Limited Domains Considered Useful

Brian E. Carpenter
The University of Auckland
brian.e.carpenter@gmail.com

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

Dirk Trossen
Huawei
dirk.trossen@huawei.com

This article is an editorial note submitted to CCR. It has NOT been peer reviewed.
The authors take full responsibility for this article's technical content. Comments can be posted through CCR Online.

ABSTRACT

Limited domains were defined conceptually in RFC 8799 to cater to requirements and behaviours that extend the dominant view of IP packet delivery in the Internet. This paper argues not only that limited domains have been with us from the very beginning of the Internet but also that they have been shaping innovation of Internet technologies ever since, and will continue to do so. In order to build limited domains that successfully interoperate with the existing Internet, we propose an architectural framework as a blueprint. We discuss the role of the IETF in ensuring continued innovation in Internet technologies by embracing the wider research community's work on limited domain technology, leading to our key insight that **Limited Domains are not only considered useful but a must to sustain innovation.**

CCS CONCEPTS

• **Networks** → **Network design principles**; *Network protocol design*;

KEYWORDS

limited domains, addressing, routing, IETF, IRTF

1 REASON FOR THIS DISCUSSION

The IP layer is widely seen as an enabler for innovation. It provides a unified API for sending packets from any network location to another, as a common denominator for any application exchanging information with another anywhere in the global IP network. Its semantics are simple: the address system identifies *network interfaces*, assigned by the network domains they attach to, with each domain obtaining part of the global address space to provide connectivity. Packets are sent in a connectionless manner, with the network making its best effort for the packets to arrive. Any IP-based device can rely on this service, which is the very driver of innovation.

But this simplicity and homogeneity is not without critics. For instance, the PlutArch efforts [19] state "*Although the homogeneity imposed by a universal IP layer has provided the scaffolding to develop today's global network, it is now an inhibitor of further innovation.*", while [6] compares use of IP addresses with computer applications binding to disk addresses rather than filenames. Such criticism arises because behaviours and requirements of applications are not as homogenous as the basic IP model makes out. End systems may or may not be mobile. They may require high reliability, or may cope well with loss. Others may struggle with the always-on nature that expects packet delivery to be successful, and yet others envision operation in highly dynamic topologies.

Catering to those varying and sometimes contradictory requirements leads to technologies that are often *limited* in scope, yet may require the global interconnection that IP packet delivery and its deployed system, the *Internet*, provides. We have the *mobile subsystem*, specified by its own standards organisation and deployed at an immense scale alongside the Internet, connecting the many cellular systems that provide a valuable mobile Internet. It has enabled Internet access in airplanes via satellite technologies, and the ever-growing Internet-of-Things, while also providing domain-specific services, such as sensor data collection, cellular radio access, etc.

The authors of [7] recognise this situation with the apt name of *Limited Domain* for those many networks at the edge of the Internet. While providing a definition and characterisation of such limited domains, RFC8799 also conveys the message that they are a *Fact of Life*; they are simply how the Internet as a whole with its connecting networks functions.

This paper pushes the discussion further. We will cycle back to the role of the universal IP layer in facilitating innovation at the ends of a communication. We will position limited domains not only as an artefact of deploying the Internet to ever more edge networks, but as filling a necessary and useful role in the innovation cycle, by fuelling the improvements needed for the whole Internet, to ensure that novel applications will find the Internet to be the best place to meet the requirements of those investing in them.

We will outline a blueprint architecture and its enabling elements that allows for "building limited domains" that are not harmful to their integration into the Internet. We will also discuss the role of the IETF and IRTF in attracting work to identify those technologies that will ultimately need interoperability for a continued innovation cycle in novel applications and supporting communication systems.

2 WHAT IS A LIMITED DOMAIN ANYWAY?

The concept of a *limited domain* was introduced in [7] to capture the trend towards network and end system requirements, behaviours, and semantics that are applied only within a limited region of the Internet. This limited region may represent a physical locality, such as within the same building, campus or even immediate proximity of an end user (e.g., through wearables, immersive devices, etc.). But the limited region may well be distributed across geographies, as an overlay on the Internet or as a parallel network. Before expanding those examples, we briefly and non-exhaustively discuss what drives the emergence of limited domains.

2.1 Drivers

Simplicity may drive limited domains at the edge of the network. Here, simplicity caters to the expected behaviour of those setting up

and maintaining the network. Simplicity is reflected in the resulting behaviour by, for instance, relying on local broadcast solutions (via a single local wireless and wired network) for aspects such as name resolution, address configuration, and multicast, without needing methods typically required in large networks, such as a dedicated naming infrastructure, spanning tree solutions for multicast, etc.

