Towards a New Architectural Framework – The Nth Stratum Concept

Martin Johnsson
Ericsson Research
martin.johnsson@ericsson.com

Jyrki Huusko
VTT Technical Research Center of Finland
Jyrki.Huusko@vtt.fi

Tapio Franti
VTT Technical Research Center of Finland
Tapio.Frantti@vtt.fi

Frank-Uwe Andersen
Nokia Siemens Networks GmbH & Co. KG.
frank-uwe.andersen@nsn.com

Thi-Mai-Trang Nguyen
University of Paris 6, France
Thi-Mai-Trang.Nguyen@lip6.fr

Miguel Ponce de Leon
Waterford Institute of Technology
miguelpdl@tssg.org

ABSTRACT
Current architectures and solutions are about to reach the limits of sustainable developments. Over the years, many new requirements have emerged, and there are observations pointing to an ever-increasing diversity in applications, services, devices, types of networks at the edge and the access. Meanwhile, the infrastructures for internetworking, connectivity, and also management remain fairly the same. A new paradigm is needed that can support a continuous high pace of innovations in all the different parts and aspects of a communication system, while at the same time keeping costs of deployment and maintenance down. This new paradigm has to embrace current trends towards increased heterogeneity, but on the other hands provide support for co-existence and interoperability between alternative and various solutions all residing within a global communication system. This paper presents a new architectural framework called the Nth Stratum concept, and which takes a holistic approach to tackle these new needs and requirements on a future communication system.

Categories and Subject Descriptors
C.2.1 [**Computer Systems Organization**]: Computer-Communication Networks – *network communications, packet-switching networks.*

General Terms
Design, Standardization.

Keywords
Architecture, Framework, Stratum, System, future Internet.

MobiMedia'08, July 7–9, 2008, Oulu, Finland.

1. INTRODUCTION
Existing systems for communication like Internet are challenged by developments occurring at various parts of these systems. Especially towards the edge, new forms of applications, access networks and devices make it increasingly difficult to keep the overall communication system consistent and coherent, as well as to maintain inter-operability. One can observe a fragmentation of the system into different non-interoperable segments. Peer-to-peer technology makes new and fairly independent application-specific networks appear on top of the existing networks as overlays, and generally with nodes in such overlays being at edges. Further to this, core features such as security, mobility, and QoS, are sometimes lacking proper support, which leads to different middlebox and adhoc solutions, and not to forget about the still occurring problems with security attacks, spam, as well as maintaining the overall stability of the routing system. The challenge is an ever-increasing need to support not only a high growth of terminal entities of various kinds but also to support mobility and multi-homing.

Cellular networks are challenged by the emergence of alternative and complementary radio access systems. The integration of these different types of networks calls for a flexible architecture framework, especially in view of the fast pace of application and device development. The Internet on the other hand lacks support for mobility, QoS, and security, which are fairly well supported by the cellular networks. However, Internet shows, so far, a clear advantage in supporting application development, making it relatively easier to roll-out new services.

Considering the status of the current systems for communication as being described above, it does not seem reasonable and feasible to continue with patching those systems with different 'fixes'. Instead we advocate that a new architectural framework is needed which can provide the necessary ontology, properties, means, and design patterns to support and sustain a high pace of application development. At the same time it maintains co-existence and interoperability between the various segments, levels and components of the communication system. It must consider and be able to resolve the issues with existing systems, while also ensuring support for emerging applications and network technologies. We do believe a holistic approach is needed to make such an architectural framework applicable for the support of global development on a large and broad scale of a future communication system.

The rest of this paper is organized as follows. In Section 2, we identify the research problem and discuss some related work. An overview of the Nth Stratum based architectural framework is presented in Section 3. The notion of Core Feature Integrity (CFI) and its use within the proposed framework is discussed in Section 4. Signalling and Management related issues are described in Sections 5 and 6 respectively. Finally, we present some concluding remarks in Section 7.

