

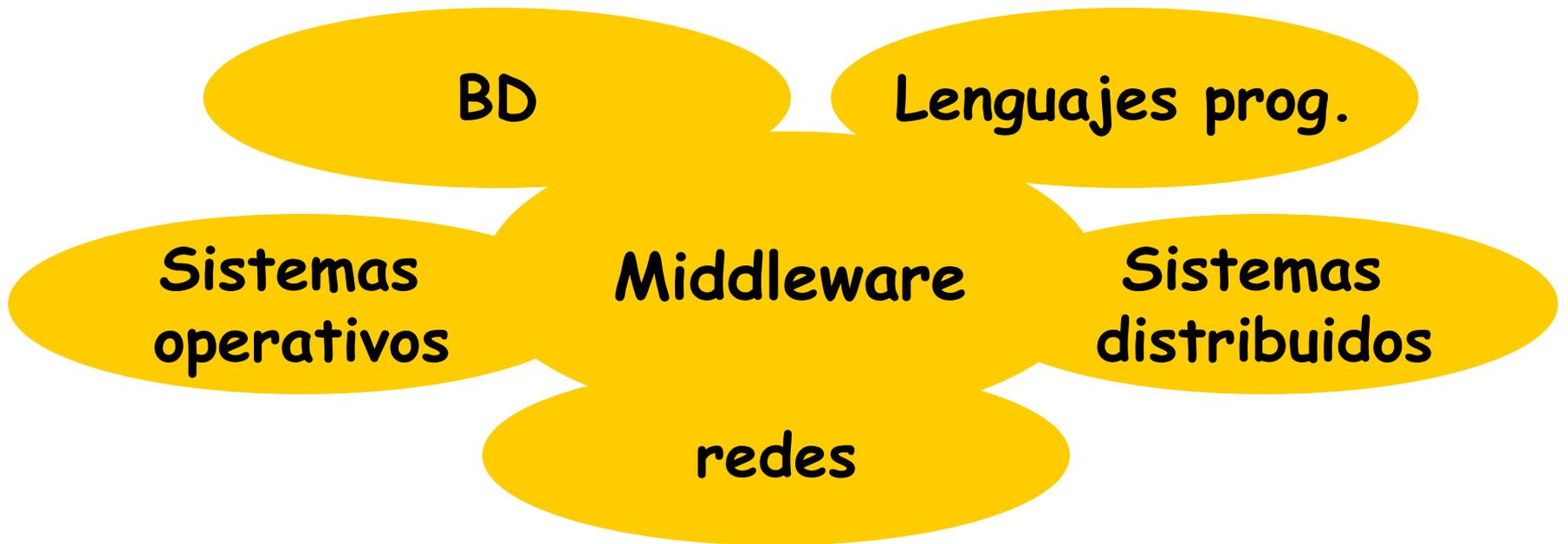
AI

**Sistemas Distribuidos y AI:
Middleware y Conciencia del
Contexto**

Middleware

- Es un software de conectividad, que junto a los mecanismos de comunicación de las aplicaciones, **permiten crear transparencia, escalabilidad e interoperabilidad** en redes distribuidas .
- Su tarea es la de **asistir a las aplicaciones y usuarios en sus procesos de comunicación**, logrando esconder la gran heterogeneidad inmersa en la red en que se encuentran.
- Residen en una **capa intermedia**, construida sobre el sistema operativo de un gran numero de plataformas de hardware.

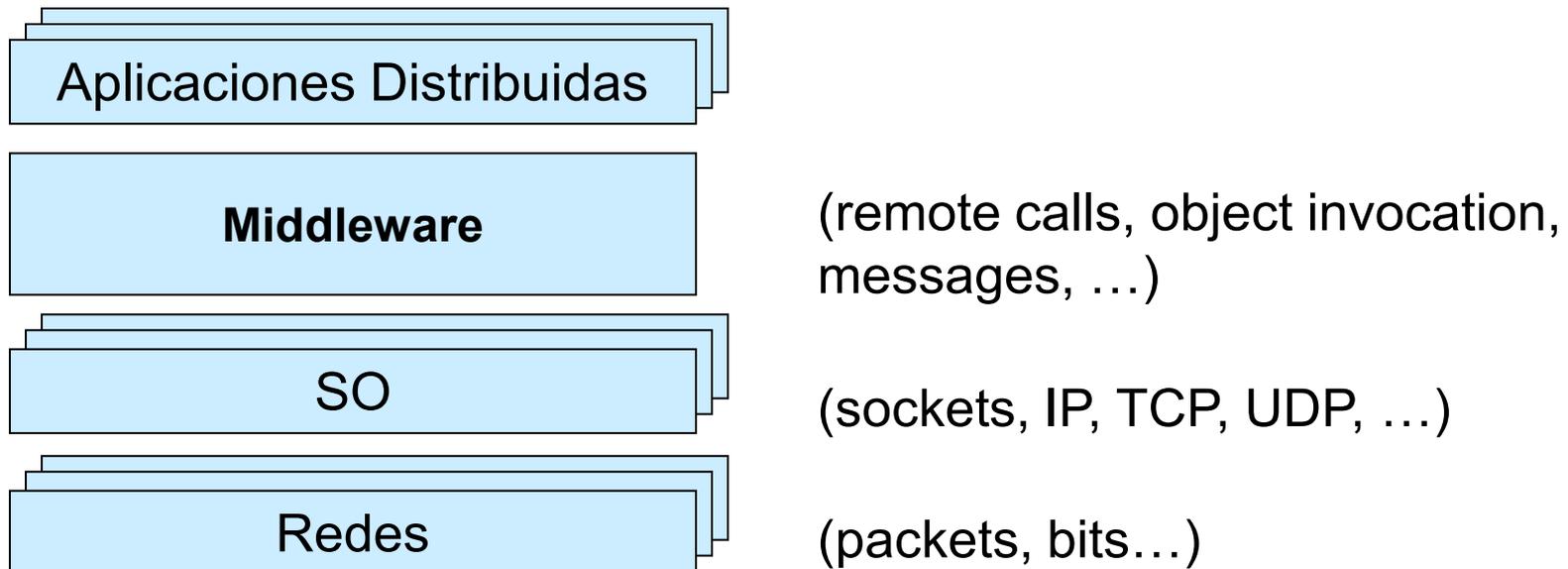
Donde esta el Middleware ?



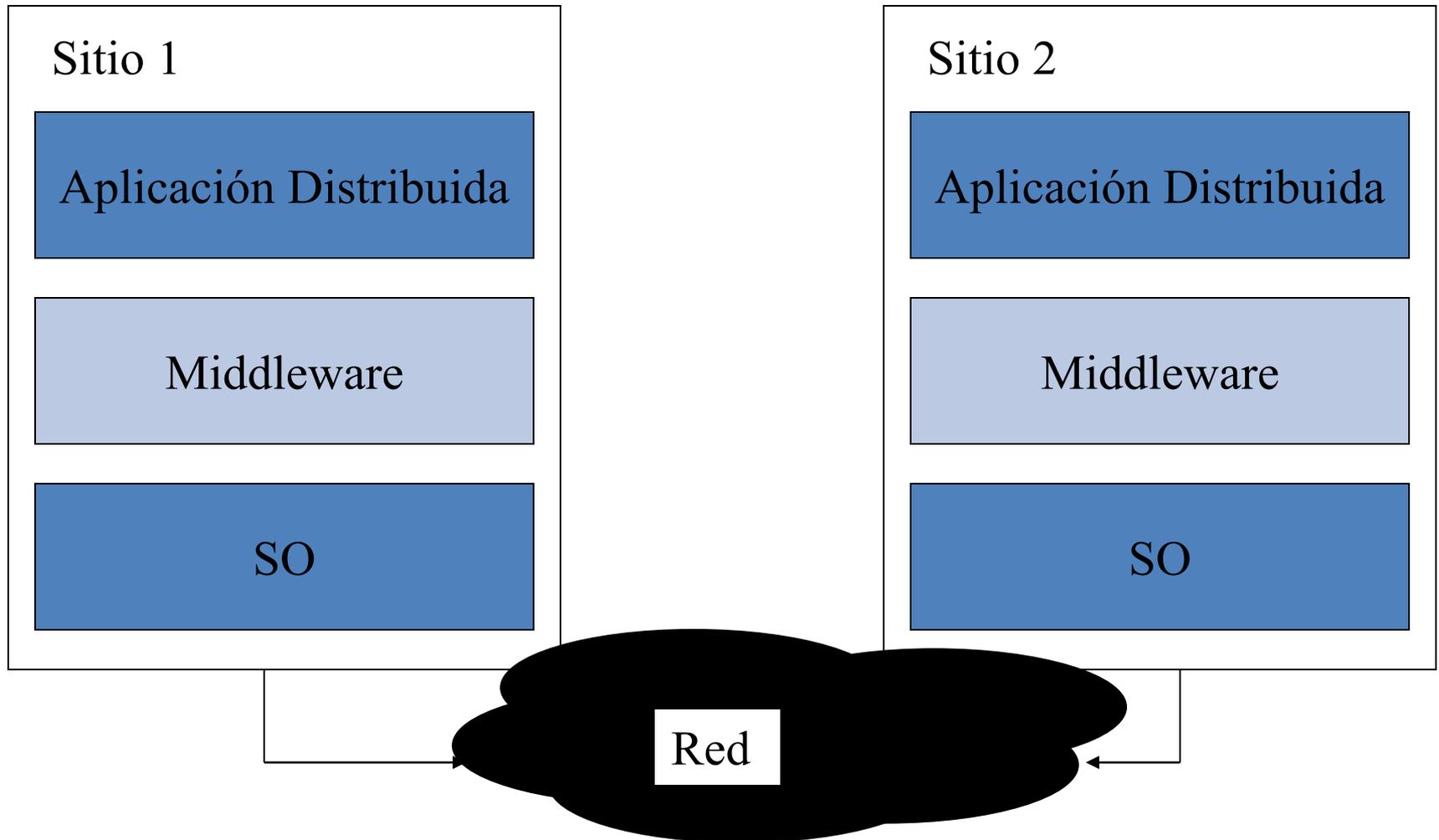
Disperso en muchas disciplinas

Middleware

- **Capa entre** sistema operativo y las aplicaciones distribuidas
- **Puente entre** las comunicaciones del sistema operativo de bajo nivel y las abstracciones del lenguaje de programación
- Proporciona una **abstracción de programación común y de la infraestructura** para las aplicaciones distribuida



Middleware



Middleware

Los Middlewares son diseñados para

- reducir la complejidad de una red de computadores,
- permitir la comunicación de las aplicaciones con los sistemas operativos presentes, y
- hacer disponibles cualquier recurso que se encuentre dentro de ella.

Middleware

Un Middleware permite

- Esconder **la heterogeneidad** de los sistemas: componentes de hardware, sistemas operativos y protocolos de comunicación
- Soportar modelos avanzados de **coordinación** entre entidades distribuidas,
- Realizar **la distribución** de los cálculos tan transparente como sea posible.
- Proveer una **interfaz estándar de alto nivel** para el desarrollo e integración de aplicaciones, de tal manera que una aplicación pueda ser fácil de reusar, portar y mantener comunicación con otras aplicaciones.
- Suministrar un **conjunto de servicios** comunes para el desarrollo de funciones de propósito general, en orden de evitar la duplicación de esfuerzos y facilitar la colaboración entre las aplicaciones

Middleware

- **Da soporte:**
 - Nombramiento, ubicación, descubrimiento servicio, réplica
 - Protocolo de manejo, fallas de comunicación, QoS
 - Sincronización, concurrencia, transacciones, almacenamiento
 - control de acceso, autenticación
- **Dimensiones:**
 - Solicitud/Respuesta vs mensajería asíncrona
 - Lenguaje Específico vs independiente del lenguaje
 - Propietario vs basado en estándares
 - Pequeña escala vs. gran escala
 - fuertemente conectados vs acoplamiento flexible

