The *Viator* Approach: About Four Principles of Autopoietic Growth On the Way to Future Hyperactive Network Architectures

Plamen L. Simeonov

Technology University of Ilmenau
Faculty for Informatics and Automation
Email: plamen.simeonov@prakinf.tu-ilmenau.de

***Abstract— Viator*, the Wandering Network approach, defines a new type of communications architecture chracterized by:**

- **flexible, multi-modal specialization of network nodes as virtual subnetworks;**
- **mobility and virtualization of the net functions in hardware und software;**
- **self-organizing multi-feedback topology-on-demand.**

Network elemens can contain several exchangeable modules capable of executing diverse network functions in parallel. They can be invoked, transported to or generated in the nodes upon delivery of mobile code about the node's behaviour.

***Index terms*— active networks, reconfigurable computing, adaptive systems, multimedia architecture, mobility, autopoiesis.**

A. Introduction

Computer networks were engineered to operate as communicating device aggregates which were product of the "control" branch in cybernetics, [1]. Therefore, inheriting the "controling" pattern[1], [2], in their overall design and deployment during the 80[ies] and 90[ies] of the XX century was a kind of a "natural" phenomenon. However, telecommunication is now increasingly becoming a complex social activity which follows much different principles of growth than the ones it has been concerned with until now. Researchers began to speak of "self-organization" which is the ability of a system to construct and change its own behavior or internal organization "from within"[2]. This field deals with more complex regulation processes through which biological and social systems adapt themselves and maintain homeostasis within a given environment, [9].

Active networks (AN) have been a subject of intensive empirical research for more than a decade, [3]. For the time being, a few general implementation strategies for the two main instances, *active nodes* and *active packets*, have been identified. Most AN approaches investigate and implement them to some detail within a specific solution, such as caching, routing, management, etc.

Recently, an integration and consolidation of the several different AN engineering approaches can be observed. This trend is particularly evident at technology frontiers such as deeply embedded networked systems, autonomous software, configurable computing, adaptive systems, etc.

However, implementations have shown that every single network issue can have a specific *active* network or open signaling solution. Therefore, a few survey papers were published trying to provide directions and goals for engineering within the field ([4], [5], [6]). These efforts were mainly focused on defining a framework for *a common programming model* of active networks. In addition, a number of basic requirements and concepts for enhanced *virtualization* have been collected to (self-)activate networking. Yet, there is still no general recipe to address all the problems with only one end-to-end active or programmable network. Despite the broad interest in the subject, cf. [8], the "killer" network of the future has not been found yet.

A close view at previous research in active and programmable networking let us realize that the concept of "activity" has been addressed in a quite narrow sense, much like the one of "intelligence" in Intelligent Networks, [7]. Although the classical AN and Open Signaling approaches still dominate the field, there is a lot more that can be done w.r.t. utilizing the "degrees of freedom" of network activity.

Along with the growing scope and number of ad-hoc solutions to active networking, the demand for their systematic categorization, evaluation and integration within a common research framework becomes increasingly evident. In particular, an *evolutionary* approach to active networking requires the development of common models for: a) the encoding of network programs in terms of mobility, safety and efficiency; b) the description and allocation of node resources; c) the built-in primitives and *behavioral* patterns available at each node, etc. The last argument reminds us that active networking is on the way to become a bit of a "social" culture hosting diverse technology development aspects where "self-organization" is going to play an essential role.

We regard networking as a *synthetic* science. Therefore, we decided to apply the "self-organization" approach to networking. The goal of this work is to provide a model-based formal theory for the design and verification of (step-wise) evolutionary, *autopoietic*[3] (i.e. self-creating, cf. [9]) communication architectures based on: i) active networks ([4],[5],[6]), ii) re-configurable computing ([10], [11]), iii) adaptive systems, [12] and iv) mobile communications ([13], [14], [15]).

[1] e.g. the Service *Control* Point (SCP) of an Intelligent Network (IN) or the Transport *Control* Protocol (TCP) of a packet switched network, etc.

[2] i.e. on its own account -- neighter in response to conditions in another system with which it may interact nor as a consequence of its membership in a larger metasystem.

[3] ***Autopoiesis*** or self-creation, is the property or condition of a system to regenerate itself by self-reproduction of its own elements and of their characteristic interactions, cf. [9].

This paper proposes a new generalized framework for the design of telematics networks as autopoietic systems. In the following, we introduce the *Viator*[4] approach to "*hyper*activity" in future networks.

B. THE WANDERING NETWORK

Table 1 summarizes the basic characteristics of an active network reference model leaned on ANTS [16] along with the available options for extension (in *italic*).

Active Nodes	Active Packets
- Have structure that could be re-configured with time. - May accommodate some residential program code for processing packets. *Could support multiple code schemes to define classes of services.* - Do processing on packets. Could be processed by packets. Could do some processing on themselves. - Could be mobile[5].	- Have structure that could be re-configured with time. - May carry program code, but do not execute it. Could support multiple code schemes. Could carry some code for AN reconfiguration. - Are processed by nodes. Could do some processing on nodes. Could do some processing on themselves. - Are mobile.

