# Consistency verification of stateful firewalls is not harder than the stateless case

LEVENTE BUTTYÁN, GÁBOR PÉK, TA VINH THONG

*Laboratory of Cryptography and Systems Security*
*Budapest University of Technology and Economics, Department of Telecommunications*
*{buttyan, pek, thong}@crysys.hu*

*Keywords: ?????*

**Firewalls play an important role in the enforcement of access control policies in contemporary networks. However, firewalls are effective only if they are configured correctly such that their access control rules are consistent and the firewall indeed implements the intended access control policy. Unfortunately, due to the potentially large number of rules and their complex relationships with each other, the task of firewall configuration is notoriously error-prone, and in practice, firewalls are often misconfigured leaving security holes in the protection system. In this paper, we address the problem of consistency verification of stateful firewalls that keep track of already existing connections. For the first sight, the consistency verification of stateful firewalls appears to be harder than that of stateless firewalls. We show that, in fact, this is not the case: consistency verification of stateful firewalls can be reduced to the stateless case, and hence, they have the same complexity. We also report on our prototype implemetation of an automated consistency verification tool that can handle stateful firewalls.**

## 1. Introduction

Firewalls are the cornerstones of improving the security of enterprise networks. Simple packet filter firewalls work in a stateless manner: they inspect the packets passing through the perimeter of the network as independent objects, and decide to accept or deny them according to a predefined static ruleset. Modern firewalls, however, are more complex and perform stateful packet inspection: they keep track of the already existing connections and they decide about the fate of a packet based on both its header information and the state of the connection that it belongs to.

In practice, firewalls are often misconfigured. Misconfiguration errors result in inconsistencies in the firewall [1,7]. An example for a critical inconsistency is when all the packets that are intended to be denied by a given rule of the firewall are accepted by some preceding rules. This is called shadowing, because the intended effect of the deny rule is cancelled by the preceding accept rules. Shadowing is a critical inconsistency, because it is likely that the deny rule is there to stop some well-known malicious traffic, however, due to the shadowing, that traffic is not actually stopped by the firewall. Such inconsistencies can easily occur when the firewall has a distributed implementation and/or when it is managed by multiple administrators; both being frequent cases in large organizations.

Checking a large firewall (i.e., hundreds of access control rules) for inconsistencies is difficult and prone to errors when it is done in an ad-hoc, non-systematic manner. Thus, several formal methods and automated tools have been proposed in the literature [1-12], but essentially all of them were designed for finding inconsistencies in stateless firewalls. However, stateful firewalls are much more broadly used nowadays due to their connection tracking feature. Finding inconsistencies in stateful firewalls has been considered to be harder than the stateless case due essentially to the potentially very large size of the state space. A first attempt to model stateful properties of firewalls is presented in [13], but that work does not propose any method to find inconsistencies in stateful firewalls.

In this paper, we propose a modeling technique for states and, for the first time, a systematic method for detecting inconsistencies in stateful firewalls. We model a state as a particular subset of the firewall ruleset that consists of all the static rules and those dynamic rules that are relevant in the given state. We show that the number of inconsistencies in any state cannot be larger than the number of inconsistencies in the designated state that includes all dynamic rules. Hence, it is sufficient to check that single designated state for inconsistencies: if no inconsistency is found in that state, then no other state can contain any inconsistencies. Moreover, if the designated state is not free of inconsistencies, then this fact proves that the firewall is inconsistent in at least one state (the designated one). Therefore, we essentially reduce the consistency verification of a stateful firewall to the consistency verification of a single static ruleset.

In order to automate the verification, we implemented a software tool in C#, which is capable of finding inconsistencies in the configuration of stateful firewalls. Our tool is based on FIREMAN [7], an approach which was originally developed for stateless firewalls. We note, however, that the real power of our approach is that *any* stateless tool could have been used. We used the

FIREMAN approach beacuse it uses Binary Decision Diagrams (BDD) for handling IP range set operations, such as intersection and union, and BDDs are conceptually simple and very efficient.

The rest of the paper is organized as follows: We define inconsistencies and inefficiencies in Section 2. We give a brief overview of the connection-tracking feature of contemporary firewalls in Section 3. In Section 4 we introduce our main theorems and their proofs, while Section 5 reports on our implementation. Finally, we conclude the paper and give some future plans in Section 6.

## 2. Inconsistencies and inefficiencies in firewalls

The configuration of a firewall consists in the ruleset that the firewall uses for filtering the traffic. A stateless rule is represented in the form *<P, action>*, where *P* corresponds to a predicate describing the criteria that a packet has to meet to match the rule, and *action* is the corresponding action that is executed when there is a match to the rule. In case of stateless rules, predicate *P* can be represented as a 5-tuple *(prot, srcaddr, srcport, dstaddr, dstport)*, where *prot* refers to a protocol (tcp, udp, icmp), *srcaddr* is the source IP address range, *srcport* is the source port range, *dstaddr* is the destination IP address range, and *dstport* is the destination port range that should be matched by a packet. In addition, an action can be *accept* or *deny*, the meaning of which should be intuitively clear.

The ruleset of a firewall may be inconsistent and/or inefficient. In this paper, we consider three types of inconsistencies: *shadowing, generalization,* and *correlation;* and one inefficiency: *redundancy*. These have also been considered in prior works of others [1,7], but in a stateless environment. In Section 4, we show that they can be defined in the case of stateful firewalls as well. In this section, we give the definitions of these inconsistencies and inefficiencies.

We use the following notation: Let *R* be a ruleset that consists of stateless rules $r_i = <P_i , action_i>$, where $P_i = (prot_i, srcaddr_i, srcport_i, dstaddr_i, dstport_i)$. We say that (*Pj* ⊆ *Pi*) if $(prot_i \subseteq prot_j) \wedge (srcaddr_i \subseteq srcaddr_j) \wedge (srcport_i \subseteq srcport_j) \wedge (dstaddr_i \subseteq dstaddr_j) \wedge (dstport_i \subseteq dstport_j)$. Similarly, $(P_j \cap P_i \neq 0)$ if $(prot_i \cap prot_j \neq 0) \wedge (srcaddr_i \cap srcaddr_j \neq 0) \wedge (srcport_i \cap srcport_j \neq 0) \wedge (dstaddr_i \cap dstaddr_j \neq 0) \wedge (dstport_i \cap dstport_j \neq 0)$.

A rule is shadowed by a preceding rule if it is a subset of the preceding rule; and the two rules define different actions:

**Definition (Shadowing)** *Rule* $(r_i = <P_i, action_i >) \in R$ *shadows rule* $(r_j = <P_j, action_j >) \in R$ *if and only if* $(i < j) \wedge (P_j \subseteq P_i) \wedge (action_i \neq action_j)$, where *i* and *j* denote the order of rules in a ruleset *R*.

A rule is a generalization of a preceding rule if it is a superset of the preceding rule and the two rules define different actions:

**Definition (Generalization)** *Rule* $(r_i = <P_i, action_i >) \in R$ *is the generalization of rule* $(r_j = <P_j, action_j >) \in R$ *if and only if* $(i > j) \wedge (P_j \subseteq P_i) \wedge (action_i \neq action_j)$.

Two rules are correlating if their intersection is not empty, they are not related by the superset or subset relations, and they define different actions. Packets that match the intersection will take the action of the preceding rule:

**Definition (Correlation)** *Rule* $(r_i = <P_i, action_i >) \in R$ *and* $(r_j = <P_j, action_j >) \in R$ *are correlating if and only if* $(P_j \cap P_i \neq 0) \wedge (P_j \not\subseteq P_i) \wedge (P_i \not\subseteq P_j) \wedge (action_i \neq action_j)$.

A rule is redundant if the removal of it would not affect the operation of the firewall. In case of masked redundancy (defined below) the successor rule is unnecessary, while in case of partially masked redundancy (also defined below) the preceding rule is unnecessary:

**Definition (Redundancy)** *Rule* $(r_i = <P_i, action_i >) \in R$ *is redundant with respect to rule* $(r_j = <P_j, action_j >) \in R$ *if and only if at least one of the following two conditions are satisfied:*

*Masked redundancy:*

$(P_i \subseteq P_j)$, *where* $(i > j) \wedge (action_i = action_j)$

*Partially masked redundancy:*

$(P_i \subseteq P_j)$, *where* $(i < j) \wedge (action_i = action_j)$

Note that not all these inconsistencies and redundancies are equally critical. Usually, only shadowing is considered to be a configuration error, while generalization and correlation are in fact often used by firewall administrators to make a ruleset compact. Nevertheless, it may be the case that some of the generalizations and correlations are not intentional, in which case, it is useful to detect them and let the administrator decide if they are harmful or not. Redundancy is not considered a serious configuration error either, but redundant rules are clearly useless, therefore, it is worth identifying and removing them, and increasing the efficiency of filtering by doing so.

## 3. Connection-tracking with iptables

In order to understand the model described in the next section, we shortly review how connection-tracking works in iptables, a stateful firewall that we used in our work. Other stateful firewalls work in a similar manner.

Iptables defines tables and chains to complete certain operations on packets at different points of the checking. We consider only the input and output chains, and the filter table for demonstration purposes. An extensive description of iptables can be found in [15].

Connection-tracking is the basis of stateful firewalls. It refers to the ability to maintain state information about a connection as an entry in a state table. Entries are inserted in and removed from the state table according to the packets the firewall is examining. For instance, we demonstrate how connection-tracking tracks a TCP connection establishment.

Suppose we have the following rules in the output and input chains of the filter tables, respectively:

1. *iptables –A OUTPUT -p tcp -m state –state NEW, ESTABLISHED -j ACCEPT*;
2. *iptables –A INPUT -p tcp -m state –state ESTABLISHED -j ACCEPT*.

Connection-tracking classifies each packet as being in different states: *NEW* (if the packet initiates a new connection), *ESTABLISHED* (if the packet is associated with a connection that has encountered packets in both directions), *RELATED* (if the packet initiates a new connection, but also associated with an already established connection), *INVALID* (not part of an existing connection). For instance, the second rule above means that only packets that belong to an established connection are permitted to enter the network.

Once a *syn* packet that initiates a TCP connection is sent in the output chain, and accepted by the first rule above that allows a *NEW* connection, the following connection table entry is created:

*tcp 6 54 SYN_SENT src=10.0.0.1 dst=154.32.43.44 sport=1506 dport=22 [UNREPLIED] src=154.32.43.44 dst=10.0.0.1 sport=22 dport=1506 use=1*

Here, *tcp* refers to the protocol of the connection (and 6 is its numerical form), the remaining time before removal of this entry is 54 seconds, *SYN_SENT* is the tcp state of the connection, *src* and *dst* are the source and destination IP addresses, *sport* and *dport* are the source and destination ports of the connection, and *UNREPLIED* refers to the connection-tracking state of the connection. In the following, the addresses and ports are listed in reverse order for the response traffic.

When a *syn+ack* packet arrives, the entry in the connection tracking table is modified as follows:

*tcp 6 60 SYN_RCVD src=10.0.0.1 dst=154.32.43.44 sport=1506 dport=22 src=154.32.43.44 dst=10.0.0.1 sport=22 dport=1506 use=1*

One can see that the TCP connection state changes to *SYN_RCVD*, while the tracked connection-state changes from *NEW* to *ESTABLISHED*. Note that the tracked connection states (*NEW, ESTABLISHED,* etc.) are different from the TCP connection establishment states (*SYN_SENT, SYN_RCVD,* etc.).

Finally, when the last part of the three-way TCP connection establishment handshake, an *ack* packet arrives from the server, the connection-tracking entry becomes:

*tcp 6 43 1995 ESTABLISHED src=10.0.0.1 dst=154.32.43.44 sport=1506 dport=22 [ASSURED] src=154.32.43.44 dst=10.0.0.1 sport=22 dport=1506 use=1*

The TCP state of the connection is altered to *ESTABLISHED* and the connection-tracking state of the connection is modified to *ASSURED*. *ASSURED* connections are not dropped from the state table when the connection is overloaded. Note that the remaining time value is increased to a previously defined timeout value.

## 4. Verification of stateful firewalls

In case of a stateless firewall, inconsistencies and inefficiencies between rules can be detected by means of static analysis of the ruleset. In case of a stateful firewall, the detection appears to be harder, because the static analysis has to be performed in all possible states of the firewall in order to be sure that the ruleset always remains consistent. In this section, we show that this is indeed not the case, and it is sufficient to verify a single designated state for inconsistencies in order to prove that the firewall's rule set is consistent in all possible states.

For doing so, we must first introduce the notion of *firewall state*:

**Definition (Stateful rules)** A firewall rule is said to be stateful if it defines state information, and is presented in the form *<P, action, stateinfo>*.

**Definition (Firewall state)** *The states of a firewall includes all the static firewall rules and those dynamic (stateful) rules that have an associated entry in the connection-tracking table.*

As an example, let us consider the following rule set, where the first three rules are dynamic (stateful) rules and the fourth rule is a static rule:

Rule 1: iptables –A OUTPUT –s 10.0.0.1 –dport 80 –m state – state NEW, ESTABLISHED –j ACCEPT
Rule 2: iptables –A OUTPUT –s 10.0.0.1 –dport 443 –m state – state NEW, ESTABLISHED –j ACCEPT
Rule 3: iptables –A OUTPUT –s 10.0.0.1 –dport 22 –m state – state NEW, ESTABLISHED –j ACCEPT
Rule 4: iptables –A OUTPUT –s 10.0.0.1 –dport 22 –j DROP

In addition, let us suppose that the following two entries have been created in the connection-tracking table (as the result of processing some packets earlier):

1. tcp 6 54 SYN_SENT src=10.0.0.1 dst=154.32.43.44 sport=6322 dport=80 [UNREPLIED] src=154.32.43.44 dst=10.0.0.1 sport=80 dport=6322 use=1
2. tcp 6 432 ESTABLISHED src=10.0.0.1 dst=154.32.43.44 sport=1506 dport=443 [ASSURED] src=154.32.43.44 dst=10.0.0.1 sport=443 dport=1506 use=1

As one can see, in this state, Rules 1 and 2 have associated entries in the connection-tracking table, while Rule 3 has no such entry. This means that in this state, no packet can match Rule 3, and therefore, it can be ignored. At the same time, packets may match Rules 1 and 2, due to the entries in the connection-tracking table, and packets may also match Rule 4, as it is a static rule (i.e., independent of any states). For this reason, Rules 1, 2, and 4 must be considered in this parti-

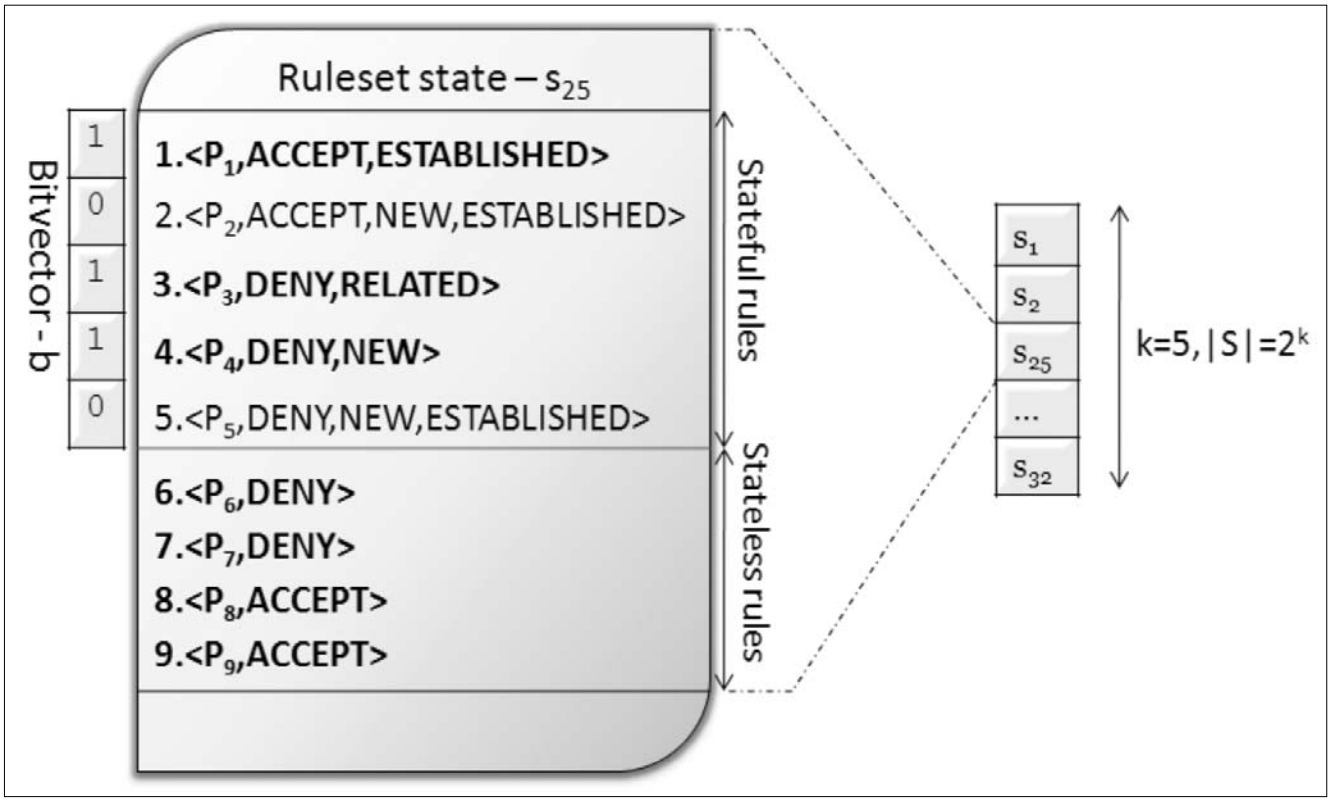



*Figure 1. Encoding the firewall state as a binary vector*

cular state. This means that, essentially, the state of the firewall can be represented by those three rules.

It is natural to encode such a firewall state as a binary vector the length of which is equal to the number $k$ of the dynamic (stateful) rules in the rule-set. This is illustrated in *Figure 1*. It trivially follows that the number of all possible firewall states is $2^k$.

We can now introduce a partial ordering $\leq$ on the set $S$ of all possble states:

**Definition (Partial ordering of firewall states)** *Let s and s' be two states of the same firewall (i.e., two binary vectors of the same length). We have $s \leq s'$ if all the dynamic rules that are included in s are also included in s'. In other words, if the i-th element of the binary vector corresponding to s is 1, then the i-th element of the binary vector corresponding to s' is also 1.*

The key idea of our work is that we show that whenever $s \leq s'$, the number of inconsistencies in $s$ cannot be larger than the number of inconsistencies in $s'$. The next theorem states this for the number of shadowings:

**Theorem (Shadowing)** *Let s and s' be two states of the same firewall such that $s \leq s'$. The number of shadowings in s cannot be larger then the number of shadowings in s'.*

**Proof:** Without loss of generality, we can assume that the $i$-th stateful rule $r_i$ of the firewall rule-set is included in $s'$, otherwise $s'$ contains no stateful rules, which means that $s = s'$, and the statement of the theorem follows trivially. Let us denote by $s''$ the state that we obtain from $s'$ by removing rule $r_i$.

Now, if there exists a (stateful or stateless) rule $r_j$ of the same firewall, such that either $r_i$ shadows $r_j$ or $r_j$ shadows $r_i$, then removing $r_i$ from $s'$ and obtaining $s''$ surely decreases the number of shadowings. Otherwise, if no rule shadows $r_i$ and no rule is shadowed by $r_i$, then removing $r_i$ from $s'$ and obtaining $s''$ does not affect the number of shadowings.

As state $s$ can be obtained from $s'$ by iteratively removing from $s'$ the dynamic rules that are not contained in $s$, the statement of the theorem can be obtained by iteratively using the above argument. ♣

Similar theorems can be stated and proven in the same way for the other types of inconsistencies and inefficiencies (see [14] for details). This leads to the following main theorem:

**Theorem (Reduction to the stateless case)** *Let $s_{all\text{-}1}$ be the state that contains all dynamic rules of the rule set. If no inconsistencies and inefficiencies exist in state $s_{all\text{-}1}$, then all states are free from inconsistencies and inefficiencies, and hence, the firewall configuration is correct.*

**Proof:** Immediately follows from the fact that $s \leq s_{all\text{-}1}$ for any state $s$ of the firewall. ♣

The consequence is that it is sufficient to verify the firewall in state $s_{all\text{-}1}$ for inconsistencies, and this can be done by using any static analysis tool.

Note that for the sake of this static analysis, the dynamic rules are converted to static rules by ignoring those parts of their predicate that refer to some state information.

## 5. Implementation

In our implementation, we used the approach called FIREMAN [7], which applies static analysis techniques to check misconfigurations, such as policy violations, inconsistencies and inefficiencies in individual firewalls as well as in distributed firewalls using symbolic model checking and Binary Decision Diagrams (BDD). Based on the concepts of FIREMAN, we implemented the methodology of stateful verification described in the previous chapter as a software tool. In the rest of this section, we briefly explain the operation of FIREMAN, and hence, our tool.

Inspired by the successfully applied software implementations of the previous works [1,7] a new application was implemented in C# that is capable of verifying a stateful firewall configuration. First of all, this tool builds upon the methodology of the aforementioned works, but uses its own Binary Decision Diagram class, to make calculations (union, intersection, subset) on IP ranges as quickly as possible.

Binary Decision Diagram is a data structure which can represent Boolean functions. It is a rooted, acyclic, directed graph which comprises several non-terminal (decision) nodes and terminal nodes with assigned value either 1 (the Boolean function is true) or 0 (the Boolean function is false). When an IP range is presented in BDD form, the number of non-terminal nodes is given by the length of the corresponding netmask. Each decision node is one of the variables of the Boolean function. Interested redears are referred to [7,14] for more details and examples.

In the following, the functioning of the application is presented. First and foremost, a valid iptables rule file has to be opened. Right after it, the application parses the rules of the file one-by-one, and tries to recognize the given parameters and their values. If a suggested parameter is not set, then default values are used instead.

An example is when one does not specify explicitly the destination port in a corresponding rule. In this case, all packets carrying one of the valid ports in the range [1, 65535] are accepted. When one rule is parsed then it is compared against with the already stored preceding rules at once. This is the task of the static analysis method that was previously mentioned. Naturally, the trivial translation of stateful rules into stateless ones is done when the current rule has state information.

According to the definition of firewall state, there is no need to distinguish rules with different state information (*NEW, ESTABLISHED, RELATED,* etc.), so they are handled uniformly. As it was explained previously, there is only one state $s_{all\text{-}1}$, which contains all the stateful (and stateless rules), that has to be checked. There are two internal lists defined, where the parsed rules are put: AcceptList and DenyList, where AcceptList contains rules with action *ACCEPT* and DenyList stores rules with action *DENY*. Note that iptables defines two declining action values: *REJECT* and *DROP*. In fact each of them refers to the internal representation of *DENY*.

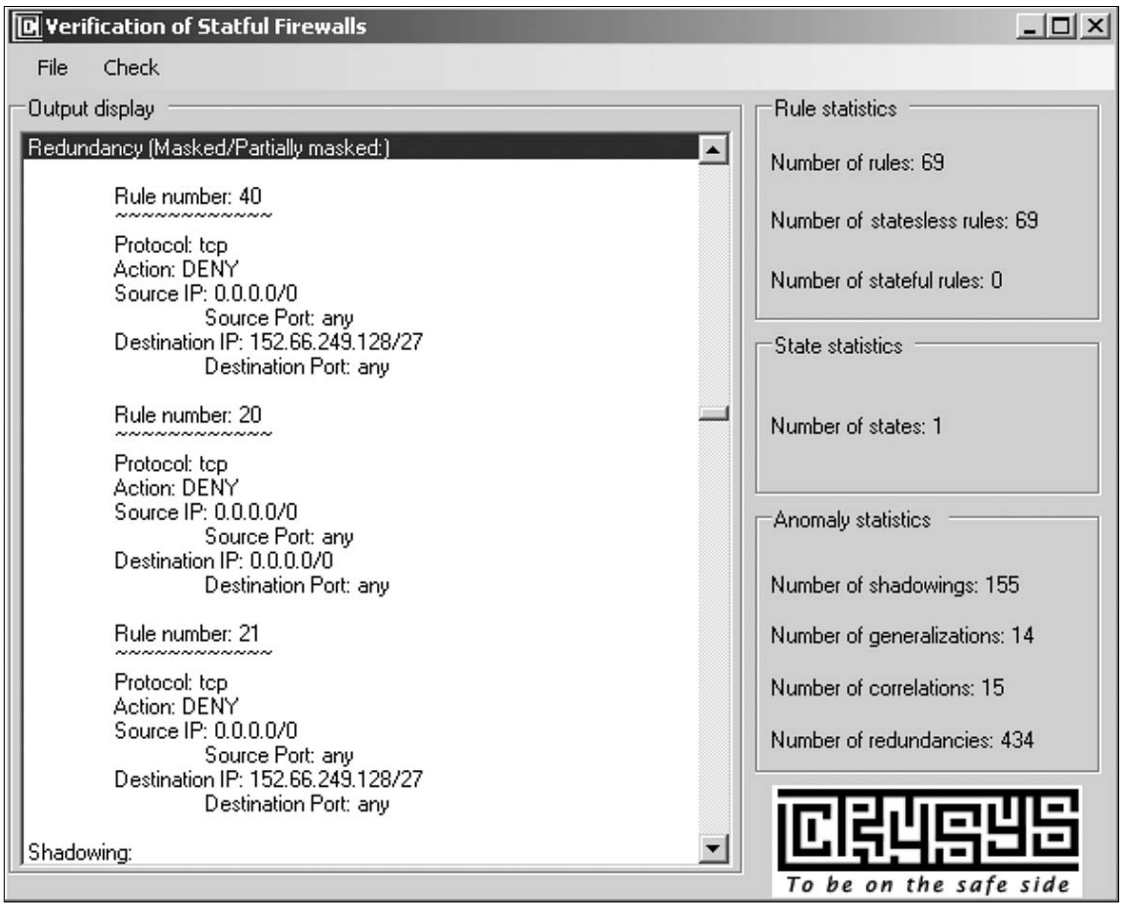



*Figure 2. Screen shot of the prototype implementation*

The pseudo-code in *Table 1* demonstrates the core of stateful verification.

```
StatefulVerfication(String firewallRuleFile){
    struct Rule {
        Boolean isStateful; Boolean isInBitVector; String protocol;
        BDD sourceIP;
        BDD destinationIP; Port sourcePort;
        Port destiantionPort; String action;
        String stateInformation; Integer numberOfRule;
    }
    Rule ruleSet[NUMBER_OF_RULES];
    ruleset = MakeInternalRepresentation(firewallRuleFile);
    forall (Rule rule in ruleset) {
        if ( rule.isStateful == true) {
            rule.isInBitVector = true;
        }
    }
    RunStatelessAnalysis(ruleSet);
}
```

*Table 1. The pseudo code of stateful verification*

It is essential to put efforts on the demonstration of the application by verifying firewalls that are used in practice. In order to satisfy these kind of requirements two firewalls at BME are analyzed by means of the implemented tool. The machine that we used for the verification was an IBM Thinkpad R40 notebook with Intel Pentium 4-M processor and 512 MB DDR RAM.

