



# Τεχνολογίες Εφαρμογών Ιστού

Δίκτυα, Αρχιτεκτονικές, Internet, Πρωτόκολλο HTTP

# Επισκοπήση

- Δίκτυα Υπολογιστών
- Αρχιτεκτονική Δικτύων
- Internet
- TCP/IP Πρωτόκολλα
- HTTP



# Δίκτυα Υπολογιστών

# Βασικές Εννοιες

- Δίκτυο Η/Υ ή Δίκτυο
  - Μία συλλογή από διασυνδεδεμένες συσκευές με δυνατότητες υπολογισμού και διαχείρισης δεδομένων/πληροφορίας
- Οι διασυνδεδεμένες συσκευές
  - Μπορούν να είναι γεωγραφικά διεσπαρμένες
  - Διασυνδέονται μεταδίδοντας πληροφορίες μέσω κάποιου μέσου (π.χ. καλώδια, αέρας)
  - Παράγουν, στέλνουν, προωθούν και λαμβάνουν πληροφορίες
- Παραδείγματα Συσκευών
  - Η/Υ
  - Εκτυπωτές
  - Modems
  - Μονάδες αποθήκευσης
- Παραδείγματα δικτύων
  - Δίκτυο τηλεπικοινωνιών (σταθερών, κινητών)
  - Δορυφορική τηλεόραση
  - Εσωτερικό και εξωτερικό δίκτυο μιας επιχείρησης (Intranet και Extranet)
  - Διαδίκτυο (Internet)

# Δομη δικτυων

- Κεντρικοί υπολογιστές (hosts)
  - Προσωπικοί υπολογιστές
  - Ισχυρά υπολογιστικά συστήματα που λειτουργούν ως εξυπηρετητές (servers)
- Περιφερειακές συσκευές δικτύου (network peripherals), όπως εκτυπωτές, modem, κ.λ.π.
- Ζεύξεις ή Γραμμές Μετάδοσης (Transmission lines, Links)
  - Φυσικά μονοπάτια επικοινωνίας διαμέσου των οποίων μεταφέρονται δεδομένα
- Στοιχεία Μεταγωγής (Switching Elements)
  - Ενδιάμεσες συσκευές για τη μεταφορά και τη δρομολόγηση των δεδομένων από το ένα δίκτυο στο άλλο

# Κατηγοριοποίηση Δικτύων

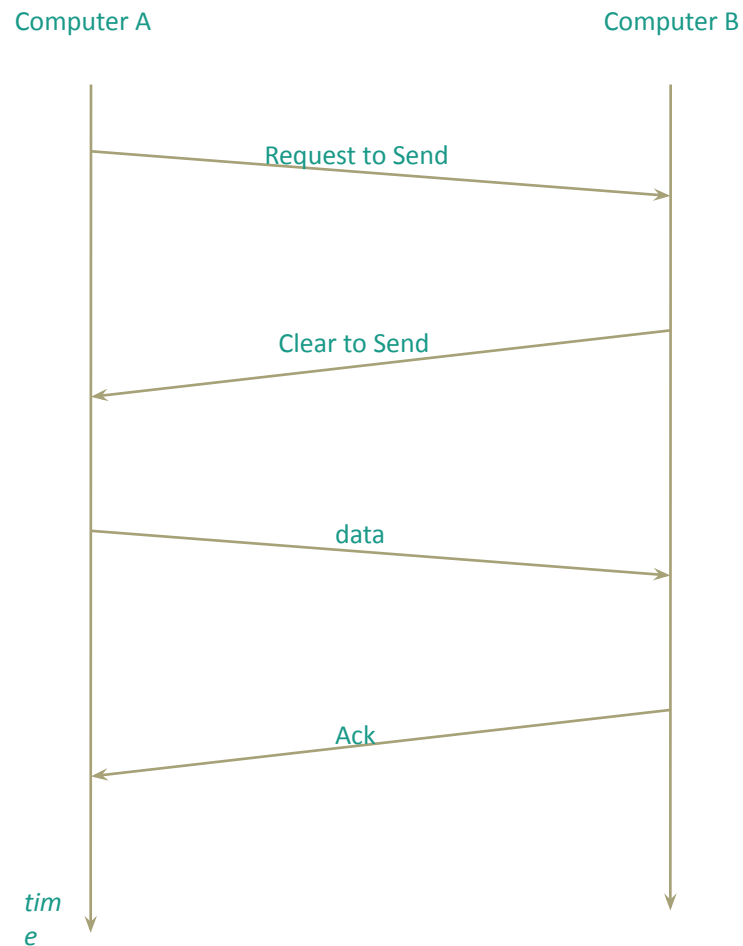
- Με βάση την τεχνολογία μετάδοσης
  - Αναλογικά
  - Ψηφιακά
- Με βάση την τεχνολογία μεταγωγής
  - Μεταγωγής κυκλώματος
  - Μεταγωγής πακέτου
- Με βάση τη γεωγραφική έκταση
  - Τοπικά
  - Μητροπολιτικά Δίκτυα
  - Ευρείας Περιοχής
- Με βάση την τοπολογία
  - Πλέγμα
  - Δίαυλος
  - Δακτύλιος
  - Αστέρας
  - Δέντρο
- Με βάση το σχήμα διευθυνσιοδότησης
  - Εκπομπής (broadcast networks)
  - Σημείου προς σημείο (point-to-point)
- Με βάση το μέσο μετάδοσης
  - Ασύρματα
  - Ενσύρματα

