Bounding the router table size in an ISP network using RINA

John Day*, Eleni Trouva §, Eduard Grasa§, Patsy Phelan‖, Miguel Ponce de Leon‖,
Steve Bunch**, Ibrahim Matta*, Lubomir T. Chitkushev* and Louis Pouzin‡‡

*Computer Science, Boston University, Massachusetts, USA, day@bu.edu, matta@bu.edu, ltc@bu.edu
§i2CAT Foundation, Jordi Girona, Barcelona, Spain, eleni.trouva@i2cat.net, eduard.grasa@i2cat.net
‖TSSG, Waterford Institute of Technology, Ireland, pphelan@tssg.org, miguelpdl@tssg.org
**TRIA Network Systems, LLC, Illinois, USA, steve.bunch@ieee.org
‡‡Eurolinc, Paris, France, pouzin@enst.fr

***Abstract*—One of the biggest problems of todays Internet is the explosion of the size of the routing tables of Internet core routers, especially due to the growth of multi-homed hosts and networks. This paper explains the benefits that the Recursive InterNetwork Architecture (RINA) brings to network service providers in terms of routing scalability: with an appropriate design the size of the router tables can be bounded. The recursive layer approach, the independence of the address space at each layer in conjunction with the use of hierarchical addressing prove to be effective tools that greatly reduce the storage requirements of routers as well as speed up the calculation of routes, resulting in more efficient and scalable routing.**

I. Introduction

RINA is an Internet architecture proposed by John Day in his book "Patterns in Network Architecture: A return to fundamentals" [1]. RINA leverages many of the "lessons learned" by previous network architectures and brings them one step further by identifying that networking can be seen as a set of recursive layers that provide distributed inter-process communication services over different scopes. The resulting architecture is surprisingly simple compared to today's protocol complexity and provides a structure that allows network designers to solve the problems identified in the current Internet. A complete description of the architecture and its features can be found in [1], [2], [3]. In this paper our goal is to consider some of the advantages of RINA for provider networks. In particular, we will consider how the architecture improves routing efficiency. RINA provides the ISPs with a tool to bound the number of routes to be stored and allows for reduced storage requirements, both in number and in length of the stored routes.

II. A provider network in RINA

Large corporate or provider networks are generally organized into a hierarchy of subnets. RINA is able to leverage this structure to considerable advantage.

A. A typical configuration

Fig. 1 depicts the basic structure that a provider network might have in RINA. In Fig. 2 the same configuration of a provider network is illustrated, in which we draw the DIFs formed, showing the ability of the architecture to provide configuration over the different scopes. Starting at the left of Fig. 2, there is a top-level DIF, named T-DIF, which covers the span of the network. This T-DIF can provide end-to-end service to the hosts located at the edges of the network. The T-DIF is supported on the left by a lower DIF with a scope consisting of the host and the first router, a typical media-specific or data link layer. The T-DIF functions like the traditional legacy architecture until the left border router. The border router determines the next hop and multiplexes the traffic to the lower layer DIF that encapsulates traffic in a new routing domain in a lower DIF (L-DIF), where traffic is routed in the normal way. To the T-DIF, the L-DIF is a single hop. The T-DIF has no knowledge of the routing decisions made by the L-DIF. The border routers can be viewed as managing a set of flows across the "hole. This top-level DIF is comprised of a ring of subnets around a central hole. After crossing the backbone, traffic is popped up a layer to the T-DIF and is routed normally within the T-DIF to the destination. This structure can repeat indefinitely to compose a provider's network. DIFs comprising the provider's network might correspond to metro areas, regions, countries and eventually the backbone of the network.

An idealized view of the configuration shown in Fig. 1 from the top would be similar to the drawing depicted in Fig. 3. Basically, there is a "necklace" of subnets around a central "hole". The subnets are not distinct DIFs but subsets of the DIFs. They might for example signify areas with the same address prefix in a hierarchical addressing scheme, as we will see in next. This configuration has several interesting properties. Next we provide the implications of such a configuration for routing but before that we need to explore the nature of addresses in RINA.

