# Techniques for Service Discovery in Smart Environments

Michele Girolami

Michele.girolami@isti.cnr.it
girolami@di.unipi.it
RHS 2013

**Wireless Networks Laboratory (WN)**


## Topics

- *Mobile Computing*
  - Service and Resource Discovery
- *Device Integration*
  - Wireless Sensor Networks
  - 802.15.4 and ZigBee

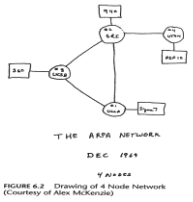
# Index

# Reference Scenario
# Internet of Things

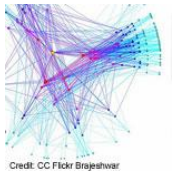
IoT paradigm

- describes the architectures for integrating *objects or things*
- identifies the Internet-based technologies that enable such architectures.

1969 Arpanet | **1990 The Toaster** | **1991 Weiser's article** | **1999 IoT is coined** | 2000-2003 First EU projects | 2011 IPv6 launch




Credit: CC Flickr - G A R N E T


Credit: CC Flickr Brajeshwar

# Reference Scenario

# IoT in the Smart Environment

A SE is a *physical place* delimited by boundaries and whose objects use the *context-information* to assist the users to accomplish his/her tasks.

The key-aspects:

- **service-orientation**
- **context-awareness**


Reference Scenario

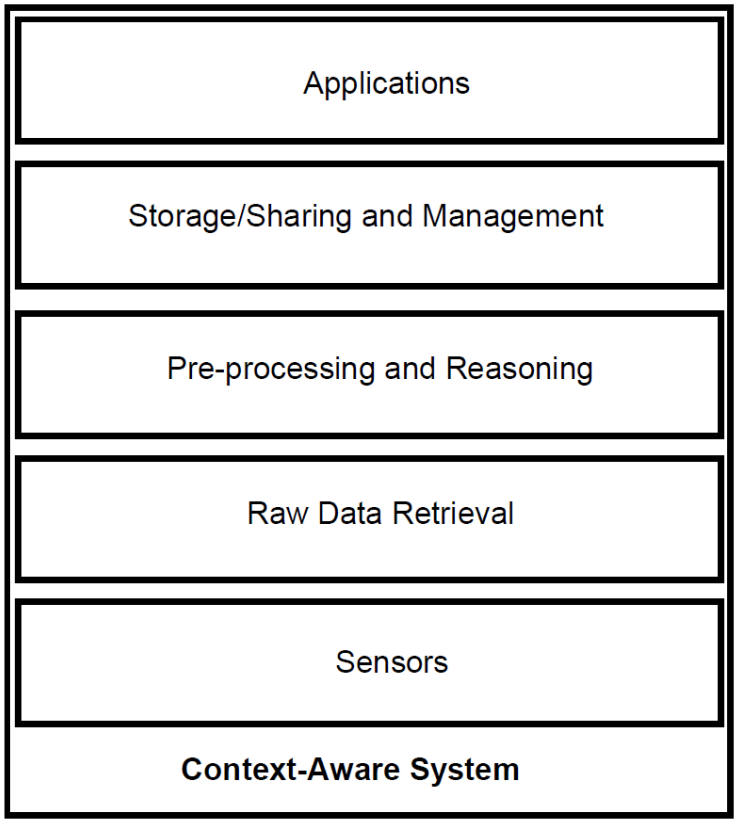
# IoT in the Smart Environment

Adaptation of the services to the changes in the environment:

**The context**

**External** – information gathered from hardware sensors

**Internal** – information that describes the user's goal and tasks.

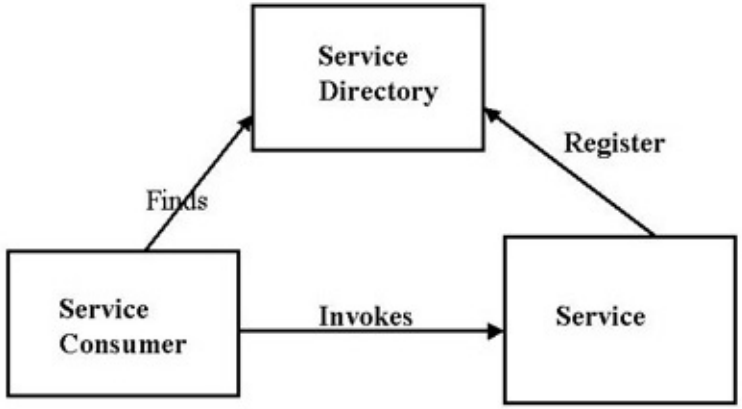


# Reference Scenario
# IoT in the Smart Environment

The objects in a SE provide **services** to humans.

Service have to be:

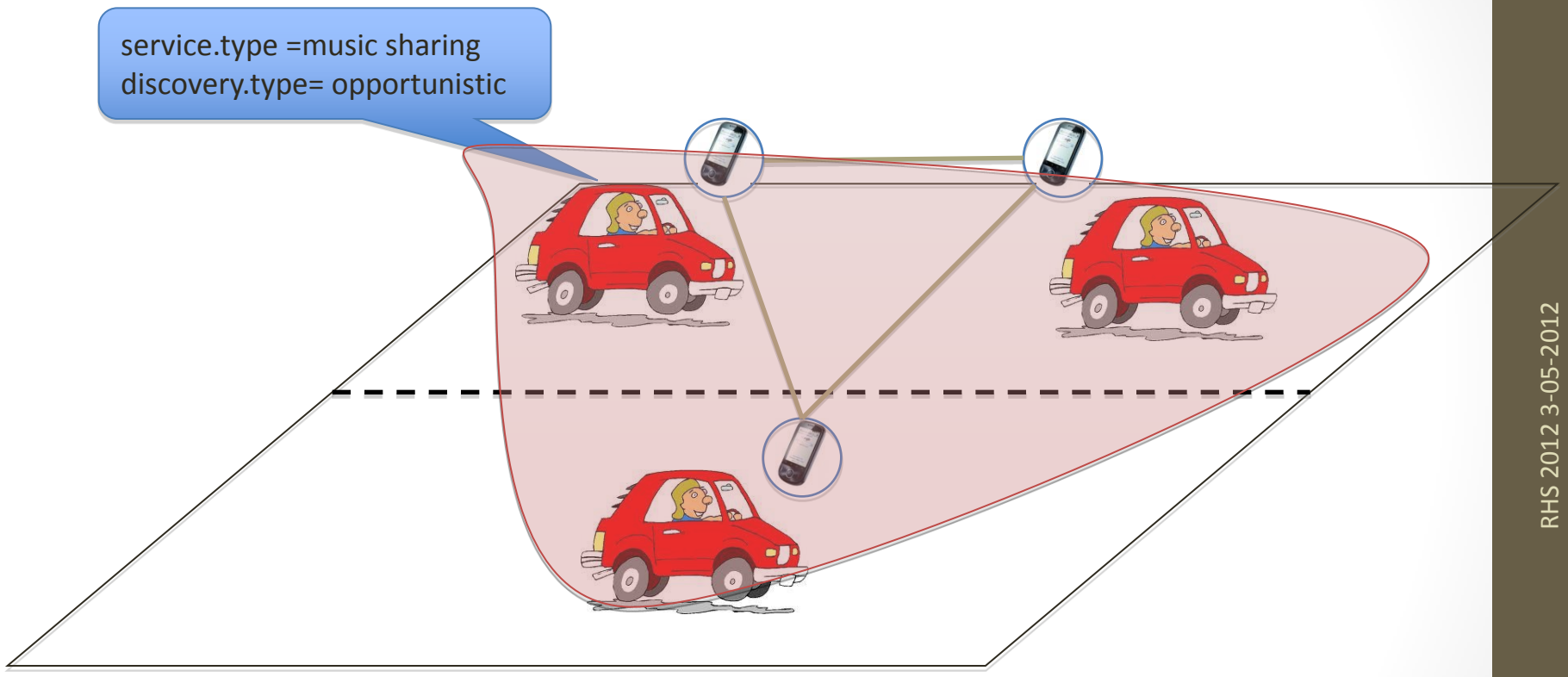
- discovered and accessed – SD protocols
- described – service description languages
- Independent form other services – compose services together




**SOA conceptual framework**

# Smart Environment

# VANET

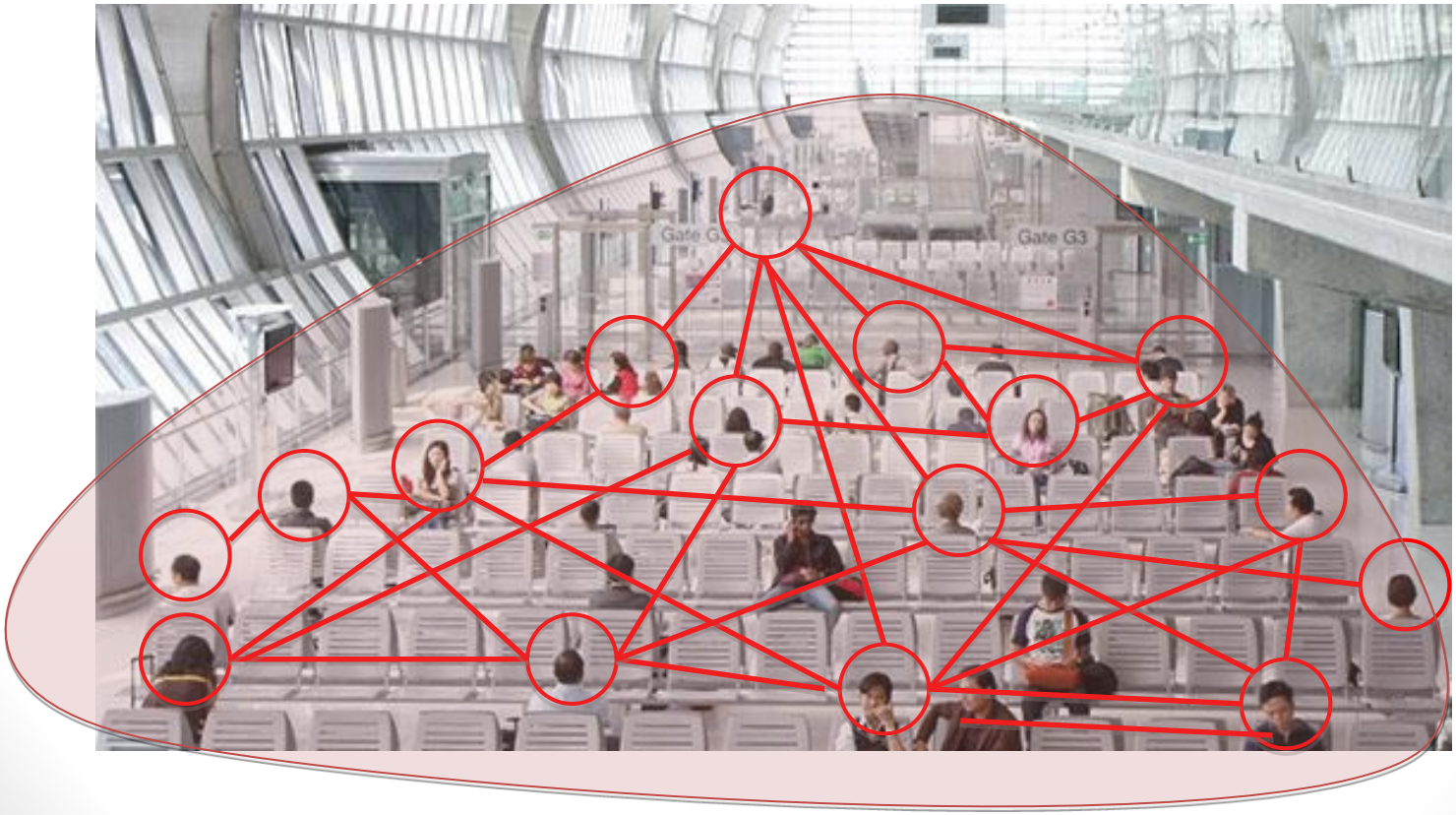


# Smart Environment
# Mobile Spaces

- SE  is a group of devices discovered in the interior (cabin) of the car.
  - Devices/Sensors installed in the car (KAN bus) or Mobile phone and game console used by passengers

- The car space can be enlarged by considering the connectivity provided by the next generation of  vehicular networks
  - Opportunistic communications,
  - Car-To-Car communication (IEEE 802.11p - WAVE )
  - Smart road sinals

# Smart Environment
# PSN – Pocket Switched Networks



# Smart Environment
# PSN – Pocket Switched Networks


Attribution (CC 2.0) :http://www.flickr.com/photos/ceiling/2374266108/sizes/l/

# Smart Environment
# PSN – Pocket Switched Networks

- **Scalability:** is the main issues in large public spaces
- **Device heterogeneity**: different hw/sw features
- **Mobility**: the person moves quickly in the environment, and its local area of interest change continuously
  - **Stationary Devices/Services**
    - Check-in desk, Restaurant, Entertainment,  Touristic assistance,
  - **Mobile Devices/Services**
    - Thousands of passengers with smart phones

