Manifesto of Edge ICT Fabric

A. Manzalini, R. Minerva
Telecom Italia
Telecom Italia – Strategy - Future Centre
Innovative Architectures
Turin, Italy
antonio.manzalini@telecomitalia.it

E. Dekel, Y. Tock
IBM Research - Haifa
Haifa, Israel
dekel@il.ibm.com

E. Kaempfer
Intel Corporation
Communications and Storage Infrastructure Group
Phoenix, United States
ernest.kaempfer@intel.com

W. Tavernier, K. Casier, S. Verbrugge, D. Colle
Department of Information Technology (INTEC),
Ghent University-iMinds, Belgium
{wouter.tavernier, koen.casier, sofie.verbrugge,
didier.colle}@intec.ugent.be

F. Callegati, A. Campi, W. Cerroni,
Department of Electrical, Electronic and Information
Engineering "G. Marconi" and CIRI-ICT
University of Bologna
Bologna, Italy
{franco.callegati, aldo.campi, walter.cerroni}@unibo.it

R. Vilalta, R. Muñoz, R. Casellas, R. Martínez
Centre Tecnològic de Telecomunicacions de Catalunya
(CTTC)
Optical Networks and Systems Department
Castelldefels, Spain
ricard.vilalta@cttc.es

Noel Crespi
Institut Mines-Telecom, Telecom SudParis
Paris, France
noel.crespi@mines-telecom.fr

Nicola Mazzocca, Elisa Maini
Department of Electrical Engineering and Information
Technology University of Naples Federico II Naples, Italy
Via Claudio, 21 80125
elisa.maini@unina.it

***Abstract*—Technology advances are making available huge amounts of processing, storage, networking capabilities at the edge (i.e., up to End-Users premises) of current networks. It is argued that these trends, coupled with new emerging paradigms such as Software Defined Networks, will impact deeply the evolution of future networks, allowing to design highly flexible architectures at the edge capable of creating a galaxy of new ICT business opportunities. This paper presents this vision by proposing a so-called "manifesto of Edge ICT Fabric": the sheer number of nodes, devices and systems being deployed at the edge, up to Users' premises, will create an ICT fabric offering an enormous processing and storage power. Using this Edge ICT fabric, which is closer to the Users, for executing network functions and services will bring several advantages, both in term of improved performance and cost savings (e.g., determined by the removal of middle-boxes). It is argued that incentives, cooperation and competition at the edge will boost the long-term value of networks: like in ecosystems, where evolution select the winning species, winning services will succeed, grow, and promote further investments, while losing ideas will fade away.**

Keywords-component; Edge Networks, SDN, NfV, Standard Hardware, Future Networks.

I. INTRODUCTION

Today communication networks include a range of deployed middle-boxes [1] such as WAN optimizers, NAT, performance-enhancing-proxies, intrusion detection and prevention systems, any sort of firewalls, other application-specific gateways, etc.

Each middle-box (typically closed and quite expensive) supports a narrow specialized function (layer 4 or higher) and it is mostly built on a specific hardware platform. Middle-boxes are deployed along most paths from sources to destinations: that's why the Internet lost its initial simple end-to-end forwarding principle. As a matter of fact middle-boxes contribute today to the network ossification, but also represent a significant fraction of the network capital and operational expenses (due to management complexities).

This paper proposes the following vision. Technological advances (e.g. standard h/w performance, embedded communications, device miniaturization, etc.) and the related costs reductions are progressively moving an incredible amount of processing, storage, communications-networking capabilities at the edge of traditional networks, i.e., which will

978-1-4799-0980-3/13/$31.00 ©2013 IEEE

be in the hands of the end Users (and by Users it is meant not only people but also machines, smart objects, appliances and any device which is attached to the network at the edge). Following such trend, end Users will be more and more empowered to "drive networks and services states" dynamically.

Furthermore, SDN (Software Defined Networks) [2] and NfV (Network functions Virtualization) [3] are creating the conditions to reinvent the network architectures, improving dramatically the flexibility through virtualization and programmability.