Table 1: Open enhancements to the AN concept

With this motivation, we defined a new network generalization for programmable active networks, the *Wandering Network* (*WN,* [17]). The new concept is based on previous research in intelligent and smart networking [18], [19], [20], and a fomalism called *WLI* (the Wandering Logic Intelligence, [21]), which extends the Columbia University model for a programmable network, [6], by three essential characteristics:

1. it is a *hyper*active network which means that it is programmable and reconfigurable, incl. the network hardware up to the gate level;
2. it is a runtime extensible and exchangeable network in terms of both software and hardware components (a *wandering network)*;
   - it is an evolutionary network which realizes *adaptive* self-distribution and replication of sub-networks:
   - by guided or autonomous node and component mobility in terms of hardware;
   - by including network engineering information in the mobile code of the active packets and applying *genetic transcoding* mechanisms in the active mobile nodes.

We distinguish between four generations of Wandering Networks (WN). The First Generation WN includes most of the traditional active network approaches as known to be programmable at the highest execution environment layer. The Second Generation WN addresses programmability at both execution environment (EE) and node operating system (NodeOS) layer. Some of the present AN systems such as ANON [28], Tempest [29] and Genesis [30] can be classified to the 2G WN. The Third Generation WN addresses programmability at the last layer of networking, an active node's hardware and switching circuitry, in addition to the 2G WN capabilities. We are not aware of any network architecture that could be classified to the 3G WN[6]. Finally, the Fourth Generation Wandering Networks is characterized by adaptive self-distribution and replication.

The *Viator* approach presented in this paper addresses 4G WN as follows:

- Active nodes may be mobile, - hence the name *ships* -, and re-configurable (in terms of software and hardware). In addition to traditional active nodes, ships can be also *modified*[7] *by shuttles*.
- A ship's runtime re-configuration can be invoked by internal procedures or upon execution of newly arrived shuttles. Autonomous mobile hardware components (*netbots*) take care for delivering their own "driver" routines (mobile code) at "docking time" on the ship.
- Active packets are called *shuttles* and carry code and data for the upgrade/degrade and re-configuration of ships. In addition, shuttles can carry genetic information about the ships' architecture and their communication patterns.
- A *code distribution mechanism* ensures that shuttle processing routines are automatically and dynamically transferred to the ships where they are required. In a WN, code distribution throughout the network and inside the ships can be *maintained by the shuttles themselves*.

Figure 1 illustrates a snapshot of such an evolutionary "always being uder construction" Wandering Network, where the different shapes of the nodes represent different functionalities at a given moment. The *Wandering Logic Intelligence* (*WLI,*. [21]), is a theory for modelling Wandering Networks. WLI generalizes AN capsules in *shuttles* as relatively autonomous mobile components including both programs and data possibly encoded in a language[8] with (semantic) references to ships and other shuttles within the same or a different flow (protocol).

[4] *Wanderer* (lat).

[5] particularly addressed by active ad-hoc networks.

[6] Of course, hardware re-configuration and programming is possible to some extent at the FPGA-level of some specialized cards in present day switches and routers. However, there is still no commercial product or research prototype that allows the runtime exchange of switching circuitry (plug-and-play modules) synchronized by driver updates in the node operation system.

[7] The capsule APIs and the execution environments can be extended by special routines allowing the accommodation and execution of code that changes a ship's configuration and resources. In this way, new functionality can not only be delivered to and injected into the active node, but also distributed and optimized throughout the node itself.

[8] This language should be capable to address in a uniform way even such issues as hypermedia content (e.g. MPEG-4/7) and related knowledge-based management systems (e.g. [26], [31]) along with the corresponding encoding/decoding routines and references to them.

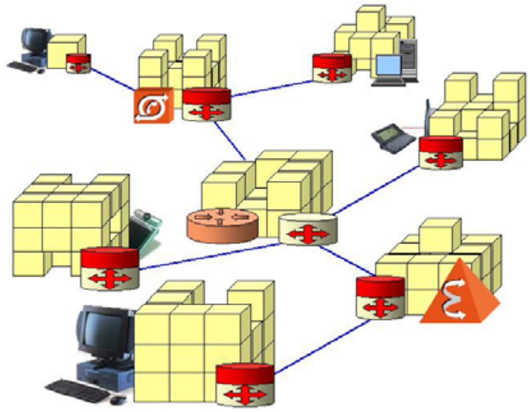

Figure 1: A Wandering Network

A Wandering Network is *self-referential*[9] in terms of its own evolution which is defined by:

1. a circular system description[10] (feedback) that may involve not only referential but also causal or instrumental relations between the components and thereby constitute a unity of its own which has the power to affect, control or modify the form or the validity of that description, and
2. a system of a system which is capable to constantly refer[11] to itself by reproducing its own elements and their specific interrelations without any need for external intervention (and even in spite of such interventions).