# Reference Scenario

# IoT – Enabling Technologies

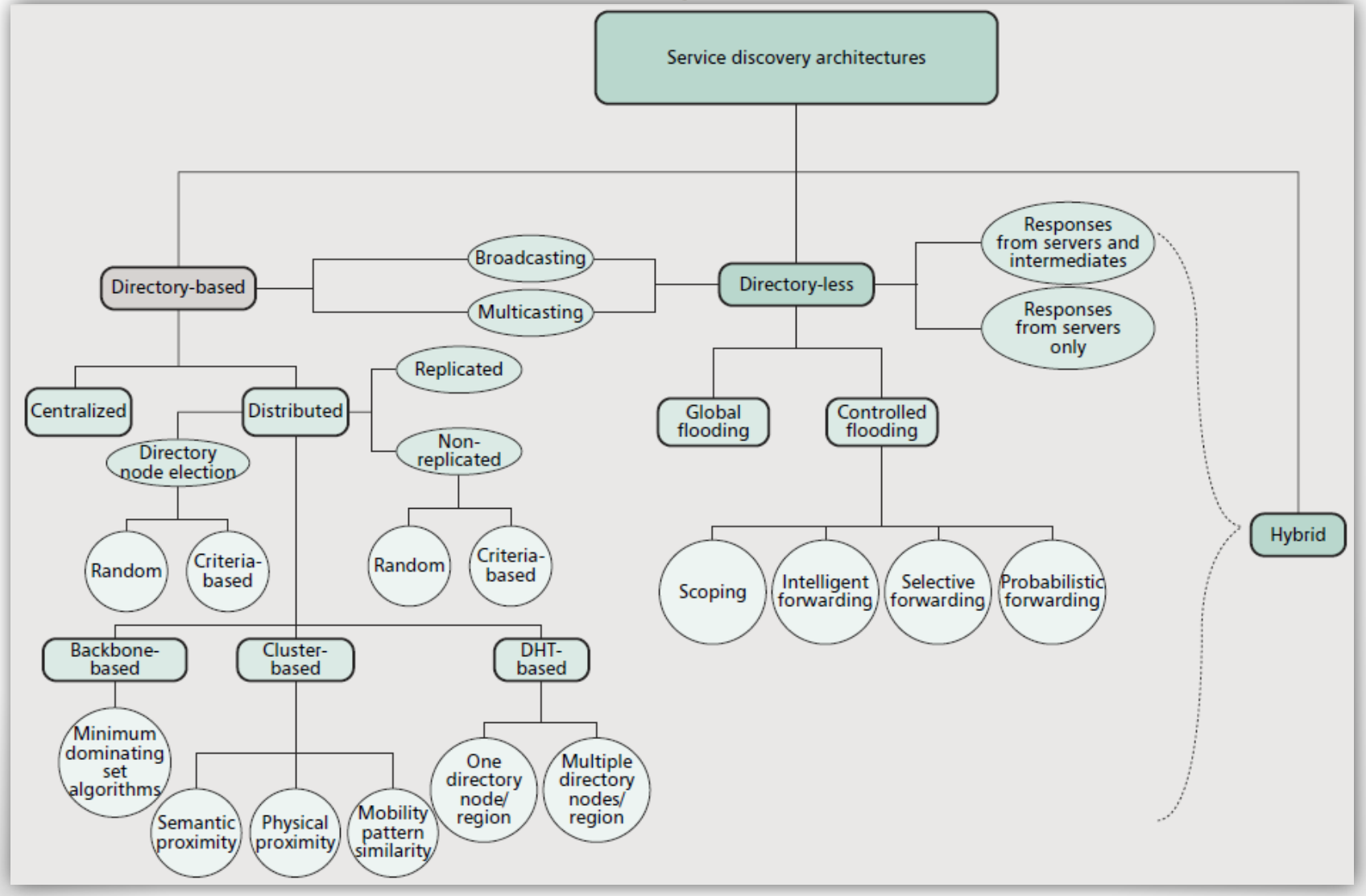
Some technologies enable the implementation of the IoT paradigm:

- **Identification, sensing and communication**
  - RFID systems
  - IPv6
  - Wireless Sensor Networks
    - ZigBee
    - 6LowPan
- **Platforms**
  - Combo technologies such as: MQTT/Rest/Json
  - OSGi Framework
  - .NET Micro Framework
  - TinyOS
  - Contiki

# Reference Scenario
# Service Discovery in the SE

# Reference Scenario

# Service Discovery in the SE



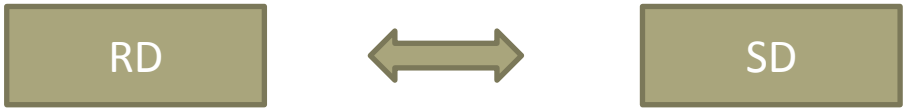
# Resource and Service Discovery

A **Resource** is any source of supply:

- File-system
- Memory space
- CPU capability

that can be provided as a **Service**

The **Resource Discovery (RD)** is any mechanism that is providing capability to locate a resource in the network



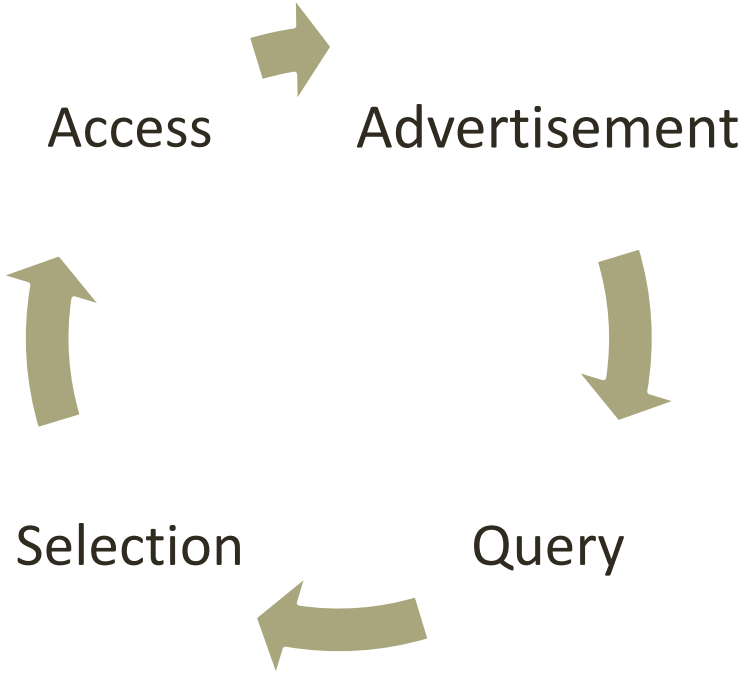

K. Vanthournout, G. Deconinck, R. Belmans, A taxonomy for resource discovery, Personal Ubiquitous Computing 9 (2) (2005) 81–89.

# Resource Discovery Design

- Centralized vs decentralized architectures
- Search technique and the query matching strategy
- Network topology (ie. star, tree or mesh topologies)
- Scale of the network:
  - internet-scale
  - enterprise-scale system
  - local-scale systems

# Resource Discovery Process

1. To advertise the resources
2. To query about the resources provided by the providers
3. To select the most suitable resource
4. To access to the resource

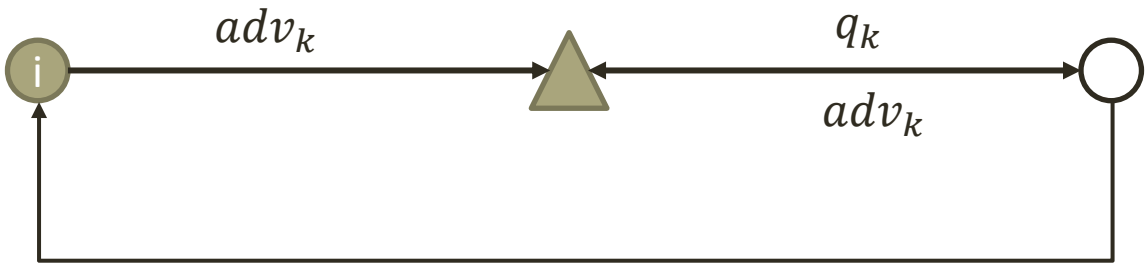


# Index

# Centralized Architecture

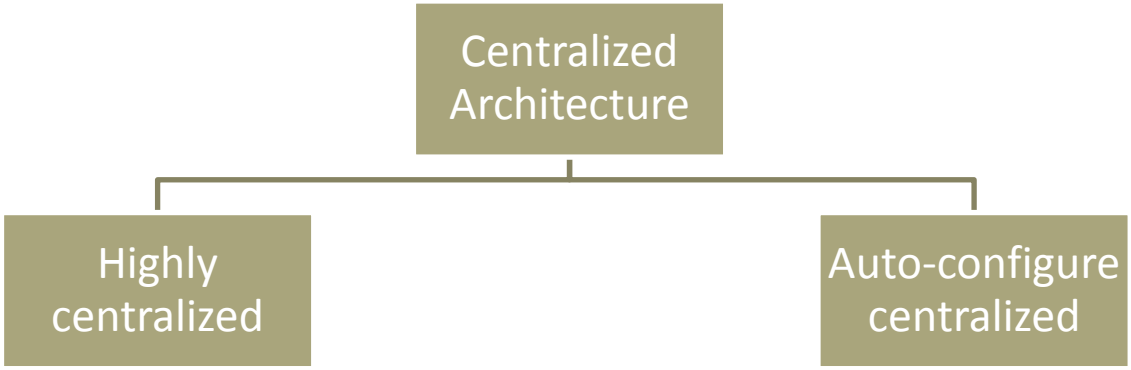
- **Resource Directory**: resources available on the network
- **Resource Provider**: advertises the resource (adv)
- **Resource Client**: queries the Service Directory

| *Node* | *Advertisements* |
|---|---|
| i | $adv_h, adv_k$ |
| j | $adv_x, adv_y$ |



## 2. RD Architectures

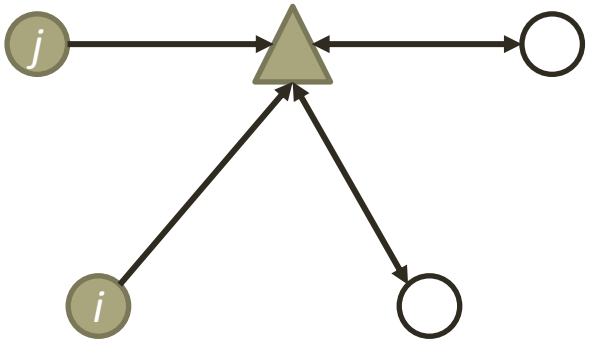
# Centralized Architecture

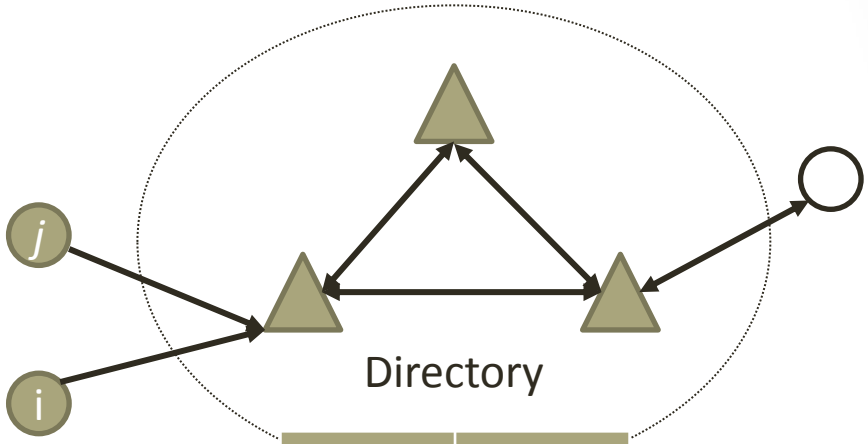



- Static configuration:
  - directory pre-configured
  - clients know the RD URLs
  - directory must be always reachable
- Variable number of RD nodes
  - can be dynamically elected
  - can be removed or added
  - need a synchronization strategy or replicas
- Dynamic configuration:
  - Clients discovers the directory with ie. m-cast or b-cast announces

# Centralized Architecture

| *Node* | *Advertisements* |
|---|---|
| i | $adv_h, adv_k$ |
| j | $adv_x, adv_y$ |




| *Node* | *Advs* |
|---|---|
| i | […] |
| j | […] |

U. C. Kozat and L. Tassiulas, “Service Discovery in Mobile Ad Hoc Networks: An Overall Perspective on Architectural Choices and Network Layer Support Issues,” *Ad Hoc Networks*, vol. 2, no. 1, 2004, pp. 23–44.

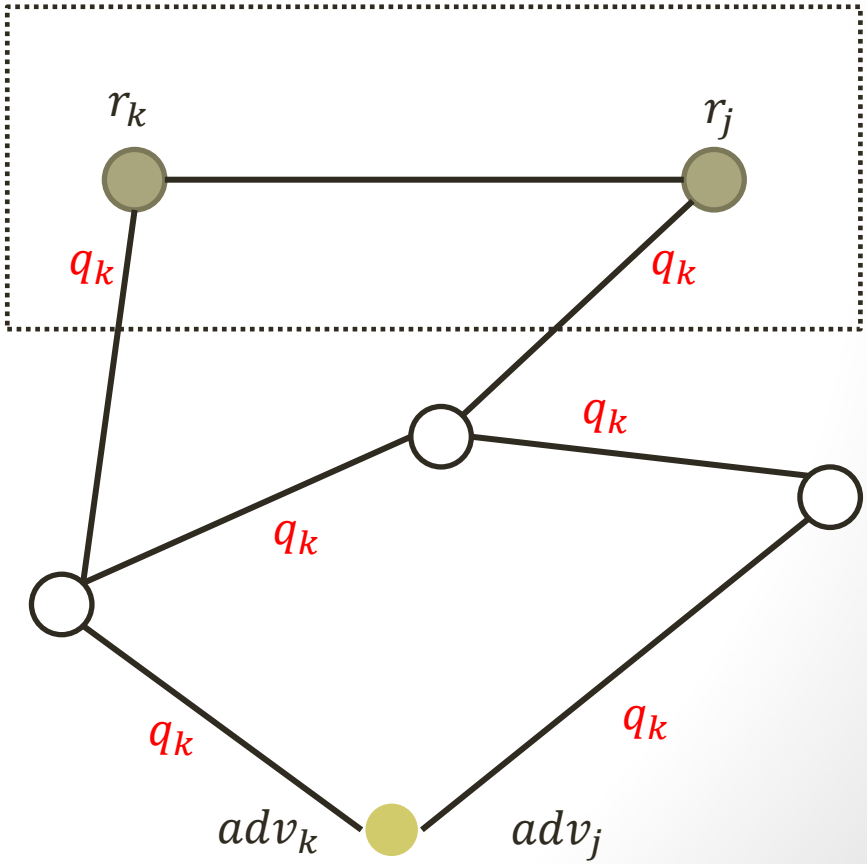
# Decentralized Architecture

- Resource Provider:
  - announces the availability of the resources to the whole network
  - answers to the client queries
- Resource Client: injects the queries into the network and waits for a set of responses

