Where on Earth is the Spatial Name System?

Ryan Gibb
University of Cambridge
ryan.gibb@cl.cam.ac.uk

Anil Madhavapeddy
University of Cambridge
anil.madhavapeddy@cl.cam.ac.uk

Jon Crowcroft
University of Cambridge
jon.crowcroft@cl.cam.ac.uk

Abstract

The existing Internet architecture lacks support for naming locations and resolving them to the myriad addressing mechanisms we use beyond IP. We propose the Spatial Name System (SNS) that allows for the assignment of hierarchical location-based names and for resolution schemes that are both global and local. Since we extend the DNS, our scheme allows for the integration of spatial names into existing applications and opens up new possibilities for sensor networks and augmented reality.

CCS Concepts

• **Networks** → **Naming and addressing**; **Network protocol design**; *Location based services*; • **Human-centered computing** → Ubiquitous computing.

Keywords

Spatial, Naming, Addressing, DNS, Network Architecture

ACM Reference Format:
Ryan Gibb, Anil Madhavapeddy, and Jon Crowcroft. 2023. Where on Earth is the Spatial Name System?. In *The 22nd ACM Workshop on Hot Topics in Networks (HotNets '23), November 28–29, 2023, Cambridge, MA, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3626111.3628210

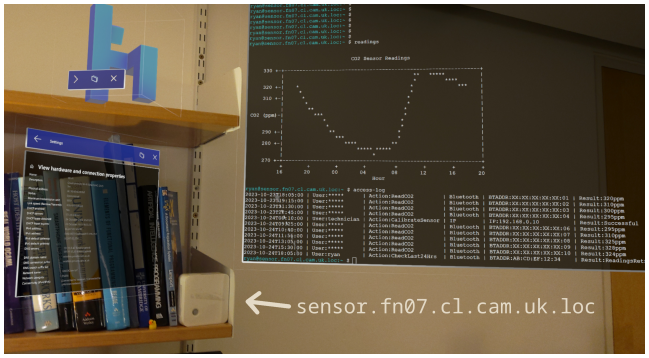

Figure 1: Connecting to networked devices just by looking at them (HoloLens 2 AR prototype)

1 Addressing the Location Gap

The lack of support for naming physical locations is an omission at the heart of the Internet architecture. While there have been many advances in *addressing* locations via multiple routing schemes, it remains difficult to refer to location-based services via *logical names*. This in turn makes it difficult to deploy network services that can be referred to by a stable name that specifies a given location, and that resolves to the addresses of the devices in that space. This matters because there are a broad class of network-connected devices with a physical presence to which location is an intrinsic part of their identity. A networked speaker in, say, the Oval Office is defined by its location: it's simply the Oval Office Speaker! If the specific device moves location its identity should change with its new location, and if the device is replaced then the replacement should assume the function of its predecessor.

Using a device's spatial location to identify it should therefore be both natural and ergonomic, but is currently neither. The emergence of augmented reality (AR) interfaces provides fresh impetus to solve this, as such interfaces should just be able to connect to networked devices in the environment and interact with them *simply by looking at them* through the AR headset (Figure 1 shows our prototype HoloLens interface). At the other extreme, consider environmental sensors in remote environments. These sensors typically have no global connectivity but can work in a local context via ad-hoc networking and often need delay-tolerant networking techniques to communicate [52]. Ideally, we could refer to these sensors by a stable domain name that does not change if the underlying sensor is swapped out for a replacement with a different network address.

This paper explores how to bridge physical and networked locations by extending the Domain Name System (DNS)—the standard for both global [39] and network-local naming [12]—and tackles several problematic aspects of the DNS towards supporting spatial naming. Firstly, the layers inherent in existing service discovery mechanisms mean that it can take seconds or even minutes to discover devices, whereas AR headsets must perform lookups in milliseconds as the user moves their gaze. Secondly, physical proximity also brings in a diverse range of connectivity options that aren't available in existing DNS resource records (RRs), such as non-IP protocols like Bluetooth. Thirdly, resolving a physical location of varying resolution to a network address is not a mechanism currently supported by the DNS standards. Finally, any modifications must be backwards compatible to maintain interoperability with existing endpoints.

HotNets '23, November 28–29, 2023, Cambridge, MA, USA

ACM ISBN 979-8-4007-0415-4/23/11.
https://doi.org/10.1145/3626111.3628210

Our spatial naming approach (§2.1) first extends the DNS with support for resolving the full set of *addresses* possible for devices, many of which are non-IP-based protocols such as Bluetooth or Zigbee (§2.2). We then extend the name resolution mechanisms to map human-readable *spatial names* to these network addresses (§2.3). This resolution can be split-horizon, and so resolution via a physically local spatial resolver could provide a layer 2 network address such as a Bluetooth Device Address, whereas resolving remotely might return a globally accessible IP address (§3.1). We then add support for querying a physical space by resolving geographic coordinates to spatial names tied to devices (§3.2). We then describe how this Spatial Name System (SNS) can be incrementally deployed as an extension to the DNS (§4.1), discuss the threat model (§4.2), and discuss alternative approaches (§4.3). We conclude by exploring the new class of applications (such as AR) that are enabled by this shift to spatial naming (§4.4).

2 Extending the DNS to Spatial Naming

We now sketch out a sample scenario of how we might use DNS for spatial lookups (§2.1), and then examine how we could extend the DNS protocol to support more address types (§2.2), and then define our spatial naming architecture (§2.3).