Efficiency may drive changes to naming and addressing, e.g., avoiding long global addresses, or avoiding frequent routing updates due to locator changes in endpoints. Often also, layering is simplified, such as in [15], to optimise node behaviour. Layers may also be conflated, such as in [20], to compress lower layer headers. These measures may be necessary due to constraints in participating nodes, e.g., computational limits or battery life, while entire layers may also be removed for optimization, as proposed in [14].

Delivery of packets may differ significantly from the Internet's best effort model. Instead, deterministic guarantees for latency or capacity are met by domain-specific queuing and resource management methods for, e.g., industrial or interactive applications.

Dynamicity of end nodes, network nodes, and service nodes may vary widely. While Internet communication may be rather stable for desktop software interacting with remote data centres, mobile endpoints, services deployed on mobile endpoints, or even entire networks, such as those in and among flying objects, exhibit an entirely different dynamicity. This may lead to the development of solutions, such as proposed in [28][9], catering to those dynamics, as we can observe in the development of the cellular access system and its range of mobility supporting solutions.

Security of participants, endpoints and nodes in the limited domain is usually very important for the stakeholders. Many enterprise networks are driven by this aspect, imposing stricter security requirements on the participation of end devices and the extension of its network[1]. Many have failed to "just setup a Wifi extension" in an office, falling foul of corporate security rules. These limited domains are often seen as special, secure places of communication — we discuss later the often perceived "cosy nature" of such security, obtained through "crunchy" armour at the edge, e.g., in the form of firewalls, sometimes creating a false sense of security.

Moving tussle boundaries may occur when structural relationships of actors in the system shift, compared to those in the Internet. For instance, systems, such as the mobile subsystem or industrial networks, may have less or no separation between service and network provider, enabling different interfaces (and enabling technologies) to be established for, e.g., steering traffic in the network. The importance of tussle boundaries and interfaces is highlighted in [11], which also emphasises the "*Design for variation in outcome, so that the outcome can be different in different places*"; an apparent advocacy for limited domains.

Restrictions to Internet access manifest themselves in many forms. Enterprise networks may block access to social media, while commercial mobile networks often require age declarations for content access. Entire countries may also regulate content access, such as through the Great Firewall[2] or the EU's proposed "upload filter"[3]). The report in [22] provides a systematic view of properties of communication systems with such restrictions.

[1]The original ARPANET itself experienced this when MILNET split off.
[2]https://en.wikipedia.org/wiki/Great_Firewall
[3]https://en.wikipedia.org/wiki/Directive_on_Copyright_in_the_Digital_Single_Market

Network management approaches may differ in limited domains to compartmentalise faults and to delimit possible legal liability. Authorisation and accountability may be designed to enable the specific legal relationships embodied in the deployment of the (limited domain) system; the mobile subsystem is an example with its usage of the IMSI (International Mobile Subscriber Identity) and the expected tracking of communication relations, enforced through legal obligations towards (wireless) service providers.

Globality is not a driver for the development of a limited domain *per se* but affects our discussion. While a set of networked actors may freely choose their method of communication, a potential need for globality and interconnection to other such domains may position the method as a limited domain in our discussion, if the Internet is chosen to meet the globality requirement.

2.2 Examples

RFC8799 [7] outlines several examples for limited domains, such as (i) locally limited networks (vehicle, home, office, campus), (ii) multi-site but single organisation networks (enterprises, universities, access networks), (iii) special purpose, e.g., sensor, Internet-of-Things, and SCADA (Supervisory Control and Data Acquisition), networks, (iv) data centre networks (single or interconnected multi-site data centres), (v) content delivery networks, and (vi) managed wide area networks for Layer2, e.g., VPN, services.

There are scenarios [7][33] where we see existing Internet technologies used for the realisation of the domain-specific behaviour of participating nodes. Those categories centre on behaviour that drives the realisation of limited domain technologies, such as constrained resources, mobility, traffic steering, service provisioning, communication with enhanced network layer security and with forwarding architectures like Software-Defined Networking (SDN).

These examples show how the development of Internet technologies is often intertwined with the very concept of Limited Domains, discussed below.

3 LIMITED DOMAINS & THE INTERNET

The concept of limited domains was documented in 2020 [7]. We argue here, however, that it has accompanied the development of the Internet from the beginning, often as an origin for technologies that have finally been absorbed by the "mainstream" Internet.

3.1 Historical Review

We spare the reader a lengthy history of the Internet but want to position the development of the Internet in the context of our discussion of Limited Domains.

