Service Discovery in Zero Configuration Networks

Dr Stuart Cheshire,
Distinguished Engineer, Apple Inc.
About the Presenter

Stuart Cheshire

• BA, MA, Sidney Sussex College, Cambridge
• MSc, Ph.D., Stanford University, California
• Distinguished Engineer, Apple Inc.
Outline

• What is Zero Configuration Networking?
  – Why / What / How
• APIs
• Power Management
• Q & A
What Is Zeroconf Networking?

• Why?
  – Local Area Communications (0–10m) chaos

• What?
  – Wide Area has converged on IP as the One True Protocol—why not Local Area too?

• How?
  – How do we make IP meet this challenge?
Why?
Wide Area Convergence

- DECnet
- Xerox XNS
- TCP/IP
- OSI
- ATM-to-the-desktop
- IBM/Microsoft NetBEUI
- AppleTalk
- Etc., etc., etc.
Wide Area Convergence

• DECnet
• Xerox XNS
• TCP/IP
• OSI
• ATM-to-the-desktop
• IBM/Microsoft NetBEUI
• AppleTalk
• Etc., etc., etc.
Local Area Chaos

- Parallel Ports
- Serial Ports
- SCSI
- ADB
- IrDA
Local Area Chaos

- Parallel Ports
- Serial Ports
- SCSI
- ADB
- IrDA
Local Area Chaos

- USB
- FireWire
- Ethernet
- 802.11
- ATA
- Bluetooth
Local Area Chaos

- USB 2, USB 3
- FireWire 800
- Gigabit Ethernet
- 802.11a/b/g/n
- Serial ATA
- Bluetooth 2, Bluetooth 3, Bluetooth 4, Bluetooth LE
- UWB (Ultra Wide Band)
- NFC (Near Field Communication)
- ZigBee
- Z-Wave
- Thunderbolt
What?
Pick One Protocol

• Wide Area Communications have already converged on IP
• Why not Local Area Communications too?
TCP/IP over Ethernet, Wi-Fi, etc., with the ease of use of USB
How?
Three Legs of Zeroconf Networking

• Addressing
• Naming
• Discovery
Addressing

• Self-Assigned IPv4 Link-Local Addressing
  – Pick random address in 169.254/16
  – ARP to see if anyone else is using it
  – If someone else is using it, try again
  – Ongoing conflict checking
• RFC 3927
  – “Dynamic Configuration of IPv4 Link-Local Addresses”
• RFC 2462
  – “IPv6 Stateless Address Autoconfiguration”
Link-Local Availability

• Self-Assigned IPv4 Link-Local Addressing first appeared in:
  – Mac OS 8.5, Summer 1998
  – Windows 98, Summer 1998
  – Mac OS X 10.0
  – ZCIP for Linux http://zeroconf.sourceforge.net/

• You may know it as Autonet, AutoIP, etc.

• IPv6 has always included Link-Local Addressing
Naming

• Multicast DNS
  – Pick desired name in ".local." subdomain
  – Issue Query to see if anyone else is using it
  – If someone else already using it, pick another
  – Ongoing conflict checking

• draft-cheshire-dnsext-multicastdns-14.txt
  – (Soon to be RFC)

• http://www.multicastdns.org/
Multicast DNS Availability

• Multicast DNS client
  – Mac OS 9.2
  – Mac OS X
  – iPhone & iPod touch
  – Linux
  – Bonjour for Windows
    http://support.apple.com/downloads/Bonjour_for_Windows
    http://www.apple.com/safari/

• Just type
  – "laserwriter.local." into your Web browser
  – "ssh mymac.local." into a terminal window
Discovery

• Raising the bar
• Should not need to know name in advance
DNS Service Discovery

• Devices already need:
  – IPv4 Link-Local Addressing
  – Multicast DNS
• Don’t need more code
  – Multicast DNS gives us Service Discovery too
• draft-cheshire-dnsext-dns-sd-10.txt
  – (Soon to be RFC)
• http://www.dns-sd.org/
Discovery Via DNS PTR

• DNS Query:

_ipp._tcp.local.  PTR  ?
Discovery Via DNS PTR

- DNS Response(s):

  _ipp._tcp.local.   PTR
  Sales._ipp._tcp.local.
  Marketing._ipp._tcp.local.
  Engineering._ipp._tcp.local.
  3rd Floor Copy Room._ipp._tcp.local.
Components of Service Name

• User-Visible Name
  3rd Floor Copy Room._ipp._tcp.local.

• Service Type (Application Protocol Name)
  3rd Floor Copy Room._ipp._tcp.local.