2.1 An Example Spatial Name

Consider `1600.penn-ave.washington.dc.usa.loc` as the DNS name for the White House. If `mic.oval-office` is a device within the White House, then a local resolution from within the room returns multiple resource records (RRs); private IPv4/6 addresses for the local wireless network, and another with Bluetooth and Zigbee addresses (§2.2). The domain names can also be combined into a fully qualified domain name, allowing the device to be named globally as a URI, e.g., `capnp://mic.oval-office.1600.penn-ave.⟷ washington.dc.usa.loc/secret`. The example here is for invoking RPCs using Capnproto [57], but more conventional URIs such as `https://` should be self-evident. The key insight here is that we can use all the existing mechanisms in the DNS to map spatial names:

- Local spatial names are completed via the resolvers appending their global location to a query, meaning clients just need to know their relative location.
- Split horizon DNS restricts lookups for a specific subdomain unless made from within that space (§3.1).
- Extensible DNS resource records (RRs) enable clients to establish non-IP connectivity to devices given physical (wireless) proximity (§2.2).
- Spatial names can operate as a subdomain of an existing DNS domain, with a top-level domain (TLD) `.loc` (for location) to scale the scheme. These could interoperate for incremental deployment.

Once we can perform these spatial lookups using standard DNS, then many existing services can "just work" as location-based services and unlock applications in many areas.

2.2 Modernising Resource Records

Modern devices all have a diverse set of connectivity options, ranging from IEEE802.11, cellular, Bluetooth or Zigbee, over which IP connectivity can be layered. Although DNS is usually used for IP addresses, it is possible to extend it to return records beyond just IPv4/6 addresses (see Table 1). Indeed, in the early days of the DNS, there were records for now-unused protocols such as X25 and ISDN [56].

Protocol	RR Type	Sample Entry
IPv4	A	`192.0.2.1`
IPv6	AAAA	`2001:db8:0:0:0:0:0:1`
Bluetooth	BDADDR	`01:23:45:67:89:AB`
802.11	WIFI	`(<ssid>, 192.0.3.1)`
LoRaWAN	LORA	`(<gw>, <devaddr>)`
Audio [38]	DTMF	`<tone-prefix>`

Table 1: Existing and extended DNS RRs

There are several benefits to extending the types of RRs available. With so many physical connectivity options available, a connecting device today needs the user to know which address to select [48] or has to perform expensive wireless scans [27] across all the protocols to determine if they are available as an option. Having a name system act as a registry for these local connectivity options is a natural extension to the DNS, as it permits connecting devices to choose the most appropriate option before committing to any one mechanism A DNS lookup against our earlier example of a microphone device might return RRs for Bluetooth and Zigbee or even audio [38], as well as IPv4/6. These addressing extensions to the DNS are backwards compatible, since they may also be encoded as `TXT` records as a fallback – a technique commonly used to interoperate with DNS middleboxes.

2.3 Spatial Names

Names serve to identify resources in a manner that can be easily interpreted by humans. Domain names are used to identify a device in a realm of administrative autonomy, authority, or control within the internet. Domain names often informally contain spatial information [13, 29, 51], but the DNS does not inherently support spatial location as a property. In contrast, spatial names are designed to identify a location, giving it an identity that can be referred to within the network.

We consider two distinct classes of location encodings: civic and geodetic [47, 55]. Civic names are a location based on structured human-readable addresses which might include elements such as the room, building number, street name, city, state/county, postal code, and country, which form a hierarchy representing containment. For example, "Oval Office, 1600 Pennsylvania Ave NW, Washington, DC".

Geodetic locations refer to a location described in terms of geographic coordinates such as latitude, longitude and optionally altitude. For example, the coordinates "38.8974° N, 77.0374° W". This may be on a grid that partitions the whole world, or be constrained to a bounded area. In many

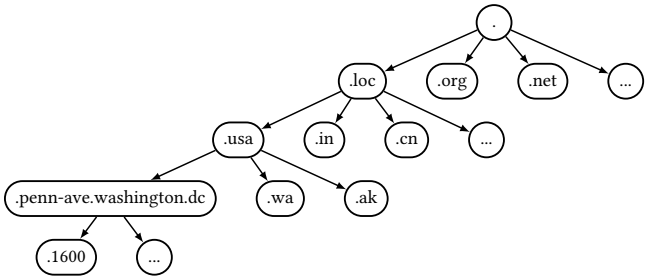

Figure 2: A Spatial Name Hierarchy

applications, knowing the exact latitude and longitude of a device is less useful than knowing its civic location. It is often more relevant to know that a device is in a particular room than its precise coordinates. Civic locations provide such human-understandable names. Another key aspect of a naming system is its ability to provide unambiguous identifiers. While geodetic locations define a point in space, two distinct physical locations can appear identical if the precision of the resolution is not sufficient or the altitude differs. In the context of naming this presents a fundamental challenge.

There is also the challenge of finding unambiguous and unique civic encodings. DNS authority delegation means that a subdomain can have its own mechanism for providing civic encodings, similar to different countries having differing postal address formats. DNS internationalization could even be used to provide locations in localized scripts [32]. While civic hierarchy also provides a delegation of authority, geodetic names have no such hierarchy as every location is on a global coordinate system. This coordinate space can be partitioned but is still one address space. Who then has the authority to add, update, and delete records at a location? Rather than build a complex globally distributed spatial database, we can instead use the civic location hierarchy to provide a more natural mechanism to delegate authority.