# Decentralized Architecture

$$q_k = \begin{cases} type = k \\ access\ cost = c \\ location = [x_1, y_1], [x_2, y_2], \\ [x_3, y_3], [x_4, y_4] \end{cases}$$

$$adv_k = \begin{cases} type = k \\ access\ cost = d \\ location = [x, y] \\ URL \end{cases}$$



# Index

# Paradigms for Propagation

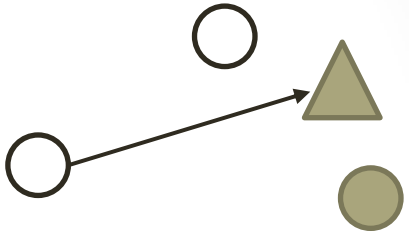
- How to propagate packets containing the queries or advertisements into the network

- The choice of the best propagation method depends on:
  - Underlying network topology
  - Communication media
  - Scale of the network

# Paradigms for Propagation

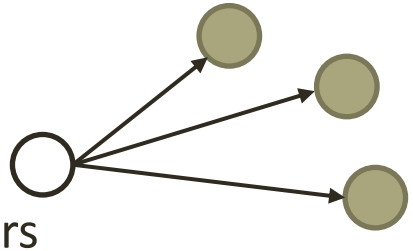
*Unicast*

- 1 sender  to 1 receiver
  - E.g. the query is sent to one directory



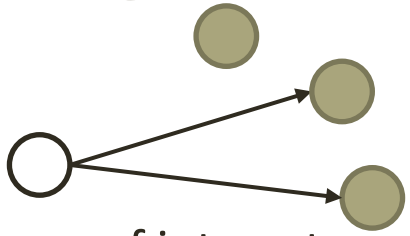
*Broadcast*

- 1 sender to all receivers
  - E.g. the query is sent to all the providers



# Paradigms for Propagation

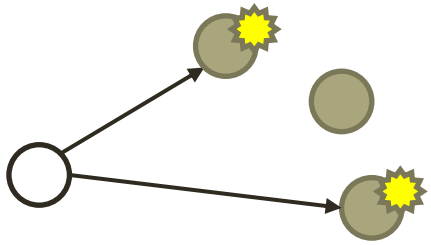
*Multicast*



- 1 sender to many receivers
  - E.g. providers and clients join some groups of interest
  - $q_k$ is only forwarded to the providers belonging to group *k*

*Anycast*

- *1 sender to many top-receivers*
- *The receivers are chosen according to several metrics like:*
  - *Closeness to the sender*
  - *Minimal load*

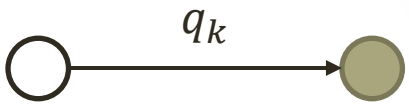


# Discovery Modes

- How clients learn about the resources the network provide
- The choice of the best discovery mode depends on:
  - Network density
  - Number of available resources
  - Popularity of the resources

# Discovery Modes

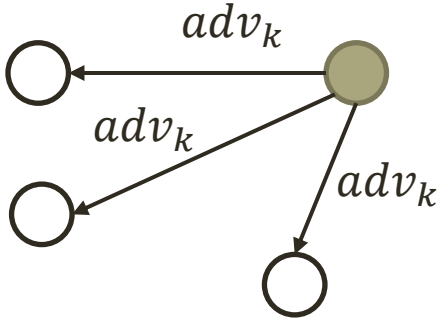
*Reactive*




- *Clients explicitly send a query to the directory agent(s) or to a set of providers*

*Proactive*

- *Clients receive resource advertisements without asking for them*
- *Providers announce/refresh the resources as soon as they are available*



# Information Delivery Modes

- How to share information about resources in the network by reducing the packet propagation

*Caching*

- Nodes cache successive resource advertisements in order not to repeat the search later
  - To manage stale-information stored in the cache
  - To keep multiple cache instances: on the resource clients and on the intermediate nodes

*Hello Messages*

- The information stored in the cache require to be updated by regular *Hello* messages
  - To tune the message rate in order to avoid much of overhead
  - To forward the messages only to 1-2 hop neighbours

# Information Delivery Modes

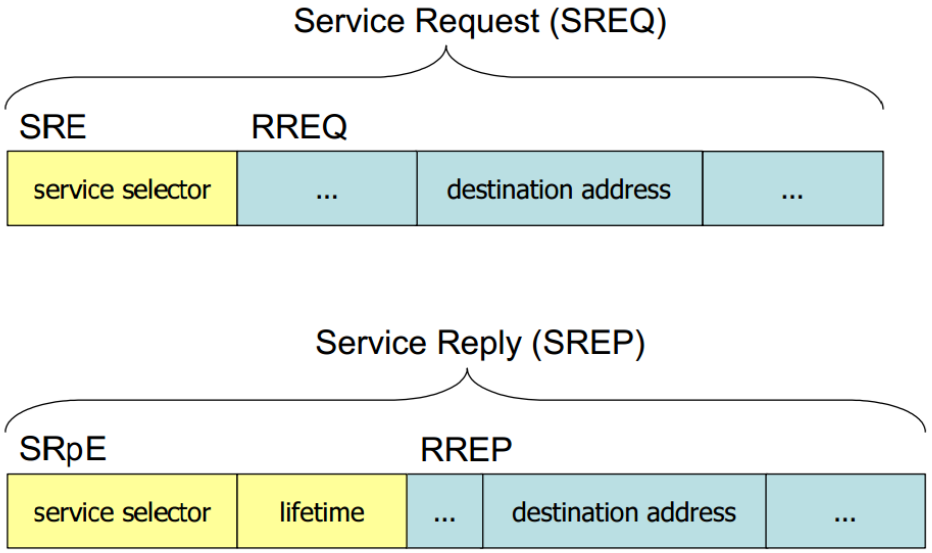
*Piggybacking*

- To exploit existing protocols in order to deliver information about available/unavailable resources:
  - routing packets by adding an extra payload delivering resource information
  - to manage constraints on the maximum packet size used in different environments on physical layer

Sotirios E. Athanaileas, Christopher N. Ververidis and George C. Polyzos, Optimized Service Selection for MANETs using an AODV-based Service Discovery Protocol

# Information Delivery Modes

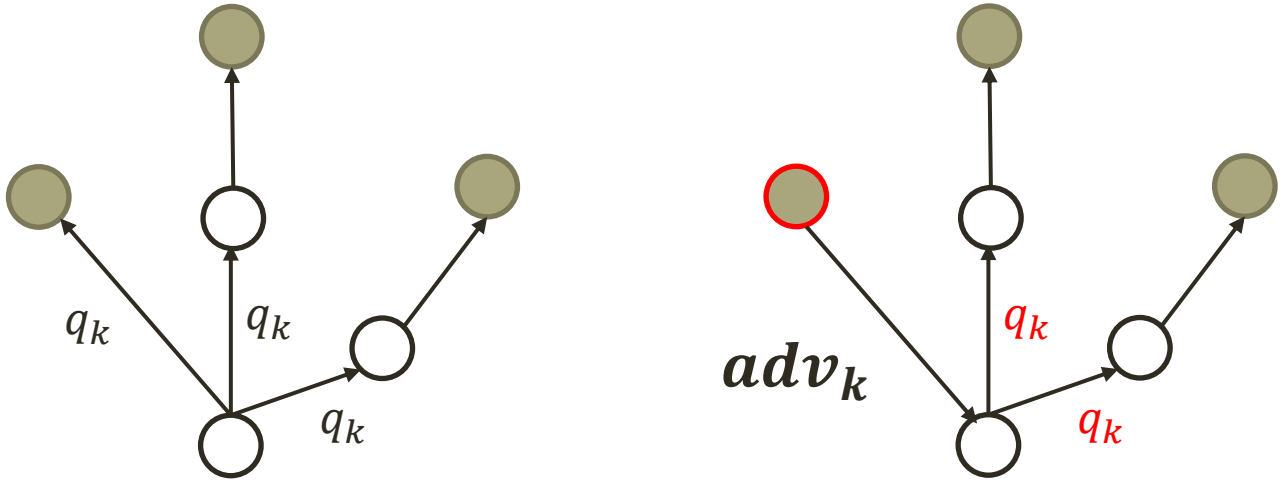
- AODV popular routing protocol for MANET
  - RREQ
  - RRESP
- AODV messages extended with
  - SREQ: service request
  - SREP: service reply



# Query Termination

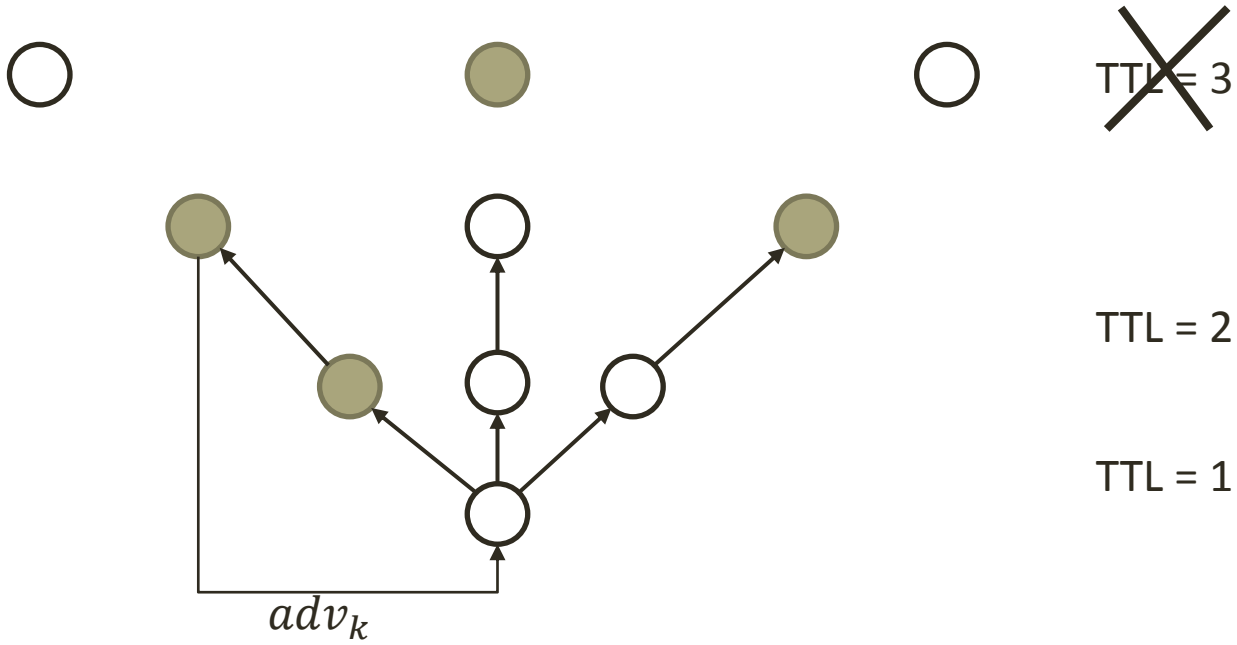
- The packet propagation injects a number of queries into the network
  - If the response has been received, all the running queries have to be terminated by avoiding:
    - network overhead
    - computation of intermediated and target nodes
  - If the response has not been already received nothing is done

# Query Termination



# Query Termination

*Iterative deepening*



# Query Termination

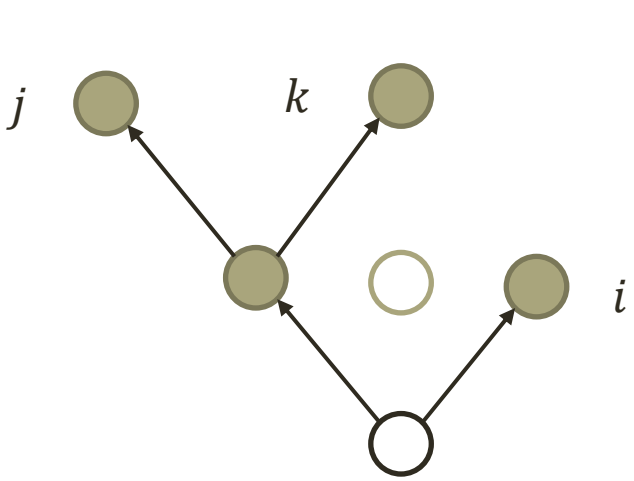
*Iterative deepening*

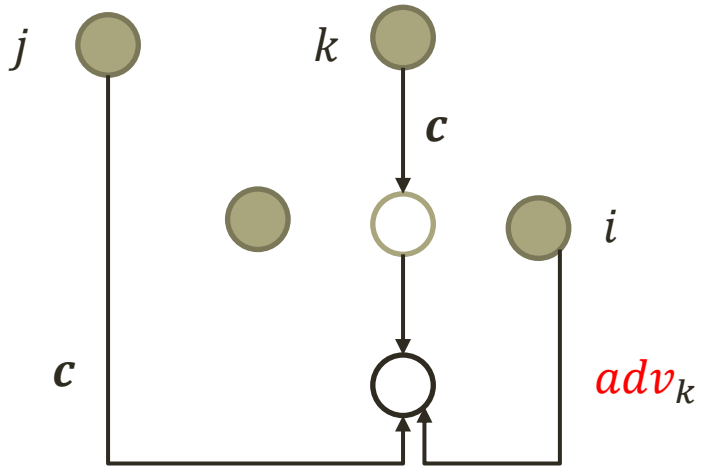
- A number of successive floods span the network by carrying the query
- TTL value of the flood increases
  - If the resource is discovered the flood iterations are suspended
  - Otherwise a new flood with TTL increased is sent
- The search diameter grows with the TTL value
  - Performs well for popular resources (hosted by lot of provider)
  - Not suitable for rare resources since big and useless area of the network are explored

# Query Termination

*Checking*

TTL = 2



# Query Termination

*Checking*

- The query replicas are sent to a number of neighbours randomly selected
- The query is forwarded toward the destination until the TTL >0
- When TTL = 0, the intermediate node sends a checking packet to the source
  - If the query is not answered the propagation continues with TTL renewed
  - Otherwise the query is terminated