B. The nature of addresses in RINA

In RINA all address spaces are private. Public addresses are merely a form of private. The address spaces for these provider DIFs belong to the provider and hence only need sufficient scope to accommodate the providers DIFs. Hence the addresses can be (and should be) shorter. With a relative

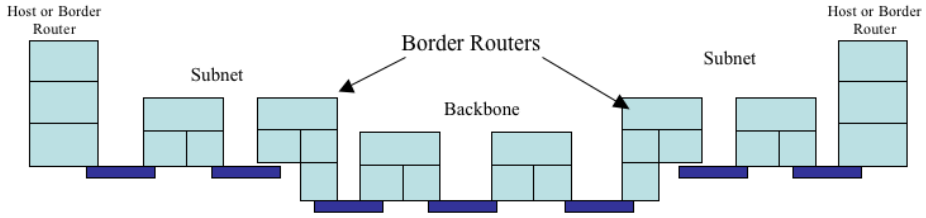

Fig. 1. A possible configuration for a provider's network in RINA

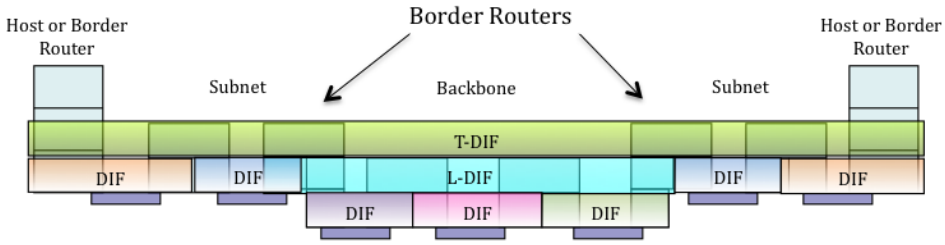

Fig. 2. RINA provides for a distinct configuration of different parts of the network that exhibit different characteristics and traffic patterns. The L-DIF can be used to configure the backbone part of the network, while the T-DIF for the provider's network.

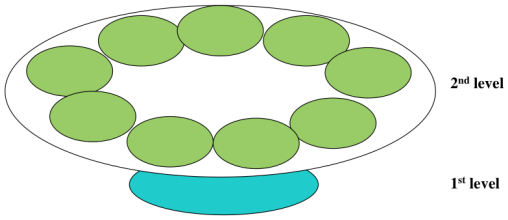

Fig. 3. Top view of a typical set for provider networks. The top rings illustrate subnets that construct provider networks and the ring below the backbone of the network.

architecture, the scope of DIFs can be designed to bound the number of elements in a DIF and hence the load imposed by routing. Consequently, the addresses length is reduced, which results in having smaller routing table entries and allows for faster routing calculations. To communicate among providers, there may be peering DIF with an address space that covers multiple providers. However, there is another approach that might be used in which DIFs are created spanning multiple providers.

Although it is not a requirement of the architecture itself, the use of a hierarchical addressing scheme will result in notable improvements in routing efficiency. Hierarchical addressing schemes aim to reduce the routing information in large networks. In a hierarchical addressing schema, one portion of the address indicates the "piece" of the network in which the destination resides, and another portion of the address distinguishes destinations within that piece. Hierarchical addresses can be chosen in such a way that they reflect the topology of the network. Moreover, a distance function that gives us the distance between nodes using their addresses might exist. Here distance is defined as the number of hops between two nodes in a DIF but any other nearness metric could also be used.

C. Routing in a provider network

RINA allows considerable efficiencies in routing in a provider network both in the lengths of the routes and the number of routes that must be calculated. This not only requires less storage for routes, but also reduces the time to compute the routes. In some cases, forwarding table updates can be accomplished without a route calculation at all.

