Providing More Than ‘Just’ Reachability Through Semantic Networking

Dirk Trossen
Huawei
dirk.trossen@huawei.com

Adrian Farrel
Old Dog Consulting
adrian@olddog.co.uk

Daniel King
Lancaster University
d.king@lancaster.ac.uk

Mohamed Boucadair
Orange
mohamed.boucadair@orange.com

Luis Miguel Contreras Murillo
Telefonica
luismiguel.contrerasmurillo@
telefonica.com

ABSTRACT

The Internet has been constantly evolving beyond its original purpose to ensure the reachability between endpoints wherever they may be located. The many technologies developed in Standard Developing Organizations (SDOs) and through proprietary solutions bear witness to this continuous evolution, often driven through specifying and implementing new communication protocols or design practises. Parts of those solutions may overload, amend, or extend packet header semantics at the risk of endangering interoperability of the solutions that make up the Internet system. This discussion paper advocates the Semantic Networking vision, explicitly exposing communication semantics as the essential abstraction for its runtime realization. We present an architecture for Semantic Networking and discuss key design considerations that may inform future research and development work, eventually leading to a new Internet architecture.

1 REASON FOR THIS DISCUSSION

The Internet was initially designed for a single purpose; ensuring *reachability* between communication endpoints connected to the Internet. To ensure reachability, the routing system finds paths through the network to forward packets to their destinations, successfully supporting the Internet’s growth towards a global system with billions of devices, content, domain names, and users. However, throughout the many years of Internet development, other needs have arisen. These include advanced hostname/address mapping, service discovery, identifier/locator separation, optimized resource usage, enhanced security, privacy preservation, network segmentation, isolation and redundancy, differentiated Quality of Service (QoS) processing. Many of these requirements have led to deploying extended, and often distinct, systems, such as Domain Name Service (DNS), firewalls, Deep Packet Inspection (DPI), Content Delivery Networks (CDNs), Distributed Denial of Service (DDoS) scrubbing and mitigators, and many more. Hence, we can observe, similar to the authors in [9], that the Internet’s purpose has evolved beyond mere reachability to accommodate the many purposes stakeholders, including users, progressively demand.

This paper discusses an approach to networking beyond mere reachability, called **Semantic Networking**, where the characteristics and requirements of communication are explicitly encoded as part of the transferred information. As we discuss in Section 2, Semantic Networking has not only been happening for a long time but it is also fundamental to the continued innovation in Internet technologies by its lead users, outlined in Section 3, within a broader set of communication semantics compared to the Internet’s original unicast semantic, as discussed in Section 4. As our core contribution, we describe an approach of Semantic Networking that aims to move beyond developing individual Semantic Networking solutions, each with their associated issues. We outline a new abstraction for Semantic Networking in Section 5, while outlining an architecture and key design considerations for moving towards realizing this architecture in Section 6.

2 ANYTHING NEW HERE?

Extensions to the original reachability purpose are almost as old as the Internet itself, with [6] providing a survey of Internet technologies, while works such as [18, 28] provide a theoretic underpinning for applying broad routing policies to the transfer of packets. Behaviours of such extensions are realized utilizing so-called ‘semantic enhancements’ to addresses and other packet header fields. This approach is referred to as **Semantic Routing** [6], where examples for this approach lead to the conclusion that most routing extensions, in fact, can be seen as an application of semantic routing. However, as also pointed out in individual Internet drafts [5, 6], those solutions are realized in isolation. Their use of specific addressing and packet header semantics, leads to issues such as fragility, complexity, efficiency, etc. [9].