# Πρωτοκολλα

- Πρωτόκολλο ανθρώπινης επικοινωνίας:
  - Όχι αυστηρό συντακτικό
  - Αντίληψη, Κατανόηση, Νόημα, Συσχέτιση
  - Μη ντετερμινιστική αντίδραση
- Πρωτόκολλο δικτυακής επικοινωνίας:
  - Αυστηρή σύνταξη και κανόνες
  - Η μηχανή αναγνωρίζει μόνο κάτι για το οποίο έχει προγραμματιστεί
  - Ντετερμινιστική απάντηση
- Για να επικοινωνήσει ο άνθρωπος με τη μηχανή πρέπει να χρησιμοποιήσει το πρωτόκολλο της μηχανής

# Πρωτοκολλα Υπολογιστων

- Σύνολο κανόνων για την επικοινωνία Η/Υ μέσω ενός δικτύου
- Ελέγχουν ή πραγματοποιούν τη σύνδεση, επικοινωνία και τη μεταφορά δεδομένων
- Υλοποιούνται από υλικό ή λογισμικό ή συνδυασμούς
- Παραδείγματα πρωτοκόλλων:
  - IP (Internet Protocol)
  - UDP (User Datagram Protocol)
  - TCP (Transmission Control Protocol)
  - DHCP (Dynamic Host Configuration Protocol)
  - HTTP (Hypertext Transfer Protocol)
  - FTP (File Transfer Protocol)
  - Telnet (Telnet Remote Protocol)
  - SSH (Secure Shell Remote Protocol)
  - POP3 (Post Office Protocol 3)
  - SMTP (Simple Mail Transfer Protocol)
  - IMAP (Internet Message Access Protocol)
  - SOAP (Simple Object Access Protocol)







# Αρχιτεκτονική Δικτύων

# Αρχιτεκτονική Δικτύων

- Τα περισσότερα δίκτυα οργανώνονται σε μια σειρά από **στρώματα (layers)** ή **επίπεδα (levels)**.
  - ευκολότερη σχεδίαση, ανάπτυξη νέων υπηρεσιών, προϊόντων και συστημάτων
- Το κάθε στρώμα χτίζεται πάνω στο κατώτερό του.
- Ο αριθμός των στρωμάτων, το όνομα τους, τα περιεχόμενά τους και η λειτουργία τους διαφέρουν από δίκτυο σε δίκτυο.
- Ένα σύνολο από στρώματα και πρωτόκολλα αποκαλείται **αρχιτεκτονική δικτύου**.
- Τα στρώματα μπορούν να προσφέρουν δύο τύπους υπηρεσιών προς τα αμέσως υψηλότερα τους στρώματα:
  - **υπηρεσίες με σύνδεση** (connection – oriented services)
  - **υπηρεσίες χωρίς σύνδεση** (connectionless services).

# Οφέλη Διαστρωματωμενων (Layered) Αρχιτεκτονικων

- Αλλαγές: Οι αλλαγές στην υλοποίηση μπορούν να περιοριστούν σε ένα επίπεδο αντί να επηρεάσουν όλη την αρχιτεκτονική
- Σχεδιασμός: Οι σχεδιαστές επικεντρώνονται σε ένα επίπεδο χωρίς να ανησυχούν για τις επιπτώσεις σε άλλα
- Κατανόηση: Η προσέγγιση με επίπεδα γίνεται ευκολότερα κατανοητή
- Εύρεση και Επίλυση λαθών: Τα λάθη μπορούν να απομονωθούν σε υλοποιήσεις λειτουργιών συγκεκριμένων επιπέδων
- Πρότυπα (Standards): Πιθανώς το σημαντικότερο όφελος. (Προ-)Τυποποιώντας τις αρχιτεκτονικές των επιπέδων καθώς και τις ενδιάμεσες διεπαφές πετυχαίνουμε διαλειτουργικότητα που είναι απαραίτητη σε ετερογενή περιβάλλοντα όπως αυτά που “ζουν” στο Internet