As *Figure 2* shows, many inconsistencies and inefficiencies have been found among the firewall rules. In detail, there are around 70 firewall rules among which there are 155 shadowings, 14 generalizations, 15 correlations and 434 redundancies. The verification time needed to discover all these inconsistencies and inefficiencies required less than 4 sec.

## 6. Conclusions and future work

So far, formal methods have been considered only for the verification of stateless firewall. In this paper, we proposed, for the first time, a formal verification method for stateful firewalls.

Our contributions are three-fold. First, we introduced a modelling technique for states, and defined the notion of inconsistency in case of the stateful environment. Second, we reduced the problem of verifying a stateful firewall to the problem of verifying a stateless firewall. More specifically, we proved that if the firewall configuration is free from inconsistencies and inefficiencies in a designated state, then it is free from these anomalies in all states. Third, we implemented our approach as a prototype stateful firewall verification tool, and used it for verifying real firewalls used in practice. Our experiments show that the tool is effective and efficient.

Regarding future work, there are many possible improvements that are yet to be done. Our approach could be extended to distributed firewalls, where multiple filters organized in some topology must function together without anomalies. It would also be interesting to extend this approach to the complex chain model of iptables.

Finally, the implementation can be extended to support other firewall products too, such as the Checkpoint FireWall-1 and Cisco ASA.

### Acknowledgments

The work presented in this paper has been partially supported by Ericsson through the HSN Laboratory at the Budapest University of Technology and Economics. Apart from this, Ericsson has no responsibility for the content of this paper.

The authors are thankful to Boldizsár Bencsáth for his help in understanding how iptables works and for his useful comments on firewall management in practice.

### Authors

**LEVENTE BUTTYÁN** received the M.Sc. degree in Computer Science from the Budapest University of Technology and Economics in 1995, and the Ph.D. degree from the Swiss Federal Institute of Technology in Lausanne in 2002. He joined the Department of Telecommunications at the Budapest University of Technology and Economics (BME) in January 2003, where he currently holds a position as an Associate Professor. His current research interests include security and privacy problems in wireless networked embedded systems, and the application of formal methods in security engineering. He has supervised several international projects at BME (e.g., UbiSecSens, SeVeCom, EU-MESH, WSAN4CIP), and he has been teaching courses on network security and electronic commerce in the MSc program. In addition, he is an Associate Editor of the IEEE Transactions on Mobile Computing, Area Editor of Elsevier Computer Communications, member of the editorial board of the Infocommunications Journal in Hungary, and he was a Guest Co-editor of the IEEE Journal on Selected Areas in Communications, Special Issue on Non-cooperative Behavior in Networking. He is a Steering Committee member of the ACM Conference on Wireless Network Security. In the last 6 years, he served on the Program Committee of around 30 international conferences and workshops, most of which were related to wireless network security.

**GÁBOR PÉK** received the B.Sc. degree in Computer Science in 2009 from the Budapest University of Technology and Economics, where he is currently doing his M.Sc. studies. He has been working on his project laboratories on several fields of security in collaboration with Levente Buttyán and his group for 1,5 years. In addition, he has experience in ethical hacking and forensics and was active on several industrial projects. His work on the analysis of stateful firewalls was awarded at the scientific student conferences (TDK) at the university and at the national levels. He also has internationally accepted publications in robot navigation algorithms. Currently, he is interested in malicious codes, botnets and mobile security.

**TA VINH THONG** received the M.Sc. degree in Computer Science from the Budapest University of Technology and Economics (BME). Since 2008, he has been working as a PhD student in the Laboratory of Cryptography and System Security (CrySyS), Department of Telecommunications, BME. His research interest is analyzing security systems using formal methods, especially, formal analysis of security protocols. His current research activities are formal verification of MANETs and formal languages.

## References


[1] E. Al-Shaer and H. Hamed, "Design and Implementation of Firewall Policy Advisor Tools." Technical Report CTI-techrep0801, School of Computer Science Telecommunications and Information Systems, DePaul University, August 2002.

[2] Ehab S. Al-Shaer and Hazem H. Hamed, Firewall policy advisor for anomaly discovery and rule editing. In: Integrated Network Management, pp.17–30, 2003.

[3] Ehab Al-Shaer and Hazem Hamed, Management and Translation of Filtering Security Polices, IEEE ICC'03, May 2003.

[4] Ehab Al-Shaer and Hazem Hamed, Modeling and Management of Firewall Policies, IEEE Transact. on Network and Service Management, Vol.1, April 2004.

[5] Ehab Al-Shaer, Hazem Hamed, Raouf Boutaba and Masum Hasan, Conflict Classification and Analysis of Distributed Firewall Policies, IEEE Journal on Selected Areas in Communications, Issue 10, Vol. 23, pp.2069–2084, October 2005.

[6] E. Lupu and M. Sloman, "Conflict Analysis for Management Policies." In Proceedings of IFIP/IEEE International Symposium on Integrated Network Management, May 1997.

[7] L. Yuan, J. Mai, Z. Su, H. Chen, C. Chuah and P. Mohapatra, FIREMAN: A toolkit for firewall modeling and analysis. In IEEE Symposium on Security and Privacy, pp.199–213, 2006.

[8] Florin Baboescu and George Varghese, Fast and scalable conflict detection for packet classifiers. Computer Networks 42(6), pp.717–735, 2003.

[9] Venanzio Capretta, Bernard Stepien, Amy Felty and Stan Matwin, Formal correctness of conflict detection for firewalls. In Proceedings of the ACM workshop on Formal Methods in Security Engineering (FMSE'07), pp.22–30, 2007.

[10] D. Eppstein and S. Muthukrishnan, "Internet Packet Filter Management and Rectangle Geometry." In Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), January 2001.

[11] P. Eronen and J. Zitting, "An Expert System for Analyzing Firewall Rules." In Proceedings of the 6th Nordic Workshop on Secure IT-Systems (NordSec 2001), November 2001.

[12] B. Hari, S. Suri and G. Parulkar, "Detecting and Resolving Packet Filter Conflicts." In Proceedings of IEEE INFOCOM'2000, March 2000.

[13] Mohamed G. Gouda and Alex X. Liu, A model of stateful firewalls and its properties. In Proceedings of the IEEE Int. Conf. on Dependable Systems and Networks, Yokohama, Japan, June 2005.

[14] Gábor Pék, Security Verification of Stateful Firewalls, Student Scientific Conference (TDK), Budapest, 2008.

[15] Oskar Andreasson, Iptables tutorial: http://iptables-tutorial.frozentux.net/iptables-tutorial.html

[16] A. Liu, E. Torng, and C. Meiners, Firewall Compressor: An algorithm for minimizing firewall policies. In Proceedings of the 27th Annual IEEE Conference on Computer Communications, 2008.

# Investigation of the impacts of user behaviour on pricing competition of Internet Service Providers: Empirical evidence and game-theoretical analysis

LÁSZLÓ GYARMATI, TUAN ANH TRINH

*Budapest University of Technology and Economics,*
*Department of Telecommunications and Media Informatics*
*{gyarmati,trinh}@tmit.bme.hu*

*Keywords: socio-economic issues, user behaviour, ISP, Internet access pricing, game-theoretic analysis*

**Socio-economic aspects of future communication networks such as pricing models for network providers, network neutrality, and Quality of Experience (QoE) become more and more important as the convergence of the networks is in progress. In this paper, we investigate the impacts of user behaviour – user loyalty in particular – on pricing strategies of Internet Service Providers (ISP) for a profitable yet sustainable Internet access marketplace. First, we propose a realistic user loyalty model, the price difference dependent loyalty model, which is based on empirical evidences. Next, we apply the loyalty model in game-theoretical analyses where optimal Internet access pricing strategies are expressed. Finally, we present the impacts of user loyalty on the prices and profits of ISPs in different scenarios based on simulation results. The simulation scripts for the investigation are available for further research [1].**

## 1. Introduction

In recent years, there has been an increasing interest in the socio-economic aspects of network systems. As an example, initiatives like the Euro-NF [2] and NSF FIND [3] promote economic incentives as a first-order concern in future network design. Economic models and pricing strategies for the edge of the networks enhance the profitability of local Internet Service Providers (ISPs) who sell Internet access for users.

Results in [4-6] initiate the discussion on customer loyalty and its impact on pricing strategies of ISPs, where a double reservation price based loyalty model is used. While [7] deals with ISPs' pricing strategies under uncertainties on static ISP markets, the authors of [8] used Stackelberg leader-follower game to handle dynamic local ISP market.

This paper presents and applies a realistic ISP customer loyalty model based on results of a survey carried out by our own. Under our price difference dependent loyalty model, the customers of the local ISPs are loyal based on the price difference of the prices of customers' current and possible future ISPs. Two price difference dependent loyalty models are used in this paper, a threshold-based model for analysing the effect of disloyal users on price setting strategies and a linear loyalty model for long-term strategies. The behaviour of service providers are investigated using game-theoretical tools.

There is broad literature in the area of modelling interactions between ISPs with game-theory, including [9-11]. They mostly assume a very simple user behaviour model: end-users choose the cheapest provider. However, this could be misleading if there are loyal customer segments present in the market, as loyalty is an important part of user behaviour. A vivid example of customer loyalty in practice is the loyalty contract between a service provider and a customer. The customers are charged with different price if they sign a contract and this difference depends on the length of the contract.

A number of empirical studies verify that user loyalty exists on ISP markets. In the USA 38% of the enterprise customers have been truly loyal to their ISPs [12]. National communication authorities of European Union's countries carry out regularly market research dealing with customer loyalty towards local ISPs. In the UK 27% of broadband users have already switched their provider at least once [13], while in Ireland 84% of subscribers have not changed their ISP in the last 12 months [14]. In Portugal, 81% of broadband customers said they did not intend to change ISP in the following 12 months [15]. In addition, only 16% of the Finnish subscribers have switched their ISP in 2007, mainly because of a better offer from a competitor [16].

The paper is structured as follows. First, in Section 2 we make a case for a loyalty model which is based on the price difference of service providers. Section 3 shows the implications of a disloyal customer segment on ISPs' pricing strategies based on a game-theoretical analysis. Section 4 presents simulation results where we quantify the effect of uncertain a priori knowledge on market shares, prices, and profits. Section 5 concludes the paper.

## 2. A case for price difference dependent loyalty in ISP markets

In order to create a realistic user loyalty model toward ISPs we carried out by our own a survey to gain insights about customer loyalty towards ISPs. Based on almost 800 answers we state that loyalty exists in the Hungarian ISP market as more than half of the persons have

not switched their ISP in the last five years. Numerous factors might have an impact on user loyalty, including Quality of Experience or customer service, but it turned out that the price difference of the ISPs also effects users' loyalty intentions toward service providers.

We asked what the minimal price is difference between the current and an other ISP when the answerers would switch their service providers. It was supposed that the two ISPs offer exactly the same service including connection speed, help center, etc. The answers are surprising as shown in *Figure 1*. Only around 5% of the answers would never leave their current ISPs, the remaining 95% said that there exists a price difference where they would become disloyal and would switch their providers. Based on the results we argue that modelling user loyalty based on the minimal price difference to switch is a realistic description of the ISP pricing problem.

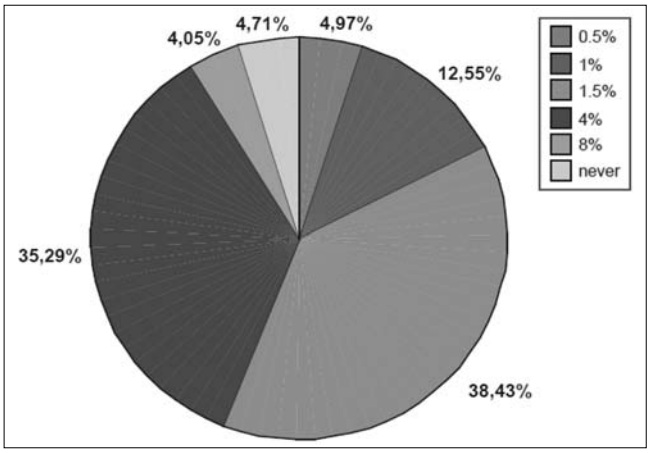



*Figure 1.*
*Minimal price difference to switch to an other ISP relative to the average salary*

Modelling loyalty based on price differences not only represents the relationship between the two prices but also includes information about the socio-economic aspects of the market. As an illustration consider two countries, a rich one and a poor one. In each country there exist two IPSs, they provide Internet access for 5 USD and for 10 USD. The price ratios are the same in both countries (0.5) but it is clear that there will be much more switchers in the poor country, where 5 USD (the price difference) worths a lot more than in the richer country.

The relation between the number of years to be a customer of the current ISP and the minimal price difference to switch is presented in *Figure 2*. The minimal price differences are not specific to the loyalty history of the subscribers, regardless of the years to be a subscriber of a specific ISP there are similar price differences where the customers would switch their ISPs.

## 3. Pricing Internet access for disloyal users

In this section we present our threshold-based loyalty model and apply it to analyse pricing strategies of service providers dealing with disloyal users. We investigate a price setting game where only two service providers exist. Customers are split into two partitions upon their loyalty: $l_1$ customers are loyal to $ISP_1$, while $ISP_2$ has $l_2$ loyal users. $d_i$ denotes the price difference meaning that if the price of $ISP_i$ is more than the other ISP's price plus $d_i$ then the user will be a switcher, she leaves her ISP for the other one. The demand function is modelled as a constant function until a border price ($\alpha$), if at least one of the ISPs sets a price less than $\alpha$, the demand is $l_1+l_2$ but above $\alpha$ none of the users buys Internet access. The service providers set their prices simultaneously after that the users select their access providers. The payoff functions of the ISPs can be expressed as

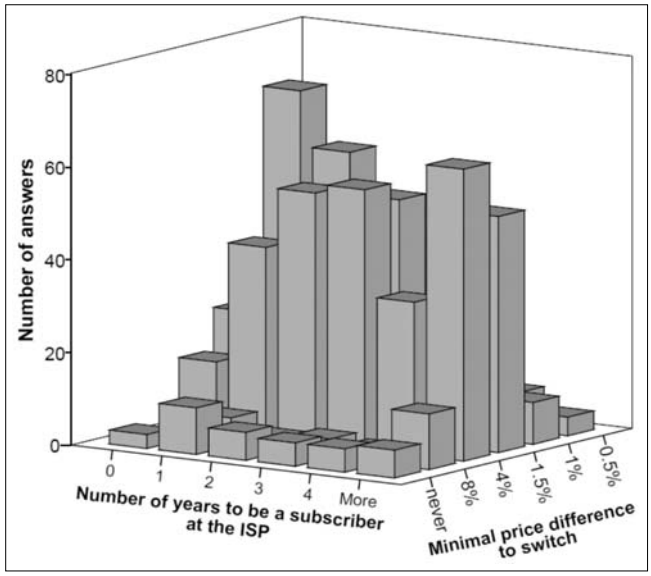
$$\Pi_1 = \begin{cases} (l_1 + l_2)p_1 & p_1 < p_2, |p_1 - p_2| > d_2 \\ l_1 p_1 & |p_1 - p_2| \le d_1 \\ 0 & p_1 > p_2, |p_1 - p_2| > d_1 \end{cases}$$

$$\Pi_2 = \begin{cases} (l_1 + l_2)p_2 & p_2 < p_1, |p_2 - p_1| > d_1 \\ l_2 p_2 & |p_2 - p_1| \le d_2 \\ 0 & p_2 > p_1, |p_2 - p_1| > d_2 \end{cases}$$

*Figure 3* illustrates the payoff function in two different scenarios. We present the payoff of $ISP_1$ at different prices ($p_1$) while the price of $ISP_2$ is fixed. *Figure 3/a* presents the payoff function if the border price is not smaller than the price of $ISP_2$ plus the price difference, while *Figure 3/b* shows the payoff if the border price is lower.

We have shown in [6] that the ISP price setting game with price difference dependent loyalty has only one pure strategy equilibrium if the ISPs have same price difference ($d$) but this assumption is not always true in the real world. One ISP can have more hard-core loyal subscribers than the others. To model this we introduce separate minimal values ($d_i$) for every ISP. Using this we can model disloyal users if we add a new virtual service

*Figure 2.*
*Relation of user loyalty and minimal price difference*



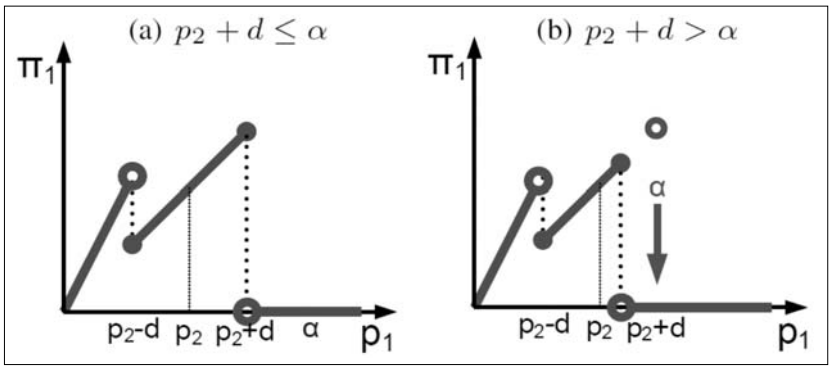



*Figure 3. Illustration of the payoff function*

provider to the market with users who have zero minimal price difference. The formal definition of game is:

- *Players:*
  the Internet Service Providers, i=1,...,n,
  $ISP_i$ has $l_i$ loyal customers with $d_i$ price difference.
- *Strategies:*
  the price of the Internet access, the decision of $ISP_i$ is $p_i$, $p_i \in [0,\alpha]$, players can have
  only pure strategies, they play single-shot game.
- *Payoff functions:*
  the payoff of $ISP_i$ is based on the above described payoff functions.

**Proposition**

If the ISPs' subscribers have different price differences there exists a pure strategy Nash equilibrium of the two-player game if the following conditions hold:

$$\frac{l_2}{l_1+l_2} \leq \frac{d_2}{\alpha}$$
$$\frac{l_1}{l_1+l_2} \leq \frac{d_1}{\alpha}$$

**Proof**

We calculate the minimal prices where an ISP will compete in the $[0,\alpha-d_i]$ and $[\alpha-d_i,\alpha]$ intervals. We deal with $ISP_1$, we can have similar conditions for $ISP_2$:

$$p_2 \in [0,\alpha-d_1]: \quad p_2 > d_2 + \frac{l_1(d_1+d_2)}{l_2}$$
$$p_2 \in [\alpha-d_1,\alpha]: \quad p_2 > d_2 + \frac{l_1\alpha}{l_1+l_2}$$

This game has a best response figure where an intersection of the graphs does not exist except at $(\alpha,\alpha)$. $ISP_1$ will compete if her payoff can be larger if she grabs the users of $ISP_2$, namely

$$(l_1+l_2)(\alpha-d_2) > l_1\alpha$$
$$l_2\alpha > d(l_1+l_2)$$
$$\frac{l_2}{l_1+l_2} > \frac{d_2}{\alpha}$$

The conclusion is the same for $ISP_2$. This means that if the ISPs' subscribers have different price sensitivities there exists a pure strategy Nash equilibrium at $(\alpha,\alpha)$ if the following conditions are satisfied:

$$\frac{l_2}{l_1+l_2} \leq \frac{d_2}{\alpha}, \quad \frac{l_1}{l_1+l_2} \leq \frac{d_1}{\alpha}$$

The proposition can be generalised for N players, where the conditions of the Nash equilibrium are similar to the conditions of two player game.

With the above introduced model we are able to handle disloyal users by creating a virtual ISP which has the disloyal users. The price sensitivity of the subscribers of the virtual ISP is small (around zero). If we look at the constraints we can see that if we have disloyal users the game will no longer have pure strategy Nash equilibrium: the value of $d_v/\alpha$ will be zero but the left side of the inequalities will be positive. As a closing word we state that if there are disloyal users in the market the Internet Service Providers can not play their pure equilibrium strategies, they have to set their prices based on probabilities and compete for the disloyal users.

## 4. Impact of long-term interaction on dynamic ISP markets

ISPs usually do not have complete knowledge about their competitors, they only have beliefs. They set their access prices based on their a priori information which will be adjusted based on the observed behaviour of the competitors. ISPs have to price Internet access not only in static markets, pricing strategies are even more crucial in dynamic markets if strategic decisions have to be made. We call a decision strategic if it has a significant long-term effect on the company. In this section, we analyse dynamic ISP markets, where a new ISP enters a local market using a linear price difference dependent loyalty model. We model game-theoretically the entry situation using a Stackelberg leader-follower game, where the incumbent ISPs are the leaders and the entrant ISP is the follower of the game.

The customers buy Internet access if the access price is at most α. The local ISP market consists of i=1,...,n companies, $ISP_i$ has $l_i$ loyal customers, thus the total number of the customers is $\sum_i l_i$. If a subscriber changes her ISP then she selects the cheapest available price. The new, entering ISP has not got any subscribers at the beginning. We assume that the subscribers' loyalty is based on the ISPs' price difference, we use a linear customer loyalty function, meaning $ISP_i$ loses

$$L_i = \frac{p_i - p_j}{\alpha} l_i$$

customers if $ISP_j$ has the lowest price ($p_j < p_i$). Every service provider plays rationally, namely selects its profit maximizing strategy. The discount factor is denoted by $0 \leq \Theta \leq 1$, the profit of ISPs is discounted at each step with Θ. We assume that each ISP has the same discount factor. $k_i$ denotes the number of rounds for $ISP_i$ looks forward. The ISPs do not know in advance the payoff of future rounds, they only have a priori knowledge.

$EP_i^{(k)}$ denotes $ISP_i$ belief about the expected access price in round $k$, similarly $EI_i^{(k)}$ is the $ISP_i$'s expected subscriber number in that round. The payoff function of the service providers is

$$\Pi_i = l_i^* p_i + \sum_{j=1}^{k_i} \Theta^j \mathrm{EI}_i^{(k)} EP_i^{(k)}$$

We assume that the ISPs have complete information when they select their strategies. However, real ISPs usually do not know the exact values of the parameters, e.g. the scope of other ISP's payoff, the price of the Internet access in the next rounds, the number of own subscribers, and the discount factor of the ISPs can also be variable. These uncertainties have an impact on the price setting decisions what we quantify later on.

The formal definition of analysed long-term price setting game is as follows.