Furthermore, the WLI model allows the creation of new capsules/shuttles (or the replication of "old" ones) in the intermediate active nodes under the supervision of the NodeOS. In addition, a special class of shuttles, called *jets* are allowed to replicate themselves and to *create/remove/ modify other capsules and resources* in the network.

The goals of the WLI theory are summarized as follows:

- to provide a formal means for the specification and verification of the generic temporal properties of active mobile nodes and packets;
- to support the reflexive dynamic adaptation of both mobile code (software) and node architecture (software and hardware);
- to provide the formal means for specification and verification of dynamic QoS and routing properties in ad-hoc mobile networks at both application (service) and packet level;
- to assist the formal transformation of the systems properties into mobile code.

[9] *(self-)reference* is the notion that the significance of a given system's behavior is meaningful only with respect to itself.

[10] incl. 2d order cybernetics, i.e. the description of a system by an observer who is part of the system itself.

[11] Self-referential systems are "(organizationally) closed" which means that the system necessarily maintains numerous exchanges with its environment, but these do not affect its basic identity.

In the following section, we present four design principles which further qualify the Wandering Network as a new generation hyperactive architecture.

C. THE WLI PRINCIPLES

The goal of *Viator* approach is to propose and demonstrate a simple and flexible mechanism for network evolution based on the emergence, change and movement of functional units within a given physical infrastructure which recognizes its own boundaries. Such a network is known as an *autopoietic system*. The following definition is closely related to the one given by Maturana and Varela in [10].

> **Definition 1:** A *Wandering Network* (WN) is a dynamic composite entity realized as a unity of a closed set of *productions* of mobile nodes, called *ships,* such that through their interactions in composition and decomposition (programming, adaptation, self-configuration[12], self-reflection[13], etc.) at all functional levels they define the network as *self-creating,* i.e. as an *autopoetic system* by:
>
> - *recursive constitution* of the same system of productions that produced the ships and their communication patterns, and
> - specification of the network extension in terms of its commuting components and its boundaries, defined by the *end-users* as a dynamic entity apart from the surrounding environment, invoking the desired changes in the information infrastructure.

The Wandering Network is based on four WLI principles:

1. Dualistic Congruence
2. Self-Reference
3. Multidimensional Feedback
4. Pulsating Metamorphosis

Next, we introduce these principles as a fundamental frame of the Wandering Network. The Dualistic Congruence describes the WLI kernel property, [21]. The next two principles are broadly used in modern software/hardware system design and network engineering. They are adopted and generalized for the purpose of this work from published research in the areas of active networking, configurable computing and adaptive systems. The fourth principle is closely related to advances in natural sciences and in particular to concepts and ideas in neurobiology and biophysics.

[12] (S*elf-)configuration* is the notion that a given system actively determines the arrangement of its constituent parts.

[13] *(Self-)reflection* in the WLI context denotes the process of monitoring and observation, partly stemming from the agency function of the individual node and partly from the reciprocal interactions with other network nodes and their environment.

1st. The Dualistic Congruence Principle

The Wandering Logic model is based on: a) the dual nature of the *ployons*, the active [mobile] network component abstractions in their two manifestations, *ships* (active mobile nodes) and *shuttles* (active gene-coded packets), and b) on their congruence.

The Dualistic Congrunce Principle (DCP) states that a ship's architecture reflects the shuttle's structure at some previous step and vice versa. Thus, *ships* are both reconfigurable computing machines and active mobile nodes in terms of hardware and software. Shuttles transport software which can activate/replace ships and their components/aggregates. A ship processing shuttles can change its state and re-configure its resources and connections *a posteriori* for further actions. In addition, it can adapt (itself) *a priori* to communications to *best-match the structure of the active packets (shuttles) at the time of delivery.* Finally, a ship can change the state of a shuttle. *Shuttles*, in turn, can be e.g. interpreted by a reconfigurable computing element inside a ship to build and/or invoke new functions. A shuttle approaching a ship can *re-configure itself* becoming a *morphing* packet to provide the desired interface and match a ship's requirements. This operation can be e.g. based on the destination address and on the class of the ship included in this address.

2nd. The Self-Reference Principle

Definition 2: The following characteristics identify a wandering network as *self-referring*:

1. Each mobiile node / ship knows best its own architecture and function, as well as *how* and *when* to display it to the external world. Ships are required to be *fair* and *cooperative* w. r .t. the information they display to the external world; otherwise they is excluded from the community.
2. Ships, are living entities: they can be born, live and die. They can also organize themselves into clusters based on one or more *feedback* mechanisms. Communication among the ships is realized through exchanging programs and data by means of *shuttles,* active packets, which may contain encoded structural information about the ships or parts of the network. Theis information can be used to maintain the operation of the network as a whole, as well as to invoke desired or necessary changes in the infrastructure via service utilization or components' feedback.
3. Each ship can acquire or *learn* other functions and extend its architecture by additional components, as well as to become a (temporary) aggregation of other nodes with a joint architecture and functionality.