Early efforts in the 1960s positioned the development of a *data communication system* alongside the dominant voice-centric communication system of the time — public switched telephone networks (PSTNs). The needs of end systems, such as timesharing computers, drove the development of crucial technologies, most prominently *packet switching*. Cornerstones like Baran's and Kleinrock's work on networking using unreliable and redundant network nodes (another crucial expectation of the behaviour of network elements) underpinned what started as the ARPANET in 1969.

Early developments in the Internet introduced many solutions that diverged from the networks it used to transfer data. For instance, it introduced different addressing, a different mode of switching (packet vs circuit switching), and different resource management (endpoint-centric), all of which addressed the needs of the community and stakeholders who drove its development. *Our takeaway is that the Internet started out as a limited domain, with the existing PSTN providing underlying carrier and interconnection capabilities.* This approach continued with extensions to the initial Internet design, e.g., in the form of multicast and IPv6 addresses, with (limited domain) overlays, such as the MBONE and the 6BONE, to deploy, extend and ultimately integrate into the Internet itself.

The Internet did not just set out to implement new behaviours but also the old ones, albeit differently realised. Video and voice transmission is an example of the latter, with technologies like *Voice over IP* disrupting the dominance of the incumbent PSTN by the 1990s[4]. Through the proliferation of fast, e.g., optical, access technologies, VOIP-based technology has long replaced legacy PSTN in many markets[5], while estimates for video delivery over Internet technologies reach up to 70% and more[8] in many markets, radically changing consumer behaviour towards individualised content delivery[6]. *This observation not only positions the historical role of the Internet as a limited domain but also as a driver for innovations that established it as the dominant domain of networking.* We will return to this aspect of the Internet's role in innovation in Section 5.

3.2 Continued Discussions on Limitations

Discussion of limitations of the IP protocol is not new. As early as 1993, efforts started towards the "next generation" of IP[4] that ultimately led to the standardisation of and slow transition to IPv6. International research projects have followed these efforts. Some studied new architectural approaches to address limitations, with NewArch[7] looking to advance addressing and routing (and propose role-based architecture concepts), while PlutArch [19] proposed to break the perceived innovation barrier of a homogenous IP system by a pluralistic architecture, more akin to other parts of society. The 2010s saw revived architectural efforts, including radical "clean-slate" projects, e.g., the US-funded FIND [8] projects, EU-funded projects of the 7th framework programme[9] and efforts in Asia efforts, leading to new technology proposals, while efforts like [32] proposed an architectural framework focussed on innovation by accommodating diversity and evolution.

We observe two aspects important to our discussion here:

(1) There is a continued desire to accommodate new requirements inadequately addressed by existing technology.
(2) Solution proposals and prototypes fall within the notion of limited domains that realise the underlying behaviours.

This positions limited domains as an *enabling concept* for innovation in technologies that may ultimately become part of the Internet. Below, we discuss the *architectural framework* within which to develop such limited domain solutions. We assert that such a framework would provide a coherent basis for facilitating innovation, by accommodating the new behaviour in a limited domain, thereby addressing the main criticism raised by the PlutArch efforts about the constraining nature of the IP design. We first look in more detail at successes and failures of limited domains, to detect the challenges that an architecture must address.

3.3 Successes & Failures

RFC8799 [7] lists several limited domain technologies that have been standardised for the purpose of interoperation. Examples include differentiated [26] and integrated services [3], service function chaining (SFC)[16], data centre network virtualisation overlays [34] and others. Common to those efforts is the clear intention for multi-vendor interoperation, often a sign of success and acceptance of the technology.

Now we look in more detail at successes and failures of entire limited domain concepts to better understand what makes a "good" limited domain and makes others fail, applying the taxonomy for limited domains, provided in [7]. With this, we extend from technologies to entire domains of use of those technologies.

One successful limited domain is the mobile (or cellular) subsystem, specifically its Internet Multimedia Subsystem (IMS)[10]. Its reason for existence is the mobile ecosystem with tight operator control over the multimedia services offered, specifically voice and interactive video[11]. IMS realises requirements at Layer 5 for managing real-time sessions by an adapted version of the IETF's Session Initiation Protocol (SIP) [17]. IMS membership is managed through domain-specific authentication, specifically over-the-air provisioning of service parameters, linked to the IMSI (international mobile subscriber identity) of the user's device, often bundled with the paid cellular service. Initially, mobile nodes joined the system only via their cellular interfaces, but more recent IMS deployments support access over the wireless LAN of a fixed network provider. IMS defines its own gateway model, bounding the limited domain towards the public Internet, including participation in Internet-based SIP peering. The topology of the IMS maps that of the mobile subsystem, with roaming through direct tunnels extending the limited domain from the home to the hosting mobile operator.