So due to human-understandability, relative disambiguity, and delegation of authority, we will use civic location encodings for spatial names. Our insight is that we can represent these spatial names as domain names and implement the SNS as an extension of the DNS. This allows us to interoperate with existing protocols to maintain backward compatibility (§4) and use an existing globally distributed key-value store to scale to the planet. Figure 2 shows an example spatial name hierarchy starting from the root with arrows denoting delegation to subdomains. The `.loc` root is shown alongside other TLDs to demonstrate DNS interoperability. Entries below `.loc` follow a possible civic location encoding.

We can assign human-meaningful hostnames to devices based on their function within a spatial domain. For example, if we have a microphone right next to a display which might overlap with geodetic names, we can assign names representational of their functionality, with the civic hierarchy ensuring these names remain unique. A microphone in the White House would have a different civic name from a microphone in 10 Downing Street due to their different spatial domains.

Traditional domain names also rely on administrative delegation, which poses the question for the SNS: who is going to administer these spatial domains? Governments, states, and city councils would be the obvious candidates for the first, second, and third-level domains of the global `.loc` TLD. However, to support incremental deployment, spatial subdomains at existing DNS domains are possible, for example `whitehouse.loc.usa.gov`. Within a spatial domain, it is possible to support zero-configuration deployment with devices assigning themselves names based on local positioning systems (§3.2), just as DHCP/DNS servers can assign devices network names based on their client identifiers.

3 Spatial Resolution Schemes

We next discuss more detailed designs for context-dependent (§3.1) resolutions, and resolutions of physical locations (§3.2), in the Spatial Name System.

3.1 Split-Horizon Resolution

The resolution of a spatial name depends on the context. The resolution of a name from within a spatial domain can return internal, local network addresses (§2.2). On the other hand, if a spatial name is resolved by a node outside, it may return a global, publicly routable IP address. We can utilize split-horizon DNS to provide these context-dependent resolutions.

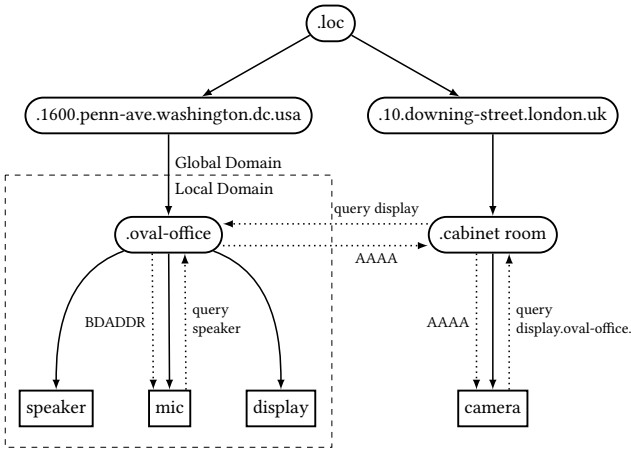

Figure 3: Context-dependent Spatial Resolution

Consider a microphone in the Oval Office (Figure 3), which can resolve the spatial name of a nearby speaker to its local Bluetooth Device Address using a `BDADDR` record. This local spatial resolution can be particularly useful where devices need to interact with each other over short distances. Now consider a camera installed in the 10 Downing Street cabinet room that needs to resolve the spatial name of a TV located in the Oval Office. The camera is outside of the Oval Office's spatial domain and hence gets the globally resolvable AAAA record corresponding to the display screen. If the device hosting the spatial name is behind Network Address Translation (NAT), a global IP could be dynamically created for a

particular port as a side-effect of the DNS resolution using, for example, the Port Control Protocol [59] or VPN tunnels that are maintained for the duration of the DNS response TTL, permitting external access to the display.

However, some devices might not be appropriate to resolve globally (such as the microphone in the Oval Office) and we need to augment the SNS with access control for this scenario. Beyond general DNS authentication mechanisms, authentication to a spatial domain can be achieved by taking advantage of physical locality. For example, authentication to a room like the Oval Office could be done by being physically present in the same space using audio beacons that chirp an encoded message to prove presence [36, 38]. With this system, the Oval Office's microphone could only respond to resolutions from devices already in the same room, refusing to return any address to resolvers outside.

3.2 Geodetic Resolution

While spatial names provide a human-readable, hierarchical understanding of location, they do not allow the discovery of devices by their geographic coordinates. Since many location-aware applications are concerned with the precise physical location of devices, we introduce a geodetic resolution to resolve a coordinate-based location to spatial names or network addresses. Geodetic resolution allows us to answer queries such as *"which devices are in this area?"*.

An explicit form of location encoding is available in the existing DNS, in the form of `LOC` records [14]. LOC records map a domain name to a location with a latitude, longitude and altitude; as well as optionally the size of the location and its precision. However, they have seen little real-world deployment [18]. For the SNS's purposes, there is no way to resolve *from* a location to a name or address. However, LOC RRs [14] could be one method used to encode these geodetic locations. Other encodings would be possible with custom TXT records, including non-global geocodings, and encodings supporting polygons.

For this to work, devices would need to have access to some form of location-aware technology. This could be as simple as a user manually registering a device's location when it's installed, or global navigation satellite systems (GNSS) – like the Global Positioning System (GPS) or Galileo – can be used to provide global geolocation to passive receivers. GNSS is limited in its accuracy indoors and in dense urban areas, however. An alternative is Indoor positioning systems (IPS) which use radio waves or other sensory information to locate objects within a building [54]. One early example is the Active BAT system [22].