According to SDN proposition, the control and data planes are decoupled, network intelligence and state management are logically centralized, and the underlying network infrastructure is abstracted. Complementarily, NfV is proposing that network processing functions to be developed in software, that can be instantiated and moved in various locations in the network.

Figure 1 is providing a scenario where the core part of the network is becoming (almost) stateless, whilst the execution of the stateful network functions is moved to the edge networks and to the Data Centers. Network functions are executed by processing resources (e.g., standard hardware) crossed by the traffic.

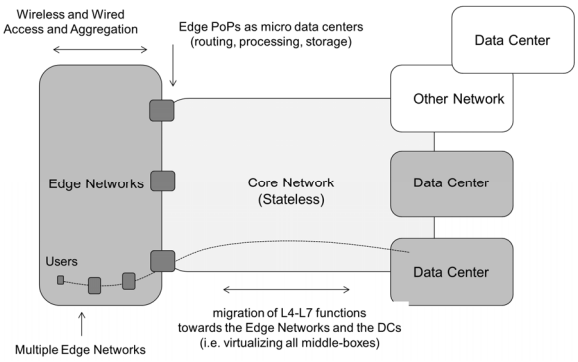

Fig. 1. Evolutionary Network Scenario: Stateful Edge Networks

The outline of the paper is the following. Section II provides some examples of use cases. Sections III focuses on the architectural aspects. Section IV makes a brief summary of the state-of-the-art. Section V elaborates about the techno economic impact and the business aspects. Finally, Section VII gives conclusions and discusses future work.

II. EXAMPLES OF USE CASES

This section describes three simple use-cases: what they have in common is the vision of the exploitation of a distributed network processing architecture at the edge of current networks. In particular, it is argued that the sheer number of nodes, devices and processing, storage systems being deployed at the edge, up to Users' premises, are offering an enormous processing and storage power. Using these resources, closer to the Users, to execute network functions and services will bring several advantages, not only in terms of flexibility but also in performance improvements.

The requirements of these three use cases (and other ones under investigations) are offering interesting technical challenges which have to be solved to transform this vision into reality. Some of these requirements and challenges are listed in the following sections.

A. Personal Data and Services following Users

This use case concerns the possibility that Users' personal data (e.g., stored in edge micro Data Centers) and services (e.g., executed by local edge resources) are following the Users when they are moving from one network attachment point to another one, even when they are moving across different edge networks (and as such the Core network has to be crossed).
In other words, it should be feasible to move data and VM executing services seamlessly with no impact on QoE perceived by the Users. It should be possible, also, for security or other policies to follow logically specific network applications (e.g., running on VMs).
It should be possible even to federate data and services associated to Users in order to build distributed virtual data centers at the edge (this is an ideal service, for example, for Universities, Enterprises, etc.), provided at costs which are a small fraction of traditional cloud computing services.

B. Harnessing idle resources at the edge

Several start-ups (Symform P2P Cloud Storage Platform represents a recent example) are starting offering storage and disaster resilience services using decentralized and distributed virtual resources, shared by Users. The concept of harnessing idle resources could be extended also to the processing idle power distributed at the edge, up to the Customers' premised. This would require the capability of properly orchestrating said local idle storage and processing resources when executing and provisioning network functions and other services. Examples of provisioned services will be CDN-like services, content sharing, aggregation, transformation, data collection, etc.

C. End-to-End Services across different Edge Networks

A Service Provider may want to provide end-to-end services to Users who are attached to edge networks belonging or operated by different Network or Infrastructure Providers. This means that it should be possible even hooking and orchestrating heterogeneous network resources and functions for the provisioning of end-to-end services.

III. OVERALL ARCHITECTURE

This paper argues that future network infrastructures will be composed by a stateless and low-complexity Core Network (i.e., all middle-boxes will be removed from the Core) interconnecting very dynamic Edge Networks (ENs) capable of executing – in orchestration with Data Centers (DCs) - all the stateful network functions and services. This will bring the Core Network – as seen from the ENs - back to initial Internet e-2-e paradigm, i.e. just forwarding packets.