2. STATE OF THE ART & PROBLEM STATEMENT

Current architectures for communication systems are typically developed around layered models like TCP/IP, OSI and 3GPP. Practice in the open Internet environment has shown that it is difficult to realize network enhancements, such as IPv6, IPSec, MobileIP or multicast [1]. The dramatic growth of the Internet has also brought into sharp focus its architectural deficiencies such as lack of support for QoS and seamless mobility, security vulnerabilities, address shortage, to name a few. Although a number of solutions have been proposed for these problems, these can, at best, be described as a patchwork of fixes to fill architectural holes. Thus, the resulting system has become complex, often with similar functionality re-appearing in different protocols and layers. The piecemeal ad hoc approach to solving problems that applies “patches” to certain parts of the protocol stack can in fact jeopardize the operation and performance of other parts of the communication system.

For instance, NAT [2] was proposed to resolve address shortage and some security issues but it changed the original end-to-end architecture to the client-server architecture. Another example is IPSec [3] which was designed to secure IP datagrams. However, it is found to be unsuitable in environments characterized by high user mobility due to constant re-establishment of IPSec tunnels. Mobile IPv6 [4] has tried to address this problem by integrating IPSec procedures with the Binding Update process, but the approach is clearly untenable in the long-run.

Several recent research efforts seek to define new architectural principles that are more flexible, support cleaner cross-layer interaction and facilitate network as well as service composition. Such efforts have been driven by the need to develop flexible and extensible architectures that can meet the needs of an ever-expanding global network.

One interesting approach is to use the notion of components (e.g., modular protocol units [5] or services, such as flow control, FEC [6]) that can be flexibly composed according to the particular requirements of applications or users. To some extent this is a revival of micro-protocols research. It focuses on protocols and their components and does not really consider the overall architecture. The Role-Based Architecture [5] proposes to get rid of the strict layering of protocols and replacing them with functional units called 'roles', organized arbitrarily for greater flexibility and richer interactions between protocols.

Other approaches stick to the layering principles. The recursive networking architecture [7] applies a generic meta-protocol to all layers to make cross-layer interactions cleaner and to avoid multiple instantiation of the same functionality at different layers. The metaprotocol is configured according to the individual requirements of the respective layer. As another example, the principle innovation of the architectural work in FARA [8] developed in the DARPA project NewArch [4] is the decoupling of end-system names from network addresses. The resulting architecture consists of entities and associations with the latter residing on a communication substrate and providing connectionless packet delivery.

The PlutArch proposal [11] facilitates inter-connectivity between networks based on different architectural principles using gatewaying functions although it does not specify what the different network architectures may look like. The Autonomic Network Architecture (ANA) project [9] concentrates on the autonomic behaviour of networks. It is working on an architecture that includes different types of networks. Definitions of primitive concepts such as network compartment and information channel have been identified. ANA does not investigate the problem of how ANA-hosted network architectures can be instantiated or be made to interoperate.

These above research efforts show that there are considerably more points in the network architecture design space than the Internet and current wireless systems. A common feature of the proposals described above is that the architectures are rather generic and high-level without actually specifying how they could be used to realize a communication system. This motivates the need for defining an architectural framework that is generic and flexible enough to accommodate diverse networking architectures and at the same time has sufficient level of detail to make it usable for instantiating specific communication networks. In this sense, we aim to reach beyond the individual, fragmented architecture research undertaken so far.

The proposed framework will incorporate the notion of architectural patterns to facilitate the design of particular network instances. Earlier work has already shown that patterns or components are a promising approach for rapid development of application-tailored protocols (see [12], [13]) but has mostly focused on single protocols and not on overall network architectures. In this context several protocol frameworks were developed (e.g. Conduits+ [14], dynamic architecture [15], and x-Kernel [16]). We extend this work by developing a framework through which different network architectures, and not just protocols, can be designed. The framework lays the foundation to ease the design of network architectures and to increase the productivity in instantiating them. We apply design patterns, together with other principles from software-engineering such as model-based design, to develop a framework that explicitly takes interoperability requirements into account. In particular, the notion of design invariants highlighted by the Ambient Networks project [17] is used to define practically useful patterns [18].

The interaction of core networking features like naming, addressing, QoS, mobility, and security is a critical aspect of future communication systems. Though these features are frequently considered in isolation, it is, in fact, their interplay that determines whether a particular pattern is appropriate for a given architecture. One of the goals here is to develop architectural patterns that enable the design and deployment of feature combinations that are tailored to particular applications. For example, security and QoS properties can be customized to user needs and may even be adapted during an on-going session. Similarly, an architecture can be customized to the individual requirements of certain types of networks, such as sensor networks where security may be traded off against energy consumption. This requires careful trade-off analysis, not only for cross-layer principles as in [19], for example, but generally for the different features. The proposed architectural framework is designed to enable the integration of new features, extending the period over which a communications system can sustain significant and partly unpredictable developments.