• Domain
  3rd Floor Copy Room._ipp._tcp.local.
Service Types

• A Service is identified by what protocol it uses
• Register your Protocol Names at:
  http://www.dns-sd.org/ServiceTypes.html
• draft-ietf-tsvwg-iana-ports-10
Lookup Via DNS SRV

• DNS Queries:

Sales._ipp._tcp.local.   SRV ?
Sales._ipp._tcp.local.   TXT ?
Lookup Via DNS SRV

• DNS Responses:
  Sales._ipp._tcp.local.    SRV    0 0 631 my-printer.local.
  Sales._ipp._tcp.local.    TXT    pdl=application/postscript
  my-printer.local.         A      169.254.12.34
DNS-SD Availability

• On OS X 10.2, iPhone, iPod touch:
  – System API
• Bonjour for Windows
  http://developer.apple.com/bonjour
• Other Platforms:
  – Use Apple’s Darwin Open Source
  – Other Independent Implementations
  – Implement from spec: draft-cheshire-dnsext-dns-sd-10.txt
• http://www.dns-sd.org/
Technology

• Link-Local Addressing
  – IPv4 (RFC 3927)
  – IPv6 (RFC 2462)

• Multicast DNS
  – http://www.multicastdns.org/

• DNS Service Discovery
  – Link-local and wide-area
  – http://www.dns-sd.org/
Why Do You Care?

• Lower support costs
• Fewer returns
• New product categories
• Network products that are a joy to use
Setting up Devices the Old Way
Apple AirPort Express
SitePlayer Telnet

$30–$140 including tax and shipping
Demo
Language Bindings (APIs)

- **Cross-platform C API**
  - Mac, Windows, Linux, etc.
  - /usr/include/dns_sd.h
- **Mac OS X:**
  - CoreFoundation
  - Cocoa (Objective C)
- **Java**
- **Python**
- **Ruby**
Three Basic Operations

Server

Register

Client

Browse (Discover Domains & Services)

Resolve
Registering a Service

DNSServiceRegister()

(char     *name,
 char     *regtype,
 char     *domain,
 Opaque16  port,
 char     *txtRecord,
 DNSServiceRegReply  callBack,
 void     *context
);


Discovery

DNSServiceBrowse

(char *regtype,
char *domain,
DNSServiceBrowserReply callBack,
void *context);

39
Resolving

DNSServiceResolve

(char *name,
char *regtype,
char *domain,

DNSServiceResolverReply *callBack,
void *context

);
DNS Service Discovery for Java

• Java API
  – Same functionality as <dns_sd.h>
• Uses system-wide DNS-SD/mDNS service
  – Requires mDNSResponder / mdnsd
• Part of Apple’s mDNS reference implementation
How the API works

• Services invoked through DNSSD factory class
• Many services are non-blocking; clients implement an interface which is called with operation results
  – Callbacks are made from a different thread
• Utility classes for
  – rescheduling callbacks on the AWT Event Thread
  – manipulating DNS-SD format TXT records
Example 1 - Registering a Service

```java
import com.apple.dnssd.*;

class MyRegistrar implements RegisterListener {
    void registerWebService() {
        DNSSDRegistration reg =
            DNSSD.register("Me!", "_http._tcp", 80, this);
    }

    public void serviceRegistered(
        DNSSDRegistration reg,
        int flags, String serviceName,
        String regType, String domain) {
        System.out.println("Registered service "
            + serviceName);
    }
}
```
Example 2 - Discovering Services

```java
import com.apple.dnssd.*;

class MyBrowser implements BrowseListener {
    void browseForPrinters() {
        DNSSDService browser =
            DNSSD.browse("_ipp._tcp", this);
    }

    public void serviceFound(
        DNSSDService browser, int flags,
        int ifIndex, String serviceName,
        String regType, String domain) {
        System.out.println("Found service "+ serviceName);
    }
}
```
Example 3 - Resolving a Service

```
import com.apple.dnssd.*;

class MyResolver implements ResolveListener {
    void resolveService(String service,
                         String type, String domain) {
        DNSSD.resolve(0,0,service,type,domain,this);
    }

    public void serviceResolved(
        DNSSDSERVICE resolver,
        int flags, int ifIndex,
        String fullName, String hostName,
        int port, TXTRecord txtRecord) {
        System.out.println("Resolved service to 
                         + hostName);
    }
}
```
Platform support and Packaging

• Consists of a Java .jar file and a JNI library
  – JNI library builds on OS X, Windows, and POSIX
• Part of the mDNSResponder project (Darwin)
• Ships with Mac OS X and Bonjour for Windows
Command-Line test tool

• Register
  `dns-sd -R <Name> <Type> <Domain> <Port>`

• Browse
  `dns-sd -B <Type> <Domain>`

• Resolve
  `dns-sd -L <Name> <Type> <Domain>`
Power Management

• Sleep Proxy
• Client transfers DNS-SD records to proxy before sleeping
• Proxy answers on behalf of sleeping client
• Proxy wakes client when appropriate
• Based on standard DNS update
• With EDNS0 option giving owner’s MAC address
  – draft-cheshire-edns0-owner-option-00
• Bonjour Sleep Proxy shipped June 2009 with Mac OS X 10.6 Snow Leopard
For More Information

Web Sites

Zero Configuration Networking
http://www.zeroconf.org/

Multicast DNS
http://www.multicastdns.org/

DNS Service Discovery
http://www.dns-sd.org/