Another successful limited domain is that of content delivery networks (CDNs) [29], providing overlaid service nodes on top of a set of hosting operators. Unlike the mobile sub-system, membership in a CDN network is implicitly controlled by access to services being redirected to a CDN instead of providing direct access. However, directly using the indirected service information may lead to "leakage" of information outside the original service's context of use; an aspect addressed by solutions for content delegation.

RFC8799 lists home networks as a typical example of limited domains, but there are aspects of home networking that can be considered failures. One example is the extension to a "smart home

[4]Work in [5] positioned the then relatively new VoIP offerings as disruptors of the incumbent PSTN industry.
[5]In other markets, however, slow access technologies still prohibit change.
[6]The MBONE, mentioned above, was largely aimed at video transmission, precisely emphasising our point.
[7]https://www.isi.edu/newarch/
[8]http://www.nets-find.net/
[9]http://www.psirp.org/, https://cordis.europa.eu/project/id/257217, https://sail-project.eu/, https://cordis.europa.eu/project/id/216041
[10]https://www.3gpp.org/technologies/keywords-acronyms/109-ims
[11]Originally positioned for any real-time service, commercial reality largely limits IMS to voice or video calls.

network", with a plethora of Internet of Things (IoT) devices deployed. To ease setup and management of those devices, particularly the reachability of IoT services from outside the network, Universal Plug and Play (UPnP) is often used for port control of the home's firewall system [2]. The goal here is automation of the setup and the removal of the human user in the management loop, i.e., convenience. Coupled with low security standards at the participating endpoints, however, the result is often malicious connection and misuse of such IoT devices; default passwords and weak diligence of end users are often cited[12] as root causes, albeit enabled by easy opening of firewall security with UPnP technology. Hence, convenience and security concerns are misaligned here.

3.4 Challenges for Limited Domains

We now use the insights from above to outline challenges that need addressing in limited domain technologies.

Semantic leakage: Packet header fields are often input to the forwarding decision. This is not limited to address fields, but may include code points, flow labels and other fields. The semantics of the fields used must be standardised for interoperability. However, limited domains commonly re-interpret them. For instance, work in [24] overrides the IPv6 address with a path identifier for efficient intra-domain forwarding by existing SDN switches. While switches in the limited domain can be programmed for the desired forwarding, any packet leaving the domain would be incorrectly interpreted. Leakage of local semantics must be avoided to prevent havoc, or at best packet dropping, albeit still wasting resources.

Security: We observe a typical security challenge in our smart home scenario, namely the "crunchy outside — soft inside" problem. Despite using a firewall as a hard security boundary, the low standards for securing devices within the network, combined with easily punched holes in the armour of the firewall, lead to adverse effects, such as misuse of resources by botnets.

Membership: Mobile subsystems handle limited domain membership with a separate domain-specific membership management system, linked to a secure enclave within the end user terminal. Any access, including limited domain services such as the IMS, over the cellular connectivity of the domain involves an explicit identification and authentication process. Less safe membership management can be found in home networks, where Wireless Protected Setup (WPS) buttons, or PINs, are increasingly deprecated due to associated risks of unauthorised usage (buttons) or brute force attacks (PINs).

Victim of own success: Limited domains may be aimed at limited deployment, in terms of physical footprint, network size, actors, traffic, or services, but then prove more successful than expected. This can raise previously ignored challenges. Thus, simple security meant for a family home may be unsuitable in small enterprises, and scalability may become a problem. When a limited domain runs out of steam, do escape mechanisms exist to deal with growth, such as nesting limited domains?

Separation of concerns: Limited domains often involve different actors and boundaries compared to other systems, including the Internet. To define an actor model for a limited domain architecture, we must identify all concerns, such as matters of economics and trust, that drive requirements and behaviours, leading to suitable interfaces for the system to work as expected. For example, vertically integrated networks, such as industrial or telecoms networks, often align differently along the service-to-network interface compared to the public Internet. Problems may arise at the interconnection between the limited domain and the Internet; the work in [11] discusses the importance of aligning at such *tussle boundaries*.

Allow for different outcomes: Furthermore, tussles may result in different outcomes in different cases, as postulated in [11][18]. Limited domain solutions may be developed for those different outcomes. Examples include industrial networking scenarios, with tighter service and network layer integration in a market that does not need to separate the network from service provisioning. This may be unacceptable in other parts of the Internet. But original limited domain designs may also encounter new tussles through changing stakeholder desires. Example here is the *smart home*, where the security desire for a limited domain may run counter the desire for ease of use through utilizing (cloud-based) management for Internet-of-Things devices.