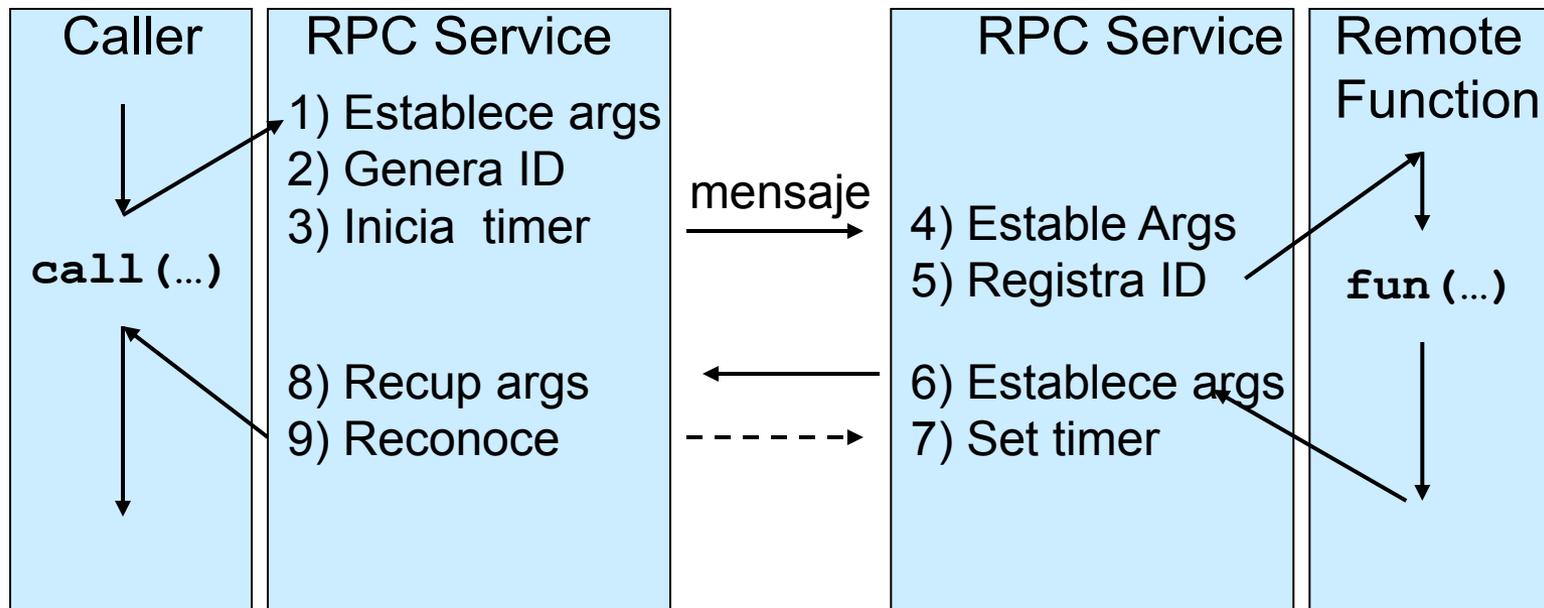
Tipos middleware

Como abstraccion para desarrollar sistemas distribuidos

1. Remote Procedure Call
2. Object-Oriented Middleware
3. Message-Oriented Middleware
4. Event-Based Middleware

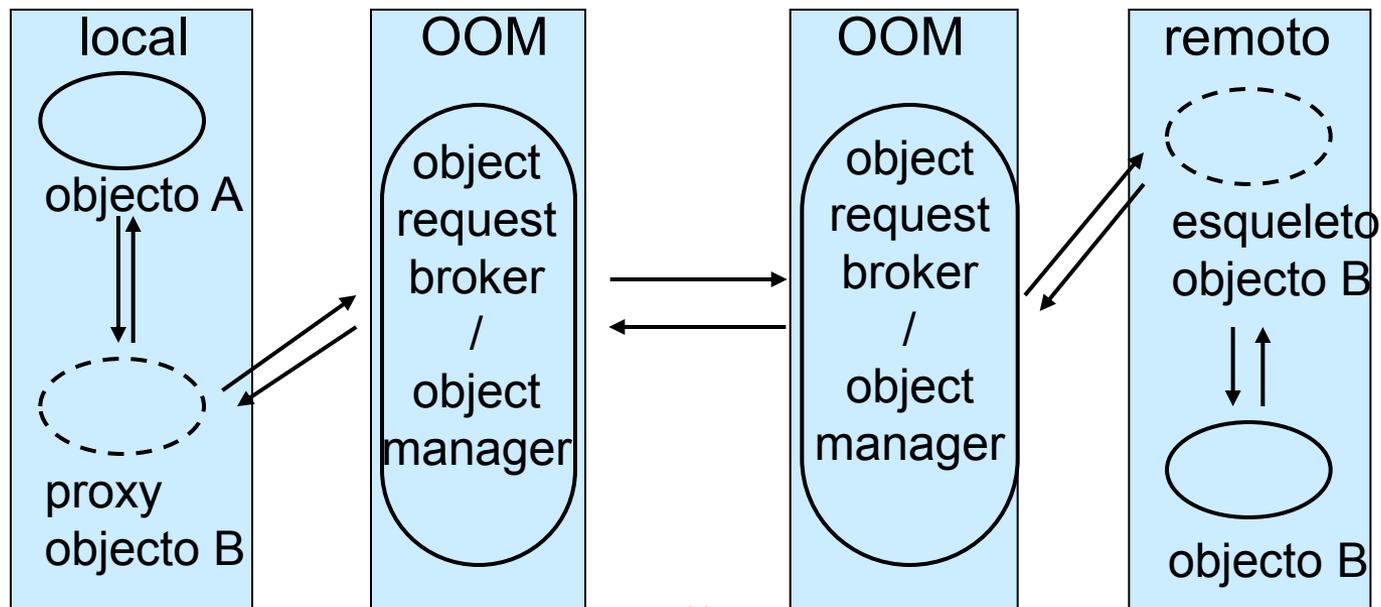
Remote Procedure Call (RPC)

- Esconde localmente remote function calls
- Modelo Cliente/servidor
- Paradigma Request/reply implementado con pase de mensaje en un servicio RPC
- Parámetros de función y retorna valores

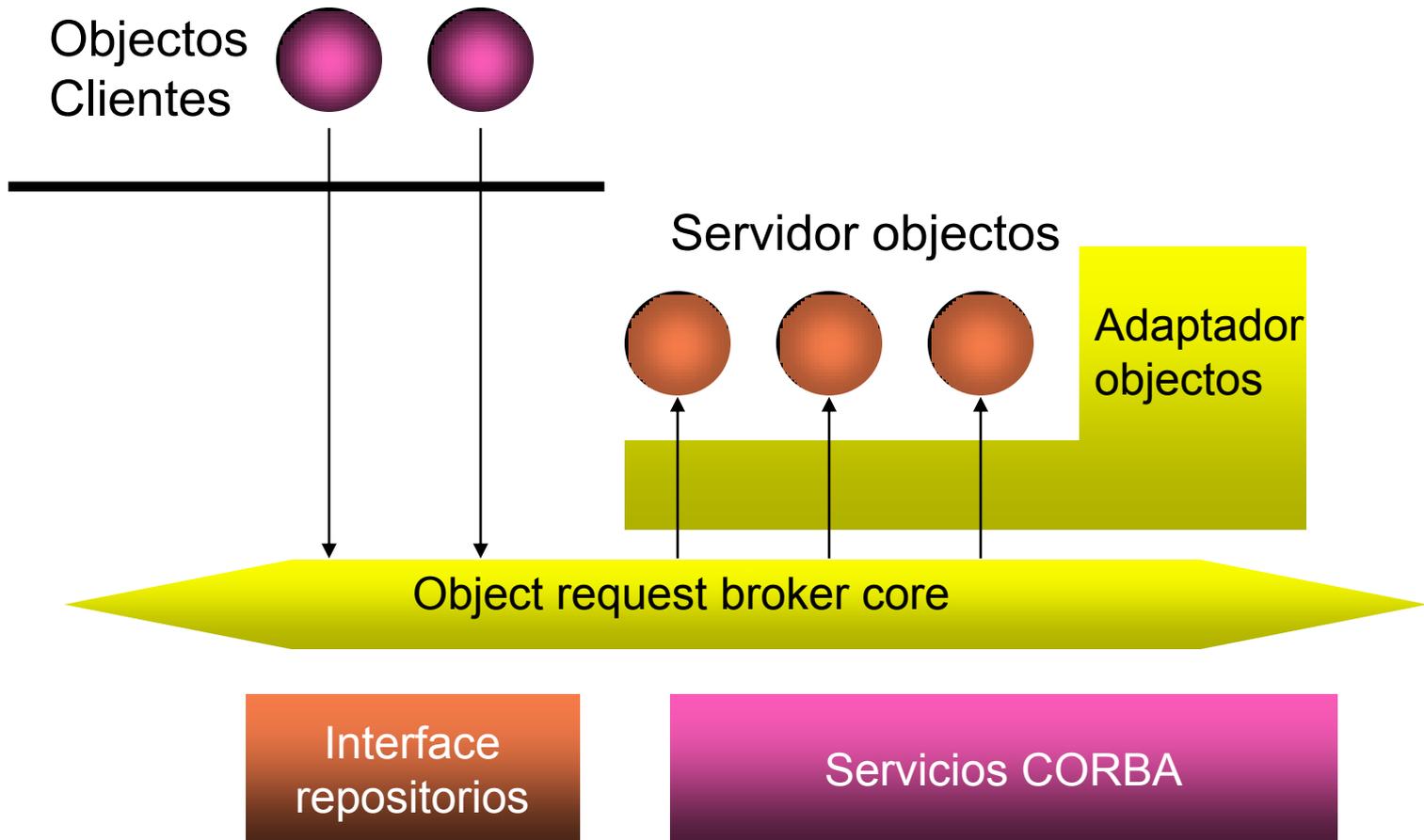


Object-Oriented Middleware (OOM)

- **Objetos** *locales o remoto*
- **referencias** **Objetos** *locales o remoto*
- Objetos Remotos visible por **remotas interfaces**
- Se esconden remotos objetos usando localmente **proxy de los objetos**
- **Remote method invocation (RMI)**

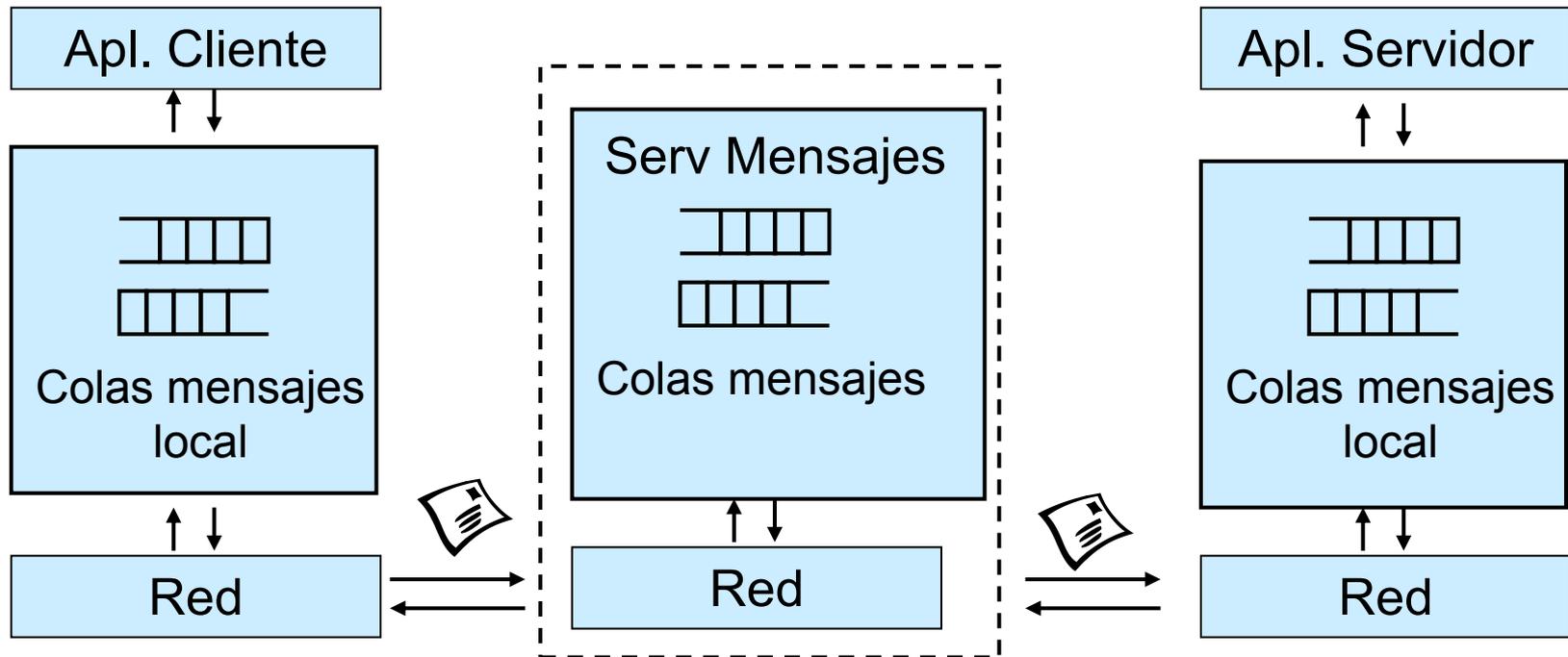


Arquitectura CORBA



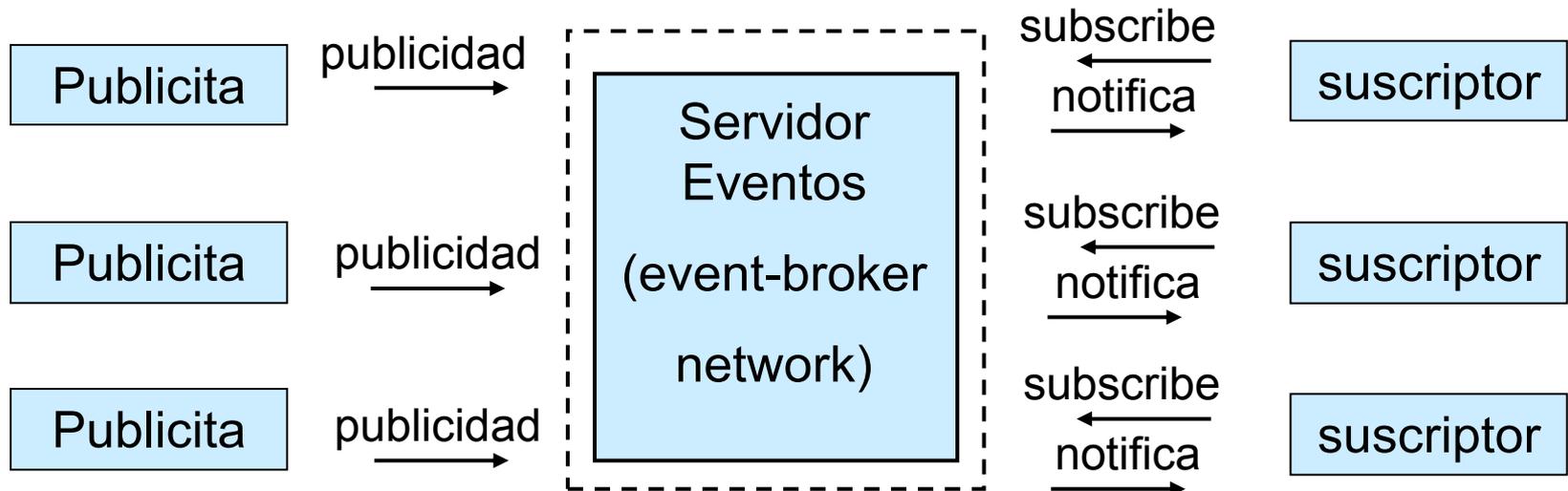
Message-Oriented Middleware (MOM)

- Comunicación usando **mensajes**
- Mensajes guardados **en colas**
- **Servidor de mensajes** entre cliente y servidor