In [1] a number of types of middleboxes are reported. Examples are Caches, Content and applications distribution boxes, Application-level gateways, Application Firewalls, etc.

Key assumption is that network functions and services will be executed in ensembles of Virtual Machines (VMs), allocated and moved across a scalable distributed platform deployed at the edge (e.g. from CPE to Edge PoPs nodes) and orchestrated with Data Centers resources.

This distributed platform (based in standard hardware) will be characterized by high flexibility, performance and self-adaptation at run-time (e.g. dynamic flocking of resources according to needs), harnessing and combining all unused resources (e.g. computing and storage power at end Users' home and in the edge micro data centers). It will be possible programming, allocating and moving a variety of virtual architectures (spanning across diverse edge networks or even across today DCs) on-demand, based on Users' requests, governance and biz requirements overcoming the current ossified networks structures. End Users will have access to a certain number of abstraction for programming, setting-up and tearing down, migrate and optimize their network functions and services (e.g. local traffic engineering, failure handling policy, local topology optimization, etc.) according to their need and service level agreement.

A. Network States

The edge will become a sort of Cloud or better a Fog of Virtual Machines (VMs) which are typically stateful and somewhat related one-another. The collection of the information related to the state of the Edge ICT Fabric resource capabilities, related usage (i.e. states) and data flows set-up, is crucial to support the global orchestration..

The semantic of the data collected, the syntax of the communication method used to collect them, and the storage strategy have to be defined. Such data must be presented in an abstracted way, to allow a general and technology-independent approach to any further use and processing.

In the Edge ICT Fabric approach there is an original challenge regarding the data collection and state analysis (DCSA), stemming from the fact that the network nodes are now VMs and therefore their state is also influenced by the overall POP DC conditions. The DCSA must take into account that the overall system conditions are not simply described by the state of the network elements but also by the state of the system hosting the VMs. Therefore the DCSA must have general communication capabilities, enabling cross boundaries communication between POP DCs, and general semantic capabilities to describe the state and condition of resources of very different nature and spanning different logical layers of the Edge ICT Fabric architecture.

Here we envisage an approach in line with previous works proposing approaches applicable to the management of state information spanning cross-layer in systems where the network and IT resources are strictly interacting [4], [5]. For instance, the Network Resource Description Language proposed in [6] can be used to describe and cross-correlate data regarding the network and the IT resources in a unified way, enabling automated and cognitive application service provisioning as demonstrated by the test examples reported in [7], [8], [9].

B. Scalable Standard Hardware Nodes

Current communication networks are built on very specialized network elements with bespoke hardware and software architectures specifically developed to provide the best performance/cost ratio for the particular function they are meant to represent. This results in an exceedingly heterogeneous selection of network elements where each node can only provide the function it was originally designed for and any new service that requires new functionality typically necessitates the deployment of new specialized types of hardware.

IT processes today are embracing a very different paradigm where services are deployed as Virtual Machines running on standard hardware resources, either on site or in the cloud. This scheme allows an IT organization to respond much faster to evolving needs and to reach higher usage rates of its computing resources.

The idea of extending this flexibility to Telecommunications networks is getting a growing attention, even more when looking at a deeper integration of processing, storage and networking resources. This is still lacking today as networks are mostly based on specialized hardware nodes.

Until recently, processor-based network equipment was typically designed around a heterogeneous system concept, in which the networking control plane ran on one processor architecture (e.g., x86 architecture processors) while the data plane executed on a different architecture, such as a multi-core MIPS platform, with specialized network acceleration features. In order to benefit from the level of performance attainable with such a system, one would accept the additional inflexibility resulting from heterogeneous software architecture as well as the complexity associated with the integration and maintenance of two different code bases. Clearly, this is not an ideal solution, and a unified system architecture, in which the control plane and data plane run on the same processor architecture while achieving the necessary cost-performance targets, is preferable for several reasons: scalable node development, development and integration is simplified; processor resource utilization is improved because the control plane and data plane can be distributed among cores with greater flexibility; and software maintenance is much easier with a common code base and a single programming environment.