4 A RING TO CONNECT THEM ALL

We now outline an *architectural framework* for the development of limited domains. This can serve as a blueprint within which limited domains can realise their specific requirements and behaviour, while enabling interconnection through the evolving Internet.

4.1 Requirements

Section 6 of [7] lists several functional requirements that we use as a starting point, covering domain identification, the nested nature of limited domains, enrollment and withdrawal from limited domains, peer verification, etc. We add the need for mechanisms for self-configuring the network with minimal human intervention.

Requirements for low level domain identification as well as cryptographic authentication are left for further study and are not addressed at the level of an architecture framework. The reasoning is found in [18], which argues that "*the architecture should accommodate variable value sets*" [18], positioning certain stakeholder requirements, e.g., encryption, privacy, or universal access, in the deployment dimension. We will follow this argument in our approach towards a limited domain blueprint.

4.2 Architecture Blueprint

Based on the preceding requirements, our blueprint focusses on four key parts: *addressing* within the limited domain, *maintaining a suitable infrastructure*, including the *membership control and management* aspects allowing participation in limited domains, and *interconnection* with the wider Internet.

4.2.1 Addressing. As noted above, the homogeneity of IP addresses (as network interface identifiers) has been criticised many times. [6] and [33] outline a number of issues with the fixed length as well as the fixed semantics of Internet addressing. As observed in both works, these limitations are circumvented through what is best described as *semantic stacking*, i.e., overloading or extending the IP addressing semantic, by using existing packet headers or adding new ones. The work in [21] proposes a prefix-based approach to

[12] https://www.csoonline.com/article/3127263/iot-botnet-highlights-the-dangers-of-default-passwords.html

structuring IPv6 addresses to support domain-specific semantics. This, however, leads to many well known problems, as [6] and [33] discuss at length. The work at [13] surveys existing techniques, showing their viability and deployment, while [12] discusses the impact of such techniques on routing and forwarding of packets.

The work in [19] suggests the concept of *context* in which addresses are embodied. Context here describes a homogenous semantic within a network region; it is dynamic in that participating nodes may decide to join or leave the context region, i.e., use the addressing context that the region represents. Thus, the region of a specific context maps to the concept of a limited domain.

It seems natural for limited domains to use this concept of context. This allows address semantics to differ between limited domains, since we would otherwise constrain any limited domain to the same addressing semantic as the Internet, squarely falling into the trap of limitation discussed above. Hence, we see support for an arbitrary semantic as key to the viability of limited domains, since a domain's semantic is directly linked to the requirements and behaviours of the domain's stakeholders.

Together with contextual semantics, we also advocate *flexible encoding* of those semantics to remove the constraints of the current fixed length (and limited semantic) Internet addresses that lead to the issues observed in [6][33], while requiring the alignment of semantics at interconnection points to the Internet. We do recognise, however, the tussle between the introduction of flexible encoding and the hardware constraints that have driven the development of ever-faster forwarding silicon.

4.2.2 'Wiring' the Network. A key aspect of configuring a limited domain network is to provide stable control and data plane connectivity for the overall functioning of the domain. Towards this goal, the concept of *Autonomic Networking* [1] envisions functions to self-configure and negotiate parameters across the network. The relevance to limited domains is the focus on stakeholder requirements in the realisation of the domain, aiming to move from human-in-the-loop management, towards incentive alignments, and externalised policy for self-configuration and self-management.

For this, the automatic setup of a secure and resilient infrastructure is imperative: We refer to this as 'wiring the network', requiring mechanisms to set up and manage a working control plane, which in turn can configure and instruct the data plane to send packets according to the semantics of the limited domain. The reference model in [23] provides more details on how to develop functional and protocol specifications for autonomic networks in an architecturally consistent manner.

Key to bootstrapping a self-forming, self-managing, and self-protecting infrastructure is the *Autonomic Control Plane* (ACP), with concepts and methods defined in [31]. An ACP provides virtual out-of-band management to replace traditional in-band management. This foresees the realisation of, e.g., routing protocols, as *autonomic functions* that build upon the ACP's capabilities for discovery and the establishment of authenticated and encrypted peer connectivity. Autonomic functions therefore do not require a working data plane in advance; [25] outlines bootstrapping a remote secure key infrastructure as another crucial element for limited domains.

Autonomic network concepts and solutions extend the proliferation of data and forwarding plane technologies like SDN[13] or P4[14] in that they should remove the need for a pre-configured data plane to provide the desired programmability through a suitable control plane protocol[15]. While the ACP in [31] foresees integration with existing non-autonomic management and control planes, we believe that only a native realisation of an autonomic control and data plane will ultimately allow for bootstrapping entirely programmable and self-configurable limited domain networks.