- *Players:*
  the Internet Service Providers, i=1,...,n, $ISP_i$ is an incumbent with $l_i$ loyal customers while $ISP_{n+1}$ is the entrant of the game.
- *Strategies:*
  $p_i \in [0,\alpha]$ the price of the Internet access at $ISP_i$, $ISP_i$ looks forward $k_i$ rounds in order to maximize its payoff, only pure strategies are allowed.
- *Payoff functions:*
  the payoffs of the ISPs are

$$\Pi_i = \left(1 - \frac{p_i - p_{n+1}}{\alpha}\right) l_i p_i + \sum_{j=1}^{k_i} \Theta^j \left(1 - \frac{p_i - p_{n+1}}{\alpha}\right) l_i P, i = 1,\ldots,n$$

$$\Pi_{n+1} = \sum_{i=1}^{n} \frac{p_i - p_{n+1}}{\alpha} l_i p_{n+1} + \sum_{j=1}^{k_{n+1}} \Theta^j \frac{p_i - p_{n+1}}{\alpha} l_i P$$

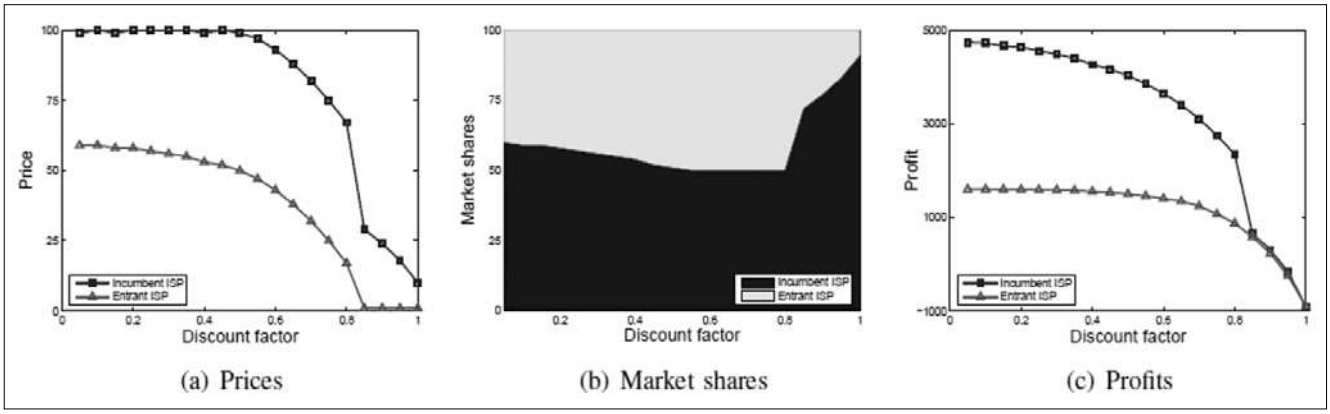
The first term of the payoff function denotes the profit of the current round, while the second describes the weighted profit of future rounds. The profit is proportional to the number of subscribers and the profit of an access. The entrant $ISP_{n+1}$ has to set the lowest price in order to have customers, otherwise $ISP_{n+1}$ can not enter the market. The algebraic calculation, based on the best response functions of the providers, yields implicit equations for the equilibrium prices. The equilibrium prices can be expressed explicitly solving a system of linear equations formulated from the implicit equations.

However, the solutions of the system are equilibrium prices only if all the prices are between 0 and $\alpha$. In other cases, the optimal prices can be computed creating an order of the ISPs, e.g. based on their number of customers, the last ISP in the hierarchy is the entrant company. Using the ordering, the equilibrium prices can be expressed step-by-step using the backward induction paradigm of game theory. In particular, first the entrant ISP selects its equilibrium price in every possible scenarios which can exist based on the decisions of the other ISPs, after that the next ISP selects its price, etc. In the followings, we present simulation results where the equilibrium prices are computed using this backward induction method, moreover we also quantify the profits and market shares of the ISPs.

The impact of incumbent's discount factor is shown when a new ISP enters a monopolistic local ISP market in *Figure 4*. The market share of the incumbent ISP is 100% ($l_1$=100), she looks forward $k_1$=5 rounds. The access prices can be between 0 and 100 while the expected future price is 40. The entrant ISP has a discount factor of 0.8 and she looks forward 15 rounds while setting the price. At smaller discount values the possible future incomes are negligible to the income of the current round, therefore the incumbent ISP sets a price as high as possible *(a)*, to maximize its profit. Accordingly, the entrant ISP can grab a significant part of the market with a small enough price, in some cases almost the half of the users select the entrant ISP *(b)*. As the expected profit of the future becomes more significant, meaning the discount factor is more than 0.8, the incumbent ISP tries to hold its subscribers by lowering its price. Because of the lower prices the profit of the ISPs are decreasing *(c)*, but the incumbent ISP has always larger profit than the entrant ISP.

We illustrate the impact of the scope of the pricing decisions on market shares. In *Figure 5,* we present three scenarios where the scope of incumbent $ISP_2$ is 5, 10, and 20 rounds respectively, while the other ISPs look forward for 15 rounds. The incumbent ISPs weight the future incomes with a discount factor of 0.9 while the entrant has only 0.2 as a discount factor. If the number of rounds is small *(a)* $ISP_2$ lost almost all of her custo-

*Figure 4.*
*The impact of discount factor on market shares, prices and profits, when an ISP enters a monopolistic market*



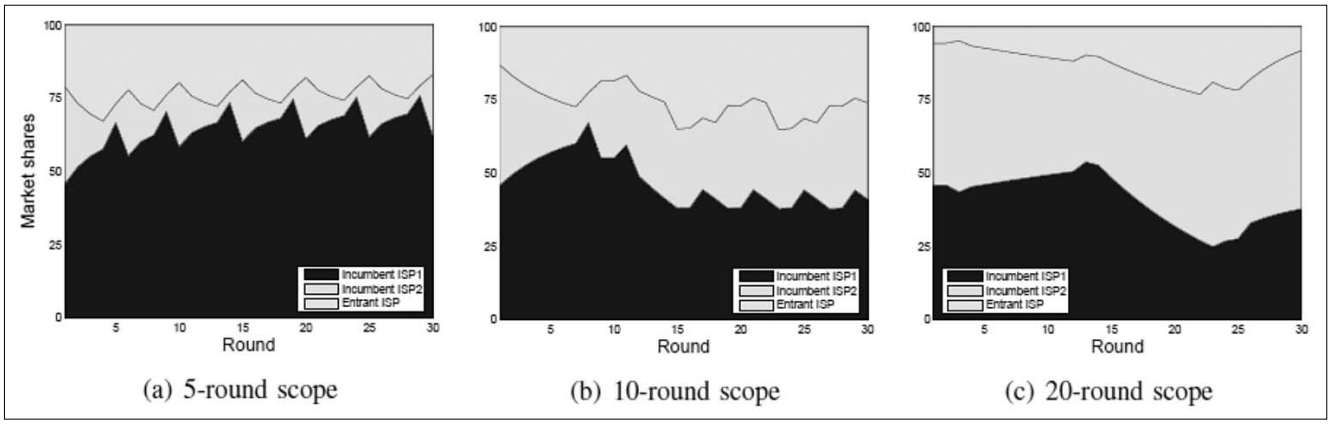



*Figure 5.*
*Market shares when ISP2 looks forward different number of rounds, the incumbents have same initial market shares*

mers, while if she sets prices based on 10 rounds her market share is the same as the entrant's *(b)*. Furthermore, if $ISP_2$ prices Internet access based on long-term decisions, where she looks forward 20 rounds, the entrant ISP's market share becomes marginal *(c)*.

## 5. Conclusions

In this paper, we have demonstrated how Internet Service Providers can include loyal customer segments in their price setting strategies in order to maximize their incomes. We have provided game-theoretical analysis for handling disloyal customers on static ISP markets, where the number of providers is constant. It turned out that if disloyal subscribers exist in the market, the ISPs have to price Internet access using mixed strategies, creating competition between the ISPs. In addition, we have applied linear price different dependent loyalty model to price access on dynamic ISP markets.

Based on the simulation results, we state that long-term Internet access pricing strategies have to be selected carefully in order to maximise the profits of ISPs.

### References

[1] Economics of Networked Systems Group, BME, http://netecon_group.tmit.bme.hu/
[2] Euro-NF: Network of Excellence on the Network of the Future, http://euronf.enst.fr
[3] NSF: Future Internet Network Design Initiative, http://find.isi.edu
[4] Chiou, J.S., The Antecedents of Consumers' Loyalty Toward Internet Service Providers, Inf. Management, Vol. 41, No. 6, pp.685–695., 2004.
[5] Trinh, T.A., Pricing Internet Access for Disloyal Users: a Game-Theoretic Analysis (extended abstract), Workshop on Socio-Economic Aspects of Next Generation Internet, Sweden, 2008.
[6] Biczók, G., Kardos, S., Trinh, T.A., Pricing Internet Access for Disloyal Users: a Game-Theoretic Analysis, SIGCOMM 2008 Workshop on Economics of Networked Systems, 2008.
[7] Gyarmati L., Trinh, T.A., How to Price Internet Access for Disloyal Users under Uncertainty, Annals of Telecommunications (submitted), 2009.
[8] Gyarmati L., Trinh, T.A., On Competition for Market Share in a Dynamic ISP Market with Customer Loyalty: A Game-Theoretic Analysis, 6th International Workshop on Internet Charging and QoS Technologies, Aachen, 2009.
[9] He, L., Walrand., J., Pricing and Revenue Sharing Strategies for Internet Service Providers, IEEE Infocom, 2005.
[10] X.-R. Cao, H.X. Shen, Wirth, P., Internet Pricing with a Game Theoretical Approach: Concepts and Examples, IEEE/ACM Transactions on Networking, pp.208–216., 2002.
[11] Shakkottai, S., Srikant, R., Economics of Network Pricing with Multiple ISPs, IEEE Infocom, 2005.
[12] Walker: The 2005 Walker Loyalty Report for Information Technology, 2005.
[13] Ofcom – Office of Communications: The Communications Market, 2008.
[14] Comreg – Commission for Communications Regulation: Consumer ICT Survey, 2008.
[15] ANACOM: Survey on the use of broadband, 2006.
[16] FICORA – Finnish Communications Regulatory Authority: Market Review, 2007.

# CSP-based modelling for self-adaptive applications

SZILÁRD JASKÓ, GYULA SIMON, KATALIN TARNAY, TIBOR DULAI, DÁNIEL MUHI

*University of Pannonia, Department of Electrical Engineering and Information Systems*
*jasko.szilard@uni-pen.hu*

*Keywords: self-adaptive, CSP based model, communication system, sensor network*

**In this paper, a CSP-based modeling approach is presented for self-adaptive systems, the proposed solution supports self-configuration, self-learning and testing as well. The behavioral elements of the system are described using the CSP language. A simple service is also defined which supports self-adaptation of subsystem components by learning from other system components. The efficiency of the system is demonstrated in two practical applications: an adaptive communication discovery protocol and a sensor networking application are implemented using the proposed approach.**

## 1. Introduction

The complexity of information-based systems is growing continuously, therefore new approaches are needed to tackle the arising problems. Self-adaptive thinking is one of the promising ways to mention, where the system is able to evaluate its own behavior, make decisions to alter this behavior, and perform the necessary reconfigurations in order to improve its performance. Such changes may be required either because of the changing environment and requirements, or because of the lack of sufficient knowledge to solve the problem.

Self-adaptive systems are usually model-based systems. Several model representations can be used to describe the behavior of systems, depending on the type of system and application. For description of systems with heavy communication between its subsystems, an elegant and efficient way is the usage of the Communicating Sequential Processes (CSP), which a process algebra-based mathematical formalism.

Efficient self-adaptive systems can be created if there are efficient tools for the support of learning, testing, adapting, and configuring methods. For reliable systems the behavior and the working flow of the system have to be exact and provable. CSP provides efficient tool sets for specification, verification and testing.

With CSP, not only the communication protocols can be described efficiently, but the behavior of the system as a whole as well. It is particularly true for event-based systems with heavy intercommunication need.

In this paper CSP-based modeling is proposed, which supports self-adaptive, self-configuring, self-learning and even self-testing behavior of complex systems. The proposed system uses a simple and robust learning mechanism: system components can discover new behavioral elements (e.g. communication protocols, algorithms etc.) in their neighbor's knowledge base, and, if necessary, these rules can be learned. The proposed approach is illustrated by two practical applications: a self-adaptive communication protocol and a sensor networking data acquisition system.

The outline of the paper is the following: Related research is briefly summarized in Section 2, and then CSP is reviewed in Section 3. Section 4 introduces the proposed CSP-based self-adaptive system, and its ability to support self-adaptive behavior is described. In Section 5 two practical applications illustrate the potential of the approach, and Section 6 concludes the paper.

## 2. Background

Managing complexity is the main driving force behind the application of self-adaptive systems. Self-adaptive systems with learning and teaching abilities can handle problems in a novel way: the designer does not need to build in all the required information for every possible case, but rather the system can handle the exceptions in run-time. Such systems are flexible and can adapt themselves to new challenges. The importance of the area is shown by the wide range of research and the high number of publications. Self-adaptive systems were created in many application areas, e.g. distributed services [1], mobile and next generation networks [2,3], self-organizing solutions [4], or even organic robot control architectures [5] and bio-inspired approaches [6]. Self-adaptive protocols also bring new possibilities in protocol design and network organization [7,8].

In the field of sensor networks, in addition to the wide range of routing applications utilizing self-adaptive features (e.g. [9]), several interesting self-adaptive and self-organizing solutions were proposed: a self-organizing system was used to create a low-cost localization system [10]; adaptive self-diagnosis services were proposed to monitor network status and degradation [11]; or in monitoring applications self-configuring nodes prolong network lifetime while providing the required sensing service as well [12].

A good summary on the application of CSP can be found in [13]. An important usage of CSP is the specification and verification of fault-tolerant systems; an elegant verification technique for this problem was proposed in [14]. CSP was used in large industrial projects as well, e.g. the International Space Station project [15], and the testing of the avionic systems of an Airbus aircraft [16].

## 3. The CSP (Communicating Sequential Processes)

CSP [17-19] is a notation for describing concurrent systems and the interaction patterns between the component processes. It has a wide range of applications from programming languages [20] to verification of safety protocols [21]. A CSP system is built up from independently running sequential processes, which communicate with each other using message passing. The mathematical background is process algebra.

Each process has its own alphabet, which is the set of all the communication events the process might use. CSP defines several operators, by the help of those operators complex process descriptions can be easily constructed. We can express sequential communicational events ($\rightarrow$), decision ($P \lessdot p \gtrdot Q$ means: if $b$ then $P$ else $Q$), recursion, different choices (deterministic: $\Box$, or non-deterministic: $\Pi$), and simultaneous behavior of processes. The language has its own built-in basic processes and we can also use inner variables.

Let us see CSP expressions for example:

$$P1 = up \rightarrow down \rightarrow P1$$

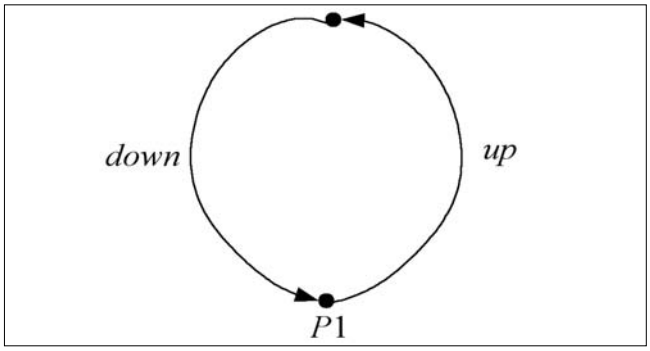



*Figure 1. P1 process*

We can see that *P1* is a process and it is shown in *Figure 1* "*up* and *down*" are events from the alphabet. The next example is:

$$COPY = left?x \rightarrow right!x \rightarrow COPY$$

It is a copy-machine and it defines two new things: the channels and an input and output operator. There are *"left"* and *"right"* channels in this example. *"?"* means input and *"!"* means output. So this machine copies *"x"* that comes in the left channel and puts it out in the right channel. Next rule is the external choice:

$$P \Box Q$$

It is a process which offers the environment of the choice of the first events *P* and of *Q* and behaves accordingly. The combination of the previous examples is:

*ExampleState =*
$c?rq \rightarrow c!rp \rightarrow ExampleState \Box c?rpEnd \rightarrow NextState$

There is an environmental (external) choice in process *"ExampleState"*, denoted by "□". One of the possible ways: to get an *"rq"* event on channel *"c"* from an other process, consequently, the *"ExampleState"* process sends *"rp"* through channel *"c"* and after that stays in the same state. The second choice is to get an *"rqEnd"* signal on channel *"c"*, which shifts the process to state *"NextState"*. CSP gives us a tool for determining the trace of processes.

Trace of a process is the set of all the possible sequences of events that can happen during the process' life. An element of a trace is written between signs "<>". Trace of a process always contains the empty trace (<>). If we follow the example mentioned above, the traces of the process can be determined:

*traces(ExampleState)=*
*(<>, <c?rqEnd>, <c?rq>, <c?rq, c!rp>,*
*<c?rq, c!rp, c?rqEnd>, <c?rq, c!rp, c?rq>,*
*<c?rq, c!rp, c?rq, c!rp, c?rqEnd>,*
*<c?rq, c!rp, c?rq, c!rp, c?rq>... )*

There are various tools for checking CSP implementations. Animators make it possible to write arbitrary process descriptions and to interact with them [22], while refinement checkers explore all of the states of a process [23]. CSP is very useful to debug failures, discover deadlock or livelock, and is an ideal language for helping verification, validation and test processes [24,34,35].

## 4. CSP-based self-adaptive system

Imagine a system where every component can automatically recognize the other components' communication and working rules. They can learn from each other new methods of operation, and the whole system can adapt to changes in the environment. Of course, it is true that if at least one element of the system knows the right rule(s) that gives a solution for the occurring problem. From this point, the adaptation process is automatic. In this case the system contains self-adaptive, self-configuring, and self-learning elements indeed.

The system builds up from nodes that can communicate with each other as can be seen in *Figure 2*. Every node has a database that stores fields of CSP rules and conditions. The CSP rule gives the communication/working flow of the node. If the condition is true the corresponding CSP rule will be activated, and all other rules will be inactive at the same time. An example will be presented in Section 5.1.

There is a special node in the network that includes the referenced data. That is useful if some nodes try to

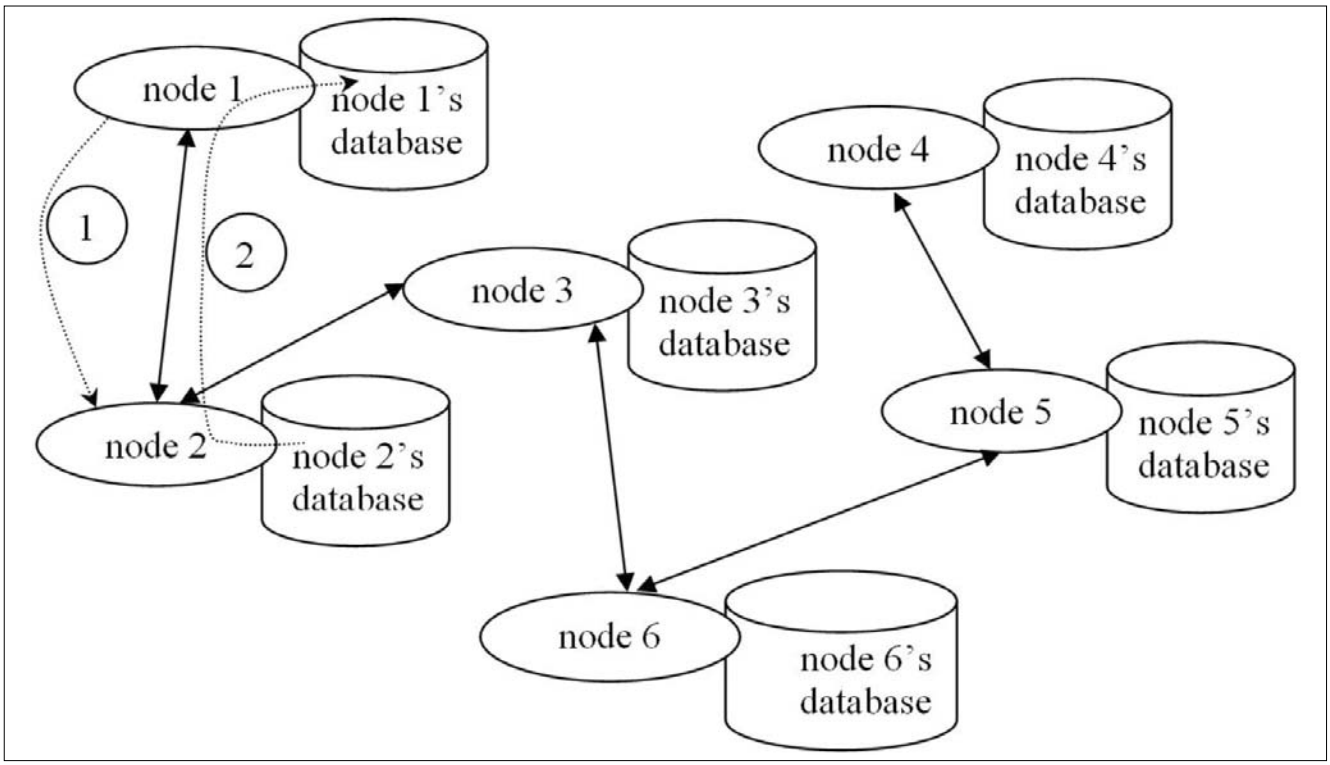



*Figure 2. Structure of the system*
*Nodes can communicate to perform normal operation (solid lines), and also can exchange information to support self-adaptivity (dashed line). In the example Node 1 requests behavioral information from Node 2, and stores the rules in its own data base.*

propagate an erroneous working mechanism. In this case, the special node orders the "bad" node(s) to delete the erroneous rule(s) and learn the good one. More details are found about it in Section 4.3.

#### 4.1 Teaching/Learning

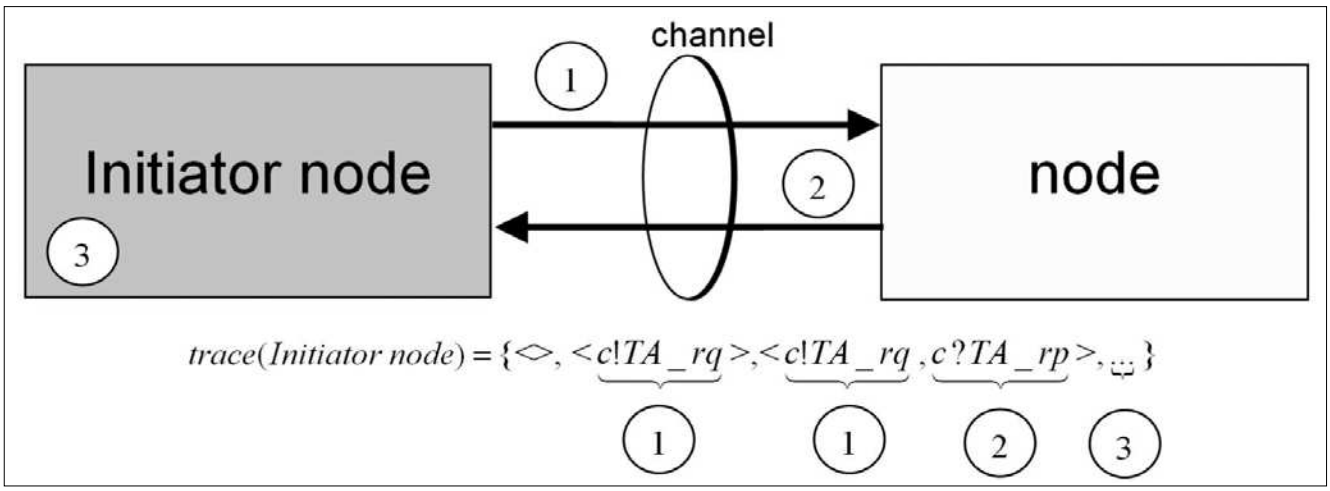
The rules are short and optimized CSP code, describing the communication and the processes in the system, and are stored in the local rule base of the nodes. Every node can check the database of other nodes. If it finds an unknown rule, it can learn it. This flow is illustrated in *Figure 2,* where Node 1 requests the rules of Node 2. Node 2 sends the requested data from its rule base. Node 1 processes the received data and if it finds new conditions and/or rules in the record, it will learn it. In Section 5.1 and 5.2 this process will be illustrated in practice.

#### 4.2 Testing

An ideal test tool could recognize every communication device automatically and would be able to determine the protocol of the other party. This is a complicated task; its complexity can be compared to learning a new language from a grammar book and a dictionary.

Computers and mobile devices communicate with each other with the help of artificial languages. These

*Figure 3. Background protocol" between initiator node and a node.*
*1. Ask the traces of the tested machine; 2. Send back the asked information; 3. Generate test cases and start the test*



protocols have several variations and versions. This fact can cause problems in many cases, because two different versions of the same protocol are sometimes not compatible.

Here, the inter-operability and the cooperation between devices are difficult. An efficient solution would be to be able to recognize the protocols used by different devices, and be able to learn and use them in the subsequent communication. The protocol and the version are recognized by extra information stored in the form of CSP rules. The CSP rules are requested by the initiative node with the help of the background protocol, shown in *Figure 3,* and described in detail in Section 4.3.

A simple and fast verification technique is when the tester node uses simple pattern matching on the received and its own CSP rules. If the rule does not match the authentic rule, the node will drop its own rule and will start the learning period by reconfiguring itself to the right CSP rule. A java-based application working with this principle will be shown in Section 5.1.

An additional, but more complicated possibility is the functional test of the used communication protocol. Based on the dictionary and the grammar book it can be decided whether the speaker is really speaking the right language: the tester has to ask questions and from the replies it can determine whether the speaker correctly speaks the language. Similarly, communication protocols can be verified by CSP rules, which include all the information required for test generation: the communication rules, interfaces and signals [25]. From the CSP model a behavior tree can be built and test cases can be generated, then the tester node can start to analyze the other node with the help of generated data [26-28].