Virtualization enables operation of multiple virtual machines (VMs), each containing a guest operating system (OS) and its associated applications, on the same physical board. The coexistence of multiple OSs is made possible by a software layer, known as a virtual machine monitor (VMM) or hypervisor, that abstracts the underlying processor cores, memory, and peripherals, and presents each guest OS with what appears to be a dedicated hardware platform. The hypervisor also manages the execution of guest OSs in much the same way that an OS manages the execution of applications.

Virtualization also provides a transition path for enabling innovative new designs while maintaining legacy applications. Virtualization makes the migration easier by allowing the

legacy applications to run in a shadow environment alongside new code, allowing compliance testing of the new code in real time.

The ability to support legacy and new code on a single platform can also ease the regulatory compliance burden in Telecom applications. Telecom Equipment Manufacturers (TEMs) can add new features on one VM, while a legacy application runs unchanged in its own VM. Because the legacy applications run unmodified, it is significantly easier to re-certify them.

TEMs can also look to virtualization to help them take advantage of multi-core processors. Many systems require re-use of legacy code written for a single-core processor. Reworking this code for multi-core execution is often impractical. In the Telecom industry, most TEMs have significant investments in validated, single-threaded code. This code is typically irreplaceable, and in some cases only exists in binaries that cannot be modified. Virtualization makes it possible to run multiple instances of this software on the same processor, each within its own VM, leading to significant improvements in cost, power, and size.

C. *Global Controller*

The Global Controller, based on the service requests and the Edge ICT Fabric states, has to compute how to direct traffic flow, routed by nodes, through these computing elements for additional processing (e.g. VMs) to execute the required services (e.g. L4-L7 middle-box network functions).

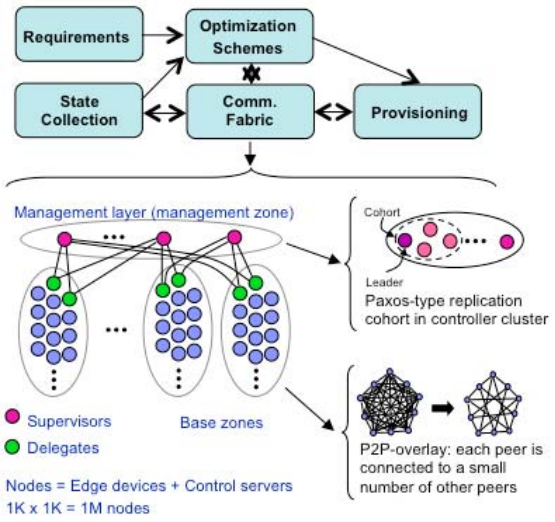

Fig. 2. Global Controller

Efficient methods are required to create optimal said paths and Edge ICT Fabric-resource allocations under continuously changing conditions. The latter involves: i) situations in which moving users need that required data and services/network functions are moving with them, as well as ii) situations of elastic service requests where resources need to scale up or down to meet the customer's demand.

These types of problems require efficient and dynamic re-allocation mechanisms for both network and processing resources (involving path computation/traffic engineering).

In addition it is necessary to design and implement scalable and dynamic discovery, control, and communication framework. This framework should enable the controller to exert its control over the edge devices, and facilitate state collection.

The discovery function is the infrastructure which the Global Controller uses to get an up to date image of the available edge assets. The discovery mechanisms will allow edge devices to join the resource pool and register their assets (i.e. resources, data and services) as candidates for utilization. Object registration and resource discovery products will be stored in a distributed data store.

The framework will implement scalable and efficient communication channels that will allow the Global Controller to manage and coordinate the edge devices. For this purpose we envision communication forms like a scalable membership service, an attribute replication service, as well as message queues, scalable publish-subscribe, and converge cast services.

An additional scope of this framework is to support the implementation of a robust controller cluster that is cloud ready, consistent, and immune to failures.