For the configuration that we introduced in Fig. 2, if we adopt a flat addressing strategy in the T-DIF, the number of routes to be stored will not be reduced. This way we will only reduce the length of the routes that are calculated because the hole is a single hop. If a hierarchical addressing strategy is adopted so that all the nodes in the same subnet around the hole have the same prefix and nodes in adjacent subnets have longer common prefixes, i.e. the address hierarchy reflects the adjacency of the subnets, then we can significantly reduce the number of routes we need to store. The routing in the T-DIF would only have to store routes to the border routers of either adjacent subnets or the hole. A border router at the edge of a hole can determine where to forward a Protocol Data Unit (PDU) based on the address only. So, no route calculation is necessary. Routes only need to be computed within the subnet. This drastically reduces the number of routes and the length of the routes that need to be stored and effectively allows a network designer to bound the number of routes at each level. In this case, the number of routes to be stored is determined by the number of elements in a subnet, not the number of elements in the network.

As an example, consider the larger network depicted in Fig. 4. Again, this is an idealistic view from the top for a possible configuration in RINA of a larger provider network. It is similar to the configuration depicted in Fig.1 but this time

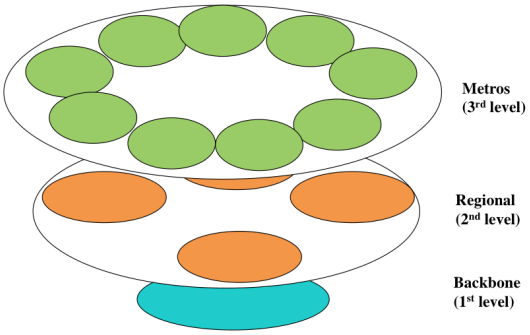

Fig. 4. A large provider might have multiple layers of DIFs to manage scaling and resource allocation.

the IPC processes running in the different systems are forming three layers of DIFs. The network consists of a backbone, a regional and a metropolitan tier. Each tier has several subnets noted in the figure as circles. Table I shows a comparison considering two cases, when the addresses are hierarchical and when not for each level of the provider network. For both cases we calculate the number of routes to be stored in a routing table and give a worst-case approximation of the number of hops, considering the length of the route equal to the diameter of the network for each tier. We assume that the subnets making up the necklaces of the two DIFs (metro and regional areas) have a diameter of D and the backbone has a diameter of 2D.

What becomes apparent from the results shown in Table I is that the use of hierarchical addresses allows the number of routes to be stored vary with the number of border routers or hosts of a single subnet as opposed to a non-hierarchical schema, where the number of routes is analogous to the number of border routers or hosts in the entire layer.

In order to explain further how hierarchical addressing can be applied in RINA, we give an example in Fig.5. The figure displays only the regional level (2nd tier) of the network illustrated in Fig.4, however hierarchical routing can be used for the other levels of the provider network accordingly. We note that the address spaces of different levels are completely independent from each other, since as we already mentioned, the addresses in RINA are internal to the DIFs. The circles in the figure denote IPC processes running in the systems (routers in the specific example) of the network and the dotted lines the links between the systems. The processes having an up arrow are processes belonging to border routers that forward the PDUs at a layer above, while the ones having a down arrow are processes belonging to border routers that forward the PDUs at a layer below. All the processes of the second tier have addresses with a common prefix (3.x.x). The second portion of the address denotes the subnet in which the router is located. For example, 3.2.x are addresses assigned to the processes of the second subnet. The suffixes of the addresses in our example are arbitrary and enumerate the processes within a subnet. In the figure we give three examples of forwarding tables of the nodes with addresses 3.1.1, 3.2.4 and 3.3.2 to show how a similar configuration achieves a reduction in the number of routes that need to be stored. The forwarding table contains entries in the format of destination address - next hop address and it is derived from the routing table of each node, in which the complete path to the destination addresses is stored. As an example we can consider the case of the first subnet and the forwarding table of the process with address 3.1.1. All the processes in the first subnet are assigned addresses of the form 3.1.x. There are three processes inside the subnet that are running in border routers able to forward a PDU to a higher layer. These are the 3.1.1, 3.1.6 and 3,1,7. There is one process running in a border router able to forward a PDU to a lower level, assigned with the address 3.1.3. In the figure we can see the forwarding table for the process assigned to address 3.1.1. There is an entry for each one of the other subnets in the network, aggregating all the addresses of a subnet in a single entry. E.g. for the second subnet, an entry exists with destination address 3.2.x and next hop address 3.1.2, which is the shortest route to the border router addressed 3.1.3 that connects the first subnet to the second one. In addition the forwarding table in 3.1.1 contains entries with destination addresses of the processes running in the border routers of this subnet that can forward PDUs to the level above. In our example, these are 3.1.6 and 3.1.7. In total the forwarding table of the process 3.1.1 contains 5 entries. If we had chosen to use a flat addressing schema instead of a hierarchical, the forwarding table in 3.1.1 would have to have one entry for each border router able to forward a PDU to a higher layer of the entire tier. In this example these are 10 entries (3.16, 3.1.7, 3.2.1, 3.2.6, 3.2.7, 3.3.3, 3.3.6, 3.4.1, 3.4.3 and 3.4.5).

