The Problem

Sometimes users/applications need resources and services (e.g. hardware devices, storage, content servers, software) that aren't available on their computers.

As the network is changing all the time, we need to be able to discover services at runtime.

Users want to be able to find services by specifying criteria – cost, trustworthiness, etc.

We'd like to describe services, e.g. using XML. This info can specify how to use the service as well as what it provides.

We'd also like the client and server to know when the other is no longer accessible. (provisioning?)

Examples

Bluetooth uses discovery and pairing to create an ad-hoc network between two devices, where each can act as server or client. Bluetooth is an interesting thing to look into if you'd like, especially recent developments.

With bluetooth the first step is to discover devices, and the second is to discover services.

Wifi Direct creates ad-hoc networks using wifi. One device is assigned as the group network, and this controls what happens, especially regarding energy use.

Using a Directory

Similar to RMI Registries, you can use a directory services which provides information about available services.

This can be done with a centralised server or purely peer-to-peer.

Example: Jini

- Similar to RMI, but with extra discovery service

Discovery

Services can be defined by their signatures.

Jini has a lookup service which stores which services are available and information about them.

Advertising of services is for a limited time – the services are leased. This is a significant difference from RMI. Clients may only get a period of access time for the service, and then will have to renew the connection. E.g. the service may be available for 1 hour, and a given client may get access only for 10 minutes.

There is a protocol (set of rules) for services to discover a lookup service and register themselves with it. This is the discovery/join protocol.

Multicast is used for finding the lookup service. Multicast addresses a subset of computers in a network, rather than specifying an individual computer (as you would with unicast) or everybody (as you would with broadcast).

Once the lookup service has been found, it sends the servers a proxy that will mediate communication between the server and the lookup registry.

The client can discover the lookup registry the same way, and then query it for the specific service it wants. It then receives a stub/proxy as with RMI.

Service access can be exclusive or non-exclusive.

Scalability

- A federation of lookup services.

Lookup services can register themselves with one another and balance the load geographically or based on requests.

A penalty here is longer resolution time to find a service if it's not found in the first step.

Central Authority

Note there is no central authority here that controls the system. The lookup is centralised, but the control isn't.