Event-Based Middleware

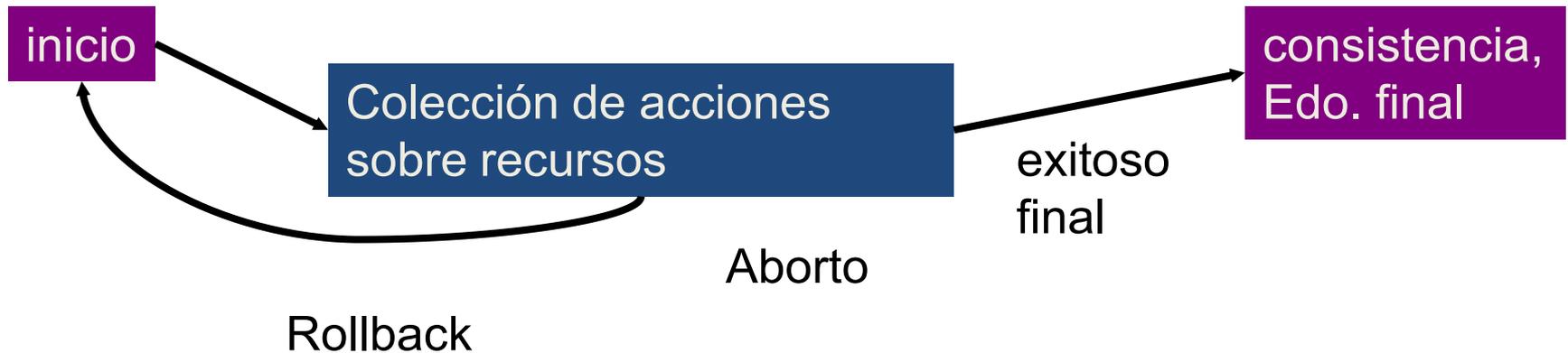
- **Editores** (publicidad): publican eventos (mensajes)
Suscriptores: expresan interés en los eventos con las suscripciones
Servicio de eventos notifica a los suscriptores interesados de los eventos publicados
Los eventos pueden tener un contenido arbitrario (pares de nombre/valor)



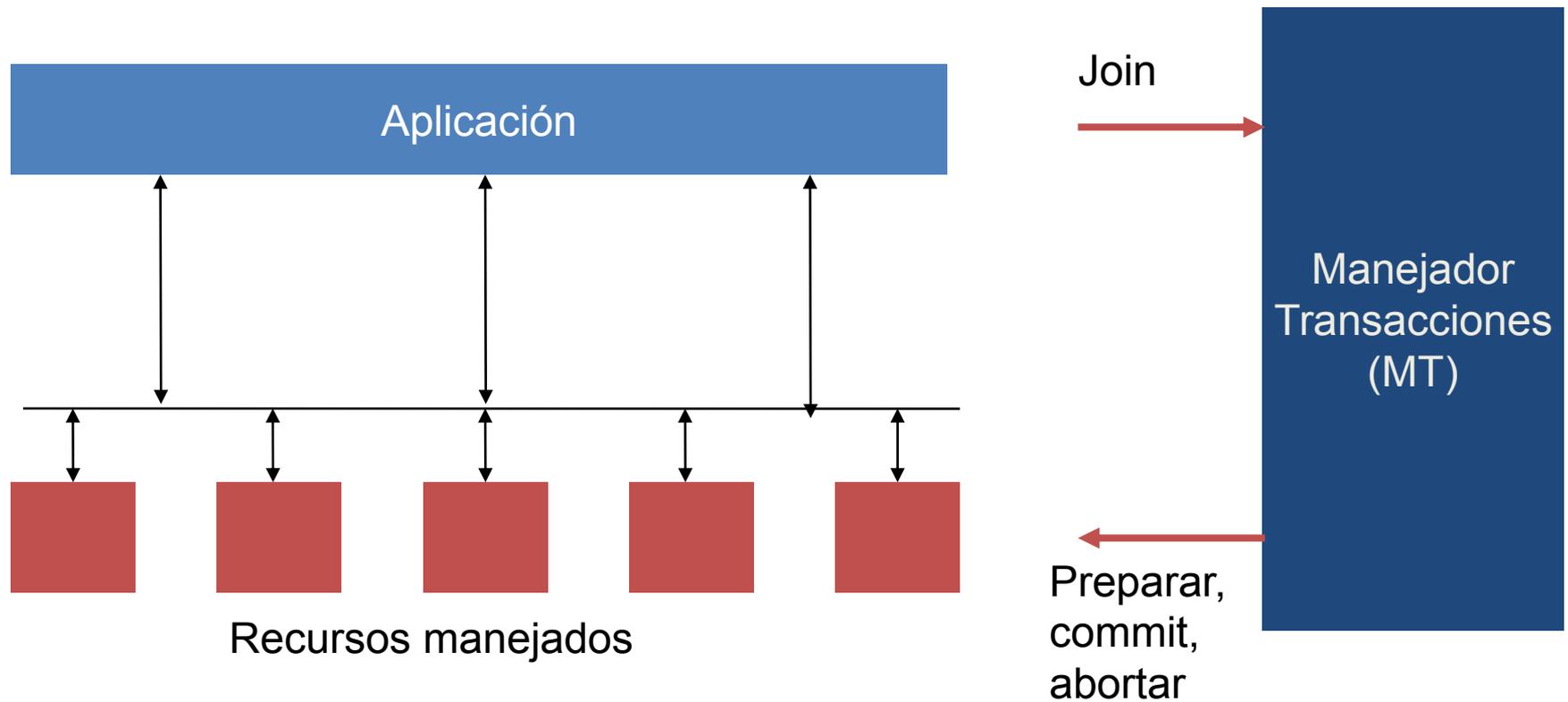
Tipos middleware

- **Procesamiento de transacciones**
 - Simplificar la coordinación de directores de recursos
- **Middleware orientado a mensajes**
 - Gestión de mensaje y de capacidades de colas,
- **Gestión de objetos distribuidos**
 - Gestión de objetos que se distribuyen a través de plataformas y organizaciones heterogéneas
- **Código móvil**
 - Permitir código de la aplicación se mueva y se ejecuta en plataformas heterogéneas