From this simple example, we see that at the very least the storage requirements for routes can be greatly reduced and if addresses are assigned appropriately the number of routes can be significantly reduced and with, appropriate network design, bounded.

III. Why RINA

The several benefits accompanying the RINA architecture makes us believe that RINA is the best choice for the next generation service provider networks. The IPC model provides the following advantages to network service providers:

Scalability RINA scales with no upper bound over any range of users, resources, bandwidth, or distance. Any limitations are in the physics, not in the architecture. This scaling is enabled by the recursion of the layer. Requirements for router computation and storage capacities are structurally reduced. As we analyzed in this paper, router table size is orders of magnitude smaller, and bounded. Router capacity is dramatically increased by our ability to aggregate flows and cap the number of flows per layer. Furthermore, the scaling is flexible. There is no requirement that a layer have no more than 100 or 10,000, or any other number of nodes, as long as it meets its operating requirements. This is facilitated not only by the recursion but by the use of topological addresses as well.

TABLE I

	Non-hierarchical		Hierarchical	
	Number of routes to be stored	Maximum number of hops	Number of routes to be stored	Maximum number of hops
Metros - DIF 3	n	$2Dn$	$m + (s-1)$	$D(m + (s-1))$
Regionals - DIF 2	$n_2 - 1$	$2D(n_2 - 1)$	$(m_2 - 1) + (s-1)$	$D((m_2 - 1) + (s-1))$
Backbone - DIF 1	$n_1 - 1$	$2D(n_1 - 1)$	$n_1 - 1$	$2D(n_1 - 1)$

The table shows the number of routes to be stored in a routing table and a worst-case approximation for the number of hops when using hierarchical addresses and when not, for the network depicted in Fig. 4. D is the diameter of each of the subnets in the metro and regional tiers, $2D$ the diameter of the backbone, s is the number of subnets in the current level, n is the number of hosts, m is the number of hosts in a single subnet, n_2 is the number of border routers in the second level that can forward PDUs to the level above, m_2 is the number of border routers of a single subnet in the second level that can forward PDUs to the level above and n_1 is the number of border routers in the backbone. Note that $m \leq n$.