In this paper, we present such a framework based on the concept of 'stratum' (pl. strata), which encapsulates any layer, aspect and property of communication, control, and management in a communication system. Within this framework, network architectures are seen as specific configurations of strata with well-defined interactions and interfaces between them. Through this concept, we aim to provide an architectural framework, which clearly improves on such important properties as extensibility, flexibility, modularity, and (self-) manageability, to meet the requirements on future communication systems supporting fast development and deployment of new protocols, applications, and services while maintaining goals on the overall system coherence and performance.

3. CONCEPT OVERVIEW

The principal and fundamental entity of the Nth Stratum concept is the Stratum. It denotes an entity which inherently has characteristics of distribution (an aspect of any communication system) and provides features/properties/functionality for a certain slice/layer/component/aspect of a communication system. Each stratum is modeled as consisting of a set of nodes containing functionality for data processing, and a medium which defines how data can be transferred between the nodes in the stratum. However, especially for reasons of designing stratum that is only a single component, or for reasons of stratum aggregation (see further below), a stratum might only contain one node. The aspect of functional distribution within a stratum implies not only that there might be several different types of nodes, but also a need for identification of either the nodes, or identification of ports of the medium through which nodes can be reached. Figure 1 below generically depicts a stratum. When specifying a stratum, the data processing in nodes, the medium, as well as naming schemes must be defined.

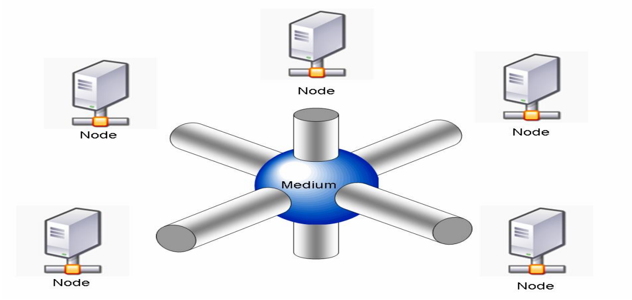

Figure 1. The nodes (N) and the Medium constituting the principal building blocks of a stratum.

In order for a stratum to execute its specified functionality, it may need to use services from one or more other strata. This also leads to the additional requirement when specifying a stratum to also define its Stratum Service Point (SSP). The SSP is similar to the Service Access Point defined within the OSI model, but where in addition, an SSP can offer information about the specific properties and features being an integral part of the stratum, and which might not be directly visible out from how particular services have been defined. The SSP defines the set of services offered by a stratum to other strata using identification schemes, parameters etc which are defined and understood by that stratum. Another stratum which likes to use the services of another stratum should define a Stratum Transition Point (STP). The STP defines how a stratum that likes to make use of an SSP of another stratum translates and resolves the use of the SSP from its own identification and parameter schemes, e.g. as a means to define and specify a name resolution mechanism. Figure 2 shows two strata, one making use of the services of another through a defined STP and SSP. An STP is specified on as-and-when-needed basis depending on what other relations to other strata a particular stratum has.

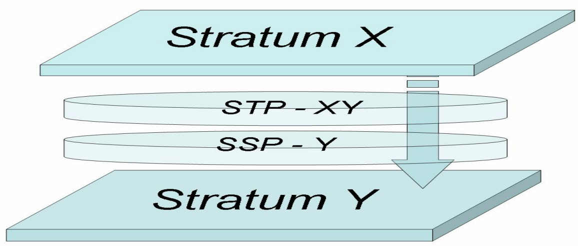

Figure 2. Relation between two strata.

Besides the possibility that one stratum makes use of the services of another stratum via an STP and SSP, two strata might also have a peering relation. Peering might occur when similar types of strata (e.g. with similar or the same identification schemes) have been deployed in different segments of a communication system. This peering relation is specified by defining a Stratum Gatewaying Point (SGP). The SGP must define the mapping of identification schemes as well other types of parameters between two peering strata. Just as with STPs, also SGPs are specified on as needed basis. Figure 3 depicts two strata in a peering relation.

