

Service Discovery in Pervasive Computing Environments

Presented by
Jamal Lewis and Shruti Pathak

CS 570 Computer Networks

Instructor Dr. Feng Zhu

Introduction

- What is it?

Pervasive Computing Environments integrate networked computing devices with people and their ambient environments enabling the device and the service to communicate with each other

- Simply means that even if the network/ protocols are different; people should be able to use it with minimum interaction with the service providers
- Few Example Service Discovery Protocols
 - MIT's International Naming System
 - UC's Berkeley's Ninja Service Discovery Service
 - Salutation Protocol

Pervasive Environment Challenges

- Pervasive computing environments are dynamic and heterogeneous
- Unlike the Enterprise Environments; it is difficult to define a network scope for pervasive computers and it is also difficult for all services to be managed by a system administrator
- Unlike the Web services, pervasive environments focus on interactions among people than between services

Integration with people

- This is the most serious challenge to pervasive computing discovery
- First challenge is to protect the private data of users
- Second challenge is to determine how much knowledge a user or a service must have for service discovery

People serve two roles:

1. Users (Require less knowledge)
 2. Service providers (Require special skill)
- The third challenge is to allow multiple service-providers to coexist at a single place

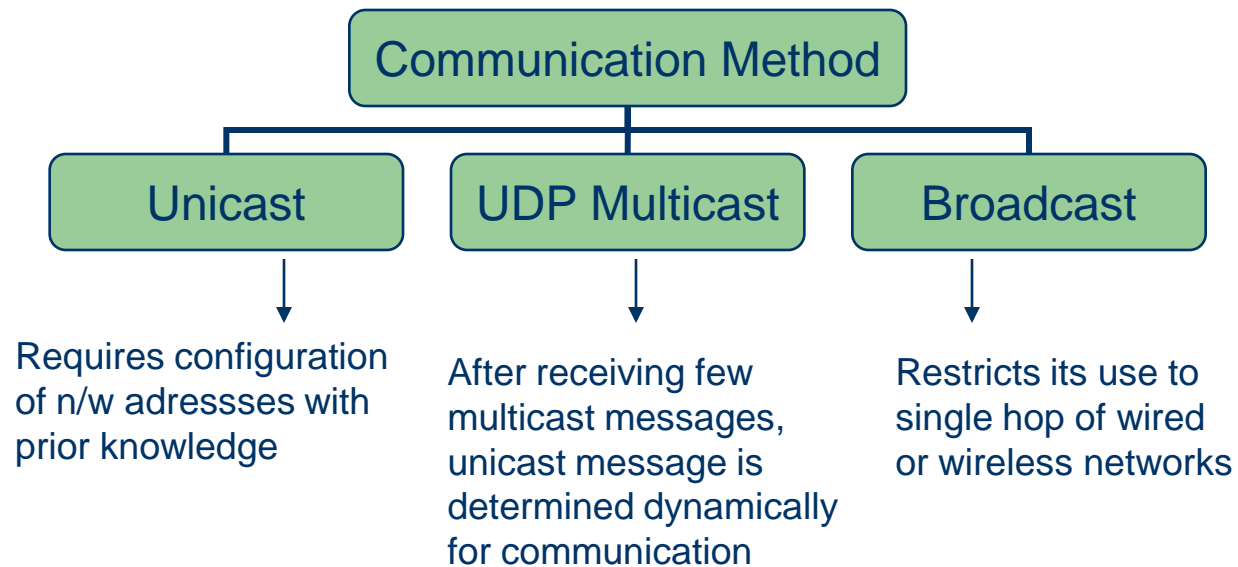
Integration with Environment

- How to define the environment that the service discovery targets?
- Pervasive Computing is heterogeneous in terms of hardware, software, network protocols and service providers
- A common protocol should be established in order to facilitate the discovery of service by the user