#### 4.3 The service process

In this section the service process will be described, which provides the means of learning and testing for the system components. In the system there always is a special node that has the reference data, being authoritative in conflict situations. Otherwise, every node is equal. A node can request data from another node using the following protocol, described in CSP:

$$Node = NormalOperation$$
$$NormalOperation = c!RD\_rq \rightarrow NormalOperation \Box c?RD\_rp \rightarrow ProcessData \Box c!TA\_rq \rightarrow NormalOperation \Box c?TA\_rp \rightarrow Tester$$
$$ProcessData = LearnNewComponent \trianglelefteq RD\_rp\_has\_new \trianglerighteq NormalOperation$$
$$LearnNewComponent = NormalOperation \trianglelefteq accomplished \trianglerighteq LearnNewComponent$$
$$Tester = c!rq \rightarrow Tester \Box c?rp \rightarrow Tester \Box c!rqend \rightarrow WaitTester$$
$$WaitTester = c?rpEnd \rightarrow NormalOperation \Box c!rqEnd \rightarrow WaitTester$$

where

$c$ = channel, ? = input, ! = output, TA = Test Alphabet, RD = Request Data, $rq$ = request, $rp$ = response

$rq \in TestAlphabets \cup OtherRequest$, $rp \in Responses$

This automaton describes one half of the basic background communication of a node. In state *NormalOperation,* with the help of *RD_rq* (request for Requested Data) signal, a request is sent out to another node asking for its database. The reply arrives in message *RD_rp,* which is processed and the receiver learns the received rules. With the help of the *TA_rq* (request for Test Alphabet) signal, which can also be sent out in *NormalOperation* state, the node asks the other node to send information needed for testing.

This information comes in *TA_rp* (this is the active CSP rule); if the process is supported by the machine under test, a transition is generated to state *Tester,* where the generated test suite is running – in the automaton it is represented by *rq-rp* (request–response) message pairs. If testing is over, the initiator node sends an *rqEnd* signal. As a response, the other node sends back an *rpEnd* signal and it takes back the automaton into the initial state.

The other part of communication runs on the responder node. For sake of clearness, this part of the automaton is shown separately, as follows:

$$Node = NormalOperation$$
$$NormalOperation = c?x \rightarrow RequestData \trianglelefteq x = RD\_rq \trianglerighteq (TestAlphabet \trianglelefteq x = TA\_rq \trianglerighteq NormalOperation)$$
$$RequestData = c!RD\_rp \rightarrow NormalOperation$$
$$TestAlphabet = c!y \rightarrow TestState \trianglelefteq y = TA\_rp \trianglerighteq NormalOperation$$
$$TestState = c?rq \rightarrow c!rp \rightarrow TestState \Box c?rqEnd \rightarrow TestEnd$$
$$TestEnd = c!rpEnd \rightarrow NormalOperation \Box c?rqEnd \rightarrow TestEnd$$

where

$c$ = channel, ? = input, ! = output, TA = Test Alphabet, RD = Request Data, $rq$ = request, $rp$ = response

$rq \in TestAlphabets \cup OtherRequest$, $rp \in Responses$

The responder node starts from the *NormalOperation* initial state. If the signal *RD_rq* (request for Requested Data) arrives via channel *c* as signal *x*, the automaton steps to state *RequestData*, otherwise the state does not change. In state *RequestData* the requested data is send back in message (*RD_rp*). If the signal *TA_rq* (request for Test Alphabet) arrives via channel *c* as signal *x*, the system turns into *TestAlphabet* state, otherwise, the state does not change. In *TestAlphabet* state if the *TA_rp* is sent back via channel *c* as signal *y*, the automaton turns into state *TestState*. Otherwise, it steps back to state *NormalOperation*.

Testing happens in *Test State* communicating with pairs of request and response messages; if it is over, the tested machine gets an *rqEnd* message. It inducts the shift to state *TestEnd*, where it is also possible to get other *rqEnd* messages. After sending an *rpEnd* message the system gets back to its normal operation at the initial state.

## 5. Applications

#### 5.1 Self-configuring communication system

Self-adaptive communication protocols will appear in the future in many application areas and they will be able to adapt to changes of the environment. In the following example a system is defined in which every node

| Communication rule (CSP trace) | Condition | Status |
|---|---|---|
| <openChannel?Data>;<openChannel!Data> | packetLost=0 | active |
| <openChannel?Data,openChannel!Ack>;<br><openChannel!Data,timerChannel!100,openChannel?Ack>;<br><openChannel!Data,timerChannel!100,openChannel?Timer,<br>openChannel!Data,timerChannel!100,openChannel?Ack>;<br><openChannel!Data,timerChannel!100,openChannel?Timer,<br>openChannel!Data,timerChannel!100,openChannel?Timer> | packetLost>0 | passive |

*Table 1. The database of a communicating node*

can test its communication partner and thus use the appropriate protocol. Simple examples include adaptivity to channel quality (in a noisy channel more robust protocol must be used); or channel safety (in a safe channel encryption is pointless; otherwise cryptographic defense of data is inevitable.)

In the demo application every communicating node has a database. This structure was defined in Section 4 and this scenario follows it. One of them is the part of describing communication rules. Here we use CSP traces, because it is equivalent to the standard CSP language, it is compact and easy to interpret (naturally, redundancies are removed from the traces for sake of compactness.) The trace(s) is/are chosen from the set of traces about the following view-points:

– the trace will be optimal as it is available,

– cover the given working/running mechanisms.

The traces give us a further advantage. If the used trace is finite we have good chance that the program of the node is livelock free. Of course, there are many CSP traces that are infinite. In this case, it has to be modified manually to work right. So the trace works like an indicator in the system and shows us if the program code includes some fatal errors. An example is presented in *Table 1*. This example includes an extra element and a status part. It is just an indicator that signals to the user which rule is active.

The simple example database contains two rules. The first (currently active) rule defines the behavior of the system when the communication channel is of good quality *(packetLost=0)*; in this case received messages are not acknowledged. The second rule explains the expected behavior if the message loss rate is unacceptable *(packetLost>0)*; in this case received messages must be acknowledged by the receiver, otherwise the sender node retries the transmission two times, after a backoff time of 100. The channel quality can be measured by any appropriate way (not included in the description); in the example condition the measured *packet Lost* variable is checked.

The main advantage of this self-adaptive communication scheme is its ability to adapt to any extreme environment without re-planning the whole system. Only a new record, containing the communication rules of the nodes has to be added to the database and the entire system will work according to the changed parameters. Naturally, not only a client/server connection can be controlled this way but the whole networked system can update its communication rules.

Note that a communicating node can never know precisely which communication rule (known or unknown) is used by its partner at the moment. The test functionality described in Section 4.2 can help in this case, because the client is able to check the other party's communication protocol, or it can learn it if that method is unknown yet, as shown in Figure 2.

In this simple example the server (the node with the authentic data in case of conflict) has rules, shown in Table 1. The client starts its operation with its database empty thus initializes the synchronization process of the communication rules. The server sends the communication rule base to the client. The client learns the new records and finishes the synchronization process.

The log of the operation can be seen in *Table 2*. In this example only a server-client communication was used, but it can be naturally extended to a larger network. The demo program, developed in Java, can be downloaded from [29].

### 5.2. Self-adaptive sensor networks

Sensor networks are special computer networks, containing potentially hundreds or thousands of embedded

*Table 2. Server and client node logs*

| server node | client node |
|---|---|
| | *Start the synchronization process of the communication rules* |
| *Send the communication rule(s) to client...* | *Ask the server communication rule(s)...done.* |
| | *Learn...done.* |
| | *The synchronization process is finished.* |

sensor nodes (called motes). Every mote is a small, usually battery-powered device, built around a low-power microcontroller running at a few MIPS with a few kilobytes of RAM, and is equipped with a wireless transmitter and the application-specific sensing capabilities [9]. Self-adaptivity is a widely researched area of sensor networks. Since inter-mote communication is usually done by ad-hoc networking, network elements must discover and adapt to unknown and potentially changing network topology.

Other interesting solutions include adaptive resource allocation [30], clustering [31], or automatic topology control [12], to achieve longer network lifetime. However, self-adaptivity is not provided at the application level. Currently the only way to change the application is the (in-network) reprogramming of the motes, where the whole memory footprint (usually tens of kilobytes) must be downloaded, inducing a serious overload on the network. CSP-based modeling provides an elegant way to include high-level self-adaptive properties in the network. The following proof of concept application was developed for Mica motes [32] running TinyOS operating system.

In the network every mote has a small built-in CSP interpreter that can translate modified CSP traces. Instead of the full CSP description, the equivalent traces were used again to (1) simplify the complexity of the interpreter and (2) decrease the message sizes. In addition to the standard CSP code, few extensions were added for the sake of efficiency: the syntax of the decision operation was changed (syntax: i(condition)(true branch)(false branch)), and the for loop, as an inseparable event was added (syntax: f(control expressions){body}).

The application layer of the motes becomes self-configuring with the help of this solution. Every mote can detect the actual program to be run, and motes can learn new application pieces, if necessary. Thus the operation can be adapted to the actual requirements. The demo application contains a simple sensor (light sensor). In mode 1 motes measure the ambient light and each mote in the network learns the position of the brightest spot. Mode 2 is more complicated: here each node builds a list of the brightest N nodes.

The CSP rules describing the operation in mode 1 are shown in *Table 3*. The measurement process is described by the first rule: the measured data is stored and broadcasted to the network. The second rule describes the diffusion process: if the received measurement data contains brightest data than the stored one, the received data is stored and broadcasted. The third rule defines the network query.

The operation in mode 2 is similarly described in *Table 4*. The measurement and diffusion processes are more complicated – note the multiple-line rules.

The network is operated in mode 1 and if requirements change, the algorithm describing mode 2 can be diffused in the network, or part of it. The application was developed in the TinyOS environment and can be downloaded from [33].

## 6. Conclusions

This paper presented a self-adaptive framework using CSP-based models to describe behavioral elements in the system. The elements of the system can perform testing operations, and – if required –, can learn new rules from other system components, thus the whole system can adapt to changing requirements. The testing and learning are supported by a simple services process.

The feasibility of the described method was illustrated by two practical applications: an adaptive communication discovery protocol illustrated self-adaptive behavior of the system when the qualities of the communication channel changes. The sensor networking application illustrated self-adaptivity on application level: changing requirements induce changes in the application program.

**Acknowledgement**

This research was partially supported by the Hungarian Government under contracts NKFP2-00018/2005 and OMFB-00247/2007.

| | |
|---|---|
| 1. | *M,S[]=M[],c!S[];* |
| 2. | *c?G[],i(G[0]>S[0])(S[]=G[],c!S[])();* |
| 3. | *c?A,c!S[]* |

*Table 3. CSP rules in mode 1*

*Table 4. CSP rules in mode 2*

| | |
|---|---|
| 1. | *M,f(i=0,i<x,i++){i(M[0]>S[i][0])*<br>*(f(j=i+1,j<x,j++){S[j][]=S[j-1][]},S[i][]=M[])()},c!S[][];* |
| 2. | *c?G[][],l=0,f(i=0,i<(x+0),i++){i(G[k][0]>S[i][0])*<br>*(f(j=i+1,j<(x+0),j++){S[j][]=S[j-1][]},*<br>*S[i][]=G[k][],k++,l=1)(i(G[k][0]=S[i][0])(k++)())},i(l=1)(c!S[][])();* |
| 3. | *c?A,c!S[][]* |

*,where*
*c=channel, M=measured data, S=stored data, G=got data, f = for,*
*i = if, A=AskData, !=out, ?=in, []=array*

## Authors

**SZILÁRD JASKÓ** received his MSc Degree in Information Technology Engineering in 2002 at the University of Veszprém. Since then, he has been a PhD student at the same university. Since 2002, he has taught at the Department of Information Systems of the University of Pannonia. He is interested in protocol modeling and testing, self-adaptive technologies, mobile telecommunication and formal languages.

**GYULA SIMON** received his M.Sc. and Ph.D. in electrical engineering from the Budapest University of Technology, Hungary, in 1991 and 1998, respectively. Currently he is an Associate Professor at the University of Pannonia. His main research interest includes adaptive signal processing and sensor networks.

**KATALIN TARNAY** is the Professor of Mobile Communication and Telecommunications Software at the University of Pannonia. Her book "Protocol specification and testing" (1989) was published by Kluwer Academic Publisher in New York. She was the coeditor of the book "Testing communicating systems" (1999). Her current research field is protocol modeling and network management.

**TIBOR DULAI** received his MSc Degree in Information Technology Engineering in 2002 at the University of Veszprém. Since then, he has been a PhD student at the same university. Since 2002, he has taught at the Department of Information Systems of the University of Pannonia. His major research areas include protocol modeling, location based technologies, mobile telecommunication and formal languages.

**DÁNIEL MUHI** received his MSc Degree in Information Technology Engineering in 2002 at the University of Veszprém. Since then, he has been a PhD student at the same university. Since 2002, he has taught at the Department of Information Systems of the University of Pannonia. He is interested in protocol modeling, formal languages, e-learning systems.

## References


[1] Conrad, M., Hof, H.J.:
A generic, self-organizing, and distributed bootstrap service for peer-to-peer networks.
In: IWSOS 2007 – Self-Organizing Systems.
Lecture Notes in Computer Science LNCS 4725,
Springer Berlin/Heidelberg (2007), pp.59–72.

[2] Djenouri, D., Badache, N.:
Cross-layer approach to detect data packet droppers in mobile ad-hoc networks.
In: IWSOS 2006 – Self-Organizing Systems.
Lecture Notes in Computer Science LNCS 4124,
Springer Berlin/Heidelberg (2006), pp.163–176.

[3] Walter, U.:
Autonomous optimization of next generation networks.
In: IWSOS 2007 – Self-Organizing Systems.
Lecture Notes in Computer Science LNCS 4725,
Springer Berlin/Heidelberg (2007), pp.161–175.

[4] Jelasity, M., Babaoglu, O., Laddaga, R.,
Nagpal, R., Zambonelli, F., Sirer, E.G.,
Chaouchi, H., Smirnov, M.I.:
Interdisciplinary research:
Roles for self-organization.
IEEE Intelligent Systems 21(2) (2006), pp.50–58.

[5] Mösch, F., Litza, M., Auf, A.E.S., Maehle, E.,
Großpietsch, K.E., Brockmann, W.:
Orca – towards an organic robotic control architecture.
In: IWSOS 2006: Self-Organizing Systems.
Lecture Notes in Computer Science LNCS 4124,
(2006), pp.251–253.

[6] Pietzowski, A., Satzger, B., Trumler, W., Ungerer, T.:
A bio-inspired approach for self-protecting an organic middleware with artificial antibodies.
In: IWSOS 2006: Self-Organizing Systems.
Lecture Notes in Computer Science LNCS 4124,
Springer Berlin/Heidelberg (2006), pp.202–215.

[7] Harangozó, Z., Tarnay, K.:
FDTs in self-adaptive protocol specification.
In: Self- Adaptive Software: Applications.
Lecture Notes in Computer Science LNCS 2614,
Springer Berlin/Heidelberg (2002), pp.105–117.

[8] Ferscha, A.G.C.:
Self-adaptive logical processes:
The probabilistic distributed simulation protocol.
In: Proceedings of the 27th Annual Simulation Symposium, IEEE Computer Society Press,
LaJolla (1994).

[9] Estrin, D., Govindan, R., Heidemann, J., Kumar, S.:
Next century challenges:
scalable coordination in sensor networks.
In: MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, New York, NY, USA,
ACM (1999), pp.263–270.

[10] Bachrach, J., Nagpal, R., Salib, M., Shrobe, H.E.:
Experimental results for and theoretical analysis of a self-organizing global coordinate system for ad hoc sensor networks.
Telecommunication Systems 26(2-4) (2004),
pp.213–233.

[11] Li, H., Price, M.C., Stott, J., Marshall, I.W.:
The development of a wireless sensor network sensing node utilising adaptive self-diagnostics.
In: IWSOS 2007: Self-Organizing Systems.
Lecture Notes in Computer Science LNCS 4725,
Springer Berlin/Heidelberg (2007), pp.30–43.

[12] Cerpa, A.E., Estrin, D.:
ASCENT:
Adaptive self-configuring sensor networks topologies.
IEEE Transactions on Mobile Computing,
3(3) (2004), pp.272–285.

[13] Peleska, J.:
Applied formal methods –
from csp to executable hybrid specifications.
In: Communicating Sequential Processes.
Lecture Notes in Computer Science LNCS 3525,
Springer Berlin/Heidelberg (2005), pp.293–320.
[14] He, J., Hoare, C.A.R.:
Algebraic specification and proof of
a distributed recovery algorithm.
Distributed Computing 2(1) (1987), pp.1–12.
[15] Urban, G., Kolinowitz, H.J., Peleska, J.:
A survivable avionics system
for space applications.
In: FTCS (1998), pp.372–381.
[16] Schneider, S.:
Concurrent and Real-time Systems the CSP Approach.
Wiley and Sons Ltd. (2000).
[17] Hoare, C.:
Communicating Sequential Processes.
Prentice Hall, NJ (1985).
[18] ROSCOE, A.:
The Theory and Practice of Concurrency.
Prentice Hall (1997).
[19] Abramsky, S.:
Interaction categories and
communicating sequential processes.
Prentice Hall (1994).
[20] INMOS: occam Programming Manual.
Prentice Hall (1984).
[21] Schneider, S., Delicata, R.:
Verifying security protocols:
An application of CSP.
In: SOFSEM'99: Theory and Practice of Informatics.
Lecture Notes in Computer Science LNCS 3225,
(2005), pp.243–263.
[22] Natanson, L.D., Samson, W.B.:
An animator for CSP implemented in HOPE.
In: Proceedings of the BCS-FACS Workshop on
Specification and Verification of Concurrent Systems,
London, UK, Springer-Verlag (1990), pp.575–594.
[23] Formal Systems (Europe) Ltd.:
Failures-Divergence Refinement:
FDR2 User Manual, (1997).
[24] Bin, E., Emek, R., Shurek, G., Ziv, A.:
Using a constraint satisfaction formulation and solution
techniques for random test program generation.
IBM Sytems Journal 41(3) (2002), pp.386–402.
[25] Jifeng, H., Hoare, C.A.R.:
SDL- and MSC-Based Specification and
Automated Test Case Generation for INAP.
Telecommunication Systems 20, pp.265–290.
[26] Engels, A., Feijs, L.M.G., Mauw, S.:
Test generation for intelligent networks
using model checking.
In: TACAS (1997), pp.384–398.
[27] Dulai, T., Jaskó, Sz., Muhi, D., Tarnay, K.:
CSP model of self-adaptive test system.
IWSAS, Washington D.C. (2003).
[28] Jaskó, Sz., Dulai, T., Muhi, D., Tarnay, K.:
Process-based model for test system.
ISDA, Budapest (2004).
[29] http://uni-pen.hu/docs/misc/javabaseddemo.zip
[30] Mainland, G., Parkes, D.C., Welsh, M.:
Decentralized, adaptive resource allocation
for sensor networks.
In: NSDI'05: Proceedings of the 2nd conference
on Symposium on Networked Systems Design &
Implementation, Berkeley, CA, USA,
USENIX Association (2005), pp.315–328.
[31] Jin, G., Nittel, S.:
UDC: a self-adaptive uneven clustering protocol
for dynamic sensor networks.
Int. J. Sen. Netw. 2(1/2) (2007), pp.25–33.
[32] Hill, J.L., Culler, D.E.:
Mica: A wireless platform
for deeply embedded networks.
IEEE Micro 22(6) (2002), pp.12–24.
[33] http://uni-pen.hu/docs/misc/tinyosbaseddemo.zip
[34] http://www.mcrl2.org/mcrl2/wiki/index.php/Home
[35] http://www.comp.nus.edu.sg/~pat/

# VoIP LAN/MAN traffic analysis for NGN QoS management

ZOLTÁN GÁL

*Center for Information Technologies*
*Centre of Arts, Humanities and Sciences, University of Debrecen*
*zgal@unideb.hu*

*Keywords: NGN, TCP, UDP, codec, QoS, DiffServ, self similarity, wavelet, fractal, entropy*

**Strict requirement is emphasized regarding QoS guarantees of the NGN (Next Generation Network) networks today. DiffServ mechanism is applied mostly for classification of protocol data units of real time and conventional information streams in LAN/MAN environment. The dependence of VoIP traffic characteristics of the delay and the jitter sensitive IP telephony vs. voice codec applied can be considered an exciting scientific question. We analyze Ethernet traffic generated by G.711, G.723, G.728 and Wideband (G.722) voice codecs. The self similar, fractal and multifractal properties of popular TCP based services (http, ftp, telnet, etc.) in LAN/MAN environment are well known for several decades. In this paper, we study the effect of UDP based current voice mechanisms on self similarity of the Ethernet data traffic. UDP traffics of the IP phones are evaluated in congested and congestionless environment using sophisticated methods of entropy and wavelet analysis. A new and efficient evaluation method, named ON/(ON+OFF) transformation is applied to the characterization of VoIP traffic.**

## 1. Introduction

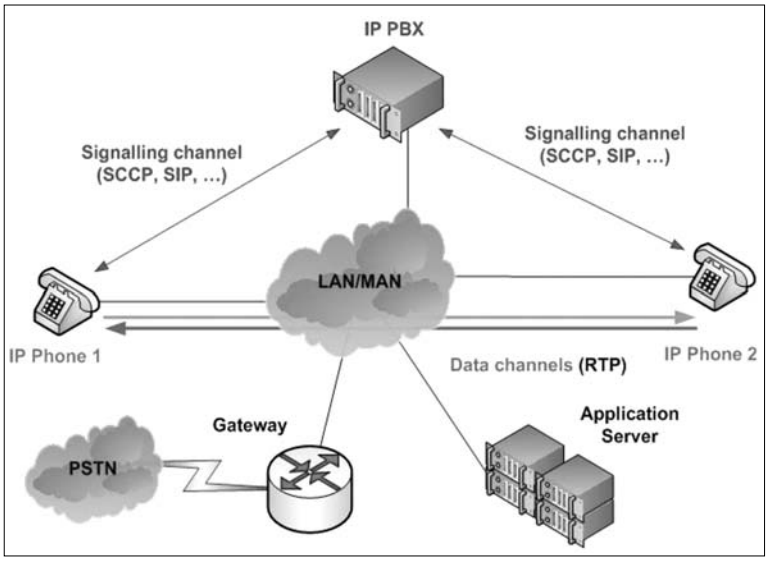
One of the most important time critical services of current and also of ITU-T NGN (Next Generation Network) communication networks in the near future is voice traffic. The VoIP (Voice over IP) technology radically transformed also the cost of telephone service and the behavior of subscribers. IP phone service exploits efficiently the Internet based network infrastructure and approximates the quality of PSTN traffic services. The best-effort transmission mechanism of IP networks cannot assure guarantees for delay sensitive voice traffic, hence for successful utilization of VoIP QoS techniques are needed among the end nodes. The evaluation characteristics of modeling aggregated voice and other type of traffics implies selection of optimal QoS mechanisms. The network traffic coming from a voice source depends strongly on the utilized voice codec type. These codecs are grouped into two classes: constant bit rate mechanisms (e.g. G.711), as well as silence supression mechanisms based on repeating ON and OFF periods of activity (e.g. G.728, GSMFR, G.722) [4].

Actual packet switching voice systems include not only IP traffic capable phone end nodes, but application servers responsible for signaling and cost accounting as well *(Figure 1)*. Signaling methods (like SSCP, SIP, etc.) in this environment are much more intelligent than those used in PSTN networks (e.g. QSIG). The voice content transmission is realized by RTP (Real Time Protocol) directly between IP phone nodes. Signaling is transmitted in TCP, and digitized voice is transmitted in UDP segments.

To interconnect VoIP and PSTN networks special gateways are used that are capable of converting signaling and voice content as well. The IP phone compiles messages from the sampled voice and assembles voice segments by a codec module *(Figure 2)*. Voice segments transmitted by RTP on top of UDP are shaped by jitter buffers at the receiving node. Modules like G.711, G.723, G.728, and GSM are called narrow-band codecs and utilize at most 64 kbps voice rate [4]. Codecs like BrandVoice32, G.722, etc. need higher bandwidth for the increased voice quality.

The main characteristics of IP voice codes are: voice bit rate, length of voice frame (80-520 bytes), duration of voice frame (0.125-20 ms), the IP packet bandwidth (24-272 kbps) and the delay of voice transfer (0.25-40 ms).