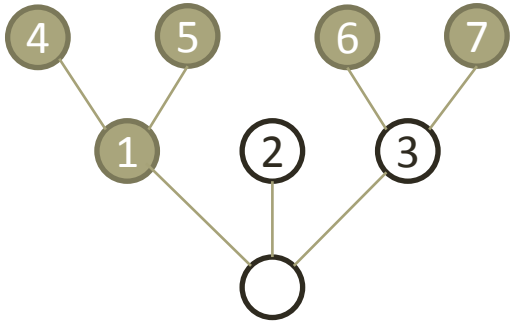
# Index

# 4. Resource Discovery Methods

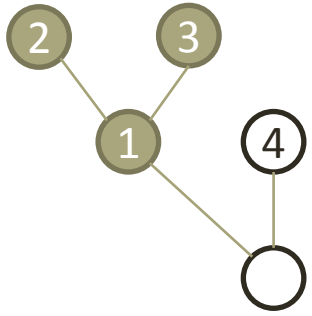
- **Uninformed methods**
  - The search is approached without **any** information abut the network
    - Systematic: the search follows a predefined approach
    - Random: the search is based on a random variable
- **Informed methods**
  - The search is approached by exploiting partial information on the network
  - Extensive use of heuristics that can lead the query to a node providing the resource

# Uninformed systematic

*BFS*




*DFS and variants*



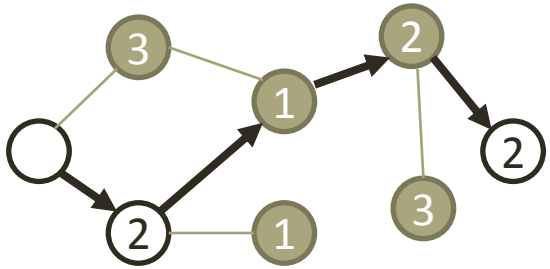

- **Limited DFS** allows to pre-set the depth of the search
- **Iterative DFS** exploits the limited DFS

At each run the depth value is increased by visiting providers far from the root

# Uninformed systematic

*Uniform-cost*

- Select the neighbours with lowest path cost
  - the search starts by exploring all the neighbours and selecting the one with lowest path cost



# Uninformed random

*Random walk*

- The source node sends a query to a number (pre-defined) of neighbors randomly selected
- The number of query replicas does not increase with the hop distance

*Probabilistic forwarding*

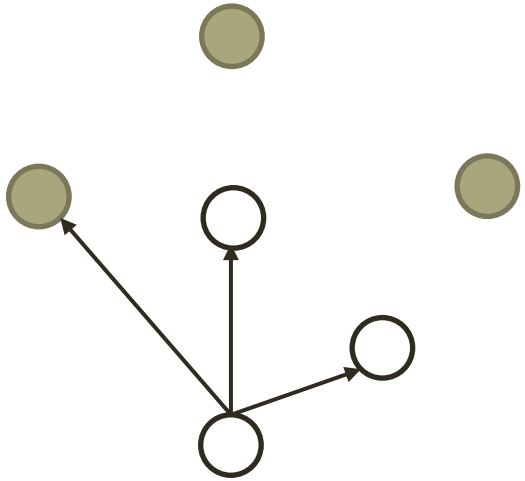
- For every node belonging to the set of out-neighbours the value $p$ = rand(0,1)

$p \geq t \rightarrow$ query forwarded to the node

# Uninformed random

*Probabilistic flooding*

- Based on flooding
- The query replicas are forwarded to the node's neighbors with a percentage p
  - *p* = 1 is the standard flooding
  - *p= 0* the propagation is stopped



## 4. Resource Discovery Methods

# Informed

- The informed search methods rely on some kinds of local information:
  - The location of the nodes
  - The traffic load
  - Available computational resources
  - Communication channel quality
  - Available bandwidth
  - Feedback considered as the percentage of success in providing a resource previously search with a query

# Informed

*Best-first*

- The methods evaluate a function $f$ on all the neighbours and selects the node with the best value of $f$.

*A**

- Based on the best-first with the function $f$ :

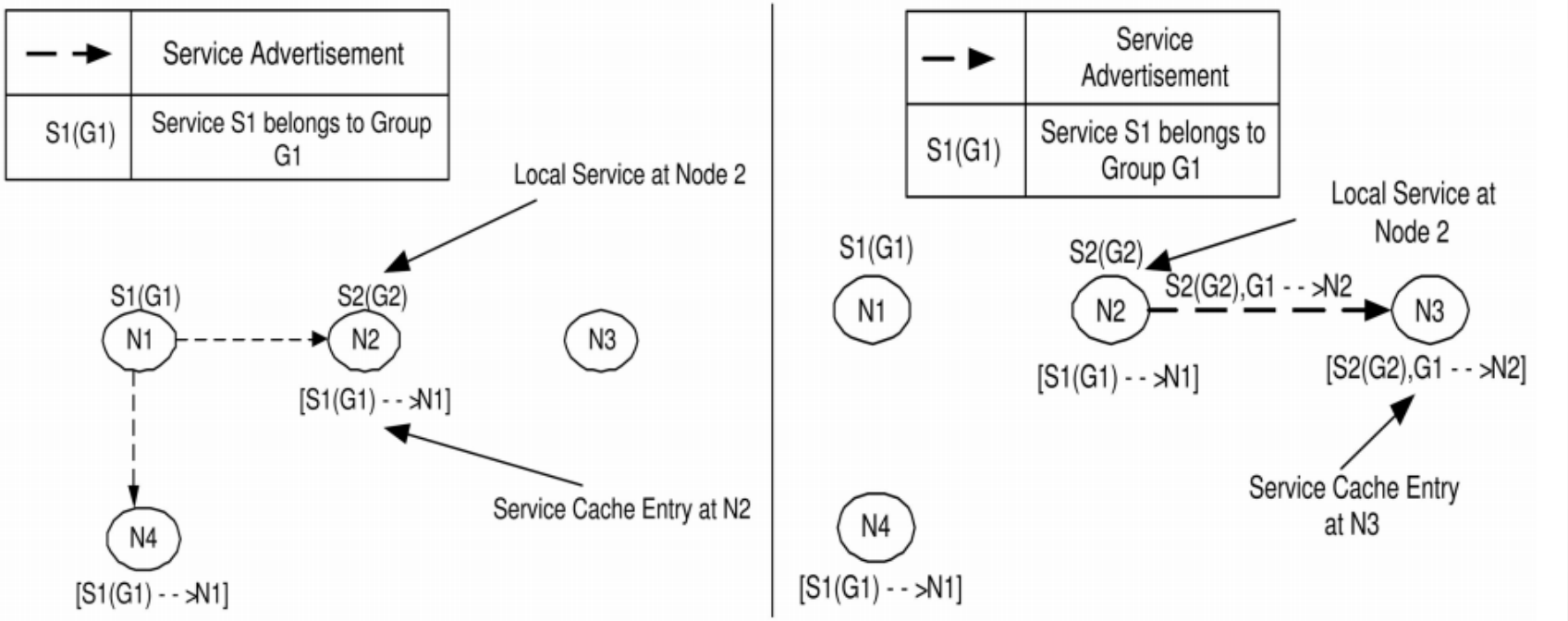
$$f(n) = g(n) + h(n)$$

$g(n)$ = path-cost from the client to the mid-node

$h(n)$ = estimated path-cost from mid-node to the provider

# Informed - GSD

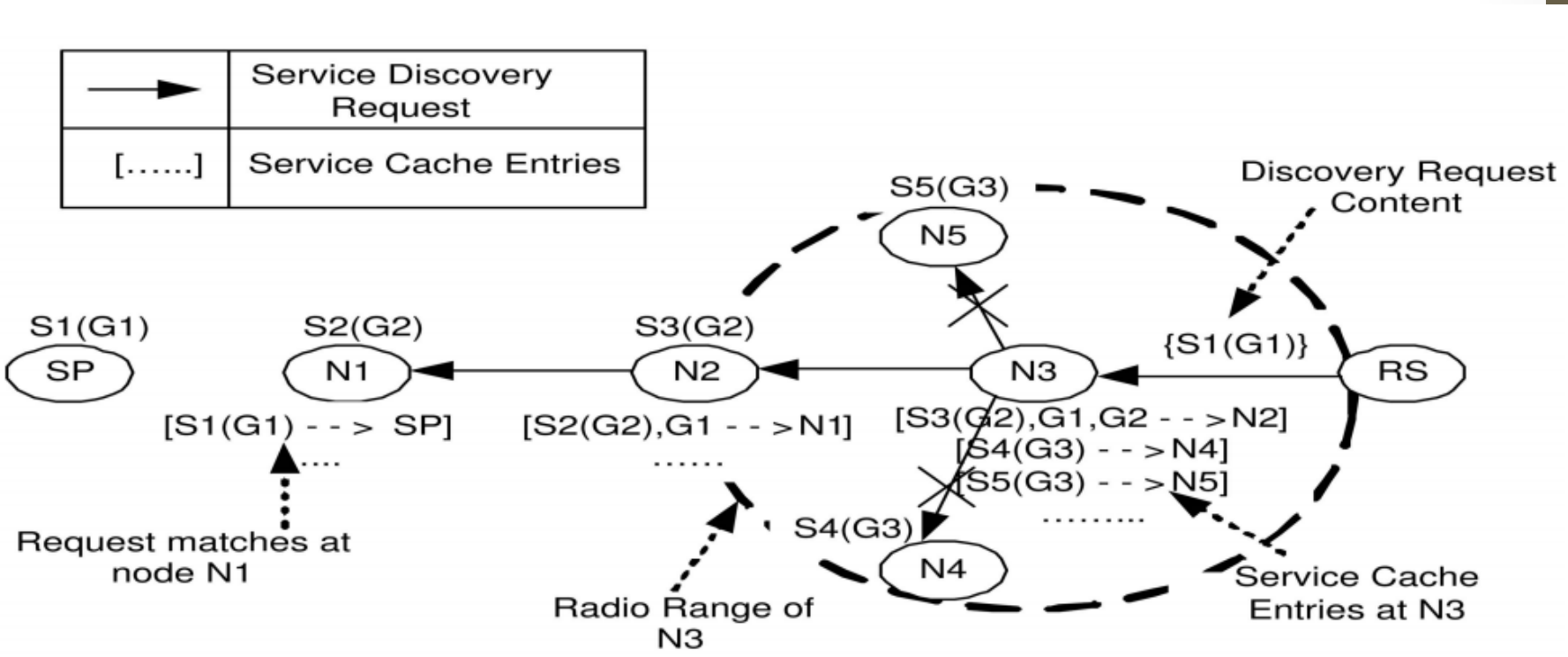
An example **G**roup-based **S**ervice **D**iscovery



## 4. Resource Discovery Methods

# Informed - GSD

An example **G**roup-based **S**ervice **D**iscovery



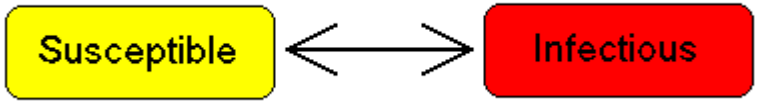
# Uninformed - Gossiping

- Mimic the spread of contagious diseases
- Inspired by some well-known virus diffusion models:

Examples [3]

**SIR**


**SIS**



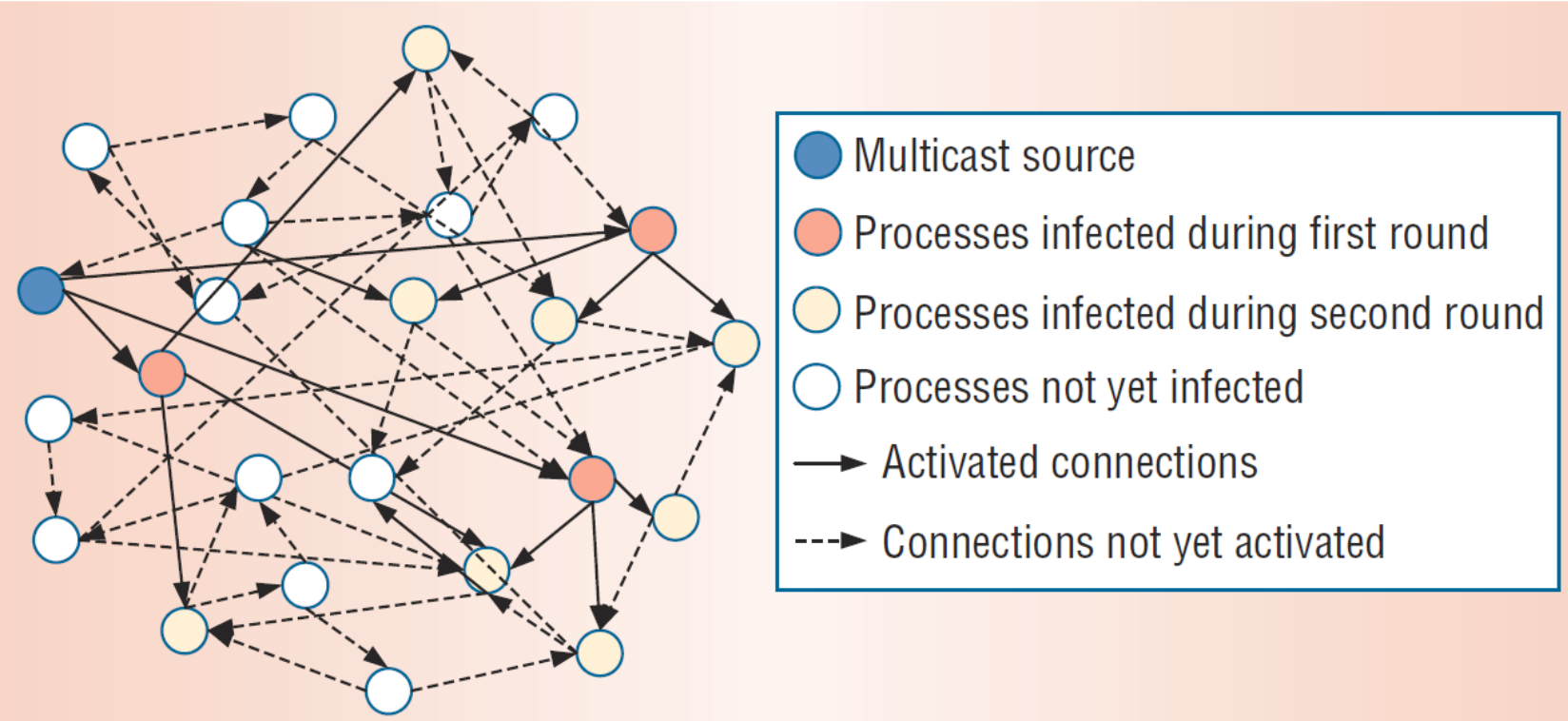
# Uninformed - Gossiping

## Example of dissemination strategy:

```
propagate_message(Message msg){
     members= getView();
     dest = fanout(members);
     send(msg, dest);
}

onReception(Message msg){
     put(msg, queue);
     propagate_message(queue.head());
}
```