The framework will be distributed and highly elastic, in order to serve the uniform growth from small scale to large scale systems. It will be based on peer-to-peer technologies that will instantiate overlay communication topologies congruent with the varied network topologies that the edge devices are embedded in.

D. *Service Provisioning*

Service provisioning will require methods for efficient service provisioning of networking and processing resources in distributed settings. The crucial challenge of this issue is to translate higher-level constructs or primitives resulting from resource allocation/optimization mechanisms into lower-level instructions and configurations.

Provisioning logic can be split into two parts: i) primitives for individual data plane and processing elements of Edge ICT Fabric, and ii) primitives for coordinated provisioning of several distributed Edge ICT Fabric. This will enable programmability of forwarding functions, middle-box functionality such as firewalling, Deep Packet Inspection, or more advanced services such as for content delivery (CDN). For these purposes, the provisioning functionality could rely and further build on the communications framework provided by the Global Controller.

A provisioning program could consist of a set of higher-level primitives describing network and resource configurations. This program can be executed by a run-time environment which translates them into lower-layer instructions such as OpenFlow, middlebox primitives, etc. Part of the program remains running in the run-time environment to react to events which lower-level elements such as network switches or middle-boxes are not able to handle (for example packets for which no (Open-)flow entry can be found, or events which cannot be processed due to lack of resources). The provisioning program can react on these events by, e.g., installing new rules, or reserving additional resources. The

role of the run-time environment is to ensure that these high-level provisioning program abstractions are correctly translated and executed into lower-level actions. Such an architecture will allow provisioning of elastic services on top of distributed Edge ICT Fabric, based on allocation schemes designed for the Global Controller.

IV. State of the Art

This section will just focus on the state of the art concerning the development of the Global Controller, which is recognised as the key architectural element.

The evolution of clustering and group communication in enterprise systems exemplifies the challenges posed by these developments. Group communication is essentially concerned with who is in the group (membership) and reliable communication between group members (for a good survey see [10]). As systems grew in size, protocols evolved from master-slave replication, to synchronous replication in cohorts, and then to protocols that allow asynchronous progress, such as Virtual Synchrony [11], [12].

Those group communication protocols employed strong consistency semantics, which limited their scalability. As systems grew in size even further, peer-to-peer techniques and gossip protocols were introduced, relaxing consistency semantics, yet increasing the scale by an order of magnitude [13], [14], [15], [16].

In an effort to increase scalability even further, and reach Internet scale deployments, pee-to-peer overlays were stacked hierarchically [17], allowing Census [18] for example, to support membership of 10000 nodes. The advent of cloud computing and the increase in size and complexity it represents brought P2P techniques to the doorstep of almost every large scale enterprise system [19], [20].

Chubby [21] and ZooKeeper [22] represent another approach for managing large distributed systems. Here a distributed hub is deployed, and all other group participants connect to it and coordinate their actions through it. Chubby and ZooKeeper were designed for coordinating data-center based distributed systems and provide strong consistency guaranties. ZooKeeper builds upon a restricted variant of Paxos [23][24], but has the crucial drawback of using a static member set. FRAPPE [25] improves upon that drawback by implementing a high performance version of reconfigurable Paxos.

One of the fundamental communication services that needs to be offered for efficient group communication is multicast – the ability to send a message from one member to a subgroup.

A popular abstraction for that service is publish-subscribe (pub/sub) messaging [26]. Pub/sub systems may be centralized or distributed; for scalable setups, generally distributed versions are employed [27]. Distributed pub/sub is generally based on overlay networks [28], [29], [30], [31], [32], and has gained traction in major enterprise systems [33], and even HPC environments [34].

V. Prototyping and Experimental Activities

Figure 3 shows the architecture of an Edge PoPs which is under development by using standard hardware (e.g., x86) and open source software, properly enhanced. This type of node can be seen as a sort of micro Data Center at the border between the core and the edge networks (e.g., a future evolution of today edge routers).