The *Self-Reference Principle (SRP)* addresses the *autopoiesis* and autonomy properties of the AN elements.

3rd. The Multidimensional Feedback Principle

The *feedback principle* in network engineering is well-known in protocol design for applications such as traffic control. However, not all degrees of freedom has been exploited until now.

Active networking introduces a new paradigm for this regulation mechanism which can be spread out to *any* service, device and application in a communicating environment. The reason is that the network offers much better opportunities to address traffic issues on a *per-service* basis than the terminal devices alone. This actually corresponds to a dynamic change (re-configuration), - in fact, a *programmability* and *adaptability* (as means) to ensure *dependability*[14] (the reason) -, of the network topology and resources in **multiple dimensions**.

An active network provides a couple of means for such a solution. Here is where the multiple dimensions come from. For instance, an AN Fusion Server can be *enabled* anywhere within the network. Since each active node controls its own resources, this implies a manipulation of the traffic on a *per-(active)-node* and a *per-configuration* basis. Then, an active packet may contain some network, user or application related data starting from look-up tables and personal configurations, and ending with programs such as encoders, compilers and compiler-compilers[15] to be mounted on the destination node: the *per-(active)-packet* and *per-method* dimensions. Furthermore, merging data within the network reduces the bandwidth requirements of the users' who are located at its (low-bandwidth) periphery. Also, user-specific multicast services within the network reduce the load on the sensors and the network backbone. Therefore, a traffic adaptation on a *per-multicast* branch base is also possible.

In addition, the routers and switches of an active network perform customized computations on the messages flowing through them: the *per-message* dimension. For example, the operator of an active network could send a trace program to each router and arrange for the program to be executed under certain conditions when their packets are processed. Besides, active routers could also interoperate with legacy routers which transparently forward datagrams in the traditional manner. Addressing subsets of legacy routers for interactions defines another dimension, the *per-interoperability-task* one. Finally, the traffic processing can be customized via a set of differentiated auxiliary services on a *per-application, per-session* and even on a *per-data-link* basis in terms of OSI. The number of such interoperating feedback dimentions is virtually unlimited. The *Multidimensional Fedback Principle (MFP)* is a fundamental characteristic of self-organizing systems.

[14] This is a generalized concept and thus differs from definitions given in other areas such as fault tolerance computing.

[15] A compiler-compiler may *activate* some mechanism for network conversion based on a language framework such as attributed gr

4th. The Pulsating Metamorphosis Principle

The process of network self-creation and self-organization is referred to as *Pulsating Metamorphosis Principle (PMP).*

Definition 3: The *Pulsating Metamorphosis Principle* postulates that:

1. There are two types of moving network functionality from the center to the periphery and vice versa inside a Wandering Network referred to as *pulsating metamorphosis*: horizontal, or inter-node, and vertical, or intra-node, transition.
2. A net function can be based on one or more facts (events, experiences). The combination of net function and facts is called a *knowledge quantum* (*kq)* in the WLI model. Knowledge quanta are a new type of capsules which are distributed via shuttles in the Wandering Network. Net functions and facts can be recorded by, stored in and transmitted between the ships. They can be selectively processed inside the ships and distributed throughout the Wandering Network (WN) in an arbitrary manner.
3. Facts have a certain *lifetime* in the Wandering Network which depends on their clustering inside the ships (knowledge base), as well as from their transmission intensity, or bandwidth ("weight"). As soon as a fact does not reach its frequency threshold, it is deleted to leave space for new facts. Since net functions are based on facts, their lifetime and the lifetime of the corresponding network constellations depends on the facts. Which facts determine the presence of a particular function inside the Wandering Network is defined individually for each function. Through the exchange and generation of new facts, it is possible to modify functions to prolong their lifetime. The lifetime of a knowledge quantum is defined by the lifetime of its network function. A modification of a net function is determined by a new set of knowledge quanta.
4. A net function can emerge on its own (the *autopoiesis* principle) by getting in touch with other net functions (i.e. states and net constellations), facts, user interactions or other transmitted information. This new property of the network is called *network resonance.*
5. Network elements can encode and decode their state in *knowledge quanta*. This mechanism is called *genetic transcoding.*

The network resonance[16] is the leading WLI characteristic. *The function defines the network and vice versa.* It can be regarded as a kind of adaptive meta policy for network development. With its help, clusters and constellations of network elements or their functions can be (self-)correlated, i.e. structurally coupled[17], and/or (self-) organized in groups, classes and patterns and stored in the cache of the single nodes/ships or in the (centralized) long term memory of the network, in order to be used later as a decision base or as a development program for processes in the network (e.g. service location, customer care, billing).

The above four principles define the overall concept framework of the Wandering Network, [17]. In the following two sections we will discuss the impact of the *Viator* approach on future network architectures.

D. YET ANOTHER NETWORK-NETWORK ?

Now, how shall we induce more activity in a Wandering Network ? For instance, we could allocate different ship or active node classes depending on their role (function) within the network.

