

ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ  
ΜΤΧΠ Πληροφοριακών Συστημάτων

*ΕΝΕΡΓΑ ΔΙΚΤΥΑ*  
*«Το σύνθετο είναι ...απλό»*

*Κωνσταντίνος Σ. Χειλάς*

Μάθημα: Τεχνολογίες Τηλεπικοινωνιών & Δικτύων  
Υπεύθυνος Καθηγητής: Α. Α. Οικονομίδης

Θ Ε Σ Σ Α Λ Ο Ν Ι Κ Η 2 0 0 0

## **ΠΕΡΙΕΧΟΜΕΝΑ**

|   |    |
|---|----|
| Περίληψη  | 1  |
| 1. Εισαγωγή   | 2  |
| 2. Τα ενεργά δίκτυα   | 4  |
| 3. Μια γενικευμένη αρχιτεκτονική                            | 7  |
| 4. Αρχιτεκτονικές προσεγγίσεις των ενεργών δικτύων          | 9  |
| 4.1. Ο προγραμματιζόμενος διακόπτης (programmable switch)   | 9  |
| 4.2. Κάψουλες (capsules)                                    | 10 |
| 5. Στόχος ένα κοινό προγραμματιστικό μοντέλο                | 10 |
| 5.1. Κωδικοποίηση   | 10 |
| 5.2. Κοινές αρχές   | 11 |
| 5.3. Η κατανομή των πόρων των κόμβων                        | 13 |
| 5.4. Σχεδιαστικοί στόχοι                                    | 14 |
| 6. Παραδείγματα εφαρμογών                                   | 15 |
| 6.1. Μετοχές - Διαχείριση χαρτοφυλακίου                     | 17 |
| 6.2. Πλειστηριασμοί στο δίκτυο σε πραγματικό χρόνο          | 18 |
| 7. Νέες εφαρμογές στο δίκτυο                                | 18 |
| 7.1. Συγχώνευση και διανομή της πληροφορίας                 | 19 |
| 7.2. Προστασία του δικτύου                                  | 19 |
| 7.3. Διαχείριση του δικτύου                                 | 20 |
| 7.4. Active Reliable Multicast (ARM)                        | 20 |
| 7.5. Ενισχυτές πρωτοκόλλου (Protocol Boosters)              | 21 |
| 7.6. Ενεργός Έλεγχος Συμφόρησης (Active Congestion Control) | 21 |
| 7.7. Internet Enhancement                                   | 22 |
| 8. Επιτάχυνση της ανάπτυξης των υποδομών                    | 23 |
| 9. Προγραμματιστικά περιβάλλοντα εκτέλεσης                  | 24 |
| 9.1. Active Network Transport System (ANTS)                 | 24 |
| 9.2. Liquid Software  | 24 |
| 9.3. Netscript  | 25 |
| 9.4. Smartpackets   | 25 |

|   |    |
|---|----|
| 10. Λειτουργικά συστήματα των κόμβων      | 25 |
| 11. Διαλειτουργικότητα (Interoperability) | 27 |
| 11.1. ANEP                                | 27 |
| 11.2. ABONE                               | 28 |
| 12. Συμπεράσματα                          | 29 |
| <br>                                      |    |
| Βιβλιογραφία                              | 30 |

## **Περίληψη**

Τα ενεργά δίκτυα (active networks) αποτελούν μία νέα προσέγγιση της αρχιτεκτονικής των δικτύων όπου τα ενεργά στοιχεία ενός δικτύου εκτελούν συγκεκριμένες ενέργειες-υπολογισμούς στα πακέτα που διακινούνται μέσα τους. Η προσέγγιση αυτή έχει σαν κινητήριες δυνάμεις της, από τη μία πλευρά προχωρημένες εφαρμογές, οι οποίες εκτελούν υπολογισμούς που ζητά ο χρήστης σε κόμβους εντός του δικτύου, κι από την άλλη τη ραγδαία ανάπτυξη της τεχνολογίας των κωδίκων στις κινητές τηλεπικοινωνίες που καθιστούν δυνατή την ανάπτυξη δυναμικών υπηρεσιών εντός των δικτύων. Υπάρχουν σήμερα αρκετές δικτυακές εφαρμογές όπου εκτελούνται προγράμματα σε κόμβους του δικτύου που επηρεάζουν τη διακίνηση των πακέτων μέσα του. Τέτοιες είναι τα firewalls, οι webproxies, οι gateways. Τα ενεργά δίκτυα επιτρέπουν σε εφαρμογές να εισάγουν προγράμματα με δυναμικό τρόπο στους κόμβους τόσο των τοπικών όσο και των δικτύων ευρείας περιοχής, τα οποία ενεργούν επί των διακινούμενων πακέτων. Με τον τρόπο αυτό καθίσταται δυνατή η άμεση και διαφανής διάθεση νέων υπηρεσιών γιατί γίνεται ευκολότερη η διαδικασία προώθησης και ανάπτυξής τους. Αυτό συμβαίνει γιατί αποσυνδέονται οι δικτυακές υπηρεσίες από την υποδομή και καινούργιες υπηρεσίες μπορούν να προστεθούν στο δίκτυο όταν ζητηθούν. Τα ενεργά δίκτυα αποτελούν μια πρόκληση στον παραδοσιακό τρόπο λειτουργίας του Internet που βασίζεται στο πρωτόκολλο IP. Το μεγαλύτερο ενδιαφέρον τους έγκειται στο γεγονός ότι στο δίκτυο δεν υπάρχει μια προκαθορισμένη υπηρεσία αλλά αυτή καθορίζεται δυναμικά κάθε φορά από τη εκάστοτε εφαρμογή. Υπάρχουν παρόλα αυτά σοβαρές επιφυλάξεις για την εφαρμογή τέτοιων τεχνικών στα δίκτυα, οι οποίες αφορούν κυρίως στις επιδόσεις των δρομολογητών, στην ταχύτητα του δικτύου και στη ασφάλειά του.

## **1. Εισαγωγή**

Το επίπεδο λειτουργίας πολλών υπολογιστικών συστημάτων εξαρτάται σήμερα και από τις δικτυακές υπηρεσίες που χρησιμοποιούνται για να μεταφέρουν την πληροφορία μεταξύ των διαφορετικών μηχανών του συστήματος. Παρόλα αυτά οι δικτυακές υπηρεσίες έχουν αναπτυχθεί με πολύ πιο αργούς ρυθμούς από ότι άλλα στοιχεία ενός κατακεμημένου υπολογιστικού συστήματος. Αυτό μοιάζει να έρχεται σε αντίθεση με τη εικόνα που μπορεί να έχει κάποιος σχετικά με την ανάπτυξη όλο και περισσότερων εφαρμογών στο χώρο των δικτύων. Οι εφαρμογές κινητού IP (Mobile IP), Ipv6, Multicast και Integrated/ Differentiated Services έχουν σαν στόχο την υποστήριξη πολυμεσικών εφαρμογών και εξυπηρέτηση περισσότερων πελατών /χρηστών σε ένα περιβάλλον κινητών υπηρεσιών. Δυστυχώς η πρόοδος στην ανάπτυξη και διάθεση αυτών των εφαρμογών προς χρήση, απέχει πολύ από την αναγνωρισμένη ανάγκη για αυτές. Έτσι καθώς όλο και περισσότερες εφαρμογές αναπτύσσονται, απαιτούν δικτυακές υπηρεσίες που να εξυπηρετούν καλύτερα τον τρόπο λειτουργίας τους.

Μια πρόσφατη παρατήρηση προς αυτή την κατεύθυνση είναι ότι ενώ είναι δυνατό να αναπτύξουμε νέες δικτυακές υπηρεσίες στα άκρα ενός δικτύου, βάζοντάς τες να «καθίσουν» πάνω στην υπάρχουσα υποδομή, είναι καλύτερο για την

λειτουργία και τη χρηστικότητα του δικτύου να τις αναπτύξουμε στους εσωτερικούς κόμβους του.

Το κύριο πρόβλημα προς αυτή την κατεύθυνση είναι ο τρόπος με τον οποίον θα αλλάξουν τα πρωτόκολλα των δικτύων. Περισσότερο από οτιδήποτε άλλο τα πρωτόκολλα είναι ο κύριος εγγυητής της συμβατότητας και της λειτουργίας σε ένα δίκτυο. Για να χρησιμοποιηθεί ένα πρωτόκολλο πρέπει πρώτα να γίνει πρότυπο (standard). Αυτό σημαίνει αρκετά χρόνια δουλειάς από τη στιγμή που κάποιος θα αναγνωρίσει μια ανάγκη ως την ημέρα που όλοι θα συμφωνήσουν για τον τρόπο που θα δρομολογηθεί η λύση της. Κι από τη στιγμή που ένα νέο πρωτόκολλο δημιουργηθεί περνάει αρκετός καιρός ως την πλήρη εγκατάστασή του γιατί αυτή πρέπει να γίνει χειροκίνητα και επειδή πολλές φορές είναι ασύμβατο με κάποιες από τις προϋπάρχουσες εφαρμογές ή πρωτόκολλα.

Παραδείγματα εφαρμογών όπως το IP Multicast [5] και το Mobile IP [6], που δεν έχουν ακόμη αναπτυχθεί δικτυακά παρά την πολυετή και αποδεδειγμένη ανάγκη για αυτές, αποδεικνύουν το αποτέλεσμα αυτής της διαδικασίας. Από την άλλη προϊόντα συγκεκριμένων κατασκευαστών αναπτύσσονται γρηγορότερα αφού ο τρόπος υλοποίησης των διαδικασιών σε αυτά εντάσσεται σε ένα ιδιοκτησιακό (proprietary) καθεστώς, διαφανές στα άκρα του δικτύου. Για παράδειγμα οι εξυπηρετητές ενός δικτύου επωφελούνται από τις υπηρεσίες προσωρινής αποθήκευσης (caching) και διανομής των πακέτων στα διάφορα σημεία ενός δικτύου. Το προϊόν CacheDirector της Cisco ελαττώνει την κίνηση ενός δικτύου επεμβαίνοντας σε επαναλαμβανόμενες αιτήσεις εξυπηρέτησης, ενώ ο LocalDirector, της ίδιας εταιρείας, καταφέρνει το ίδιο κατανέμοντας την κίνηση σε πολλούς εξυπηρετητές. Η επεμβάσεις στα πακέτα που κινούνται στο δίκτυο γίνονται μέσα στους δρομολογητές με τρόπο διαφανή στο τελικό σύστημα. Επειδή ο τρόπος υλοποίησης δεν αποτελεί τυποποιημένη διαδικασία (standard), δημιουργείται το εξής πρόβλημα. Ενώ οι νέες υπηρεσίες είναι ικανές να αντιμετωπίσουν ένα συγκεκριμένο πρόβλημα μέσα σε έναν απλό δρομολογητή, είναι απαγορευτική η εφαρμογή των διαδικασιών αυτών σε ένα ευρύτερο δικτυακό περιβάλλον.

Τα ενεργά δίκτυα προσπαθούν να επιλύσουν το πρόβλημα της αργής ανάπτυξης υπηρεσιών εισάγοντας προγραμματιστικές διαδικασίες στην ίδια την υποδομή των δικτύων και επιτρέποντας με αυτό τον τρόπο στις νέες υπηρεσίες να εξαπλωθούν πιο γρήγορα. Βέβαια η ευελιξία αυτή επιτυγχάνεται σε βάρος της ασφάλειας και της απόδοσης. Οι υποστηρικτές των ενεργών δικτύων πιστεύουν ότι η ικανότητά του

δικτύου να προσαρμόζεται στην κάθε εφαρμογή θα δικαιολογεί την αυξημένη ανάγκη να προστατεύσουμε το δίκτυο στο σύνολό του.