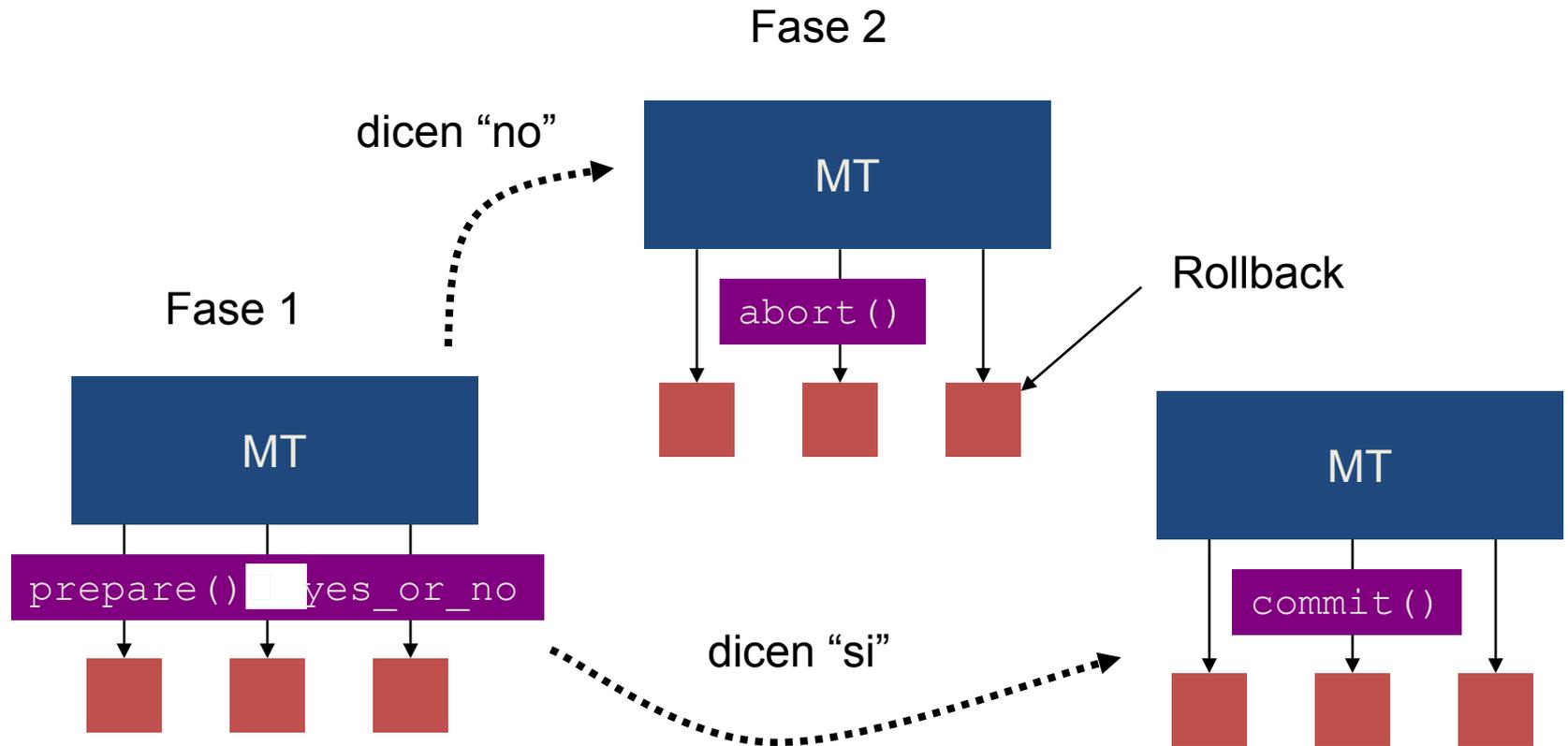
Procesamiento de transacciones



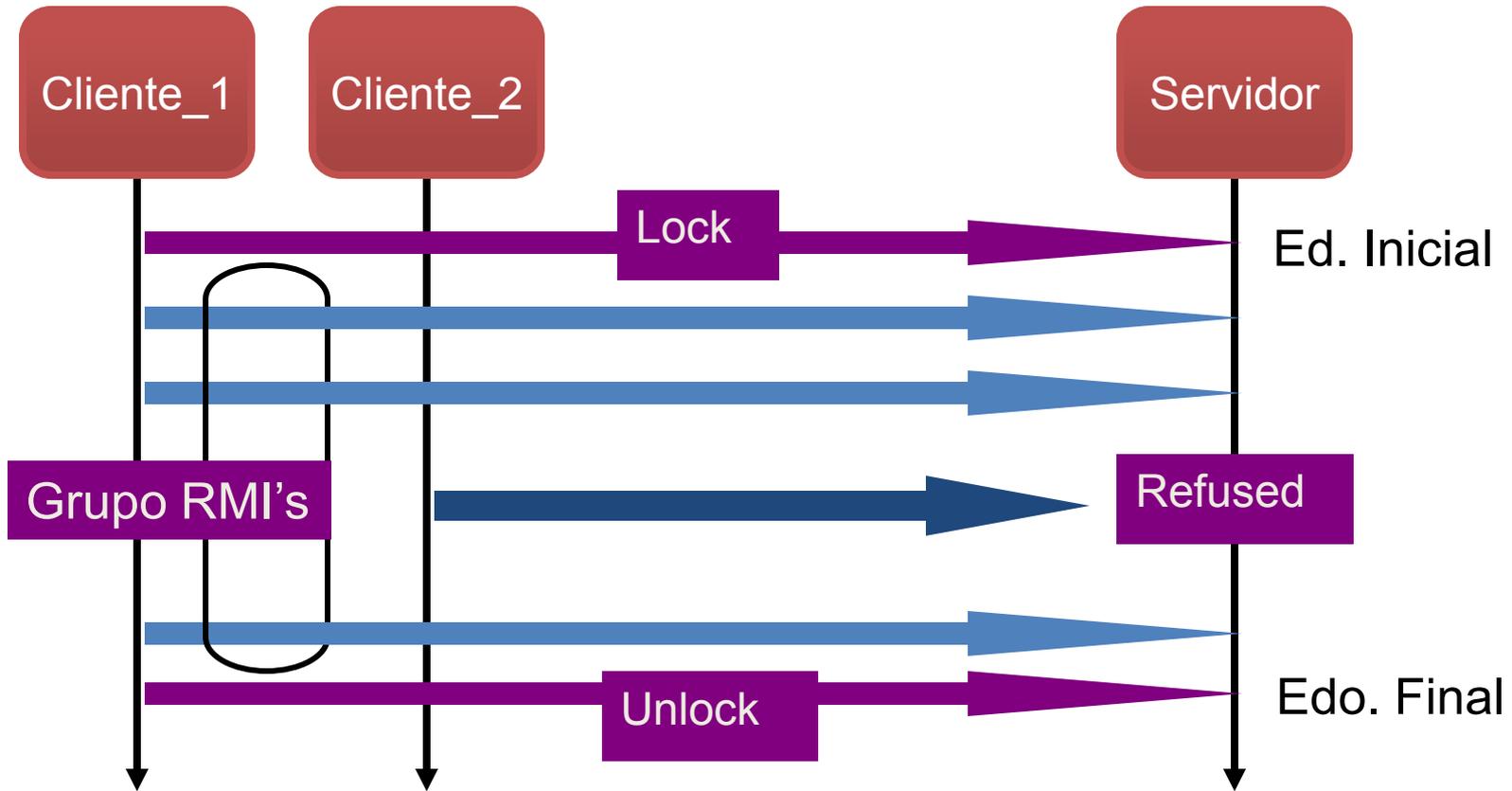
Procesamiento de transacciones



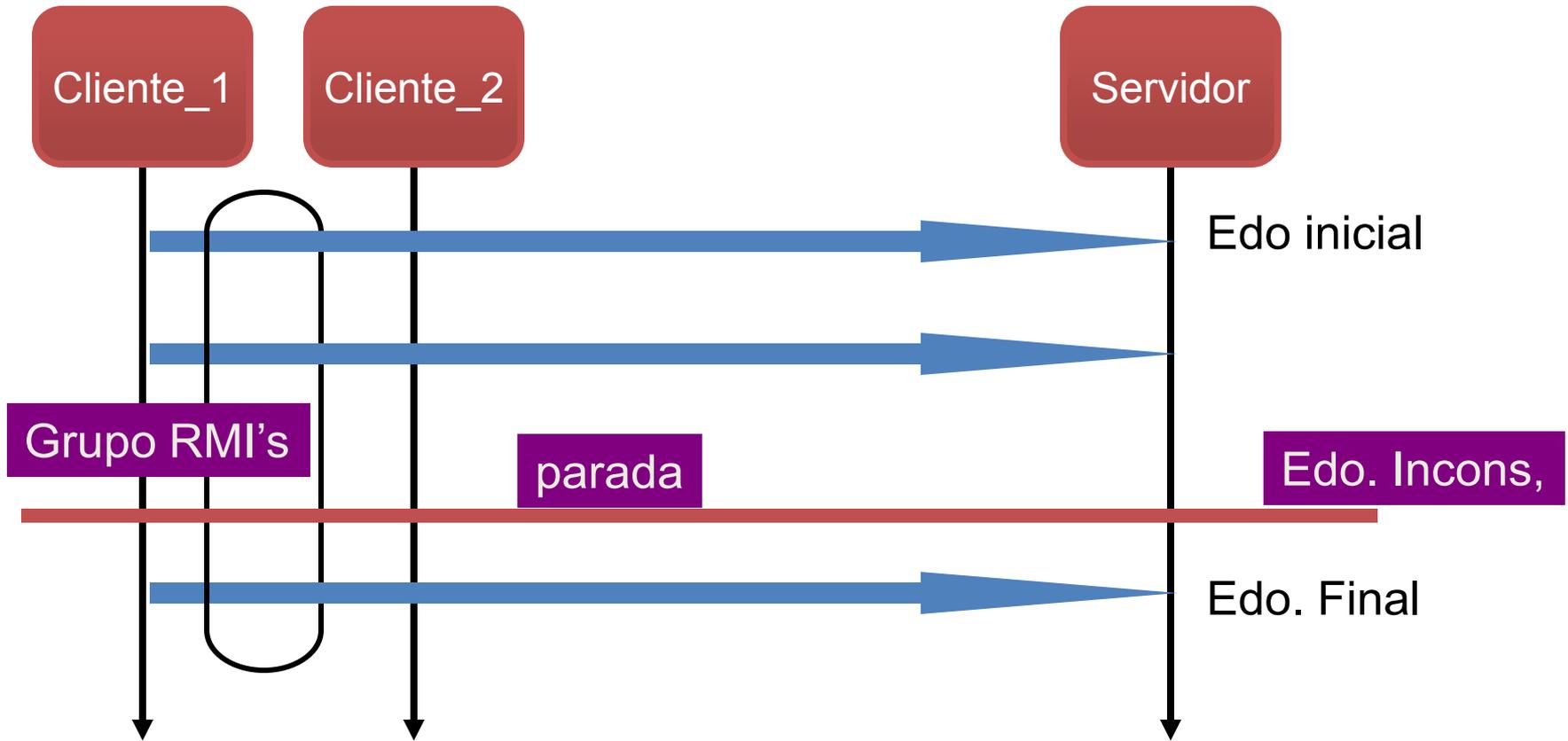
Procesamiento de transacciones



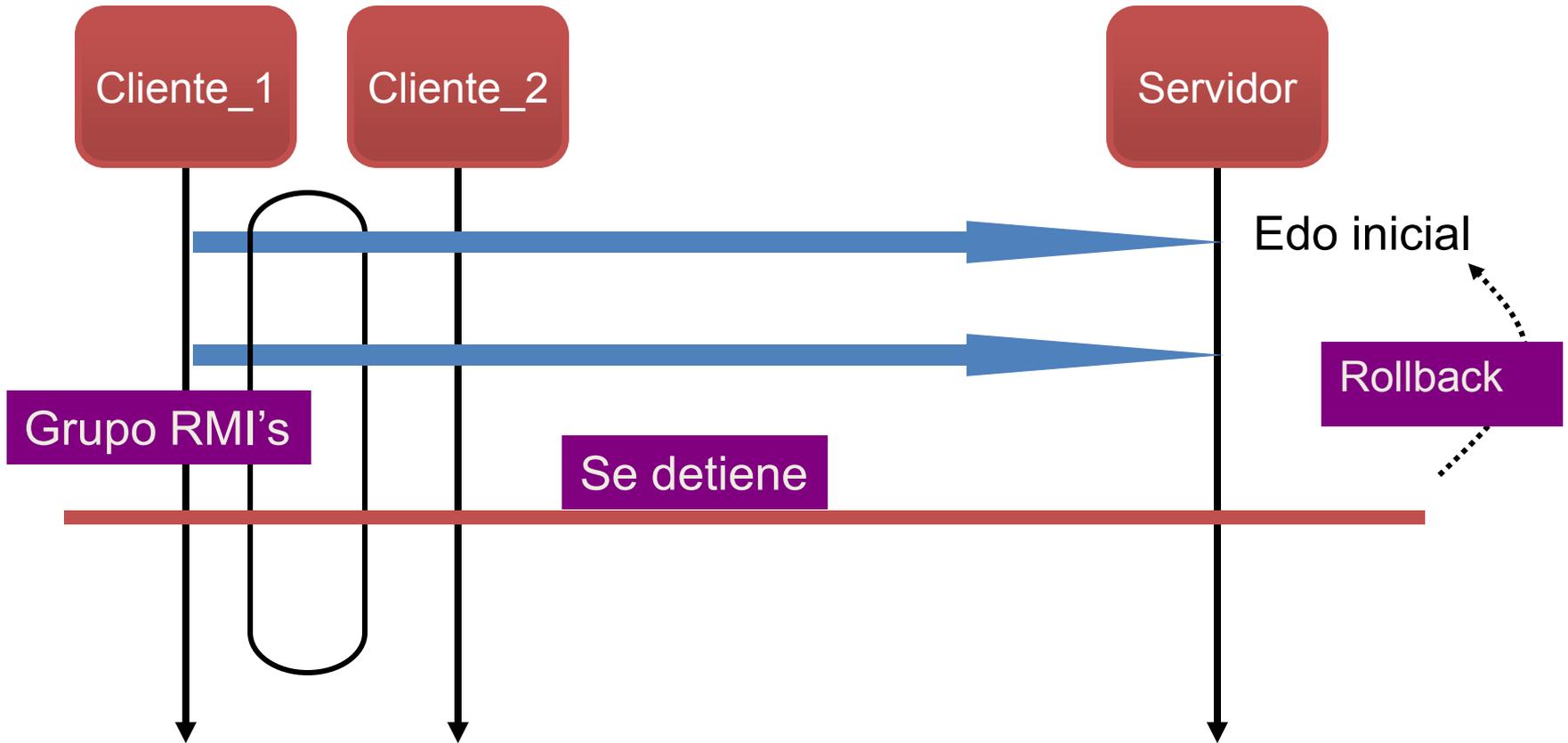
prevenir conflictos



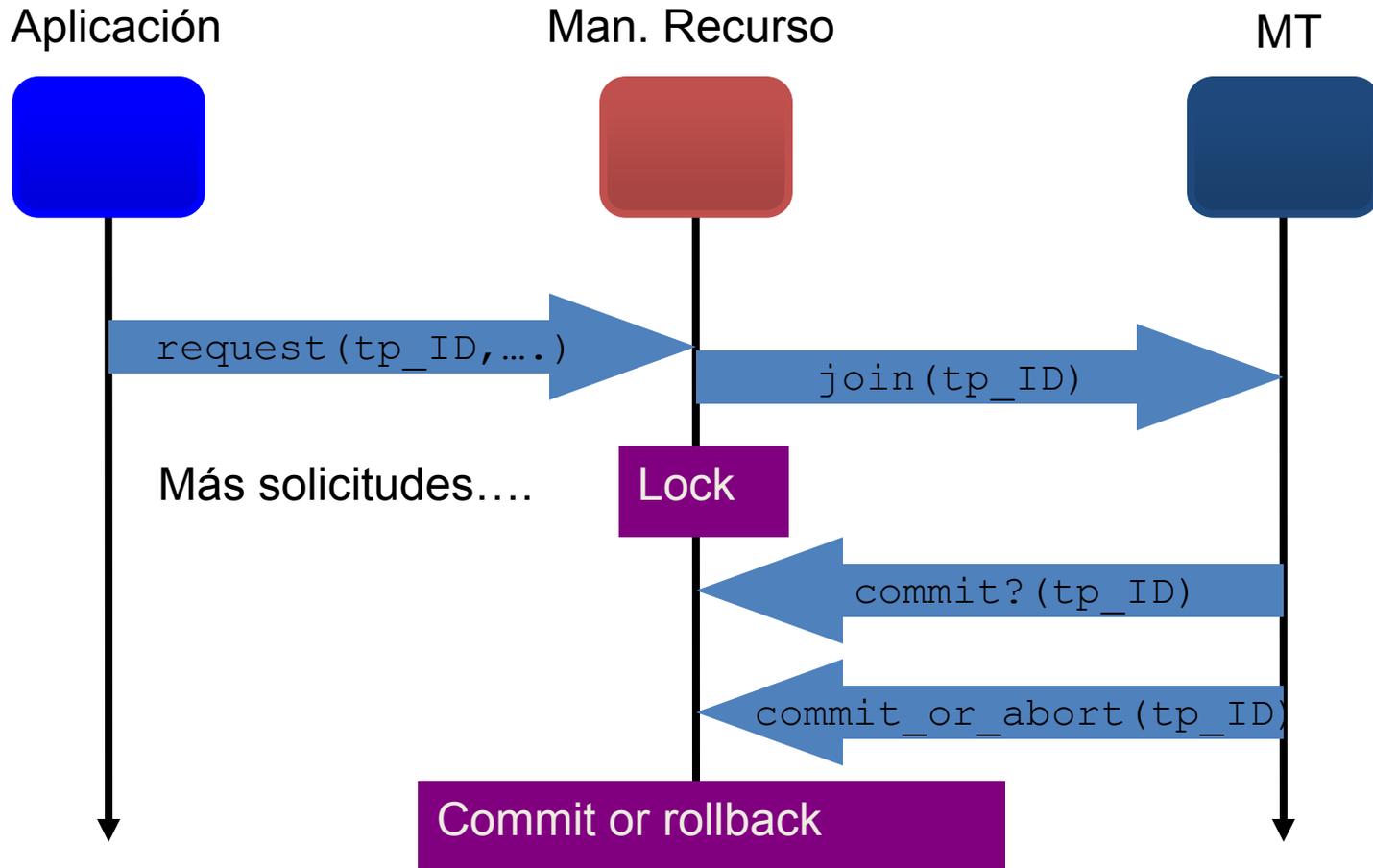
Aborto



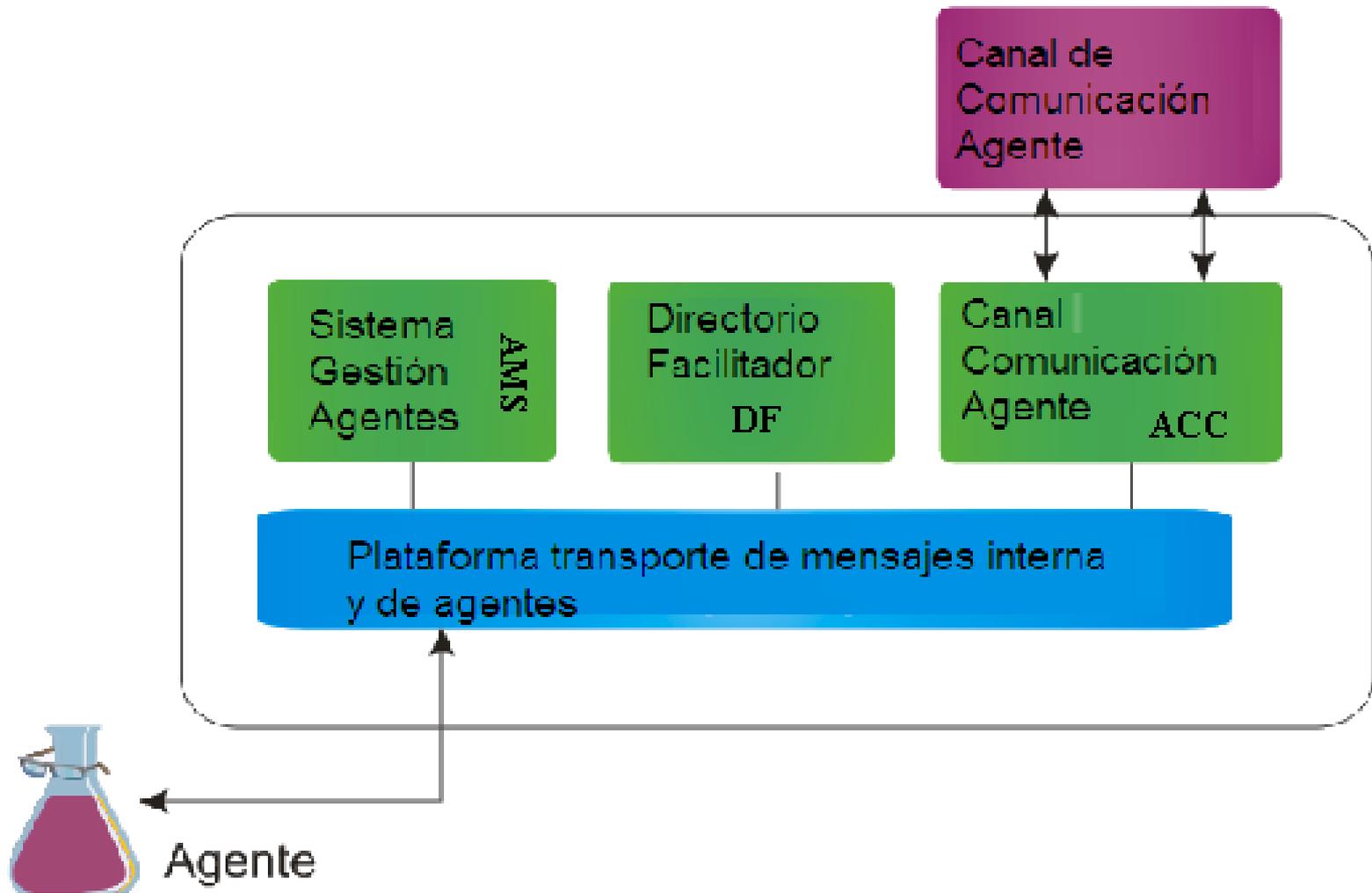
Rollback



Procesamiento de transacciones



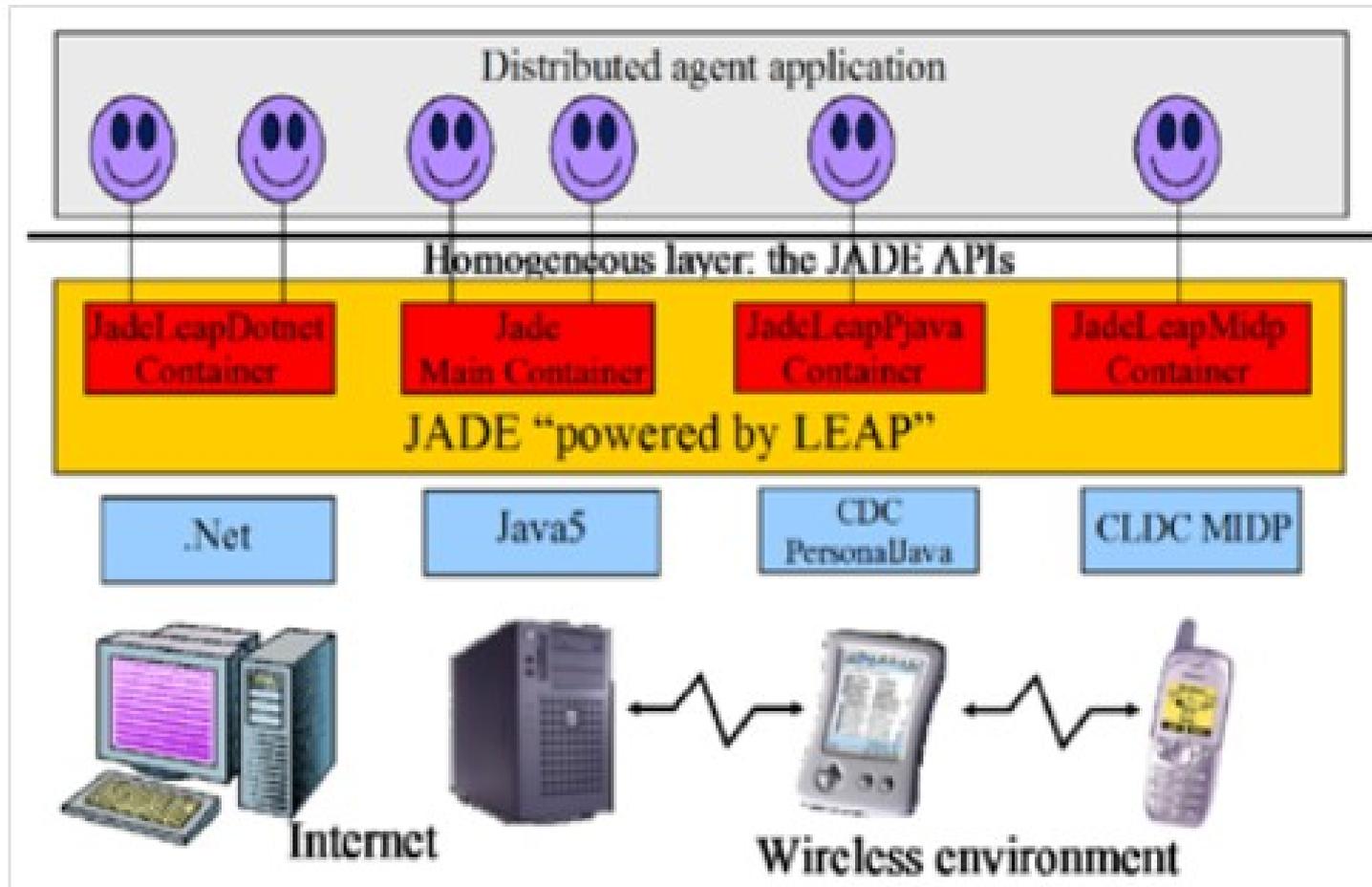
Middleware para Agentes



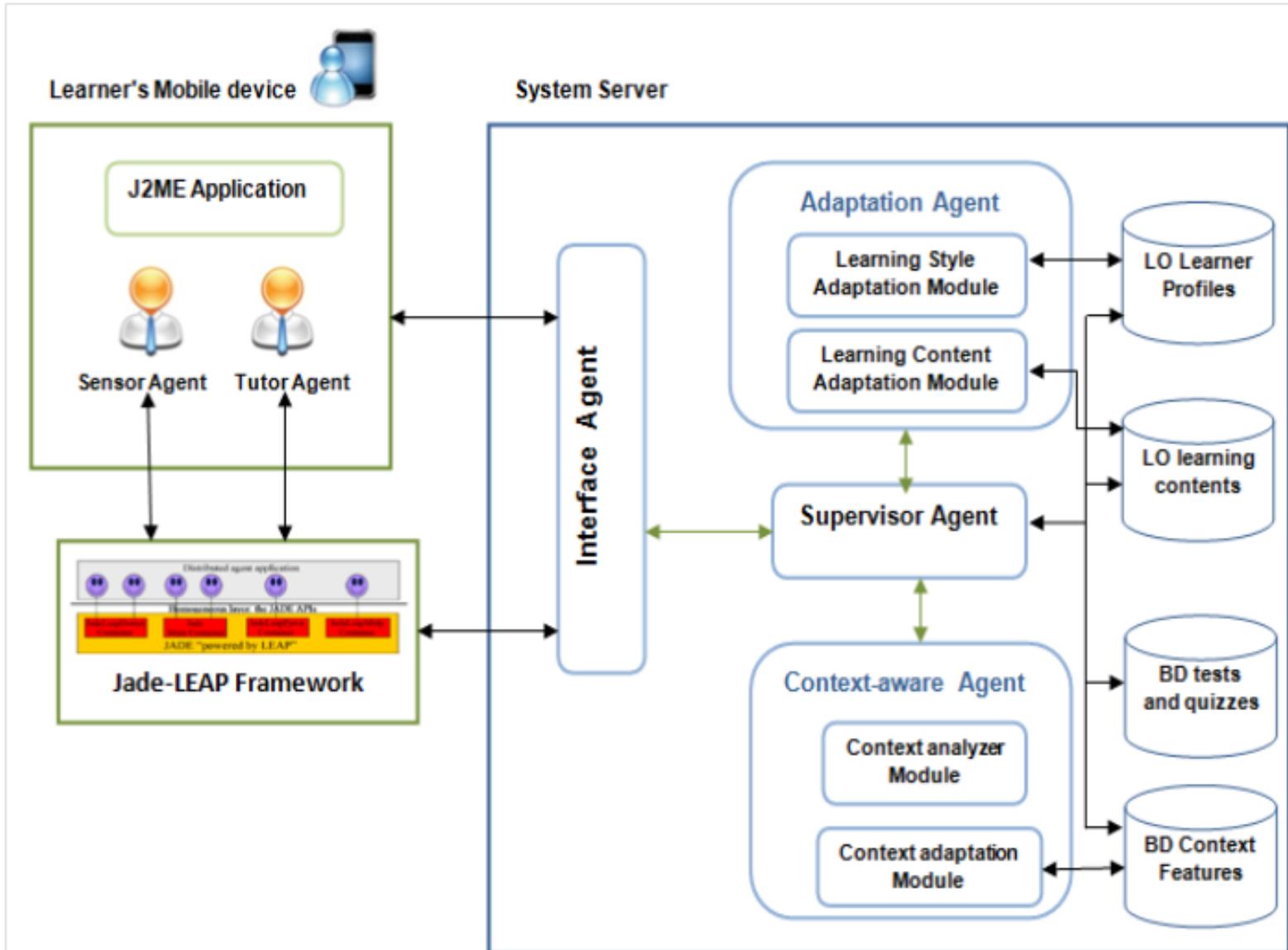
Middleware para Agentes

- *El Sistema de Gestión de agentes (AMS)* es el agente que ejerce el control y supervisión sobre el acceso y uso de la plataforma. Es responsable de mantener un directorio de los agentes residentes y de sus ciclos de vida.
- *El canal de comunicación de los agentes (ACC)* proporciona la ruta de acceso para el contacto entre los agentes dentro y fuera de la plataforma. El ACC es el método de comunicación predeterminado, que ofrece un servicio de enrutamiento confiable y ordenado.