Wetherall and Tennenhouse, [23], define four basic classes of capsule mechanisms or network functions:

- *Fusion* : the active node is delivering *less* data than it receives, e.g. filtering of an MPEG-4 video stream content.

- *Fission* : the active node is delivering more data than it receives, e.g. generating additional packets for multicasting.

- *Caching* : the active node *stores* incoming data for later use upon request, e.g. storage of web pages for local processing and reducing the data flow.

- *Delegation*: the active node is performing tasks on behalf of another active node which are delegated by means of capsules, e.g. becoming a unified messaging node which migrates closer to a nomadic user while she moves.

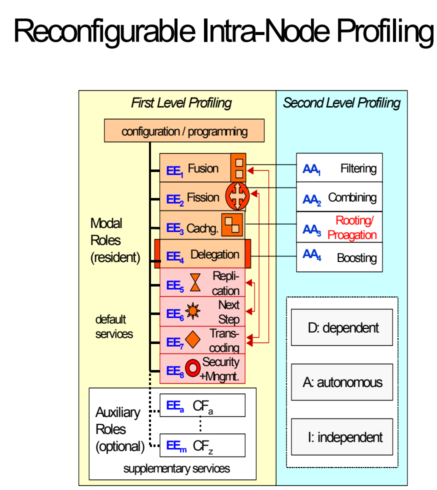

Figure 2: A ship's internal organization

[16] The analogy of this special property of the Wandering Network in biology is known as Sheldrake's theory of the Morphing Resonance, [22].

[17] "Structural coupling" is the term for structure-determined (and structure-determining) engagement of a given entity with either its environment or another entity.

Kulkarni and Minden, [24], propose seven *classes of protocols* in active networks, some of which could be regarded as specific instantiations of the ANTS capsule mechanism classes:

- ***Filtering*** (cf. fusion): packet dropping or some other kind of bandwidth reduction technique;
- ***Combining*** (cf. fission): joining packets from the same stream or from different streams;
- ***Transcoding*** : transforming user data / content into another form;
- ***Security Management*** : capsule authorization and resource access control;
- ***Network Management*** : self-configuration, self-diagnosis, self-healing[18] via event reporting, accounting, configuration management and workload monitoring;
- ***Routing Control*** : overlaying and managing several virtual topologies on top of the same physical network infrastructure as an application-layer service;
- ***Supplementary Services*** *: adding new feature to the packets* without altering, but depending on their contents, e.g. content-based buffering.

The *Viator* approach combines the above two classification schemes ([23], [24]) with the concepts of nomadic services, [20], and reconfigurable/ programmable *functional specialization* of the node/ship both in terms of hardware and software (cf. Figure 2, [17]). To retain the simplicity of the WLI model, we postulate that each active node (or *ship*) can be assigned exactly one single function at a time. Thus, an active node could behave e.g. as a fusion server during a session, and then become a network cache proxy for another. Furthermore, we distinguish between *modal* (basic) functions resident at each node/ship and *auxiliary* (optional) ones that can be transported, installed and enabled via capsules/shuttles to be later customized by the user. By default, we consider that each function is assigned a single "registry" execution environment (EE) with the modal functions being priorized for access.

Viator regards this role distribution (except the routing control) on the time scale as a *horizontal inter-node functional wandering (self-organization)* of the active nodes (ships), cf. Figure 3. We call the capsule mechanisms (functions) identified by Whetherall and Tennenhouse "First Level Profiling", and the protocol classes (functions) of Kulkarni and Minden – "Second Level Profiling".

In *Viator*, routing control is considered as a special class of *virtual vertical intra-node overlay of functional wandering (self-organization),* cf. Figure 4. This class is interdependent from all of the other functional classes (node roles). For instance, we can generate a QoS oriented network topology on demand.

Furthermore, we combined the security and network management classes into one single class, cf. Figure 2.

[18] A *self-healing* network is a fault-tollerant network which adapts automatically to defects in its node connectivity, functional specialization and performance disturbances to provide the best possible level of service (QoS) to the communicating parties. Self-healing in the WLI context implies reflection (monitoring) and detection of service facility and hardware failures, automatical re-routing around the failure, as well as automatic aggregation and reconstruction of the disrupted functionality.

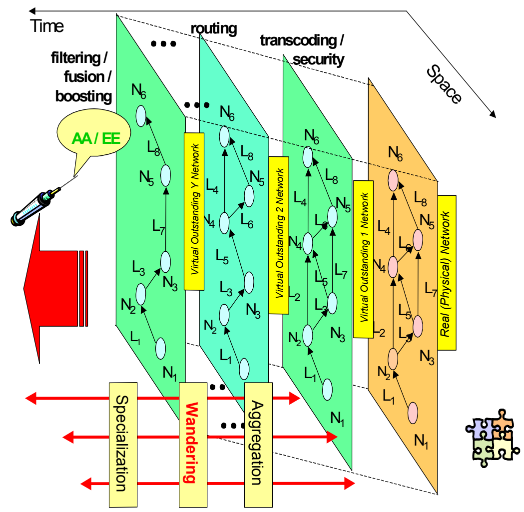

Figure 3: Horizontal network wandering[19]

Also, in order to address the performance enhancements, we included the protocol boosters as an additional class to the categorization of Kultarni and Minden along with an instantiation of the delegation mechanism of Wetherall and Tennenhouse. Finally, we assigned two additional roles to the First Level Profiling: *Replication* and *Next-Step* for packet/function replication and ship state description respectively.