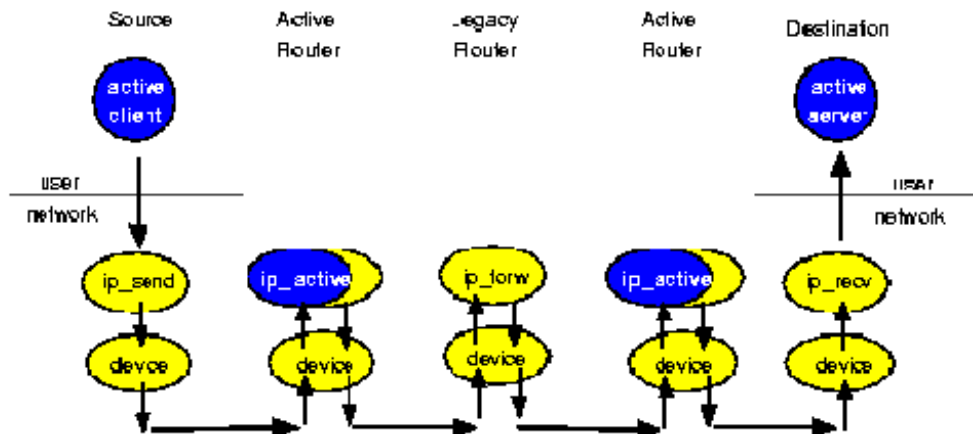
Μερικά ερωτήματα που γεννιούνται είναι:

- Με ποιόν τρόπο μπορεί να χρησιμοποιηθεί μια δικτυακή υπηρεσία για να βελτιωθεί η επίδοση των εφαρμογών;
- Πως μπορούμε να κάνουμε την δικτυακή υποδομή προγραμματίσιμη χωρίς να υποβαθμίσουμε τις επιδόσεις και την ασφάλεια του δικτύου;

## **2. Τα ενεργά δίκτυα**

Τα ενεργά δίκτυα είναι δίκτυα στα οποία μπορείς να προσθέσεις προγράμματα ή να τα προσαρμόσεις σε συγκεκριμένες εφαρμογές και σήμερα είναι γνωστά με τη μορφή των προσωρινών χώρων αποθήκευσης στο δίκτυο (web cache servers) ή των τοίχων προστασίας (firewalls). Σε ένα ενεργό δίκτυο, οι δρομολογητές ή άλλα διακοπτικά στοιχεία εκτελούν συγκεκριμένους υπολογισμούς στα μηνύματα που περνάνε από μέσα τους. Για παράδειγμα ο χρήστης ενός ενεργού δικτύου μπορεί να στείλει ένα ανιχνευτικό πρόγραμμα το οποίο θα εγκατασταθεί σε όλους τους δρομολογητές του δικτύου και θα εκτελεί συγκεκριμένες πράξεις στα πακέτα του χρήστη. Στο παρακάτω σχήμα (Σχήμα 1) βλέπουμε πως μπορεί να επαυξηθεί η λειτουργία των δρομολογητών σε ένα IP δίκτυο με τέτοιο τρόπο ώστε να εξειδικεύεται η επίδρασή τους πάνω στα datagrams που ρέουν μέσα στο δίκτυο. Αυτοί οι ενεργοί δρομολογητές μπορούν να συνεργαστούν και με άλλους συμβατικούς που διαχειρίζονται τα datagrams με τον παραδοσιακό τρόπο.

Τα δίκτυα αυτά λοιπόν είναι ενεργά με την έννοια ότι εκτελούν κάποιους υπολογισμούς ή επιφέρουν αλλαγές στα περιεχόμενα των πακέτων. Εκτός από αυτό η ενέργειές τους μπορούν να ρυθμιστούν ανάλογα με το χρήστη ή την εφαρμογή. Αντίθετα οι υπολογισμοί που εκτελούν επί των πακέτων τα παραδοσιακά δίκτυα πακέτων, όπως το Internet, είναι περιορισμένης έκτασης. Αν και οι δρομολογητές επεμβαίνουν στις επικεφαλίδες (headers) των πακέτων, δεν κάνουν κανενός είδους επεξεργασία στα δεδομένα, τα οποία μεταφέρουν με διαφάνεια. Πέρα και από αυτό οι τυχόν υπολογισμοί στις κεφαλίδες των πακέτων καθορίζονται από τις ενέργειες του χρήστη ή την εφαρμογή που δημιουργεί τα πακέτα.



Σχήμα 1. Επεξεργασία εντός των κόμβων ενός ενεργού δικτύου

Στην πιο επιτηδευμένη τους μορφή, τα ενεργά δίκτυα, είναι εύκολα προσαρμοζόμενες υποδομές που επιτρέπουν στους σχεδιαστές των δικτύων να προγραμματίσουν επίπεδα ελέγχου, τα οποία να είναι σε θέση να ελέγξουν το λογισμικό (software) και το υλικό (hardware) που χρησιμοποιείται στη διαχείριση της συμπεριφοράς του συστήματος μεταφοράς. Διάφοροι μηχανισμοί προγραμματισμού, όπως τα συστήματα ανοιχτής σηματοδότησης (open signalling), εξαπλώνονται όλο και περισσότερο.

Η έρευνα όμως στον χώρο αυτό έχει πάει ένα βήμα παραπέρα. Τα υπάρχοντα συστήματα επιτρέπουν στους σχεδιαστές να τροποποιήσουν τις υποδομές μεταγωγής πακέτων άμεσα και γρήγορα χρησιμοποιώντας είτε ένα μοντέλο που ανακατεύει τα «ενεργά» πακέτα με άλλα είτε θεωρεί όλα τα πακέτα ενεργά – εφόσον τα έχει συνδέσει με μεθόδους ενθυλάκωσης (encapsulation).

Μια γενική αρχιτεκτονική ενεργών δικτύων περιέχει πολλαπλά περιβάλλοντα εκτέλεσης (execution environments). Κάθε τέτοιο περιβάλλον δέχεται έγκυρα προγράμματα και πακέτα τα οποία είτε εκτελεί είτε αλλάζει την κατάστασή τους και μετά τα μεταδίδει. Τα συγκεκριμένα αυτά περιβάλλοντα είναι ουσιαστικά το ίδιο το περιβάλλον του ενεργού δικτύου που έχει σαν πυρήνα του μια συγκεκριμένη γλώσσα ή μοντέλο.

Η κύρια πρόκληση στην υλοποίηση αυτής της γενικής αρχιτεκτονικής είναι η εξισορρόπηση της ευχρηστίας (usability), της ευελιξίας (flexibility), της ασφάλειας και των επιδόσεων.



Το μεγαλύτερο εμπόδιο στην ευχρηστία είναι η χρήση μοντέλων και περιορισμών με τα οποία ο χρήστης δεν είναι εξοικειωμένος. Η ευελιξία αναφέρεται στην ευκολία με την οποία το υποσύστημα του δικτύου μπορεί να προσαρμοστεί σε διαφορετικές εφαρμογές – καθήκοντα. Τα θέματα ασφαλείας άπτονται της ικανότητας του δικτύου και των εφαρμογών να αναγνωρίσουν και να καθορίσουν τις κύριες οντότητες στο δίκτυο, μηχανές και χρήστες. Με τη σειρά τους οι κύριες αυτές οντότητες πρέπει να συσχετίζονται με δικαιώματα επί των εφαρμογών, των πόρων του δικτύου (π.χ. εύρος ζώνης) αλλά και επί αλλήλων. Τέλος τα θέματα των επιδόσεων υποδεικνύουν το πως πρέπει να αναπτυχθούν τα ενεργά δίκτυα για να μη δημιουργούν σημεία συμφόρησης (bottlenecks).

Σε γενικές γραμμές τα συστήματα πρώτης γενιάς τείνουν να χρησιμοποιούν μοντέρνες γλώσσες προγραμματισμού, όπως η Caml και η Java, παράλληλα με τεχνικές κρυπτογράφησης, για να επιτύχουν ευχρηστία και ασφάλεια. Οι διαφορές τους στην ευχρηστία και τις επιδόσεις, προκύπτουν κυρίως από τις διαφορετικές τεχνικές κρυπτογράφησης που χρησιμοποιούν ενώ πολλές φορές προκύπτουν και λόγω των διαφορετικών εφαρμογών που καλούνται να υποστηρίξουν.

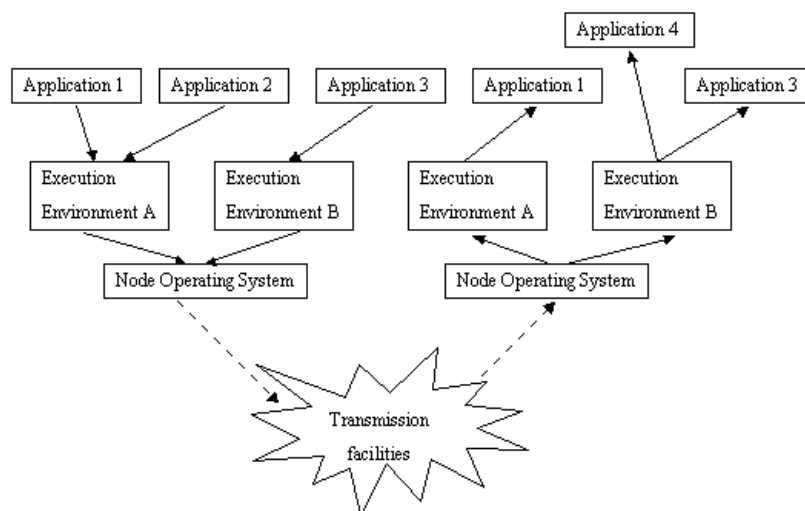
Η γενική ιδέα της ενεργούς δικτύωσης γεννήθηκε μέσα από συζητήσεις σχετικές με το μέλλον των δικτύων στα πλαίσια της ερευνητικής επιτροπής του DARPA μεταξύ του 1994 και 1995. Εκεί αναγνωρίστηκαν πολλά προβλήματα των σημερινών δικτύων, όπως: η δυσκολία ολοκλήρωσης νέων τεχνολογιών και προτύπων σε ένα διαμοιραζόμενο δικτυακό περιβάλλον, χαμηλές επιδόσεις λόγω επαναλαμβανόμενων ενεργειών σε διάφορα επίπεδα των πρωτοκόλλων καθώς και η δυσκολία να ενταχθούν νέες υπηρεσίες στις υφιστάμενες υποδομές. Η ιδέα των μηνυμάτων που μεταφέρουν διαδικασίες (procedures) και δεδομένα, μοιάζει να είναι ένα φυσικό βήμα πέρα από την παραδοσιακή μεταγωγή κυκλώματος και πακέτου. Είναι ένα τρόπος να προσαρμόζεται γρήγορα το δίκτυο σε μεταβλητές απαιτήσεις. Σε συνδυασμό με ένα ευκατανόητο περιβάλλον εκτέλεσης προγραμμάτων στους κόμβους του δικτύου, η προσέγγιση αυτή αποτελεί τη βάση για να προσεγγιστούν τα δίκτυα σαν μια σύνθεση από μικρότερα στοιχεία με συγκεκριμένες ιδιότητες. Οι υπηρεσίες διανέμονται εύκολα και προσαρμόζονται άμεσα στις διάφορες εφαρμογές. Οι αναφορές για την συνολική συμπεριφορά του δικτύου μπορούν να γίνουν με βάση τις ιδιότητες μεμονωμένων στοιχείων.

### 3. Μια γενικευμένη αρχιτεκτονική

Στο παρακάτω σχήμα (Σχήμα 2), φαίνεται μια άποψη για τη μορφή ενός κόμβου σε ένα ενεργό δίκτυο. Μία ή παραπάνω εφαρμογές τροφοδοτούν ένα υπολογιστικό περιβάλλον που τροφοδοτεί το λειτουργικό σύστημα ενός κόμβου. Ένα απλό, μονολιθικό υπολογιστικό περιβάλλον, μπορεί να είναι σε θέση να διαχειρίζεται τους πόρους ενός συστήματος αλλά μπορεί να μην λειτουργεί σωστά αν πρέπει και να μοιράσει αυτούς τους πόρους.

Η ερευνητική κοινότητα έχει αναπτύξει ένα πλαίσιο που περιγράφει τα ενεργά στοιχεία και τις διεπαφές (interfaces) μεταξύ των στοιχείων. Το πλαίσιο αυτό θεωρεί σαν παραδοσιακό δίκτυο εκείνο που περιέχει κόμβους που διασυνδέονται με διαφορετικών τύπων κανάλια και ζεύξεις. Ορίζει επίσης ένα δίκτυο μεταγωγής πακέτων που σημαίνει ότι η επικοινωνία λαμβάνει χώρα με τη μετακίνηση πακέτων μεταξύ των κόμβων του δικτύου. Το δίκτυο απλά φέρνει σε επικοινωνία τους τελικούς χρήστες αντί να προσφέρει επεξεργαστικές υπηρεσίες.

Ένα ενεργό δίκτυο διαφέρει από την κλασσική δομή μέσω των διαδικασιών που δεν ορίζει. Αντί να οριστεί το πως συνεργάζονται οι κόμβοι για την παροχή δικτυακών υπηρεσιών (π.χ. μέσω της μεθόδου της καλύτερης προσπάθειας (best-effort method)), περιγράφονται οι λειτουργικές σχισμές (functional slots) που πρέπει να ενεργοποιηθούν άμεσα για να παρασχεθεί μια συγκεκριμένη δικτυακή υπηρεσία. Οι σχισμές αυτές προσθέτουν στο σύστημα ένα βαθμό ελευθερίας που επιταχύνει την ανάπτυξη του δικτύου και προσαρμόζει νέους τύπους δικτύων, αλγορίθμους και εφαρμογές.



Σχήμα 2. Μια γενικευμένη αρχιτεκτονική ενός ενεργού δικτύου.