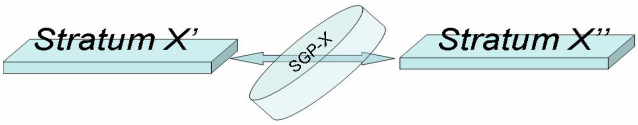

Figure 3. Two strata of similar types, X' and X'', having a peering relation via a Stratum Gatewaying Point (SGP-X).

In the following subsections, further elaborations and aspects of a stratum are described.

3.1 Inheritance

The Nth Stratum will support the principles of inheritance. Inheritance can be used to define more generic features and properties of the communication system, and the main purpose to use it is to ensure that the Nth Stratum concept is able to specify "the endoskeleton" of the architectural framework. That is to say that the ambition is to keep such a framework consistent and coherent through its development over a long period of time.

There are many aspects, features, and properties of networking that lend them to be generically specified and defined, possibly at different levels of abstractions supported by an inheritance hierarchy. Some examples are identities (names, cryptographic ids), locators, policies, QoS, security, mobility and self-management properties. Not only should it be a matter of providing consistent and coherent definitions for each of such properties, but also for them as a whole, especially when there are critical dependencies between them.

Inheritance of properties and features in the Nth Stratum concept is supported by the notion of the abstract stratum. Such a stratum might, and especially on the level of abstraction it is defined, have the properties defined for a stratum above either only partially defined, deferred, or not defined at all. Compared with a stratum, an abstract stratum can not be put into execution in a run-time environment.

3.2 Stratum as a Component, Stratum Aggregation

In the general sense, an individual stratum basically constitutes a module of a communication system, encapsulated by means of its Stratum Service Point. There are no limitations as to how "thin", "thick", or "narrow" the definition of a stratum needs to be. It can surely be so that the definition of strata is used as a means to develop generic components for a communication system, and which then can be included or aggregated into a definition of a more specific and/or "broader" stratum. Such a generic component can even be defined as an abstract stratum, and where certain settings, features, properties etc can be left unspecified in the abstract stratum, but which then can turn into a concrete specification when inherited by some stratum.

3.3 Stratum Instantiation

Stratum is the specification of a module, and for its execution, it needs to be instantiated. The instantiation occurs within the nodes of a communication system. As nodes interconnect, and in accordance with the stratum specification (i.e. the medium), they start to form a distributed instantiation of the stratum. Here we do assume that stratum generally inherits and implements self-management capabilities, which shall provide support for such properties as self-organisation, self-configuration, self-healing, and possibly also other self-* properties. This instantiation may stretch to any scope, possibly globally. Thus, the definition of a stratum must carefully consider its scope of operation, and e.g. analyze whether an instantiated stratum might cross domain borders, in which case, the definition should include mechanisms for the control and management of domain borders. If there are specific needs to control the scope of the instantiation of a stratum, the Stratum Gatewaying Point shall be used.

3.4 Horizontal and Vertical Strata

Given these principle and basic constructs of inheritance, modularization, and instantiation, the Nth Stratum framework allows for strata to be "stacked" in an arbitrary way with no specific ordering or relationships defined, but where restrictions might apply as from design principles and the basic framework constructs. Within this context, we differentiate between the so-called horizontal and vertical strata.

The horizontal and vertical strata have different characteristics and meaning. Although, both represent conceptual as well as concrete, instantiated strata, in essence they distinguish between the connectivity/service/application aspects of a system on one hand (the horizontal), and the overall configuration, governance, performance, consistency and coherency of the system and the framework as a whole on the other hand (the vertical).

A set of horizontal strata can be aimed to generally provide network connectivity and infrastructure services. Such kind of strata generally implements "in-network" management-related functionality such as self-* properties. Their horizontal position indicates that they may relate to all strata, and offer services (e.g context awareness) to other strata via the STP/SSP.

Vertical strata are the ones which can have an impact on the horizontal strata such as the overall governance, composition, and monitoring of horizontal strata, selecting and building the configuration of horizontal strata. On the vertical, conceptual stratums can also exist and are defined as "abstract" strata. They are considered as a set of "libraries" which will be used to build a concrete "network architecture".