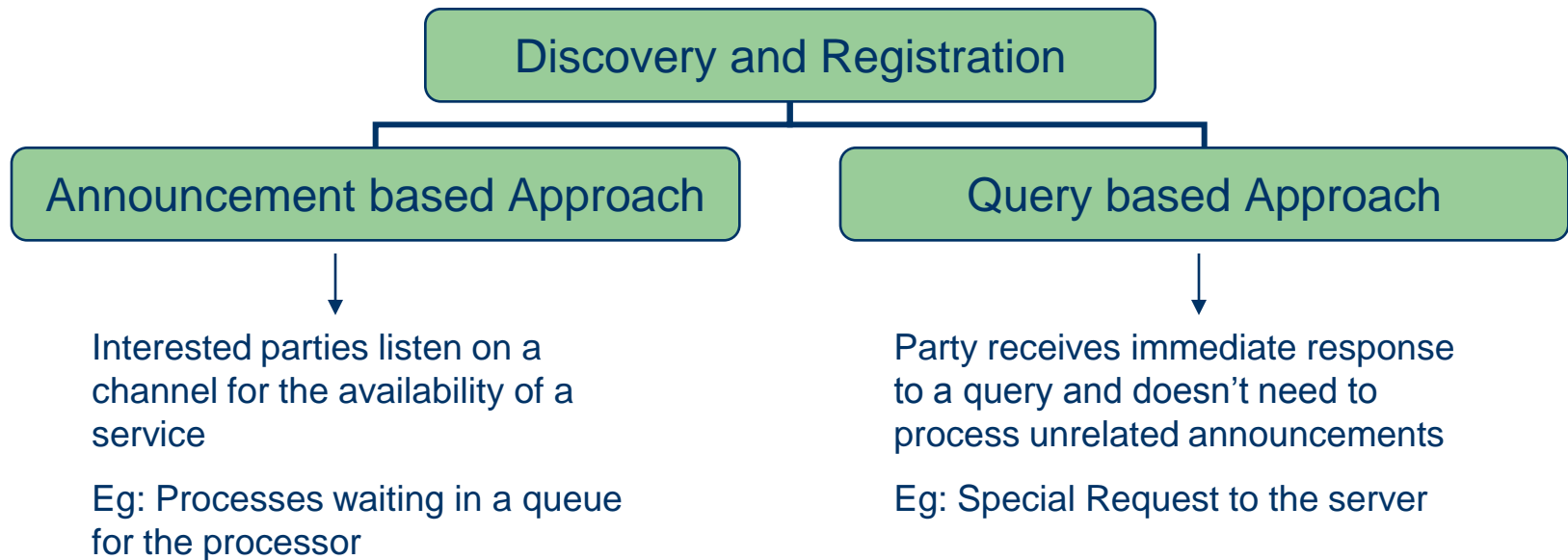
Service and Attribute Naming

- Two types of Service and attribute naming: **Template-based** and **Template-based and predefined**
- **Template -Based**
 - defines a format for service names and attributes
 - Example: Apple's Rendezvous is based on Internet's DNS which defines how service names can be composed
- **Template-Based and Predefined**
 - gives commonly used attributes and names
 - eliminates ambiguity regarding name and attributes in client, services, and directory interaction.

Initial Communication Method



Discovery and Registration



Service Discovery Infrastructure

- Uses two service discovery infrastructure models
 - **Directory Based Model**
 - Has a dedicated component that maintains service information and processes queries announcements
 - Example of Directory Based Model would be Microsoft's Active Directory
 - **Non-Directory Based Model**
 - No dedicated component
 - When a query arrives, every service processes and service that matches query responds
 - Example: Switch that broadcast a request to all systems on network in order to find where a new computer is located.

Service Information State

- Two service information states: **Soft State** and **Hard State**
 - **Soft State**
 - Most service discovery protocols maintain status as a soft state
 - Before service expiration, a client or directory polls the service or service then announces itself to renew registration lease.
 - Soft state simplifies system design and keeps service up to date.
 - **Hard State**
 - Requires fewer services and housekeeping jobs
 - Clients and services poll periodically to verify info is up to date.

Discovery Scope

- Proper discovery scopes minimize unnecessary computation on client, services, and directories.
- **Network topologies, user roles, context information**, or a combination helps to properly define service discovery scope session targets.
- Based on **Network Topology, User Roles, Context Info**, or a combination of either
 - **Network Topology**
 - Uses LAN and single hop wireless network range protocols
 - One can assume that the clients, services, and directories belong to same administrative domain
 - Setback to that is pervasive computing environments can include multiple, coexisting administrative domains as well as different underlying networks that may not be connected