# Υπηρεσίες Με Συνδεση

- Απαιτείται η εγκατάσταση μιας (φυσικής ή λογικής) σύνδεσης μεταξύ πομπού και δέκτη πριν αρχίσει η μεταφορά της πληροφορίας, καθώς και η απεγκατάστασή της μόλις λήξει η μεταφορά.
- Λειτουργεί σαν σωλήνας: ο πομπός σπρώχνει την πληροφορία από το ένα άκρο και ο δέκτης τα λαμβάνει με την ίδια σειρά στο άλλο άκρο

# Υπηρεσίες Χωρίς Συνδεση

- Δεν απαιτούν την εγκατάσταση μιας σύνδεσης αλλά μεταφέρουν τα πακέτα ανεξάρτητα το ένα από το άλλο.
- Τα πακέτα φτάνουν στον προορισμό τους με λάθος σειρά, κάποια πακέτα μπορεί να φτάσουν και να περιέχουν λάθη και κάποια άλλα μπορεί να χαθούν.

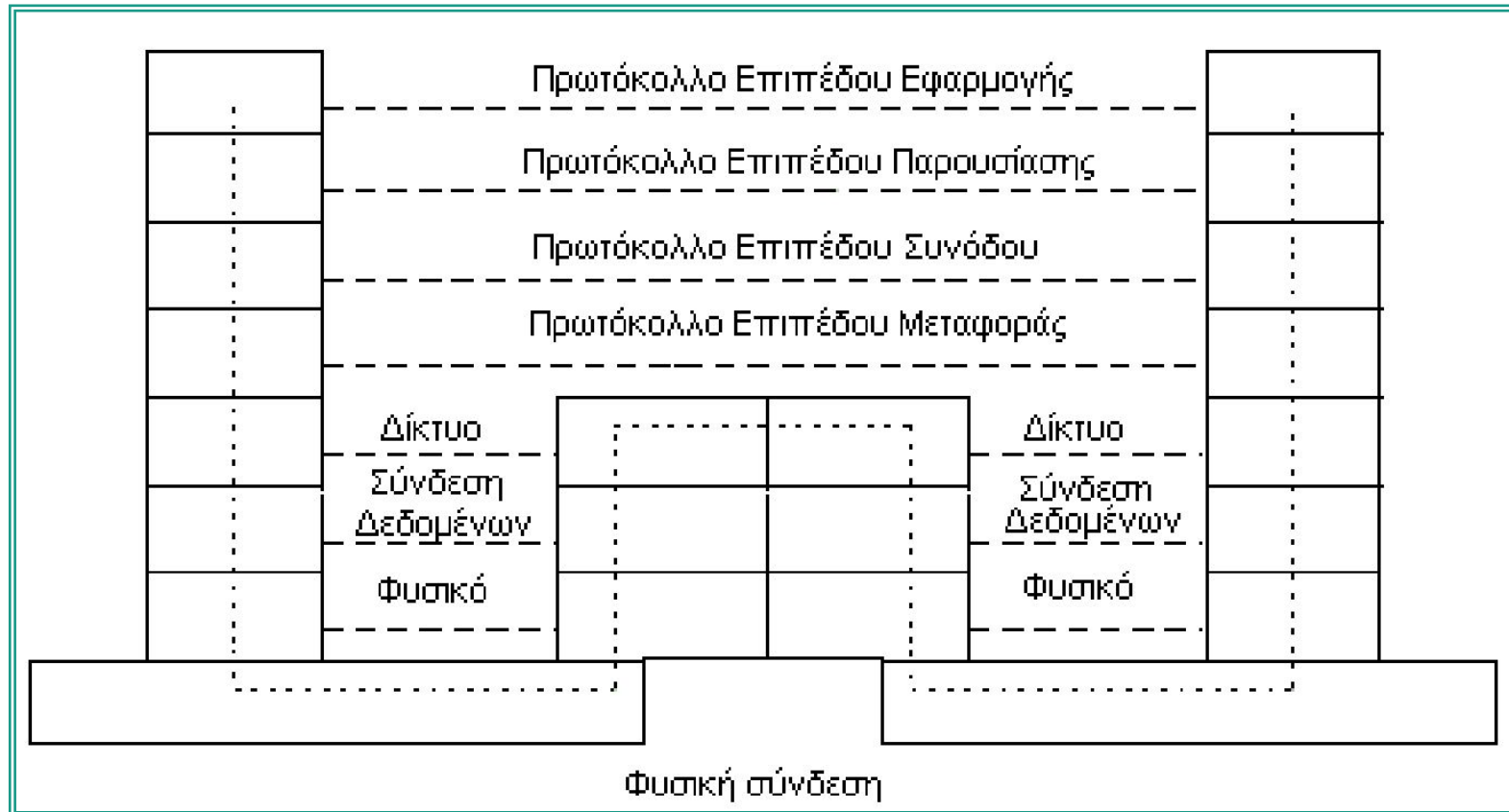
# Το Μοντελο Αναφοράς OSI

- Η προσπάθεια για την μοντελοποίηση των δικτύων οδήγησε στην ανάπτυξη του προτύπου OSI (Open Systems Interconnection - Διασύνδεση Ανοιχτών Συστημάτων) από τον οργανισμό ISO.
- Η βασική ιδέα που οδήγησε στην ανάπτυξη του OSI είναι η ομαδοποίηση και η ταξινόμηση όλων των λειτουργιών που εκτελούνται στα δίκτυα σε ένα και μοναδικό μοντέλο.
  - Οι λειτουργίες ταξινομούνται ανάλογα με τη φύση και το είδος της εξυπηρέτησης που προσφέρουν.
- Το OSI αποτελείται από 7 επίπεδα.

# Το Μοντελο Αναφοράς OSI

- Το κάθε ένα από τα 6 κατώτερα επίπεδα προσφέρει τις υπηρεσίες του στο ανώτερό του.
- Οι υπηρεσίες και οι πληροφορίες παρέχονται στο ανώτερο επίπεδο μέσω διασυνδέσεων (interfaces) ανάμεσα στα επίπεδα.
- Τα τρία κατώτερα επίπεδα ασχολούνται με:
  - τη μετάδοση των bits,
  - τη μετάδοση των πακέτων μέσα από μία ζεύξη και
  - τη μετάδοση των πακέτων από άκρο σε άκρο.
- Τα ανώτερα επίπεδα δομούν υπηρεσίες επικοινωνίας για τις εφαρμογές των χρηστών.

# Το Μοντελο Αναφοράς OSI





# Φυσικο Επίπεδο (Physical Layer)

- Πρόκειται για το κατώτατο επίπεδο
- Ασχολείται με τις φυσικές ιδιότητες του δικτύου