# Uninformed - Gossiping

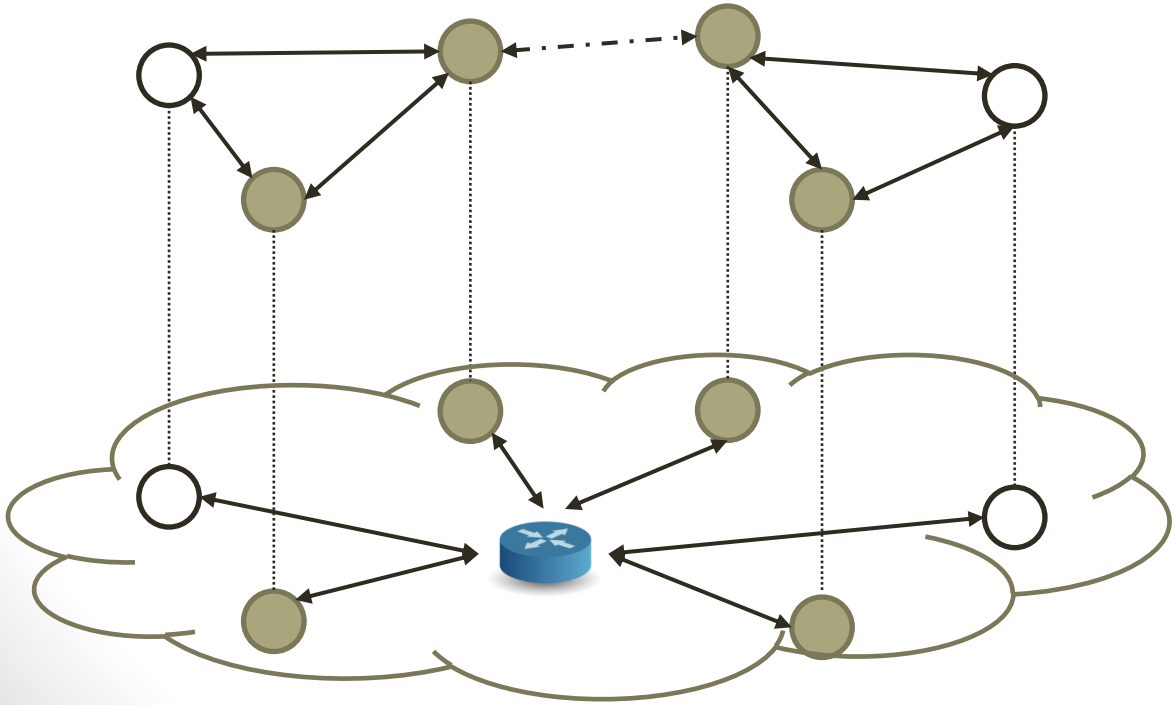



Taken from [1]

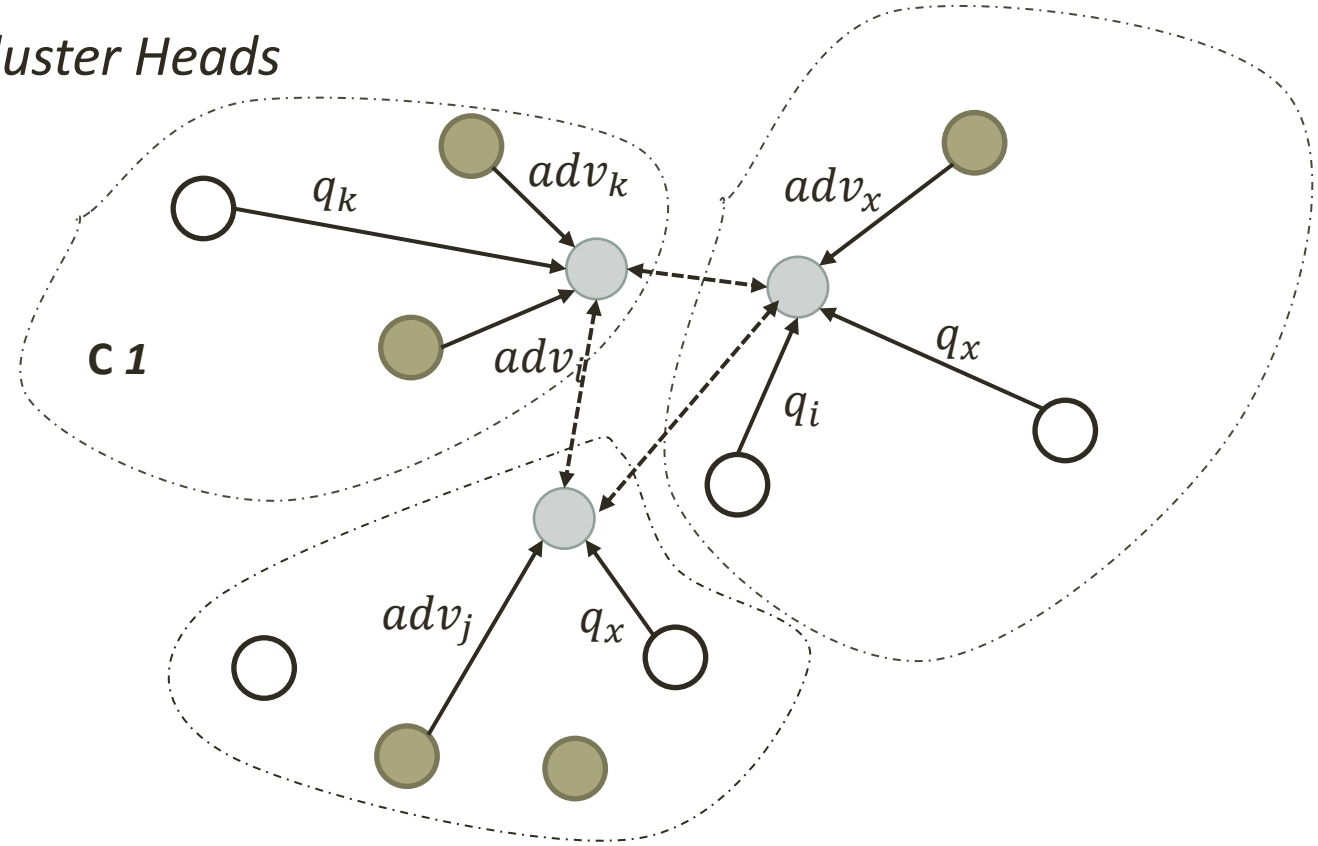
# Index

# 5. Clustering and overlay networks

- Modify the network topology in order to increase the system performance
  - Nodes are organized into clusters sharing common properties
  - Queries and service advertisements are managed in an efficient way by the cluster



# Examples of clusters




$adv_k$ = advertisement of resource k
$q_k$ = query for resource k

# Examples of clusters

Examples of cluster rules:

- Similarity on the services
  - Similar services within the same cluster
- Location of the nodes
  - Nearby nodes within the same cluster
- Quality of the service
  - Services with similar QoS parameters within the same cluster

# Index

# 6. Service Discovery frameworks

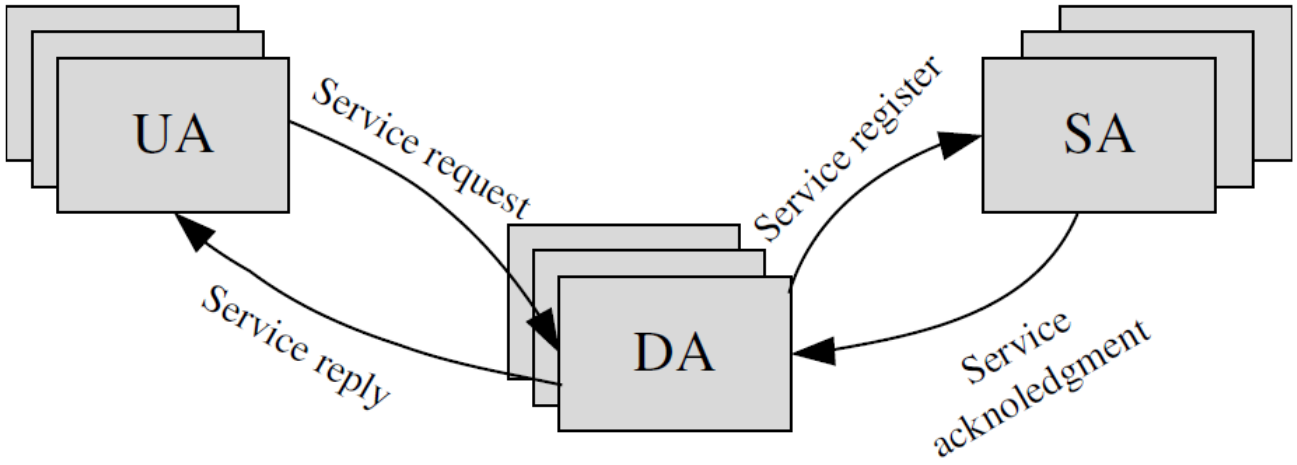
- Review of widely used service discovery frameworks
- Designed for administrated networks (hence not for p2p)
- Centralized and decentralized architectures

1. SLP Service Location Protocol
2. UPnP Universal Plug and Play
3. Bluetooth Service Discovery
4. Bonjour

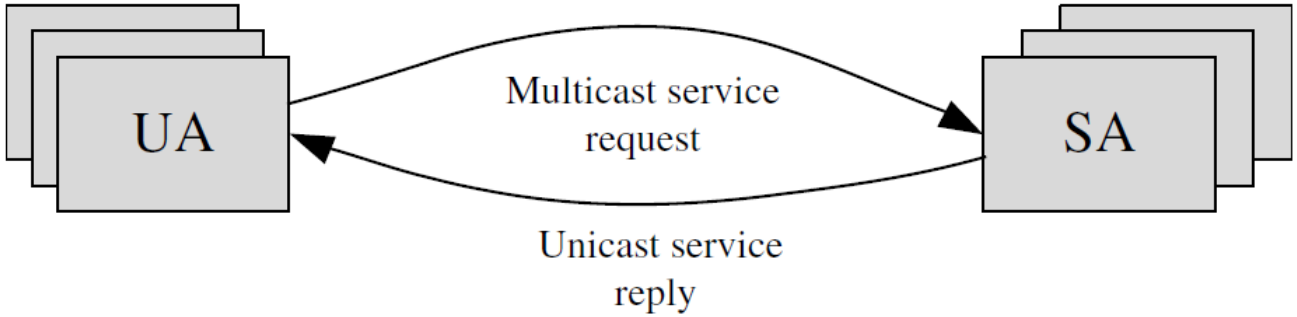
# SLP

- SLP is an IETF standard
- Defined by a number of RFCs (2165, 2608, 2609 and 2914)
- SLP relies on a centralized architecture suitable for
  - Large-Enterprise networks
  - LAN
- Supports 2 modes:
  - Centralized mode with Directory Agents (DAs)
  - Distributed mode without DAs

# SLP




A. Centralized approach. SLP with device agents.



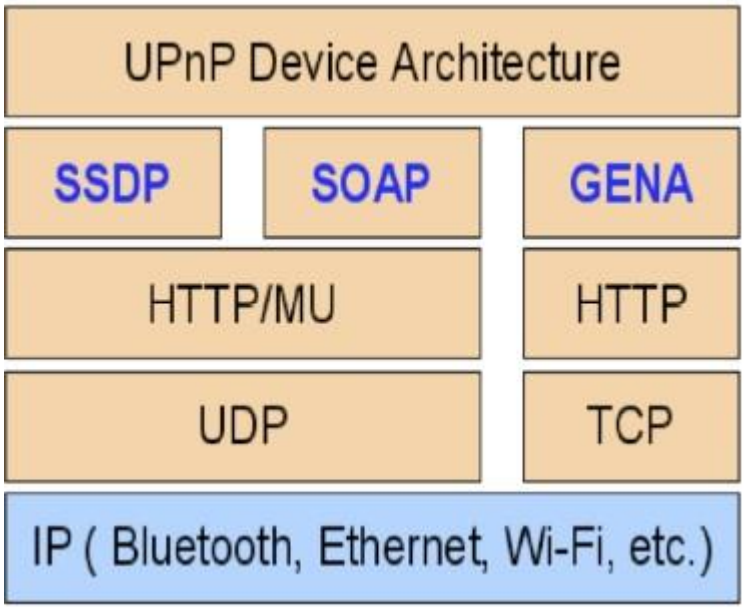

B. Distributed approach. SLP without device agents.

# SLP

- The service replies contain:
  - URL **service:servicename:protocolname://hostname**.
  - Attributes: <key, value>
  - Scope: string classifying the services
- UAs query the DA or SAs by specifying:
  - The type of the service
  - A list of attributes
  - The service scopes