The various current forms of Semantic Routing reflect service providers’ and operators’ requirements and design practices. So, while there is an economic incentive for long term stability and sustainability of the Internet, there is also clearly a desire to accommodate new needs. *Limited domains* [4] are defined as a concept to realize stakeholder behaviours and requirements by changing key technologies, such as routing and addressing, albeit with a limited deployment scope. As observed in [2], the ability to deploy novel methods in limited domains, while still relying on the public Internet for their interconnection to provide global reachability, is

---

FIRA ’22, August 22, 2022, Amsterdam, Netherlands

ACM ISBN 978-1-4503-9328-7/22/08…$15.00
https://doi.org/10.1145/3527974.3546974

often seen as a key driver of innovation, with both [2, 4] providing samples of well-known limited domain solutions.

So if Semantic Routing has already been happening within limited domains, what is new here?

Firstly, as recognized in [9], the development of extensions within the limited set of packet fields and semantics, poses significant risks not only because of extra complexity in managing the deployment of solutions, but also because of the scope of their applicability and the risk of jeopardizing the stability of the Internet. Secondly, the use of semantic enhancements lacks an explicit and prescriptive usage model across design options, which may increase applications' complexity (and likely their usage by end-users). Initiatives like **intent-based networking** [25] attempt to tackle this by allowing the application or an end-user to express an intent to the service provider, which interprets this intent by means of relevant configuration to (dynamically) instantiate (forwarding and routing) policies that will be further enforced by the network. However, the complexity of interpreting an intent to trigger the subsequent provisioning steps may add extra complexity, while inaccurate interpretation may lead to misconfiguration, with the need to update the configuration engine when specifications evolve. Thirdly, the increasing complexity of operating networks may stifle innovation for developing new communication semantics by narrowing the economic forces to those who can afford to handle that complexity, while risking the economically viable operation of the Internet as a stable and assured system.

We believe that a structured approach could overcome some of the aforementioned issues, bringing together advances in intent-based networking, programmable data planes, and novel routing and forwarding solutions in an architecture that consolidates desired communication semantics at runtime; We refer to this approach as ***Semantic Networking*** *in the remainder.*

3 DRIVING THE NEED

As argued in Section 2, the need for some form of semantic enhancements for different traffic forwarding policies has been acknowledged for some time. IP packet transfer provides a common packet delivery system with a basic semantic that ensures end-to-end reachability, including as connectivity between semantic-rich environments. In contrast, current solutions address their specific requirements, to some extent, through inserting additional information in addresses and packet headers at large. Semantic networking bridges this dichotomy by allowing applications to explicitly request specific semantics for data they generate and receive, without negatively impacting other semantically defined forwarding and routing policies, including basic IP delivery.

We see the need for a Semantic Networking architecture driven by the large variety of lead usages and users, which may encourage additional service-specific semantics. This leads to a continued 'pull' towards semantic-rich communication beyond merely reaching another network location. While existing efforts to develop solutions may continue to follow the solution-specific approach they have pursued thus far, Semantic Networking instead proposes a common architecture to realize semantic-rich networking.

3.1 New Lead Usages

New task-centric computing solutions , e.g., proposed by Ray [24], utilize distributed schedulers, together with a distributed and fault-tolerant data store, to steer traffic for applications in an AI/ML-enabled domain. In Semantic Networking, task-specific traffic scheduling could be realized through carrying relevant information in packets and executing Semantic Forwarding actions to enforce semantically-defined traffic steering policies, while dynamically reconfiguring the involved network functions whenever necessary.

One example of such a distributed computing use case is that of *Digital Twins* [12], which represent physical entities (objects, network, systems etc.) as digital equivalents (that is, as entities in a virtual and simulated copy of the network). Digital Twin techniques can be used to assess the efficiency of a network design or a traffic forwarding policy.

While relying on protocols for *Distributed Data Systems* (DDS) [14] in Digital Twin designs, Semantic Networking could provide linkages between data streams at lower levels of the networked system, e.g., for traffic engineering purposes across data streams, while data- and task-specific traffic scheduling could be supported by programming suitable semantics and forwarding behaviours.