- Το φυσικό επίπεδο επιφορτίζεται με:
  - την μετατροπή των bits σε ηλεκτρικά ή οπτικά σήματα,
  - το ρυθμό μετάδοσης των δεδομένων
  - το συγχρονισμό πομπού και δέκτη.

# Ζητήματα Που Επιλυονται

- Πόσα volts πρέπει να χρησιμοποιηθούν για την αναπαράσταση του 1 και του 0
- Πόσα microseconds διαρκεί ένα bit,
- Αν η μετάδοση μπορεί να γίνει και προς τις δύο κατευθύνσεις
- Πώς η αρχική σύνδεση αποκαθίσταται και πώς απορρίπτεται
- Πόσα pins έχει το NIC και πόσα pins χρειάζεται για τη σύνδεση

# Επίπεδο Συνδεσης Δεδομενων (Data Link Layer)

- Το επίπεδο αυτό εξασφαλίζει την αξιόπιστη μεταφορά των δεδομένων, είτε στο δέκτη (περίπτωση απλού τοπικού δικτύου) είτε σε μια κομβική συσκευή, διαμέσου μιας γραμμής.
- Παρέχει υπηρεσίες με σύνδεση είτε χωρίς σύνδεση, καθώς και λειτουργίες όπως έλεγχος ροής και λαθών.
- Κάποιες επιπλέον λειτουργίες του επιπέδου αυτού είναι:
  - Πλαισίωση: Διάσπαση σε πιο μικρά και εύχρηστα πακέτα (frames).
  - Φυσική Διευθυνσιοδότηση: Στο κάθε πλαίσιο αναγράφονται οι διευθύνσεις των κόμβων αφετηρίας και προορισμού του. Οι κόμβοι, που συνδέονται στην ίδια γραμμή ή σε διαφορετικές γραμμές, μπορεί να είναι ο πομπός, ο δέκτης ή κάποιες ενδιάμεσες συσκευές (π.χ. Κάποιος δρομολογητής).
- Έλεγχος Πρόσβασης: Για συσκευές που συνδέονται στην ίδια γραμμή ελέγχεται ποια από όλες έχει την πρόσβαση στην γραμμή, κάθε χρονική στιγμή.

# Ζητήματα Που Επιλυονται

- Δημιουργία και αναγνώριση των ορίων των πλαισίων – συνήθως προσθέτοντας ειδικές ακολουθίες bits στην αρχή και το τέλος του πλαισίου
- Επίλυση προβλημάτων που προκαλούνται από απώλεια, αποστολή διπλών ή λανθασμένων πακέτων
- Έλεγχος ενός γρήγορου πομπού ο οποίος μεταδίδει δεδομένα προς ένα αργό δέκτη
- Έλεγχος μέσου μεταξύ πλαισίων επιβεβαίωσης λήψης (ack) και πλαισίων δεδομένων εάν η γραμμή είναι δικατευθυντική