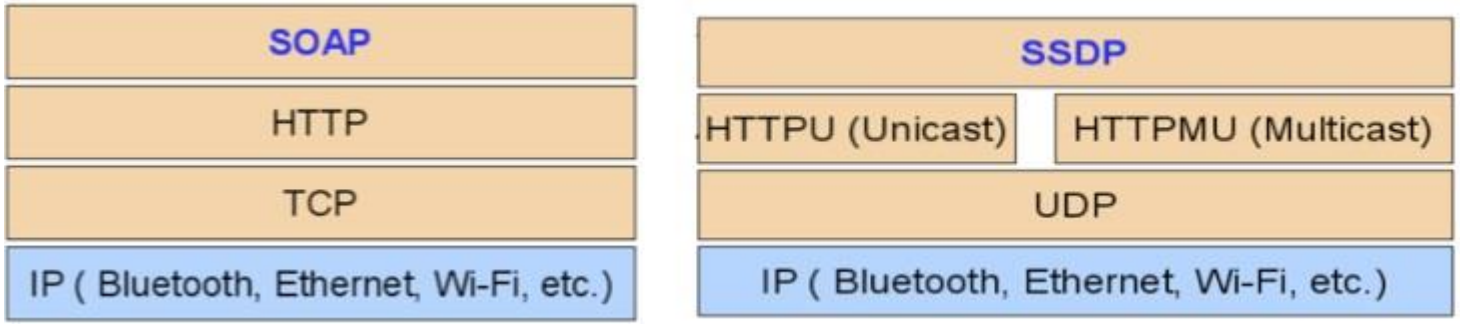
# UPnP

- Universal Plug and Play FW defines a protocol stack for:
  - Addressing
  - **Discovery**
  - Description
  - Control
  - Eventing
  - Presentation



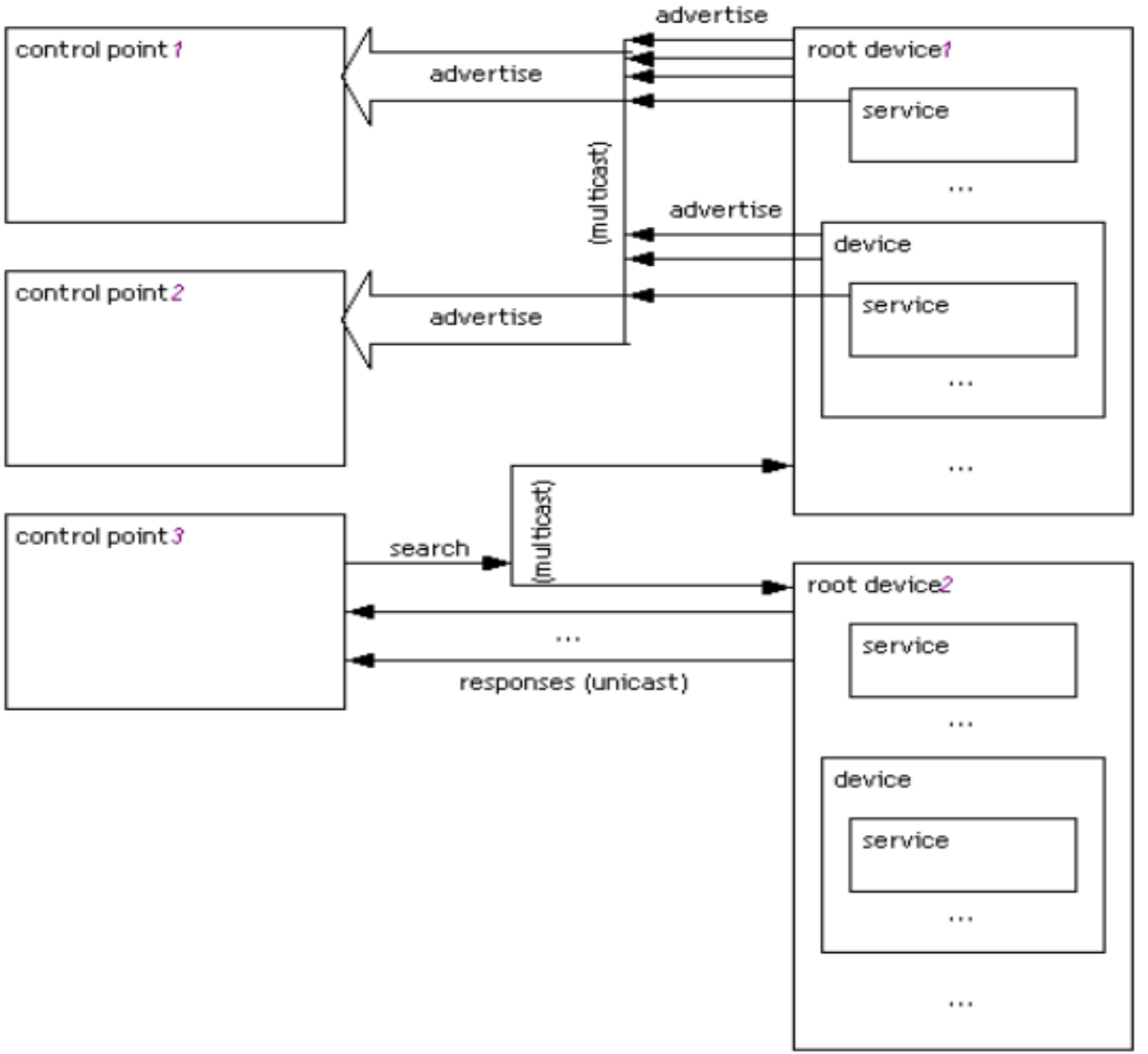
# UPnP

- UPnP relies on the SSDP protocol for the discovery




- Completely distributed query-based
- Roles of nodes:
  - Control Points ($\cong resource\ clients$)
  - Controlled Devices ($\cong resource\ providers$)

# UPnP

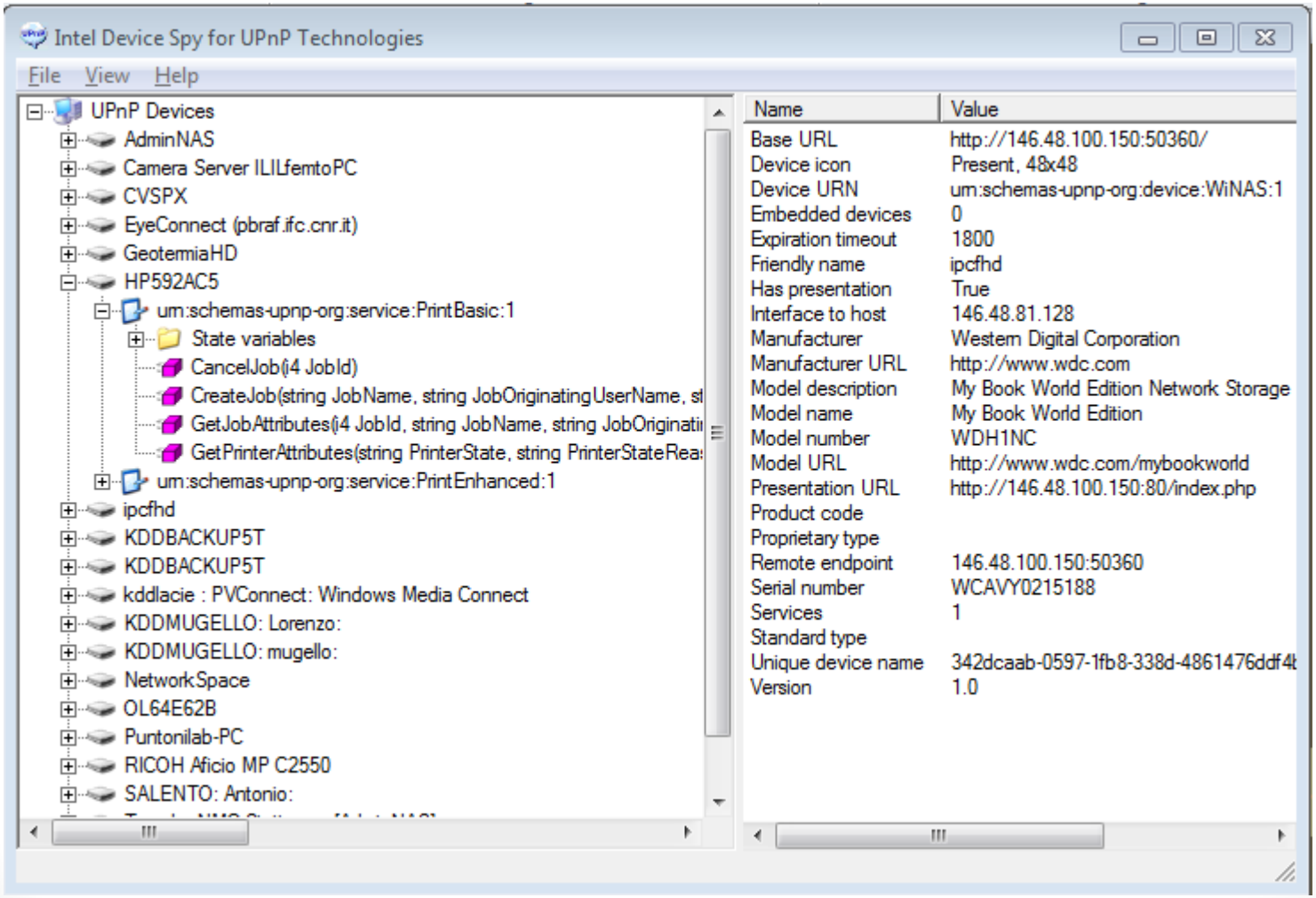

control point 1
control point 2
control point 3
advertise
advertise
(multicast)
advertise
advertise
root device1
service
...
device
service
...
...
search
(multicast)
root device2
service
...
device
service
...
...
...
responses (unicast)

# UPnP

- Controlled Devices receive an XML URL describing the Controlled Device
  - Every controlled device runs a HTTP server
  - XML document provides a tree-based description of the device

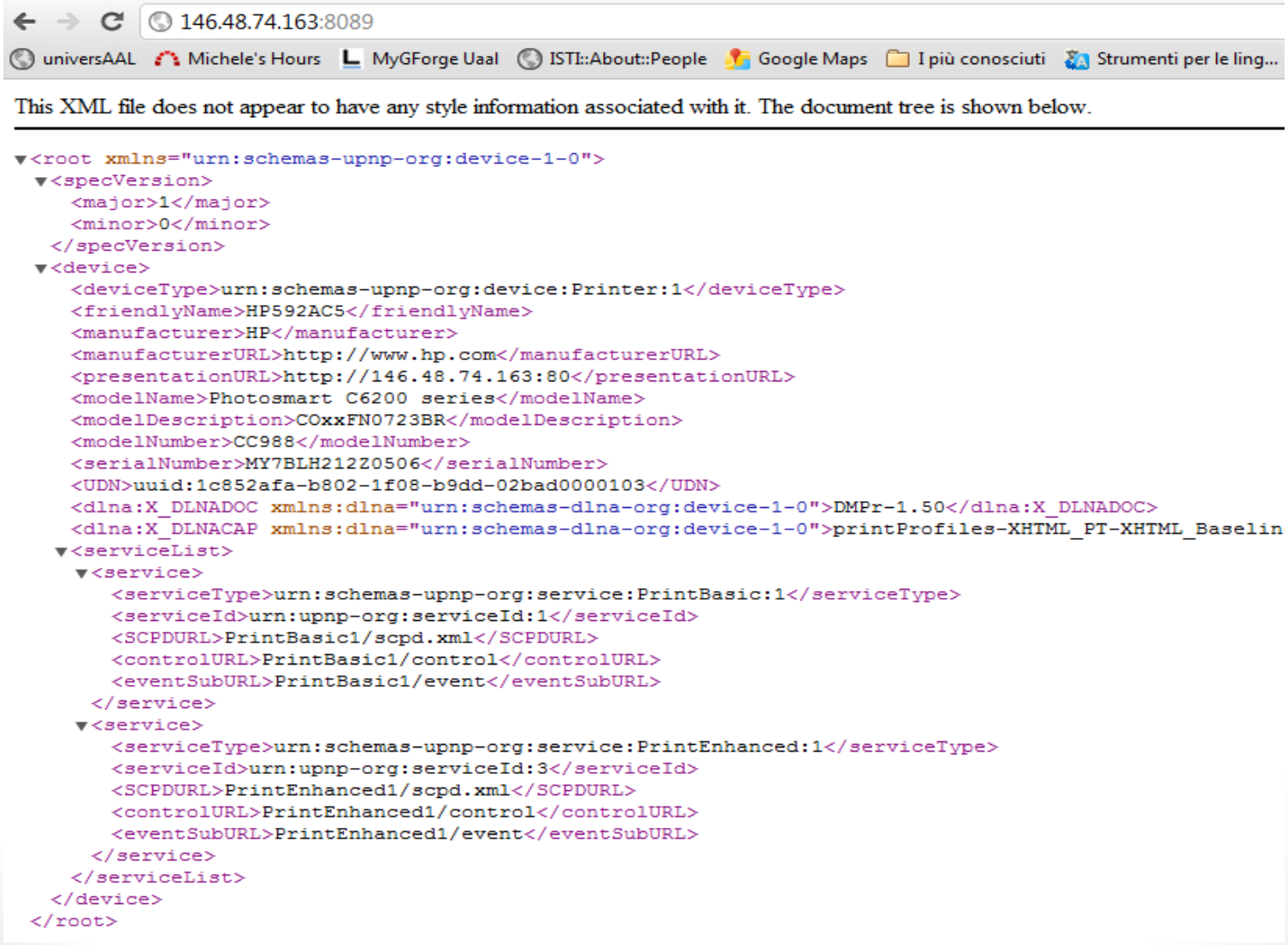
- UPnP also defines:
  - how to access to the service
    - To invoke remote procedures through SOAP messages
  - how to be notified by the service
    - To register to the control variables and to receive asynchronous HTTP messages