Όπως φαίνεται στο σχήμα (Σχήμα 2), ένας χρήστης του δικτύου (μια εφαρμογή), αλληλεπιδρά με το προγραμματιστικό περιβάλλον (Execution Environment - EE) που βρίσκεται τόσο στους κόμβους του δικτύου όσο και στα άκρα του. Αυτό το περιβάλλον είναι υπεύθυνο για όλα τα θέματα της διεπαφής χρήστη – δίκτυο. Σε αυτά περιλαμβάνονται η σύνταξη και η σημειολογία των πακέτων, η διευθυνσιοδότηση και οι λειτουργίες ονοματολογίας. Κάθε προγραμματιστικό περιβάλλον εξάγει προς το χρήστη είτε ένα API, είτε μια διευρυμένη εικονική μηχανή Java (Java Virtual Machine), είτε μια διευρυμένη πρίζα (socket) διεπαφής.

Είναι δυνατό πολλά προγραμματιστικά περιβάλλοντα να τρέχουν πάνω σε ένα κόμβο οπότε το λειτουργικό σύστημα του κόμβου αναλαμβάνει πολλές λειτουργίες διαχείρισης και ελέγχου των πόρων. Για να εξασφαλιστεί αυτή η λειτουργία, κάθε υπολογιστικό περιβάλλον αποκτά πρόσβαση στους πόρους του κόμβου μόνο μέσω του λειτουργικού συστήματος. Το τελευταίο δρα ως ενδιάμεσος για όλα τα θέματα που αφορούν στην ασφάλεια, στην αποθήκευση πληροφορίας, στη δημιουργία των πινάκων δρομολόγησης, στο διαθέσιμο υπολογιστικό εύρος ζώνης και στην επικοινωνία των πακέτων.

Έτσι τα πακέτα του τελικού χρήστη βλέπουν το δίκτυο χτισμένο σε τρία επίπεδα. Στο πρώτο είναι το API (Application Programming Interface) του λειτουργικού συστήματος που είναι αμετάβλητο σε όλο το δίκτυο και ορίζει τα βασικά δομικά στοιχεία που είναι διαθέσιμα στο προγραμματιστικό περιβάλλον. Κάθε ξεχωριστό επεξεργαστικό περιβάλλον εκτελεί μια σειρά αφαιρέσεων βασισμένο σε αυτά τα δομικά στοιχεία και τις παρουσιάζει στον τελικό χρήστη. Για να έχει πρόσβαση στην επιθυμητή υπηρεσία, ο τελικός χρήστης χρησιμοποιεί αυτές τις αφαιρέσεις.

Το βασικό μοντέλο λειτουργίας είναι ίδιο για όλα τα ενεργά δίκτυα. Τα πακέτα φτάνουν σε μια φυσική ζεύξη του δικτύου. Το λειτουργικό σύστημα του κόμβου τα ταξινομεί ανάλογα με τις επικεφαλίδες τους και τα στέλνει στα κατάλληλα λογικά κανάλια. Κάθε κανάλι είναι συσχετισμένο με μια επεξεργασία ενός πρωτοκόλλου που μπορεί να περιλαμβάνει και ελέγχους ασφαλείας. Το κανάλι παραδίδει μετά το πακέτο είτε σε ένα επεξεργαστικό περιβάλλον για παραπέρα μετάφραση και επεξεργασία είτε το ξαναστέλνει έξω στο δίκτυο.

Ένα επεξεργαστικό περιβάλλον μπορεί επίσης να είναι υπεύθυνο και για τη δημιουργία πακέτων. Τα πακέτα αυτά μπορεί να τύχουν της ίδιας επεξεργασίας με οποιοδήποτε πακέτο που κινείται στο δίκτυο. Σε ένα ενεργό δίκτυο, σε αντίθεση με τα συμβατικά δίκτυα στα οποία ένα πακέτο απλά κινείται μέσα στο δίκτυο, υπάρχει μια

ακαθόριστη σχέση ανάμεσα στα πακέτα που εισέρχονται σε έναν κόμβο και σε αυτά που εξέρχονται. Το γεγονός αυτό δίνει στους σχεδιαστές ένα μεγάλο βαθμό ελευθερίας. Σημαίνει επίσης, ότι τα υπολογιστικά περιβάλλοντα (EE) πρέπει να ενεργοποιούν όλους τους μηχανισμούς που είναι απαραίτητοι για να εξασφαλισθεί η ευστάθεια του δικτύου. Το λειτουργικό σύστημα των κόμβων (NodeOS) μπορεί μόνο να εξασφαλίσει ότι οποιαδήποτε αστάθεια σε ένα υπολογιστικό περιβάλλον δεν θα επηρεάσει τους πόρους που διατίθενται στα υπόλοιπα.

Για να απλοποιηθεί η επικοινωνία (porting) των υπολογιστικών περιβαλλόντων προς τα υποκείμενα λειτουργικά συστήματα, οι προτεινόμενες αρχιτεκτονικές ενεργών δικτύων ορίζουν μια διεπαφή EE-to-NodeOS ή μια διεπαφή NodeOS.

#### **4. Αρχιτεκτονικές προσεγγίσεις των ενεργών δικτύων**

Γενικά υπάρχουν δύο προσεγγίσεις στην υλοποίηση των ενεργών δικτύων. Η προσέγγιση του προγραμματιζόμενου διακόπτη (programmable switch) και η προσέγγιση της κάψουλας (capsule).

##### **4.1. Ο προγραμματιζόμενος διακόπτης (programmable switch)**

Η αρχιτεκτονική αυτή διατηρεί την υπάρχουσα μορφοποίηση των πακέτων ή των κυψελίδων και παρέχει έναν διακριτικό μηχανισμό μέσω του οποίου φορτώνονται τα προγράμματα. Αυτή η οπτική διατηρεί και τη σημερινή πρακτική στα δίκτυα που ζητά μετάδοση δεδομένων εντός ζώνης (in-band) και διαχείριση εκτός ζώνης (out-of-band).

Οι χρήστες πρώτα στέλνουν ένα πρόγραμμα σχετικό με την εφαρμογή τους στους δρομολογητές του δικτύου. Αφού το πρόγραμμα εγκατασταθεί, αποστέλλονται τα δεδομένα. Μόλις ένα πακέτο φτάσει σε κάποιον δρομολογητή, εξετάζεται η επικεφαλίδα του και ενεργοποιείται το κατάλληλο από τα ήδη εγκατεστημένα προγράμματα για να επιδράσει επί των δεδομένων.

Οι διαφορετικοί μηχανισμοί «φορτώματος» και εκτέλεσης των προγραμμάτων είναι χρήσιμοι όταν πρέπει να δοθεί ιδιαίτερη προσοχή σε μία από τις δύο διαδικασίες. Όταν για παράδειγμα ο κατασκευαστής επιτρέπεται να «μπει» σε έναν δρομολογητή για να εισάγει μια νεότερη έκδοση του λογισμικού του. Ειδικοί μηχανισμοί ελέγχουν την ταυτότητα του κατασκευαστή αλλά και την ποιότητα του

κώδικα που εισάγεται. Διαχωρίζοντας την εισαγωγή των προγραμμάτων από την επεξεργασία των μηνυμάτων μοιάζει να είναι ιδιαίτερα ελκυστικό όταν η επιλογή των προγραμμάτων γίνεται από τους διαχειριστές του δικτύου αντί από τους χρήστες.

#### **4.2. Κάψουλες (capsules)**

Η αρχιτεκτονική προσέγγιση της κάψουλας (capsule) πηγαίνει λίγο μακρύτερα. Μετατρέπει τα παθητικά πακέτα των σημερινών δικτύων, σε ενεργά προγράμματα που ενθυλακώνονται (encapsulated) στα πλαίσια που μεταδίδονται και εκτελούνται στους κόμβους του δικτύου καθώς κινούνται μέσα από αυτό. Κάθε πακέτο γίνεται ένα πρόγραμμα! Όταν μια κάψουλα φτάσει σε έναν ενεργό κόμβο, τότε αξιολογούνται τα δεδομένα που μεταφέρει με έναν τρόπο αντίστοιχο με αυτό που ένας Postscript εκτυπωτής μεταφράζει τα περιεχόμενα ενός αρχείου που έχει λάβει.

Τα ψηφία που φτάνουν στην είσοδο ενός κόμβου, υπόκεινται σε διαδικασία που αναγνωρίζει τα όρια της κάψουλας, πιθανόν με τον ίδιο τρόπο που χρησιμοποιεί το κλασσικό επίπεδο σύνδεσης στα δίκτυα. Τα περιεχόμενα της κάψουλας μεταφέρονται σε ένα μεταβατικό χώρο εκτέλεσης για περαιτέρω επεξεργασία. Υποθέτουμε ότι τα πακέτα αποτελούνται από προγράμματα που εκτελούν κάποιους βασικούς υπολογισμούς επί των δεδομένων της κάψουλας ή έχουν τη δυνατότητα να εμφανίσουν άλλα προγράμματα και διαδικασίες που με τη σειρά τους προσφέρουν πρόσβαση σε πόρους εξωκείμενους του δικτύου. Η εκτέλεση της κάψουλας μπορεί να έχει σαν αποτέλεσμα τον προγραμματισμό της διακοπής της διαδικασίας διοχέτευσης άλλων πακέτων από τον κόμβο και τη μεταβολή της κατάστασης του.

### **5. Στόχος ένα κοινό προγραμματιστικό μοντέλο**

Τα προγράμματα που θα χρησιμοποιηθούν στα ενεργά δίκτυα πρέπει να είναι ικανά να διακινούνται εντός της δικτυακής υποδομής και να εκτελούνται σε διαφορετικές πλατφόρμες. Αυτό υποδεικνύει την υιοθέτηση κοινών μοντέλων για την κωδικοποίηση των προγραμμάτων, τις ενσωματωμένες στους διάφορους κόμβους βασικές λειτουργίες και τέλος, την περιγραφή και κατανομή των πόρων των κόμβων.

#### **5.1. Κωδικοποίηση**

Οι στόχοι στο θέμα της κωδικοποίησης είναι:

α) η κινητικότητα (mobility) – είναι η ικανότητα να μεταφέρονται και να εκτελούνται τα προγράμματα σε διαφορετικές πλατφόρμες.

β) η ασφάλεια – η ικανότητα να περιορίζονται οι πόροι στους οποίους μπορεί να έχει πρόσβαση το κάθε πρόγραμμα.

γ) απόδοση (efficiency) – να ενεργοποιούνται τα παραπάνω χωρίς συμβιβασμούς στην απόδοση του δικτύου, τουλάχιστον στις πιο απλές και κοινές περιπτώσεις.

Η κινητικότητα μπορεί να επιτευχθεί σε διάφορα επίπεδα απεικόνισης του προγράμματος, π.χ. να εκφράσουμε το πρόγραμμα μέσω κάποιας γλώσσας υψηλού επιπέδου όπως η Tcl, ή να υιοθετηθεί μια ενδιάμεση απεικόνιση που να είναι ανεξάρτητη από την πλατφόρμα όπως η Java, ή να μεταφέρουμε το πρόγραμμα με δυαδικό τρόπο όπως το Omniware. Στον παρακάτω πίνακα (Πίνακας 1), φαίνονται οι σημερινές τεχνολογίες που επιτρέπουν την ασφαλή και αποδοτική εκτέλεση των προγραμμάτων σε διάφορα επίπεδα. Αναμένεται και οι τρεις προσεγγίσεις να αποδειχθούν χρήσιμες.

Μια πιθανή προσέγγιση στη συνεργασία των κόμβων μπορεί να είναι μέσω ενδιάμεσης κωδικοποίησης για την υποστήριξη της κινητικότητας του κώδικα. Με αυτό τον τρόπο τόσο οι κατασκευαστές όσο και οι χρήστες ενθαρρύνονται να αναπτύξουν δικούς τους κώδικες στους κόμβους υπό την προϋπόθεση ότι θα είναι δυνατή η δημιουργία κοινά κατανοητού ενδιάμεσου κώδικα. Υπάρχουν ήδη αρκετές τεχνολογίες που προσφέρουν μεταφραστές (compilers) και λειτουργικά συστήματα για την υποστήριξη τέτοιων διαδικασιών.

## **5.2. Κοινές αρχές**

Οι υπηρεσίες που θα είναι ενσωματωμένες στους κόμβους μπορεί να περιέχουν πολλές κατηγορίες διαδικασιών [15]. Μπορεί να είναι διαδικασίες οι οποίες να επιτρέπουν στο πακέτο να μεταβάλλεται από μόνο του, π.χ. να αλλάζει την επικεφαλίδα του, το μήκος του κ.α. Άλλες μπορεί να επιτρέπουν πρόσβαση στο περιβάλλον του κόμβου, π.χ. στην κατάσταση της ζεύξης και τη διεύθυνση του, ενώ άλλες να επεμβαίνουν στον έλεγχο ροής, στην προώθηση των πακέτων, στην αντιγραφή ή στην απόρριψή τους. Ενδιαφέρουσες λειτουργίες είναι εκείνες που διαχειρίζονται προγραμματισμένες ενέργειες πάνω σε πολλά πακέτα ταυτόχρονα.