3.5 System Aspects

The Nth stratum concept allows for dynamic composition and configuration of a communication system, or segments thereof. The specific organization and configuration of the strata to be used is up for (local) optimization criteria according to the requirements and needs of (local) network operators.

The result might be that the organization and configuration of strata could differ between different segments of a communication system. It is for further study if certain strata, let alone some abstract strata that provide the overall framework for the entire communication system, needs to be made mandatory for any segment of a communication system, or whether "inter-segment" communication shall be done via Stratum Gatewaying Points.

3.6 Stratum Framework Example

The management of the horizontal strata is taken care of by three different vertical strata as depicted in Figure 4: Governing Stratum, the Performance Optimization Stratum, and the Knowledge Stratum. These are explained further in Section 6.

In the case shown in Figure 4 where, for example, two different horizontal strata have been configured to be used for communication; the Virtualization Stratum allows for flexible allocation of the underlying physical resources (and where a 'folding point' could be an integral part of a defined SGP (not depicted) between peering Virtualization strata) while the Connectivity Stratum can e.g. be used for real-time applications (some type of "application stratum" not shown in figure) or e.g. for expedited delivery of data.

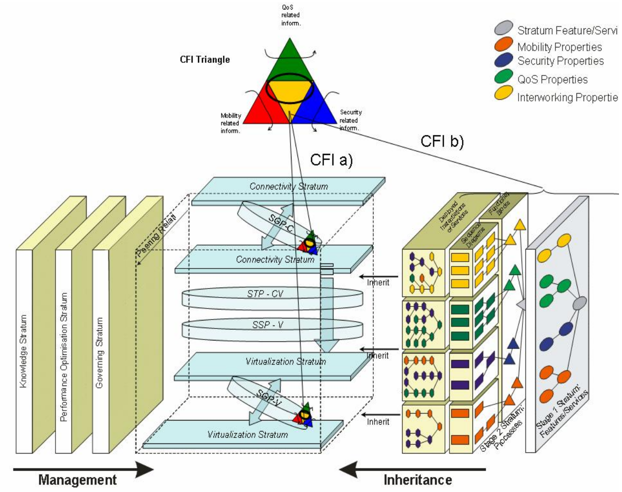

Figure 4. Nth stratum, horizontal and vertical strata, and CFI methods a) and b).

The two different strata in the example above used for communication in turn inherit characteristics from different abstract strata. This is further elaborated upon in Section 4. Many other strata can be envisaged for the purpose of defining and possibly also enforcing common properties and mechanisms, e.g. a (common) signaling system, see Section 5. It shall be further noted that there is no specific limitation regarding neither how many strata can be defined, nor which specific strata that can be used in certain "segments" of a communication system.

4. CORE FEATURE INTEGRITY (CFI) DESIGN

In many network designs, functionality is modeled in a way that treats them as separate "entities" (e.g. Ambient Networks [20]), sometimes as "blocks" (e.g. IEEE [21]) or also "domains" (e.g. security domains in 3GPP [22]), depending on the purpose of the individual design. This "divide & conquer" type of separation may help simplify the design, but sometimes it leads to a situation where the strict separation of functionality negatively affects their cooperation. More detailed specification of the functions usually shows their inter-relatedness and inter-dependence, but the abovementioned design approach does not explicitly support it, since its main purpose is to separate, and not unify functionality. While not stating that the classic design approach is generally a problem, especially when executed with great care, we want to avoid the described problem by taking another approach, namely thinking about conceptually stressing the interoperation and interdependence of functions right from the start.

We see at least the well-known triple {mobility, security and QoS} (c.f. Moby Dick [1], Ambient Networks [6], SeQoMo [23]), in the center of the overall functionality of a future communication system and will refer to them as "core features". These will also extend to self-management capabilities and also likely cover also naming & addressing aspects. Core features are abstract placeholders for the actual mechanisms and protocols to be deployed in a system.. If designed independently of each other, they provide operators with flexibility to a certain degree, because mechanisms may be replaced quite simply, but they constitute a problem when it comes to offering an integrated service. More specifically, a mobile node's changing connectivity to access networks may be well handled by an appropriate mobility mechanism, but will QoS and security be adapted or established accordingly? Parallel, simultaneous use of the existing technologies such as IPSec together with DiffServ and Mobile IP seems not easily feasible today. Many more examples can be constructed, showing that the core features have many interdependencies [24]. This is why it seems useful to design and implement the core features together from the start, not even separating between them any more, and to create just one block of functions, resembling the "service oriented architecture" (SoA) approach [18].