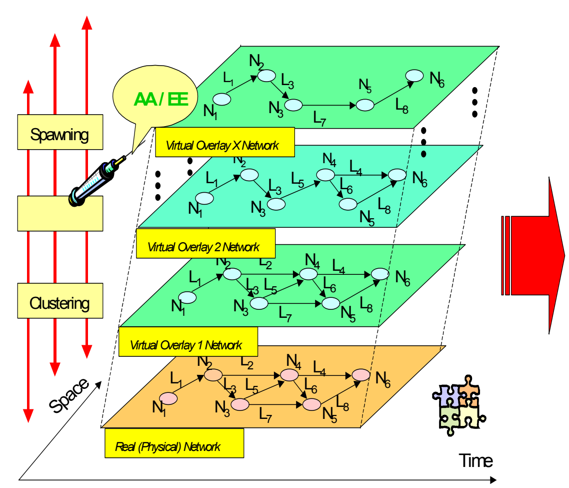

Figure 4: Vertical network wandering[20]

The first two roles of the First Level Profiling correspond partially to the functions "Forward and Copy" (FaC) and "Oracle" suggested by Raz and Shavitt, [25] to enhance the AN architecture framework.

[19] (ex-pulsing) - inter-node functional autopoiesis generating virtual outstanding networks of the same physical infrastructure

[20] (in-pulsing) – *intra*-node functional *autopoiesis* generating virtual *overlay* networks over the same physical infrastructure

A capsule/shuttle replication could be quite useful for deploying knowledge-based services such as selective "activation" of the network topology and thus adding an additional level of flexibility to the AN model (e.g. to change a node's routing algorithm and/or table). The Next-Step function operates as an internal programmable switch which stores the next node role to come. It is a standard module for each node/ship. Since most of the network traffic carries large amounts of rich multimedia content, a *transcoding* function for congestion control and local, feedback-enabled content-, user- and resource-dependent QoS management is also useful. To complete the model, routing and propagation of functionality were included in the Second Level Profiling as dependants of the caching class which refers in turn as a bootstrapping mechanism to the node state (Next Step) and all other instances of the functional classes in the First Level Profiling.

The above architecture and mechanisms open a new dimension of systemic hyperactivity in networking. The two schemes of functional autopoiesis on figures 3 and 4, horizontal inter-node and vertical intra-node wandering , are operating in parallel to realize an adaptive virtual topology by utilizing the *Pulsating Methamorphosis Principle (PMP).*

E. SUMMARY AND OUTLOOK

The applications of the Wandering Network model for multimedia communications are numerous. For instance, adaptive QoS management and routing in ad-hoc mobile networks is one of them, [15]. In particular, we applied the WLI model framework for the formal specification and verification of a generic adaptive routing protocol for active ad-hoc wireless networks. We are satisfied with the results which delivered four DIN A4 pages of bug-free TLA^+ code, [33], with Lamport's TLC model checker, [34], running on top of SUN's JRE-SE-1.3.1 under Microsoft's Windows 2000 within a man-month. This work is for further study.

The Wandering Logic Intelligence (WLI) is an open, hierarchical and dynamically structured model which allows to address specific problems in communications architectures, services and multimedia applications with a great degree of differentiation and flexibility.

The Wandering Network is a new type of active mobile architecture for adaptive communications defined by the following characteristics:

- flexible, multi-modal[21] specialization of network nodes as virtual subnetworks;
- mobility and virtualization of the net functions as hardware und software;
- self-organization as multi-feedback-based topology-on-demand.

[21] The single network nodes can execute multiple functions, and thus perform multiple roles in the network, in parallel, such as e.g. a protokol booster, fusion server, etc., as sub-classes of the generic roles: server, client and agent [17]. These functions can be realized as programmable software of configurable, i.e. resident, or plug-and-play hardware. Therefore we talk of a programmable network topology in WLI.

The *Viator* approach defines the constitution of a Wandering Network based on the four WLI principles: *Dualistic Congruence*, *Self-Reference*, *Multidimensional Feedback* and *Pulsating Metamorphosis*.

The *Viator* model comprises a self-organizing[22] mobile active network with both nodes (ships) and packets (shuttles) being *active* (i.e. executable), exchangeable, *re-configurable and programmable* (down to the level of gate components). A ship processing the shuttle contents can change its state and re-configure its resources and connections to perform accordingly. Similarly, a shuttle approaching a ship can re-configure itself becoming a *morphing* packet to provide the desired interface at the dock and match a ship's requirements. This operation can be based on the destination address and on the class of the ship included in this address. The assumption in this case is that the sender ship was not taking care about arranging this procedure for the shuttle.