*Figure 1. IP phone and VoIP architecture*



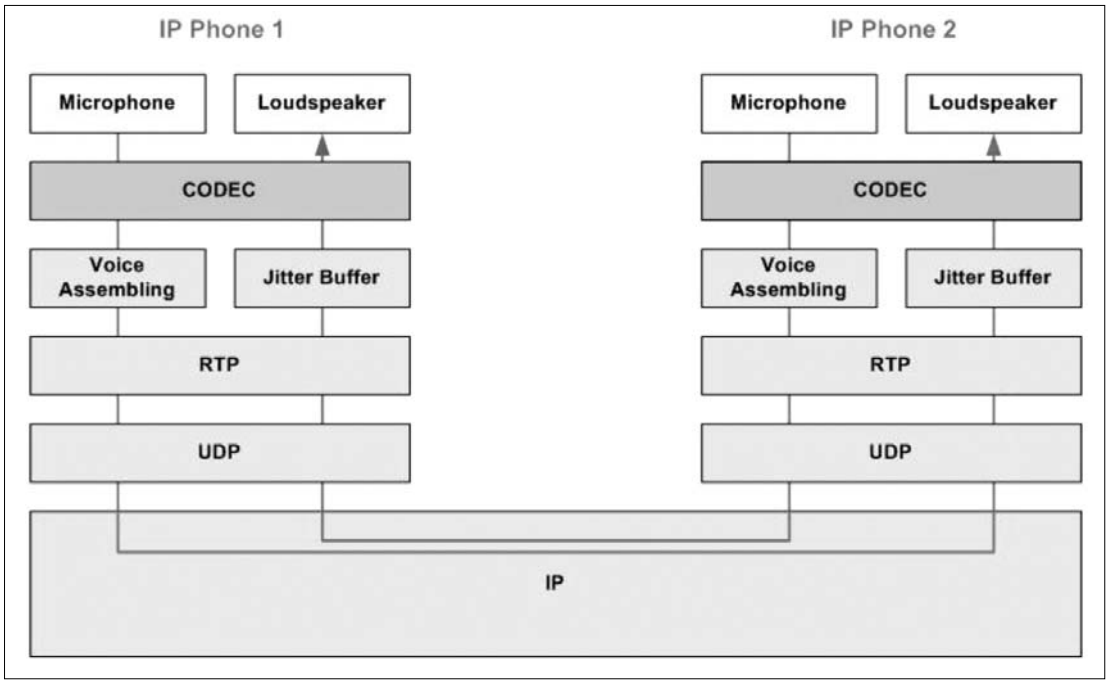



*Figure 2. VoIP transmission model*

These parameters depend on the codec type. The delay is influenced by eight mechanisms: sample buffering, coding, packet assembling, transmitting, transfering through LAN/MAN, receive buffering, decoding and playing back. To guarantee interactivity, this delay can be at most 200 ms (high voice quality) and 400 ms (eligible quality) respectively.

Different applications need distinct requirements regarding transferring the traffic through the LAN/MAN. The necessary network resource is a function of the traffic volume generated. Some applications are more tolerant of the transfer delay and delay variation, but others cannot tolerate a given transmission error limit excess. Basic QoS parameters are bandwidth, latency, jitter and transmission error rate [4]. The QoS algorithms manage the network resource utilization: dedicated bandwidth allocation, monitoring the transmission error characteristics, congestion avoidance and management, traffic shaping, traffic prioritization.

The three well known QoS mechanisms are: *Best-Effort, Intserv* and *Diffserv*. Intserv is based on IETF RSVP (Resource ReserVation Protocol) for QoS between end user nodes. Diffserv works among intermediate nodes on L2-L7 layers of OSI. The most important feature of "coloring" the IP traffic is the DSCP (Diffserv Code Point) field in the IP packet header. The ingress interface needs classification, marking, policing, shaping (e.g. FIFO, FQ-Fair Queue, WFQ-Weighted Fair Queue, WRED-Weighted Random Early Detected, "tail-drop", LLQ-Low Latency Queuing), while the egress interface needs congestion avoidance, policing, and shaping tasks, respectively. The voice is classified as the traffic with the highest priority.

Modeling packet switched networks is mainly characterized by manipulating only the packet arriving time series as a stochastic process [8]. There are only few papers that study both the length and the arriving time of packets in evaluation of PDU real time transmission performance [9]. In this paper, we evaluate both arrival time and length of packets as time series for evaluating the performance of QoS mechanisms. Because the bandwidth of the wired channel between two intermediate nodes in most cases (i.e. IEEE 802.3 protocol family) is technology dependent and is fixed, the frame size in our case can be transformed linearly into time dimension. Utilizing a new transformation, called ON/(ON+OFF), two time series can be obtained: channel utilization and packet/channel intensity. These measures can be used favorably for complex evaluation of the different QoS mechanisms.

In Section 2, performance characteristics of the IP networks and a special measure based on entropy, called Corvil bandwidth, are presented. In Section 3, wavelet analysis of self similar processes is introduced. Sophisticated methods of entropy and wavelet analysis applied on traces of IP phones in congested and congestion-less environment will be presented in Section 4. In the concluding section we also outline potential future work.

## 2. Performance characteristics of IP networks, entropy, Corvil bandwidth

The performance of current IP network applications is influenced by three factors: *bandwidth, statistical multiplexing* and *QoS mechanisms* [2]. The task of measuring *bandwidth* is relatively simple because intermediate nodes (routers, switches) are able to store five minutes of average values based on SNMP MIB objects. The detected values provide a measure of traffic traversing the network, but do not calculate exactly the necessary bandwidth for different network applications. Packet discarding and jitter are strongly influenced by the

behavior of traffic at the millisecond time scale. There is no detailed information to predict the throughput of applications. In case of medium sized networks VoIP applications tolerate several 10 milliseconds of jitter. Practical experience of given network applications resulted in some empirical rules.

For example, in Best-Effort IP services, only 60% of the five minutes periods with a load above 95% for the network resources is recommended. If the load intensity exceeds this 60% threshold, the infrastructure needs to be improved. The percentages above are empirical and cannot be generalized for any network services. In VoIP environment the 60% load intensity referred above should be decreased to 40%.

An important tool of service providers is the *gain of statistical multiplexing,* which shares the resources of the packet switched networks randomly. When transmitting ten pieces of different video streams with the same characteristics on circuit switched network exactly ten times the bandwidth of a single stream is needed. In the case a packet switched environment significantly less aggregated bandwidth is needed. The reason is that the asynchronity of short bursts of the different streams making aggregated traffic is more shaped. The difference between the sum of individual bandwidths and the effective aggregated bandwidth is called gain of statistical multiplexing and is a measure of IP network performance characteristics.

Traffic shaping, policing and differentiated queuing are the most efficient *QoS mechanisms*. Dimensioning the bandwidth for robust statistical reliability, guaranteeing the prescribed gain of statistical multiplexing, and setting the QoS mechanisms can be managed in two ways: the first method is realization of highest gain with the requested quality guarantees, and the second method is based on extra dimensioning of network resources to produce performance. In the first case special services cannot be assured, while expensive network infrastructure is needed in the second case.

The uncertainty feature of network traffic is observed by the non deterministic relation among these three factors [3]. The network bandwidth, the traffic load, and the QoS objectives are essentially inter-related. Modification of either of them, influence the relation between others. The bandwidth necessary for a guaranteed delay depends not only on the network load, but also on the type of traffic (VoIP, data). The following set of formula shows the relationship among quality, network and traffic:

$$\left.\begin{aligned} \text{Quality} &= f_Q(\text{Network}, \text{Traffic}) \\ \text{Network} &= f_N(\text{Traffic}, \text{Quality}) \\ \text{Traffic} &= f_T(\text{Quality}, \text{Network}) \end{aligned}\right\} \quad (2.1)$$

The CB (Corvil Bandwidth) technology gives proprietary solution to the equation system above in a given network environment. The method is based on intensive sampling of the traffic to extrapolate rules for dimensioning the network resources. The bandwidth is the simplest metric and can be measured relatively easily. Today SLAs (Service Level Agreement) specify the acceptable packet discard rate and delay parameters with SNMP, which are evaluated on weekly or monthly time scales.

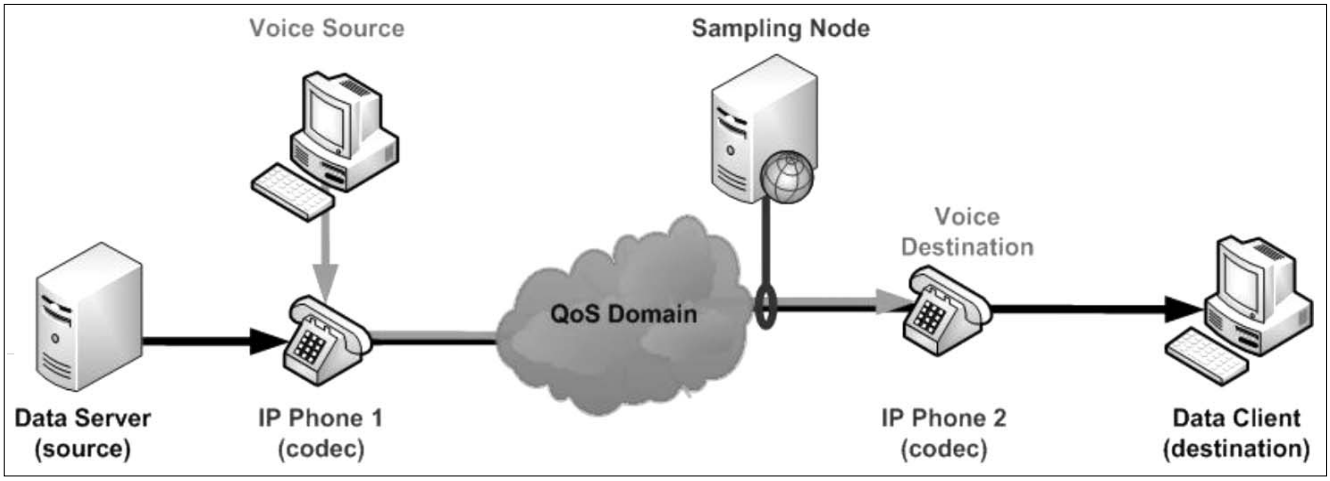
These are too rough values to assure guarantees for special or real time applications. The CB technology is based on Large Deviation theory applied to queuing systems, which analyzes the entropy, being the key statistical characteristic of a given queuing system. The entropy of a packet trace describes queue management and multiplexing of different traffics inside of network devices.

$$\text{CB} = f_Q(\text{Corvil Entropy}, \text{QoS}) \quad (2.2)$$

Some CB dimensioning rules of traffic classes or interfaces are the following: the queue processing delay is 0.001...1 sec and the size threshold is 1...2000 packets; the percentage of protected packets is 1...100% in steps of 0.0001%; the periods of protection rules are: 5 minutes, 1/2/4/ hours, 1 day, 1 week. The CB measured for 5 minutes differs strongly from the bandwidth measured by SNMP because CB considers sub-millisecond features of traffic. The real bandwidth need is set as a function of transfer delay and packet discard rate.

All the considerations regarding entropy suggest statistical analysis be effectuated at sub-millisecond time scales.

*Figure 3. Measurement environment of the VoIP traffic*



## 3. Wavelet analysis of self similar processes

A real valued $\{Y(t), t \in R\}$ process is self similar (H-ss) with Hurst parameter (H> 0), if for $\forall a > 0$ exists: $Y(at) \stackrel{def}{=} a^H Y(t)$ [8]. A real valued $\{Y(t), t \in \boldsymbol{R}\}$ process is H-sssi, if it is H-ss with stationary increments. If $\{Y(t)\}$ H-sssi has finite deviation, then $0 < H \leq 1$. A discrete increment time series can be created by $X_k = Y(k) - Y(k-1), k = 1,2, \ldots$ . Let $X^{(m)}$ and $r^{(m)}(\cdot)$ be the m-aggregated time series and the autocorrelation function of X, respectively, where $X_k^{(m)} = \frac{1}{m}\sum_{i=(k-1)m+1}^{km} X_i$. In case of $0 < H < 0.5$ the process is called short range dependent (SRD), and for $0.5 < H < 1$ the process is long range dependent (LRD). If the process is LRD, then the form of autocorrelation function of the increment process is: $r(k) = \frac{1}{2}[(k+1)^{2H} - 2k^{2H} + (k-1)^{2H}]$. For exactly self similar process the deviation of aggregated increment process is $var\left(X^{(m)}\right) = m^{2H-2} var(X)$, and $r^{(m)}(k) = r(k)$. It can be observed that for LRD $var\left(X^{(m)}\right) > m^{-1} var(X)$, and for SRD $var\left(X^{(m)}\right) < m^{-1} var(X)$. Process X is asymptotically self similar, if for k high enough, holds: $\lim_{m\to\infty} r^{(m)}(k) = r(k)$ [7].

The discrete wavelet transformation (DTW) is a time-frequency decomposition, which assigns two-variable coefficients to the time series *X* with *n* elements, in the following formula [1]:

$$d_{j,k} = \int X(s)\psi_{j,k}(s)ds, j \in \boldsymbol{Z}, k \in \boldsymbol{Z} \qquad (3.1)$$

where any wavelet has the following unique expression:

$$\psi_{j,k}(s) = 2^{-j/2}\psi(2^{-j}t - k) \qquad (3.2)$$

Several mother wavelets exist and each has the following features:

$$\int t^k \psi(t)dt \equiv 0, \ \forall k = 1,2, \ldots, N-1 \qquad (3.3)$$

The wavelet decomposition is a linear combination of mother wavelet functions and coefficients $d_{j,k}$:

$$X(t) = \sum_{j\in\boldsymbol{Z}} \sum_{k\in\boldsymbol{Z}} d_{j,k}\psi_{j,k}(t) \qquad (3.4)$$

The wavelet coefficients can be used to evaluate scale, and frequency dependence features of LRD processes. The second order log-scale diagram (2-LD) is a log-linear graph of the estimated second moment dependent of the octave *j*:

$$\mu_j = \frac{1}{n_j}\sum_{k=1}^{n_j} |d_{j,k}|^2 \approx 2^{j(2H-1)}, \text{ where } n_j = 2^{-j}n \qquad (3.5)$$

The average of square sum of the wavelet coefficients is called energy function, $\mu_j$ of the time series *X*. Conforming to (3.5), the logarithm of the energy function is a linear function of octave *j*.

$$y_j = log_2(\mu_j) \approx (2H-1)j + c \qquad (3.6)$$

For the estimation of the Hurst parameter linear segment or segments of 2-LD can be utilized. If more than one linear segment can be identified, the process is multifractal, otherwise is monofractal. The *H* parameter for the linear octave segment $[j_1, j_2]$ can be evaluated by the WLS (Weighted Less Squares) method with the following formula [1]:

$$\widehat{H}(j_1, j_2) = \frac{1}{2}\left|\frac{\sum_{j=j_1}^{j_2} S_j j y_j - \sum_{j=j_1}^{j_2} S_j j \sum_{j=j_1}^{j_2} S_j y_j}{\sum_{j=j_1}^{j_2} S_j \sum_{j=j_1}^{j_2} S_j j^2 - \left(\sum_{j=j_1}^{j_2} S_j j\right)^2} + 1\right|, \quad where\ S_j = \frac{n\ ln^2 2}{2^{j+1}}\ are\ the\ weights. \qquad (3.7)$$

## 4. Measurement environment and analysis of the measured processes

### a) Analysis of VoIP traffics in congested environment

The link between the source and destination is 10 Mbps Ethernet, on which (T) TCP and (U) UDP traffics were generated at the maximum load of the channel with Java based IPerf server and client entities running on the Data Server and the Data Client, respectively. Both voice and data traffic were transmitted through the LAN interface of the IP phones *(Figure 3)*.

One minute long songs, (H) hard rock (Limp Bizkit – Eat You Alive) and (P) piano (Wolfgang Amadeus Mozart – Concert for horn and orchestra KV KV 285d C major Adagio non troppo) were repeated on the source and transmitted from the IP Phone_1 to IP Phone_2. Different types of codecs (G.728, GSM, G.711, WideBand-G.722) were utilized at the IP phones, while the voice traffic inside of QoS LAN/MAN domain was regulated by DSCP values set for 0x00-"best-effort", 0x02-low price, 0x04-reliable, 0x08-performance, 0x10-low latency. The eighty different traffic traces were created by varying the transport protocol, the song type, the codec type, and the DSCP value: [(T,U) x (H,P) x (G.728,GSM,G.711,WB) x (0,2,4,8,16)] = 2x2x4x5 = 80 traces. These traces were captured by the program Wireshark with 1 μsec accuracy.

### b) Analysis of VoIP trunk traffics in congestion-less environment

The aggregated traffic on the voice VLAN of IP/PBX gateway was captured for a population of 1500 IP phones [6]. The voice trunk link was 100 Mbps Ethernet and the capturing task was effectuated with 1 μsec accuracy in university environment on a working day for a one hour time interval.

For both a) and b) scenarios arrival time and the length of the Ethernet frames were captured. In case of voice transmission the Ethernet MTU (1500 B) is at least two times higher than the VoIP packet size, so there were no packet fragmentations. Because Ethernet is the mostly applied technology in the access and the dist-

ribution network layers today, the measurement scenario and the packet traffic behaviors can be considered for LAN and MAN environment as well. The 1 μsec accuracy of the L2 PDU capturing is caused by the Corvil bandwidth considerations in Section 2.

Metrics studied in network traffic analysis are related mostly to the effect of the network resources utilization (e.g. frame inter-arrival time, frame size). Lots of dominant papers discuss statistical features (LRD, fractal, multifractal) of packet switching based on measures [9,7]. A new transformation, called ON/(ON+OFF) is proposed and introduced in this paper, which offers load evaluation of the network resources directly. Our implication is that analyzing the finite set of network resources (channel load, channel activation intensity) provides more suggestive measures of the communication processes and gives more clear view of the instantaneous network resources state.

The length of the frame given in bits can be converted in time dimension, making it possible to calculate two time series $L_i$, the average transmission time interval ($ON_i$), and $tan(\varphi_i)$, the channel load. The evaluation time interval $T = M * (T_{ON} + T_{OFF})$ was fixed, and for each measurement, $T = 100\ ms$. $M_i$ is the intensity of frame arriving and can be considered as the intensity of channel activation in the evaluation time period $i$. The instantaneous channel load and the phase of voice traffic for evaluation period $i$ can be calculated with the formulae (4.1). Basic time diagrams of the ON/(ON+OFF) transformation are presented in *Figure 4*.

$$\left.\begin{array}{ll} M_i\ [\,], & Intensity\ of\ channel\ activation \\ L_i[sec], & \text{Average}\ frame\ processing\ time \\ D_i = \sqrt{L_i^2 + T^2}\ [\text{sec}], & Square\ average\ time \\ \tan(\varphi_i)\ [\%] = \frac{L_i}{T} * 100, & \text{Average}\ channel\ load \\ \varphi_i[Rad] = \tan^{-1}(\frac{L_i}{T}), & \text{Average}\ channel\ phase \end{array}\right\} \quad (4.1)$$

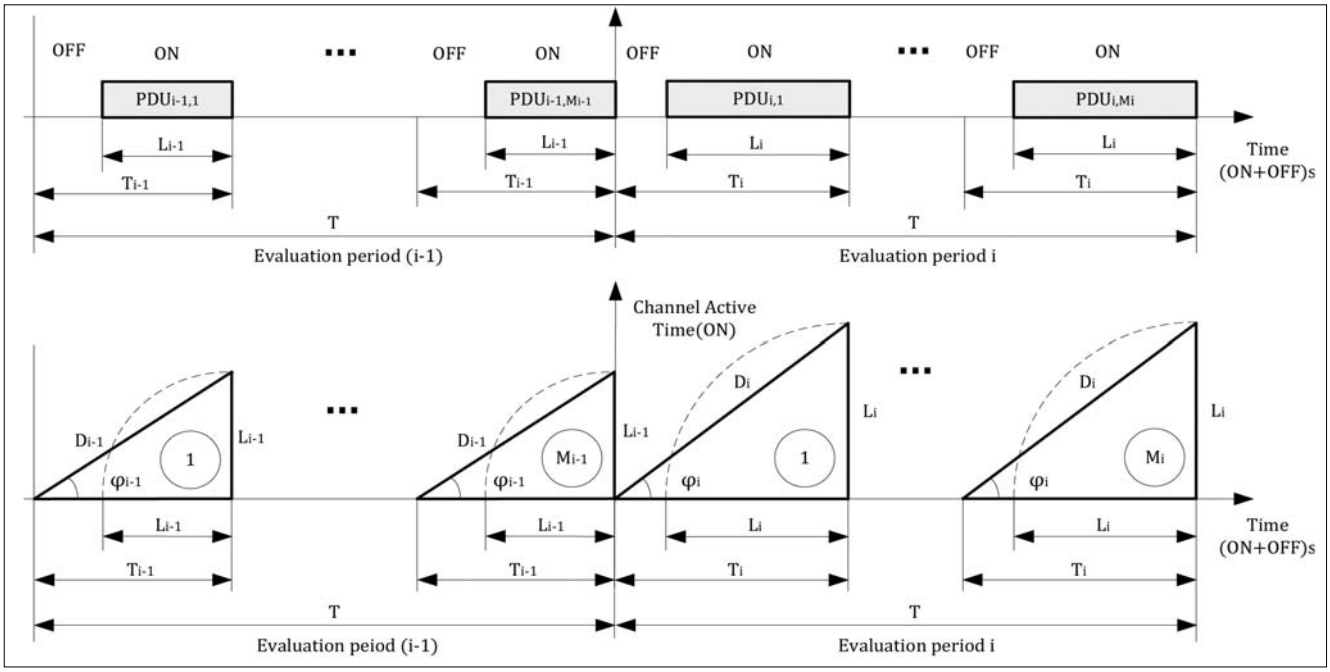
**For VoIP congested environment** the main characteristics of the four types of codecs are presented in *Figure 5*, and the relative deviation of the eighty intensity series can be seen in *Figure 6*. The relative deviation of the channel load and the intensity for voice are practically the same. In Figure 6, the dark colors mean smaller values, and the light colors mean larger values.

In case of DSCP=0 ("best effort"), TCP data traffic and G.711 voice codec, the relative deviation of voice traffic intensity is small, but for other types of codecs this can be as high as 20% (for GSM). When the voice traffic is treated with QoS mechanisms, this relative deviation remains at low values. In case of UDP data traffic the relative deviations of voice loads are high, but with TCP data traffic these are small. This phenomenon is caused by the TCP flow control mechanisms, which decrease and shape the TCP data traffic in favor of the UDP voice traffic. For UDP data traffic there is no flow control, and the voice traffics without QoS have larger deviations. The dynamics of songs has an effect on the voice traffic load only for low bit rate codecs (G.728, GSM).

*Figures 7–10* present the channel load time series and their wavelet transforms in cases of UDP data traffic, GSM codec, "best-effort"/QoS and piano/hard rock song environments. Although the characteristics of the two time series are comparable, the main difference is shown expressively by the wavelet transforms.

*Table 1* and *Figure 11* show the estimated Hurst parameter, ($\hat{H}$) of the channel load and of the intensity for all eighty traces. Irrespective of the data flow transport protocols, in cases of G.728 and GSM codecs, the channel load time series are not self similar, why the $\hat{H} > 1$. In contrary, for G.711 and G.722 (WB) codecs the voice time series are self similar and LRD. Irrespective of the song dynamics and data flow transport protocols, in cas-

*Figure 4. Basics of the ON/(ON+OFF) transformation*



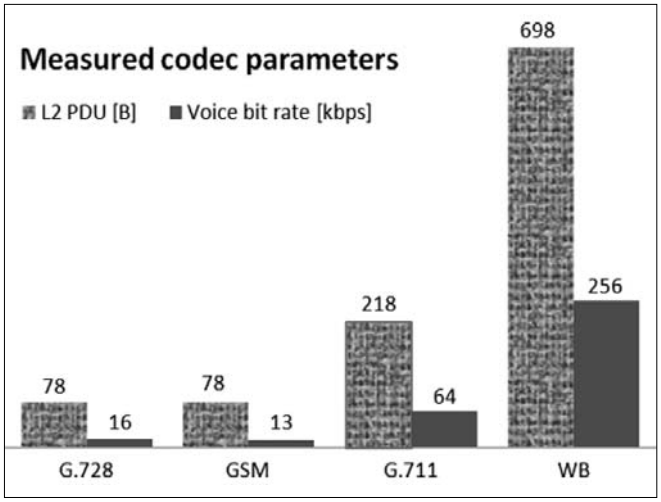



*Figure 5. Characteristics of voice codecs*

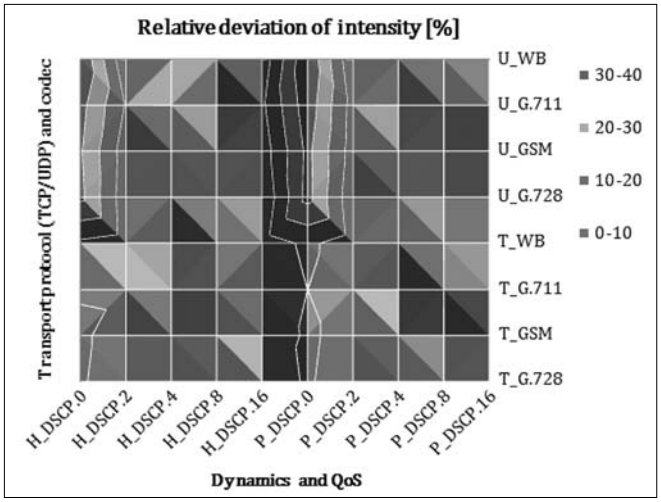



*Figure 6. Intensity, M*

es without QoS (i.e. DSCP=0), the voice channel load is H-sssi and LRD, and for the estimated Hurst parameter, $\hat{H} \in [0.56, 0.91]$.

The estimated Hurst parameter, $\hat{H}$ of voice traffic load varies contrary with the bandwidth of voice codec (Figure 5 vs. Table 1). The experienced voice quality at the receiving IP phone was better for codecs with higher bandwidth, and the congestion of voice traffic was not subjectively detectable when QoS mechanisms were active.