Πίνακας 1. Τεχνολογίες κωδικοποίησης των προγραμμάτων (M-κινητικότητα, S-ασφάλεια, E-απόδοση)

| Project                                 | M | S | E | Περιγραφή  |
|---|---|---|---|--|
| Safe-Tcl<br>(πηγαίο)                    | X | X |   | Είναι μια γλώσσα που παρέχει ασφάλεια μέσω της μετάφρασής της [11]. Ο περιορισμός της εκτέλεσης του προγράμματος μέσα στο επιτρεπτό περιβάλλον γίνεται από τον μεταφραστή.   |
| Java<br>(ενδιάμεσος)                    | X | X | X | Η Java χρησιμοποιεί ένα ενδιάμεσο σετ εντολών για να επιτύχει κινητικότητα [12]. Παραδοσιακά, η ασφαλής εκτέλεση ενός ενδιάμεσου κώδικα εξαρτάται από την προσεκτική μετάφρασή του. Μια από τις σημαντικές συνεισφορές της Java σε αυτό είναι ότι απαλλάσσει τον διερμηνέα (interpreter) από ένα μεγάλο κομμάτι αυτής της ευθύνης. |
| Omniware<br>(αντικείμενα)               | X | X | X | Ο κώδικας με αντικείμενα Omniware [13], βασίζεται σε απομόνωση των σφαλμάτων μέσω λογισμικού. Προϋποθέτει ένα πολύ συγκεκριμένο σετ κανόνων σχετικά με την αλληλουχία εντολών και της αριθμητικής των διευθύνσεων οι οποίες δημιουργούν ένα κουτί ("sandbox") μέσα στο οποίο περιορίζεται η εκτέλεση του προγράμματος.             |
| Proof-Carrying<br>Code<br>(αντικείμενα) |   | X | X | Ο PCC χρησιμοποιεί μιά νέα μέθοδο για να πετύχει ασφάλεια [14]. Προσθέτει στο πρόγραμμα μια επίσημη απόδειξη των ιδιοτήτων του. Ο παραλήπτης ελέγχει μόνο την εγκυρότητα του αντιγράφου που είναι απλούστερο από το να ελέγξει ολόκληρο το πρόγραμμα. Για την ώρα η διαδικασία είναι εφικτή μόνο για μικρά προγράμματα.            |

**Πίνακας 2. Τεχνολογίες λειτουργικών συστημάτων**

| <b>Πρόγραμμα</b> | <b>Περιγραφή</b>  |
|------------------|---|
| Scout            | Υποστηρίζει εργασίες προσανατολισμένες στην επικοινωνία [16]. Ασχολείται με τον καταμερισμό και τον προγραμματισμό των πόρων εφαρμόζοντας βελτιώσεις που σκοπό έχουν την αύξηση της διαμεταγωγής και την ελάττωση του χρόνου αδρανούς λειτουργίας.  |
| Exokernel        | Ο exokernel επιτρέπει σε προγράμματα να αποκτούν χαμηλού επιπέδου πρόσβαση στους πόρους του συστήματος με ασφαλή τρόπο [17]. Δημιουργεί ένα λεπτό επίστρωμα πάνω από το υλικό, που επιτρέπει στα προγράμματα να κάνουν τις δικές τους παρεμβάσεις στις υπηρεσίες του λειτουργικού.  |
| SPIN             | Βασίζεται στις ιδιότητες της γλώσσας Modula 3 και σε έναν αξιόπιστο μεταφραστή που δεν ξεπερνά το περιβάλλον όπου είναι περιορισμένος [18]. Τα προγράμματα που έχει υπογράψει ο μεταφραστής μπορούν να φορτωθούν δυναμικά στο λειτουργικό σύστημα.  |
| ‘C               | Η ‘C και ο VCODE επιτρέπουν την άμεση δημιουργία κώδικα [19]. Αυτό επιτρέπει στα πηγαία προγράμματα να συγκεκριμενοποιούν τη λειτουργία τους ή ακόμη και να δημιουργούνται τη στιγμή που τρέχουν. Με αυτές τις τεχνολογίες μπορεί οι ενεργοί κόμβοι να είναι σε θέση να μεταφράσουν πολύ κοινά προγράμματα σε διαδική κωδικοποίηση. |

### **5.3. Η κατανομή των πόρων των κόμβων**

Εκτός από την κωδικοποίηση και τις βασικές διαδικασίες (primitives), πρέπει να υπάρχουν κοινά μοντέλα που να αφορούν στους πόρους των κόμβων και στον τρόπο με τον οποίο επικοινωνούν μεταξύ τους οι «πολιτικές» διαδικασίες που ασχολούνται με την κατανομή τους. Οι πόροι που πρέπει να μοντελοποιηθούν είναι τόσο φυσικοί όπως το εύρος ζώνης, η επεξεργαστική και αποθηκευτική χωρητικότητα, όσο και λογικοί όπως οι πίνακες δρομολόγησης και η βάση που περιέχει τις πληροφορίες διαχείρισης του κόμβου (MIB). Η ασφαλής κατανομή των πόρων είναι ένα ευαίσθητο θέμα.



#### 5.4. Σχεδιαστικοί στόχοι

Η αρχιτεκτονική των ενεργών δικτύων καλείται να ακολουθήσει πέντε βασικούς στόχους.

- Να ελαχιστοποιήσει τα τυποποιημένα πρωτόκολλα που απαιτούνται για την κατασκευή και ανάπτυξη υπηρεσιών τελικού χρήστη. Η ελάττωση των πολλών παγκόσμιων συμφωνιών που χρειάζονται, εξυπηρετεί τόσο ερευνητικούς όσο και εμπορικούς σκοπούς. Οι σχεδιαστές μπορούν να πειραματίζονται με ένα «ζωντανό» δίκτυο γιατί το δίκτυο είναι τόσο ευέλικτο που μπορεί να εξυπηρετεί όλες τις ανάγκες κατά τη λειτουργία του. Μια πλειάδα τηλεπικοινωνιακών υπηρεσιών μπορεί να υλοποιηθεί μέσω των προγραμματιστικών περιβαλλόντων.
- Να μεγιστοποιήσει την ευελιξία σχετικά με τις υπηρεσίες που υποστηρίζονται. Μια βασική προϋπόθεση για αυτό είναι να μπορεί να υποστηρίξει πολλές υπηρεσίες ταυτόχρονα. Οι πόροι του δικτύου που ελέγχει το λειτουργικό σύστημα του κόμβου (NodeOS), είναι αυτοί που χρειάζεται το επεξεργαστικό περιβάλλον. Υπό αυτή την οπτική η αρχιτεκτονική μοιάζει με αρχιτεκτονική πυρήνα στην οποία τα ΕΕ πολυεπεξεργάζονται. Επειδή πολλά επεξεργαστικά περιβάλλοντα μπορούν να τρέχουν ταυτόχρονα σε έναν κόμβο, υπάρχει η δυνατότητα να προχωρά ταυτόχρονα και η έρευνα σε διάφορα προγραμματιστικά μοντέλα. Αυτό είναι σημαντικό για δύο λόγους: πρώτον, αν και οι ερευνητές αποκτούν όλο και μεγαλύτερη αίσθηση για την λειτουργία των μοντέλων πρέπει σε κάθε περίπτωση να επιλεγεί το καλύτερο. Δεύτερον, οι σχεδιαστές μπορούν να οδηγούνται σε νέα μοντέλα υποστηρίζοντας ταυτόχρονα και τα παλιά.
- Να επιτρέψει τη διασύνδεση δικτύων που λειτουργούν υπό διαφορετική διαχείριση. Αυτό αναγνωρίζει την ύπαρξη διαφορετικών σχέσεων εμπιστοσύνης μεταξύ διαχειριστικών μονάδων. Επίσης σημαίνει την πρωταρχική ανάγκη ρύθμισης θεμάτων ασφαλείας. Στη γενικότερη περίπτωση το λειτουργικό σύστημα του κάθε κόμβου παρέχει υπηρεσίες ασφαλείας στα επεξεργαστικά περιβάλλοντα.

- Να υποστηρίζει την κλιμάκωση σε μέγεθος και ταχύτητα. Αυτό σημαίνει να γίνεται άμεση διευθέτηση της κίνησης πακέτων που δεν χρειάζονται επεξεργασία.
- Να συμπεριλάβει σημερινά πρωτόκολλα, όπως το πρωτόκολλο του Ίντερνετ (IP) ως ειδικές περιπτώσεις. Στη γενικευμένη αρχιτεκτονική ο σωρός πρωτοκόλλων του IP μπορεί να αντιμετωπιστεί σαν ένα υπολογιστικό περιβάλλον, παρά την απλή μορφή του API και του ρόλου του.

## **6. Παραδείγματα εφαρμογών**

Η έρευνα στα ενεργά δίκτυα κινείται από τις «ωθήσεις» τις τεχνολογίας και τις «έλξεις» των χρηστών. Η «ώθηση» αναφέρεται στην μεγάλη συλλογή εφαρμογών όπως: προσωρινοί χώροι αποθήκευσης στο δίκτυο (web caches), πύλες μετατροπής πρωτοκόλλων βίντεο (video gateways), συσκευές κινητών ή απλών δικτυακών ενδιάμεσων (proxies), δρομολογητές κ.α., που εκτελούν υπολογισμούς εξαρτώμενους από τον χρήστη σε κόμβους μέσα σε ένα δίκτυο. Σε πολλές περιπτώσεις οι υπηρεσίες αυτές, που είναι και οι κύριοι οδηγοί της αγοράς (πίνακας 3), υλοποιούνται σε κόμβους όπως τα firewalls και εκτελούν επεξεργασίες ανάλογες της εφαρμογής. Ο σκοπός είναι να αντικαταστήσουμε αυτές τις ad hoc εφαρμογές με μια γενικευμένη ικανότητα των κόμβων του δικτύου να είναι προγραμματίσιμοι από τους χρήστες.

Η «ώθηση» της τεχνολογίας είναι η ανάπτυξη των ενεργών τεχνολογιών που καθιστά τους στόχους μας προσεγγίσιμους. Η οπτική των διαχειριστών των δικτύων δημιουργεί προβληματισμούς σχετικά με την ασφάλεια και απόδοση των δικτύων. Παρόλα αυτά η σημερινή πρόοδος στον τομέα των γλωσσών προγραμματισμού, των μεταφραστών και των λειτουργικών συστημάτων ανοίγει νέους ορίζοντες στην εκτέλεση κινητών προγραμμάτων. Παρόμοιες τεχνικές εφαρμόζονται σήμερα μεταξύ εφαρμογών και τελικού χρήστη, π.χ. μέσω εφαρμογών σε γλώσσα Java (Java applets). Τα ενεργά δίκτυα έχουν σκοπό να εισάγουν τις τεχνικές αυτές εντός του δικτύου αλλάζοντας ριζικά την οπτική μας για ένα δίκτυο.

Έτσι λοιπόν τα ενεργά δίκτυα μπορεί να έχουν αντίκτυπο στη μεταρρύθμιση των δικτύων και στις νέες εφαρμογές που θα ενεργοποιηθούν.