Both extremes, i.e. the functional separation as well as the complete integration have strengths and weaknesses, so the optimum may be found in between, as in any tradeoff situation. It would be desirable to have the flexibility in an overall architectural framework to design functions and services in a way that permits to define or control the level of their integration. With relation to the Nth Stratum concept, we see at least two complementary architectural constructs for CFI, also displayed in Figure 4.

(a) Exploiting the gatewaying and peering properties: Different strata are interworking (peering), according to the model, via specialized SGPs. At these locations, the core features will naturally "meet" and can therefore interwork, according to Figure 3 and Figure 4. Whenever any communication takes place, these locations would be appropriate as interaction points, where functional synchronization may take place.

(b) Following the inheritance path: The rightmost "stage one" abstract stratum (Figure 4) does not standardize the usage of a stratum, but it describes the communication service in terms of the perceptions of the user receiving the stratum service, allowing for an understanding of the service without regard to implementation. The static aspects of a service can be efficiently described by attributes such as QoS or QoC (Quality of Context), mobility and the aforementioned other sub-attributes. The stage 2, i.e. less abstract stratum defines the derivation of a functional stratum model based on stage 1. This would include the functional entity actions and allocation of functional entities to physical locations. The stage 3, or least abstract stratum gives examples of signalling flows for the stratum based on specific protocol formats, and the requirements identified for switching and service nodes in an instantiated stratum.

As an example, if we were to implement a network based on the Nth Stratum framework that needs to support both sensor networks (SN) and an IP based network, we can either say that the SN part is one stratum and the IP network another stratum and then design an SGP that does the conversions, or we can follow the inheritance principle (known from OOP) and flexibly scale simple core features to ever more complex and specialized inherited classes. For example, QoS methods are derived from a root element (such as "best effort" in the case of QoS), which all inherited strata will have to support in some way, while more specialized strata in the inheritance path can use more sophisticated methods. The same principle applies to mobility: support for simple handover as the base mechanism and inherited high-quality handover with real-time or capabilities for seamlessness. But, most importantly, any relations defined at a more abstract stage between QoS and mobility, such as pre-handover QoS negotiations will be available in any of the inherited classes, too. So, if simple core feature interworking procedures are defined in a more abstract stratum, these will quite automatically be conserved and possibly evolved into more complex ones in the inherited strata, which has exactly the desired effect not to neglect core feature interworking. The gatewaying approach corresponds more to the modular, less integrated design mentioned in the beginning of the chapter, while the inheritance can be used to focus on the full functional integration. The framework thus gives designers choices for controlled functional interworking.

5. SIGNALLING AND QoS ASPECTS

In the next generation Internet, importance of the just-in-time service provisioning increases. It is foreseen that for instant service provisioning with guaranteed QoS and QoE the protocols and communication mechanisms, i.e., architecture should be well designed and optimized.

Currently, e.g., multimedia delivery and mobility management mechanisms need to rely on cross-layer communication with inherent deficiencies for the improvements. At the same time new middleware functionalities are introduced to improve, for example, security and reliability, which implies increased complexity on cross-layer information controllers and managers, and protocol communication and hierarchies. However, information needs to be transferred also through the network, not only inside the protocol stack, expanding the overhead of communication dramatically, which gives way for the new stratum based architecture. In the stratum approach, all and only the required functionalities on the particular strata are inherited from the abstract stratum. All the properties in the abstract stratum are designed to be in line with the architectural framework. Therefore, it enables overall improvement of the scalability, minimization of the redundancy in a protocol stack and real-time control signalling for emerging needs instead of solutions that may serve a short-term purpose but significantly impair the long-term flexibility of the Internet (or other communication systems).