Discovery Scope cont'd

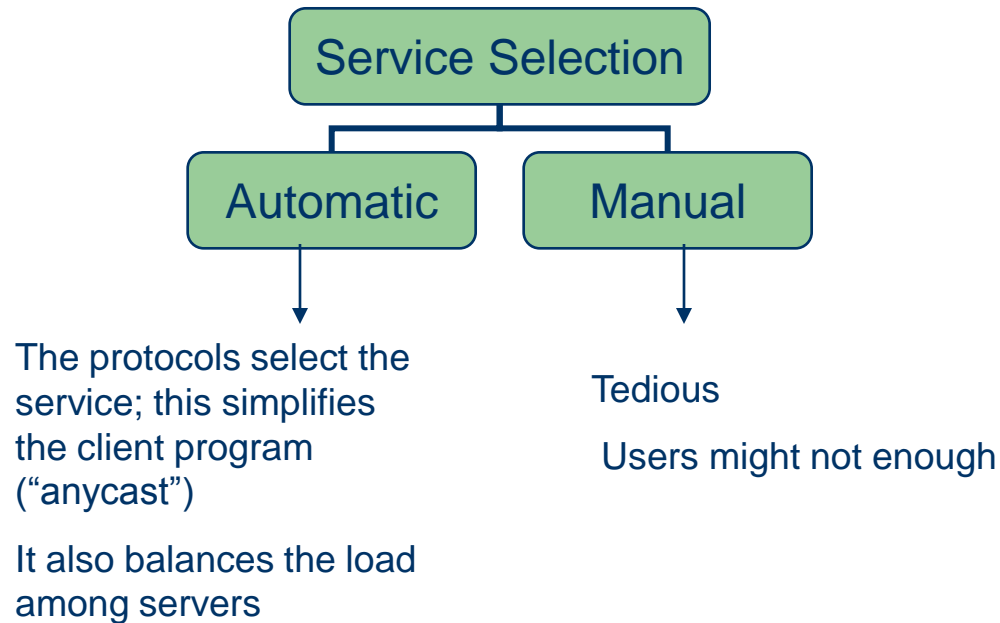
- **User Roles**

- Users authenticate with a domain or supply the designated domain as an attribute.
 - *User must have prior knowledge of target domain
- Implementation of this should reflect an ambient environment according to user role

- **Context Discovery Scope**

- Defined by temporal, spatial, and user activity information
- Proper use can save users time and effort in discovery agencies

Service Selection



Service Invocation

- Invocation Involves →

Level 1: Network's Service Address

Level 2: Underlying Communication Mechanism

Level 3: Operations specific to application domain

Service Usage

- **Explicit Release:** A client must explicitly release a service's resources once service usage is granted
- **Lease-based mechanism:** A client and the service negotiate the usage period (user can cancel/ renew it later)

This service handles dynamic conditions of the pervasive systems in a better way

Service Status Inquiry

- Used by clients to keep up with service events or status by polling or service event notifications.
- Two types of service status inquiry: **Polling** and **Service Event Notifications**
 - **Notification**
 - Clients register with a service and the service notifies client of something interesting such as a expiration date or upgrade to software
 - **Polling**
 - Used services generate events frequently or change status quickly

Security and Privacy

- Service discovery protocols must provide security and privacy to protect devices, services, and users
- Harder to implement changes due to changing environment
- Only current solution to environment changes is have people with special skills
- Scope of possible intrusion is increased due to wireless networks in a pervasive computing environment
- Clients, services, and directories should exchange sensitive information with legitimate parties
- What is legitimacy?
 - Refers to both valid credentials and access privileges on services
- Isn't always easy to acquire

Security and Privacy cont'd

- One way to verify legitimacy is to progressively exchange credential and information
- Compared to service discovery functionality, support for security and privacy in existing service discovery protocols is still in its infancy stages
- Because of different protocols being used, pervasive computing requirements cannot be met
- But with some revisions in discovery protocols and new protocols, we are able to support more security features
- With further research or possibly assimilating these protocols into maybe a “suite”, we can increase security and privacy

Conclusion

- Service discovery for unfamiliar protocols needs to be addressed more
- In order to compute at anytime or anywhere, these discovery protocols must work in unfamiliar computer environments
- These must become more intelligent to compensate for user's lack of knowledge, special skills, and unwillingness to trust the environment

References

- Zhu, Mutka, and Ni. *Service Discovery in Pervasive Computing Environments*. IEEE ppg. 2005 81-90.