4.2.3 Membership Management & Control. An important aspect of infrastructure management is membership management for network participants, both end systems and network nodes. The *discovery* and *identification* of members and domains is the key initial step, as also outlined in the ACP work[31].

Assigning *roles* to network nodes is another key aspect, particularly roles defining interaction with other domains. Aligned with the autonomic networking principles discussed above, *dynamic role* assignment should be supported, replacing careful configuration of, e.g., border nodes by automatic discovery and setup, including their relationship to the limited domain (i.e., facing 'inwards' or 'outwards').

Also, mechanisms must exist to ascertain *eligibility* for participating in the limited domain and for nodes to determine which domain they may (or may not) join in a given role. This must be accompanied by *secure enrollment* in the chosen domain[25]. Preferably, enrollment should be dynamic in nature, allowing nodes to flexibly join and leave limited domains, if necessary due to application behaviours or changed user behaviour.

While border nodes, including end systems, have received particular attention in, e.g., the mobile subsystem, using the IMSI (international mobile subscriber identity) for membership management for mobile end systems, membership management for domain-internal network nodes is becoming increasingly important. This is due to the possibility to use general purpose hardware (including end user devices) to instantaneously deploy, e.g., purely SW-based realisations of limited domain technologies. Here, again, autonomic network concepts should be used to aid configuration and bootstrapping of the limited domain, rather than relying on manual, human-in-the-loop, solutions.

4.2.4 Interconnection (global routing). Section 5 of RFC8799 [7] outlines the scope of protocols in limited domains. We pay specific attention to those limited domains that interconnect over the Internet, either transparently or through explicit translation. Such interconnection primarily requires that border nodes have determined the inward/outward facing nature of their interfaces to perform suitable interconnection functions on those interfaces.

Plutarch [19] refers to those interconnection functions as *interstitial functions*, translating the context of one limited domain to and from that of the interconnecting one, with [6][33] outlining a number of methods employed for this purpose, such as compression techniques or proxies.

[13] https://opennetworking.org/sdn-definition/

[14] https://p4.org/p4-spec/docs/P4-16-v1.2.0.html

[15] For instance, Openflow as the control protocol for SDN requires a routable IP in-band infrastructure to exist, positioning the bootstrapping of programmable forwarding actions in SDN as a Munchhausen trilemma.

An important aspect is that of exposing *chaining contexts* to end systems, such as for facilitating choices among context chains (e.g., chaining over the public IPv6 Internet or using an IPv4 transit network as part of the chain). For this, interfaces must exist for conveying context information as well as for instructing interconnection points of the forwarding choice across the chosen contexts - the FARA work [10] proposes *forwarding directives* for this purpose, allowing end nodes to hold multiple *context associations*.

But what layer is best to realise interconnection? Layer 3 seems a natural choice, tasked with 'interconnection' in the common understanding of the network layer, but offering only restricted semantic choices with the most commonly deployed choice (the Internet protocol), as Plutarch [19] so pointedly expressed. Given the wide notion of context, proposals for a more flexible structure of Internet addresses have been made, e.g., in [30], targeting both optimised limited domain addressing and richer semantics across multi-context interconnection networks to reach another limited domain. Alternatively, could the best layer here be simply the application layer with protocols like HTTP or MQTT [27] providing the necessary context translation through service-level chaining? We leave this discourse for another stage, discussed next.

The figure below illustrates key points of our discussion in this paper through a communication from an end system in LD1 (on the left) to another end system in LD1 at the right of the figure as well as to an end system in the public Internet, for which another limited domain (LD2) as well as the public Internet is utilized. Domain-specific *interconnection points* perform the necessary context translation at the boundaries of the individual limited domains.

As noted in our discussion before, the *context* of each limited domain is an important input into that interconnection, while we also see the context being reflected in the *address* of the data packet itself. Encapsulation may be the approach for this, as exercised in many existing approaches, but we can also foresee the use of a flexible addressing capability (as suggested in [30]) as way of accommodating context information; an approach that may be feasible through the advances of forwarding plane technologies.