Another example of highly distributed services is the Distributed Consensus System (DCS), e.g., realized through Distributed Ledger Technologies (DLTs) [10, 15]. A DCS implements smart contracts for applications such as wallets or cryptocurrencies. Information about these smart contracts, i.e., transactions and ledger information, is typically diffused through a randomized multipoint operation. This diffusion is inherently dynamic in its group of recipients and is currently realized through unicast replications rather than network-level multicast. Furthermore, dynamic and static constraints in selecting IP-based DLT endpoints lead to a 'chatty' communication for the discovery and maintenance of the P2P network, which has been recognized as the source of significant inefficiency [30].

Semantic Networking enables diffusion-based multicast semantics, and the constraints mentioned above may be encoded for performing routing and forwarding decisions that would avoid the cost of current DLT designs, stemming from the inefficiencies identified in [30].

Also, the increasing mobile Internet has been driving new usage patterns, such as applications for mobile (or multi-access) edge computing [11]. Here, developers may take advantage of compute resources located at the network edge, hence closer to their premises. However, the 'closest' compute resource may not be the best one because of, e.g., resource limitations, overload situations, or user mobility. In addition, selecting the 'best' computational resource may need to accommodate application-specific requirements, which in turn need exposure to facilitate compute-aware networking decisions. Semantic enhancements here may include service identification and metric information for semantic-rich decisions to select the most appropriate compute resource.

3.2 New Lead Users

Following the concept of *Limited Domains* [4], emerging usages are increasingly realized in stakeholder and user-specific deployments. For example, *Cellular networks* realize their own communication semantics and solutions to connect to and over the Internet. *Industrial Networking*, with use cases like Digital Twins for smart factories, is another lead use case that is likely to rely upon Internet technologies. *Autonomous driving*, which is set to expand at a 33% annual growth between 2018 and 2028 [27], assumes inter-vehicle communication as well as edge-supported infrastructure services, while other industries like *satellite communications* or *unmanned aerial vehicles* (UAV) raise specific requirements.

4 COMMUNICATION SEMANTICS

Communications can be distinguished according to the *relationship* between senders and receivers and the *selection of the path(s) and endpoint(s)* for the delivery of packets, leading us the following distinct semantics.

The *Unicast* semantic consists of sending a packet from a sender to a single receiver identified by the destination address and possibly enhanced using, e.g., port information.

In *Anycast*, a packet is sent from the sender to any one of a set of receivers. This set may be identified by an address [17] or a content/service name [19, 31].

In *Multicast*, a packet is sent from a sender to all members of a group of receivers.

These relationship semantics can be further constrained through path and endpoint selection semantics:

Multicast relations may be defined as (i) by configuration, (ii) dynamically formed through a membership protocol [3], (iii) through requests towards the sender [29], or (iv) through diffusing towards a sub-group of a larger group, e.g., in Distributed Ledger Technologies (DLTs) [30].

In *Bestcast*, the network applies constraints to determine the best path to the receiver based on the destination address, the state of the network and the compute resources, and information supplied with the packet. Bestcast may also be achieved by extending the anycast address to include multiple virtual unicast representations of the same receiver. The choice of a specific receiver may also determine the network path to reach this receiver. The choice may be made within the network or using a server-based scheduler and a database akin to DNS Resource Records.

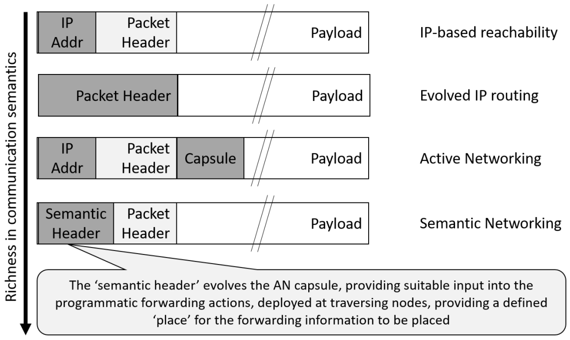
The *Chaincast* semantic steers a packet through a specific set of nodes deduced from the value of the destination address, with typical examples being Service Function Chaining [16] and Segment Routing Network Programming [13].