As mentioned, the Global Controller is orchestrating the local controllers by using historic and real-time network data: these data have to be elaborated to create sort of maps needed to orchestrate virtualized network functions and services in a reliable and efficient way.

In other words the Global Controller should be capable of handling the "network states", which are stored in the Network Big Data (NBD).

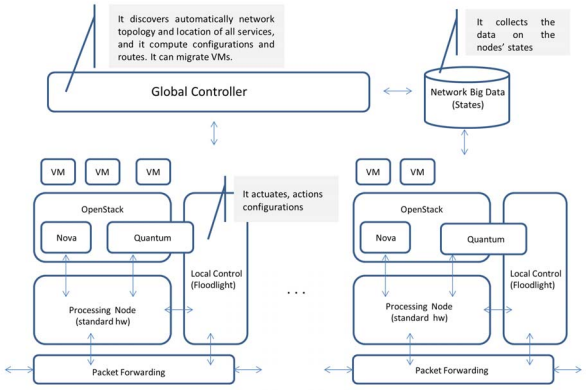

Fig. 3. Example of Edge PoP nodes, Global Controller and NBD

NBD provides a sort of infrastructure map representing the states of the resources and the network connectivity. NBD stores, filters and processes historic and real-time data collected over time (through monitoring points) and refreshed continuously. For example network performance are monitored at a high frequency, for immediate response to faults and degradations. NBD provides also actual maps used to direct end Users to the best Edge PoPs to respond to their requests.

The development of the nodes prototypes will be followed by a set of experiments both in "local" test-beds and in a distributed environment developed by interconnecting some local test-beds.

Test-beds will be interconnected through VPN connections (e.g., L2 tunnels for control and data interconnection) in order to demonstrate challenging features such as VMs live migration across WAN.

VI. Socio Economic Issues and Business Models

In the near future the edge will look like a distributed network processing architecture, deeply integrating processing, storage and networking resources. This will impact dramatically future networks evolution. Technology and business developments will be more and more strictly intertwined in the future. Obviously, a certain technology will be adopted not only if it is advantageous (reducing costs) and

trusted but also if it will be able to enable desired business ecosystems (with the related foreseen business models); on the other hand, newly designed potential ecosystems will look for enabling solutions and technologies capable to bring them into reality.

The introduction of this increased functionality at the edge opens up opportunities to execute network functions and applications closer to the customer increasing his quality of experience, offloading network traffic and device functionality. Additionally this will increase the reach of applications allowing an even smoother deployment of high end applications on a worldwide scale with the constraints mainly imposed by the Edge ICT Fabric. Of course the advent of new applications and the success of the functionality installed at the edge will be depending on the economics of the applications installed and running on this edge. Indeed, the impact of the price for using the functionality at the edge will be an important aspect here. However the ecosystem and potentially additional incentives and pricing schemes could prove at least as important.

Installing the functionality at the edge involves new investments in both software and hardware and as well in the training and equipment at the developer's side. For a developer to step into this approach, he will also need to have a clear view on the costs associated to using each of the new functionalities at his disposal. Setting the price from a provider point of view will be a task involving typical investment modeling, simulation and analysis. Checking the profitability of an application developer will use this pricing in a dedicated investment analysis.

As already touched upon, the use of ICT functionality at the edge will most probably have an impact on the load of the access, metro and core network as well as on the device constraints at the customer side. It will open up a huge field of opportunities to develop and distribute applications and open the market at this point to new players. It is mandatory to tackle any economic calculation in such context in a multi-actor manner in which the full business model and ecosystem is taken into account in the calculation of one actor's business case. Adding this business model view on top of the previously mentioned detailed investment analysis for edge ICT and application providers is the key to understanding the impact of edge ICT and creating a successful business model for it.

VII. CONCLUSIONS

This paper has proposed a so-called Manifesto of Edge ICT Fabric, which is a vision arguing that the recent advances in standard hardware technologies and open source software are creating the conditions for exploiting highly innovative network architectures, mainly at the edge of current networks.

