TECNOLOGÍAS MIDDLEWARE PARA EL DESARROLLO DE SERVICIOS EN ENTORNOS DE COMPUTACIÓN UBICUA

Autora: Mª Celeste Campo Vázquez
Director: Andrés Marín López
7 de Mayo de 2004

Índice

- Introducción
- Estado del arte
- PDP: *Pervasive Discovery Protocol*
- SDA: *Service Discovery Agent*
- Conclusiones y trabajos futuros

Índice

- Introducción
  - Motivación
  - Objetivos
- Estado del arte
- PDP: *Pervasive Discovery Protocol*
- SDA: *Service Discovery Agent*
- Conclusiones y trabajos futuros

3

Introducción

- [Weiser, 1991]

  "The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it"

- Evolución computación distribuida:
  - Movilidad
  - Ubicuidad

4

Motivación

- Nuevos actores:
  - Dispositivos personales
  - Dispositivos de función específica
  - Sensores y actuadores
- Nuevos escenarios:
  - Protocolos de comunicación inalámbricos
  - Dinámicos y cambiantes
- Nuevos retos:
  - Limitaciones de los dispositivos ? cooperación
  - Coste de las comunicaciones ? autonomía
  - Importancia de las distancias ? ámbito local

Objetivos

- Título:
  - “Tecnologías middleware...
  - para el desarrollo de servicios...
  - en entornos de computación ubicua”
- Tecnologías:
  - Paradigma de agentes móviles para el desarrollo de servicios
  - Protocolos de descubrimiento de servicios como base para la cooperación entre dispositivos

Objetivos

- Estudio del estado del arte
- Paradigma de agentes para entornos ubicuos:
  - Adaptación arquitectura FIPA
  - Implementación
- Protocolo de descubrimiento y anuncio de servicios:
  - Diseño
  - Estudio de prestaciones
  - Implementación
- Integración de las propuestas

Índice

- Introducción
- Estado del arte
  - Tecnologías relacionadas
  - Descubrimiento de servicios
  - Agentes
- PDP: *Pervasive Discovery Protocol*
- SDA: *Service Discovery Agent*
- Conclusiones y trabajos futuros

Tecnologías relacionadas

- Protocolos de comunicación inalámbricos:
  - Formación de redes de forma espontánea: redes ad-hoc
  - Restricciones: calidad cambiante, conectividad no transitiva e intermitente, coste de las comunicaciones,...
- Dispositivos:
  - Heterogeneidad
  - Java 2 Micro Edition
- Proyectos de investigación

Descubrimiento de servicios

- Facilitar a los dispositivos el descubrimiento de los servicios de la red sin necesidad de configuraciones
- Definidos en el IETF:
  - SRVLOC:
    - SLP (1997), SLP v2 (1999)
    - SSDP (1999)
  - Propuestas del ZeroConf:
    - DNS-SD
    - Multicast DNS / LLMNR

Descubrimiento de servicios

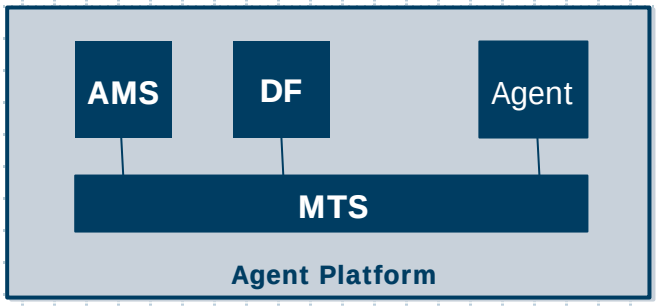
- Integrados dentro de una tecnología de red:
  - SDP de Bluetooth
  - IAS de IrDA
- Independientes de la tecnología de red:
  - Salutation
- Asociados a un *framework* de desarrollo de servicios:
  - Jini
  - OSGi
  - JXTA

Agentes

- Definición de agentes (propiedades):
  - Autonomía, comunicación, cooperación, coordinación, movilidad e inteligencia
- Beneficios de la tecnología de agentes móviles:
  - Independencia de la disponibilidad de la red
  - Reducción del tráfico de la red
  - Distribución de tareas
  - Flexibilidad de las aplicaciones
- No sustituye a otros paradigmas (cliente/servidor):
  - Mejores prestaciones en ciertas aplicaciones:
    - Acceso a BD [Papstravou et al. 1999],
    - Gestión de red [Clitho and Magedanz, 2002]
    - Recolección de datos en redes de sensores [Qi et al., 2001]
  - Movilidad de código paradigma válido en computación ubicua:
    - Proyecto one.world [Grimm et al., 2002] (Proyecto Portolano - U. Washington)

Agentes

- Modelo de referencia FIPA



AMS Agent Management System
DF Directory Facilitator
MTS Message Transport System

13

Agentes

- Plataformas de agentes en computación móvil:
  - LEAP (Lightweight Extensible Agent Platform) - IST-1999-10211
    - Aplicación en redes de telefonía móvil
    - Arquitectura modular: en el terminal móvil no existe una plataforma compatible con FIPA
    - No soporta movilidad en dispositivos limitados
  - MAE de U. de Monash [Mihailescu and Kendall, 2002]
    - Aplicaciones de comercio electrónico en redes de telefonía móvil
    - Plataforma soporta movilidad a nivel agentes
    - No es compatible con FIPA
- Plataformas de agentes en computación ubicua:
  - FIPA Ad-Hoc (2002)

14

Conclusiones

- Computación ubicua:
  - Plantea nuevos retos no abordados hasta ahora
- Descubrimiento de servicios:
  - Protocolos existentes no han sido diseñado para entornos ubicuos
  - Existen varias propuestas para abordar este problema
  - Contribución: Diseño de un nuevo protocolo de descubrimiento teniendo en cuenta los nuevos retos
- Agentes:
  - Paradigma de la computación adaptado a entornos ubicuos
  - Viabilidad de implantación en dispositivos limitados
  - Necesidad de adaptación del estándar FIPA
  - Contribución: Adaptación de la arquitectura FIPA a entornos ubicuos

Índice

- Introducción
- Estado del arte
- PDP: *Pervasive Discovery Protocol*
  - Estudio del problema
  - Diseño razonado
  - Descripción del protocolo
  - Análisis de prestaciones
- SDA: *Service Discovery Agent*
- Conclusiones y trabajos futuros