Multiple optimality criteria may be applied to unicast traffic to select the "best" path to the receiver [21], while the selection between multiple underlying networks or network connections can be made in any of these semantics.

5 ABSTRACTIONS

Key to realizing the communication semantics is the *abstraction* provided at the network level. IPv4 and IPv6 header formats include well-defined source and destination IP address fields that capture unicast, anycast, and multicast semantics.

However, as observed in [6], many extensions to IP have long been developed, extending the abstraction used at the network level to one of the addresses plus additional information encoded in various packet header fields albeit without a well-defined ontology for doing so but following an ad-hoc, solution-driven approach.



The 'semantic header' evolves the AN capsule, providing suitable input into the programmatic forwarding actions, deployed at traversing nodes, providing a defined 'place' for the forwarding information to be placed

Figure 1: Evolution of Abstractions

Active Networking [7] provided a first vision of lifting those ad-hoc approaches onto the level of programmatic extensions, exposed through an in-packet *Capsule*. With this, Active Networking envisioned *'programmable networks that perform computations on the user data that is passing through them.'* [7] albeit limiting the programmability to the semantics defined according to the IP addressing model it was based upon. However, concerns around complexity and security [8] in providing what are possibly arbitrary computations by end users led to this abstraction not being widely adopted.

Figure 1 illustrates the evolution of these abstractions towards Semantic Networking, realizing communication semantics beyond the programmatic approach that Active Networking proposed, by encoding within the packet header the communication semantic as well as the data needed for successful network traversal. Unlike Active Networking, Semantic Networking decouples data and programmatic actions, relying on frameworks like SDN and P4 with a-priori agreed (rather than arbitrary) programmatic actions in the intermediary network elements and placing more emphasis on the management and control planes.

6 KEY DESIGN CONSIDERATIONS

Figure 2 shows a generic functional architecture for Semantic Networking with the components across Application layer, Management Plane, Control Plane, or Forwarding Plane. Many components are familiar or enhanced for semantics purposes, but it is worth detailing some.

The *Intent Service Manager* prepares communication semantics for the *Orchestration Engine* using intents received from the customer's business processes and pre-configured mapping information from the *Ontology* component, the latter providing language transformations that provide mappings from intent to communication semantics, while the *Repository* realizes the storing of such pre-configured or pre-computed mappings from communications actions (as expressed by applications or the Intent Service Manager) to actions that the Orchestration Engine should apply to the network based on a set of constraints and metrics.

The *Network Topology, Capabilities, and State* store contains the most up-to-date record of the managed network and provides a

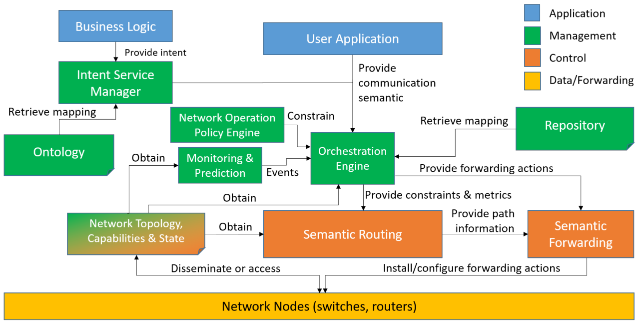


Figure 2: Architecture for Semantic Networking

basis for performance monitoring, fault correlation, route selection, and network orchestration. While the *Semantic Routing* component determines potential paths that satisfy the constraints within the current network state, the *Semantic Forwarding* component is responsible for determining the forwarding actions that should be taken in the forwarding plane and then programming those actions into the forwarding components in the data plane. That is, the Semantic Forwarding component is not the actual packet forwarder, but is similar to an SDN controller that issues configuration commands to the network nodes.

In the following, we outline key design considerations for the main components shown in Figure 2.