- *El Directorio Facilitador (DF)* es el agente que presta servicios de páginas amarillas a la plataforma de agentes.

Middleware para Agentes



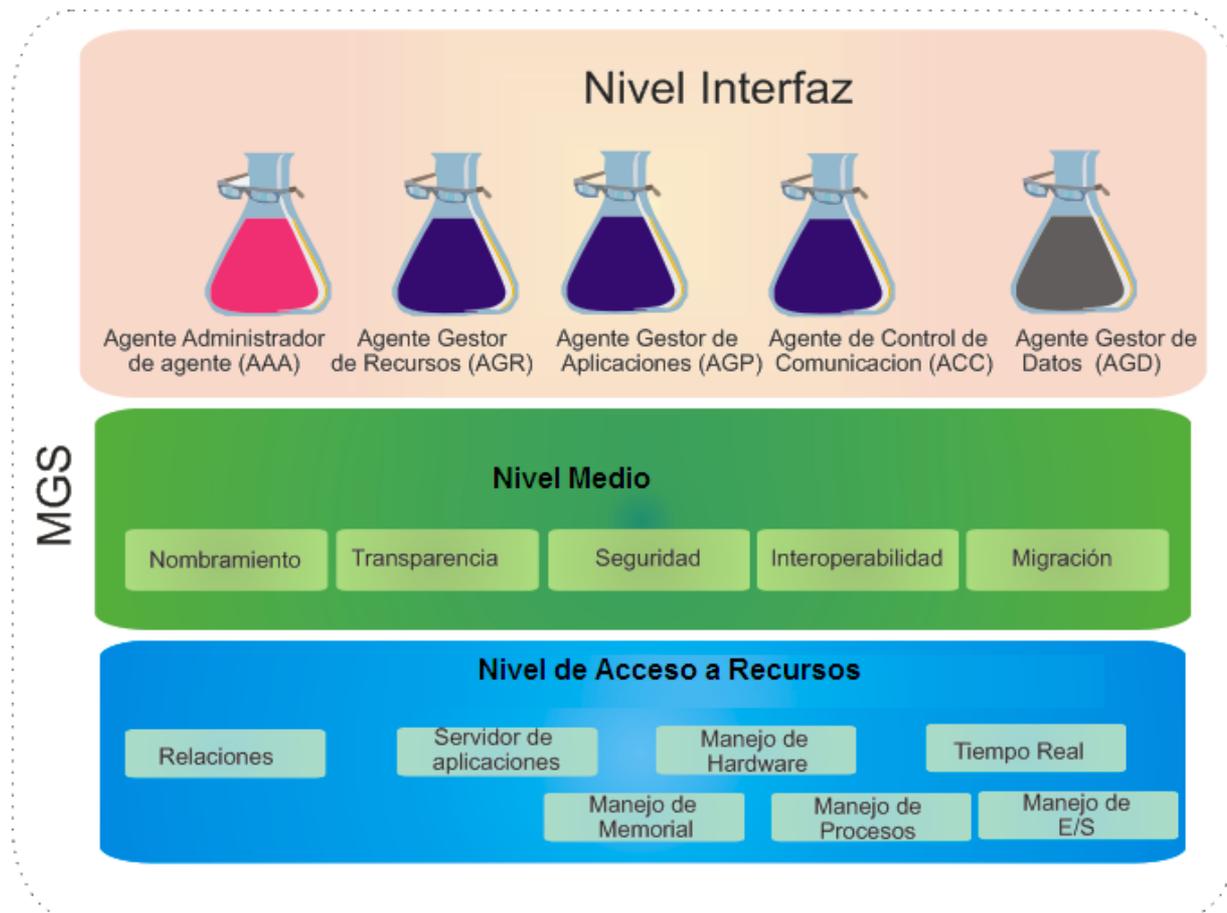
Middleware para un sistema multiagente de aprendizaje móvil



Middleware para agentes



Sistema Multiagentes



Middleware para agentes

MGS debe proveer mecanismos que permitan a los Agentes (comunidad de agentes del SMA):

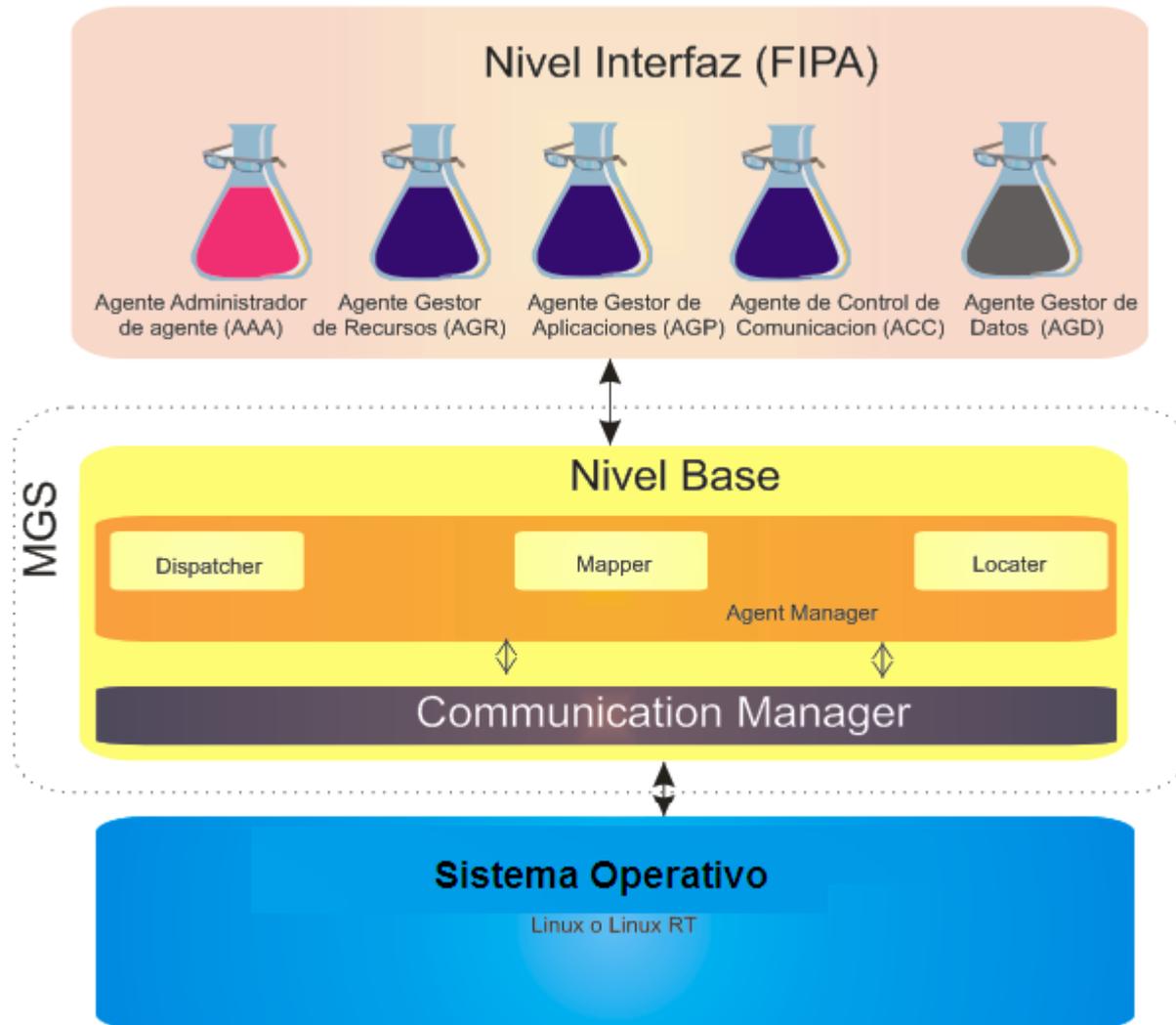
- Conocer los servicios disponibles agrupados por niveles, por tipos de servicio, o alguna otra clasificación
- Conocer los agentes que prestan un determinado servicio y dónde están, cómo se invocan,
- ¿Qué requerimientos se deben cumplir para accederlos?.
- La movilidad, implica el proceso de migración
- Comunicarse entre agentes
- Su activación y desactivación, la posibilidad de invocarlos, etc.