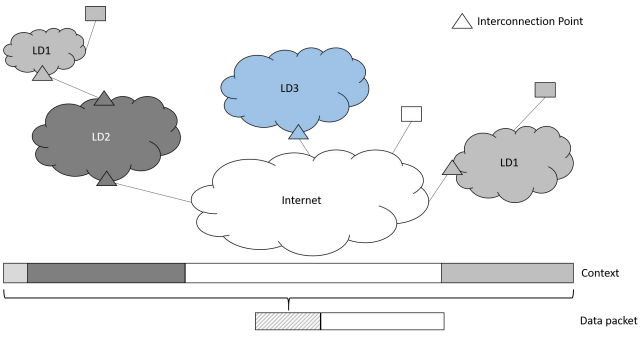

Figure 1: End-to-end traversal of Limited Domains

The figure also highlights another key aspect of our discussion, nmamely the role of the Internet in such interconnection in that the Internet itself can be seen as yet another limited domain in the chain of interconnection albeit possibly (and likely) representing a substantial part of the overall end-to-end communication. Furthermore, it also cycles back to the recognition made in [19] in that simply aligning the context (and therefore the resulting addressing) to single one may be constraining the limited domains in their capabilities when interconnecting via the Internet, therefore possibly hampering the innovation that may have motivated the introduction of the limited domain in the first place. An approach to architecting limited domains and their interconnection must take this possible constraint (and its avoidance) into account.

5 ROLE OF THE IETF

Section 5 of RFC8799 lays out the involvement of standard organisations like the IETF in the development and deployment of limited domains, noting that such standardisation in the IETF's global scope is not always required. However, we can see in a number of limited domain technologies that the IETF has a clear role in driving standardisation, such as for segment routing, service function chaining, and many others. We interpret the existence of such limited domain technologies in the IETF as recognising that *limited domains are simply a fact of life*, both in deployment and standardisation alike.

With this in mind, the IETF's role may *de facto* extend beyond standardising protocols (or extensions) that need interoperation across the whole Internet to include solutions that may only interoperate within a limited domain boundary. Our view of limited domains as incubators for (possible future) Internet technology may further drive this extended scope for standardisation, while considering the recurrent concern that if a limited solution is accidentally or intentionally deployed outside or across a limited domain boundary, it may have harmful results. A strong design constraint is to avoid such harm by construction.

We assert that such avoidance is best accomplished by a deeper architectural understanding on how to build "good" limited domains; an exercise that Section 4 only started and that will need continuation. While the recognition of limited domains is a good starting point, we see the role of the IETF as going beyond standardisation, specifically through its sister organisation, the IRTF. Here, we observe that many limited domain technologies have their origin in the research community, investigating the differing requirements and behaviours that define them. Bringing such work to the IRTF is crucial to sustain innovation that ultimately needs standardisation, while also fostering the aforementioned architectural insights.

6 GOING FORWARD

We outlined in this paper that limited domains are more than just a fact of life. They have been essential since the very origins of the Internet and its progress through the many extensions we have seen initially as limited domain technologies, before standardisation. We see a continued role for standards organisations like the IETF and its research-facing sister organisation, but we strongly believe that we need a clear architectural view that will guide the identification and development of limited domain technologies, which in turn will sustain the innovation of the Internet to come.

This paper has set out an architectural framework as a possible foundation for this architectural view, driven by our insight that **Limited Domains are not only considered useful but a must to sustain innovation!**

ACKNOWLEDGMENTS

Useful comments and hints have come from Artur Hecker, Luigi Iannone, Yihao Jia, and others, who share no responsibility for the resulting text.

REFERENCES

[1] M. Behringer, M. Pritikin, S. Bjarnason, A. Clemm, B. Carpenter, S. Jiang, and L. Ciavaglia. 2015. *Autonomic Networking: Definitions and Design Goals.* RFC 7575. IETF. http://tools.ietf.org/rfc/rfc7575.txt

[2] M. Boucadair, R. Penno, and D. Wing. 2013. *Universal Plug and Play (UPnP) Internet Gateway Device - Port Control Protocol Interworking Function (IGD-PCP IWF).* RFC 6970. IETF.

[3] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. 1997. *Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification.* RFC 2205. IETF.

[4] Scott Bradner and Allison Mankin (Eds.). 1996. *IPng - Internet Protocol Next Generation.* Addison-Wesley.

[5] C. Fine C. Vaishnav. 2006. A dynamic assessment of VoIP innovation, adoption and their interaction with CALEA regulation. In *Technology Policy Research Conference.*

[6] B. Carpenter. 2014. IP addresses considered harmful. In *ACM Computer Communication Review*, Vol. 44. ACM. Issue 2.

[7] B. Carpenter and B. Liu. 2020. *Limited Domains and Internet Protocols.* RFC 8799. IETF.

[8] Cisco. 2020. *Cisco Annual Internet Report (2018-2023) White Paper.* Technical Report.