6.1 Ontology

Vital to providing semantic-rich communication is a common foundation in the concepts that are used for configuring actions in the nodes of the network. An *ontology* captures those concepts of the Semantic Networking domain. Standardization is a common form of defining an ontology, while advances in Semantic Web and Ontology [26] may be used such that such the ontology can be used programmatically, thereby removing manual configuration tasks. Furthermore, techniques for *semantic mediation* [23] support the translation of concepts that may be defined by different organizations for incorporation into a single networking environment. With this, we postulate the use of techniques that are widely used in enterprise architectures for the management and configuration of Semantic Networking environments.

6.2 Repository

The ontology leads to mapping communication semantics onto concrete sets of *actions*, *constraints*, and *metrics* for the required forwarding behaviour that derives from the communication semantics. This information needs to be provided to the Semantic Routing function, described in Section 6.3, through a *repository*. To do so, (distributed) database solutions (including conventional routing protocols) may be utilized as well as DLTs [10]. The latter may enable using highly distributed repository entities without strong trust requirements, which makes DLTs suitable for emerging communication scenarios, such as drone networking, where a-priori trust is unlikely. However, using DLTs may also pose a challenge for the provider network itself in terms of efficiency and latency, as initially observed in [30], requiring a better understanding of the large-scale efficient use of DLTs.

6.3 Semantic Routing

Due to the demand for emerging applications [1], IP networks are expected to accommodate packet delivery quality beyond current IP-based delivery based on the least-cost path to the destination. Instead, constraints for advanced packet delivery may include throughput, jitter, latency, packet or connectivity loss, but also compute aspects, such as load or compute capabilities. Requirements include relative or absolute guarantees, and predictable elastic changes under contention on these performance factors. This puts significant pressure on network operators as they have to know the relevant information about how individual packets should be treated to meet the expectations of users and applications.

One option for a routing function to utilize richer information is to enforce routing policies within an overlay, e.g., by using IP to route packets between network nodes that can interpret the semantics defined at a higher layer. Several approaches, including Service Function Chaining (SFC) [16] and Information-Centric Networking (ICN) [31] can be used for that purpose. Alternatively, network operators may use techniques to modify the default forwarding behaviour, e.g., by considering heuristics based on packet inspection, on information carried in the packet, or configured via policies into the routers; the latter may utilize advances in multi-optimality routing [18], while using techniques such as those described in [28] to ensure correctness of the intended routing policy. These mechanisms are sometimes called *Semantic Routing* [6], and techniques include *Preferential Routing*, *Policy-based Routing*, and *Flow Steering*.

Operators or service providers may also apply policies to the traffic as it enters the network. These policies may map packet fields (e.g., addresses, DS field, etc.) to specific paths, or result in packets being encapsulated within an additional IP header, as proposed in [19, 32]. In some environments, the destination is located within the source's network, and the network operator can apply locally meaningful Semantic Addressing policies. In other scenarios, a packet will require a path that spans several domains, where each domain may enforce its own traffic forwarding policies. In these environments, there is no consistency or guaranteed traffic performance unless a mutually agreed Service Level Agreement (SLA) is used for traffic crossing the domains.

6.4 Network Topology, Capabilities & State

Techniques to manage network information, e.g., topology, capabilities and routing-related states, must ensure that information is available in a timely manner, while scaling from small to possibly very large networks. An additional problem lies in the constraining of communication relations, following our semantic model outlined in Section 4. With this, the physical network topology information is, in fact, extended into a set of *virtual network topologies*, each of which represents the path and endpoint selections of its governing policy, using methods outlined in [18] to construct such topologies and techniques such as proposed in [28] to ensure their correctness, e.g., in terms of loop freeness.

In a *distributed control plane* model, each network node would have the necessary logic (control plane entity) to communicate with other network nodes, combining resource discovery, reachability, signalling, and connection or link management functions to facilitate Semantic Routing. Here again, DLTs [10] have been proposed, e.g., for the Border Gateway Protocol (BGP) [22] to disseminate network information, while approaches described in [18] have shown that distance vector protocols may be used for multi-optimality routing with semantic-rich constraints. An approach that provides global reachability across domains that are based upon the Identifier/Locator separation is described in [20].