To perform geodetic resolution, we need to associate values (names or addresses) with these geodetic locations. The exact semantics of the geodetic coordinate could vary, but we'll take the physical area as a simple example. When performing a resolution we check a queried area for any values with intersecting coordinates; i.e., we find the overlap between our queried area and areas associated with values.

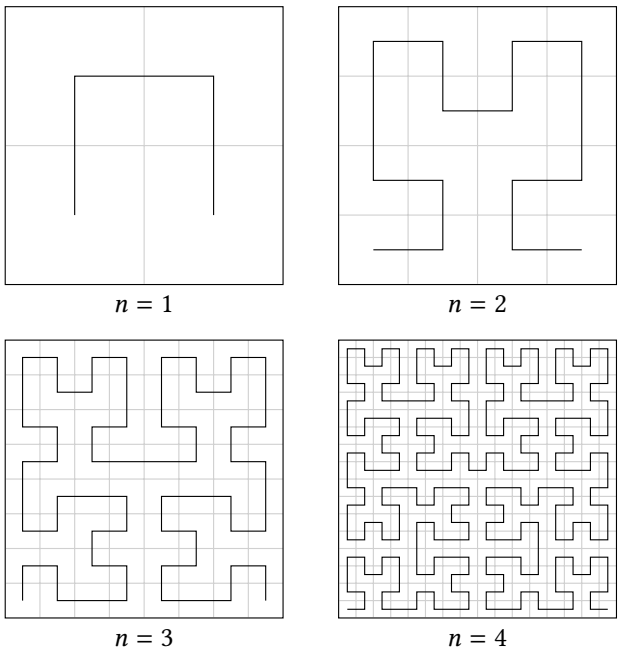

Figure 4: Hilbert Curves of Order n

A naive solution to this would involve checking every device's location in a domain against the query area, but this would be $O(n)$ for n devices. Instead, we can use existing work from spatial indexing [9, 35, 45] to optimize our queries. One possibility would be using space-filling curves to partition an area and provide a spatial index. Hilbert curves are one such curve which preserve locality when mapping a 1D space to a higher dimension [23]. We can efficiently lookup overlapping interval ranges of this Hilbert curve to calculate the intersection in logarithmic complexity [16]. Hilbert curves with varying order can be used to provide varying precision (see Figure 4). We emphasise that this is just one possible scheme, however, and alternatives such as R-trees [8, 21] may be more efficient for sparse locations.

We have discussed geodetic resolution in the context of a local spatial domain, but we can extend this system to global resolution using the SNS hierarchy. Spatial domains down the hierarchy could have geodetic mappings, so a query to "38.8974° N, 77.0374° W" (the Oval Office) would start at '.loc', which would return '.usa' as the next domain to check, operating like normal iterative DNS. The resolver could iterate down the hierarchy until it reaches the `oval-office` nameserver. Note that these high-level spatial domains, like country code second-level domains, will have very complex geometries, which further motivate the work into novel spatial indexing schemes for this globally distributed geodetic database. There is also the question of handling ambiguity – what if you query a point right on the border? Returning a set of RRs in the DNS authority section [1] could be used to point the resolver to multiple spatial domains, which it can then pursue concurrently.

4 The Spatial Name System

We will now discuss how the SNS can be implemented as an extension of the DNS (§4.1), then consider the security and privacy implications of using the DNS (§4.2), give an overview of related work (§4.3) and describe the applications the SNS enables (§4.4).

4.1 Spatial Deployment

While the spatial split-horizon (§3.1) and geodetic resolution (§3.2) necessitate modifications to clients and nameservers, implementing the SNS as an extension of the DNS allows it to be rolled out incrementally across authoritative DNS nameservers and stub resolvers, work harmoniously with existing middleboxes, and work with end devices [34]. This allows the SNS to interoperate with existing DNS features and DNS-based protocols. The DNS domain space can be used with the proposed `.loc` TLD, and existing DNS resolver infrastructure can be used to perform queries.

While our motivating examples are for static devices, we can layer a limited form of mobility using existing DNS mechanisms. If a device moves between spatial domains and wants to retain communication with its identity at its former location, it can use a `CNAME` record to point to the new location. If a device moves geodetic location, updates to the geodetic mapping within a local spatial domain could be done using dynamic DNS updates [58]. Aside from naming, existing network mobility techniques can be used for established connections [6, 40, 43].

DNSSEC [25] operates as usual, which enables us to have authenticated answers to spatial queries. This also allows us to securely provision public keys with the SNS using `SSHFP` records [19]. This, in turn, enables the transparent replacement of devices at a spatial name, as even their public keys can be replaced through the naming system.

DNS Service Discovery (DNS-SD) [10] uses standard DNS protocols, including mDNS [11] for the local link, to discover services available in a given domain. With SNS, this domain becomes a spatial domain, such as a particular room, floor, or building. DNS-SD augmented with spatial information makes service discovery more context-aware since it is no longer about finding a service anywhere on the network but rather about finding it in the spatial environment.

4.2 Security and Privacy Considerations

The DNS has a long history of problems with misconfiguration and security. Their impact is potentially far more severe for the SNS given the sensitivity of location information.

First, consider the risk of unauthorised access to devices or locations. Sometimes the mere existence of a name could be sensitive information. However, the DNS does not implement a generic form of access control, every record is instead public. The closest form of preventative measures to restrict the information accessible is Transaction Signatures (TSIG) for zone transfers [15] and DNSSEC next secure record version 3 (NSEC3) RRs preventing zone enumeration [5]. Future work on SNS access control (§3.1) could provide access control to resolving and addressing a particular device. This would be augmented with fine-grained application layer access control to specific services.