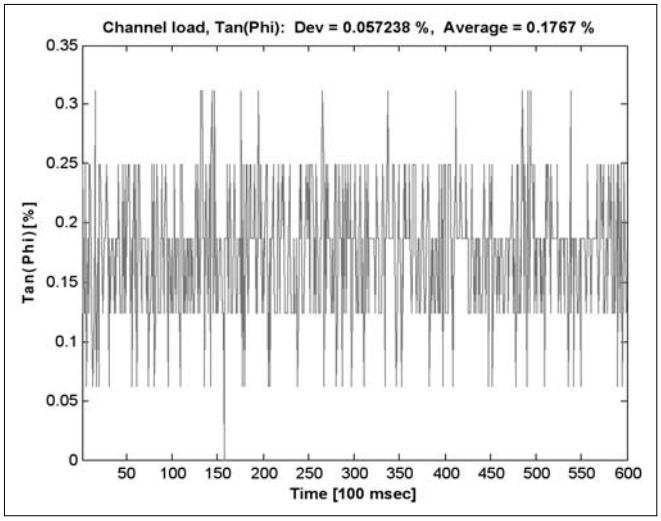



*Figure 7. Channel load, Tan(φ) – UDP, GSM, no QoS, Piano –*

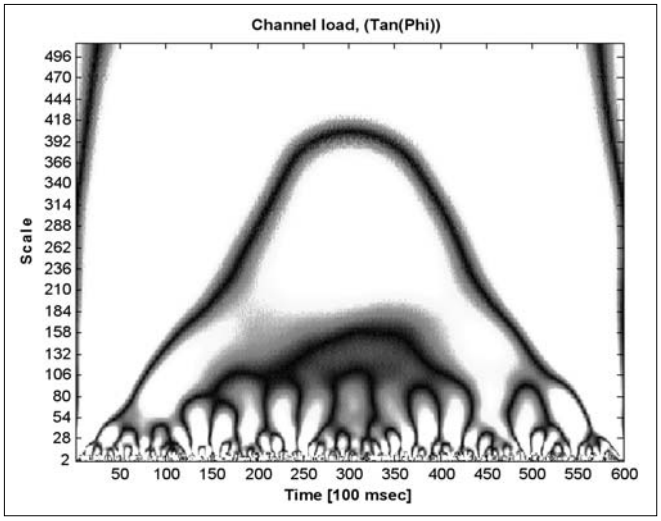



*Figure 8. Wavelet transform, Tan(φ)*

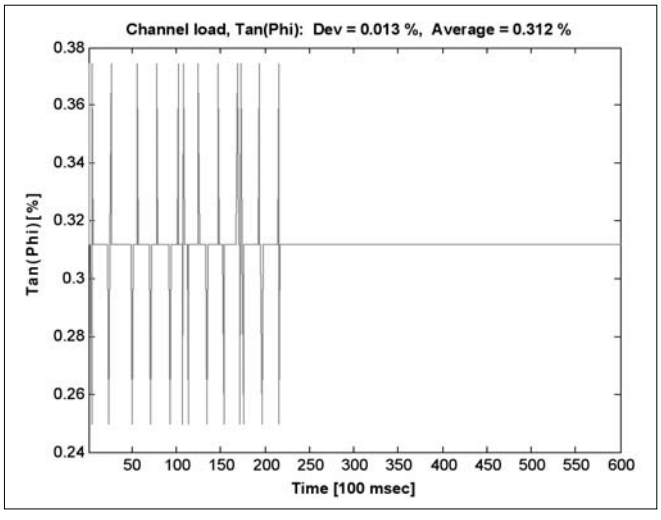



*Figure 9. Channel load, Tan(φ) – UDP, GSM, DSCP=16, Hard Rock –*

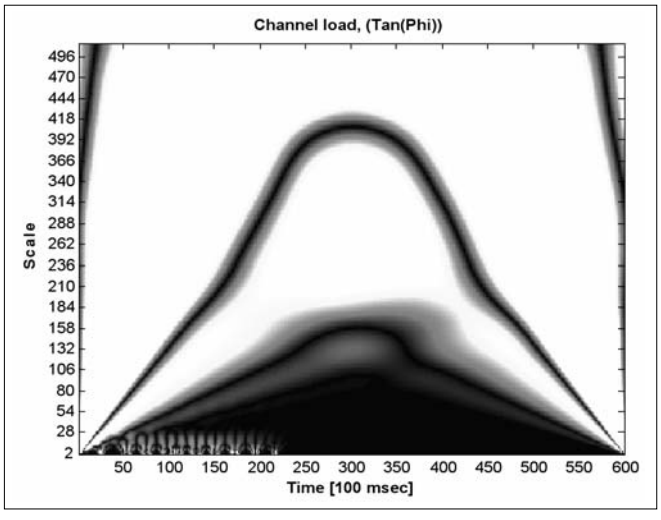
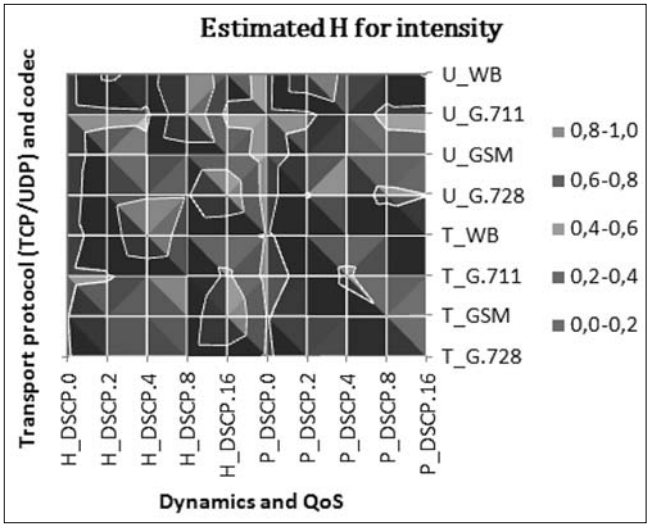



*Figure 10. Wavelet transform, Tan(φ)*

| | T G.728 | T GSM | T G.711 | T WB | U G.728 | U GSM | U G.711 | U WB |
|---|---|---|---|---|---|---|---|---|
| H_DSCP.0 | 0,86 | 0,87 | 0,79 | 0,59 | 0,92 | 0,91 | 0,73 | 0,57 |
| H_DSCP.2 | B | B | 0,73 | 0,76 | B | B | 0,75 | 0,93 |
| H_DSCP.4 | B | B | 0,83 | B | B | B | 0,80 | 0,81 |
| H_DSCP.8 | B | B | 0,74 | 0,76 | B | B | B | B |
| H_DSCP.16 | B | B | B | 0,74 | B | B | 0,77 | 0,79 |
| P_DSCP.0 | 0,87 | 0,85 | 0,78 | 0,57 | 0,91 | 0,89 | 0,72 | 0,56 |
| P_DSCP.2 | B | B | 0,83 | 0,77 | B | B | 0,80 | B |
| P_DSCP.4 | B | B | B | 0,79 | B | B | 0,96 | 0,82 |
| P_DSCP.8 | B | B | 0,75 | 0,77 | B | B | 0,78 | 0,81 |
| P_DSCP.16 | B | B | 0,74 | 0,81 | B | B | 0,76 | 0,90 |

*Table 1. Estimated Hurst parameter of Tan(φ)*

*Figure 11. Estimated Hurst parameter of M*




The intensity time series are LRD and for each analyzed case $\hat{H} \in [0.52, 1]$. Irrespective of the data flow transport protocols, in cases without QoS the estimated Hurst parameter of the voice traffic intensity is higher but near 0.5, $\hat{H} \in [0.51, 0.6]$. In case of G.711 codec, the song with high dynamics and *DSCP*= 8, performance optimization QoS mechanism causes high value for estimated Hurst parameter of voice traffic intensity. The intensity of this type of song traffics produces higher $\hat{H}$, than songs with low dynamics (Figure 11.)

**For congestion-less VoIP environment**, *Figures 12-13* present the channel load and the intensity of voice trunk, *Figures 14-15* present the 2-LD charts of them, according to relation (3.7). Although the shifting averages of the graphs indicate correlation, the characteristics of these two time series differ significantly because of the local maximums of the intensity. The 1 second length shifting average function of the channel load indicates the number of simultaneous voice sessions. The relative deviation of the channel load is 53%, and of the channel intensity is just 44%. The VoIP trunk traffic in congestion-less environment exhibits a multifractal feature as well.

The wavelet estimation of Hurst parameter of the channel load is $\hat{H} = 0.88$ and is less scale dependent, but for the intensity time series $\hat{H} = 0.61$ and for larger octaves is radically varying (Figures 14-15). The Ethernet link is congestion-less because of the small channel load, and the aggregated voice traffic is self similar and LRD.

## 5. Conclusions and future works

Voice transfer over IP networks is the most critical real-time network application and providing this service is a complex task and for service providers in IP LAN/MAN

*Figure 12. VoIP trunk channel load*

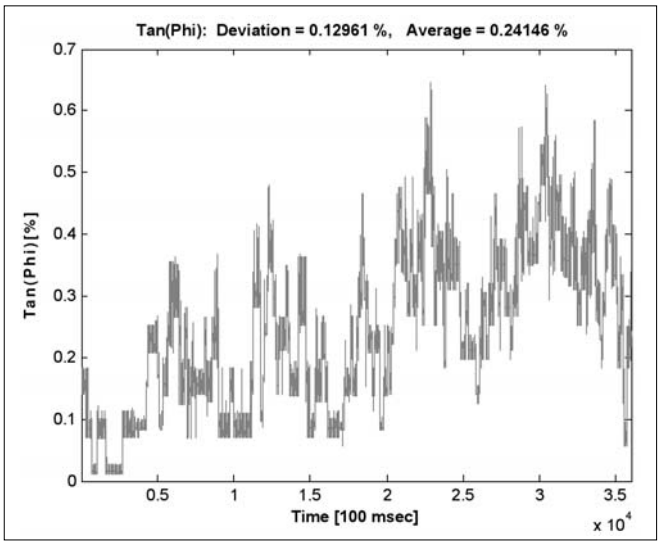
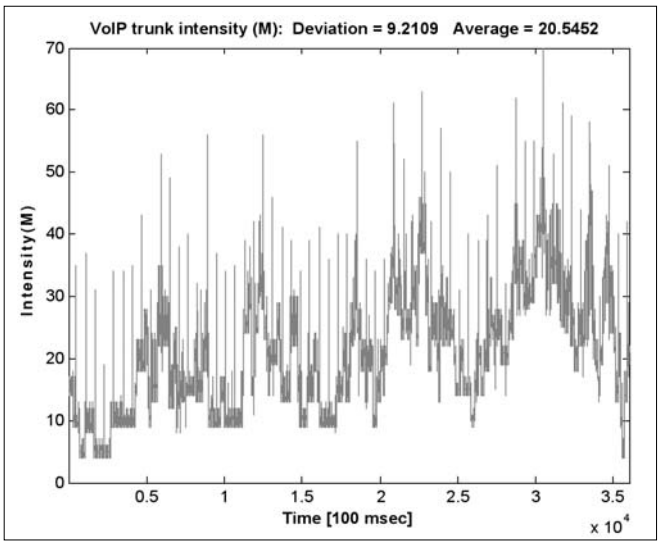



*Figure 13. VoIP trunk channel intensity*



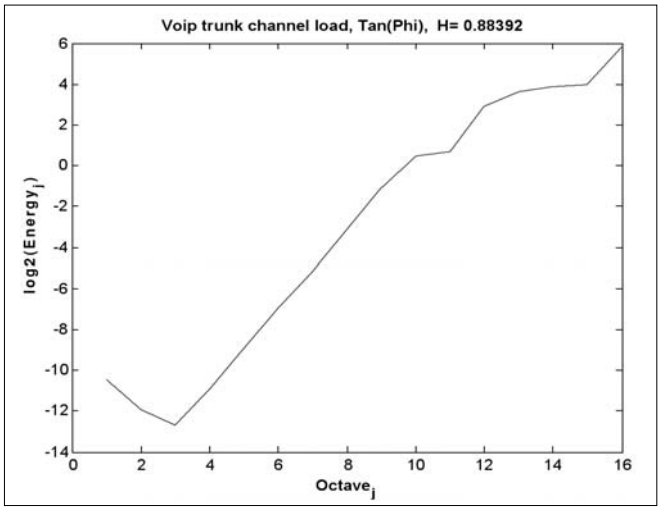



*Figure 14. VoIP trunk channel load – H estimated ($\hat{H}$) with wavelet method –*

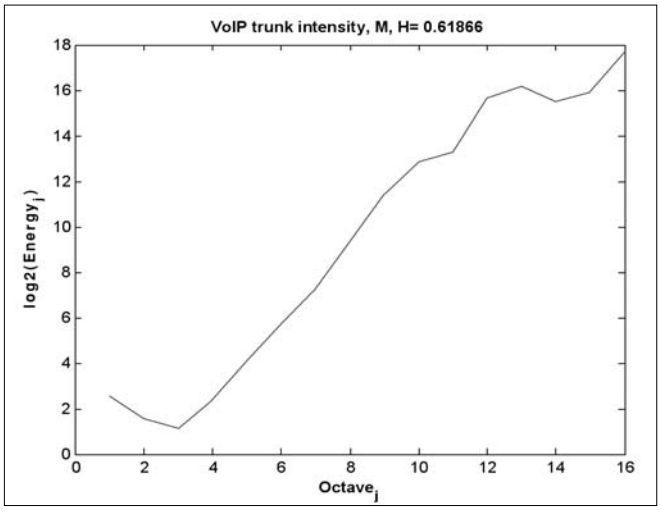



*Figure 15. VoIP trunk intensity*

environment it could be more difficult task than real-time video transmission service. The baseline QoS traffic classes based on the packet loss concealment threshold gives this priority order because only 20 ms are allowed for VoIP and higher values are permitted for real time video. The type of the codec determines the quality of the voice channel. The VoIP channel works on UDP transmission protocol, where there is no feedback, no flow control and no voice frame length modification during the session. The measured and subjectively sensed quality of the codec depends of both the physical channel bit rate and the packet switched protocol data unit size. In case of analyzed scenarios the increasing order of voice codec quality is: G.728, GSM, G.711, WB (G.722). This order was detected by independent human users during the measurements as well. The fractal phenomenon of L2/L3 voice traffic in congested LAN/MAN environment produces significant degradation of the voice quality. Wavelet analysis is an efficient tool for evaluating the H parameter, the fractal and scale dependent features of packet switched voice traffics, together with offering suggestive classification method of QoS controlled packet traffics. Based on the proposed ON/(ON+OFF) transformation VoIP network operators can determine more exactly the necessary NGN resources for a given number of voice subscribers not only in best-effort, but in congested and QoS controlled wired LAN/MAN environment, too.

QoS is becoming a service in the control plane of modern network protocols and strongly influences the self similar and fractal nature of transmission processes of the packet switched protocol data units. The quality of voice transmissions, the self similar and LRD features of these traffics are greatly influenced by the QoS mechanisms. This impact opens new directions in the area of QoS based flow control research area. More complex analysis is needed for employing both the channel load and intensity metrics of the IP packet switched protocol data unit traffic simultaneously. Based on the results of deep traffic research, optimal configuration profiles can be set for intermediate nodes to provide the necessary NGN QoS services. For this reason measurements and statistical analysis of communication processes are required at 10...100 µs time scale to cover the Corvil bandwidth based entropy characteristics and its macro effects during the session time of real-time applications.

## References

[1] Patrice Abry,
Lois D'échelle, Multirésolutions et Ondelettes,
Habilitation Travaux de Recherche,
Université Claude Bernard Lyon, Mars 2001.

[2] Corvil Ltd, Whitepaper (2004):
An Introduction to Corvil Bandwidth Technology.

[3] Corvil Ltd, Whitepaper (2008):
Managing Performance in Financial Trading Networks.

[4] T.D. Dang, B. Sonkoly, S. Molnár,
Fractal Analysis and Modelling of VoIP Traffic,
NETWORKS 2004 – Conf. Proc.,
Vienna, Austria, June 13-16, 2004.

[5] Z. Gal, Gy. Terdik, E. Iglói,
Multifractal Study of Internet Traffic,
2000 WSES International Conference on Applied and Theoretical Mathematics, Vravrona, Greece,
December 1-3, 2000. (http:/www.worldses.org)

[6] Gál Zoltán, Balla Tamás,
The impact of QoS on infocommunication applications,
Híradástechnika, 2007/4, pp.7–16. (in Hungarian)

[7] E. Iglói, Gy. Terdik,
Superposition of Diffusions with Linear Generator and its Multifractal Limit Process,
ESAIM: Probability and Statistics, 2003, Vol. 7, pp.23–88.

[8] Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V.,
On the self-similar nature of Ethernet traffic (ext. ver.),
IEEE/ACM Transactions on Networking (TON),
Vol. 2, Issue 1, February 1994, ISSN:1063-6692

[9] Park, K., Willinger, W. Eds.,
Self-Similar Network Traffic and Performance Evaluation.
Wiley-Interscience, New York, 2000.
ISBN: 978-0-471-31974-0

# Personal Electronic Nurse – medical monitoring system

ANDRÁS TÓTH*, LÓRÁNT VAJDA**, FERENC VAJDA*

*Budapest University of Technology and Economics,
Department of Control Engineering and Information Technology
{totha, vajda}@iit.bme.hu

**Bay Zoltán Foundation for Applied Research, Institute for Applied Telecommunication Technologies
vajda@ikti.hu

*Keywords: telecare, patient monitoring, sensor networks, wireless communications*

**A common problem in Telecare systems is that the physician and nurses are unable to be with a patient with sufficient frequency. This paper introduces a wireless telecare system for patients in and out of the hospital. The primary goal of this study is to establish real time monitoring of the patient's vital signs in a system called the Personal Electronic Nurse (PEN). The biomedical data for each patient, such as, ECG, blood pressure and body temperature are pre-processed on the PEN Mobile Device and transmitted to a Data Collection System using wireless communication and embedded technologies. The system is able to monitor patients' vital data and to display their personal information in a continuous fashion. The main features of the system have been implemented and tested. The results are promising and show the system can the various modules to collect sensor data and send it to the collector system successfully.**

## 1. Introduction

In the past, the medical care monitoring for a patient was managed only by measuring of vital signs manually, documenting the measured values on paper, and communicating over phone lines or handheld devices. Nowadays, sensor network solutions are becoming an important part of medical administration systems. Important features of these systems are low energy consumption, small size and reliability. Patients that use such solutions are accepting the monitoring systems as long as they do not feel that they are watched all the time by an intelligence system ("Big Brother" effect). Therefore, it is not enough to create small and low weight devices, but it is getting important to use wireless technologies.

We have developed a system called Personal Electronic Nurse (PEN) that facilitates collaborative and time-critical patient care in the telecare community. First, automated vital data collection is accomplished and can be processed locally and accessed remotely by the care personnel (nurse, doctor, etc.). One of the most important features of the system is its scalability so it can serve from one patient to hundreds and could expand from one room to a national service.

The modularity of the system is highly defined in terms of software and hardware. At the design phase we decided to use currently available, stable and trusted platforms and technologies. This enabled us to create a reliable and cost effective system which is very compatible and easily made in comparison with other similar systems in the market.

The main contributions of this paper are the following:

- A review of similar systems is presented with a list of pros and cons.
- Based on the conclusions a system plan and realization method is given.
- Finally, the functioning system and future plans are presented.

## 2. Overview of wireless telecare solutions

A rapidly increasing number of healthcare professionals believe that wireless technology will provide improved data accuracy, reduce errors, costs and improve the overall patient care service quality. In our vision the problems raised in such systems can be solved on three levels:

(i) Collect and preprocess the sensor data locally by defining intelligent data filters and saving data streams on local storage;
(ii) Prepare a dynamic data collector system;
(iii) Develop of easy-to-use display devices.

A good example of a telecare system developed especially for out-of-hospital situations is the MobiHealth [1] project, which uses 3G technology for patient monitoring. The system is able to analyze all the collected vital data continuously, getting the disease characteristics and analyzing the causes. MobiHealth can be successfully used for large distances, however, it is too complex and expensive. On the other hand, the system uses continuous streaming of all of the patients vital data on the wireless link, which is not desirable at all in such systems, since this solution leads to a large overhead.

An other solution dealing with sensor systems for telecare solutions was developed by the Shimmer mote community [9]. This mobile mote uses two wireless channels for communication, Bluetooth and Zigbee, which

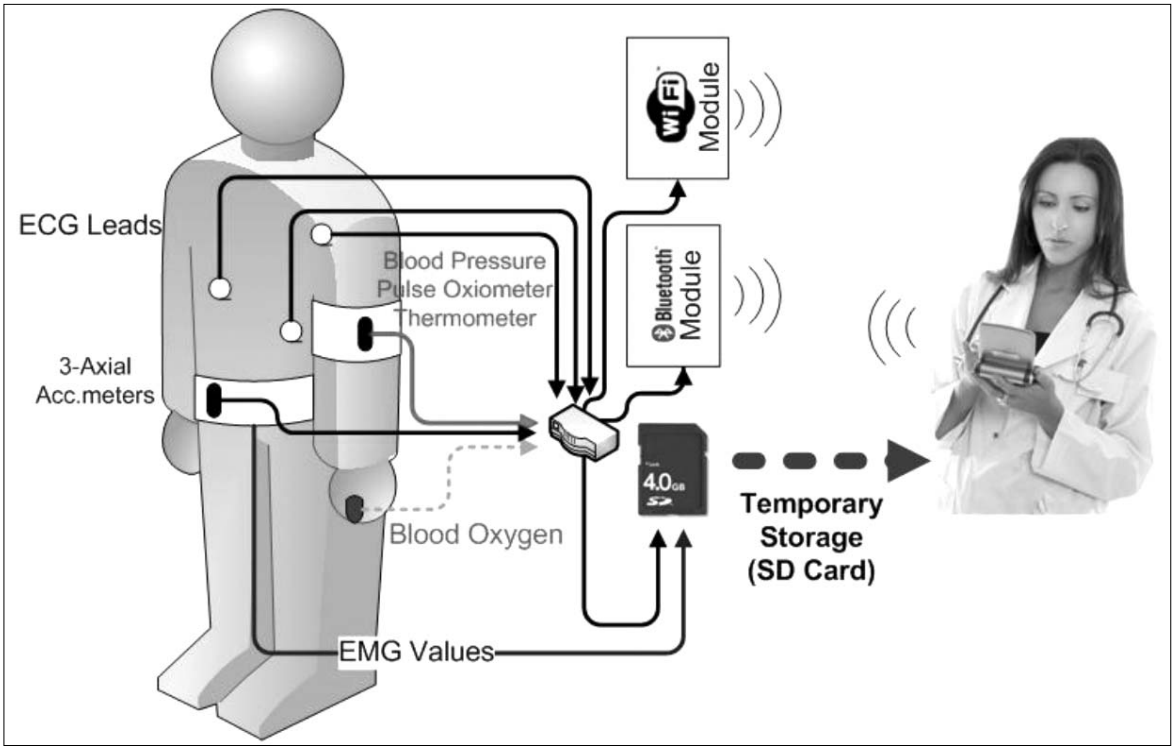



*Figure 1. Data storage solution*

are available in the mobile device. Having two wireless radios switched on all the time on increases energy consumption. Another drawback of this system is that from the development point of view Shimmer mote is very expensive.

There are several solutions available that can solve data processing and data display problems. One of the satisfying approaches is the Code Blue project's [7] platform independent VitalDust application. Unfortunately, it has a proprietary protocol, which cannot be integrated with other third party systems, and is undesirable from the present medical care point-of-view.

## 3. System design and implementation

Using existing standards and recommendations in an overall system makes it easy expandable and compatible with other systems and helps solving the problem of scalability.

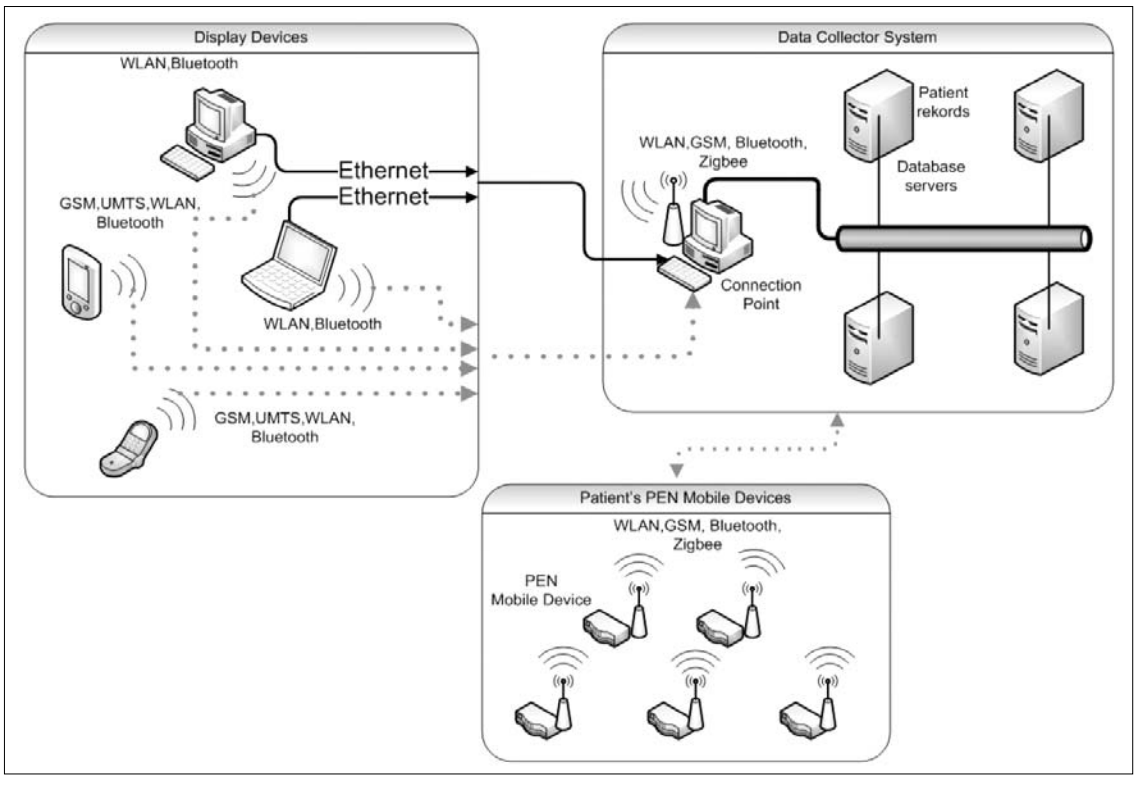



*Figure 2. Planned system modules*

The system has three main components *(Figure 2)*.