# UPnP


Intel Device Spy for UPnP Technologies
File View Help
UPnP Devices
AdminNAS
Camera Server ILILfemtoPC
CVSPX
EyeConnect (pbraf.ifc.cnr.it)
GeotermiaHD
HP592AC5
um:schemas-upnp-org:service:PrintBasic:1
State variables
CancelJob(i4 JobId)
CreateJob(string JobName, string JobOriginatingUserName, st
GetJobAttributes(i4 JobId, string JobName, string JobOriginati
GetPrinterAttributes(string PrinterState, string PrinterStateRea
um:schemas-upnp-org:service:PrintEnhanced:1
ipcfhd
KDDBACKUP5T
KDDBACKUP5T
kddlacie : PVConnect: Windows Media Connect
KDDMUGELLO: Lorenzo:
KDDMUGELLO: mugello:
NetworkSpace
OL64E62B
Puntonilab-PC
RICOH Aficio MP C2550
SALENTO: Antonio:
Name
Value
Base URL
http://146.48.100.150:50360/
Device icon
Present, 48x48
Device URN
urn:schemas-upnp-org:device:WiNAS:1
Embedded devices
0
Expiration timeout
1800
Friendly name
ipcfhd
Has presentation
True
Interface to host
146.48.81.128
Manufacturer
Western Digital Corporation
Manufacturer URL
http://www.wdc.com
Model description
My Book World Edition Network Storage
Model name
My Book World Edition
Model number
WDH1NC
Model URL
http://www.wdc.com/mybookworld
Presentation URL
http://146.48.100.150:80/index.php
Product code
Proprietary type
Remote endpoint
146.48.100.150:50360
Serial number
WCAVY0215188
Services
1
Standard type
Unique device name
342dcaab-0597-1fb8-338d-4861476ddf4
Version
1.0

# 6. Service Discovery Frameworks

# UPnP



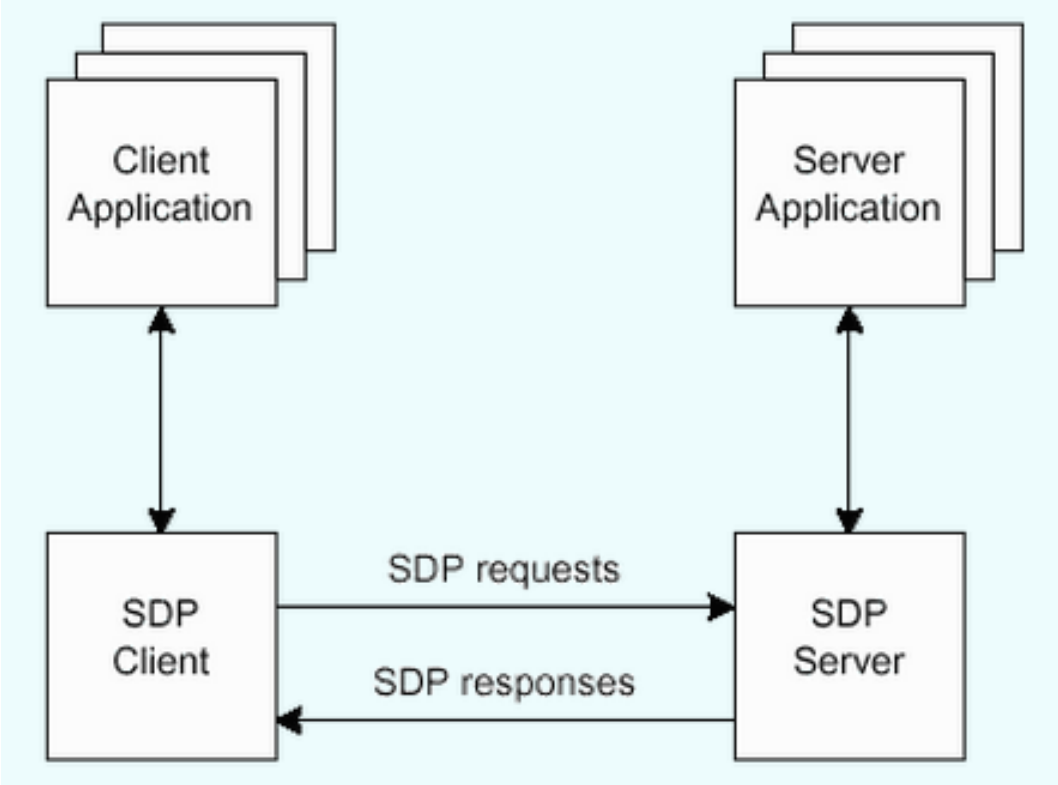
# Bluetooth Service Discovery

- Bluetooth allows multiple devices to cooperate in a master-slave relationship
  - a Piconet composed of
    - 1 master device
    - n slaves
- Designed for resource-constrained environments and to spend minimal bandwidth
- Bluetooth  is not designed for IP-based networks
- Service Discovery in Bluetooth is powered by SDP

# Bluetooth Service Discovery

- Each device can act as SDP client or server
  - Client discovers services provided by other devices
  - Service provides services
- Every service is described by a *service record (*set of service attributes )
- Every service belongs to a *service class:*
  - Type of the service
  - Set of attributes describing the specific service
- Services and attributes are uniquely identified with pre-defined IDs

# Bluetooth Service Discovery



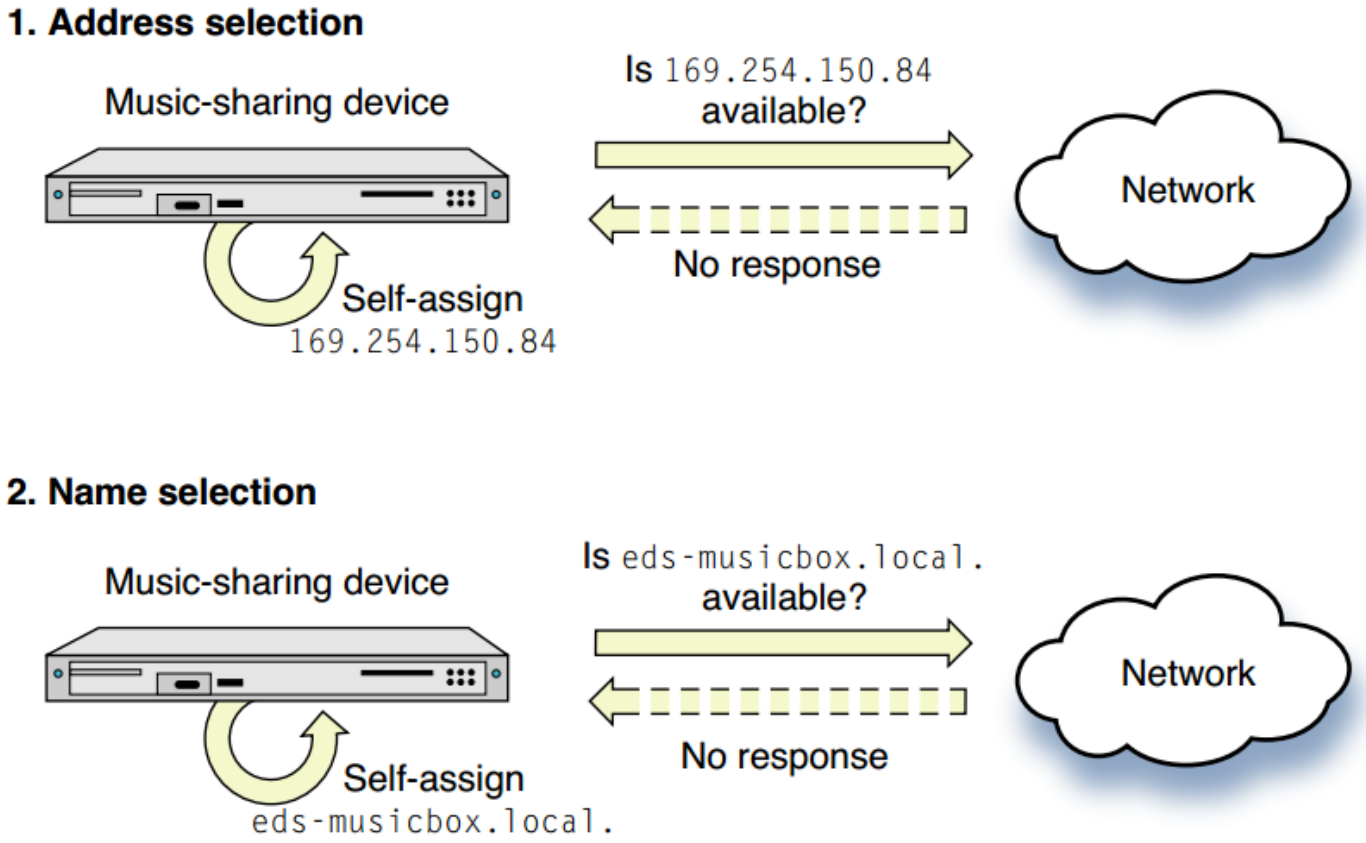
# Bluetooth Service Discovery

- SDP defines 3 search modes
  - Service Search: to search for a specific service identified by an ID. The client will receive a bunch of service records
  - Attribute Search: to search for a set of attributes with respect to a specific service
  - Service and Attribute Search: to search for a service and to fetch a list of relevant attributes

# Bonjour

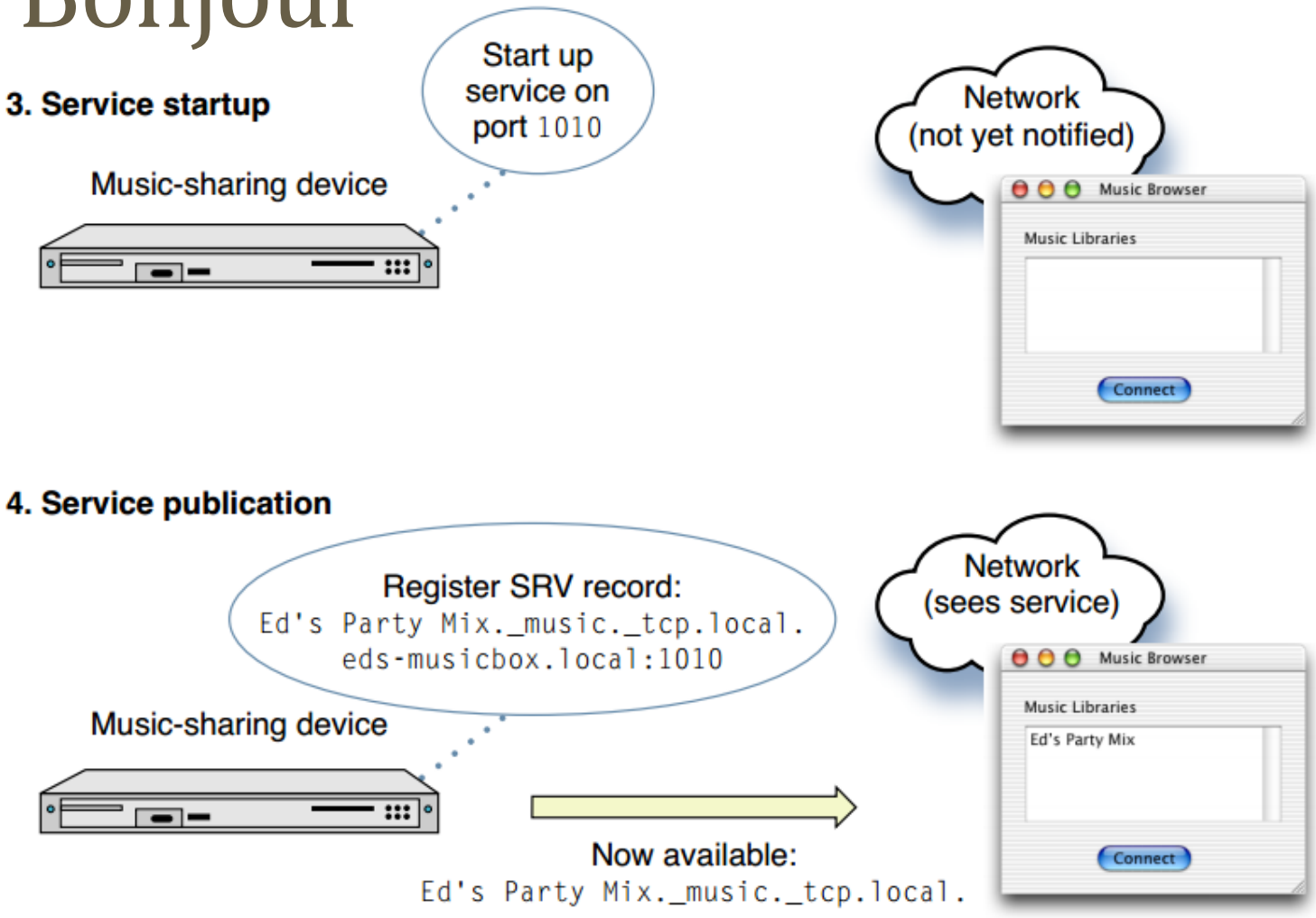
- Bonjour protocol is promoted and supported by Apple
  - Successor of Appletalk
  - Implementation of Zeroconf IETF protocol
- Bonjour is designed for local and ad-hoc IP-based networks
  - Decentralized architecture
  - Relies on multicast and DNS technologies
- Bonjour covers 3 areas:
  - Addressing
  - Naming
  - Service Discovery