A stratum based approach with inherent real-time signalling entities enables instant connection establishment and optimized just-in-time type service provisioning. Real-time signalling here means that control information is transmitted, if necessary at once even one control bit a time (without conventionally collecting a bunch of them before transmission) for immediate actions in the communication system. We would like to note that every time when payload bit(s) are transmitted, transmission requires (media independently) minimum amount of wave phases for receiver's synchronization and this of course itself increases redundancy. However, it is considered that real-time functionalities of the stratum approach would decrease overall redundancy in communication systems because separation of control signalling bearers also helps to minimize the total amount of control information by decreasing redundancy due to time delays. The amount of control information is, for example in cellular system like GSM and 3G, more than 60% from the overall traffic and in most of the cases the signalling is not even close to real-time in a strict sense. The real-time control signalling may require to "permanent" allocation of carrier(s) that could be used for other purposes if control information is not transferred). In other words, there would be control channel "open" all the time for the instant control information transmission.

6. MANAGEMENT ASPECTS

The next generation Internet is expected to be adaptable and auto-piloted. The network should be able to decide and make use of the protocols and algorithms which are best suitable to the user’s needs and network context in real time. The ability of supporting several protocols or algorithms and switching between them without the perception of user is called dynamic protocol selection and configuration. For instance, when a DoS attack is detected, the network can change its current routing protocol in order to be immune to this attack. When the volume of real-time traffic in a network reaches a threshold, the network can change QoS configuration or algorithms to adapt to traffic conditions. The transport protocol can be appropriately selected in function of application’s needs and available access links.

In the Nth Stratum concept, dynamic strata selection and configuration is realized by the two vertical strata, Governing stratum and Knowledge stratum as illustrated in Figure 4. The set of algorithms and protocols used by a communication system to provide connectivity can be considered as part of the configuration of a stratum. The Governing Stratum is responsible to decide in real-time the set of strata needed for a communication system and how they are individually configured. In order to make a decision, the Governing Stratum can obtain information from the Knowledge Stratum which is responsible for providing a detailed view of the network state (round trip delay, packet loss rate, and network load) and the network context (temperature over the wireless link, the nature of the link, underway failures, security problems, etc.).

The performance of a network depends on three factors: the algorithm used by the Governing Stratum, the network picture provided by the Knowledge Stratum, and the specialized protocols and algorithms supported in horizontal strata. An efficient algorithm used by the Governing Stratum should be able to calculate the optimized configuration of the horizontal strata in function of new user demands and network conditions. The change of security mechanism used in the network to react to an attack detected is an example of how the Governing stratum changes the configuration of the horizontal strata. The more the network picture provided by the Knowledge stratum reflects status of the real network, the more efficient the action taken by the Governing Stratum will be. If the signs of the attack is not present or is present but late in the Knowledge stratum, the Governing stratum cannot react in time. Finally, the richness of protocols and algorithms supported in horizontal strata stands in proportion to how efficient and how well the Governing stratum can perform, which could potentially be mitigated by a Performance Optimization stratum (see Figure 4).

7. CONCLUSION

In this paper, we have presented a novel idea for an architectural framework based on the Nth Stratum concept. It provides a holistic and systematic approach to development and design of network architectures for future communication systems. This paper describes fundamentals of the Nth Stratum concept, and should be considered as a starting point for the further work of detailing the specification of this concept. We like to proceed with how to describe, e.g. the structure, syntax, and semantics, the functions and features of a stratum, including also the Stratum Access/Transfer/Gatewaying Points. We also need to describe a design process and guiding principles for the development of strata as part of an architectural framework. Eventually, a number of 'principal strata' will emerge, not precluding those might very well evolve over time, both in numbers and in quality and content, and which all-together forms the foundation of a new architectural framework.

8. ACKNOWLEDGMENTS

The research for this paper was carried out in partially European Commission funded 7th framework programme (FP7) project 4WARD (FP7 project reference: 216041). The authors would like to thank also all the colleagues in the project and during the project preparation phase for lively and innovative discussion about the new architecture models for future Internet. The authors would like to thank especially Dr. Nadeem Akhtar for giving valuable input in project preparation phase, which made this paper possible.

9. REFERENCES

[1] P. Mähönen, D. Trossen, D. Papadimitriou, G. Polyzos, and D. Kennedy: "The Future Networked Society", http://www.future-internet.eu/, December 2006.

[2] K. Egevang, P. Francis, The IP Network Address Translator (NAT), RFC 1631, Available: http://www.ietf.org/rfc/rfc1631.txt, May 1994.

[3] S. Kent, K. Seo, Security Architecture for the Internet Protocol, RFC 4301, Available: http://www.networksorcery.com/enp/rfc/rfc4301.txt, Dec. 2005.