As a matter of fact, the sheer number of nodes, devices and systems being deployed at the edge, up to Users' premises, are offering an enormous processing and storage power. Using these resources, closer to the Users, to execute network functions and services will bring several advantages, both in term of improved performance and cost savings (e.g., determined by the removal of closed middle-boxes in the Core Network).

The paper has described three use cases whose study has allowed the derivation of a number of high level requirements. Starting from this, an example of architectural concept has been also proposed which will be prototyped by using open source software and standard hardware.

From a management viewpoint, it will be important solving the problem of orchestrating ensembles of VMs executing network functions and services. The problem has two facets: VMs placement and traffic routing between VMs (moving of VMs is allowed, locally and globally, but it has a cost). Placement, move and routing decisions should aim at minimizing "network costs" while achieving "close-to-optimal performance" in executing network functions and services. This means solving "live" (by using the NBD) a double "constrained optimization problem".

From the economic and business viewpoints, the paper has also argued that the exploitation of these principles at the edge of current networks will transform this area into a fertile ground for the flouring of new ICT ecosystems and business models. Future steps in this direction includes also modeling and simulating diverse cooperation-competition business strategies for Telco-ICT ecosystems at the edge.

REFERENCES

[1] IETF RFC3234 "Middleboxes: Taxonomy and Issues", February 2002;

[2] White paper "Software-Defined Networking: The New Norm for Networks", Open Networking Foundation;

[3] White paper "Network Functions Virtualisation", ETSI;

[4] A. Campi, F. Callegati: "Network Resource Description Language." In Proc. IEEE GLOBECOM Workshops, pp. 1–6. Nov. 30 2009–Dec. 4 2009;

[5] Chinwe Esther Abosi, "Towards a Service Oriented Framework for the Future Optical Internet", PhD Thesis, School of Computer Science and Electronic Engineering, University of Essex, April 2011;

[6] F. Callegati, A. Campi, W. Cerroni, D. Simeonidou, G. Zervas, R. Nejabati: "SIP-enabled OBS Architectures and Protocols for Application-aware Optical Networks." Computer Networks, vol. 52, no. 10, pp. 2065–2076, July 2008. doi:10.1016/j.comnet.2008.02.016. Invited Paper;

[7] F. Callegati, A. Campi, W. Cerroni: "Automated transport service management in the future Internet: concepts and operations." Journal of Internet Services and Apllications, vol. 2, no. 2, pp. 69 – 79, 2011. ISSN 1867-4828. doi:10.1007/s13174-011-0026-y;

[8] F. Callegati, W. Cerroni, A. Campi: "Application scenarios for cognitive transport service in next-generation networks." Communications Magazine, IEEE, vol. 50, no. 3, pp. 62 –69, march 2012. ISSN 0163-6804;

[9] B. Martini., Cerroni W., Gharbaoui M., Campi A., Castoldi P., Callegati F. Integrated Signaling Framework for Joint Reservation of Application and Network Resources for the Future Internet. In: IEEE Global Telecommunications Conference. Houston, 5-9 Dec. 2011, p. 1-5;

[10] Chockler G., Keidar I.., and Vitenberg R., "Group communication specifications: a comprehensive study," ACM Computing Surveys, vol. 33, no. 4, pp. 427–469, 2001;

[11] Birman K. and Joseph T., "Exploiting virtual synchrony in distributed systems," in SOSP'87, 1987;

[12] Distribution and Consistency Services (DCS), IBM;

[13] van Renesse R., Minsky Y., and Hayden M., "A gossip-style failure detection service," in Middleware'98: Proc. of the IFIP Int'l Conf. on

Distributed Systems Platforms and Open Distributed Processing, 1998, pp. 55–70;

[14] Ganesh A., Kermarrec A. M., and Massoulie L., “Peer-to-Peer Membership Management for Gossip-Based Protocols” IEEE Trans. Comput., vol. 52, no. 2, pp. 139–149, 2003;

[15] Allavena A., Demers A., and Hopcroft J. E., “Correctness of a gossip based membership protocol,” in PODC ’05: Proc. of the annual ACM Symp. on Principles of Distributed Computing, 2005, pp. 292–301;