Second, there is the risk of privacy violation. SNS query privacy is potentially of more concern than traditional DNS resolutions as an adversary could use them to track user's locations. While DNS-over-TLS [28] and DNS-over-HTTPS [26] encrypt the queries, recursive resolvers can correlate client IPs with unencrypted queries [37]. There are existing schemes for private DNS resolutions that aim to disassociate the query from the address making the query [46, 50] that could be used to mitigate this.

Third, there is the risk of address spoofing. An attacker could manipulate DNS responses to redirect traffic to a malicious party. DNSSEC can work as normal to mitigate this [25].

We also consider the risk of service disruption via a denial of service attack to disrupt the resolution process and hence the functionality of local devices. To address this, as well as accommodate this context-dependent operation, we propose deploying authoritative nameservers to the edge of the network. These nameservers would hold authority over the devices within their respective spatial domains. It is not necessary to have a unique nameserver for each room; instead, a single nameserver could manage a local area. These nameservers would interact with devices joining the network via protocols such as DHCP, promptly assigning them global spatial names, as well as managing NAT. By moving the responsibility of DNS operations to the edge of the network, we can support low-latency name resolution for local devices as well as offline operation, ensuring continued functionality for local devices even in the face of service degradation or disconnection from the wider internet.

4.3 Related Work

4.3.1 Geographic Routing and Addressing. In the Internet, a host's topological location (denoted by an IP address) is vital for routing and addressing. Current Internet protocols relegate support for spatial location to inference from other network properties such as domain names, propagation delays, and IP addresses, which are limited in their accuracy and precision [17, 20, 31, 42, 49]. This is intentional as location-independence is desirable for most of the Internet, but there is also a class of devices for which spatial location is core to their function.

Geographic routing (georouting) schemes, such as Greedy Perimeter Stateless Routing [30], do support spatial location in the network by making forwarding decisions based on nodes positions in order to route to a location. Similarly, geographic casting (geocasting) makes spatial location explicit by encoding it in network addresses [41]. Geocast uses geographical coordinates as part of the addressing scheme, utilizing IPv6's support for geographic-based unicast addresses with the reserved binary prefix '100' [24]. It operates by utilizing geographic coordinates or areas as destinations

for packets and defines a geographical region of hosts in the network as the destination for a network message. These messages are then routed to all devices within this specified region, making it possible to target data delivery to a specific geographic area. These approaches overload the meaning of IP addresses with both topological and spatial locations. In contrast, a spatial name service allows us to decouple logical names from the concrete location addressing schemes used.

4.3.2 The Intentional Naming System. Another system that *does* solve this at the naming layer is the Intentional Naming System (INS) [4]. The INS uses a language based on attributes and values to name services and resources. This is considerably more complex than domain names and includes multiple dimensions such as resource location and resource type. While this certainly has its appeal, the INS does not support interoperation with the DNS.

4.4 Implications

We have described a design for a spatial name system that builds over the DNS, but it is only worth building if it enables a new class of location-aware applications.

Urban device management. Many modern "smart" devices require a remarkable amount of configuration, including many extraneous features [33]. The SNS offers the possibility of separating the management of device functions ("living room light") from the address management of those devices on local networks. Once they start using the SNS, they can be operated locally in an offline-first manner via a direct wireless connection, by resolving their spatial name to a local address that only requires physical proximity. This in turn allows for functionality that requires Internet access to be separately activated, which greatly reduces the attack surface of the device [44]. It may even be possible to support DNS lookups for the non-IP connectivity mechanisms to permit SNS lookups with only physical proximity.

Spatial names for augmented reality. Naming also needs to progress in terms of latency. The emergence of augmented reality headsets brings a demand for real-time spatial name discovery (§1). This requires precise positioning information, both from the headsets and from the environment containing the devices that are being looked at. The SNS provides the protocol framework for devices to advertise their location, and building it over the DNS allows for caching and broadcast-based discovery. With new schemes such as space-filling curves for specifying the precision of a spatial lookup (§3.2), modern headsets such as the upcoming Apple Vision Pro can convert gaze tracking into real-time connectivity to local devices. And since the SNS is just an extension to DNS, these visual connections can also be secure by the use of SNS-based TLS certificates.

Environmental sensor management. On the other end of the spectrum are environmental sensors in remote environments where there is no global access to the Internet (Figure 5).

Figure 5: A camera trap in a Costa Rican rainforest

The SNS simplifies some management concerns: the devices could, for example, sign their readings using certificates issued from the spatial name. It would even be possible to obtain proof of physical access from the device (for example via local audio) and subsequently use that in a global SNS URI to authenticate future remote access to that device (either directly or via delay tolerant protocols [53]).

Any network service can be spatial. Finally, the compatibility of the SNS with existing DNS means that almost any existing applications can become location-aware, simply by naming them appropriately. For example, we could create URIs for spatial locations, and with global names we can also provision TLS certificates for a location using ACME's DNS-01 challenge [7]. We could use federated messaging services to generate virtual entities tied to physical locations. The Matrix protocol [2], for example, uses domain names for server endpoints, and these could now be resolved by location. ActivityPub [3] and any other protocols that use URIs can be deployed just by assigning them to a spatial name instead of a domain name. APIs for a location would be accessible, e.g. to view a camera or brew a cup of coffee. The spatial name, along with a VPN, permits these devices to be managed from afar as well.