(1) The *PEN component*,

which contains mobile devices. These devices can use several wireless communication channels depending on the user demands. Our main goal is to develop a mobile element, which can measure and deliver the patients vital data. To accomplish this, an effective data collecting and processing network is needed too, which can securely gather and display all of the patient' data. Measured data in the PEN component is pre-processed based on pre-defined filters and rules. If the available bandwidth of the communication channel cannot deliver the continuous streaming, temporary data storage must be included in the PEN. It is also important that the data be represented in a standardized format. An internal storage solution is also helpful to move the collected vital data, for example, to a PDA used by a doctor or nurse by simply changing the storage memory *(Figure 1)*. In this way, only high priority data has to be sent through the (wireless) communication channel. A remote configuration and management scheme has to be done in the PEN, which allows for a remote update of the PEN units software, driver upload and management.

(2) *Data Collector System* –

Used for data storage of patient records. There is also a higher intelligence requirement for data processing and representation than in the PEN unit.

(3) *Display Devices* –

These can be a PDA, PC or other mobile phone device which is capable to run a lightweight application software for visualization of the measured and processed data.

### 3.1 Wireless technologies

As stated, currently available solutions involve only fixed communication technologies in the system. We have developed a solution which can handle various types of communication connections (Bluetooth, Zigbee, GSM, WLAN) switching transparently between them as needed.

*Figure 3* represents a multi-communication setup. In this way, if a user uses the PEN in a Bluetooth area, then Bluetooth technology is used for communication. Moving the user in a WLAN environment, the communication is changed transparently to WLAN. If a new communication module is purchased, that can also be used with this system by changing the communication module (of course, providing a driver for it is necessary). Therefore, the communication module can be changed on the fly and the needed driver parameters are stored on the unit itself or can be downloaded from the storage memory.

### 3.2 Modularity and open devices

Developing any new device always raises the question of how to reuse the existing modules which would otherwise have to be redesigned from the ground up. In our vision, at the design phase, we use as much as we could from existing industrial standard modules. In this way, stability is increased, and less expensive system can be produced.

## 4. Processing system plans

As mentioned before, the essence of PEN is to incorporate existing standards and solutions. To do this it was

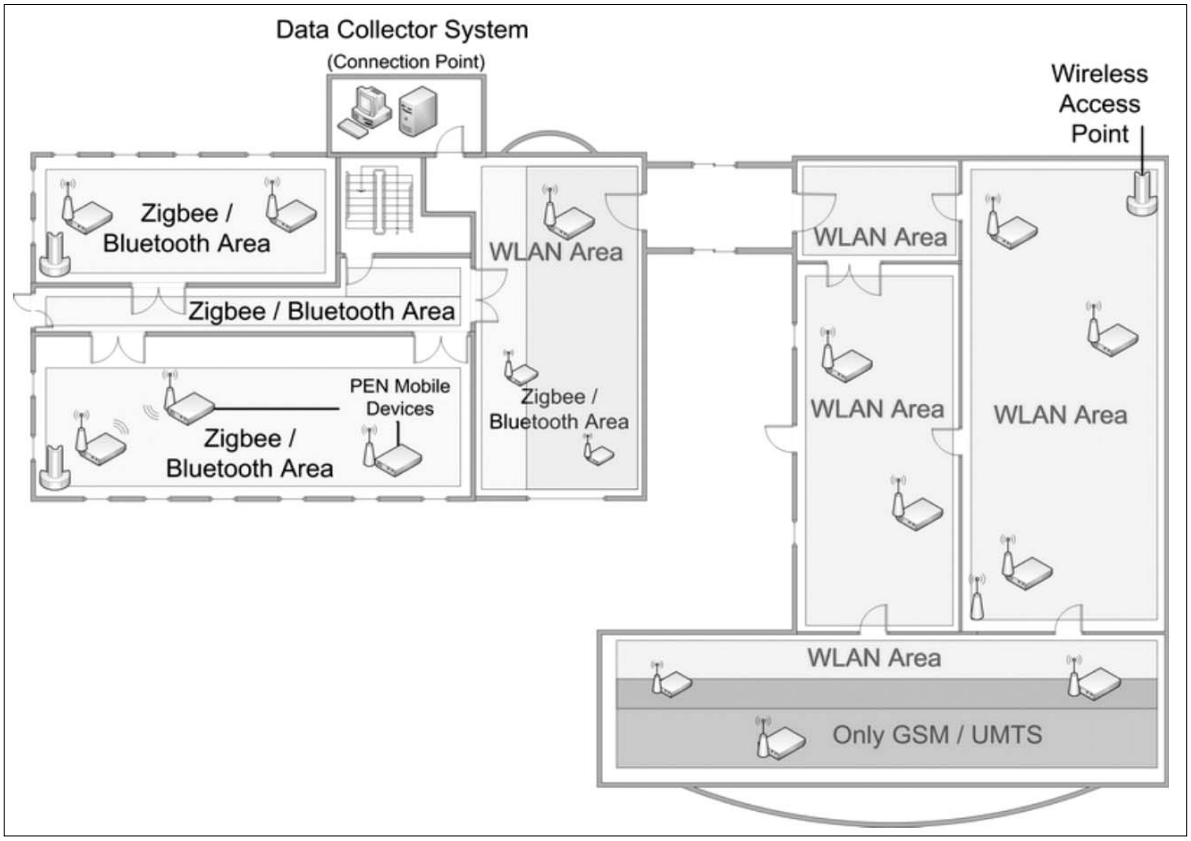



*Figure 3. Multi-communication setup*

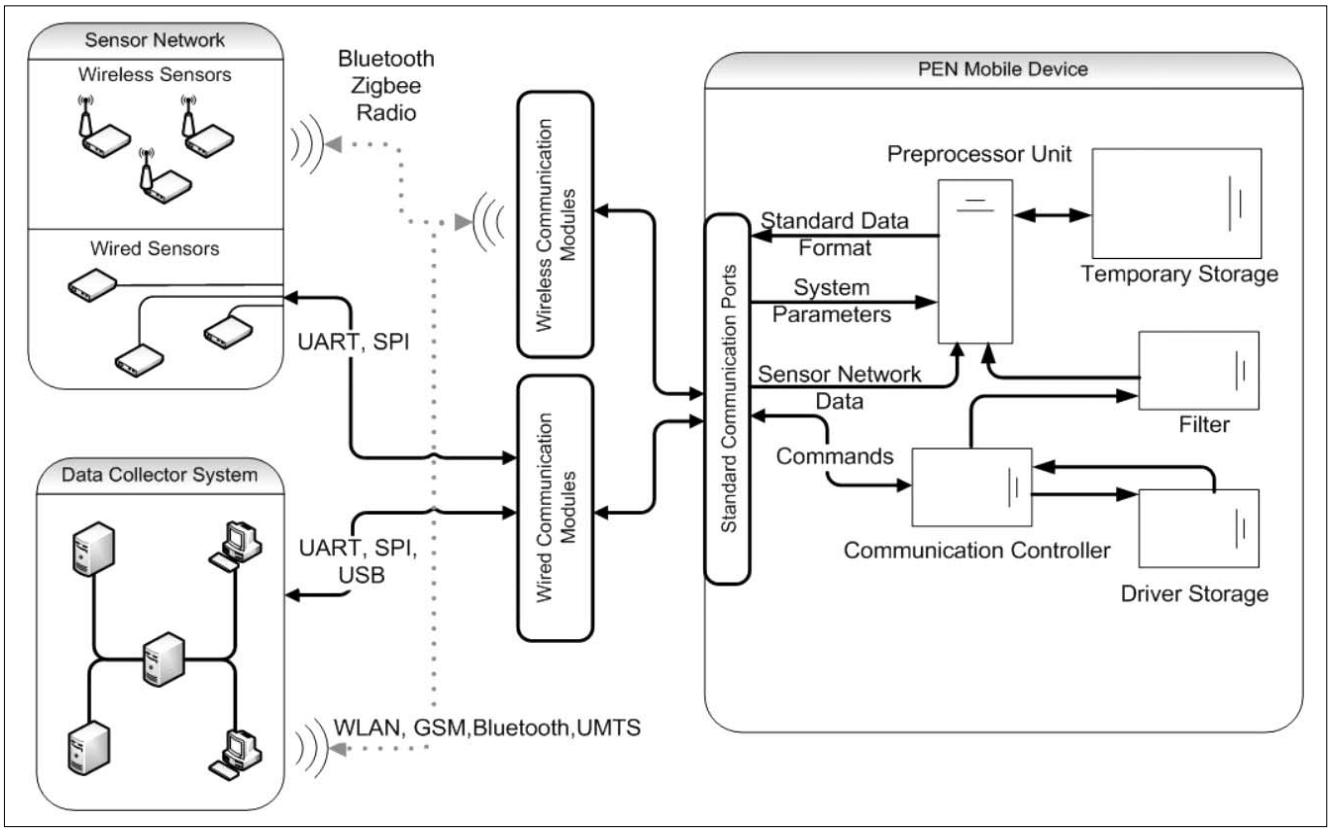



*Figure 4. Modular architecture of the system*

necessary to create a complex driver architecture for the communication and sensor modules. The main problem we solved was how to find the right driver system for the connected elements, and make the system ready to use it.

#### 4.1 PEN Mobile device

During system design we understood that the resources in the mobile device were limited. Therefore creating complex services on the PEN itself is not possible. Moreover, the energy consumption of these nodes has to be kept on a low level. In a mobile device the wireless communication modules are the main energy consumers. The less these modules are used, the lower the power consumption. In this way, within the PEN, the measured data has to be preprocessed (by filtering methods), compressed, stored on an internal storage and sent only at communication times of the wireless modules. Data compression is also useful to avoid communication overhead.

The above mentioned filtering methods can be divided into two main parts:

(1) Development of a uniform standardized sensor data format, which allows the system to understand and analyze the measured information without reference to the sender sensor type.

(2) Design of algorithms which can fit the measured parameter of a user into a general User Profile. Therefore, the application has to learn about the user specific parameters and edit the measured values in the mobile device. If a measurement shows an emergency scenario, then the communication channels are immediately used to send out alerts. Additional measurements are stored on the internal storage of PEN and sent out only if needed by the Data Collection System.

Depending on the wireless communication technology used, the system can perform stable, multi-hop communication [2] and if needed, extends the range of the radio communication and decreases energy consumption.

The modular architecture of the PEN mobile device can be seen in *Figure 4*. Different internal units of the PEN can be seen as well with the internal interfaces between the units represented. The possible connection types between sensor networks, PEN and Data collector modules are shown as well.

#### 4.2 Data Collector System

The main parts of this module are the *data collector, processor* and *storage* units. It is important to have all components operate in a fast but safe mode. At this time, several groups have developed solutions for such purposes. Unfortunately, these solutions are too specific and unique to include them in our system.

To have a scalable and compatible system, we have decided to use the recommendations of the NESSI [3] community. NESSI is founded to specify recommendations for standardized software and service interfaces. Moreover, the system is designed to handle a multi-connection architecture.

#### 4.3 Display Devices

In such a system, a module is needed, which allow users to login into the system safely, check the measur-

ed vital data, check the processed and analyzed results and perform modifications on the system, if necessary. Of course authentication mechanisms and user rights have to be set carefully. For this component, NESSI recommendations are also used to access the database and data handlers.

## 5. System Realization

In this project, a model-level plan of the PEN system is completed.

Based on the results, a prototype of the PEN Mobile Device has been created and tested. The main part of the PEN is the highly spread Atmel microcontroller, the ATMega128, used in many embedded systems. The Atmel controller is the heart of the mobile device, having high performance, low power consumption and a low costs. Moreover, this platform has several available software solutions to manage file systems on external storage or control the standard communication channels simultaneously.

The software components for ATMega128 are written in programming language ANSI C. The controller has an 8 channel 10-bit Analog to Digital converter for measuring the signs of analog medical sensors. Moreover, it has various standard communication channels such as two TTL serial ports, which can use UART and SPI communication standards. The ATMega128 can raise his calculation abilities up to 16 MIPS. Important point at the system development phase was that the ATMega128 had a free downloadable development platform, called AVRStudio [4].

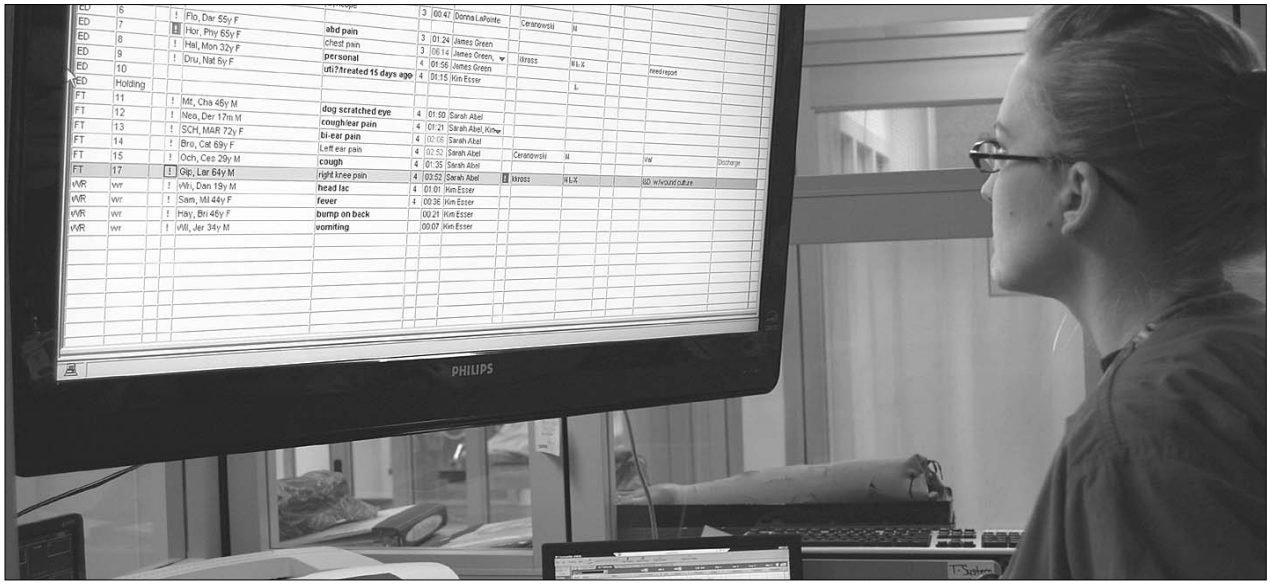
For the wireless communication channels in the prototype we have used Bluetooth (Rainsun BT20) and Zigbee (Telegesis ETRX2) technologies. The module of Rainsun [5] is a high quality Bluetooth solution with low power consumption in idle mode (about 20uA). It has also 8 input-output channels, which can be programmed by the user. An 8 MB flash memory can be found to store the Bluetooth stack and the user applications. This module supports the full Bluetooth protocol Stack up to HCI, with full speed and full piconet support. It makes driver system and easier to control the unit. UART communication port is also available with programmable baud rate up to 1.5 M to connect to the main controller.

The Zigbee Telegesis [6] module is also a good solution implement the Zigbee communication channel. It uses a very reliable core, which is based on Ember EM 250 Zigbee chip, having AT-style command set. The size of the chip is small and easy to integrate in the PEN mobile device. A 128 kB flash memory is integrated containing the full Zigbee stack support and the user's firmware or application.

It can act as an End device, Router or Coordinator and it makes easy to build complex ad-hoc network from the PEN mobile devices. In this module, the security is also guaranteed by using the AES-128 hardware supported encryption. These modules communicate with the Atmel controller on UART. Drivers and parameters to handle the wireless modules are stored in a temporary storage space on the internal storage memory.

For internal storage memory the system is using a Secure Digital memory card handled by the controller on a standard SPI channel. This memory is used for the measurement data storage as well as for driver space too. This module is made by Panasonic. The main controller can communicate with the storage card through SPI communication channel. With this connection mode the communication data rate can be up to 2 Mbps.

The device transfers the collected and filtered data to the database server on the computer. In this database, we can assign one record to each patient. These records can be assigned to doctors or nurse. To visualize collected and processed values a Java based patient record indicator is used which is developed at the Bay Zoltán Foundation for Applied Research. To store and forward data, the system uses an XML based scheme, which is built on the basic structure of with the Sensor ML [8] technology.



## 6. Conclusions and future work

During our work a prototype of a PEN Mobile Device was developed and implemented. Several standards and requirements are designed to be compatible with other system. We have developed also modules and interfaces based on industry accepted and highly standardized technologies. Remote firmware upgrades and remote device management are accomplished too. Therefore, the final developed system has a highly dynamic and scalable modular design with low power consumption and development costs.

Several performance and capacity tests of the system were done. At this time, we collaborate with a medical university and are designing pre-processing and post-processing algorithms for PEN devices and Data Collector Module for ECG and EMG sensors. The full technical performance testing results of the PEN will be published in an internal Technical Report and will be made available after validation.

As future plans, new sensor attachment to the system and testing are included. Also, GSM and UMTS communication channels are to be developed. If all the tests are successful, a network of a higher number of devices is to be tested in a real-life scenario with ECG and EMG devices included.

In the AAL (Ambient Assisted Living) Laboratory of Zoltán Bay Foundation for Applied Research, where the system was developed, our goal is that the PEN system will become an industry standard for safe, reliable and high speed patient monitoring and care.

### Authors

**ANDRÁS TÓTH** received his M.Sc. in 2008 at Budapest University of Technology and Economics, Hungary and doing his PhD studies at Budapest University of Technology and Economics, Department of Control Engineering and Information Technology. He is currently working as a researcher in Bay Zoltán Foundation for Applied Research on projects connected to the sensor network and sensor data fusion in medical systems topics.

**LÓRÁNT VAJDA** received his M.Sc. in 2000 at Technical University of Timisoara, Romania and did his PhD studies at Budapest University of Technology and Economics. He is currently working as a researcher in Bay Zoltán Foundation for Applied Research on international and national projects connected to the AAL topic. He is also working with Wireless Personal Area Network and sensor networking technologies. He has authored/coauthored several refereed international papers, journals and conference contributions.

**FERENC VAJDA** received his Ph.D. in Electrical Engineering in 2006 from Budapest University of Technology and Economics, and is currently working as an assistant professor at the department of Control Engineering and Information Technology. His main research activities focus on processing 2D/3D images and on various areas of health information technologies.

### References

[1] The MobiHealth Project.
Innovative gprs/umts mobile services
for applications in healthcare.
http://www.mobihealth.org/

[2] Alec Woo, Terence Tong, and David Culler,
"Taming the underlying challenges of
reliable multihop routing in sensor networks.
In the 1st ACM Conference on Embedded Networked
Sensor Systems", SenSys 2003,
November 2003.

[3] Networked European Software & Services Initiative.
http://www.nessi-europe.com/Nessi/

[4] Atmel AVRStudio.
http://www.atmel.com/dyn/products/
tools_card.asp?tool_id=2725

[5] Rainsun Group.
http://www.rainsun.com/

[6] Telegesis manufacture.
http://www.telegesis.com/

[7] David M., Thaddeus F-J.,Matt W. and Steve M.,
"CodeBlue: An Ad Hoc Sensor Network Infrastructure
for Emergency Medical Care",
MobiSys 2004 Workshop on Applications of
Mobile Embedded Systems (WAMES 2004),
June 2004.

[8] Open Geospatial Consortium Inc.,
"Sensor Model Language (SensorML)
for In-situ and Remote Sensors", 2004.

[9] SHIMMER research:
http://shimmer-research.com

# Utilization of UML diagrams in designing an events extraction system

MIHAI AVORNICULUI

*Babes-Bolyai University, Department of Computer Science, Cluj-Napoca, Romania*
*mavornicului@yahoo.com*

*Keywords: event extraction, RUP method, UML diagram*

**The system proposed and investigated in this paper is intended to be a helpful instrument when looking for pieces of information on different web pages, in cases when a special type of event is needed to be centralized. The system can be useful, for example, when information about weather forecast for a specific geographical region is needed to be collected from different web pages (Yahoo, Google, CNN site, etc.). To develop such a system we will use the UML diagrams.**

## 1. Introduction

Nowadays the amount of information on the Internet reaches high proportion. In finding specific information on the Internet some general instruments (engines of search) have become popular, which automatically run through all the existent web pages in order to update the databases containing the latest information on the Internet. In most cases the search is done based on a number of strings stored in the database of the search engine. The result of such a search is, generally, a large amount of links to different web pages.

To systematize the searching process and to obtain a result in a concrete form, an other stage is useful, a stage in which the information returned by the search engine is processed, and the response is generated in a more organized form [3,5,6].

The centralization of a specific type of event is useful, first of all, to realize some news services. These services must offer updated information about a specific type of event, and – if possible – in real time.

Take sport events for example. A user of this type of news service might want to find out what sport events take place in a certain geographical region (a city, a country, etc.) during a certain period of time (in that very moment, a day before, or the next week, etc.). All this information has to be obtained from centralized information. In this way, useful data can be obtained about a specific event from various sources. These sources (the websites from where the information was taken) can complete each other concerning the information content about a specific event.

The system will recognize the events of a specific type (weather, sport, politics, and text data mining) depending on the way it will be drifted (the dictionary that it possesses). These events can be transmitted to the user or the entire context in which the event appeared, and can be taken, to show the initial form in which the event was included.

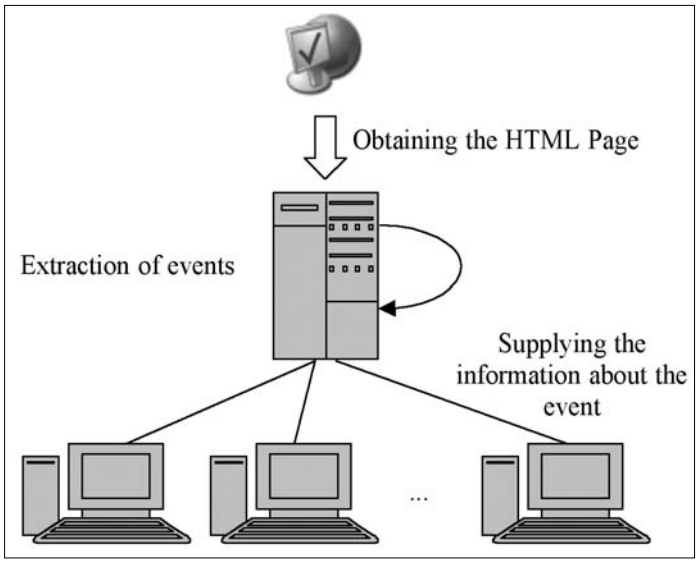
## 2. The conceptual model

To be able to realize what we intended, the system (the components) has to handle, in general, three aspects [1]:

• Taking the HTML documents from the Internet and saving them in a database in order to process them later: in order to realize this aspect, the system must gather and run through a number of webpage addresses. This list of addresses can depend on type of the event we want to identify. The application will recursively run through the list of addresses and will save all the documents it meets in the local database.

• Document processing and obtaining the information needed: the processing of documents and the extraction of the information needed is based on a dictionary of concepts which describes the types of events. This dictionary of concepts has to be flexible in order to be able to identify different variations of types of events. All the identified events will be stored in a database in order to be able to refer to them later.

*Figure 1. The structure of the system*



• Giving the users a way of access to the collected information: finally, an access to the extracted information has to be given to the user. For example, in case of sport events, a list of arranged sport events can be presented to the users, according to the date when it will take place or when it took place. The search on different criteria will be allowed. This offers a quick access to the information requested, eliminating the necessity of day-by-day search.

The *Figure 1* relevantly presents the structure of the system.

In case of human users, HTML documents will be offered with the needed information, and if the user is a computer programme, then the information will be transmitted using a generic form, for example an XML document, in order to be able to process it easily.

## 3. The application of the RUP method

RUP (Rational Unified Process) is an iterative method. Each iteration has one or more aims. In RUP, the aim is to produce a functional software which can add value and deliver it to the customer. The iterations are determined by a time limit. This means that each facility must be realised in a certain period of time [8,10,11].