Middleware para agentes

MGS requiere una serie de servicios para asegurar que puede cumplir su cometido.

- Acceso a las tablas de proceso para mantener control de los agentes en ejecución.
- “Copiado” del estado de un agente para manejo de su migración.
- Mecanismo para controlar las comunicaciones.
- Esquemas de manejo de seguridad (por usuario, por servicio, por recurso, etc.).
- Esquemas de virtualización de recursos.

Implantación Middleware para agentes

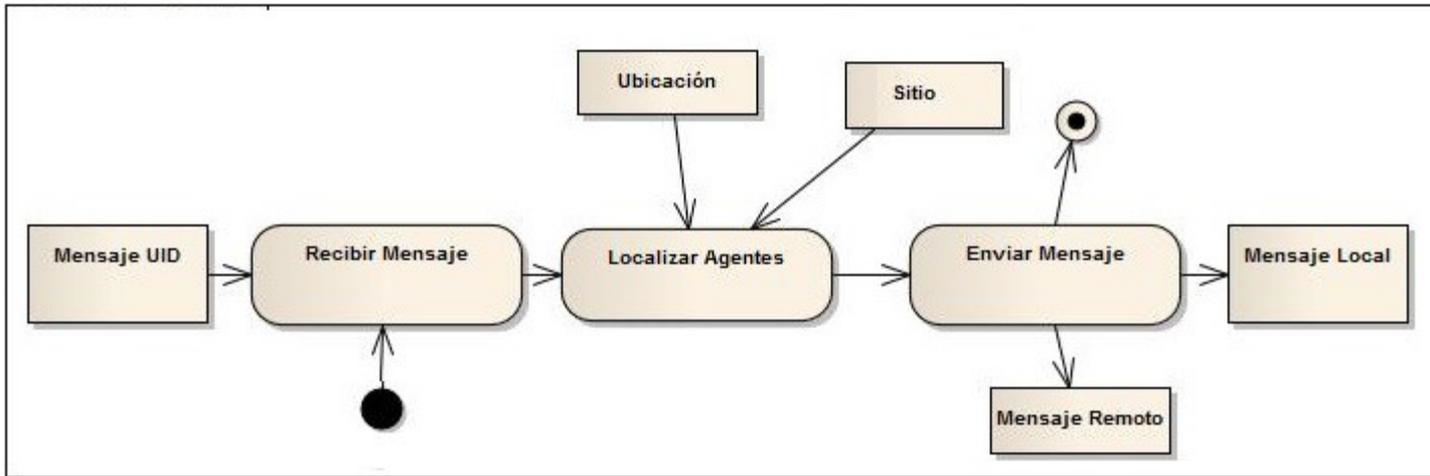


Implantación Middleware para agentes

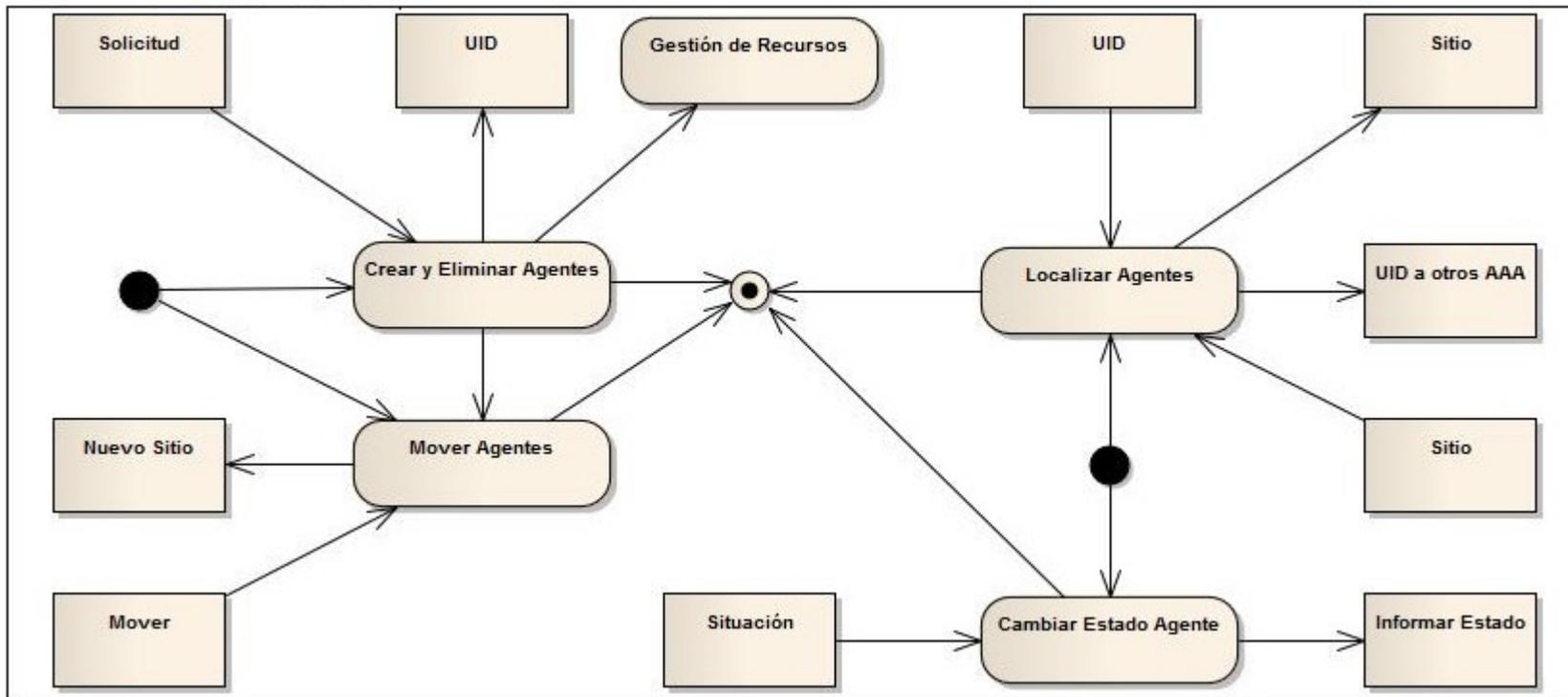
AAA

- ***Despachador (Dispatcher)***: se encarga de despachar invocaciones a los agentes.
- ***Mapper***: se encarga de otorgar identificadores ´ unicos y de gestionar los recursos del sitio para los agentes y procesos.
- ***Localizador (Locator)***: Se encarga de localizar agentes usando sus identificadores ´ unicos.

Implantación Middleware para agentes



ACC



AAA

Middleware Reflexivos

Middleware flexible (OOM) para, por ejemplo, aplicaciones móviles y sensibles al contexto (adaptación al contexto), a través de la **supervisión y sustitución** de componentes

- Interfaces para la **reflexión**
 - Los objetos pueden inspeccionar el comportamiento middleware
- Interfaces para la **personalización**
 - Reconfiguración dinámica en función del entorno

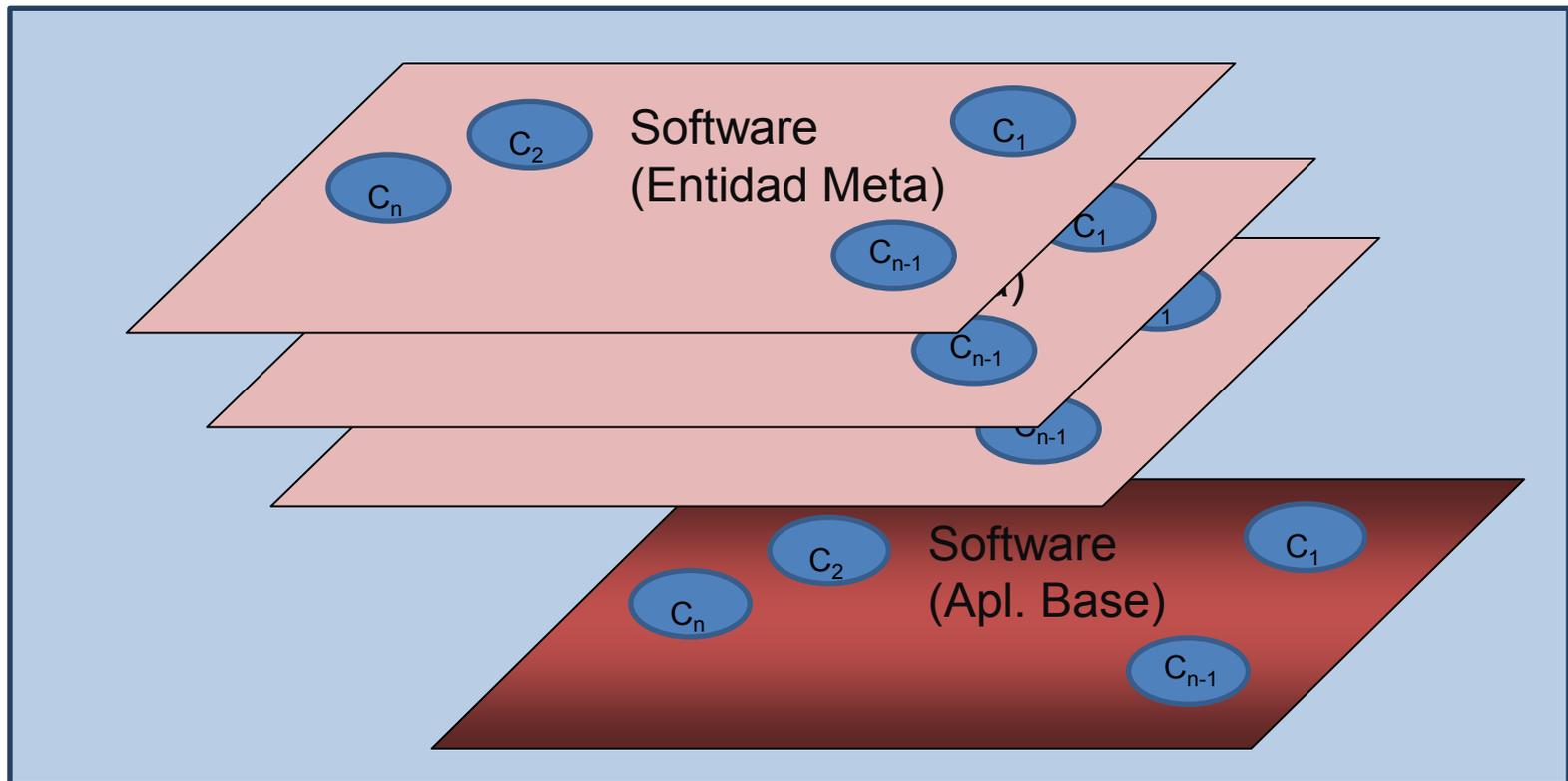
Middleware Reflexivos

Reflexión confieren la capacidad de desarrollar sistemas computacionales sensibles a su ambiente.

- Dos tipos de fases en la reflexión:
 - **Introspección**; La capacidad de un componente para observar y razonar acerca de su propio estado de ejecución.
 - **Intersección**; Es la habilidad de un componente para modificar su propio estado de ejecución (estructura) o alterar su propia interpretación o significado (comportamiento).

