Naming

An Overview

Joseph Kehoe[1]

[1]Department of Computing and Networking
Institute of Technology Carlow

CDD101, 2018

Table of Contents

Table of Contents

Introduction

- A string of bits or characters refering to an entity
- An entity can be many things e.g. Printer, Server, file, disk, user, etc.
- Entities can be acted on (through an interface)
- Therefore they have access points
- Access points are known as **addresses**
- An entity may have more than one access point (e.g. Phone number)
- Entity can change access point over time

Identifiers

- Entity name should ideally be independent of its address **Location Independence**
- Entities need **Identifiers**

An Identifier has the following properties:

1. Refers to at most one entity
2. Each entity is refered to by at most one identifier
3. An identifier is permanent. It always refers to the same entity (e.g. PPS)

Institiúid Teicneolaíochta Cheatharlach
INSTITUTE of TECHNOLOGY CARLOW
At the Heart of South Leinster

Name Spaces

- A namespace is a labeled directed graph where:
- Each leaf node refers to an entity
- Non leaf nodes are kown as directory nodes (e.g. Domain Names)
- Absolute path ending in leaf node refers to unique entity
- Relative path ending in leaf node may refer to more than one entity depending on starting point
- Looking up a name is known as **Name Resolution**

Institiúid Teicneolaíochta Cheatharlach
INSTITUTE of TECHNOLOGY CARLOW
At the Heart of South Leinster

Name Space Implementation

Name Space graph is divided into three parts:

Global Level Top Level of graph. Here graph is very static (e.g .edu, .org, .ie in Domain names)

Administrative Layer Middle layer. Graph is still static in nature, does not change much

Managerial Layer Lower level. Most likely to change over time

- Static layers can be cached
- Use high performance and reliable servers (replicated) to run top levels

Institiúid Teicneolaíochta Cheatharlach
INSTITUTE of TECHNOLOGY CARLOW
At the Heart of South Leinster

Table of Contents

Locating Mobile Entities

- Allowing mobility for entities creates problems for naming
- use of identifiers helps (seperates naming from location)
- Identifiers tend not to be human friendly
- A location service is used to locate entities

Institiúid Teicneolaíochta Cheatharlach
INSTITUTE of TECHNOLOGY CARLOW
At the Heart of South Leinster

Mobile Entity Solutions

Many different approaches to handling mobile entities. All have issues.

Broadcast-Multicast Broadcast identifier to all and wait for response from current "location"

Forwarding Pointers When entity moves old reference keeps pointer to new location forming chains of pointers to entity

Home Based Approach Original location of enity is the home location. This always keeps track of current location of entity

Hierarchical Approach Similar to DNS. Caching can help here if entities are not highly(?) mobile

Institiúid Teicneolaíochta Cheatharlach
INSTITUTE of TECHNOLOGY CARLOW
At the Heart of South Leinster

Table of Contents

Removing Unreferenced Entities

Similar to garbage collection for pointers but much worse

- How do we track references to an entity when those references come from many distributed machines?
- Reference counting helps but we must count across machines
- **Weighted Reference counting** helps. But it still has issues
- **Reference listing** is where each skeleton keeps a list of every proxy that links to it (Java RMI)