According to Kruchten, RUP has the following characteristics [4]:

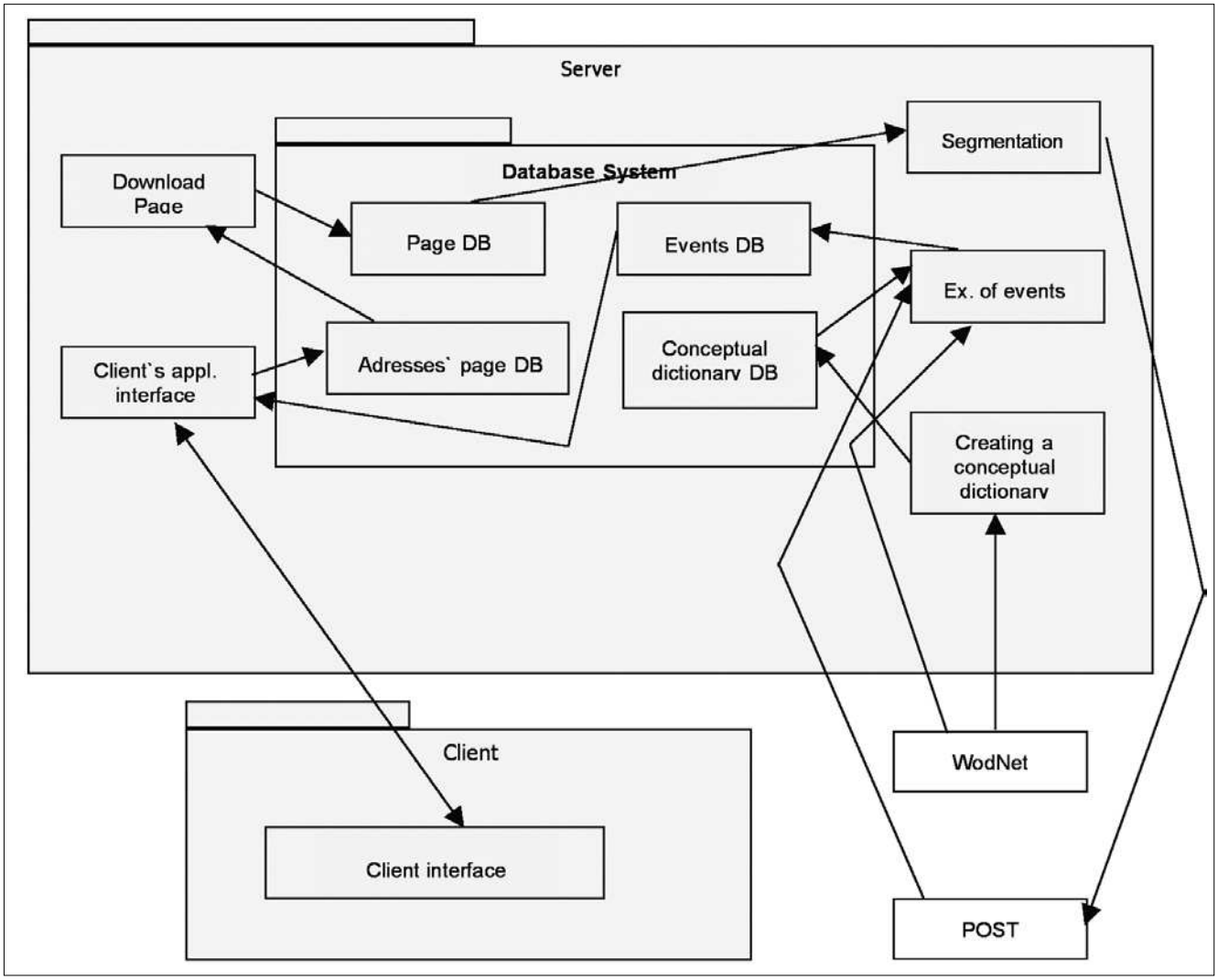
• *Interactive development of a software product* – proposes developing in short increments of certain iteration chains. This ensures a real-time detection of risks and an adequate addressing of them.

• *Processing demands* – denotes a continuous process of demand identification of a system that evolves in time and the demand factors that have the greatest impact on the system. Processing these demands requires a disciplined manner of evaluating, associating priorities and monitoring. It is better that communication had a well defined set of demands as a base.

• *Uses architecture based on components* – it is more flexible, making the extensibility of the particular application possible. Components can be redesigned or extended without compromising on the evolution of the whole system.

• *Uses visual instruments of modelling contributing to the understanding of extremely complicated systems*

*Figure 2. The diagram of the modules*



| Module | Function |
|---|---|
| Addresses' page Database | Contains the list of web pages for a certain domain, on which the search will take place. |
| Events Database | Contains the identified events, indexed according to their domain, time, locations and criteria. |
| Conceptual dictionary Database | Contains the concepts which are going to be looked for, indexed according to domains. |
| Download Page | Harvests periodically web pages from the Internet with addresses in the page addresses Database, and stocks them in Pages Database. This module can recursively fetch the pages indicated by the links contained by the current page, from the same server, to a certain depth of harvesting. |
| Segmentation | Divides the text from the web pages of the pages database in segments. |
| POST | This module has the role to process the segments and to annotate them with the adequate parts of speech from the segment. |
| Extraction of events | Extracts the events from the annotated segments from the Post module, on account of the concepts from the conceptual dictionary and from the WorldNet, depositing them into the events database. |
| Creating the conceptual dictionary | Constructs the specific concepts of a certain domain through a learning algorithm, and after that deposits them into the Conceptual database. |
| WordNet | Supplies relations between words. |
| Client's application interface | Gets requests of looking for events from the Events database. |
| Client's interface | Assures the interface with the client, the introduction of events' queries, options set ups, displaying the results according to several criteria. |

*Table 1. The functions of the modules*

– using UML models the complexity of a system can be effectively processed among more developers.

• *Permanently verifies the quality of the produced software* – this represents a constant occupation that runs at the level of every iteration. From this perspective errors are discovered in time and revising costs are reduced.

• *Controls changes brought by developed software* – dealing with these changes represents a key to the success of developing an IT system. If one of the team members causes a change to the system, every member has to be warned, who is affected by that change.

Analysing the characteristics of the RUP method, we can conclude the following regarding the advantages of the method when it is used to develop systems of event extractions [1]:

• RUP is a method that enables developing systems with a flexible and extendible architecture. Event extraction systems are these kinds of systems so they comply with these demands.

• RUP focuses on dealing with aspects of potential risk in time. This characteristic is a plus for every system.

• It doesn't imply a fixed set of tasks in the initial stage so they can be refined as the project evolves. From the point of view of an event extraction system tasks can not be specified in the first stages so this characteristic is in favour of event extraction systems.

• RUP stresses on the final product and on the conformity of this with the demands of the final users. This is clearly an advantage of event extraction systems.

• RUP leaves the evolution of the system entirely on the users will. From the point of view of developing an event extraction system, this characteristic can turn into an important disadvantage. It is possible that the user does not have any knowledge of EDI or XML message formats so he could jeopardize the flexibility of the system.

• RUP takes over the advantages offered by UML. For a lot of IT systems this represents an advantage.

• RUP makes possible to control the quality of the developed system. The quality of the system is in close relationship with its reliability. The better the quality the more reliable the system is. This characteristic brings an advantage to developing an event extraction system.

• The time needed to develop a system using RUP is much less than in case of other methods, so this is also considered an advantage.

• When it is adopted RUP becomes a repetitive and predictable process for the developing team. This leads to a high efficiency in case of developing large and reliable software.

We consider that RUP represents a method that can be successfully used for developing event extraction systems while avoiding certain disadvantages of the method. Among these is the fact that RUP does not contribute in an explicit way to the development of some implementing instructions, for it is a perspective method in comparison with the more agile XP.

## 4. The modules of the system

Just like in the case of other systems, this system is also composed of modules. Now we will identify the system's modules *(Figure 2)*.

In *Table 1* the functions of each module is presented.

## 5. Detailed design of the system

The system will naturally contain several classes. The classes are not isolated elements of the system. Consequently, we will have to identify the relations between classes.

In the followings we will present the attributes and methods of the classes.

*The Addresses' page DB class:*

- Is an XML database.
- Contains the list of starting pages' addresses.
- In this database there are only the starting pages, but their harvesting module will search and fetch the pages that were referred to in the starting pages, from the same server, for a certain depth of searching

Example:

```
<xml>
    <domain name="weather">
        <url name="www.yahoo.com/weather" depth=3>
        <url name="www.cnn.com/weather" depth=2>
        .....
    </domain>
</xml>
```

*The Pages Database:*

- Is an XML database.
- Contains in XML format the indexes of brought pages, and, in separate files, the pages.
- Only the pages containing text are downloaded and kept (extract pieces of information).
- For each page there is an index in the page' s database which contains information about the URL of the page, the last update time, the name of the file where it is stored, a unique identifier for each page and the list of the page's identifiers to which the page is related to.

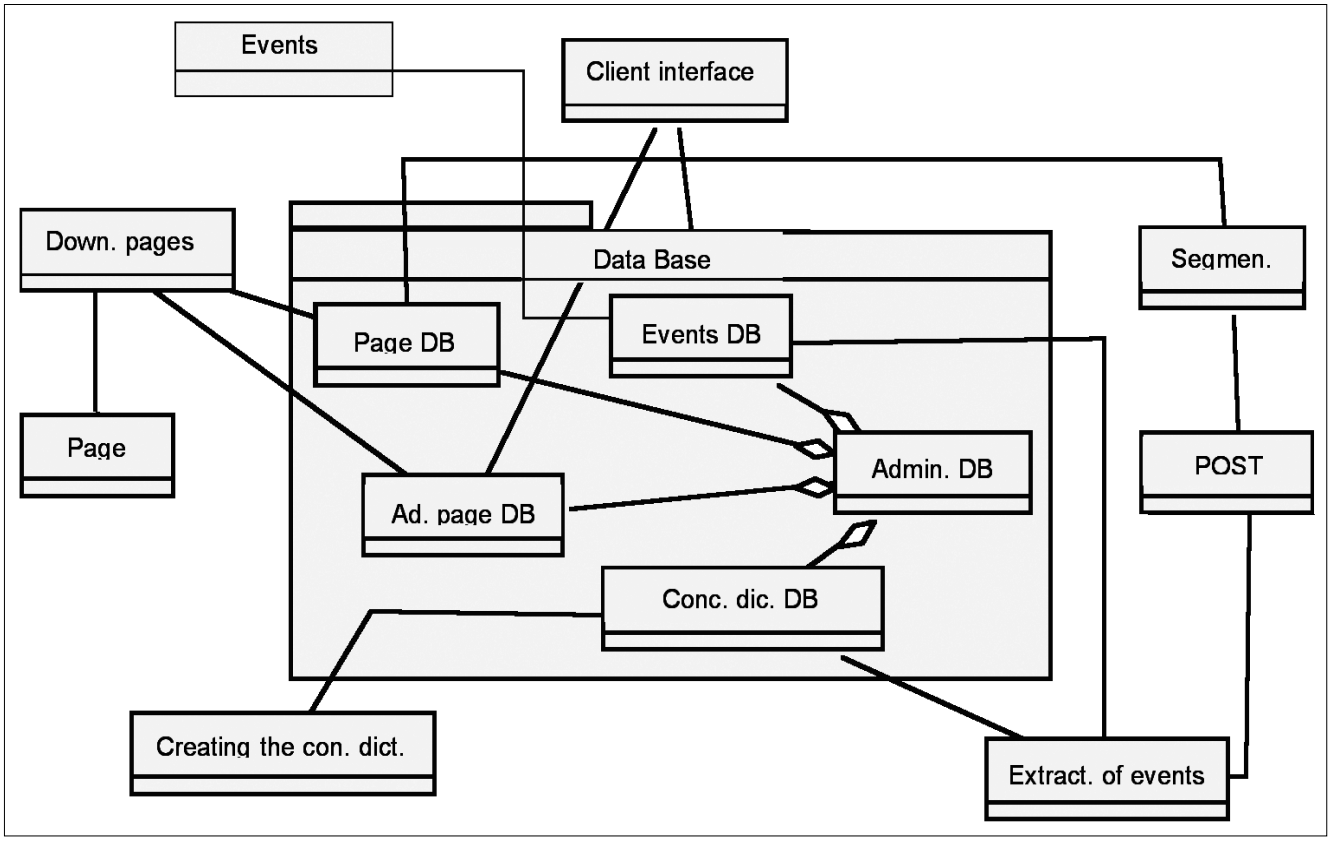
Example:

```
<xml>
    <domain name="weather">
        <URL page="www.yahoo.com/weather/"
        lastUpdated="22.10.2008"
        fileName="weather_com_yahoo_www" id=1>
        </page>
        <URL page="www.yahoo.com/weather/europe"
        lastUpdated="22.10.2008"
        fileName=
        "europe_weather_com_yahoo_www" id=2>
        <upPage>1</upPage>
        </page>
    </domain>
<xml>
```

*The Events database class:*

- Is an XML database.
- Keeps the identified events, indexed by domain, time, location and other criteria.
- An event contains the name of the event and a number of attributes specific to the event, ex. {**rain**, Cluj-Napoca, today}.

*Figure 3. Relations between classes*



*The Conceptual dictionary database class:*
- Is an XML database.
- Contains the concepts that are going to be looked for, indexed by domain.
- Concept – the collection of all information (events) of a certain type, which respects the set of syntaxes imposed by the concept ex: {**rain**, Madrid, today}, {**rain**, Bucharest, tomorrow}, {**rain**, Sofia, yesterday}, {**rain**, X, Y}, for each valid X and Y, plus a series of valid syntaxes for the rain concept and attributes (complements) of X and Y.
- For one concept the following information must be kept:
  – the name of the concept- releaser of the concept
  – type of concept
  – the list of attributes
  – the syntax of the concept – the relative position of the attributes to the name of the concept

*The Page download class:*
- Fetches periodically pages from the Internet from the addresses contained by the addresses' page database. This module can recursively fetch the pages indicated in the current page, from the same server, to a certain depth of harvesting.
- This module uses an external application, specialised in fetching recursively web pages, but it can apply the function internally.
- Has page analysing function to extract new links which are going to be used, plus an elimination function, which erases information (ex: scripts) which are not relevant for the application.

*The Segmentation class:*
- Divides the text of the web pages from their database into different segments.
- A segment is a part of a text, which can be a sentence or a complex sentence and which can be seen as an entity of atomic information.
- The post applications accept this kind of segments as input data.
- The segmentation is done on account of the HTML tags which help to delimit them (ex: <P> <BR> etc.) but also on the account of the text delimiters.

*The POST class:*
- This module has the role to process the segments and to annotate them with the adequate part of speech from the segment.
- The module interfaces the programme with a Part of Speech Tagger application, which receives a segment and annotates it.

*Event Class:*
- It contains the most important attributes of events.
- Interacts with the Events database.

*The Extraction of events class:*
- Extracts the events from the annotated segments from the Post modules, on account of the concepts from the conceptual dictionary and from the WordNet, depositing them into the events database.
- This module is one of the most complex modules of the system, together with the Creating the conceptual dictionary module and the segmentation module.
- A concept is formed of a trigger and a series of attributes, which can be located relatively to the trigger in a certain schema (which implies a certain syntax).
- Uses a pattern adjusting algorithm to identify the possible attributes, which are checked later using the pieces of information from the WordNet.

*Admin DB class:*
- It is an XML data base
- It holds necessary information to the data bases: Events, Page, Ad. Page and Conc. Dic.

*Creating the conceptual dictionary class:*
- Constructs the specific concepts of a certain domain through a learning algorithm, and then deposits them into the Conceptual database.
- The construction of the conceptual dictionary can be done either by identifying manually the representative concepts for a certain domain, which are provided to the module to enter them into the database; or by using the training and learning algorithm of certain concepts on special learning pages.
- The model of the learning algorithm leads to the extraction of concepts which match well and which can well identify the events from similar pages to those from which the learning has been done.
- The construction of dictionary is one of the most important components in this stage in extracting the events.

*Client's application interface class:*
- Gets the requests for looking for events, which are provided by the Events database.
- The events are only taken out from the database, their search and identification being made separately by the Events Extraction module.

In order to implement the system we can use Java Server Pages (JSP). JSP is the most popular method to create Web interfaces for the applications which are Java based.

## 6. Conclusions

Using UML in the development of event extraction systems is opportune from several points of view. UML offers powerful tools of modelling behaviours aspects. Classes contain both data and their associated processes. It also offers a complete vision above the groupings of different components under the form of packets and their physical places. Event extraction systems are complex systems that present a more dynamic evolution than other types of IT systems.

For this reason it is necessary to use a flexible instrument of analysis and design that permits the future expansion of the system.

### Author

**AVORNICULUI MIHAI-CONSTANTIN** graduated from the Babes–Bolyai University of Sciences and obtained his M.Sc. degree in Databases and Electronic Commerce in 2005. He obtained a five-months SOCRATES/ERASMUS scholarship in 2004 which he spent at Johannes Kepler University in Linz. Since 2005 he has been responsible for the specialization of informatics in economics at Babes–Bolyai University of Sciences. He has been author or co-author of several lecture notes, textbooks and monographs during 2002-2007. Main research areas include databases, object oriented programming and modeling. Mr. Avornicului is currently is working toward his Ph.D. degree.

### References

[1] Avornicului, C., Avornicului, M., The Use Of Objective Methods for Developing Events Extraction Systems, Annals of the Tiberiu Popovici Seminar, Cluj-Napoca, October 10-12, 2008., pp.11–22.

[2] Avornicului M., Planning and management of information systems, ÁBEL Publishers, Cluj-Napoca, 2007 (in Hungarian).

[3] Han, J. and Kamber, M., Data Mining: Second Edition Concepts and Techniques. Morgan Kaufman Publishers, 2006.

[4] Kruchten, P.B., The Rational Unified Process: An Introduction – IEEE Software, 1998.

[5] Lin, B., Web Data Mining – Exploring Hyperlinks, Contents an Usage Data, Springer, 2007.

[6] Lin, T.Y., Xie, Y., Wasilewska, A., Lian, C.J., Data mining: Foundations and Practice, Springer Berlin, 2008.

[7] Markov, Z., Larose, D.T., Data Mining the Web, John Wiley & Sons, 2007.

[8] Raffai M., The UML 2 modeling language, Palatia Printers and Publishers, 2005 (in Hungarian).

[9] Sieg, A., Mobasher, B., Burke, R., Ontological User Profiles for Personalized Web Search. Proceedings of AAAI Workshop on Intelligent Techniques for Web Personalization, AAAI Press Technical Report WS-07-08, July 2007., pp.84–91.

[10] Sommerville, I., Software Engineering, 7th Edition, Addison-Wesley, 2004.

[11] http://rup.hops-fp6.org/process/ovu_proc.htm

# The 60 years' anniversary of HTE

GYULA SALLAI
*President of HTE*
ROLLAND VIDA
*Chair of International Affairs Committee of HTE*
CSABA A. SZABÓ
*Editor-in-Chief, Infocommunications Journal*

**2009 is a special year not only for the IEEE, which celebrates its 125th anniversary, but also for HTE – the Scientific Association for Infocommunications, Hungary, which was founded 60 years ago, on January 29, 1949.**

The organization, formerly known as the Scientific Society for Telecommunications, is a voluntary and autonomous professional society of engineers and economists, researchers and businessmen, managers and educational, regulatory and other professionals concerned in the fields of telecommunications, broadcasting, electronics, information and media technologies in Hungary.

With a membership of over 1300 individuals and 100 corporate members (large companies, small and medium enterprises, research labs and educational institutions), HTE has become one of the most important scientific associations in the country, with continuously broadening international relations and activities as well. Being an IEEE Sister Society has certainly enabled us to maintain or even further enhance our reputation as an important player in the field of infocommunications, for which we are of course grateful to the IEEE.

One of the important activities of HTE is the organization of high quality international scientific conferences. After having organized the World Telecommunications Congress (WTC) in 2006, and the 16th IST Mobile and Wireless Communications Summit in 2007, last year we brought to Budapest the 13th International Telecommunications Network Strategy and Planning Symposium – Networks 2008. We are happy to say that the five-day conference was a great success, as it attracted more than 260 specialists from 30 countries, in addition to the 75 participants that registered only to the tutorial sessions that were held the first two days. The program of the symposium was divided in 21 technical sessions, which included 90 presentations altogether. In addition, there were plenary talks and panel sessions that involved prestigious speakers from regulatory bodies, industry and academia *(www.networks2008.org)*.

And the 60th anniversary of HTE cannot be truly celebrated of course without organizing a similarly prestigious conference. That is why we were happy to help the Communications Society in organizing the IEEE Wireless Communications and Networking Conference (WCNC 2009), which was held in Budapest in April 2009, after being hosted by cities like Hong Kong and Las Vegas the last couple of years. WCNC is the premier conference related to wireless communications, and with a technical program including 10 tutorials and more than 500 research papers, it was certainly one of the major scientific events in 2009.

Besides this, the entire year 2009 will be celebrated by HTE as an anniversary year, with many dedicated symposiums, workshops and publications presenting the history of the association, and the evolution of its activities. As our readers have already been informed about, in 2009 we significantly restructured the "Infocommunications Journal", the scientific journal published monthly by the HTE. Until recently, our journal was a predominantly Hungarian language journal (in fact, the only one in our field), fulfilling the important mission of disseminating information on the state-of-the-art of different areas in telecommunications and information technology among Hungarian professionals and also trying to preserve and develop the professional language in this field. We also published English issues twice a year, which were compiled mostly from the best research papers published in Hungarian during the preceding half a year period.

As of January 2009, we increased the number of English issues to four, with the objective to become a quarterly international journal. We publish original research papers in the aforementioned areas after rigorous peer reviewing process. Our newly appointed International Advisory Committee, consisting of high-standing representatives of the international professional community supports the Hungarian editorial team in maintaining the quality of published papers.

"Infocommunications Journal" is intended to become a recognized international publication forum for researchers not only from Hungary but also from neighboring countries, and in principle, from all over the world. We will be particularly happy to publish theoretical and experimentation research results achieved within the framework of European ICT projects.

As a conclusion, we hope that the 60th anniversary of our association will be celebrated through attractive, high quality events, and that in 2069 we will be able to inform you about another 60 successful years in the life of HTE.

*This communication is based on the article published in IEEE Global Communications Newsletter, March 2009.*

# IEEE WCNC 2009:
## Explores newest advances in wireless communications and cooperative systems

LAJOS HANZÓ
*lh@ecs.soton.ac.uk*

**Nearly 600 industry professionals, academics and government officials joined the IEEE Wireless Communications & Networking Conference (WCNC) in April to explore the latest advancements in wireless cellular communications and cooperative systems. In all, more than 550 technical papers, sessions, panels and keynotes highlighted the future of wireless communications, systems and applications as well as the newest technologies, applications, market trends and business implications.**

At the invitation of WCNC 2009 General Chair Lajos Hanzo of the University of Southampton, the four-day event held in Budapest, Hungary officially commenced with an opening salutation from IEEE President John Vig. Citing "the huge momentum in contributions," Vig thanked HTE and the society's worldwide fellowship of members, volunteers and strategic network of partners, including IEEE Press & Wiley publishers, for their ongoing support, which over the past 57 years helped to make IEEE ComSoc one of the world's foremost technical communications organizations. Dr Vig also participated in a meeting co-organized by a number of local IEEE Chapters.

An IEEE Life Fellow and an active IEEE participant for the past 30 years, Vig also attended the 'Green Radio' panel discussion later that evening, which was convened and moderated by Lajos Hanzo. The panel included distinguised British Professors Aghvami and McLaughlin as well as Dr Hoshyar, who collaborate under the recent 'Green Radio' initiative of the UK's Virtual Centre of Excellence known as VCE, also funded by the Engineering and Physical Sciences Research Council (EPSRC).

Andrea Goldsmith, Professor of Electrical Engineering at Stanford University, delivered one of the morning keynotes on "The Next Wave in Wireless Technology: Challenges and Solutions." During her address Prof. Goldsmith emphasized the "exponential worldwide growth enjoyed by the wireless communications industry" over the past few decades and "the role of next generation, high-performance wireless networks, which must be designed to support significant increases in data rates, coverage, spectral and energy efficiencies, reliability with the aid of new networking paradigms." According to her, "The next wave of wireless technology is upon us. Wireless communication systems are increasingly expected to deliver higher data-rates (Gbps) with low latency and reliable coverage in both indoor and outdoor environments, while supporting new services. "But, there are numerous challenges ranging from the size and cost of devices to the management of interferences at the system level. As a result, we must make more efficient use of the wireless spectrum and create an innovative vision, which treats interference as a friend that can be exploited through cooperation, cognition and cross-layer protocol designs, including sophisticated relay strategies."

Continuing the theme, Gerhard Fettweis, Vodafone Chair in TU Dresden, commented on the "Current Frontiers in Wireless Communications: Fast & Green & Dirty," while addressing the future's hottest research challenges. This includes "enabling high cellular data rates with increased spectral efficiency and fairness," "overcoming analog impairments with the aid of sophisticated RF design" and tackling "the challenges of designing 'green radio.'" "Yesterday, we believed the cellular phone would be the "black hole" of integration, encompassing all wireless standards, allowing communication over an increasing number of air interfaces," stated Fettweis. "Today we see that we were wrong: e.g. DVB-enabled phones have only a modest market share and UWB is currently out. A better insight into the factors deciding the success or failure of the diverse solutions is needed. The "wireless roadmap" of the past gives us researchers valuable input towards understanding what sort of solutions will be needed in the future as well as the challenges that will no doubt keep future generations of researchers busy."

For more information on next year's conference and paper submission guidelines, interested parties are urged to visit: *www.ieee-wcnc.org/wcnc*. The IEEE WCNC 2010 "Call for Papers" deadline is September 18, 2009.