[4] D.Johnson, C. Perkins, J. Arkko, Mobility Support in IPv6, RFC 3775, Available: http://www.ietf.org/rfc/rfc3775.txt, June 2004.

[5] R. Braden, R. Braden, T. Faber, Mark Handley, From protocol stack to protocol heap: role-based architecture, ACM SIGCOMM Computer Communication Review, Volume 35, No. 1, January 2003.

[6] Rudra Dutta, George N. Rouskas Ilia Baldine Arnold Bragg, Dan Stevenson, The SILO Architecture for Services Integration, controL, and Optimization for the Future Internet, Proc. Of the ICC 2007, Dublin, 2007.

[7] J. Touch, Y. Wang, V. Pingali; A Recursive Network Architecture; ISI Technical Report ISI-TR-2006-626, December 2006.

[8] D. Clark, R. Braden, A. Falk, V. Pingali, FARA: Reorganizing the Addressing Architecture, Proceedings of the ACM SIGCOMM workshop on Future directions in network architecture, 2003.

[9] Autonomic Network Architecture, http://www.ana-project.org

[10] D. Clark et al, New Arch:Future Generation Internet Architecture, Technical Report, Available: http://www.isi.edu/newarch/iDOCS/final.finalreport.pdf.

[11] J. Crowcroft, S. Hand, R. Mortier, T. Roscoe, A. Warfield, Plutarch: An Argument for Network Pluralism, Proc. of the Workshop on Future Directions in Network Architecture (FDNA) at ACM SIGCOMM 2003 (August 2003).

[12] M. Jung, E. Biersack; A Component-Based Architecture for Software Communication Systems; IEEE Engineering of Computer Based Systems (ECBS) 2000, April 2000.

[13] E. Exposito, P. Senac, M. Diaz; Compositional architecture pattern for QoSoriented communication mechanisms; Proc. of 11th International Multimedia Modelling Conference, January 2005.

[14] H. Hüni, R. Johnson, R. Engel; A framework for network protocol software; Proc. of Annual Conference on Object-Oriented Programming Systems, Languages and Applications, OOPSLA 95, October 1995.

[15] S. O'Malley, L. Peterson; A dynamic network architecture; ACM Transactions on Computer Systems; Vol. 10, No. 2, May 1992.

[16] N. Hutchinson, L. Peterson; The x-kernel: An architecture for implementing network protocols; IEEE Transaction on Software Engineering, Vol. 17, No. 1, January 1991.

[17] Ambient Networks, http://www.ambient-networks.org

[18] B. Ahlgren, M. Brunner, L. Eggert, R. Hancock, S. Schmid; Invariants – A New Design Methodology for Network Architectures; SIGCOMM 2004 Workshop on Future Directions in Network Architecture (FDNA'04), August 2004.

[19] V. Kawadia, P.R. Kumar; A Cautionary Perspective on Cross Layer Design; IEEE Wireless Communications, Vol. 2, No. 1, February 2005.

[20] 3G Security, 3GPP Technical Specification, 3GPP TS 33.200

[21] X. Fu, T. Chen, A. Festag, G. Schäfer, and H. Karl, SeQoMo Architecture: Interactions of Security, QoS and Mobility Components, Technical Report TKN-02-008, Telecommunication Networks Group, Technische Universität Berlin, April 2002.

[22] C. Kappler, P. Mendes, C. Prehofer, P. Poyhonen, D. Zhou, A Framework for self-organizing network composition, Proc. of the 1st IFIP International Workshop on Autonomic Communication, Berlin, 2004

[23] J. Jahnert; J. Zhou. Jie; R.L. Aguiar, V. Marques, M.Wetterwald, E.;Melin, J.I. Moreno, A Cuevas, M. Liebsch, R. Schmitz, P. Pacyna, T. Melia, P. Kurtansky, Hasan; D. Singh; S. Zander, H. J. Einsiedler, B. Stiller, “The ‘Pure-IP’ Moby Dick 4G architecture”, Computer Communications, Volume 28, Issue 9, 2 June 2005.

[24] F.-U. Andersen, L. Caviglione, Survey of IPv6 functional interoperability for mobile Internet, 2nd International Conference on, Mobile Technology, Applications and Systems in GhuangZhou, 2005.