Πίνακας 3. Πρωτοποριακές δικτυακές εφαρμογές

| ΚΑΤΗΓΟΡΙΑ            | ΠΕΡΙΓΡΑΦΗ  |
|----------------------|--|
| Firewalls            | Οι «πύρινοι τοίχοι» είναι στην ουσία φίλτρα που καθορίζουν ποια πακέτα θα περάσουν με διαφάνεια και ποια θα μπλοκαριστούν. Αν και εκτελούν εργασίες δρομολόγησης όπως ένας συμβατικός δρομολογητής, εκτελούν επίσης και ενέργειες που καθορίζονται από τον χρήστη. Η ανάγκη να ενημερώνονται διεξοδικά πριν την χρήση νέων πρωτοκόλλων είναι ένα κώλυμα στη χρήση τους. Μέσω των ενεργών διαδικασιών είναι δυνατή η άμεση ενημέρωσή τους και μάλιστα αυτοματοποιημένα μέσω λογισμικού που θα εισάγουν μέσα τους όσες εταιρείες έχουν την άδεια να το κάνουν. |
| Web Proxies          | Οι συσκευές αυτές παρέχουν υπηρεσίες διαφανείς στο χρήστη δομημένες έτσι ώστε να ταιριάζουν στην εξυπηρέτηση και προσωρινή αποθήκευση ιστοσελίδων.   |
| Nomadic Routers      | Είναι ένας δρομολογητής που παρεμβάλλεται ανάμεσα στο δίκτυο και τον τελικό χρήστη. Το σύστημα ελέγχει και προσαρμόζεται ανάλογα στον τρόπο με τον οποίο ο χρήστης συνδέεται στο δίκτυο. Έτσι αν η σύνδεση είναι μέσω modem ενεργοποιεί διαδικασίες συμπίεσης και διαχείρισης του διαθέσιμου εύρους ζώνης ή αν ο χρήστης βρίσκεται εκτός του τοπικού δικτύου της εταιρείας του ενεργοποιεί διαδικασίες αυξημένης ασφάλειας [7].  |
| Transport Gateways   | Είναι κόμβοι τοποθετημένοι σε στρατηγικά σημεία στο δίκτυο οι οποίοι γεφυρώνουν δίκτυα με διαφορετικό εύρος ζώνης ή χαρακτηριστικά αξιοπιστίας, π.χ. ένα ενσύρματο κι ένα ασύρματο δίκτυο. Για να υποστηρίξει ασύρματη πρόσβαση σε ενσύρματα δίκτυα η διαδικασία TCP snooping διατηρεί πληροφορίες για κάθε σύνδεση στους ασυρματικούς σταθμούς βάσης [8].   |
| Application Services | Μια τέτοια περίπτωση είναι η εξυπηρέτηση μιας τηλε-διάσκεψης μεταξύ μελών με διαφορετικούς περιορισμούς στο εύρος ζώνης, π.χ. ISDN (H.320) με IP (H.323).  |

Παραθέτονται παρακάτω δύο παραδείγματα που επιδεικνύουν το εύρος των περιπτώσεων εφαρμογών που καλούνται να εξυπηρετήσουν τα ενεργά δίκτυα. Καμία από τις δύο περιπτώσεις δεν μπορεί να υλοποιηθεί σήμερα στο επίπεδο δικτύου.

### **6.1. Μετοχές – Διαχείριση χαρτοφυλακίων**

Πολλοί άνθρωποι χρησιμοποιούν σήμερα το Internet για την παρακολούθηση και διαχείριση του χαρτοφυλακίου των μετοχών τους. Η γρήγορη πρόσβαση σε ενημερωμένες τιμές τίτλων είναι εξαιρετικά κρίσιμη, ιδιαίτερα σε περιόδους υπερφόρτωσης των εξυπηρετητών (γιατί πιθανά αυτό σημαίνει ότι είναι και περίοδοι έντονης διακύμανσης της αγοράς). Συστήματα μνήμης προσωρινής αποθήκευσης (cache) δεν είναι χρήσιμα σε αυτές τις περιπτώσεις. Αυτό συμβαίνει για διάφορους λόγους. Αρχικά τα συστήματα αυτά δεν αποθηκεύουν τιμές μετοχών γιατί είναι δυναμικά δεδομένα. Δεύτερον, ακόμη κι αν μπορούσαμε να υπερπηδήσουμε το εμπόδιο της προσωρινής αποθήκευσης δυναμικά μεταβαλλόμενων δεδομένων θα αντιμετωπίζαμε τη διαφοροποίηση των σελίδων που κάθε χρήστης ενδιαφέρεται να κατεβάσει. Ακόμη κι αν υποθέσουμε ότι δύο χρήστες έχουν τους ίδιους τίτλους θα διαφέρει η ποσότητα που έχει ο καθένας. Έτσι η πιθανότητα να επαναληφθεί αίτηση για την ίδια σελίδα είναι σχεδόν ανύπαρκτη. Ακόμα περισσότερο αν θεωρήσουμε ότι τα δεδομένα της σελίδας που έχει αποθηκευτεί θα είναι άχρηστα σε οποιονδήποτε χρήστη μετά από παρέλευση ενός τετάρτου της ώρας.

Σε ένα ενεργό δίκτυο τα πράγματα μπορούν να είναι διαφορετικά. Αρχικά ένα ενεργό πρωτόκολλο θα ειδικεύεται να αποθηκεύει τιμές ανά μετοχή. Μετά κάποιο πεδίο στα πακέτα του χρήστη θα μπορεί να ενημερώνει τον εξυπηρετητή για το χρονικό περιθώριο ενημερότητας που ενδιαφέρει τον χρήστη. Έτσι ένας χρήστης που δεν ενδιαφέρεται για την πιο πρόσφατη τιμή θα μπορεί να έχει αμεσότερη εξυπηρέτηση, ενώ κάποιος που θέλει την πιο καινούργια τιμή θα περιμένει περισσότερο χρόνο (εξαρτώμενο από το διαθέσιμο εύρος ζώνης) για να εξυπηρετηθεί.

Η βασική ιδέα είναι ότι οι τιμές αποθηκεύονται μαζί με μια σήμανση του χρόνου. Τα πακέτα των αιτήσεων του χρήστη ελέγχονται στον κόμβο κι αν διαπιστωθεί ότι βρίσκονται εντός των χρονικών ορίων που υπάρχει πληροφορία εξυπηρετούνται αλλιώς η αίτηση προωθείται προς τον εξυπηρετητή.

## **6.2. Πλειστηριασμοί στο Δίκτυο σε πραγματικό χρόνο (On-line Auctions)**

Σύμφωνα με μετρήσεις, οι τόποι του Διαδικτύου με τις περισσότερες επισκέψεις είναι εκείνοι που φιλοξενούν ζωντανές δημοπρασίες. Ένας εξυπηρετητής με τέτοια εφαρμογή συλλέγει και επεξεργάζεται τις εντολές των πελατών για τα διαθέσιμα προϊόντα. Εκτός από αυτό ο εξυπηρετητής είναι υπεύθυνος για την ενημέρωση των τρεχόντων τιμών. Λόγω των καθυστερήσεων του δικτύου είναι πολύ μεγάλη η πιθανότητα η ανταπόκριση του εξυπηρετητή στα ερωτήματα των πελατών να φτάνει σε αυτούς καθυστερημένα. Οι ετεροχρονισμένες τιμές που φτάνουν στον πελάτη προξενούν πιθανές αιτήσεις του για αγορά οι οποίες όταν φτάσουν στον εξυπηρετητή πρέπει να απορριφθούν λόγω της νέας καθυστέρησης που έχει δημιουργηθεί.

Όλες οι σημερινές λύσεις τέτοιων εξυπηρετητών υλοποιούν τις εφαρμογές αυτές πάνω στον ίδιο τον εξυπηρετητή. Με τη χρήση ενεργού δικτύου είναι δυνατή η αποφόρτιση της κίνησης στον εξυπηρετητή με τον εξής τρόπο. Το φιλτράρισμα των εντολών των πελατών γίνεται σε κόμβους μακριά από τον κύριο εξυπηρετητή. Μόλις αυτός βρεθεί σε κατάσταση υπερφόρτωσης, στέλνει στους κόμβους μια συνεχή ενημέρωση για τις τιμές των προϊόντων. Όσες εντολές από τους πελάτες φτάνουν στους κόμβους με τιμές μικρότερες από την τρέχουσα τιμή του προϊόντος απορρίπτονται και στέλνεται στον πελάτη μία σχετική ενημέρωση. Όσες εντολές περνάνε το κριτήριο της τιμής στέλνονται στον εξυπηρετητή για περαιτέρω επεξεργασία. Αυτή η διαδικασία απελευθερώνει πόρους στον εξυπηρετητή και στο δίκτυο γύρω από αυτόν. Τα φίλτρα στους ενεργούς κόμβους μπορούν ακόμα να κρατάνε αρχεία με τις απορριφθείσες αιτήσεις τα οποία στέλνουν πίσω στον εξυπηρετητή μετά το πέρας της δημοπρασίας.

Ο εξυπηρετητής της δημοπρασίας και εκείνος της χρηματιστηριακής εφαρμογής μοιάζουν στο ότι χρησιμοποιούν διαδικασίες προσωρινής αποθήκευσης στους κόμβους του δικτύου. Παρόλα αυτά το πρωτόκολλό του είναι διαφορετικό γιατί αναθέτει συγκεκριμένα κομμάτια της λειτουργίας του (το φιλτράρισμα των εντολών) σε προγράμματα-αντιπροσώπους σε άλλα σημεία του δικτύου.

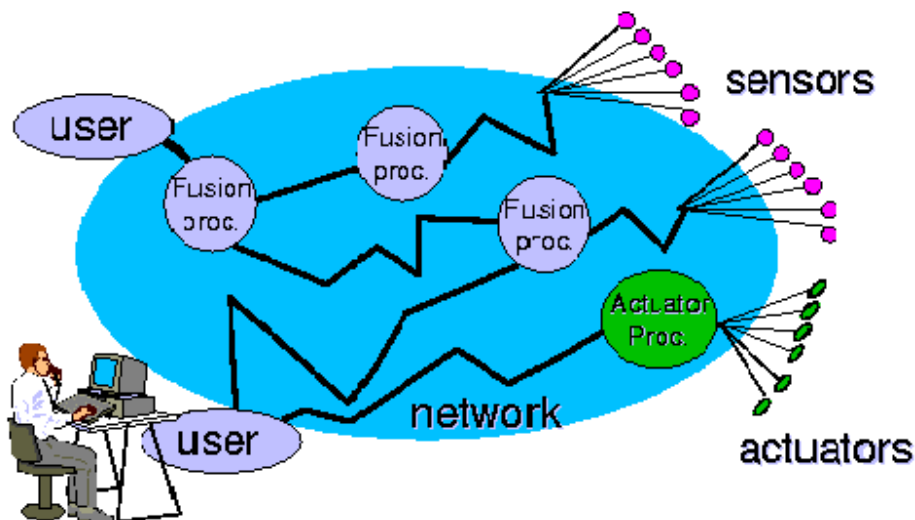
## **7. Νέες εφαρμογές στο δίκτυο**

Ποιες μπορούν να είναι οι καινούργιες εφαρμογές μέσα στα δίκτυα; Η συγχώνευση της πληροφορίας, η προστασία του δικτύου και η ενεργός διαχείρισή

του. Οι εφαρμογές των ενεργών δικτύων αναπτύσσονται συνεχώς. Περιλαμβάνουν από αξιόπιστες εφαρμογές εκπομπής προς πολλούς κόμβους (multicast) μέχρι διαχείριση του Internet και έλεγχο συμφόρησης.

### 7.1. Συγχώνευση και διανομή της πληροφορίας

Μέσω του ιστού και του Mbone έχει ξεκινήσει η εποχή των πολυ-χρηστικών και πολυ-τοπικών (Multi-user, multi-site) εφαρμογών. Πολλές εφαρμογές απαιτούν την ένωση και τη διανομή πληροφοριών. Τα υπάρχοντα συστήματα στηρίζονται σε υπηρεσίες με περιορισμένες δυνατότητες. Στο σχήμα 3 φαίνεται η περίπτωση ενός δικτύου όπου μέσω της κατανομής της εφαρμογής επιτυγχάνουμε ελάττωση των υπολογισμών και της αποθήκευσης στο δίκτυο. Στην περίπτωση του σχήματος κάθε χρήστης μπορεί να δει σε πολλά σημεία, εικόνες που προέρχονται από κατανεμημένους αισθητήρες. Πέρα από αυτό κάθε χρήστης μπορεί να επιλέξει το είδος της συμπίεσης ή της κωδικοποίησης που επιθυμεί. Η σύνδεση των δεδομένων ελαττώνει το εύρος ζώνης που χρειάζεται ο κάθε χρήστης και ελαφρώνει έτσι τα άκρα του δικτύου που έχουν ούτως ή άλλως χαμηλές ταχύτητες.



Σχήμα 3. Μία περίπτωση ένωσης και διανομής πληροφορίας μέσα στο δίκτυο

### 7.2. Προστασία του δικτύου

Προστασία της πληροφορίας σημαίνει η σωστή πληροφορία να φτάνει στον σωστό παραλήπτη τη σωστή στιγμή. Κατά καιρούς έχουν προταθεί πολλές λύσεις για

την υλοποίηση μηχανισμών ασφάλειας και εγκυρότητας στα δίκτυα. Τα ενεργά δίκτυα πιθανόν να δεχθούν την ενσωμάτωση σε αυτά ενός μηχανισμού που να ελέγχει όλους τους πόρους του δικτύου και την πληροφορία που ρέει εντός του. Αυτό εξαλείφει την ανάγκη για διαφορετικά πρωτόκολλα ασφαλείας σε διαφορετικά επίπεδα των πρωτοκόλλων. Επιτρέπει να προγραμματίσει κάποιος την πολιτική ασφαλείας του δικτύου βάση των χρηστών ή βάση της χρήσης.