The *Wandering Network* differs from the well-known research frameworks of Open Signaling and Active or Programmable Networks essentially by two characteristics:

- adaptive function migration, and
- pulsating metamorphosis.

The WN elements have a temporal character; they can be created, configured and removed. Functions can change their hosts (ships), wander and settle down in other hosts, thus creating a valuable statistics about the frequency of usage of wandering functions in the network[23]. The results obtained after a careful evaluation of this data can be used for the design of new network architectures and topologies. The main distinction from other AN approaches elsewhere is that the active nodes (ships) are considered to be *mobile* and *configurable* both in terms of software and hardware.

The essential contributions of the WLI model and the resulting Wandering Network are summarized as follows:

1. Role Change: The *role* of the network node within a particular virtual architecture can change during its operation. The new functionality is either resident on the node and waiting to be activated, i.e. it is not yet involved in the next step virtual scheme, or transferred via active networking to the destination node.

2. Parallel Roles: The execution of the parts of a distributed algorithm can be performed within the different roles of an active node's / ship's, configuration.

3. Node Genesis ("N"-geneering): encoding and embedding the structural information about a mobile node, the ship, and its environment into the executable part of the active packets, the shuttles.

[22] It is essential, that the Wandering Network is based on the principle of a*utopoiesis* [9], or self-organization and self-creation which is determined through the exploitation of the network by the user, through the adaptive mobility of its nodes and terminals and through the target analysis and evaluation of the "experiences" of the actively adaptive network elements, which may even lead to a new life cycle of creation and organization of new functions and protocols in the network. The WLI approach is not the intended distribution of fixed, injected, programmable, or even "elastic" control functions, [32], inside or outside the network.

[23] e.g. in connection with the maintenance of a Virtual Home Environment for end users.

Finally, the Wandering Network represents an aggregate and further development of concepts pursued elsewhere in the research community. We deeply appreciate our colleagues' efforts and contributions worldwide and strongly believe that the *Viator* approach will become another step towards the realization of an evolutionary global *vivid* Active Network, as it has been percieved by the pioneers of this idea, [35].

F. REFERENCES

[1] Gregory Bateson, "*Steps to an Ecology of Mind*", Ballantine Books, New York, 1972, (reprint University of Chicago Press, 2000), ISBN 0-226-03905-6.

[2] Bateson, G., "*Mind and Nature: A Necessary Unity*", Ballantine Books, New York, 1979.

[3] Tennenhouse, D., "What's New at DARPA/ITO ?*"*, July 28th, 1999, http://www.cra.org/Activities/snowbird/ slides/ tennenhouse/index.html.

[4] Tennenhouse, D. et al*., "*A Survey of Active Network Research*", IEEE Comm. Mag.*, January 1997, pp. 80-86.

[5] Calvert, K. L. et al., "Directions in Active Networks," *IEEE Communications Mag.,* Oct. 1998, pp. 72-78.

[6] Campbell, A. et al., *"*A Survey of Programmable Networks*"*, http://www.columbia.comet.edu.

[7] Faynberg, I. *et al.*, "The Intelligent Network Standards", McGraw-Hill, 1997, ISBN 0-07-021422-0.

[8] Chen, T., Jackson, A. W., (Eds), "Commentaries on Active Networks and End-to-End Arguments", *IEEE Network*, May/June 1998, pp. 66-71.

[9] Maturana, H. R., Varela, F. J., "*Autopoiesis and Cognition*", D. Reidel Pub. Co., 1980, ISBN 9027710163.

[10] Compton, K., Hauck, S., "Reconfigurable Computing: A Survey of Systems and Software", Northwestern University, Dept of ECE, *Tech. Report*, 1999.

[11] Bondalapati, K., Prasanna, V. K., "Reconfigurable Computing: Architectures, Models and Algorithms", Tech. Report, USC, Dept. of EE, http://maarc.usc.edu/.

[12] DARPA Information Technology Office, *"Adaptive Computing Systems Program"*, http://www.darpa.mil/ ito/acs.

[13] Simeonov, P. L., Miloucheva I., K. Rebensburg, K.J. Turner, A. J. M. Donaldson,*, "*Prototype Performance Evaluation of Multimedia Service Components", *Proc. of ICCCN'94,* September 11 - 14, 1994, pp. 254-261.

[14] Spaniol, O., Meggers, J., *"*Active Network Nodes for Adaptive Multimedia Communication*"*, *Proc. of SMARTNET'99*, Nov. 1999.

