# Resource Discovery

The resource discovery should allow the system to identify nodes in the network and what they are publishing and then subscribe to the relevant information as required. There will also be a mapping of resources to nodes to ensure that each node is aware of the resources it has and where to find the resources it needs.

# Assumptions:

The Resource Discovery system will be on a publish/subscribe model, where nodes will be able to exchange data under the abstraction of “topics”. The network will be an approximate graph structure.

# Requirements:

* Each node must be able to publish its available resources to the system.
* Each node must be able to find the resources its requires by searching through the network
* The node must then be able to subscribe to the resources it requires
* Each resource must have a relevant tag so that to can be easily identified when searching
* Each node must have a list of the tags which it has available to it
* Each node must carry a list of the resource tags available to its neighbours
* The resource list of each node must be updated periodically as required

# Research:

[1] ZeroConf

[2] Avahi

ZeroConf is a set of technologies that creates a computer network based on TCP/IP. ZeroConf uses the assignment of unique network addresses for each networked device, the resolution of computer hostnames and automatic identification of network services.

Avahi is a major implementation of ZeroConf and is the expected implementation which will be used for this project. Avahi is a Linux implementation of ZeroConf which is ideal for this project and also has a Python language binding which is also useful as Python will be one of the target languages for the project.

# Chosen Solution:

Each node, upon starting a new service, publishes to all of its neighbours that it has that service at distance 0. The neighbours likewise publish that service. Each node is now aware of the services it has and that which belong to its neighbours. Therefore, if a node receives a request for a service it doesn’t have it sends the request on to whichever of its neighbours has that service. Given its close correlation with Avahi, this will be the tool used to implement this.

# Deadline:

The deadline for this part of the project has been set as 4 weeks or the 15th of November, allowing time to combine with the Message Passing & Queuing and the Message Spec. to form the complete distributed communications system.