Estudio del problema

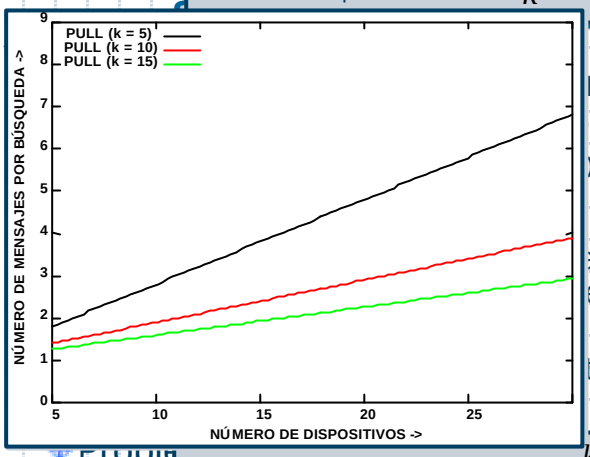
- Estudio teórico:
  - Funcionamiento distribuido:
    - Pull
    - Push
  - Funcionamiento centralizado
- Estudio de las propuestas del IETF:
  - SLP
  - SSDP
  - DNS-SD

Estudio teórico
Solución I - Distribuida modo *PULL*

- Los dispositivos que buscan servicios preguntan al entorno por difusión y los servidores que ofrecen el servicio responden por unicast
- Ventajas:
  - Cuando se necesita un servicio se pregunta al entorno, entonces se tiene una versión actualizada de los servicios disponibles
  - Implementación más sencilla tanto en clientes como en servidores
- Problema:
  - Número de mensajes por búsqueda

Estudio

ESTUDIO ANALÍTICO:
- Número de dispositivos en la red = n
- Número de tipos de servicios = k



rvicios preguntan
rvidores que
or unicast

pregunta al
ersión actualizada

to en clientes como

- Número de mensajes por búsqueda = $\frac{k ? n ? 1}{k}$
- Tasa de servicios descubiertos = 100%
- Tasa de servicios falsos descubiertos = 0%

18

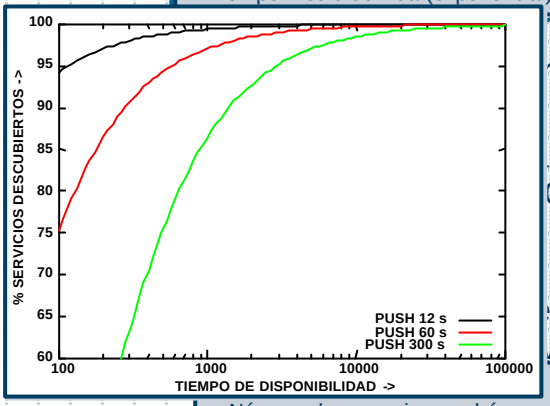
Estudio teórico
Solución II - Distribuida modo *PUSH*

- Los servidores anuncian periódicamente los servicios que ofrecen
- Los clientes almacenan localmente estos anuncios para consultarlos cuando quieren descubrir un servicio
- Problemas:
  - Mayor complejidad en clientes y servidores
  - Lento, dependiendo del intervalo entre anuncios
  - Si los anuncios son muy frecuentes, se consume mucho ancho de banda y si son poco frecuentes, se puede intentar acceder a servicios que no están disponibles

19

Estudi

ESTUDIO ANALÍTICO:
- Tiempo entre anuncios = T_a
- Tiempo medio entre peticiones (exponencial) = ?
- Tiempo medio de vida (exponencial) = ?



riódicamente los

lmente estos
cuando quieren

ntes y los servidores
valo entre anuncios

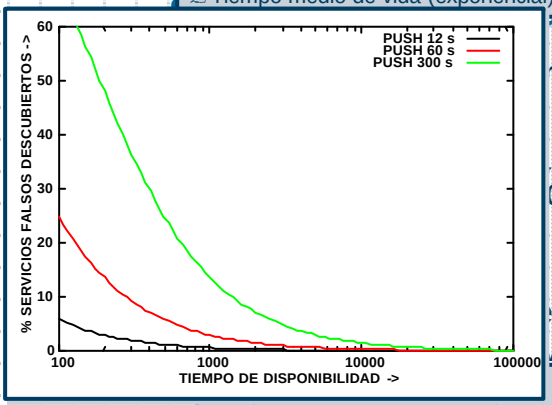
muc
pued
disp

- Número de mensajes por búsqueda = $\frac{?}{T_a}$
- Tasa de servicios descubiertos = $p ? \frac{?}{T_a} ? 1 ? e^{\frac{? T_a}{?}}$
- Tasa de servicios falsos descubiertos = $f ? 1 ? p$

19

Estudi

ESTUDIO ANALÍTICO:
- Tiempo entre anuncios = T_a
- Tiempo medio entre peticiones (exponencial) = ?
- Tiempo medio de vida (exponencial) = ?



riódicamente los

lmente estos
cuando quieren

ntes y los servidores
valo entre anuncios

muc
pued
disp

- Número de mensajes por búsqueda = $\frac{?}{T_a}$
- Tasa de servicios descubiertos = $p ? \frac{?}{T_a} ? 1 ? e^{\frac{? T_a}{?}}$
- Tasa de servicios falsos descubiertos = $f ? 1 ? p$

19

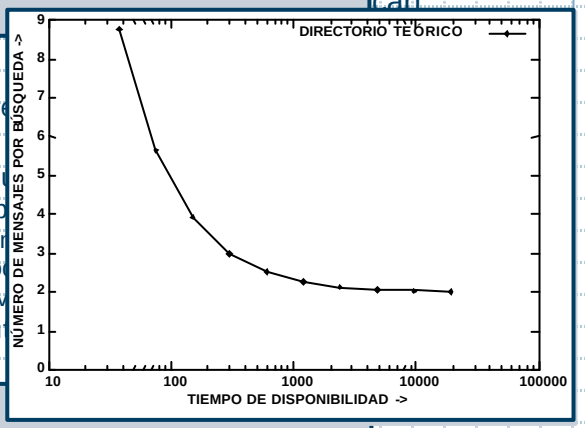
Estudio teórico
Solución III – Modo centralizado

- Directorio, nuevo elemento donde los servidores registran sus servicios y los clientes los buscan
- Ventajas:
  - Escalabilidad
  - Uso en áreas extensas
- Problemas:
  - Dependencia de un sistema central
  - ¿Cómo se descubre al directorio?:
    - Configuración manual
    - Anuncios periódicos del directorio
    - Búsqueda activa de los servidores y clientes
  - Entorno cambiante:
    - ¿Quién asume el papel de directorio?
    - Búsquedas de directorios frecuentes, por lo tanto costosas.

ralizado

dores
can

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
    1 fijo
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño de la caché = 100



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

Estudio propuestas IETF

- SLP:
  - Modo centralizado
  - Modo distribuido: modo pull con caché opcional
- SSDP:
  - Modo distribuido: modo pull y modo push
- DNS-SD:
  - Modo centralizado basado en DNS.
  - Modo distribuido basado en Multicast DNS (modo pull con respuestas multicast) o en LLMNR (modo pull)

¿Es necesario un nuevo protocolo?