5 Conclusions

We have outlined how spatial locations could be represented as an extension to the DNS, enabling a new class of low-latency location-based services to be built that interact naturally with networked devices in the environment. We believe it is possible to make this system backwards compatible with the DNS, globally scalable and privacy-aware, and unlock new networked application domains such as AR headsets.

Acknowledgements We thank Patrick Ferris, Srinivasan Keshav, Magnus Skjegstad and the anonymous reviewers for their insightful feedback and Ian Lewis and Kathy Ho for the sensors. We acknowledge PhD funding from the Huawei Hisilicon Scholarship at Cambridge for the lead author.

References

[1] 1987. *Domain Names - Implementation and Specification.* Request for Comments RFC 1035. Internet Engineering Task Force. https://doi.org/10.17487/RFC1035

[2] 2014. Matrix. (2014). https://matrix.org/

[3] 2016. ActivityPub. (2016). https://www.w3.org/TR/activitypub/

[4] William Adjie-Winoto, Elliot Schwartz, Hari Balakrishnan, and Jeremy Lilley. 1999. The Design and Implementation of an Intentional Naming System. In *Proceedings of the Seventeenth ACM Symposium on Operating Systems Principles (SOSP '99)*. Association for Computing Machinery, New York, NY, USA, 186–201. https://doi.org/10.1145/319151.319164

[5] Roy Arends, Geoffrey Sisson, David Blacka, and Ben Laurie. 2008. *DNS Security (DNSSEC) Hashed Authenticated Denial of Existence.* Request for Comments RFC 5155. Internet Engineering Task Force. https://doi.org/10.17487/RFC5155

[6] Ran Atkinson and S. N. Bhatti. 2012. *Identifier-Locator Network Protocol (ILNP) Architectural Description.* Request for Comments RFC 6740. Internet Engineering Task Force. https://doi.org/10.17487/RFC6740

[7] Richard Barnes, Jacob Hoffman-Andrews, Daniel McCarney, and James Kasten. 2019. *Automatic Certificate Management Environment (ACME).* Request for Comments RFC 8555. Internet Engineering Task Force. https://doi.org/10.17487/RFC8555

[8] Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger. 1990. The R*-Tree: An Efficient and Robust Access Method for Points and Rectangles. In *Proceedings of the 1990 ACM SIGMOD International Conference on Management of Data (SIGMOD '90)*. Association for Computing Machinery, New York, NY, USA, 322–331. https://doi.org/10.1145/93597.98741

[9] Jon Louis Bentley. 1975. Multidimensional Binary Search Trees Used for Associative Searching. *Commun. ACM* 18, 9 (Sept. 1975), 509–517. https://doi.org/10.1145/361002.361007

[10] Stuart Cheshire and Marc Krochmal. 2013. *DNS-Based Service Discovery.* Request for Comments RFC 6763. Internet Engineering Task Force. https://doi.org/10.17487/RFC6763

[11] Stuart Cheshire and Marc Krochmal. 2013. *Multicast DNS.* Request for Comments RFC 6762. Internet Engineering Task Force. https://doi.org/10.17487/RFC6762

[12] S. Cheshire and M. Krochmal. 2013. RFC 6763: DNS-Based Service Discovery. (2013).

[13] Ovidiu Dan, Vaibhav Parikh, and Brian D. Davison. 2022. IP Geolocation through Reverse DNS. *ACM Transactions on Internet Technology* 22, 1 (Feb. 2022), 1–29. https://doi.org/10.1145/3457611

[14] Ian Dickinson, Paul A. Vixie, Christopher Davis, and Tim Goodwin. 1996. *A Means for Expressing Location Information in the Domain Name System.* Request for Comments RFC 1876. Internet Engineering Task Force. https://doi.org/10.17487/RFC1876

[15] Donald E. Eastlake 3rd, Ólafur Guðmundsson, Paul A. Vixie, and Brian Wellington. 2000. *Secret Key Transaction Authentication for DNS (TSIG).* Request for Comments RFC 2845. Internet Engineering Task Force. https://doi.org/10.17487/RFC2845

[16] Ryan Thomas Gibb. 2022. Spatial Name System. (Nov. 2022). https://doi.org/10.48550/arXiv.2210.05036 arXiv:cs/2210.05036

[17] Phillipa Gill, Yashar Ganjali, and Bernard Wong. 2010. Dude, Where's That {IP}? Circumventing Measurement-based {IP} Geolocation. In *19th USENIX Security Symposium (USENIX Security 10)*. https://www.usenix.org/conference/usenixsecurity10/dude-wheres-ip-circumventing-measurement-based-ip-geolocation

[18] John Graham-Cumming. 2014. The Weird and Wonderful World of DNS LOC Records. (April 2014). http://blog.cloudflare.com/the-weird-and-wonderful-world-of-dns-loc-records/

[19] Wesley Griffin and Jakob Schlyter. 2006. *Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints.* Request for Comments RFC 4255. Internet Engineering Task Force. https://doi.org/10.17487/RFC4255

[20] Bamba Gueye, Artur Ziviani, Mark Crovella, and Serge Fdida. 2004. Constraint-Based Geolocation of Internet Hosts. In *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement (IMC '04)*. Association for Computing Machinery, New York, NY, USA, 288–293. https://doi.org/10.1145/1028788.1028828

[21] Antonin Guttman. 1984. R-Trees: A Dynamic Index Structure for Spatial Searching. *ACM SIGMOD Record* 14, 2 (June 1984), 47–57. https://doi.org/10.1145/971697.602266