### **7.3. Διαχείριση του ενεργού δικτύου**

Οι διαδικασίες διαχείρισης δικτύων απαιτούν τη συλλογή και σύγκριση δεδομένων, όπως η απαρίθμηση γεγονότων. Θα ήταν καλό να υπήρχε η δυνατότητα για φιλτράρισμα των πληροφοριών στο δίκτυο ώστε να απορρίπτονται γεγονότα που αναφέρονται αλλά δεν μας ενδιαφέρουν. Πιθανόν να είναι επιτεύξιμη η δυνατότητα ενσωμάτωσης ενός επιπέδου «υπευθυνότητας» στα ενεργά στοιχεία του δικτύου, π.χ. δρομολογητές, έτσι ώστε εισάγοντας ένα μικρό πρόγραμμα στα γειτονικά τους στοιχεία να ελέγχουν τον εαυτό τους. Με τον ίδιο τρόπο τα ενεργά δίκτυα μπορούν να εισάγουν μεθόδους για αναγνώριση σφαλμάτων έτσι ώστε να αυξάνεται η βιωσιμότητα του δικτύου μετά από κάποια καταστροφική εξωτερική επίδραση, π.χ. θεομηνία ή είσοδο κάποιου αναρμόδιου στο σύστημα.

### **7.4. Active Reliable Multicast (ARM)**

Η εφαρμογή αυτή αναπτύσσεται στο MIT και περιλαμβάνει δρομολογητές σε ένα δέντρο πολλαπλής εκπομπής (multicast tree) [24]. Οι δρομολογητές αυτοί επιδεικνύουν πολύ καλύτερη συμπεριφορά κατά την αποκατάσταση της λειτουργίας τους από τους συμβατικούς. Η βελτιωμένη αυτή συμπεριφορά οφείλεται σε τρεις λόγους.

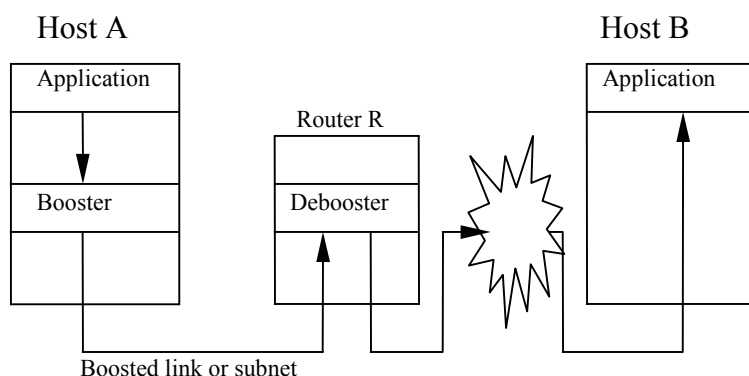
- Την καταστολή διπλότυπων πακέτων NACK, η οποία απαγορεύει την κατάρρευση των NACK.
- Την αποθήκευση στην προσωρινή μνήμη (cache) της καλύτερης από τις προσπάθειες (best-effort) επανεκπομπής πακέτων. Είναι ένα χαρακτηριστικό ιδιαίτερα χρήσιμο στην περίπτωση της αποκατάστασης.
- Τοπικοποιεί την εκπομπή, γεγονός που ελαττώνει το απαιτούμενο εύρος ζώνης για την επανεκπομπή των δεδομένων.

Για την υλοποίηση του ARM δεν επιβάλλεται η χρήση συγκεκριμένων δρομολογητών κι ούτε είναι απαραίτητο να είναι ενεργοί. Το αποτέλεσμα είναι ένα ανθεκτικό σύστημα, με ικανότητα κλιμακωτής ανάπτυξης και μικρές απαιτήσεις σε εύρος ζώνης.

### 7.5. Ενισχυτές πρωτοκόλλων (Protocol Boosters)

Είναι στοιχεία ενός πρωτοκόλλου τα οποία μπορούν είτε να εισαχθούν στο πρωτόκολλο είτε να αφαιρεθούν. Η ιδέα που κρύβεται από πίσω είναι η ικανότητά μας να χτίζουμε τα πρωτόκολλα δυναμικά [25]. Αυτό συντελεί τόσο στις επιδόσεις του πρωτοκόλλου όσο και στο ρυθμό ανάπτυξης των τεχνολογιών των δικτύων.

Οι ενισχυτές πρωτοκόλλων αναπτύχθηκαν για τον έλεγχο των λαθών και της συμφόρησης και τα ενεργά δίκτυα παρέχουν την ιδανική υποδομή για την εφαρμογή αυτή.



Σχήμα 4. Ο τρόπος επαύξησης της λειτουργίας ενός δικτύου με τη χρήση protocol booster

### 7.6. Ενεργός Έλεγχος Συμφόρησης (Active Congestion Control)

Λόγω της ευελιξίας των, τα ενεργά δίκτυα προσφέρουν πιθανές λύσεις για τη βελτίωση του ελέγχου της συμφόρησης. Το σύστημα αξιοποιεί ελέγχους κατάστασης για να μειώσει την καθυστέρηση όταν σηματοδοτείται συμφόρηση προς τα συστήματα που εκπέμπουν. Τα πακέτα του ACC περιέχουν 4-8 bytes για το χαρακτηρισμό της κατάστασης του σημείου που αναφέρει τη συμφόρηση, πιθανόν με αναφορά σε ένα παράθυρο συμφόρησης που ρυθμίζεται από τον πομπό. Όταν έχουμε συμφόρηση σε έναν δρομολογητή και συγκεκριμένα όταν γίνεται απόρριψη ενός πακέτου, τότε ο δρομολογητής :



- Ορίζει το μέγεθος του παραθύρου της συμφόρησης,
- Σβήνει τα πακέτα που δεν μπορούν να σταλούν με το νέο μέγεθος παραθύρου, και
- Ενημερώνει τον αποστολέα για το νέο μέγεθος του παραθύρου.

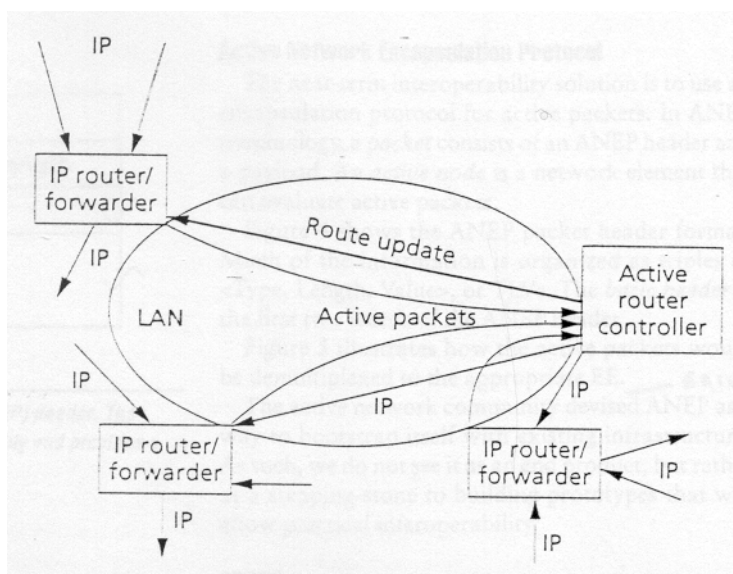
Οι κόμβοι πέρα από το σημείο της συμφόρησης, έχουν για την κίνηση μια εικόνα που παραπέμπει σε άμεση αντίδραση του αποστολέα. Το ACC αποδεικνύεται ιδιαίτερα ισχυρό στις περιπτώσεις δικτύων που έχουν μεγάλο γινόμενο (εύρους ζώνης x καθυστέρηση).

Ο Faber [26], προσάρμοσε το ACC στον έλεγχο συμφόρησης του TCP/IP θεωρώντας ότι αυτό λειτουργούσε πάνω σε IP δρομολογητές. Οι προσομοιώσεις έδειξαν βελτίωση στην διαμεταγωγή της τάξης του 18%, σε περιπτώσεις απότομων διακυμάνσεων της κίνησης.

### 7.7. Internet Enhancement

Ένας τρόπος για να εξαπλωθούν γρήγορα τα ενεργά δίκτυα είναι μέσω της διεύρυνσης του τρόπου λειτουργίας του Διαδικτύου.

Η έρευνα που γίνεται στο Πανεπιστήμιο της Πενσυλβάνια αντιμετωπίζει τη διεύρυνση του αυτή με τη μορφή ενεργών δρομολογητών ελεγκτών [27], που διαχειρίζονται μια ομάδα δρομολογητών επεμβαίνοντας σε αληθινό χρόνο στους πίνακες δρομολόγησης.



Σχήμα 5 Ένας ενεργός ελεγκτής δρομολογητών καθώς διαχειρίζεται μια ομάδα δρομολογητών.

## **8. Η επιτάχυνση της ανάπτυξης των υποδομών**

Καθώς τα παραδείγματα των πρωτοπόρων εφαρμογών του πίνακα 3 αποδεικνύουν, ήδη μέσα στα δίκτυα συμβαίνουν υπολογισμοί. Η μεγάλη απαίτηση για τέτοιου είδους εφαρμογές δείχνει ότι η πραγματικότητα των δικτύων πρέπει να προσανατολιστεί προς αντίστοιχες αρχιτεκτονικές.

Σε πιο βασικό επίπεδο, η διαδικασία ανάπτυξης των δικτύων είναι ώριμη για ανανέωση. Ο ρυθμός ανάπτυξης των δικτύων είναι ιδιαίτερα αργός και καθώς το διαδίκτυο αναπτύσσεται συνεχώς, είναι δύσκολο να επιταχυνθεί αλλά ακόμα και να παραμείνει σταθερός αυτός ο ρυθμός. Η ικανότητα της ανάπτυξης ενός δικτύου εξαρτάται από τους διασυνδεδεμένους σε αυτό κόμβους. Γενικά ένας τύπος δικτύου σήμερα, από την επίδειξη του προτύπου ως την πλήρη ανάπτυξη και εφαρμογή του χρειάζεται δέκα χρόνια. Η όλη διαδικασία περνά από το στάδιο της προτυποποίησης, της υιοθέτησης της τεχνολογίας από τους μεγάλους κατασκευαστές, την προμήθεια από τους χρήστες και την τελική εγκατάσταση. Σήμερα υπάρχουν κάποιες τεχνολογίες των οποίων καθυστερεί η ολοκλήρωση της υιοθέτησής τους. Για παράδειγμα η μετατροπή του πρωτοκόλλου IP στη νέα έκδοσή του Ipv6.

Το πρωτόκολλο IP αποτελεί τη βάση της επικοινωνίας στο Internet ορίζοντας μια συγκεκριμένη μορφή για τα πακέτα και συγκεκριμένη διευθυνσιοδότηση. Έτσι οι δρομολογητές ακόμη κι όταν προέρχονται από διαφορετικούς κατασκευαστές εκτελούν αντίστοιχες εργασίες. Ο τρόπος ανανέωσης του πρωτοκόλλου στη νέα έκδοσή του απαιτεί στην ουσία την αλλαγή όλης της υποδομής του δικτύου, αφού το IP είναι η βάση του δικτύου. Ένας τρόπος είναι με την εγκατάσταση υπερστρωμάτων όπως το MBone.

Σε αντίθεση, τα ενεργά δίκτυα δεν ακολουθούν ένα συγκεκριμένο τρόπο επίδρασης επί των πακέτων αλλά ορίζουν ότι κάθε κόμβος θα υποστηρίζει αντίστοιχα με τους άλλους υπολογιστικά μοντέλα. Με τον τρόπο αυτό αυξάνεται ο βαθμός αφαίρεσης στο δίκτυο επιτρέποντας στις εφαρμογές να προσαρμόσουν την επεξεργασία των μηνυμάτων έτσι ώστε να ταιριάζει στους στόχους τους.

Η ικανότητα να φορτώνει κάποιος νέες υπηρεσίες στην υποδομή του, θα φέρει τα δίκτυα σε μια κατάσταση όπου η ανάπτυξή τους θα οδηγείται από τους χρήστες. Στόχος είναι να ξεφύγει το δίκτυο από τη σημερινή κατάσταση, όπου το υλικό και το λογισμικό είναι στενά συνδεδεμένα.

## **9. Τα προγραμματιστικά περιβάλλοντα εκτέλεσης**

Τα ενεργά δίκτυα βασίζονται στην ικανότητα να προσθέτουν με ευκολία προγράμματα στην υποδομή του δικτύου. Αυτό κάνει σημαντική την επιλογή του προγραμματιστικού περιβάλλοντος και της γλώσσας του. Αυτά που θα περιγραφούν παρακάτω αποτελούν μερικές από τις ερευνητικές δουλειές κατασκευαστών και ινστιτούτων και έχουν σα στόχο τους την βελτιστοποίηση διάφορων προγραμματιστικών στόχων. Αυτό που προσφέρουν εξαρτάται από την κατάλληλη σύζευξη ανάμεσα στην ενεργή και παθητική κυκλοφορία, το μέγεθος της υπολογιστικής ισχύος στην υποδομή και το χρόνο ζωής του ενεργού κινητού κώδικα.