- No han sido propuestos para entornos ubicuos
- Requisitos:
  - Minimizar consumo (número de transmisiones)
  - Funcionar sin infraestructura fija
  - Simple y poco costoso computacionalmente
- Criterios de diseño:
  - Adaptarse a entornos dinámicos como estáticos
  - Maximizar la cooperación entre los dispositivos
  - Adaptarse a las características de las aplicaciones

Diseño razonado

- Propuesta 1:
  - **No emplear directorio de servicios**
- Justificación:
  - Dispositivos limitados y móviles
  - Coste de la búsqueda de directorio
- Aunque si existe dispositivos fijos, que se comporte como un directorio

Diseño razonado

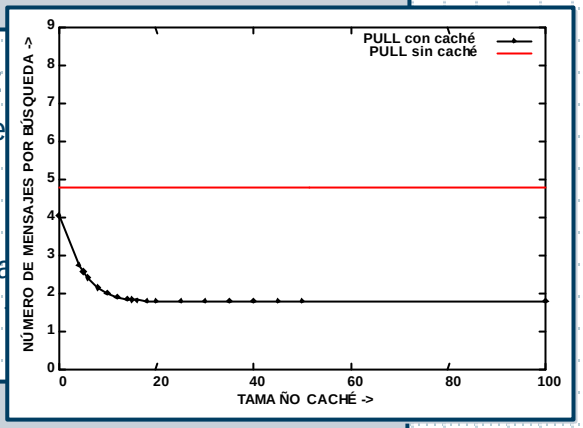
- Propuesta 2:
  - **Introducir caché de servicios remotos**
- Justificación:
  - Disminuye el número de mensajes transmitidos por búsqueda
- Problemas:
  - Disminuye la tasa de servicios descubiertos
  - Aumenta la tasa de servicios falsos descubiertos

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

ry Protocol

Introducción

- Justificación:
  - Disminuye e
    búsqueda
- Problemas:
  - Disminuye la
  - Aumenta la



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

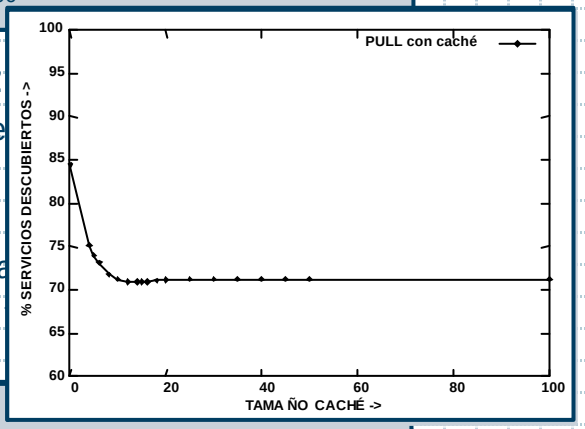
24

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

ry Protocol

Introducción

- Justificación:
  - Disminuye e
    búsqueda
- Problemas:
  - Disminuye la
  - Aumenta la



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

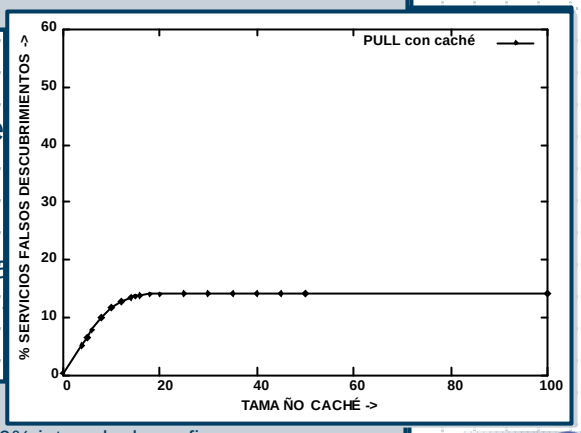
24

ry Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

Introducción

- Justificación:
  - Disminuye e
    búsqueda
- Problemas:
  - Disminuye la
  - Aumenta la



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

Pervasive Discovery Protocol

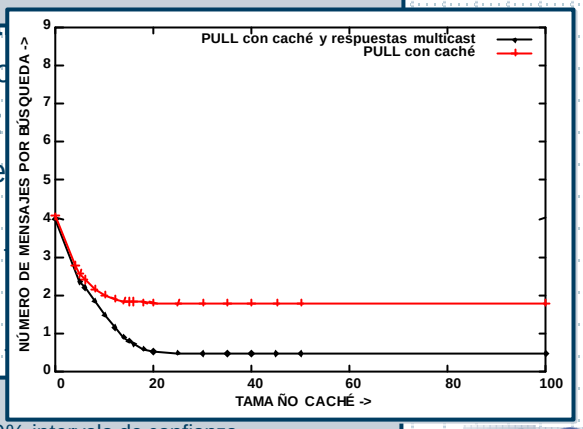
Diseño razonado

- Propuesta 3:
  - **Mezclar características de *pull* y *push***
  - Modo pull con caché y respuestas multicast
- Justificación:
  - Disminuye el número de mensajes transmitidos por búsqueda
  - Aumenta la tasa de servicios descubiertos
- Problema:
  - Aumenta la tasa de servicios falsos descubiertos

ery Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

- Modo pull co
- Justificación:
  - Disminuye e búsqueda
  - Aumenta la
- Problema:
  - Aumenta la



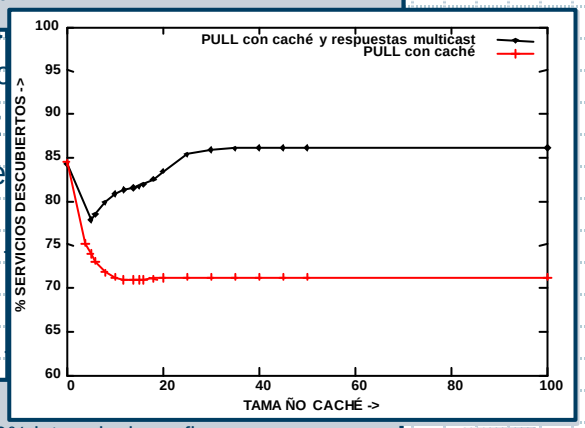
PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