# Επίπεδο Δικτύου (Network Layer)

- Εξασφαλίζει τη σύνδεση των τερματικών συσκευών (πομπού και δέκτη) είτε σε ένα δίκτυο είτε στο διαδίκτυο και επιφορτίζεται με τη μεταφορά και δρομολόγηση των πακέτων δεδομένων, διαμέσου των διάφορων κόμβων (π.χ. δρομολογητές και πύλες) που παρεμβάλλονται.
- Οι κυριότερες υπηρεσίες που προσφέρει είναι:
  - Λογική Διευθυνσιοδότηση: Πρόκειται για την παροχή διευθύνσεων ώστε να είναι δυνατός ο προσδιορισμός και η ταυτοποίηση του πομπού και του δέκτη μέσα σε ένα διαδίκτυακό ή σε ένα ευρύ δικτυακό περιβάλλον.
  - Δρομολόγηση: Είναι ο μηχανισμός με τον οποίο γίνεται η προώθηση των πακέτων από τον πομπό στον δέκτη, διαμέσου των κομβικών συσκευών του διαδικτύου ή ενός δικτύου WAN.
  - Έλεγχος Ροής – Συμφόρησης: Ελέγχεται από το σύστημα αν κάποια ενδιαμέση συσκευή είναι συνωστισμένη με πακέτα, έτσι ώστε να δρομολογηθούν κάποια πακέτα από άλλο μονοπάτι ή να ελαττωθεί ο ρυθμός αποστολής κάποιων πακέτων.

# Ζητήματα Που Επιλυονται

- Υλοποίηση του μηχανισμού δρομολόγησης
- Έλεγχος συμφόρησης
- Accounting
- Διασύνδεση ετερογενών δικτύων

# Επιπεδο Μεταφοράς (Transport Layer)

- Το επίπεδο μεταφοράς επιβλέπει τη μεταφορά των δεδομένων από τον πομπό στον δέκτη. Οι υπηρεσίες που προσφέρει αποσκοπούν στην ακρίβεια και στην καλή ποιότητα της μεταφοράς.
- Οι κυριότερες υπηρεσίες του επιπέδου είναι οι ακόλουθες:
  - Διευθυνσιοδότηση Διεργασιών: πρόκειται κατά κάποιον τρόπο για την ταυτότητα των προγραμμάτων που επικοινωνούν και ανταλλάσσουν τις πληροφορίες. Τα προγράμματα αυτά τρέχουν στις τερματικές συσκευές.
  - Τμηματοποίηση & Επανασύνδεση Δεδομένων: Οι πληροφορίες που στέλνονται διασπώνται σε μικρότερα τμήματα από τον πομπό και συνδέονται ξανά στον δέκτη. Αυτό διευκολύνει την καλύτερη διαχείριση των δεδομένων.

# Επίπεδο Μεταφοράς (Transport Layer)

- Έλεγχος Σύνδεσης: Υπάρχουν δύο δυνατότητες, η προσφορά υπηρεσιών χωρίς σύνδεση και οι υπηρεσίες με σύνδεση. Στη δεύτερη περίπτωση δημιουργείται μια σύνδεση ανάμεσα σε πομπό και δέκτη, την οποία ακολουθούν όλα τα πακέτα, ώστε να υπάρχει εγγύηση για την αξιοπιστία στην μεταφορά, αφού το επίπεδο μεταφοράς μπορεί να ελέγξει καλύτερα τη ροή, την αλληλουχία και τα σφάλματα.
- Αξιόπιστη Παράδοση: Αυτό σημαίνει ότι γίνεται έλεγχος της ροής των πακέτων που φθάνουν, έλεγχος για την ύπαρξη σφαλμάτων, ή για τυχόν διπλές παραδόσεις ίδιων πακέτων, ή έλεγχος για απώλειες πακέτων κ.α.



# Ζητήματα Που Επιλυονται

- Υλοποίηση μίας σύνδεσης μεταφοράς μέσω πολλών συνδέσεων δικτύου εάν το επίπεδο συνόδου απαιτεί υψηλές επιδόσεις ή την πολύπλεξη πολλών συνδέσεων μεταφοράς στην ίδια σύνδεση δικτύου εάν οι συνδέσεις δικτύου είναι ακριβές
- Παροχή διαφορετικών τύπων υπηρεσιών για το επίπεδο συνόδου
- Υλοποίηση ενός τύπου ελέγχου ροής

# Επίπεδο Συνοδου (Session Layer)

- Το επίπεδο αυτό χρησιμοποιείται στον έλεγχο του διαλόγου που ανοίγει ανάμεσα στις πλευρές που επικοινωνούν.
- Παρέχει λειτουργίες δημιουργίας, διατήρησης και συγχρονισμού του διαλογικού περιβάλλοντος.
- Ως επίπεδο πολλές φορές είναι προαιρετικό επειδή οι λειτουργίες του καλύπτονται από τις λειτουργίες των πιο κάτω επιπέδων (σύνδεσης δεδομένων, δικτύου και μεταφοράς).
- Από τις πιο σημαντικές δραστηριότητες του επιπέδου αυτού είναι η δυνατότητα να κρατάει κάποια σημεία ελέγχου κατά την αποστολή μεγάλων αρχείων, με αποτέλεσμα να διασφαλίζει την μη απώλεια αρχείων σε περιπτώσεις σφαλμάτων ή προσωρινής διακοπής της σύνδεσης.