When using a *centralized control plane*, a *controller* interacts with the nodes directly, akin to SDN techniques. While this architecture simplifies the control logic implementation, it may encounter scalability limitations as the size and dynamics of the network increase, while also requiring a suitable controller-node communication protocol to exchange policy information. Furthermore, increasing diversity in service-specific policies may grow the number of virtual network topologies, eventually becoming a problem for a central controller.

A key decision will be the choice of the control plane model, with both having their strengths and weaknesses: central control is conceptually simpler since, as a single point of deployment of policies and business logic, it is easier to deploy and requires less state synchronization. It may, however, present a bottleneck or single point of failure, with latent fault-tolerance issues. A hybrid model may also be feasible, combining the strengths of both models.

6.5 Semantic Forwarding

Forwarding packets in a network involves examining information carried in a packet and using that to determine the interface out of which to send the packet. That determination has historically been a simple look-up of the destination address in a table using longest-first matching.

As discussed in the previous subsections, the forwarding table is installed either by a routing component running on the forwarding device, or from a central controller using a management protocol, utilizing a Semantic Routing algorithm that is aware of the capabilities and state of the network, the desired traffic service behaviours, and the ways that the packets can be marked using additional fields and information carried in the packets.

The network components that perform Semantic Forwarding are instructed about which information to collect from each packet, and what operation to perform to determine the next hop on the forwarding path. The operation could be another simple look-up in the forwarding table using the information fields (or a hash of them) as the key, or it could be a more complex algorithm that is aware of the meaning of the fields as well as the state and capabilities of the network.

Unlike in Active Networking [7], where the forwarding algorithm or function is carried in each packet, in Semantic Networking, the algorithm is installed in the forwarding components for repeated use. That may be achieved a priori (for example, by specifying the forwarding behaviour as a standard and building it into the components), or by distributing the expected behaviour from a controller using a control or management plane protocol.

6.6 Programmable Infrastructure

As outlined in Section 3, new networks must accommodate emerging use cases and application requirements. However, rolling out new functionality incurs high costs and requires new protocols, specialized skills and hardware. In addition, implementing new protocols often increases the network device state requirements and packet processing, wasting valuable memory space, CPU cycles or silicon function features that only support a limited set of protocols. Deploying a more general programmable method and forwarding capabilities would facilitate network flexibility, performance, dynamicity, and efficiency without the protocol overhead and dedicated silicon functions.

Programmable hardware enables several functional packet processing steps, including classification, modification, dropping, and forwarding. In addition, most programmable packet processing systems also provide additional services, such as manipulation, scheduling, filtering, metering, and traffic shaping. These capabilities would instantiate network intent with minimal user input, enforce packet forwarding behaviour and support end-to-end connectivity objectives for specific applications and traffic types.

6.7 Deployment Considerations

Let us briefly discuss considerations for deploying Semantic Network-based systems. Most importantly, a Semantic Networking architecture should enable the evolution of networks to be designed and deployed to better accommodate new services and operational constraints, avoiding a standstill in service innovation. In particular, the interfaces controlling the relevant engines that are responsible for defining and dynamically instantiating semantically-inferred differentiated packet delivery, should not be restricted by design to a set of specific fields, but should instead supported the definition of future parameters and selection criteria.

Also, the architecture should not require that all domains involved in packet delivery must implement the same techniques (let alone adopt the same configuration parameters). Instead, the selection of those techniques should be a local decision, while it is the responsibility of each domain to ensure the appropriate mapping onto inter-domain links.

Works such as [21] outline the use of DC-internal virtual hosting to steer traffic within and possibly across point-of-presence (PoP) data centres. Semantic networking deployments must reconcile these techniques with desires to enable distributed, non-PoP services in future, e.g., 6G, use cases.