### **9.1. Active Network Transport System (ANTS)**

Το περιβάλλον αυτό εξελίσσεται στο MIT βασισμένο στη γλώσσα Java και στο περιβάλλον Java Virtual Machine. Τα χαρακτηριστικά της Java κάνουν το ANTS [3, 20, 21] κατάλληλο για την υποστήριξη πολλών εφαρμογών, γεγονός που σημαίνει ότι κερδίζει βάση στο χώρο των προγραμματιστών.

Το περιβάλλον αυτό αρχικοποιεί τους κόμβους του δικτύου, μεταφέροντας τον κινητό κώδικα μέσα στα πακέτα της πρώτης σύνδεσης. Τα επόμενα πακέτα χρησιμοποιούν τον κώδικα χωρίς να τον ξαναμεταφέρουν.

Στα χαμηλότερα επίπεδα της αρχιτεκτονικής του, το ANTS αποτελείται από κάπουλες που μεταφέρουν τόσο τον κώδικα όσο και δεδομένα. Παρόλα αυτά ο κώδικας είναι ιδιαίτερα συμπαγής γιατί περιορίζεται στο να αναφέρει το πρωτόκολλο που θα ενεργήσει επί των πακέτων. Το PAN (Practical Active Networks) μεταλλάσσει την αρχιτεκτονική του ANTS με τέτοιο τρόπο ώστε να συνδέεται πιο στενά με το λειτουργικό σύστημα του κόμβου και να υποστηρίζει παράλληλα κι άλλα συστήματα με κινητούς κώδικες.

### **9.2. Liquid Software**

Στο Πανεπιστήμιο της Αριζόνα ασχολούνται με το θέμα της διαφορετικής ταχύτητας της Java σε σύγκριση με άλλα στοιχεία ενός ενεργού δικτύου. Το λογισμικό τους περιέχει εργαλεία της Java που αντιμετωπίζουν το πρόβλημα της ταχύτητας με δύο τρόπους [22]. Πρώτα χρησιμοποιούν προγράμματα μετατροπής Java σε C. Δεύτερον χρησιμοποιούν γρήγορους μεταφραστές (compilers) στα σημεία

εκτέλεσης των προγραμμάτων. Αυτή η προσέγγιση διατηρεί την ευελιξία της Java ενώ αυξάνει την ταχύτητα.

### **9.3. Netscript**

Η έρευνα στο Πανεπιστήμιο Κολούμπια βασίζεται στη συγγραφή κώδικα σε Netscript για τη σύνθεση βασικών υπηρεσιών σε πιο σύνθετες [23]. Χρησιμοποιώντας σαν περιβάλλον εκτέλεσης διερμηνείς (interpreters) Netscript δημιουργείται μια ροή διαδικασιών επί της ροής των δεδομένων με σαφώς ορισμένες εισόδους και εξόδους. Η γλώσσα προγραμματισμού που χρησιμοποιείται είναι κι εδώ η Java.

Τα εργαλεία του περιβάλλοντος Netscript χρησιμοποιούνται για τη επίδειξη δημιουργίας δυναμικών «πύρινων τοίχων» (firewalls) και εξειδικευμένων συναρτήσεων δρομολόγησης.

### **9.4. Smartpackets**

Το περιβάλλον αυτό του BBN κινεί τον κώδικα μέσα σε IP πακέτα. Η διάρκεια ζωής του κώδικα σε κάθε κόμβο είναι τόση όσο χρειάζεται για να εκτελεστεί. Βασίζεται στις γλώσσες Sprocket και Spanner. Εξειδικεύεται στην επεξεργασία MIBs και αποτελεί ένα πολύ ευέλικτο και ασφαλές εργαλείο επισκόπησης και ελέγχου του δικτύου.

## **10. Λειτουργικά συστήματα των κόμβων**

Η διεπαφή του λειτουργικού συστήματος του κόμβου ορίζει τέσσερις βασικές λειτουργίες. Τη σύνδεση νημάτων (thread pool), την σύνδεση μνήμης (memory pool), τα κανάλια (channels) και τις ροές (flows). Οι τρεις πρώτες ενσωματώνουν τα τρία είδη πόρων του συστήματος. Η τέταρτη έχει να κάνει με τη συγκέντρωση του ελέγχου και τον προγραμματισμό των άλλων με τέτοιο τρόπο ώστε η όλη λειτουργία να προσομοιάζει τις απαιτήσεις μιας δικτυακής εφαρμογής. Οι διεπαφές των λειτουργικών συστημάτων των κόμβων υλοποιούνται στα λειτουργικά συστήματα Scout, Amp, Exokernel [17], SPIN [18]. Άλλες υλοποιήσεις είναι στο στάδιο του σχεδιασμού.

Η έννοια των ροών (flows) είναι από τις πιο ευαίσθητες στα ενεργά δίκτυα. Ξεκινάμε από τη θέση ότι θα ήταν ιδιαίτερα σημαντικό να δίνεται η δυνατότητα στις κάψουλες να αφήνουν πίσω τους κάποια πληροφορία όταν περνάνε από κάποιον κόμβο. Μπορεί κανείς να ανοίξει μια σύνδεση ορίζοντας σε μια κάψουλα να εκτελείται σε κάθε κόμβο που περνάει, αφήνοντας ταυτόχρονα μια μικρή πληροφορία για μια κατάσταση ελέγχου. Τα επόμενα πακέτα που ακολουθούν το ίδιο κανάλι μπορεί να περιέχουν κώδικα που εντοπίζει και εκτιμά την κατάσταση της σύνδεσης σε κάθε κόμβο.

Μια αντίστοιχη προσέγγιση είναι να χρησιμοποιηθούν ροές που είναι κάτι πιο απλό από μια σύνδεση. Κάθε κάψουλα μπορεί να περιέχει κώδικα που δοκιμάζει να εντοπίσει την κατάσταση ροής στους κόμβους που περνάει. Κατά κάποιον τρόπο οι κάψουλες ροής είναι πιο ανθεκτικές από αυτές που χρησιμοποιούνται στις συνδέσεις με την έννοια ότι η κατάσταση ροής είναι δεν είναι κρίσιμη για την εκτέλεση του περιεχομένου της κάψουλας. Αν μια κάψουλα ροής περάσει από ένα κόμβο που δεν έχει κάποια σχετική κατάσταση σύνδεσης, τότε εκτελεί το περιεχόμενό της και αφήνει πίσω της τα αποτελέσματα της εκτέλεσης για εξυπηρέτηση των επόμενων πακέτων. Από την άλλη, οι κόμβοι του δικτύου μπορούν να αντιμετωπίσουν τις καταστάσεις ροής σαν κάποια αδύναμη (soft) τιμή που έχει αποθηκευτεί προσωρινά και την οποία απορρίπτουν αν χρειαστούν πόρους. Υπό αυτό το σκεπτικό οι ροές είναι λιγότερο απαιτητικές ως προς την σταθερότητα λειτουργίας των κόμβων. Σε κάθε περίπτωση είναι καταστάσεις πολύ πιο ισχυρές από τα σημερινά συστήματα αφού αφήνουν πίσω τους προγράμματα αντί για στατικούς πίνακες.

Ο στόχος είναι να δημιουργηθούν νέα σχήματα πέρα από την έννοια της ροής και της σύνδεσης. Για παράδειγμα, οι κάψουλες μπορούν να είναι έτσι προγραμματισμένες που να είναι σε θέση να δώσουν «ραντεβού» σε κάποιον κόμβο. Η πρώτη που φτάνει στον κόμβο θα ενεργοποιεί μια κατάσταση του κόμβου και μετά θα πέφτει σε «λήθαργο» (sleep). Αφού φτάσει και η τελευταία θα μπορούν να πραγματοποιήσουν υπολογισμούς σε συνεργασία. Μια περίπτωση είναι εκείνη των πακέτων που μεταφέρουν αποσπάσματα κινούμενης εικόνας από διάφορους αισθητήρες.

Τέλος παρατηρούμε ότι κάψουλες ικανές να αλλάζουν καταστάσεις του κόμβου (π.χ. αποθήκευση) προσφέρουν έναν ενοποιημένο μηχανισμό διαχείρισης των λειτουργιών των κόμβων. Τόσο τα πρωτόκολλα δρομολόγησης, η ανανέωση των πινάκων όσο και η χρήση εντολών του SNMP, μπορούν να γίνουν με κάψουλες.

Σταθερότερες υπηρεσίες διαχείρισης των κόμβων μπορούν επίσης να υλοποιηθούν με αυτόν τον τρόπο με τη διαφορά ότι το μεταβατικό περιβάλλον στο οποίο εκτελούνται θα είναι πιο σταθερό, πιθανόν να επιτρέπεται η ανανέωσή του μόνο κατά την αρχικοποίηση (reset) του κόμβου.

## **11. Διαλειτουργικότητα (Interoperability)**

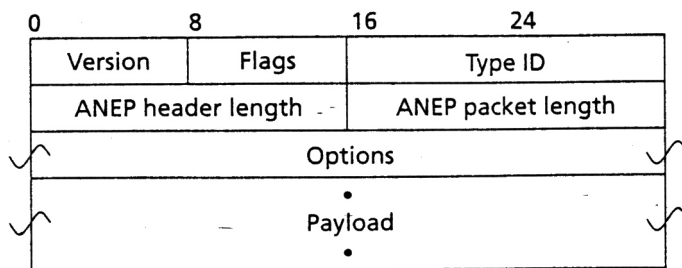
Δύο προσπάθειες γίνονται για να υποστηριχθεί η διαλειτουργικότητα των περιβαλλόντων επεξεργασίας. Το Πρωτόκολλο Ενθυλάκωσης Ενεργών Δικτύων (Active Network Encapsulation Protocol - ANEP), έχει σαν στόχο την επιτάχυνση των σχετικών πειραμάτων και τη χρήση των υφιστάμενων υποδομών σε διάφορες εφαρμογές. Η "Ραχοκοκαλιά" Ενεργών Δικτύων (Active Network Backbone - ABONE) χρησιμοποιείται για δοκιμές σε διάφορα περιβάλλοντα και τη χρήση τους σε εφαρμογές.

### **11.1. ANEP**

Η βραχυπρόθεσμη λύση για τη εφαρμογή ενεργών δικτύων είναι η χρήση ενός πρωτοκόλλου ενθυλάκωσης για ενεργά πακέτα. Σύμφωνα με την ορολογία του ANEP, ένα πακέτο αποτελείται από την επικεφαλίδα τύπου ANEP και την ωφέλιμη πληροφορία. Ως *ενεργός κόμβος* ορίζεται κάθε στοιχείο του δικτύου που μπορεί να διαχειριστεί ενεργά πακέτα.

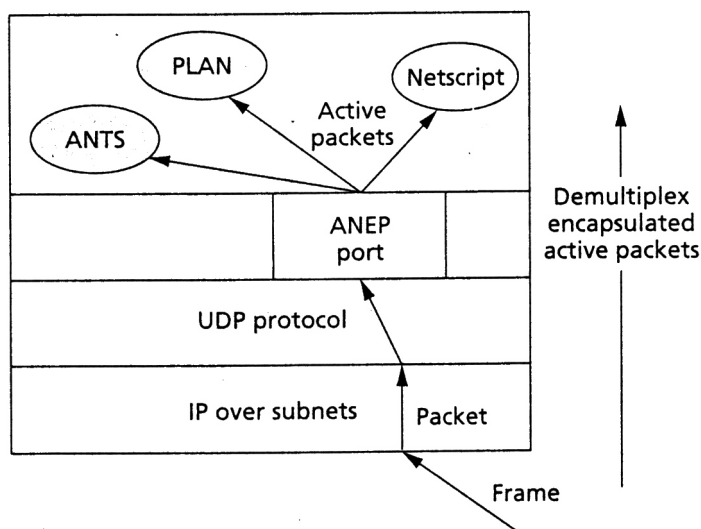
Στο παρακάτω σχήμα (Σχήμα 6) φαίνεται η μορφή της επικεφαλίδας. Η πληροφορία της είναι οργανωμένη σε τριάδες της μορφής <Τύπος, Μήκος, Τιμή> (<Type, Length, Value>, ή TLV). Η βασική επικεφαλίδα (basic header) είναι οι δύο πρώτες λέξεις της συνολικής επικεφαλίδας. Η επικεφαλίδα επιτρέπει στην ανεξάρτητη επιλογή των περιβαλλόντων επεξεργασίας και παρέχει τον τρόπο με τον οποίο μπορούμε να περάσουμε παραμέτρους που θα βοηθήσουν τα διάφορα περιβάλλοντα στη διερμηνεία των πακέτων.

Στο σχήμα 7 αναπαριστάται ο τρόπος με τον οποίο τα ενεργά πακέτα αποπλέκονται για να οδηγηθούν στα κατάλληλα υπολογιστικά περιβάλλοντα. Καθώς τα πακέτα φτάνουν στη διεπαφή (interface) του δικτύου, αποπλέκονται διαδοχικά στα επίπεδα IP, UDP και ANEP, για να οδηγηθούν τελικά στο κατάλληλο περιβάλλον επεξεργασίας.