# Ζητήματα Που Επιλυονται

- Έλεγχος διαλόγου: μπορεί να επιτρέψει στα δεδομένα να κινηθούν και προς δύο κατευθύνσεις ταυτόχρονα ή προς μία κατεύθυνση. Εάν τα δεδομένα μπορούν να πάνε προς μία κατεύθυνση, το επίπεδο συνόδου μπορεί να βοηθήσει στην ανάθεση μετάδοσης εκ περιτροπής.
- Παροχή σκυτάλης για τη διαχείριση της εκτέλεση λειτουργιών οι οποίες δεν επιτρέπονται ταυτόχρονα από τις δύο πλευρές. Μόνο η πλευρά η οποία έχει στον έλεγχο της τη σκυτάλη μπορεί να εκτελέσει κάποια συγκεκριμένη ενέργεια
- Συγχρονισμός και ανοχή σε σφάλματα με χρήση σημείων ελέγχου

# Επίπεδο Παρουσιασης (Presentation Layer)

- Ασχολείται με τη μορφή που στέλνονται τα δεδομένα και αν θα πρέπει αυτή να αλλάζει ώστε να είναι κατανοητά και από τις δύο πλευρές.
- Το επίπεδο αυτό επιφορτίζεται με θέματα όπως:
  - τη σύνταξη των δεδομένων,
  - αν αυτά στέλνονται σε συμπιεσμένη μορφή,
  - αν είναι κωδικοποιημένα ή κρυπτογραφημένα
  - διάφορα άλλα μέτρα ασφαλείας.

# Ζητήματα Που Επιλυονται

- Συντακτικό και semantics της πληροφορίας που μεταδίδεται
- Ποια κωδικοποίηση χρησιμοποιείται στην επικοινωνία

# Επίπεδο Εφαρμογής (Application Layer)

- Το επίπεδο αυτό παρέχει στο χρήστη ένα σύστημα διεπαφής (interface) για την εκτέλεση διαφόρων εφαρμογών (π.χ. e-mail και telnet).
- Κάποιες από τις υπηρεσίες που προσφέρει το επίπεδο εφαρμογής είναι:
  - Συμφωνία για τους μηχανισμούς ασφάλειας (μηχανισμοί κρυπτογράφησης, κωδικοποίησης και αυθεντικοποίησης)
  - Προσδιορισμός της ταυτότητας αυτών που επιθυμούν να επικοινωνήσουν, με βάση το όνομα ή τη διεύθυνσή τους.
  - Συμφωνία για την υπευθυνότητα στους μηχανισμούς ανάκαμψης και χειρισμού σφαλμάτων.
  - Προσδιορισμός του εάν είναι διαθέσιμος αυτός με τον οποίο επιθυμεί ο χρήστης να επικοινωνήσει.
  - Ανάκτηση της άδειας για επικοινωνία.

# Ζητήματα Που Επιλυονται

- Διαλειτουργικότητα μεταξύ τερματικών συσκευών και εφαρμογών (π.χ. browsers, λειτουργικά συστήματα, συστήματα διαχείρισης αρχείων)