25

ery Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

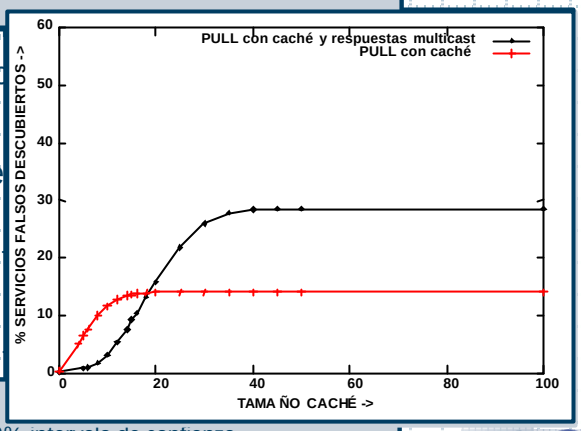
- Modo pull co
- Justificación:
  - Disminuye e búsqueda
  - Aumenta la
- Problema:
  - Aumenta la



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

25

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100



- Modo pull co
- Justificación:
  - Disminuye e búsqueda
  - Aumenta la
- Problema:
  - Aumenta la

PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

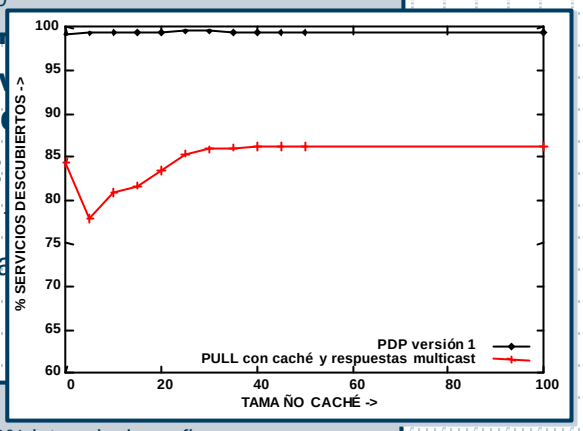
Diseño razonado

- Propuesta 4 (PDP v1):
  - **Enviar petición siempre**
  - **Incluir servicios ya conocidos en los mensajes de búsqueda**
- Justificación:
  - Aumenta la tasa de servicios descubiertos
  - Disminuye la tasa de servicios falsos descubiertos
- Problema:
  - Aumenta el número de mensajes por búsqueda

ery Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

- Incluir serv
  de búsque
- Justificación:
  - Aumenta la
  - Disminuye la
- Problema:
  - Aumenta el



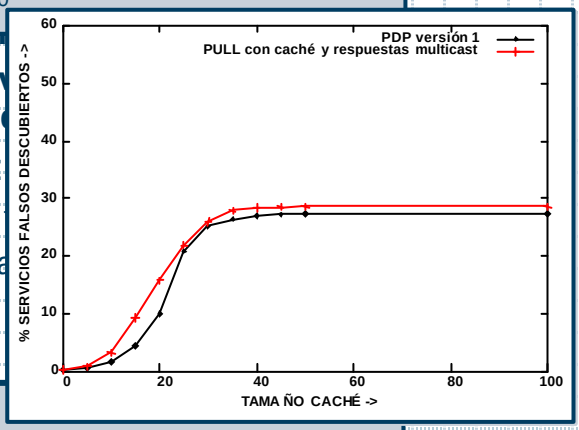
PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

26

ery Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

- Incluir serv
  de búsque
- Justificación:
  - Aumenta la
  - Disminuye la
- Problema:
  - Aumenta el



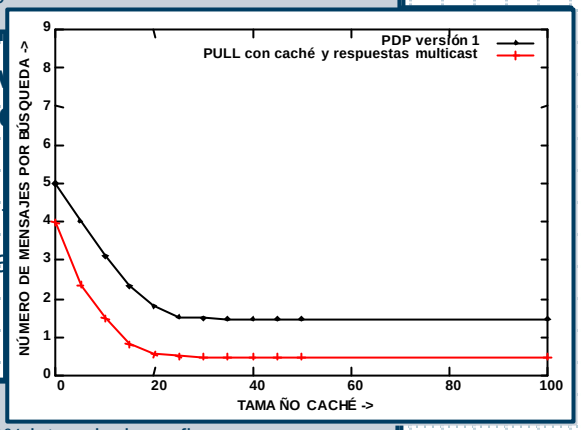
PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

26

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

- **Incluir serv**
- **de búsque**

- Justificación:
  - Aumenta la
  - Disminuye la
- Problema:
  - Aumenta el



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

Diseño razonado

- Propuesta 5 (PDP v2):
  - **Responder si conoces el servicio no sólo si lo ofreces**
  - **Transmitir sólo servicios no anunciados en otras respuestas**
  - **Priorizar respuestas de dispositivos con**:
    - **Mayor tiempo de disponibilidad**
    - **Mayor número de servicios conocidos**
- Justificación:
  - Comportamiento similar a un directorio: los dispositivos más limitados responden a menos peticiones de búsqueda
- Problema:
  - Aumenta la tasa de servicios falsos descubiertos

ery Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 40
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 500 seg. (20%), 2500 seg. (20 %), 4500 seg. (20%), 6500 seg. (20 %) y 9500 seg. (20%)
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 10 (500 seg. y 2500 seg.), 40 (4500 seg. y 6500 seg.) y 100 (9500 seg.)

lo ofreces

respuestas

Priorizar respue

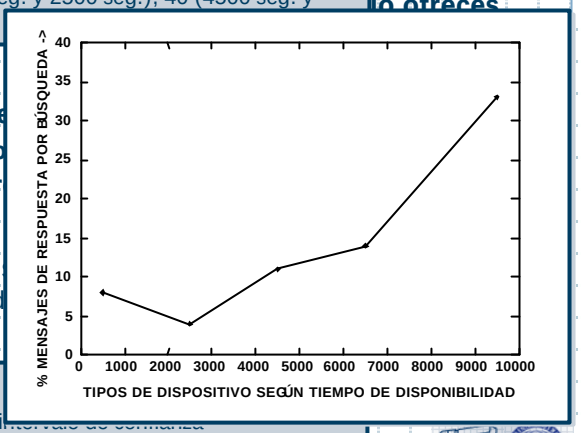
Mayor tiempo

Mayor númer

Justificación:

Comportamiento
limitados respond

Problema:



PRECISIÓN:
- 90 % nivel de confianza – 10% i

27

ery Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

Transmitir sólo
respuestas

Priorizar respu

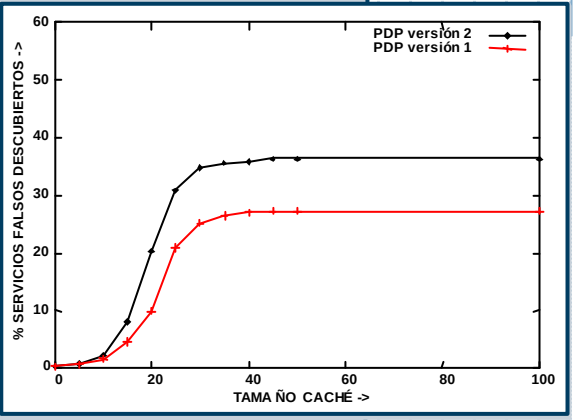
Mayor tiempo

Mayor núme

Justificación:

Comportamiento
limitados respond

Problema:



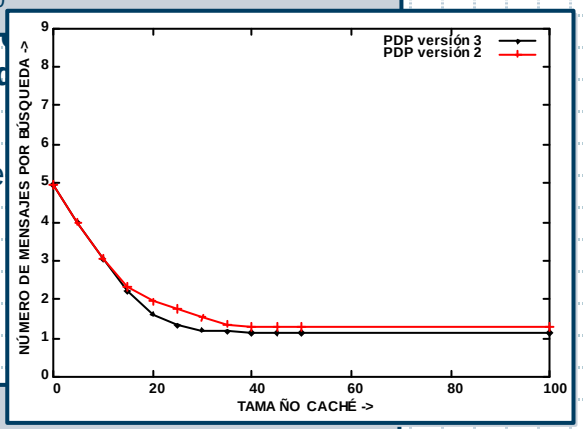
PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

27

Diseño razonado

- Propuesta 6 (PDP v3):
  - **Actualizar caché con servicios incluidos en los mensajes de búsqueda**
- Justificación:
  - Disminuye el número de mensajes por búsqueda

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

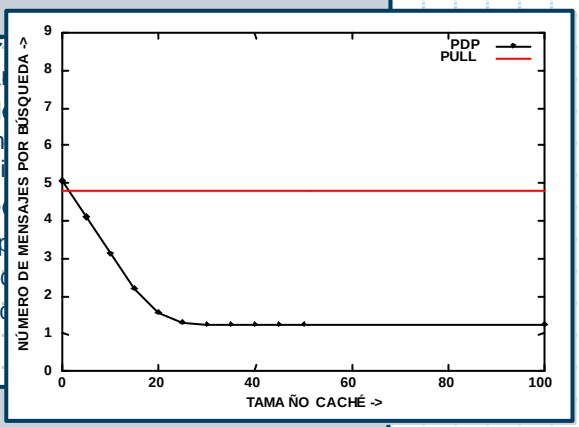


PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

Diseño razonado

- Propuesta 7 (PDP):
  - Introducir mecanismos de consistencia de cachés:
    - Primer mecanismo:
      - Tiempo de expiración en caché es el mínimo entre el tiempo de disponibilidad local (donde se almacena) y el tiempo de disponibilidad del dispositivo que lo anuncia
    - Segundo mecanismo:
      - Mensaje para borrar servicios almacenados en la caché:
        - Detección de cambio de red o apagado de dispositivo
        - Detección de errores de acceso al servicio
- Justificación:
  - Reducir el número de entradas falsas en las cachés

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

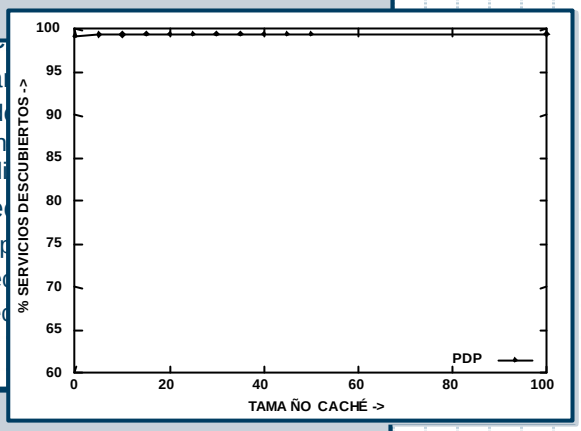
ry Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

Introducir the

- Primer meca
  - Tiempo d de dispon disponibili
- Segundo me
  - Mensaje p
    - Dete
    - Dete

- Justificación:



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

29

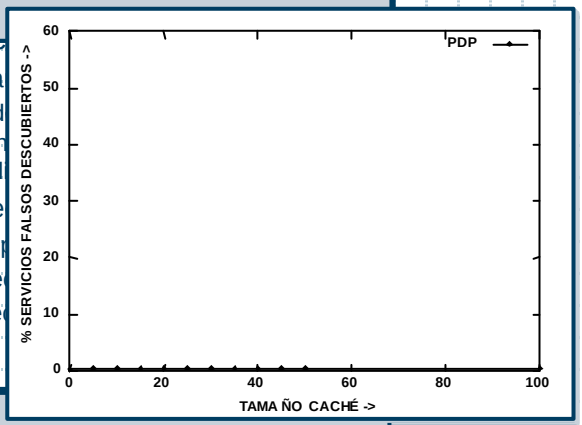
ry Protocol

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100

Introducir the

- Primer meca
  - Tiempo d de dispon disponibil
- Segundo me
  - Mensaje p
    - Dete
    - Dete

- Justificación:



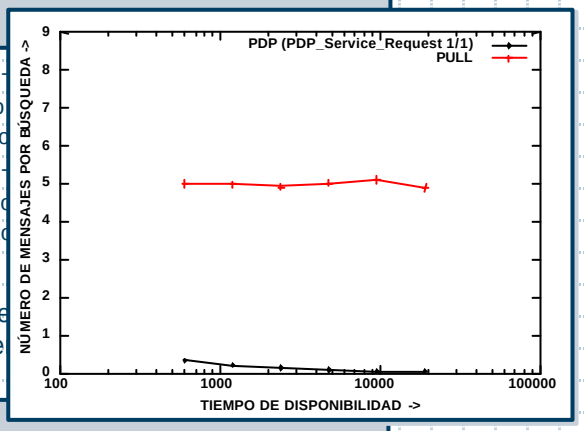
PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

29

Diseño razonado

- Propuesta 8 (PDP 1/1 – 1/n):
  - Diferenciar tipos de búsqueda:
    - Una petición – una respuesta (1/1):
      - El servicio que se ofrece es independiente del servidor que lo ofrece
      - Sólo responde un servidor
    - Una petición – varias respuestas (1/n):
      - Obtener todos los servidores que ofrecen un tipo de servicio
      - Obtener todos los servidores de todos los tipos: "ALL"
- Justificación:
  - Reducir el número de mensajes por petición al tener en cuenta el uso que hacen las aplicaciones de las respuestas obtenidas

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 37.5 seg. – 19200 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 100



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

Descripción del protocolo

- Descripción detallada (tipo RFC):
  - Formato de descripción de servicios de SLP ("**`service URL`**")
  - Dispositivos:
    - PDP_SA anuncia servicios
    - PDP_UA localiza servicios
    - Tiempo de disponibilidad
    - Caché de servicios por interfaz de red
  - Tipos de mensajes:
    - PDP_Service_Request (1/1 y 1/n)
    - PDP_Service_Reply
    - PDP_Service_Deregister
- Consideraciones sobre seguridad