Σχήμα 6. Η μορφή της επικεφαλίδας του ANEP.

Οι άνθρωποι που ασχολούνται με τα ενεργά δίκτυα επινόησαν το ANEP σαν έναν τρόπο για να προχωρήσουν την έρευνα με βάση την υπάρχουσα υποδομή. Με αυτό το σκεπτικό δεν αντιμετωπίζεται σαν τελικό προϊόν αλλά σαν ένα σκαλοπάτι στην ανάπτυξη προτύπων που θα επιτρέπουν την ενδολειτουργία.



Σχήμα 7 Ο τρόπος με τον οποίον αποπλέκονται τα πακέτα προς τα διάφορα περιβάλλοντα επεξεργασίας μέσω του πρωτοκόλλου ANEP.

## 11.2. ABONE

Ένα πρότυπο με στόχο τη διαλειτουργικότητα είναι το ABONE. Στόχος του είναι να υποστηρίξει την ανάπτυξη δικτύων με περισσότερους από 1000 κόμβους. Σήμερα οι 100 κόμβοι του είναι μηχανές Unix στις οποίες τρέχει το πρόγραμμα anetd. Αυτό διαχειρίζεται ενεργούς κόμβους δικτύου και την εκκίνηση του επεξεργαστικού περιβάλλοντος. Προς το παρόν το ABONE είναι επίπεδο αν και οι δημιουργοί του

έχουν προτείνει μια ιεραρχική δομή. Στα άκρα του δικτύου βρίσκονται οι φιλοξενητές (hosts) με τοπική διαχείριση και καθώς προχωράμε προς την αρχή του δικτύου παρεμβάλλονται δρομολογητές. Η ιεραρχική δομή αναμένεται να επιταχύνει την ανάπτυξη του δικτύου προς το στόχο των 1000 κόμβων. Το ABONE υποστηρίζει τα περιβάλλοντα ANTS, Netscript και PLAN/Alien.

Για τα επόμενα χρόνια έχει προγραμματισθεί η ανάπτυξη εφαρμογών πάνω από το ABONE ώστε να επιδειχθούν και να αναπτυχθούν η αξιοπιστία, η ευελιξία, η ασφάλεια, η ευχρηστία και οι υψηλές αποδόσεις. Στο χρόνο αυτό θα επιδιωχθεί ο περιορισμός των περιβαλλόντων επεξεργασίας (EE) ώστε να καινούργια χαρακτηριστικά και υπηρεσίες να δουλεύουν πάνω από αυτά που θα απομείνουν.

## **12. Συμπεράσματα**

Τα ενεργά δίκτυα αποτελούν μια μεγάλη αλλαγή στα πρότυπα της συμπεριφοράς των δικτύων. Δημιουργούν περιβάλλοντα εντός του δικτύου που επιτρέπουν στα πακέτα που κινούνται μέσα σε ένα δίκτυο να μεταφέρουν τις υπηρεσίες που χρειάζονται και να τις υπολογίζουν εντός τις δικτυακής υποδομής. Στο χώρο αυτό γίνεται σήμερα αρκετή έρευνα που αρχίζει να αποδίδει αποτελέσματα. Οι αρχιτεκτονικές που προτείνονται οδηγούν στην ανάπτυξη διαδραστικών στοιχείων στα δίκτυα ενώ όλο και πιο ισχυρά επεξεργαστικά περιβάλλοντα δημιουργούνται. Αναλύονται ταυτόχρονα τα θέματα ασφαλείας που προκύπτουν και διερευνώνται οι πιθανές λύσεις. Η ιδέα ωριμάζει καθημερινά και οι άνθρωποι που ασχολούνται με το πεδίο αυτό υπόσχονται να φέρουν τα μοναδικά χαρακτηριστικά των ενεργών δικτύων σε εφαρμογή στον δικτυακό κόσμο. Άλλωστε ήδη σήμερα υπάρχουν εφαρμογές που στηρίζονται σε υπολογισμούς και επεξεργασία πακέτων εντός του δικτύου. Ήρθε η ώρα να εξερευνήσουμε καινούργια αρχιτεκτονικά μοντέλα στα δίκτυα με στόχο μια νέα γενιά δικτύων που θα είναι πιο ευέλικτα και θα επιταχύνουν το ρυθμό ανάπτυξης και ανανέωσης των υποδομών.



## **ΒΙΒΛΙΟΓΡΑΦΙΑ**

[1] D. L. Tennenhouse et al., “A survey on Active Network Research”, IEEE Communications, Jan. 1997, pp. 80-96.

Γενικό άρθρο από μία ομάδα που εργάζεται εντατικά στο θέμα των ενεργών δικτύων. Αποτελεί πολλή καλή πρώτη αναφορά αφού συνοψίζει το ερευνητικό έργο όλων των ομάδων – εργαστηρίων που ασχολούνται με το θέμα.

[2] J. M. Smith et al., “Activating Networks: A Progress Report”, IEEE Computer, Apr. 1999, pp. 32-41.

Άρθρο της ίδιας ομάδας, όπου περιγράφονται γενικά τα διάφορα περιβάλλοντα εφαρμογής των ενεργών δικτύων, τα λειτουργικά συστήματα των κόμβων του δικτύου, θέματα ασφάλειας και μερικές εφαρμογές.

[3] D. Wetherall, U. Legedza, and J. Guttag., “Introducing New Internet Services: Why and How”. IEEE Network Spl. Iss. Active and Programmable Networks, Jul. 1998.

Διερευνάται το αποτέλεσμα των ενεργών δικτύων στις διάφορες εφαρμογές καθώς και στον τρόπο ανάπτυξης και εφαρμογής τους. Παρουσιάζεται ιδιαίτερα η αρχιτεκτονική ANTS που προσθέτει επεκτασιμότητα στη επίπεδο δικτύου και επιτρέπει την κλιμακωτή ανάπτυξη ενεργών κόμβων στο Internet.

[4] D. Tennenhouse, and D. J. Wetherhall. “Towards an Active Network Architecture”. Multimedia Computing and Networking (MMCN '96). 1996. San Jose, CA.

[5] S. E. Deering. Host Extensions for IP multicasting. Request For Comments 1112, Aug 1989.

Είναι το κείμενο όπου ορίζεται ο τρόπος με τον οποίο καθορίζονται οι προεκτάσεις που απαιτούνται για την εγκατάσταση ενός οικοδεσπότη (host) με τέτοιο τρόπο ώστε να υποστηρίζει multicasting μέσω του IP πρωτοκόλλου.

[6] C. Perkins, Ed. “IP Mobility Support”. Request For Comments 2002, Oct 1996.

Το κείμενο αυτό καθορίζει τις προσαυξήσεις που είναι απαραίτητες στο πρωτόκολλο IP, ώστε να επιτραπεί η διαφανής δρομολόγηση των IP datagrams στο Internet και να καταστεί δυνατή η σύνδεση κινητών τερματικών στο Δίκτυο.

[7] Kleinrock, L. "Nomadic Computing". Intl. Conf. On Mobile Computing and Networking. 1995. Berkeley, CA: ACM.

Περιγράφονται οι διαδικασίες μέσω των οποίων ένας δρομολογητής που παρεμβάλλεται ανάμεσα στο δίκτυο και τον τελικό χρήστη ελέγχει και προσαρμόζεται ανάλογα στον τρόπο με τον οποίο ο χρήστης συνδέεται στο δίκτυο και παρέχει διαφορετικά επίπεδα υπηρεσιών.

[8] Balakrishnan, H. et al. "Improving TCP/IP Performance over Wireless Networks". Intl. Conf. On Mobile Computing and Networking. 1995. Berkeley, CA.

Αναφέρεται στην υποστήριξη υπηρεσιών ασύρματης πρόσβασης σε ενσύρματα δίκτυα μέσω ενεργοποίησης της διαδικασίας TCP snooping και διατήρησης των πληροφοριών για κάθε ξεχωριστή σύνδεση στους ασυρματικούς σταθμούς βάσης

[9] Amir, E., S. McCanne, and H. Zhang. "An Application Level Video Gateway". ACM Multimedia '95. 1995. San Fransisco, CA.

Μια εφαρμογή πύλης για μεταφορά εφαρμογών κινούμενης εικόνας ανάμεσα σε διαφορετικά περιβάλλοντα πρωτοκόλλων.

[10] M. Fry, and A. Ghosh, "Application Level Active Networking", Computer Networks, April 1999, Vol.31, Issue 7, pp. 655-667

[11] N. Borenstein, "E-mail with a mind of its own: The Safe-Tcl Language for Enabled Mail", IFIP International Conf., 1994, Barcelona, Spain.

[12] J. Gosling, and H. McGilton, "The Java Language Environment: A White Paper", 1995, Sun Microsystems: Mountain View, CA.

[13] A. Adl-Tabatabai, et al. "Efficient and Language-Independent Mobile Programs". ACM SIGPLAN Conf. On Programming Language Design and Implementation (PLDI '96). 1996, Philadelphia, PA: ACM.

[14] G. Nacula, P. Lee. "Safe Kernel Extensions Without Run-Time Checking". Second Symp. On Operating System Design and Implementation (OSDI '96). 1996. Seattle, WA

[15] D. Wetherhall, and D. Tennenhouse, "The Active IP Option". 7<sup>th</sup> ACM SIGOPS European Workshop. 1996. Connemara. Ireland

[16] D. Mosberger, and D. Tennenhouse. "Making Paths Explicit in the Scout Operating System". 1996. Technical Report 96-05, Dept. of Computer Science, University of Arizona.

[17] D. R. Engler, M. F. Kaashoek, and J O'Toole, Jr. "Exokernel: An Operating System Architecture for Application Level Resource Management". 15<sup>th</sup> ACM Symp. On Operating Systems Principles. 1995.

[18] B. Bershad et al. "Extensibility, Safety and Performance in the SPIN Operating System". 15<sup>th</sup> ACM Symp. On Operating Systems Principles. 1995.

[19] D. R. Engler, W. C. Hsieh, and M. F. Kaashoek. "C: A Language for High-Level, Efficient, and Machine-Independent Dynamic Code Generation". 23<sup>rd</sup> Annual ACM Symp. On Principles of Programming Languages (POPL'96). 1996. St. Petersburg, FL.

[20] D. Wetherhall, J. Guttag, and D. Tennenhouse. "ANTS: Network Services Without the Red Tape". IEEE 1999. pp. 42-48.

[21] <http://www.sds.lcs.mit.edu/activeware/> Η κεντρική σελίδα της ομάδας του MIT που ασχολείται με τα ενεργά δίκτυα. Από εδώ ξεκινάνε συνδέσεις για άλλα ιδρύματα που ασχολούνται με το θέμα ενώ περιέχει πολλά άρθρα της ερευνητικής ομάδας και RFCs.

[22] <ftp://ftp.cs.arizona.edu/xkernel/Papers/tr96-11.ps>.

[23] [http://www.es.columbia.edu/dcc/netscript/body\\_index.html](http://www.es.columbia.edu/dcc/netscript/body_index.html)

[24] L. Lehman, S.J. Garland, and D. L. Tennenhouse, "Active Reliable Multicast," Proc. INFOCOM 98, IEEE CS Press, Los Alamitos, Calif., 1998, pp.581-589.

[25] D. C. Feldmeier et al., "Protocol Boosters," IEEE J. Selected Areas in Comm., Apr. 1998, pp. 437-444.

[26] T. Faber, "ACC: Using Active Networking to Enhance Feedback Congestion Control Mechanisms," IEEE Network, May/June 1998, pp 61-65.

[27] <http://www.cis.upenn.edu/~switchware>

[28] D. L. Tennenhouse, S. J. Garland, L. Shrira, and M.F. Kaashoek, "From Internet to Active Net".

[29] Walid S. Dabbous, "High Performance protocol architecture". Computer Networks & ISDN Systems 29, 1997. pp 735-744.

[30] [www.sciencedirect.com](http://www.sciencedirect.com). Δικτυακός τόπος όπου υπάρχουν άρθρα σε διάφορα επιστημονικά θέματα. Πρέπει να γραφτεί κάποιος συνδρομητής για να χρησιμοποιήσει τις υπηρεσίες του.

[31] Y. Yemini, S. DaSilva, and D. Florissi, "The Netscript Project". DARPA An. Workshop 1997.

[32] K.L. Calvert, ed. "Architectural Framework for Active Networks version 1.0". Active Network Working Group, RFC Draft, Univ. of Kentucky. July 27, 1999.

[33] [www.cabletron.com](http://www.cabletron.com) ο δικτυακός τόπος της Cabletron Systems. Καλή αναφορά με πολλά white papers σε δικτυακά θέματα (VPN, Voice over Packet, IP Multicast, ...)