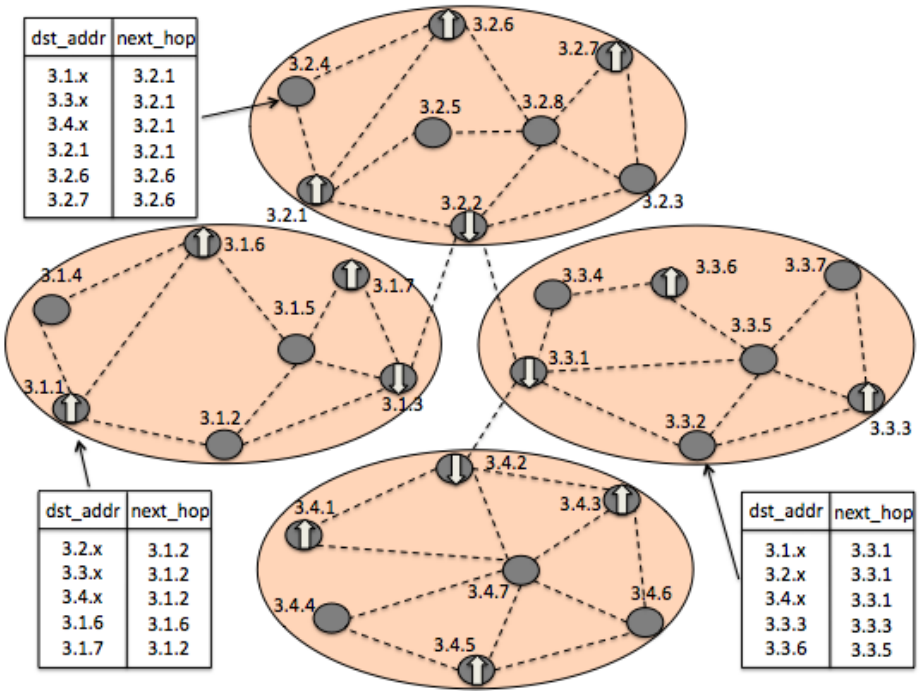

Fig. 5. An example of hierarchical addressing for the second level (regional) of the network depicted in Figure 4.

Greater Robustness and More Effective Response to Change Response to change is faster thanks to load balancing, quicker convergence due to much smaller routing table sizes, more responsive flow management, and, on the manpower side, simpler and more effective operational management. RINA provides all of the flexibility and survivability of connectionless networking while supporting all the service capabilities of connection-oriented networking. Data flows under hostile environments are far more reliable due to highly tunable, situation-specific policies.

Congestion Control The most important contributor to the effectiveness of a congestion control scheme is time to notify, i.e. the largest component of the time to react. As average network diameter increases, the effectiveness of any congestion control scheme will decline. With RINA by putting congestion control in lower DIFs, not only can time to notify be shortened, even bouned, but the effects of the congestion can be localized to a single DIF in a specific part of the network.

Full Support for Multiple Classes of QoS RINA provides multiple classes of QoS independent of applications. We take a more analytical approach that makes it simpler to accommodate new classes of QoS. Our approach, along with the recursive structure of our architecture, provides a solution that leverages the burstiness of traffic rather than covering it up. Layers aggregate flows into higher bandwidth flows. Each layer manages flows in a given bandwidth range, multiplexing them onto higher bandwidth flows in lower layers. Thus, the number of flows to be managed at a given layer can be bounded or even held constant. Resource allocation becomes more efficient and scalable. We also provide the means for service providers to collaborate on providing QoS without divulging sensitive information about their networks, removing a major barrier to interoperation. We also provide the means for service providers to collaborate on providing QoS without divulging sensitive information about their networks,

removing a major barrier to interoperation. There is a range of long-sought services that conventionally require significant overbooking of capacity, or separate networks. These include voice and video, including multicasting of both, and teleconferencing. RINA makes these capabilities available without special provisioning as QoS can be appropriately enforced at each DIF (layer). In this way we avoid the huge inefficiencies that occur with over-provisioning.

Multihoming and Mobility are a part of the architecture as they are completely supported by the architectural structures and topological addresses; no special protocols nor mechanisms are required and no special burden is placed on the routers. Mobility is equivalent to dynamic multi-homing and reduces to updating changes in the addresses of protocol state machines to reflect their position as they move with respect to the topology of the layer/subnet. There are no mobility-related scaling problems as with the conventional, centralized solutions.