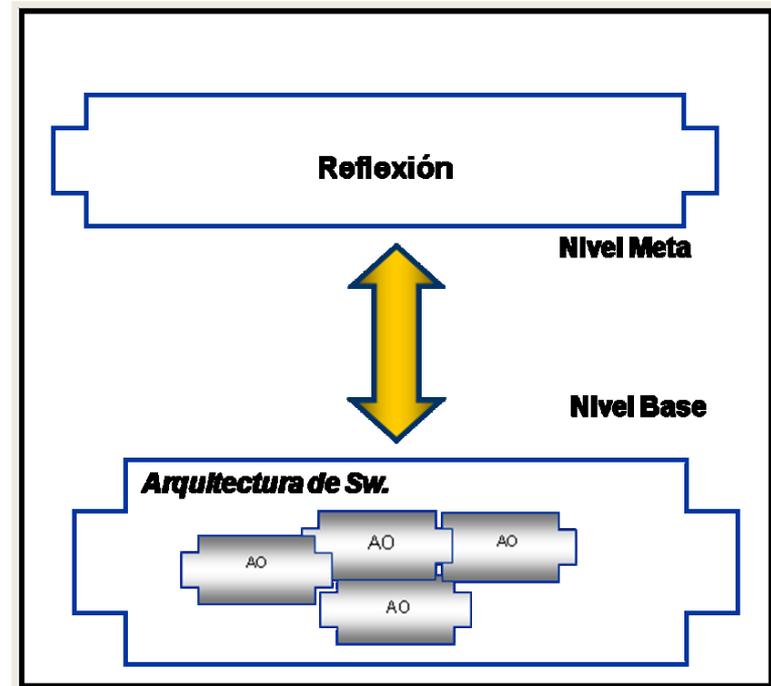
Computación Reflexiva

auto-conciencia y auto-referencia (**Introspección**) de los sistemas o programas, lo cual les permite cambiar (**Intercesión**) su “comportamiento”, según los requerimientos y necesidades del entorno de ejecución

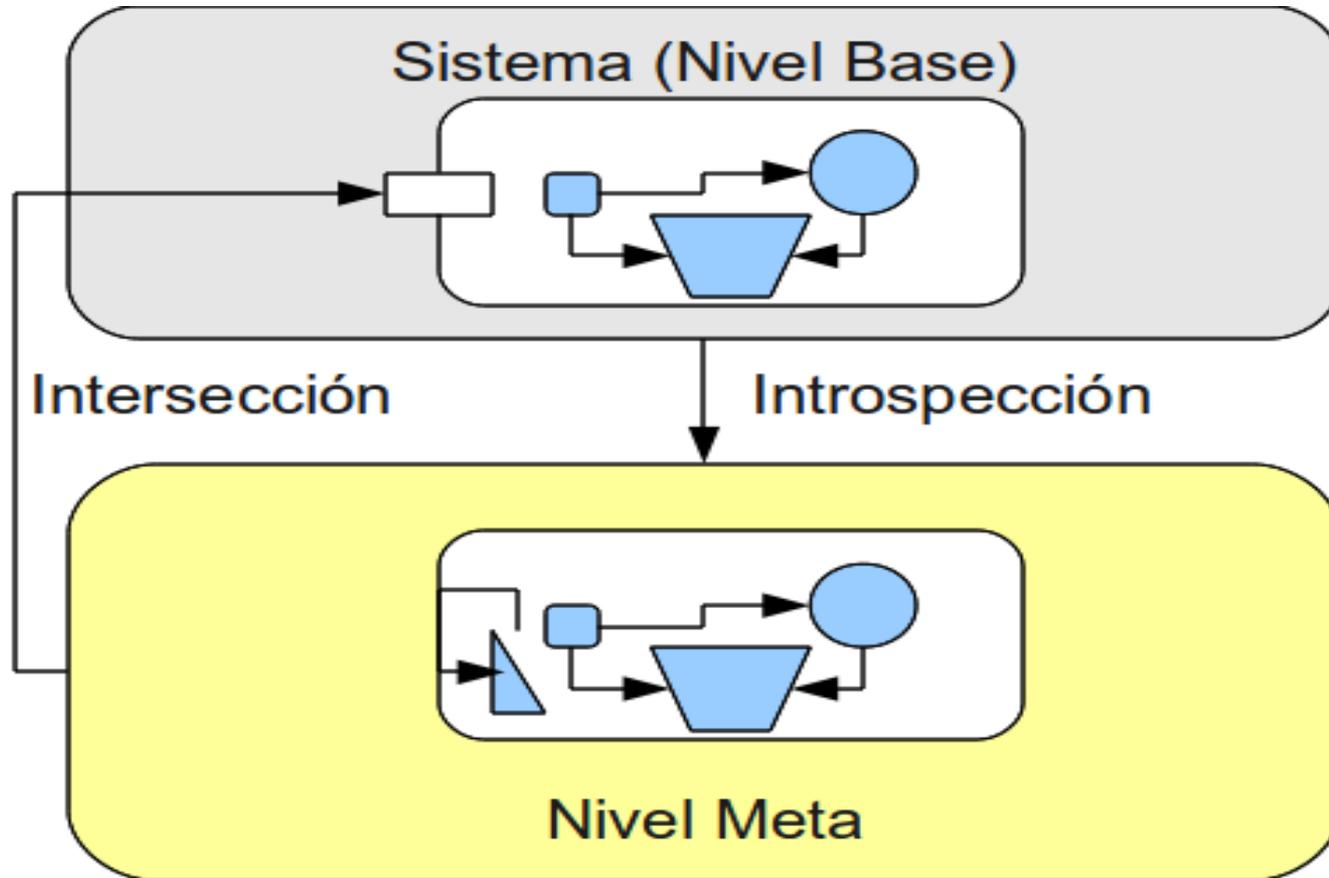


Middleware Reflexivo:

- El middleware reflexivo permite incorporar, entre otras cosas, las siguientes características al nivel base:
 - Auto-conciencia
 - Auto-Eficiencia
 - Planes Preventivos
 - Contextualmente Consciente

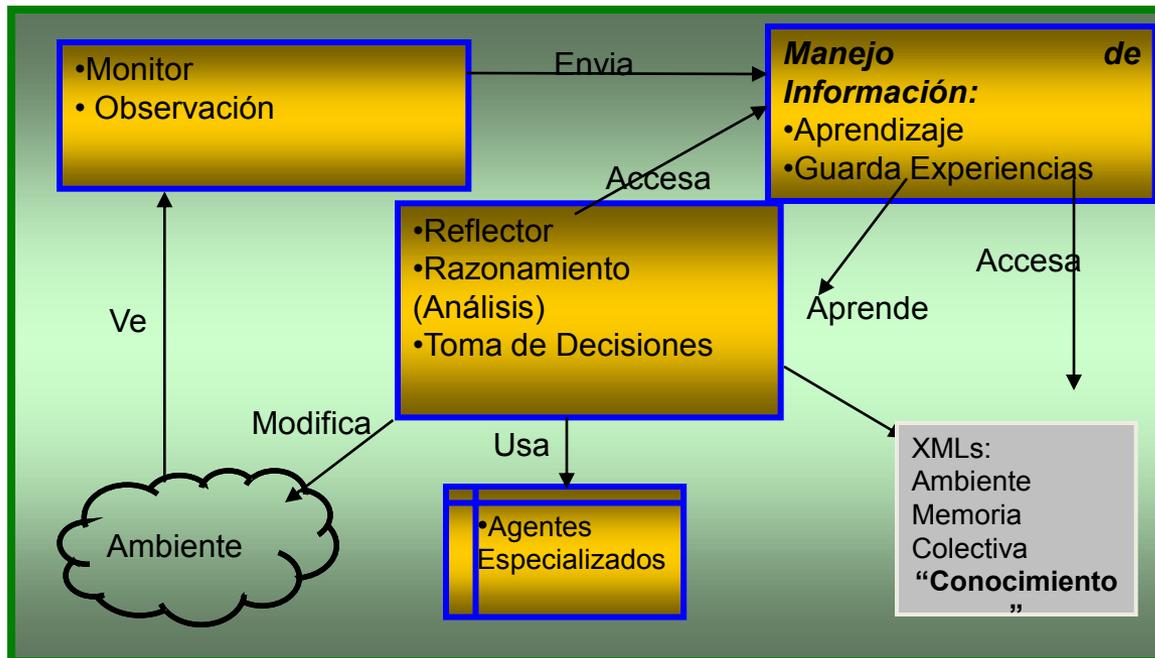


Middleware Reflexivos



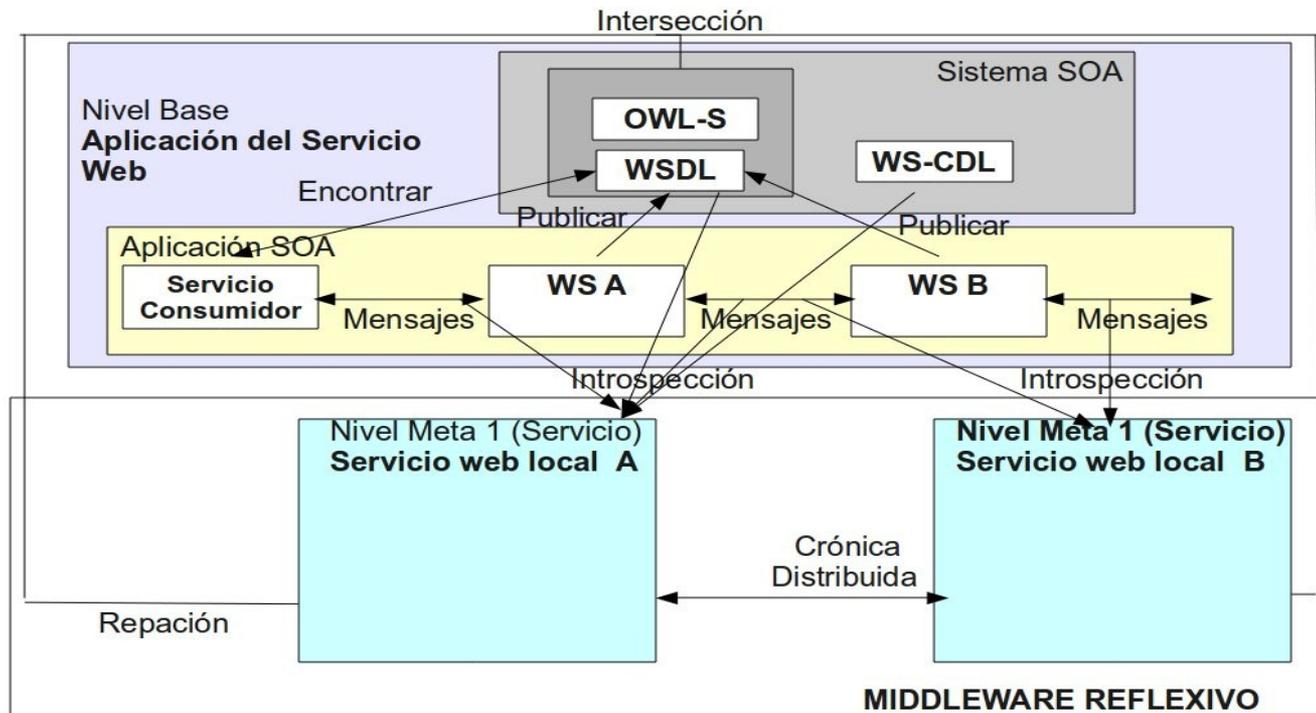
Middleware Reflexivo (MR)

- **Agente Monitor:** Observa el nivel base (introspección).
- **Agente Reflector:** Procesa y analiza la información del nivel base para tomar decisiones. (Intercesión)
- **Agente Manejador de Información:** Escribe y lee los archivos XML que almacenan la información relacionada con los eventos del sistema base



Arquitectura (ARMISCOM)

Diseñar un middleware reflexivo para la gestión de fallas en el proceso de composición de servicios web



Sistemas Conscientes de Contexto

- Las computadoras modernas están divorciados de la realidad
 - No se dan cuenta de quién, dónde, y lo que los rodea
 - Conduce a la falta de conciencia
- Las computadoras con extremadas limitaciones en la entrada
 - Sólo conscientes de entrada explícita
 - Se puede tomar un montón de esfuerzo para hacer cosas simples
- *Informática Context-Aware*
 - Hacer computadores más conscientes de los mundos físico y social en que vivimos

Sistemas Conscientes de Contexto

“Un sistemas es consciente del contexto si este
usa el contexto para proveer información
relevante y/o servicios para el usuario, donde
la relevancia depende de las tareas del
usuario.”

Contexto como E/S



Contexto:

- estado usuario
- estado del ambiente
- estado del sistema computacional
- historia de las interacciones usuario-PC
- ...

Sistemas Conscientes de Contexto

Ejemplos	Tipo contexto	Humano
Auto apagado luz	habitacion	Conveniencia
Sistemas archivos	personal	Buscar info
Recuerdos en calendarios	Tiempo	Memoria
Alarma incendio	habitacion	Seguridad
Escaner id	identificación	Eficiencia

Sistemas Conscientes de Contexto

Ejemplos cercanos	Tipo contexto	Humano
Auto apagado celular reuniones	Identidad	Conveniencia
fotos	Tiempo	Buscar inf
recordar	Localidad	Memoria
Alertas salud	Proximidad	Seguridad
Servicio envio cosas	Actividad	Eficiencia
	Historia	
	...	

Sistemas Conscientes de Contexto

Enfoques para la construcción de aplicaciones sensibles al contexto:

- ***Modelo de contexto fuera de la aplicación.*** Son aplicaciones que realizan todas las funciones de adquisición, pre-procesamiento, almacenaje y razonamiento del contexto fuera de los límites de la aplicación.
- ***Modelo de contexto implícito.*** Son aplicaciones que usan librerías, marcos de trabajos y herramientas para realizar las tareas de adquisición de información del contexto, pre-procesamiento, almacenaje y de razonamiento.
- ***Modelos de contexto explícito:*** Son aplicaciones que usan infraestructuras de gestión del contexto o soluciones de middleware.
 - las acciones como adquisición, pre-procesamiento, almacenaje y razonamiento del contexto se encuentran fuera de la aplicación.
 - la gestión del contexto y la aplicación son claramente separados, y pueden ser desarrollados y extendidos independientemente

Sistemas Conscientes de Contexto

Definiciones importantes

- **Modelo de Contexto.** “identifica un subconjunto concreto del *contexto que es realísticamente accesible por los sensores, aplicaciones y usuarios, y pueden ser usados para la ejecución de las tareas.* El modelo de contexto es usado por una aplicación consciente del contexto dada, y es usualmente explícitamente especificado por el desarrollador”
- **Atributo del contexto:** “*es un elemento de un modelo de contexto que describe el contexto. Un atributo de contexto tiene un identificador, un tipo y un valor, y opcionalmente, un colección de propiedades describiendo específicamente sus características*”

Sistemas Conscientes de Contexto

Definiciones importantes

- **Calidad de Contexto (QoC):** conjunto de parámetros que expresan la calidad de los datos del contexto.

QoC se basa en tres parámetros:

- la validez de los datos del contexto,
- la precisión de los datos del contexto y
- que tan actualizados están los datos de contexto.

Se dice que la QoC depende de la calidad de los sensores físicos, la calidad del proceso de envío, entre otras cosas.

Sistemas Conscientes de Contexto

Niveles de conciencia

- **Personalización:** Esto permite al usuario ajustar sus preferencias, gustos, y expectativas del sistema manualmente. Por Ejemplo, un usuario puede ajustar la temperatura deseada en un ambiente.
- **Consciencia de Contexto Pasiva.** El sistema constantemente monitorea el ambiente y ofrece las opciones apropiadas para que un usuario pueda tomar acciones. Por ejemplo, cuando un usuario entra a un supermercado, el dispositivo móvil alerta acerca de la lista de descuento.
- **Consciencia de contexto Activa:** El sistema continuamente y automáticamente monitorea la situación y actúa autónomamente. Por ejemplo, si el detector de humo y de temperatura detecta fuego en la habitación en un sistema inteligente, este inmediatamente alerta al cuerpo de bomberos y al propietario de la casa a través de métodos apropiados (por ejemplo, una llamada telefónica).