[22] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. 2002. The Anatomy of a Context-Aware Application. *Wireless Networks* 8, 2 (March 2002), 187–197. https://doi.org/10.1023/A:1013767926256

[23] David Hilbert. 1891. Ueber die stetige Abbildung einer Line auf ein Flächenstück. *Math. Ann.* 38, 3 (Sept. 1891), 459–460. https://doi.org/10.1007/BF01199431

[24] Bob Hinden. 1995. *IP Version 6 Addressing Architecture.* Request for Comments RFC 1884. Internet Engineering Task Force. https://doi.org/10.17487/RFC1884

[25] Paul E. Hoffman. 2023. *DNS Security Extensions (DNSSEC).* Request for Comments RFC 9364. Internet Engineering Task Force. https://doi.org/10.17487/RFC9364

[26] Paul E. Hoffman and Patrick McManus. 2018. *DNS Queries over HTTPS (DoH).* Request for Comments RFC 8484. Internet Engineering Task Force. https://doi.org/10.17487/RFC8484

[27] Xueheng Hu, Lixing Song, Dirk Van Bruggen, and Aaron Striegel. 2015. Is There WiFi yet? How Aggressive Probe Requests Deteriorate Energy and Throughput. In *Proceedings of the 2015 Internet Measurement Conference (IMC '15)*. Association for Computing Machinery, New York, NY, USA, 317–323. https://doi.org/10.1145/2815675.2815709

[28] Zi Hu, Liang Zhu, John Heidemann, Allison Mankin, Duane Wessels, and Paul E. Hoffman. 2016. *Specification for DNS over Transport Layer Security (TLS).* Request for Comments RFC 7858. Internet Engineering Task Force. https://doi.org/10.17487/RFC7858

[29] Bradley Huffaker, Marina Fomenkov, and kc claffy. 2014. DRoP: DNS-based Router Positioning. *ACM SIGCOMM Computer Communication Review* 44, 3 (July 2014), 5–13. https://doi.org/10.1145/2656877.2656879

[30] Brad Karp and H. T. Kung. 2000. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom '00)*. Association for Computing Machinery, New York, NY, USA, 243–254. https://doi.org/10.1145/345910.345953

[31] Ethan Katz-Bassett, John P. John, Arvind Krishnamurthy, David Wetherall, Thomas Anderson, and Yatin Chawathe. 2006. Towards IP Geolocation Using Delay and Topology Measurements. In *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement*. ACM, Rio de Janeriro Brazil, 71–84. https://doi.org/10.1145/1177080.1177090

[32] John C. Klensin. 2010. *Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework.* Request for Comments RFC 5890. Internet Engineering Task Force. https://doi.org/10.17487/RFC5890

[33] Monica Kowalczyk, Johanna T. Gunawan, David Choffnes, Daniel J Dubois, Woodrow Hartzog, and Christo Wilson. 2023. Understanding Dark Patterns in Home IoT Devices. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 179, 27 pages. https://doi.org/10.1145/3544548.3581432

[34] Christian Kreibich, Nicholas Weaver, Boris Nechaev, and Vern Paxson. 2010. Netalyzr: Illuminating the Edge Network. In *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement (IMC '10)*. Association for Computing Machinery, New York, NY, USA, 246–259. https://doi.org/10.1145/1879141.1879173

[35] J. K. Lawder and P. J. H. King. 2000. Using Space-Filling Curves for Multi-dimensional Indexing. In *Advances in Databases*, Gerhard Goos, Juris Hartmanis, Jan Van Leeuwen, Brian Lings, and Keith Jeffery (Eds.). Vol. 1832. Springer Berlin Heidelberg, Berlin, Heidelberg, 20–35. https://doi.org/10.1007/3-540-45033-5_3

[36] Patrick Lazik, Niranjini Rajagopal, Oliver Shih, Bruno Sinopoli, and Anthony Rowe. 2015. ALPS: A Bluetooth and Ultrasound Platform for Mapping and Localization. In *Proceedings of the 13th ACM Conference*

on Embedded Networked Sensor Systems (SenSys '15). Association for Computing Machinery, New York, NY, USA, 73–84. https://doi.org/10.1145/2809695.2809727

[37] Minzhao Lyu, Hassan Habibi Gharakheili, and Vijay Sivaraman. 2022. A Survey on DNS Encryption: Current Development, Malware Misuse, and Inference Techniques. *Comput. Surveys* 55, 8, Article 162 (Dec. 2022). https://doi.org/10.1145/3547331

[38] Anil Madhavapeddy, David Scott, and Richard Sharp. 2003. Context-Aware Computing with Sound. In *UbiComp 2003: Ubiquitous Computing*, Anind K. Dey, Albrecht Schmidt, and Joseph F. McCarthy (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 315–332.