Análisis de prestaciones

- Prestaciones del protocolo, respecto:
  - Tiempo de disponibilidad
  - Número de dispositivos
  - Tamaño de la caché
- Comparativa con otros protocolos, en:
  - Escenarios homogéneos
  - Escenarios heterogéneos
  - Escenarios con un dispositivo fijo
  - Soporte a aplicaciones:
    - Tipo buscador
    - Tipo "una petición- una respuesta"

Pervasive Discovery Protocol

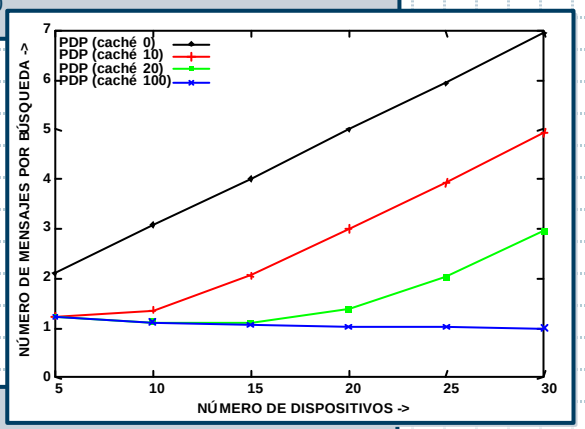
Análisis de prestaciones

33

y Protocol

PARAMÉTROS SIMULACIÓN:

- Número medio de dispositivos = 5 – 30
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100



PRECISIÓN:

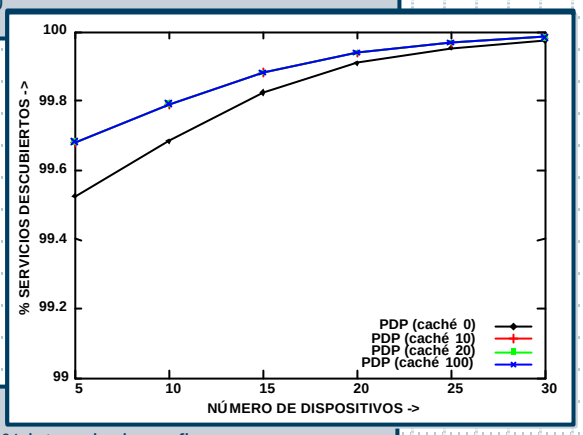
- 90 % nivel de confianza – 10% intervalo de confianza

33

y Protocol

PARAMÉTROS SIMULACIÓN:

- Número medio de dispositivos = 5 – 30
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100



PRECISIÓN:

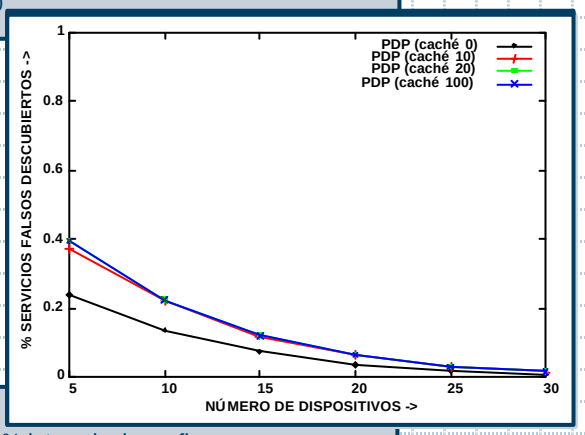
- 90 % nivel de confianza – 10% intervalo de confianza

33

y Protocol

PARAMÉTROS SIMULACIÓN:

- Número medio de dispositivos = 5 – 30
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 0 - 100



PRECISIÓN:

- 90 % nivel de confianza – 10% intervalo de confianza

33

Pervasive Discovery Protocol

Análisis de prestaciones

Consideraciones sobre tiempo de disponibilidad

- Modela la movilidad de un dispositivo
- Concepto existente en todos los protocolos de descubrimiento de servicios con cachés locales
- Propuestas nuevas en PDP:
  - Calcular el tiempo de expiración como el mínimo del tiempo de disponibilidad local y remoto
  - Priorizar las respuestas de los dispositivos para que respondan antes los menos limitados (menos móviles)
- ¿Cómo influyen los errores en este parámetro?
  - Estudios de simulación recomiendan estimar a la baja el tiempo de disponibilidad

34

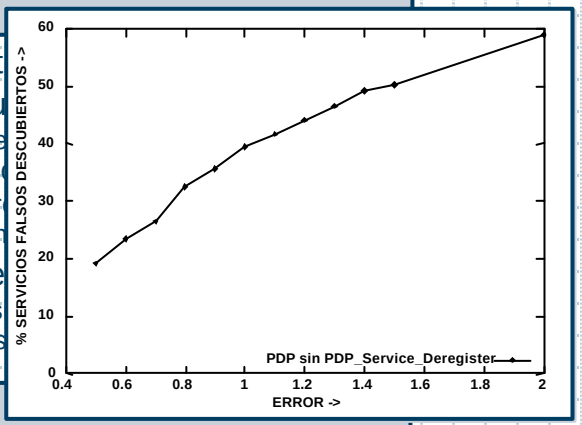
Pervasive Discovery Protocol

Análisis de prestaciones

Consideraciones sobre tiempo de disponibilidad

PARAMÉTROS SIMULACIÓN:
- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg.
  - Tiempo de disponibilidad = ERROR * Tiempo medio de vida
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 100

descubrimient
- Propuestas nu
  - Calcular el tie de disponibili
  - Priorizar las r respondan an
- ¿Cómo influye
  - Estudios de s tiempo de dis



PRECISIÓN:
- 90 % nivel de confianza – 10% intervalo de confianza

34

Pervasive Discovery Protocol

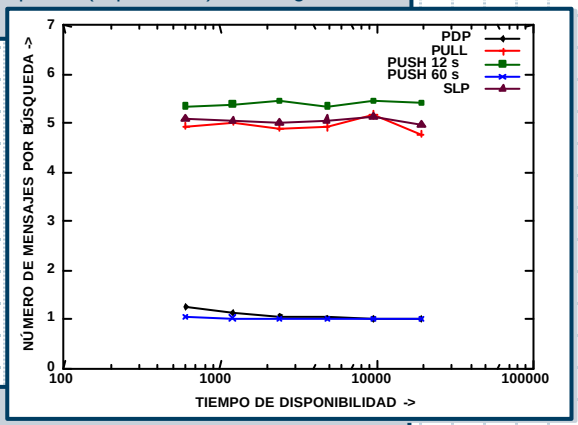
Análisis de prestaciones

PDP vs Otros protocolos

35

PARAMÉTROS SIMULACIÓN:

- Número medio de dispositivos = 20
- Número de tipos de servicios = 5
- Cada dispositivo:
  - Tiempo medio de vida (exponencial) = 600 seg. – 19200 seg.
    20% 10 veces mayor que los demás
  - Tiempo medio entre búsquedas (exponencial) = 60 seg.
  - Tamaño caché = 100

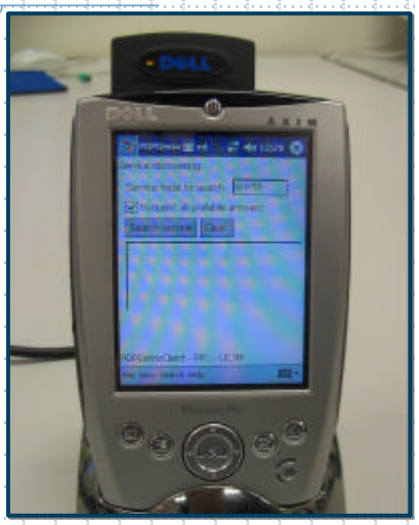


PRECISIÓN:

- 90 % nivel de confianza – 10% intervalo de confianza

35

Implementación



- J2SE
- J2ME:
  - Pocket PC
  - J9 IBM
  - Tamaño: 18 KB.

Índice

- Introducción
- Estado del arte
- PDP: *Pervasive Discovery Protocol*
- SDA: *Service Discovery Agent*
  - Motivación
  - Objetivos
  - Descripción SDA
  - FIPA Ad-Hoc
- Conclusiones y trabajos futuros

Motivación

- Adaptación de estándares FIPA a entornos móviles:
  - Nuevos elementos en la arquitectura: MA y CA [FIPA00014,2002]
  - Características de terminales móviles [FIPA00091,2001]
  - Transporte de mensajes [FIPA00092,2001], [FIPA0093,2001]
- Adaptación a entornos ubicuos:
  - Creación del comité técnico FIPA Ad-Hoc a principios de 2002
- Servicio de páginas amarillas (DF):
  - Fundamental en entornos cambiantes y sin infraestructura
  - Federación de DF no es un mecanismo válido:
    - Búsqueda de un servicio concreto siempre implica comunicación con sistemas remoto
    - No existen mecanismos para la formación dinámica de federaciones de DFs

Objetivos

- Adaptación del servicio de páginas amarillas definido en FIPA (DF) para su funcionamiento en redes sin infraestructura.
- Requisitos:
  - Mantener el DF como servicio de páginas amarillas para los agentes residentes en la plataforma
  - Eliminar la federación de DF
  - Emplear un nuevo mecanismo adaptado a las características de los entornos ad-hoc

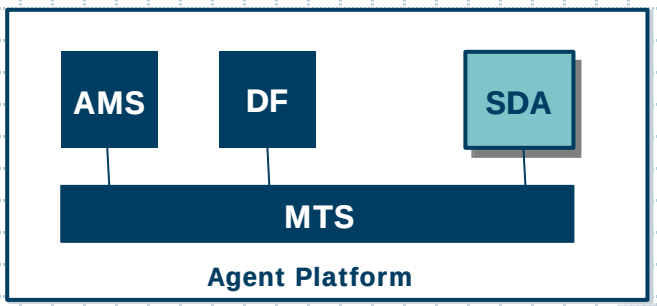
Descripción SDA

- Service Discovery Agent (SDA): nuevo elemento en la arquitectura FIPA que realiza búsquedas servicios/agentes remotos en redes ad-hoc
- Solución 1:
  - SDA implementa su propio mecanismo de descubrimiento
  - Problema:
    - Definición de mensajes ACL de difusión
- Solución 2:
  - SDA emplea un mecanismo de descubrimiento existente
  - Problema:
    - Si emplea cualquier mecanismo: ¿cómo se consigue la interoperabilidad entre plataformas que emplean distintos mecanismos?
  - Emplear un único mecanismo eficiente en este tipo de entornos ? ?PDP

Service Discovery Agent

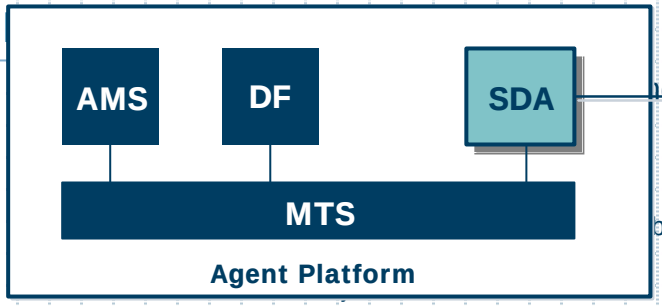
Descripción SDA

- Service Discovery Agent (SDA): nuevo elemento en la arquitectura FIPA que realiza búsquedas servicios/agentes remotos en redes ad-hoc
- Solución 1:
  - SDA implementa su propio mecanismo de descubrimiento



40

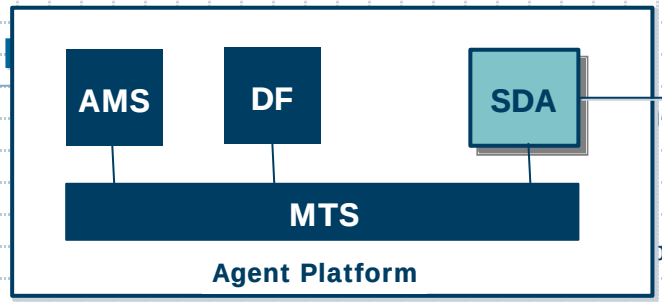
Service Discovery Agent



- Solución 2:
  - SDA emplea un mecanismo de descubrimiento existente

40

Service Discovery Agent



- Definición de mensajes ACL de difusión
- Solución 2:
  - SDA emplea un mecanismo de descubrimiento existente
  - Problema:
    - Si emplea cualquier mecanismo: ¿cómo se consigue la interoperabilidad entre plataformas que emplean distintos mecanismos?
  - Emplear un único mecanismo eficiente en este tipo de entornos ? ?PDP

40

Descripción SDA – PDP

- Funcionalidad
  - Interacción con el DF:
    - Registro del SDA en el DF (sda@*home_agent_platform_name*)
    - Ofrece la misma funcionalidad que el DF a los agentes pero en búsquedas remotas: **`register`**, **`modify`**, **`deregister`**, **`search`**.
  - Interacciona con PDP:
    - PDP_SA para el registro, modificación y eliminación de registro de agentes.
    - PDP_UA para realizar búsquedas de servicios remotos

