesador por la red
- **Radar HiPer-D (ARPA)**
 - (HiPer-D: Programa de computación distribuida de altas prestaciones)
 - coordinar cientos de ordenadores en múltiples barcos
- **Sistema de aterrizaje autónomo en portaaviones**
 - multicast para sensores y procesadores redundantes