Using DNS for Global Discovery of Environmental Services

Introduction and Problem Statement
- Embedded sensor devices getting pervasive in everyday lives.
- Due to Internet penetration in real world, sensor devices become globally addressable.
- Web of Things reuses well-accepted and understood Web standards to interconnect IP-enabled resource-constrained devices.
- Embedded sensors can fully integrate to Web: directly by embedding Web servers on them or indirectly by means of gateways.
- In future, Web-enabled sensor devices will constitute large majority of Web population, exposing their functionalities as open RESTful Web APIs.
- As yet, no standardized, scalable and flexible ways to globally discover these embedded devices, based on their characteristics and capabilities.

Proposal
- Exploit existing Internet infrastructure to achieve real-time discovery of embedded devices and environmental services.
- Utilize Domain Name System (DNS) as a scalable, pervasive, global meta-data repository for embedded devices. Extend for supporting location-based discovery of Web-enabled physical entities.
- Service-centric Vs Domain-centric Discovery (DNS-based Service Discovery, Multicast DNS)
- Enable global, real-time, location-based discovery of pervasive services, offered by Web-enabled sensor devices.

Device Registration
Include a new top-level DNS domain: .env
.env domain intended to support embedded devices and environmental services which are enabled to the Web and registered to the DNS.
Real-time registration of devices and RESTful Web services through Dynamic DNS (DDNS)
Each sensor device will be assigned a unique hostname that has following format: Devicename_.http_tcp.service.location.env

Service Discovery
Begins when user types in Web browser URL ending with .env label.
- asks about all sensor devices offering service and deployed in location.
- receives list of relevant instances; then constructs relevant URL.
By invoking this URL in Web browser, DNS translates it to actual IPv4/IPv6 address of corresponding sensor device.
When IP address is resolved and request is forwarded to appropriate sensor device, device needs to describe its functionality. Example description languages:
SensorML, EEML, WADL. WADL is an XML-based language that provides a machine-readable (RESTful) description of HTTP-based applications.

Practical Issues
- Freshness of Information: Dynamic DNS Update Leases constitutes a method of extending DDNS to contain an update lease time. Hence, devices may be forced to re-register frequently to the DNS.
- DNS Management: Since DNS follows a hierarchical structure, addition of .env top-level domain not expected to be a complicated task.
- Information Reliability: Partially by checking IP addresses of sensor devices, if they fall in locations claimed during the registration process. (A user-based feedback/reputation system similar to eBay?)
- Increased Traffic: Traffic increase expected to be severe, especially considering existing infrastructure. Hundreds (only?) DNS servers would be needed.
- Unique Devicename Assignment: Upon a conflict, .env DNS server should automatically select new name for device, typically by appending a digit at end of its name.
- Uniform Naming of Services and Locations: To avoid language ambiguities, grammar mistakes etc., electronic directory services such as X.5009 could be used, where distributed database would contain unique translations for services and locations.
- Caching of Information: During registration, sensor devices could include their latest measurements, stored in TXT resource records. However, how long would the measurement stay fresh?

General Procedure
1. Device Registration (devicename, service, location)
2. Device Acknowledgment
3. PTR DNS Request (service.location.env)
4. DNS Response
   (List of devicenames for service.location.env)
5. A/AAAA DNS Request
   (devicename.service.location.env)
6. DNS Response (IPv4/IPv6 Address Translation)
7. WADL Request
8. WADL Response
9. GET service
10. Environmental Data in XML/JSON

Challenges/Potential
- Automation: Select sensor device from list returned by .env DNS server, parse its WADL file and construct HTTP requests.
- Personalization: Select only devices that meet particular user preferences. For example, devices having positive online feedback or belonging to well-known authorities such as governmental organizations.
- User Participation: A culture could be created around concept of sharing environmental services with rest of the world.
- Generalization: Support any kind of physical devices and/or pervasive services. To achieve this, standardized “domain vocabularies” need to be created, for facilitating the construction of queries by end users. For example, a user that wishes to park his car in Barcelona could just need to type in his Web browser parking.cars.Barcelona.
- Knowledge Inference: Defining extended environmental ontologies would encourage automatic information retrieval, generalized inferences and advanced Web mashup development very easily. For example, when temperature in Porto is obtained, then general temperature of Portugal can be automatically inferred.
- M2M Communication: Ad hoc, plug and play machine-to-machine communication would be enabled by discovering and interacting automatically with Web-enabled physical devices, deployed anywhere around the world.

Andreas Kamilaris and Andreas Pitsillides
Department of Computer Science, University of Cyprus
kami@cs.ucy.ac.cy, Andreas.Pitsillides@ucy.ac.cy
http://www.netri.cs.ucy.ac.cy/