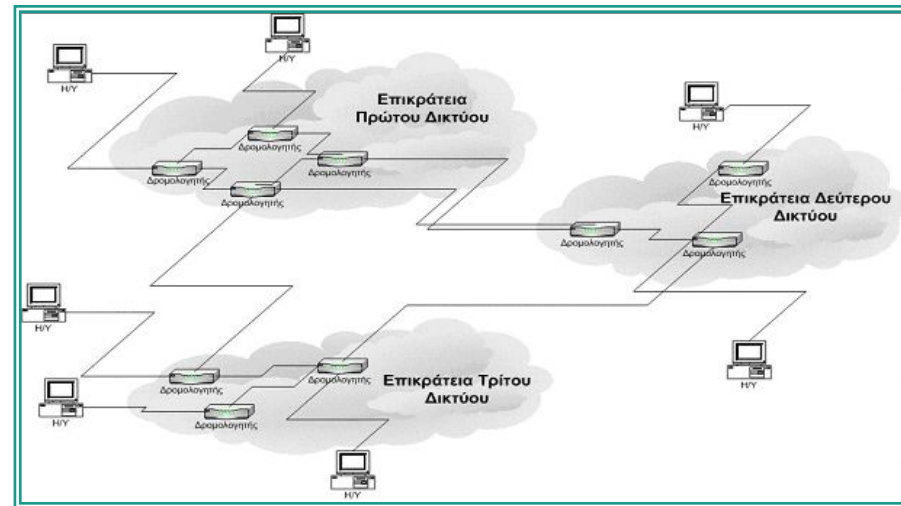


# Internet



# Ορισμος

- Γενικά, διαδίκτυο (internet ή internetwork) είναι ένα σύνολο από δύο ή περισσότερα δίκτυα (LAN, MAN, WAN) που συνδέονται μεταξύ τους με διάφορα στοιχεία μεταγωγής
- Πλέον το **Διαδίκτυο (Internet)** είναι ένα WAN που καλύπτει ολόκληρο τον πλανήτη
- Ξεκίνησε σαν ένα ερευνητικό πρόγραμμα της κυβέρνησης των ΗΠΑ, επιχορηγούμενο από την Advanced Research Projects Agency (ARPA). Αρχικά ονομάζονταν ARPANET
- Το Internet αναπτύχθηκε γρήγορα στις δεκαετίες των 80-90
- Το 1983 ήταν συνδεδεμένοι στο Internet λιγότεροι από 600 Η/Υ, ενώ τώρα υπάρχουν πολλά εκατομμύρια



# Το Internet Σήμερα

- Με τον όρο Διαδίκτυο εννοούμε ένα παγκόσμιο πληροφοριακό σύστημα που:
  - Είναι λογικά διασυνδεδεμένο μέσω ενός μοναδικού παγκόσμιου χώρου διευθύνσεων.
  - Είναι ικανό να υποστηρίξει επικοινωνίες χρησιμοποιώντας πρωτόκολλα της αρχιτεκτονικής TCP/IP ή/και άλλα πρωτόκολλα.
  - Παρέχει, χρησιμοποιεί ή καθιστά προσπελάσιμες, είτε δημόσια είτε ιδιωτικά, υπηρεσίες υψηλού επιπέδου βασισμένες στις επικοινωνίες.

# Απο Τι Αποτελείται Το Διαδίκτυο;

- Υπολογιστικά Συστήματα
  - Από Υπερ-υπολογιστές μέχρι ένα κινητό νέας γενιάς
- Τηλεπικοινωνιακές γραμμές
  - Οπτικές ίνες, χαλκός, μικροκύματα
- Δρομολογητές
  - Διευθυνσιοδότηση, αποσύνθεση/σύνθεση πακέτων, δρομολόγηση, έλεγχος ροής και λαθών, ασφάλεια, ...
- Πρωτόκολλα
  - TCP, IP, HTTP, FTP, SMTP, ...

# Το Μοντέλο Πελάτη - Εξυπηρετητή

- Το μοντέλο πελάτη – εξυπηρετητή είναι κυρίαρχο στο INTERNET.
- Καθορίζει τον τρόπο λειτουργίας και εκτέλεσης εφαρμογών (π.χ. FTP, HTTP, SMTP, TELNET) στο Διαδίκτυο.
- Όλες οι μηχανές στο Διαδίκτυο που ζητάνε τις υπηρεσίες άλλων αποτελούν τους πελάτες (Clients), ενώ αυτές που προσφέρουν τις υπηρεσίες τους στους πελάτες είναι οι εξυπηρετητές (Servers).



# Το μοντελο Πελατη - Εξυπηρετητη

- Στον εξυπηρετητή όλες οι εφαρμογές – υπηρεσίες που μπορούν να προσφερθούν, πραγματοποιούνται από ένα διαφορετικό τμήμα εξυπηρέτησης, το οποίο διακρίνεται από τα άλλα με τη βοήθεια των αριθμών θυρών (Port Numbers)
- Υπάρχει, δηλαδή, ένας διαφορετικός αριθμός θύρας για κάθε μορφή εφαρμογής (π.χ. η θύρα με αριθμό 80 του Server στο galaxy.hua.gr προσφέρει υπηρεσίες HTTP)
- Ο πελάτης μπορεί να στείλει περισσότερες από μία αιτήσεις ή ο εξυπηρετητής να στείλει μια σειρά απαντήσεων – χωρίς απαραίτητα να έχει προηγηθεί κάποια αίτηση
- Άλλοι τύποι μηνυμάτων που μπορεί να σταλούν είναι οι αιτήσεις ύπαρξης του εξυπηρετητή, οι επιβεβαιώσεις και οι απαντήσεις μη ορθής αίτησης