[39] Paul Mockapetris and Kevin Dunlap. 1986. Implementation of the Domain Name System. In *Proceedings of the 2nd Workshop on Making Distributed Systems Work (EW 2)*. Association for Computing Machinery, New York, NY, USA, 1–2. https://doi.org/10.1145/503956.503991

[40] Robert Moskowitz, Petri Jokela, Tom Henderson, and Pekka Nikander. 2008. *Host Identity Protocol*. Request for Comments RFC 5201. Internet Engineering Task Force. https://doi.org/10.17487/RFC5201

[41] Julio C. Navas and Tomasz Imielinski. 1997. GeoCast—Geographic Addressing and Routing. In *Proceedings of the 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '97)*. Association for Computing Machinery, New York, NY, USA, 66–76. https://doi.org/10.1145/262116.262132

[42] Venkata N. Padmanabhan and Lakshminarayanan Subramanian. 2001. An Investigation of Geographic Mapping Techniques for Internet Hosts. *ACM SIGCOMM Computer Communication Review* 31, 4 (Oct. 2001), 173–185. https://doi.org/10.1145/964723.383073

[43] Charles E. Perkins. 2002. *IP Mobility Support for IPv4*. Request for Comments RFC 3344. Internet Engineering Task Force. https://doi.org/10.17487/RFC3344

[44] Jingjing Ren, Daniel J. Dubois, David Choffnes, Anna Maria Mandalari, Roman Kolcun, and Hamed Haddadi. 2019. Information Exposure From Consumer IoT Devices: A Multidimensional, Network-Informed Measurement Approach. In *Proceedings of the Internet Measurement Conference (IMC '19)*. Association for Computing Machinery, New York, NY, USA, 267–279. https://doi.org/10.1145/3355369.3355577

[45] Hanan Samet. 1984. The Quadtree and Related Hierarchical Data Structures. *Comput. Surveys* 16, 2 (June 1984), 187–260. https://doi.org/10.1145/356924.356930

[46] Paul Schmitt, Anne Edmundson, Allison Mankin, and Nick Feamster. 2019. Oblivious DNS: Practical Privacy for DNS Queries. *Proceedings on Privacy Enhancing Technologies* 2019, 2 (April 2019), 228–244. https://doi.org/10.2478/popets-2019-0028

[47] Henning Schulzrinne. 2006. *Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information*. Request for Comments RFC 4776. Internet Engineering Task Force. https://doi.org/10.17487/RFC4776

[48] David Scott, Richard Sharp, Anil Madhavapeddy, and Eben Upton. 2005. Using Visual Tags to Bypass Bluetooth Device Discovery. *SIGMOBILE Mob. Comput. Commun. Rev.* 9, 1 (jan 2005), 41–53. https://doi.org/10.1145/1055959.1055965

[49] Yuval Shavitt and Noa Zilberman. 2011. A Geolocation Databases Study. *IEEE Journal on Selected Areas in Communications - JSAC* 29 (Dec. 2011), 2044–2056. https://doi.org/10.1109/JSAC.2011.111214

[50] Sudheesh Singanamalla, Suphanat Chunhapanya, Jonathan Hoyland, Marek Vavruša, Tanya Verma, Peter Wu, Marwan Fayed, Kurtis Heimerl, Nick Sullivan, and Christopher Wood. 2021. Oblivious DNS over HTTPS (ODoH): A Practical Privacy Enhancement to DNS. *Proceedings on Privacy Enhancing Technologies* 2021, 4 (Oct. 2021), 575–592. https://doi.org/10.2478/popets-2021-0085

[51] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson. 2004. Measuring ISP Topologies with Rocketfuel. *IEEE/ACM Transactions on Networking* 12, 1 (Feb. 2004), 2–16. https://doi.org/10.1109/TNET.2003.822655

[52] Jing Su, James Scott, Pan Hui, Jon Crowcroft, Eyal De Lara, Christophe Diot, Ashvin Goel, Meng How Lim, and Eben Upton. 2007. Haggle: Seamless Networking for Mobile Applications. In *Proceedings of the 9th International Conference on Ubiquitous Computing (UbiComp '07)*. Springer-Verlag, Berlin, Heidelberg, 391–408.

[53] Jing Su, James Scott, Pan Hui, Jon Crowcroft, Eyal de Lara, Christophe Diot, Ashvin Goel, Meng How Lim, and Eben Upton. 2007. Haggle: Seamless Networking for Mobile Applications. In *UbiComp 2007: Ubiquitous Computing*. Vol. 4717. Springer Berlin Heidelberg, Berlin, Heidelberg, 391–408. https://doi.org/10.1007/978-3-540-74853-3_23

[54] Zain Bin Tariq, Dost Muhammad Cheema, Muhammad Zahir Kamran, and Ijaz Haider Naqvi. 2017. Non-GPS Positioning Systems: A Survey. *Comput. Surveys* 50, 4 (Nov. 2017), 1–34. https://doi.org/10.1145/3098207

[55] Martin Thomson, James Winterbottom, and Hannes Tschofenig. 2009. *GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations*. Request for Comments RFC 5491. Internet Engineering Task Force. https://doi.org/10.17487/RFC5491

[56] Robert L. Ullmann, Paul V. Mockapetris, Craig Everhart, and Louis A. Mamakos. 1990. *New DNS RR Definitions*. Request for Comments RFC 1183. Internet Engineering Task Force. https://doi.org/10.17487/RFC1183

[57] Kenton Varda. 2014. Cap'n Proto. (2014). https://capnproto.org/

[58] Paul A. Vixie, Susan Thomson, Yakov Rekhter, and Jim Bound. 1997. *Dynamic Updates in the Domain Name System (DNS UPDATE)*. Request for Comments RFC 2136. Internet Engineering Task Force. https://doi.org/10.17487/RFC2136

[59] Dan Wing, Stuart Cheshire, Mohamed Boucadair, Reinaldo Penno, and Paul Selkirk. 2013. *Port Control Protocol (PCP)*. Request for Comments RFC 6887. Internet Engineering Task Force. https://doi.org/10.17487/RFC6887