[15] Simeonov, P. L. (Ed.), "*MediaPEP* (Media Performance Enhanced Proxy), A Scaleable and Dependable *ActiveUMTS* QoS Management Server, An Internet Protocol Booster and An Adaptive Media Transcoder Switch Architecture for E2E-IP Integration of Mobile Wireless Interactive Telepresence Applications" (The "Odysseus-2001" Project), *White Paper*, November, 2000, http://www.prakinf.tu-ilmenau.de/IPI/FGT/.

[16] Wetherall, D., J., Guttag, J., Tennenhouse, D., L., "ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols*", Proc. of IEEE OPENARCH'98*, San Francisco, CA, April 1998.

[17] Simeonov, P. L., "The Wandering Network, a Glance at an Evolving Reality" (in German: "Das Wandernde Netzwerk"), Proc. of *Netobjectdays'2001*, 2. Joint GI conference "Object Orieneted Programming for a Networked World", 11-13 September, 2001, Erfurt, Germany, http://www.prakinf.tu-ilmenau.de/IPI/FGT/.

[18] Simeonov, P. L., Hofmann, P., "*A Distributed Intelligent Computer/Telephony Network Integration Architecture for Unified Media Communication"*, Proc. of 2IN'97 (IFIP), Paris, September 2-5, 1997, pp. 3-8, Chapman & Hall, D. Gaiti (Ed.), ISBN 0-412-82950-9.

[19] Simeonov, P. L., "*The SmartNet Architecture or Pushing Networks beyond Intelligence"*, Proc. of ICIN'98 (5th International Conference on Intelligent Networks), 12-15 May, Bordeaux, France, pp. 182-185.

[20] Simeonov, P. L., "*On Using Nomadic Services for Distributed Intelligence"*, Proc. of ICCCN'99, October 11 - 13, 1999, Boston, MA, USA, pp. 228-231, IEEE Press, ISBN: 0-7803-5794-9, http://www.icccn99.cstp.umkc.edu/; also in the Journal of Microprocessors and Microsystems, Elsevier Science Publishers, October, 2000.

[21] Simeonov, P. L., "A Path Towards the Wandering Logic of Intelligence", *Proc. of PONMS'99* (Modal & Temporal Logic based Planning for Open Networked Multimedia Systems), AAAI 1999 Fall Symposium Series, November 5-7, 1999, North Falmouth, MA, USA, Working Notes, pp. 78-84, AAAI Press.

[22] Sheldrake, R., "*A New Science of Life: The Hypothesis of Morphing Resonance*", London, Blond & Briggs, 1981, ISBN 0-89281-535-3.

[23] Wetherall, D., Tennenhouse, D., *"*The ACTIVE IP Option*", Proc. 7th ACM SIGOPS Euro. Workshop*., Connemara, Ireland, 1996.

[24] Kulkarni, A., B., Mindem, G. J., "Active Networking Services for Wired/Wireless Networks*", Proc. INFOCOM'99*, pp. 1116-1123, New York, USA, 1999, http://www.ieee-infocom.org/1999/program.html.

[25] Raz, D., Shavitt, Y., "Active Networks for Efficient Distributed Network Management*", IEEE Comm. Mag.*, Vol. 38, No. 3, March 2000.

[26] Simeonov, P. L. and M. Plessow, "*Netlike Schematics and Their Structural Description"*. Proc. of the VII Workshop on Informatics in Industrial Automation. ZKI, Akademie der Wissenschaften, Berlin, 31 Oct. - 4. Nov. 1989, pp. 144-161.

[27] Hoffman, D., *"Visual Intelligence",* W. W. Norton & Co. Inc., 1998, ISBN 0-393-04669-9.

[28] Tschudin, C., "Active Network Overlay Network (ANON)", *RFC Draft*, December, 1997.

[29] Van der Merwe, J. E., Rooney, S., Leslie, I. M., Crosby, S. A., "The Tempest – A Practical Framework for Network Programmability", *IEEE Network*, November, 1997.

[30] Campbel, A. T., De Meer, H. G., Kounavis, M. E., Miki, K., Vicente, J. B., Villela, D., « The Genesis Kernel : A Virtual Network Operating System for Spawning Network Architectures », Proc. of OPENARCH'99, New York, 1999.

[31] Reschke, D., Jungmann, M., Paradies, Th.,, "Component-oriented approach for hypermedia-based knowledge management applications", *Proc. International Forum on Multimedia & Image Processing* (IFMIP 98), Anchorage, Alaska, Mai 1998.

[32] Bos, H., Isaacs, R., Mortier, R., Leslie, I., « Elastic Network Control : An Alternative to Active Networks », Jour. Of Comm. And Networks, Vol. XX, No. Y, March, 2001.

[33] Lamport, L., "The Temporal Logic of Actions", ACM Toplas, 16, 3, pp. 872-923, May, 1994; also in: http://www.research.digital.com/SRC/personal/lamport/tla/.

[34] Lamport, L. "Specifying Systems", Draft, August 2001, http://www.research.compaq.com/SRC/personal/lamport/tla/book.html.

[35] Tennenhouse, D. L. *et al.*, « From Internet to ActiveNet », *RFC Draft*, January, 1996.