Lastly, using explicit signals to identify the nature of communication for any Semantic Networking service may be misused to track the end user or device activity, raising privacy concerns. Means to seek users’ consent should be developed to control whether additional information can be carried in data packets and where such additional information can be shared only with trusted networks.

7 WHY HAVE THIS DISCUSSION NOW?

As described earlier in this paper, the uses of the Internet have developed over the years, and these additional uses have put pressure on the Internet routing system to provide new and enhanced features. This has led to a wide range of technologies being proposed and developed at the application layer, in transport protocols, and within networking.

Semantic Networking has had many realisations through different solutions. These mechanisms have often been limited to specific network environments, technologies, and application demands. Recent developments, e.g., around 5G, cloud networking, distributed computing, and programmable networking have led to a cluster of proposals and ideas for new methods of Semantic Networking.

That means that now is a good time to step back from the individual engineering approaches and to try to understand the many issues and concerns that they introduce.

This can be expressed through generic problem statements, a re-envisioned architecture for routing and forwarding, and potentially a generalised protocol solution that will address the needs of current and future requirements, while providing an interoperable and backwards compatible way of operating the networks of tomorrow.

8 CONCLUSIONS

Adding semantic extensions to existing IP packet delivery allows for richer communication semantics and is driven by needs expressed through new usages and users alike. So far, these extensions have led to a plethora of solution-specific approaches, so far unnoticed from network architects.

We believe that it is now time to explore new architecture models that explicitly recognize the need for semantic-rich communication in a move away from those point solutions. We assert that such models will further accelerate the development of suitable technologies, while ensuring proper architectural evolution. This paper provides a starting point through insights into a Semantic Network architecture, the abstractions as an evolution to existing IP networking, as well as design considerations for its key components.

REFERENCES

[1] ITU-T Focus Group 'Network 2030'. 2019. *Network 2030: A Blueprint of Technology, Applications and Market Drivers Towards the Year 2030 and Beyond.* Technical Report. https://www.itu.int/en/ITU-T/focusgroups/net2030/Documents/White_Paper.pdf

[2] D. Trossen B. Carpenter, J. Crowcroft. 2021. Limited domains considered useful. In *ACM Computer Communication Review*, Vol. 51. 22–28. Issue 3.

[3] B. Cain, S. Deering, I. Kouvelas, B. Fenner, and A. Thyagarajan. 2002. *Internet Group Management Protocol, Version 3.* RFC 3376. IETF. http://tools.ietf.org/rfc/rfc3376.txt

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

[5] A. Farrel D. King, D. Jang. 2022. *Challenges for the Internet Routing Infrastructure Introduced by Changes in Address Semantics.* Technical Report. https://datatracker.ietf.org/doc/html/draft-king-irtf-challenges-in-routing

[6] D. Jang D. King, A. Farrel. 2022. *A Survey of Semantic Internet Routing Techniques.* Technical Report. https://datatracker.ietf.org/doc/html/draft-king-irtf-semantic-routing-survey

[7] D. J. Wetherall D. L. Tennenhouse. 1996. Towards an Active Network Architecture. In *ACM SIGCOMM Computer Communication Review.*

[8] A. D. Keromytis J. M. Smith D. S. Alexander, W. A. Arbaugh. 1999. Security in Active Networks. In *Lecture Notes in Compute Science.*

[9] S. Jiang D. Trossen, Z. Lou. 2022. *Continuing to Evolve Internet Routing Beyond 'Mere' Reachability.* Technical Report. https://datatracker.ietf.org/doc/draft-trossen-rtgwg-routing-beyond-reachability/

[10] Maya Dotan. 2020. Survey on Blockchain Networking : Context , State-of-the-Art , Challenges. 1, 1 (2020), 1–36.

[11] ETSI. 2022. *Multi-access Edge Computing (MEC).* Technical Report. https://www.etsi.org/technologies/multi-access-edge-computing?tab=2

[12] A. Liu F. Tao, H. Zhang and A. Y. C. Nee. 2019. Digital twin in industry: State-of-the-art. 15, 4 (2019), 2016.