# Bonjour



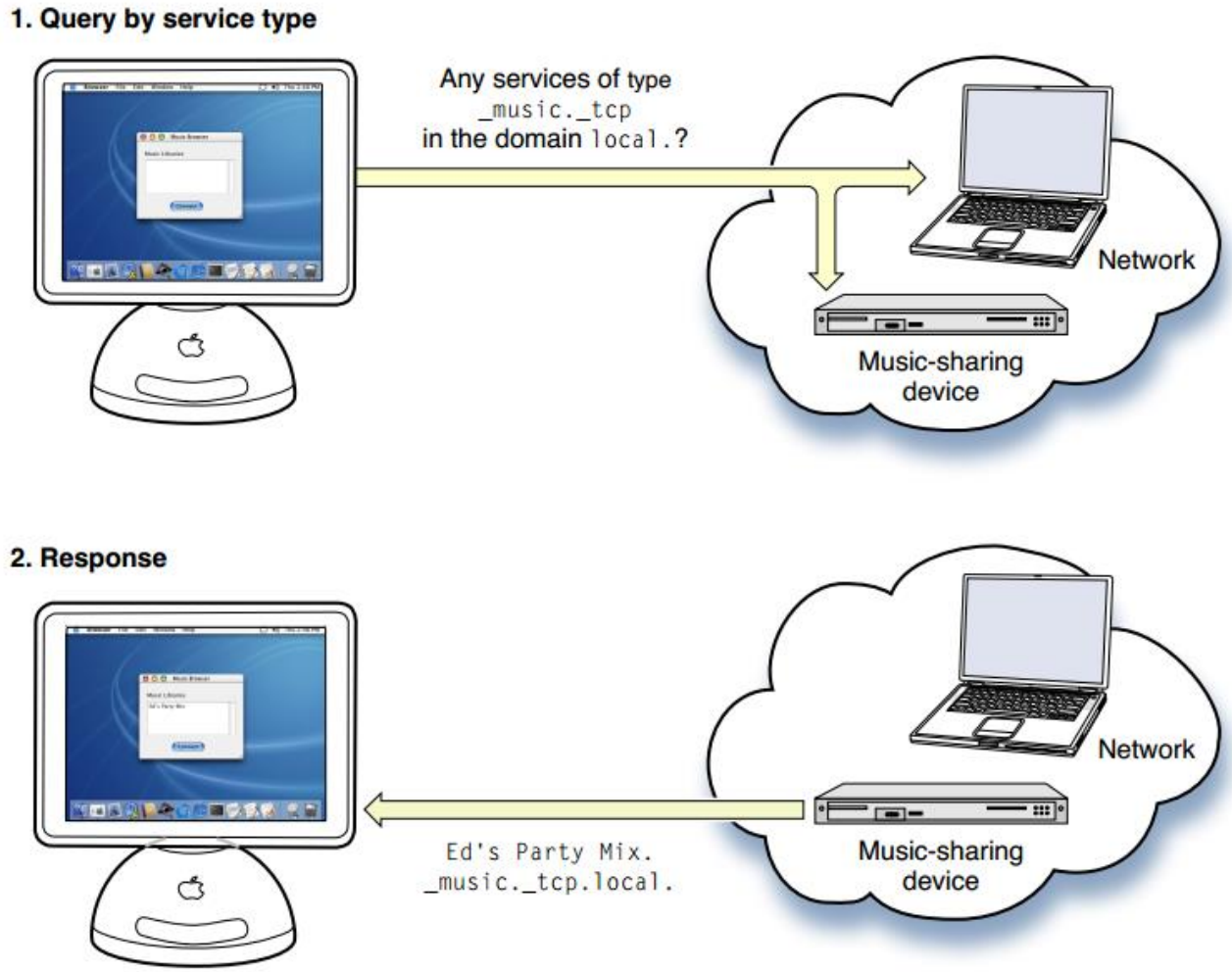

https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/NetServices/NetServices.pdf

# Bonjour




https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/NetServices/NetServices.pdf

# Bonjour




https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/NetServices/NetServices.pdf

# Index

# Challenges

*Wireless Sensor Networks*

- Decentralized infrastructure
- The size of the network spans over a wide range
- Communication media with low bandwidth
- Devices with limited memory
  - No XML parsing allowed
  - Cost-based algorithm for efficient service discovery

*PSN*

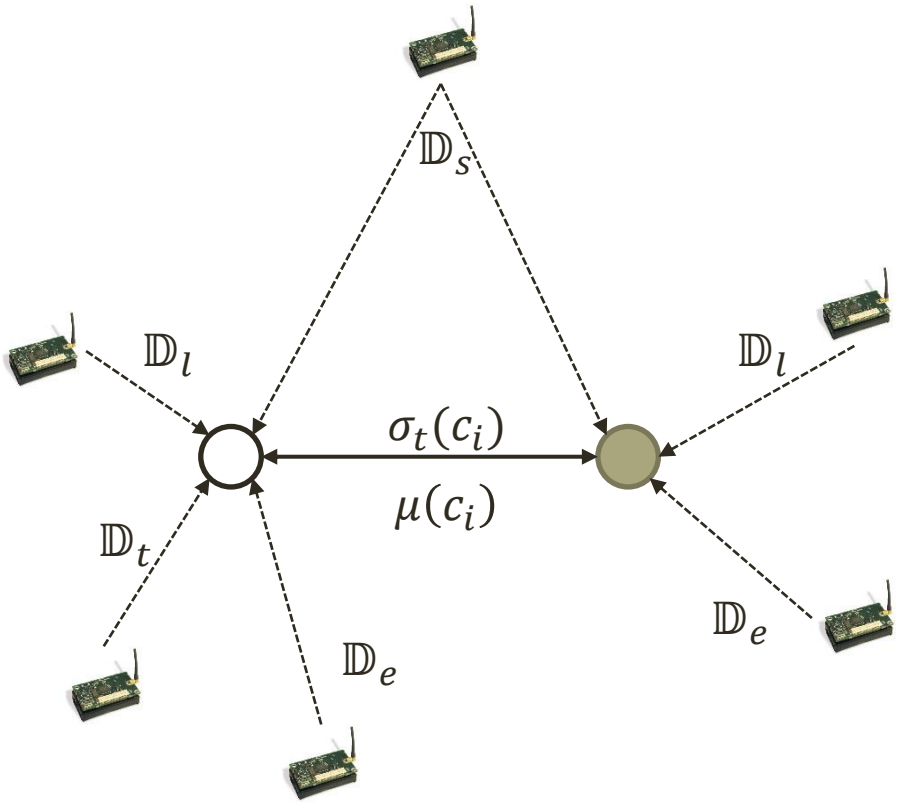
- Exploit user mobility to propagate advertisements and queries
- Reduce battery usage of devices

# Challenges

*Exploit Context-Information*

- To exploit the context-information to refine the search
- To push to the clients all the **needed** services instead of the **discovered** ones
  - $\mathbb{D} = context$
  - $C = \{\mathbb{D}_0, \dots, \mathbb{D}_{n-1}\}, \mathbb{D}_i \in \mathbb{D} = i - th\ dimension\ of\ the\ context$
    - $\mathbb{D}_{location}, \mathbb{D}_{time}, \mathbb{D}_{environment}, \mathbb{D}_{physical\ status}, \mathbb{D}_{device\ status}$
  - $\sigma_t(c_i) = v \in \mathbb{R}^n, \qquad v = (d_0, .., d_{n-1})$
  - $\mu(c_i) = s, \qquad s = \{adv_i, \dots, adv_k\}_{\sigma_t(c_i)}$

# Challenges


$\mathbb{D}_s$
$\mathbb{D}_l$
$\mathbb{D}_l$
$\sigma_t(c_i)$
$\mu(c_i)$
$\mathbb{D}_t$
$\mathbb{D}_e$
$\mathbb{D}_e$

# Contacts

Michele Giroami

- Wireless Network Lab. ISTI- CNR
- Departement of Computer Science Pisa

- michele.girolami@isti.cnr.it
- girolami@di.unipi.it

# Bibliography

*Surveys*

- Ververidis, C.N.; Polyzos, G.C.; , "Service discovery for mobile Ad Hoc networks: a survey of issues and techniques," *Communications Surveys & Tutorials, IEEE* , vol.10, no.3, pp.30-45, Third Quarter 2008
- Elena Meshkova, Janne Riihijärvi, Marina Petrova, Petri Mähönen, ” A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks” Computer Networks 52 (2008) 2097–2128
- Mian, A.N.; Baldoni, R.; Beraldi, R.; , "A Survey of Service Discovery Protocols in Multihop Mobile Ad Hoc Networks," Pervasive Computing, IEEE , vol.8, no.1, pp.66-74, Jan.-March 2009
- Koen Vanthournout , Geert Deconinck , Ronnie Belmans, “A taxonomy for resource discovery”, Personal and Ubiquitous Computing Journal
- Wenge Rong, Kecheng Liu, "A Survey of Context Aware Web Service Discovery: From User's Perspective," sose, pp.15-22, 2010 Fifth IEEE International Symposium on Service Oriented System Engineering, 2010

# Bibliography

*Resource Discovery Surveys*

- Zhu, F.; Mutka, M.W.; Ni, L.M.; , "Service discovery in pervasive computing environments," Pervasive Computing, IEEE , vol.4, no.4, pp. 81- 90, Oct.-Dec. 2005
- Sivavakeesar, S.; Gonzalez, O.F.; Pavlou, G.; , "Service discovery strategies in ubiquitous communication environments," Communications Magazine, IEEE , vol.44, no.9, pp.106-113, Sept. 2006
- W. Keith Edwards, "Discovery Systems in Ubiquitous Computing," IEEE Pervasive Computing, vol. 5, no. 2, pp. 70-77, April-June 2006

# Bibliography


*Semantic Service Discovery*

- S Mokhtar, D Preuveneers, N Georgantas, V Issarny, Y Berbers, “EASY: Efficient semAntic Service discoverY in pervasive computing environments with QoS and context support”, Journal of Systems and Software (2007) Volume: 81, Issue: 5, Publisher: Elsevier, Pages: 785-808
- Fei Li , Katharina Rasch , Hong-linh Truong , Rassul Ayani , Schahram Dustdar, ” Proactive Service Discovery in Pervasive Environments“
- Helal, S.; Desai, N.; Verma, V.; Choonhwa Lee; , "Konark - a service discovery and delivery protocol for ad-hoc networks," Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE , vol.3, no.
- Klein, M.; Konig-Ries, B.; Obreiter, P.; , "Service rings - a semantic overlay for service discovery in ad hoc networks," Database and Expert Systems Applications, 2003. Proceedings. 14th International Workshop on , vol., no., pp. 180- 185, 1-5 Sept. 2003
- Juan Ignacio Vázquez, and Diego López de Ipiña, “mRDP: An HTTP-based lightweight semantic discovery protocol.”, Computer Networks, Vol. 51, Nr. 16 (2007) , p. 4529-4542.
- Bellavista, P.; Corradi, A.; Montanari, R.; Toninelli, A.; , "Context-aware semantic discovery for next generation mobile systems," Communications Magazine, IEEE , vol.44, no.9, pp.62-71, Sept. 2006
- Chakraborty, D.; Joshi, A.; Yesha, Y.; Finin, T.; , "Toward Distributed service discovery in pervasive computing environments," Mobile Computing, IEEE Transactions on , vol.5, no.2, pp. 97- 112, Feb. 2006

# Bibliography


Cross-layer protocols

- Rae Harbird, “Adaptive Resource Discovery for Ubiquitous Computing”, In Proceedings of the 2 nd Workshop on Middleware for pervasive and ad-hoc computing
- Ververidis, C.N.; Polyzos, G.C.; , "AVERT: Adaptive SerVicE and Route Discovery ProTocol for MANETs," Networking and Communications, 2008. WIMOB '08. IEEE International Conference on Wireless and Mobile Computing, , vol., no., pp.38-43, 12-14 Oct. 2008

*Service Discovery  Architectures*

- Dhanakoti, N.; Gopalan, S.; Sridhar, V.; Subramani, S.; , "A distributed service discovery and selection framework in pervasive service environments," Telecommunications, 2005. advanced industrial conference on telecommunications/service assurance with partial and intermittent resources conference/e-learning on telecommunications workshop. aict/sapir/elete 2005. proceedings , vol., no., pp. 452- 457, 17-20 July 2005
- Yuanmin Chen; Wei Mao; Xiaodong Li; , "Federation framework for service discovery in ubiquitous computing," Communication Technology, 2008. ICCT 2008. 11th IEEE International Conference on , vol., no., pp.600-602, 10-12 Nov. 2008
- ZHANG Li, SHI Zhen-lian, SHEN Qi, "A Service Discovery Architecture based on Anycast in Pervasive Computing Environments," compsac, vol. 2, pp.101-108, 2007 31st Annual International Computer Software and Applications Conference, 2007

# Bibliography

*Service Discovery Frameworks*

- SLP E. Guttman et al., “Service Location Protocol, Version 2,” IETFRFC 2608, June 1999
- UPnP: Microsoft Corporation, “Universal Plug and Play: Background”;http://www.upnp.org/resources/UPnPbkgnd.htm, 1999.
- Salutation: Salutation Consortium, “Salutation Architecture Specification”;http://web.archive.org/web/20030623193812/www.salutation.org/, 1999 (the Salutation Consortium was disbanded on June 30, 2005).
- Bluetooth: “Specification of the Bluetooth System”; http://www.bluetooth.com, Dec. 1999.

# Backup slides

# Query Termination

*Chasing wave*

- A number of query replicas are sent toward the neighbours
- For every hop the query leaves a *marker* in order to keep trace of path
- As soon as the query is received, the client sends a number of chasing packets in order to kill the running queries

# Super Node Clustering

- Given the graph, a number of nodes are elected with the role of Cluster Heads (CH)
  - CHs collect partial information about the available resources
- The queries are forwarded to the a CHs that can reply differently:
  - by forwarding the query to the provider
  - by answering on behalf of the provider

# Super Node Clustering issues

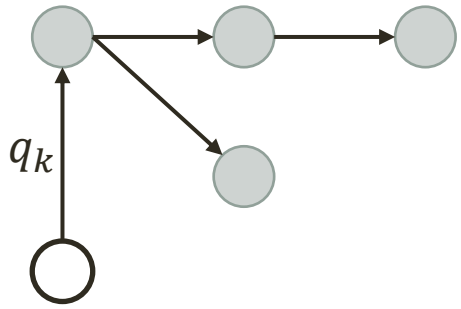
- The election of CHs require to share and to evaluate the function/non-functional properties of the candidates (resource computation, load factor, hw features)

- A node acting as CH can pass the role to another node dynamically
- The CHs generate non-negligible amount of traffic due to:
  - Synchronization of CHs about the available resource
  - Election and maintenance of the CHs

# Examples of clusters

*Tree-based clustering*

- CHs are arranged as a tree.




*Locality and Logical Clustering*

- Nodes are clustered on the basis of
  - Distance in terms of number of hops
  - Type of resource provided: all the