[9] T. Clausen and P. Jacquet. 2003. *Optimized Link State Routing Protocol (OLSR).* RFC 3626. IETF. http://tools.ietf.org/rfc/rfc3626.txt

[10] A. Falk V. Pingali D. Clark, R. Braden. 2003. FARA: Reorganizing the Addressing Architecture. In *Workshop on future Directions in Network Architecture (FNDA) at ACM SIGCOMM.*

[11] K. Sollins R. Braden D. Clark, J. Wroclawski. 2002. Tussle in Cyberspace: Defining Tomorrow's Internet. In *ACM SIGCOMM.*

[12] A. Farrel D. King, D. Jang. 2021. *Challenges for the Internet Routing Infrastructure Introduced by Changes in Address Semantics.* Technical Report.

[13] D. Jang D. King, A. Farrel. 2021. *A Survey of Semantic Internet Routing Techniques.* Technical Report.

[14] M. Reed M. Al-Naday J. Riihijarvi D. Trossen, S. Robitzsch. 2020. *Internet Services over ICN in 5G LAN Environments.* Technical Report.

[15] C. Gomez, J. Crowcroft, and M. Scharf. 2021. *TCP Usage Guidance in the Internet of Things (IoT).* RFC 9006. IETF.

[16] J. Halpern and C. Pignataro. 2015. *Service Function Chaining (SFC) Architecture.* RFC 7665. IETF.

[17] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg. 1999. *SIP: Session Initiation Protocol.* RFC 2543. IETF.

[18] D. Trossen I. Brown, D. Clark. 2010. Should Specific Values Be Embedded In The Internet Architecture?. In *ACM Conext Workshop on Re-Architecting the Internet.*

[19] R. Mortier T. Roscoe-A. Warfield J. Crowcroft, S. Hand. 2003. Plutarch: An Argument for Network Pluralism. In *Workshop on future Directions in Network Architecture (FNDA) at ACM SIGCOMM.*

[20] V. Jacobson. 1990. *Compressing TCP/IP Headers for Low-Speed Serial Links.* RFC 1144. IETF.

[21] Sun Q. Farrer I. Bo Y. Jiang, S. and T. Yang. 2013. *Analysis of Semantic Embedded IPv6 Address Schemas.* Technical Report.

[22] S. Khattak. 2017. *Characterization of Internet censorship from multiple perspectives.* Technical Report.

[23] B. Carpenter T. Eckert L. Ciavaglia J. Nobre M. Behringer, Ed. 2021. *A Reference Model for Autonomic Networking.* RFC 8993. IETF.

[24] N. Thomos D. Trossen G. Petropoulos S. Spirou M. J. Reed, M. Al-Naday. 2016. Stateless multicast switching in software defined networks. In *IEEE ICC.*

[25] T. Eckert M.Behringer K. Watsen M. Pritikin, M. Richardson. 2021. *Bootstrapping Remote Secure Key Infrastructures (BRSKI).* RFC 8995. IETF.

[26] K. Nichols, S. Blake, F. Baker, and D. Black. 1998. *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers.* RFC 2474. IETF.

[27] OASIS. 2019. *MQTT Version 5.0.* Technical Report. OASIS. https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html

[28] C. Perkins, E. Belding-Royer, and S. Das. 2003. *Ad hoc On-Demand Distance Vector (AODV) Routing.* RFC 3561. IETF. http://tools.ietf.org/rfc/rfc3561.txt

[29] L. Peterson, B. Davie, and R. van Brandenburg. 2014. *Framework for Content Distribution Network Interconnection (CDNI).* RFC 7336. IETF.

[30] Y. Tian X. Gong R. Moskowitz R. Shoushou, D. Yu. 2019. Routing and Addressing with Length Variable IP Address. In *Proceedings of the ACM SIGCOMM 2019 Workshop on Networking for Emerging Applications and Technologies (NEAT'19).*

[31] S. Bjarnason T. Eckert, M. Behringer. 2021. *An Autonomic Control Plane (ACP).* RFC 8994. IETF.

[32] H. Balakrishnan N. Feamster I. Ganichev A. Ghodsi P. B. Godfrey N. McKeown G. Parulkar B. Raghavan J. Rexford S. Arianfar D. Kuptsov T. Koponen, S. Shenker. 2011. Architecting for Innovation. In *ACM Computer Communication Review*, Vol. 41. ACM. Issue 3.

[33] L. Iannone D. Eastlake Peng Liu Y. Jia, D. Trossen. 2021. *Challenging Scenarios and Problems in Internet Addressing.* Technical Report.

[34] L. Yong, L. Dunbar, M. Toy, A. Isaac, and V. Manral. 2017. *Use Cases for Data Center Network Virtualization Overlay Networks.* RFC 8151. IETF.