Ejemplos de Contexto

- Identidad
- Espacial: localidad, orientación, velocidad
- Temporal: fecha, tiempo, época del año
- Ambiental: temperatura, luz
- Social: gente cerca, actividad, calendario
- Recursos: cerca, disponible
- Sicológicos o fisiológicos: presión de sangre, tasa de latido del corazón, tono de voz

Sistemas Conscientes de Contexto

Deben soportar:

- ***La Presentación***. El contexto puede ser usado para decidir que información y servicios necesitan ser presentados al usuario.
- ***La Ejecución***. La ejecución automática de servicios es una característica crítica en los sistemas conscientes de contexto.
- ***El Etiquetado***. Existen un gran número de sensores en los ambientes inteligentes. Estos sensores pueden producir un gran volumen de datos que deben ser recolectados, analizados, fusionados e interpretados. El contexto necesita ser etiquetado para ser procesados y entendidos luego.

Sistemas Conscientes de Contexto

Los tipos de contextos:

- *Categorización Operacional*: Categoriza el contexto basado en cómo se adquiere, modela y trata
- *Categorización Conceptual*: Categoriza el contexto basado en el significado y las relaciones conceptuales entre los contextos

Sistemas Conscientes de Contexto

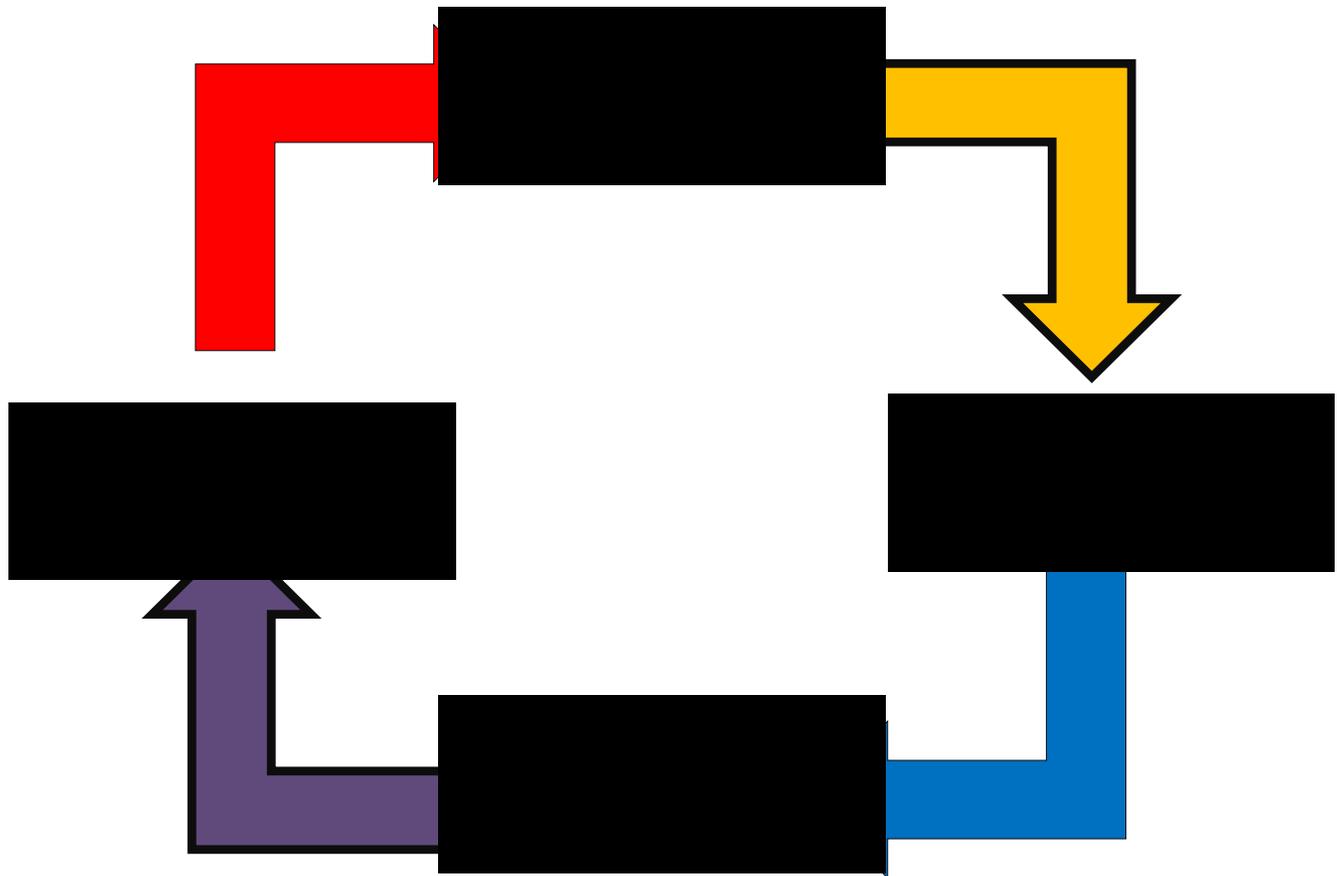
Los tipos de contextos:

- **Contexto Primario:** Cualquier información que se recupera sin necesidad de utilizar el contexto existente, y sin realizar ningún tipo de operación de fusión de datos de sensores (por ejemplo, lecturas de los sensores).
- **Contexto Secundario:** Cualquier información que pueda ser calculada usando el contexto primario.
 - El contexto secundario se puede calcular mediante el uso de las operaciones de fusión de datos desde los sensores, o a partir de operaciones sobre el comportamiento del sistema (por ejemplo, identificar la distancia entre dos sensores).
 - Ella recupera contexto como números de teléfono, direcciones de correo electrónico, etc.

		Categorías de Contexto (Perspectiva Operacional)	
		Primaria	Secundaria
Localizacion		Localizacion de Datos desde un sensor GPS (Por Ejemplo Longitud y Latitud)	Distancia do dos sensores computados usando los valores de GPS. Imagen de un Mapa recuperado de un proveedor de servicios de mapas
Identidad		Identidad del usuario basada en etiquetas RFID	Recuperar una lista de amigos desde los usuarios de un perfil de facebook. Identificar la cara de una persona haciendo uso de un sistema de reconocimiento facial
Tiempo		Leer el la hora desde un reloj	Calcular la estacion de basada en la informacion del tiempo. Predecir el tiempo basado en la actividad actual y el calendario
Actividad		Identificar la actividad de la apertura de la puerta desde un sensor de la puerta	Predecir la actividad de un usuario basado en el calendario del usuario Buscar la actividad de un usuario basado en los sensores del telefono movil tales como GPS, Giroscopio y acelerometro

Sistemas Conscientes de Contexto

Ciclo de vida del Contexto



Ciclo de vida del Contexto

Adquisición del contexto

- **La Responsabilidad.** La adquisición del contexto (por ejemplo, los datos del sensor) puede llevarse a cabo principalmente mediante dos métodos: Se solicita, Se informa.
- **La Frecuencia.** El contexto puede ser generado a partir de dos tipos de eventos: eventos instantáneos y eventos en un intervalo de tiempo. Instantáneos, Intervalo de tiempo
- **La fuente.** tres categorías sobre la base de la procedencia del contexto: Adquisición directa desde el sensor, Adquisición a través de una infraestructura de middleware, Adquisición desde servidores de contexto
- **Los Sensores.** Los sensores pueden ser divididos en tres categorías, Sensores físicos, Sensores virtuales, Los sensores lógicos
- **El proceso de adquisición.** Hay tres formas de adquirir el contexto: Por Sentido, Por derivación, Manual.

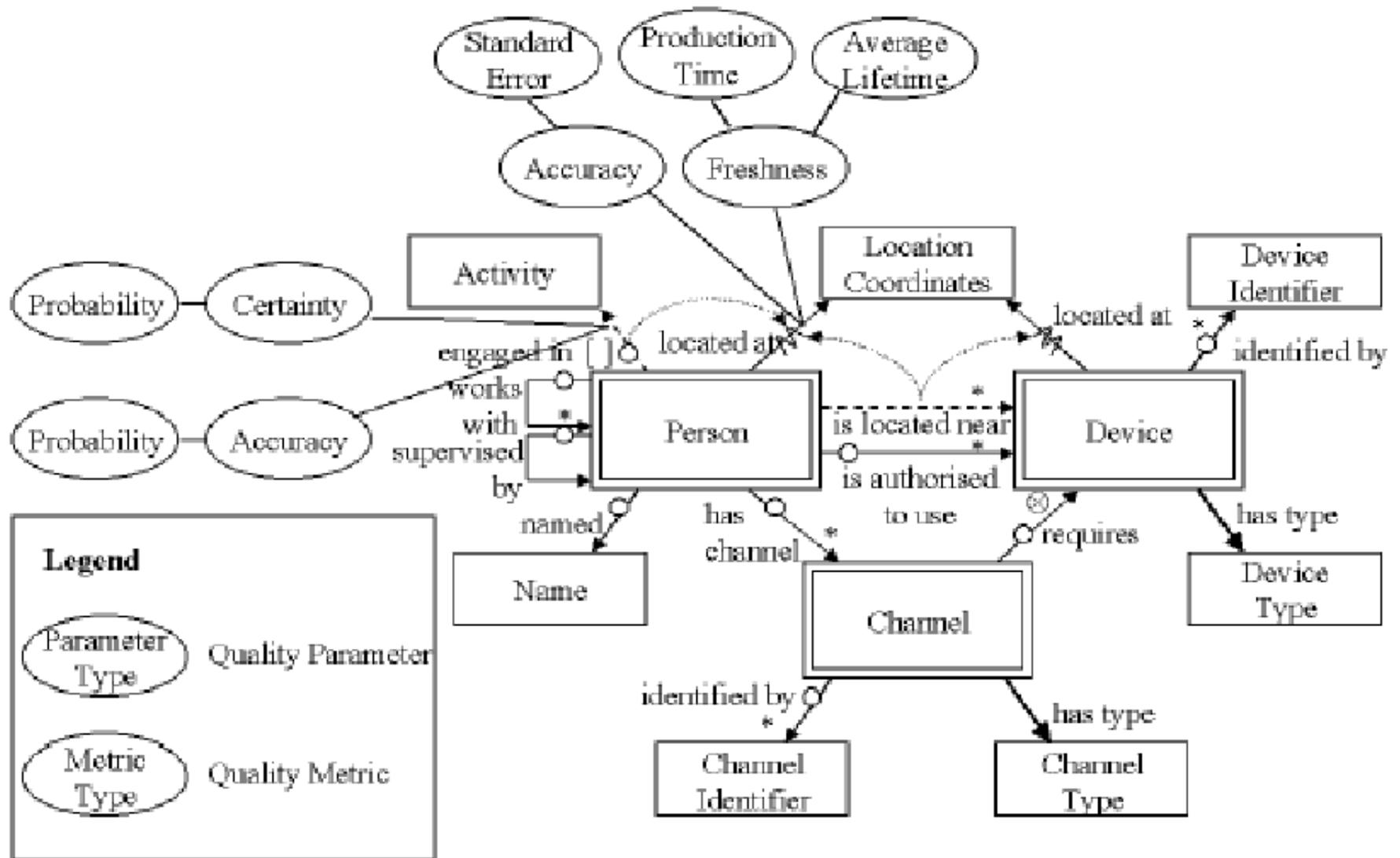
Ciclo de vida del Contexto

Modelado del Contexto

también es referido como la representación del contexto. Pueden ser estáticos o dinámicos.

- Por lo general, hay dos **pasos en la representación de contexto**:
 - **Modelado del Contexto**: el contexto se define en términos de atributos, características, relaciones en el contexto.
 - **Organización del contexto** : el resultado de la etapa de modelado de contexto necesita ser validado. Además, la nueva información del contexto necesita ser fusionada y añadida en el repositorio de información del contexto existente. Por último, la nueva información del contexto se pone a disposición para su uso cuando sea necesario.
- la información de contexto tiene seis **estados**: listo, en marcha, suspendido, en reanudación, ha caducado, y terminado.
- **Técnicas más populares de modelado**: *clave-valor, marcado, grafo, basado en objetos, basado en lógica, y basado en ontologías.*

Ejemplo de modelo



Ciclo de vida del Contexto

Razonamiento del Contexto

Método para deducir nuevos conocimientos basada en la información del contexto disponible.

‘El razonamiento surge, entre otras cosas, por dos características del contexto: **es imperfecto** (es decir, es desconocido, ambiguo, impreciso o erróneo) y **es incierto**.

- El proceso de razonamiento consta de varios pasos:
 - **Tratamiento previo**: En esta fase se limpian los datos de los sensores.
 - **Fusión de datos**: Se trata de combinar datos de sensores para producir datos más precisos, más completos, e información más fiable.
 - **Inferencia del Contexto**: Es el proceso en sí de razonamiento, se deducen nuevas informaciones, conocimiento, etc.
- **Las categorías más empleadas de razonamiento de contexto son**: razonamiento basado en reglas, razonamiento difuso, razonamiento ontológico y razonamiento probabilístico.

Ciclo de vida del Contexto

Diseminación del Contexto

- Los dos métodos utilizados para la distribución del contexto son:
 - **Query (consulta)**. El consumidor de contexto hace una petición en términos de una consulta,.
 - **Suscripción (también llamado de publicación/suscripción)**: El consumidor de contexto se puede suscribir a un sistema de gestión de contexto. El sistema le devolverá resultados periódicamente, o cuando se produce un evento (umbral de violación). En particular, los consumidores pueden suscribirse a un sensor específico, a un evento.

Sistemas Conscientes de Contexto

La ***tercera generación de sistemas conscientes de contexto*** se caracterizan por adoptar en su mayoría:

- la **ontología** como modelos de contexto,
- se enfocan en la **protección de la privacidad**,
- y como reto **la escalabilidad**

Cyberguide



Cyberguide

Georgia Tech
Abowd et al

- GPS
 - Precisa localidad
- Coloca en screen donde se encuentra
- Predefine puntos de interés
- Asocia a guía de viajero
 - Fotos, etc.
 - Caminos a seguir
- Contexto: sitio, tiempo

Cyberguide