[13] C. Filsfils, P. Camarillo, J. Leddy, D. Voyer, S. Matsushima, and Z. Li. 2021. *Segment Routing over IPv6 (SRv6) Network Programming.* RFC 8986. IETF. http://tools.ietf.org/rfc/rfc8986.txt

[14] DDS Foundation. 2022. *What is DDS?* Technical Report. https://www.dds-foundation.org/what-is-dds-3

[15] Juan A Garay. 2017. Basic Properties of the Blockchain. 991, 2016 (2017), 2016.

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

[17] R. Hinden and S. Deering. 1998. *IPv6 Multicast Address Assignments.* RFC 2375. IETF. http://tools.ietf.org/rfc/rfc2375.txt

[18] M. A. Ferreira J. L. Sobrinho. 2020. Routing on Multi Optimality Criteria. In *ACM SIGCOMM.*

[19] R. Khalili Z. Despotovic A. Hecker G. Carle K. Khandaker, D. Trossen. 2022. CArDS: Dealing a New Hand in Reducing Service Request Completion Times. In *IFIP Networking.*

[20] D. Phung S. Secci M. Boucadair, C. Jacquenet. 2019. LISP-MSX: Decentralized Interconnection of Independent LISP Mapping Systems. 57, 1 (2019), 35–41.

[21] V. Giotsas S. Kerola M. Majkowski P. Odinstov J. Sitnicki T. Chung D. Levin A. Mislove C. A. Wood N. Sullivan M. Fayed, L. Bauer. 2021. The Ties that un-Bind: Decoupling IP from web services and sockets for robust addressing agility at CDN-scale. In *ACM SIGCOMM.*

[22] M. McBride. 2022. *BGP Blockchain.* Technical Report.

[23] Ahmed Moujane, Chiadmi Dalila, and Laila Benhlima. 2006. Semantic Mediation: An Ontology-Based Architecture Using Metadata. In *4th International Multiconference on Computer Science and Information Technology CSIT.*

[24] S. Wang A. Tumanov R. Liaw E. Liang M. Elibol Z. Yang W. Paul M. I. Jordan I. Stoica P. Moritz, R. Nishihara. 2018. Ray: A Distributed Framework for Emerging AI Applications. In *13th USENIX Symposium on Operating Systems Design and Implementation.*

[25] Yang C. Chen D. Song Y. Pang, L. and M. Guizan. 2020. A Survey on Intent-Driven Networks. 8 (2020), 22862–22873.

[26] Pablo N. Pirnay-Dummer. 2012. *Ontology and Semantic Web.* Springer US, Boston, MA, 2507–2510. https://doi.org/10.1007/978-1-4419-1428-6_527

[27] Adroit Market Research. 2022. *Autonomous Driving Market by Type...Global Forecast 2018 t0 2028.* Technical Report. https://www.adroitmarketresearch.com/industry-reports/autonomous-driving-market

[28] J. L. Sobrinho T. G. Griffin. 2005. Metarouting. 35 (2005), 1–12. Issue 5.

[29] Dirk Trossen. 2022. *Realizing Forward Requests Return Multicast Semantic with BIER.* Internet-Draft. IETF. Work in Progress.

[30] Dirk Trossen, David Guzman, Mike McBride, and Xinxin Fan. 2021. Impact of Distributed Ledgers on Provider Networks. 935 (2021).

[31] J. D. Thornton M. F. Plass N. H. Briggs R. L. Braynard V. Jacobson, D. K. Smetters. 2009. Networking Named Content. In *ACM Conext.*

[32] S. Gu G. Zhuang F. Li Y. Li, Z. Han. 2021. Dyncast: Use Dynamic Anycast to Facilitate Service Semantics Embedded in IP address. In *Workshop on Semantic Addressing and Routing for Future Networks at IEEE Intl. Conference on High Performance Switching and Routing.*