Service Discovery Agent

Descripción SDA – PDP

DF	SDA	PDP
`register` (df_agent_description)	**`register`** (df_agent_description)	df_agent_description como servicio local
`modify` (df_agent_description)	**`modify`** (df_agent_description)	Modifica servicio local Envía PDP_Service_Reply gratuito
`deregister` (df_agent_description)	**`deregister`** (df_agent_description)	Elimina servicio local Envía PDP_Service_Deregister

Descripción SDA – PDP

DF	SDA	PDP
search (max-depth = 0, max-results)		
search (max-depth != 0, max-results)	**search** (max-results)	max-results = 1 PDP_Service_Request _1/1
		max-results > 1 PDP_Service_Request _1/n
		max-results < 1 PDP_Service_Request _1/n

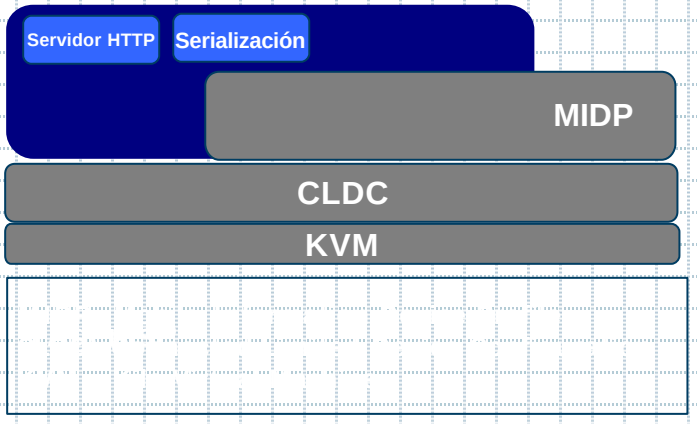
Ontología SDA

- **service-agent-ontology**:
  - Basada en **fipa-agent-management** [FIPA00023].
- Objetos:
  - **fipa-agent-management:**
    - **df-agent-description**
    - **agent-identifier**
    - **service-description**
  - **adhoc-search-constraints**
- Funciones:
  - **fipa-agent-management:**
    - **register**
    - **deregister**
    - **modify**
  - **search**

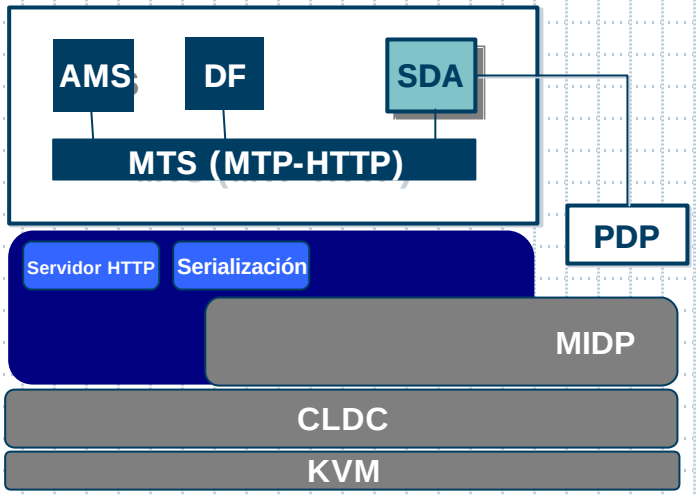
TC FIPA Ad-Hoc

- Participación:
  - Febrero 2002: respuesta al Call for Technology:
    - UMBC (University Maryland, Baltimore County)
    - Siemens AG
    - MIT Media Lab Europe
    - UC3M (Universidad Carlos III de Madrid)
  - Junio 2002: contribución al White Paper "*Agents in Ad-Hoc Environments*"
  - Julio 2002: invitación al FIPA 26
  - Miembro del grupo
- Propuestas aceptadas:
  - **Eliminar la federación de DF**
  - **Mantener el DF obligatorio**

Implementación



Implementación



44

Índice

- Introducción
- Estado del arte
- PDP: *Pervasive Discovery Protocol*
- SDA: *Service Discovery Agent*
- Conclusiones y trabajos futuros:
  - Conclusiones
  - Trabajos futuros
  - Resultados de la investigación

45

Conclusiones

- Pervasive Discovery Protocol:
  - Propuestas:
    - Funcionamiento distribuido mezcla de modo push y pull
    - Cooperación y coordinación de los dispositivos
    - Tiempo de disponibilidad
  - Resultados:
    - Análisis de prestaciones:
      - Reduce el número de mensajes por búsqueda
      - Alcanza tasa de servicios descubiertos altas y servicios descubiertos falsos bajas
      - Tiende a un directorio de forma automática
    - Descripción detallada del protocolo
    - Implementación en dispositivos limitados

Conclusiones

- Service Discovery Agent:
  - Propuestas:
    - Mantener el DF como servicio de páginas amarillas para los agentes
    - Sustituir el mecanismo de federación de DFs por SDA-PDP
  - Resultados:
    - Fácil de integrar en la arquitectura FIPA
    - Proporciona un servicio de páginas amarillas eficiente en redes ad-hoc
    - Viabilidad de implementación en dispositivos limitados

Conclusiones y trabajos futuros

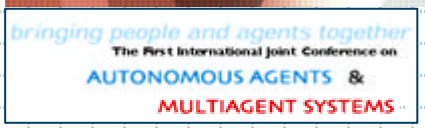
Trabajos futuros

- Pervasive Discovery Protocol:
  - Formato de descripción de servicios
  - Aprendizaje del tiempo de disponibilidad
  - Aprendizaje del tamaño de la caché
  - Nuevos mecanismos para generar retardo aleatorio antes de responder
  - Problema de los nodos ocultos
- Agentes:
  - Adaptación del AMS
  - Adaptación del MTS
- Computación ubicua:
  - Seguridad basada en relaciones de confianza distribuida
  - Soporte para desarrollo de aplicaciones basadas en contexto

48

Conclusiones y trabajos futuros

Resultados de la investigación



49

Resultados de la investigación

- Proyectos de investigación:
  - E-Ticket: Arquitectura de Aplicaciones de Tarjetas Inteligentes para el Sector de Servicios, CICYT-FEDER 2FD1997-1269-C02-01. Diciembre 1999 - Diciembre 2001
  - Beca Cátedra Nokia – UC3M
  - UBISEC: Ubiquitous Networks with a Secure Provision of Services, and Context Delivery. Sixth Framework Programme. Contract no. 506929. Enero 2004 – Diciembre 2005
  - Everyware: servicios personalizados en un entorno de computación ubicua. MCyT TIC2003-08995-C02-01. Diciembre 2003 - Diciembre 2006

TECNOLOGÍAS MIDDLEWARE PARA EL DESARROLLO DE SERVICIOS EN ENTORNOS DE COMPUTACIÓN UBICUA

Autora: Mª Celeste Campo Vázquez
Director: Andrés Marín López
7 de Mayo de 2004