Applications are able to operate on whatever layer has sufficient scope to reach all their correspondents. For example, corporate applications may operate on top of a layer whose scope is limited to their corporate network, while the corporation's website may operate on a public global layer. Corporate applications would thus be invisible to the outside world. This has the effect of Network Address Translation (NAT) without explicitly requiring NATs. That is, the meaning of NAT is transformed; since every layer has its own address space, NATs are an integral part of the architecture. Provider networks have their own address spaces on which these organizational networks (layers) float. Messaging, peer-to-peer, mail relaying, and transaction processing are essentially instances of a DIF with different policies and concrete syntaxes. Proxies and caching are part of normal layer operation, i.e. relaying and routing, with appropriate policies and parameters.

RINA allows for a seamless transition One of the many benefits of RINA is that no migration to the new architecture is required. Adoption is the way RINA can spread, as it can be used over and under the layers of the current stack, as well as along with the current infrastructure. RINA can be used over IP, under IP, or along side IP. Deployment can begin with one pair of devices and proceed by adding one device at a time. There is a range of adoption strategies. The first and simplest deploys DIFs under IP, much as MPLS is deployed today. This allows the network to more effectively manage flows and provide better performance to users. Second, a common layer that emulates a Sockets API could be used to encapsulate existing applications to provide them with some of the benefits of layer operation without modifying legacy applications. This could take two forms: a pure Sockets API that maps to the DIF with policies and parameters that make it look like TCP, or a Sockets API with options that would allow it to accept directives and thus enable the DIF to provide an improved service and in turn enable the applications to make better use of the DIF. This would require only minor changes to the application. In either case, the number of legacy applications is sufficiently small that the layer could be aware of what the applications were and use policies appropriate to those applications. Third, new applications could use an API that directly accessed the layer's capabilities, to obtain all the benefits described here. Furthermore, a RINA structure could be wrapped around TCP/IP to yield some (but not all) of the benefits of RINA. This structure could exist in a subnet and interact transparently with traditional TCP/IP systems without them being any the wiser. And of course, a dual-stack approach could be continued used as long as desired to internetwork legacy applications over TCP and legacy applications over a DIF. This would be no more difficult than current dual stack/NAT transition plans. However, we do not foresee this as common, since it creates security problems.

IV. Conclusions, On-going and Future Work

We have developed a breakthrough advance in Internet architecture, with all the characteristics required for quick adoption. Rooted in fundamental insights into the behavior of networks, RINA is an extension of existing technology and new insights that lead to a complexity collapse. It has a straightforward operational model that solves the major problems in Internet infrastructure, such as, as we unfold in this paper, the exponential growth of the router tables. RINA enables critical, long-sought services, modes of use and operating characteristics, and improves security. It reduces time and cost of network engineering, development and operations. Finally, RINA has a straightforward adoption path that permits internets and maintenance of legacy equipment value.

It appears that all of the architectural elements are in place to accommodate IPC, i.e networking, what primarily remains now is to explore the policies and configurations for specific forms of DIFs. Our architectural attention has been forced (by the problem) to consider in more detail the implications of "If a DIF is a Distributed Application that does IPC, what is a Distributed Application?". Meanwhile, our current work includes the development of a prototype that implements RINA and the refinement of the current specifications [4]. Future work includes experimentation with the developed prototype over networks of different physical media and further research, measurements and performance comparisons of RINA versus the current Internet architecture.

Acknowledgment

This work has been partially supported by the Spanish Government, MICINN, under research grant TIN2010-20136-C03. The work of Ibrahim Matta and John Day was supported in part by NSF grant CNS-0963974.

References

[1] J. Day, *Patterns in Network Architecture: A Return to Fundamentals*, Prentice Hall, 2008.

[2] J. Day, I. Matta and K. Mattar *Networking is IPC: A Guiding Principle to a Better Internet*, Madrid, Spain: Proc. of ReArch08, 2008.

[3] E. Trouva, E. Grasa, J. Day, I. Matta, L. T. Chitkushev, P. Phelan, M. Ponce de Leon and S. Bunch *Is the Internet an unfinished demo? Meet RINA!*, Prague, Czech Republic, TERENA Networking Conference, 2011

[4] Pouzin Society website, http://www.pouzinsociety.org/