[16] Gupta I., Chandra T. D., and Goldszmidt G. S., “On scalable and efficient distributed failure detectors,” in PODC’01: Proc. of the annual ACM Symp. on Principles of Distributed Computing, 2001, pp. 170–179;

[17] Ganesh A. J., Kermarrec A. M., and Massouli´e L., “HiScamp: self-organizing hierarchical membership protocol,” in Proceedings of the 10th workshop on ACM SIGOPS European workshop, ser. EW 10, 2002, pp. 133–139;

[18] Cowling J., Ports D. R. K., Liskov B., Popa R. A., and Gaikwad A., “Census: Location-Aware Membership Management for Large-Scale Distributed Systems,” in USENIX, 2009;

[19] Rodrigo Rodrigues and Peter Druschel, Peer-to-Peer Systems. In Communications of the ACM Vol. 53 No. 10, Pages 72-82;

[20] Ken Birman, Gregory Chockler, Robbert van Renesse: Toward a cloud computing research agenda. SIGACT News 40(2). June 2009;

[21] Burrows M., "The chubby lock service for loosely-coupled distributed systems," in Proceedings of the 7th symposium on Operating systems design and implementation, ser. OSDI, USENIX Association, Berkeley, USA, 2006;

[22] Hunt P.,Konar M.,Junqueira F. P., andReed B., "ZooKeeper: wait-free coordination for internet-scale systems," in Proceedings of the 2010 USENIX conference on USENIX annual technical conference, ser. USENIXATC'10;

[23] L. Lamport. The part-time parliament. ACM Trans. Comput. Syst., 16(2):133–169, 1998;

[24] [SCCF15] L. Lamport. Paxos made simple. ACM SIGACT News, 32(4):18–25, December 2001;

[25] [SCCF16] V. Bortnikov, G. Chockler, D. Perelman, A. Roytman, S. Shachor, and I. Shnayderman, "FRAPPE: Fast Replication Platform for Elastic Services", LADIS 2012;

[26] Eugster P. T., Felber P. A., Guerraoui R., and Kermarrec A. M., "The many faces of publish/subscribe," ACM Comput. Surv., vol. 35, no. 2, pp. 114-131, Jun. 2003;

[27] Hosseini M., Ahmed D., Shirmohammadi S., and Georganas N., "A survey of Application-Layer multicast protocols," IEEE Communications Surveys & Tutorials, vol. 9, no. 3, pp. 58-74, 2007.

[28] Roie Melamed, Idit Keidar: Araneola: A scalable reliable multicast system for dynamic environments. J. Parallel Distrib. Comput. 68(12), 2008;

[29] Gregory Chockler, Roie Melamed, Yoav Tock and Roman Vitenberg, Constructing Scalable overlays for pub-sub with many topics, in PODC 2007;

[30] Gregory Chockler, Roie Melamed, Yoav Tockand Roman Vitenberg, SpiderCaast, a scalable interest aware overlays for topic-based pub/sub communication, in DEBS 2007;

[31] Sarunas Girdzijauskas, Gregory Chockler, Ymir Vigfusson, Yoav Tock and Roie Melamed, Magnet: Practical subscriptions clustering for Internet-scale publish/subscribe. In DEBS 2010;

[32] Castro M., Druschel P., Kermarrec A. M., and Rowstron A. I. T., "Scribe: a large-scale and decentralized application-level multicast infrastructure," IEEE Journal on Selected Areas in Communications, vol. 20, no. 8, 2002;

[33] Bortnikov V., Chockler G., Roytman A., and Spreitzer M., "Bulletin board: a scalable and robust eventually consistent shared memory over a peer-to-peer overlay," SIGOPS Oper. Syst. Rev., vol. 44, pp. 64-70, Apr. 2010;

[34] Y. Tock, B. Mandler, J. Moreira, T. Jones, "Scalable Infrastructure to Support Supercomputer Resiliency-Aware Applications and Load Balancing", SC'11 poster.