# Επικοινωνία στο Διαδίκτυο

- Οι υπολογιστές στο Διαδίκτυο βρίσκονται σε σταθερή επαφή/επικοινωνία μεταξύ τους με μια σειρά από τηλεφωνικές γραμμές, καλώδια, δορυφορικά συστήματα κλπ που επιτρέπουν την γρήγορη μεταφορά ηλεκτρονικής πληροφορίας
- Έχει συμφωνηθεί ένας κοινός τρόπος επικοινωνίας που επιτρέπει στους υπολογιστές να ανταλλάσσουν μεγάλες ποσότητες πληροφορίας ακόμα και αν το κάθε μηχάνημα εσωτερικά μιλάει διαφορετική γλώσσα υπολογιστών
- Αυτή η επικοινωνία είναι εφικτή εξαιτίας μιας σειράς πρωτοκόλλων
- Πρωτόκολλο: Σύνολο συμφωνημένων κανόνων για το πώς ακριβώς θα μεταφέρεται η πληροφορία από υπολογιστή σε υπολογιστή στο Διαδίκτυο
- Η αρχιτεκτονική TCP/IP είναι ένα σύνολο από πρωτόκολλα που επιτρέπει σε διαφορετικούς υπολογιστές με διαφορετικά λειτουργικά συστήματα να επικοινωνούν μεταξύ τους



# TCP/IP

# TCP/IP

- Το λογισμικό που διαχειρίζεται την επικοινωνία στο Διαδίκτυο είναι συνέπεια μιας συλλογής πρωτοκόλλων που λέγεται TCP/IP
- Το Internet Protocol (IP) καθορίζει το περιεχόμενο και διευθυνσιοδότηση της πληροφορίας καθώς αυτή μεταφέρεται
- Το Transmission Control Protocol (TCP) υπαγορεύει πως κατακερματίζονται και ανασυνθέτονται τα πακέτα και χειρίζεται χαμένη πληροφορία
- Δρομολόγηση πακέτων μέσω δρομολογητών (routers)
- Στο Internet, κάθε συνδεδεμένος Η/Υ αναγνωρίζεται από μια μοναδική διεθνή διεύθυνση, την IP address



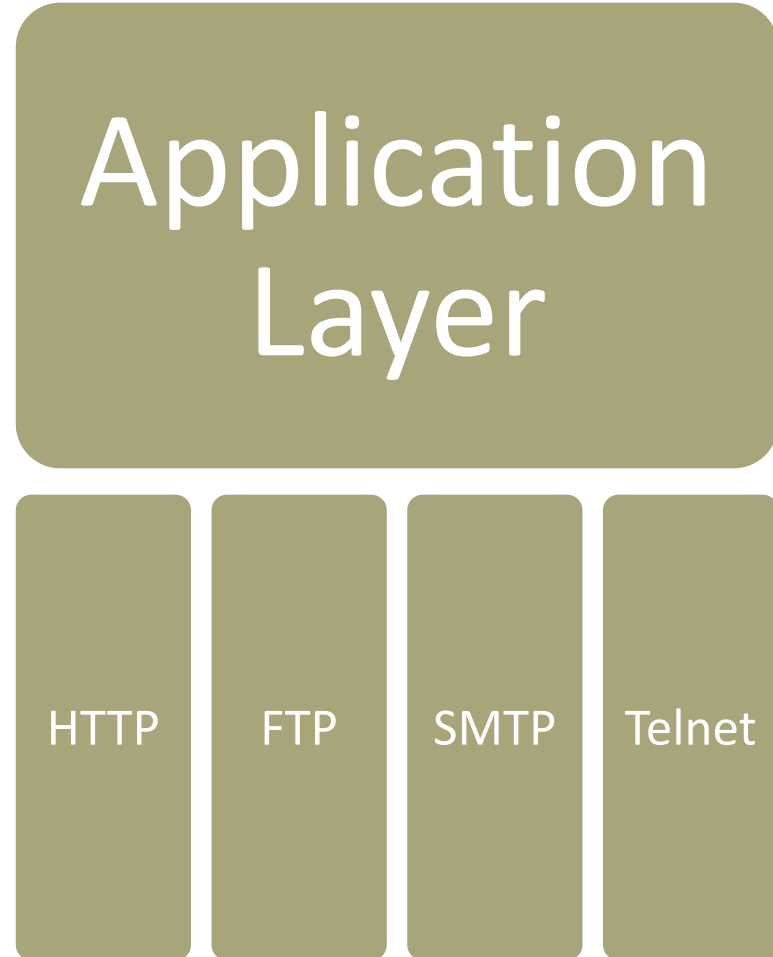
# Η Ιστορία Του TCP/IP

- Το αμερικάνικο υπουργείο άμυνας σχεδίασε το μοντέλο TCP/IP με στόχο την κατασκευή ενός δικτύου που θα επιβίωνε κάτω από οποιεσδήποτε συνθήκες.
- Είναι μια απλοποιημένη μορφή του μοντέλου OSI (το TCP/IP προηγήθηκε του OSI)
- Κάποια επίπεδα έχουν τα ίδια ονόματα με το μοντέλο OSI



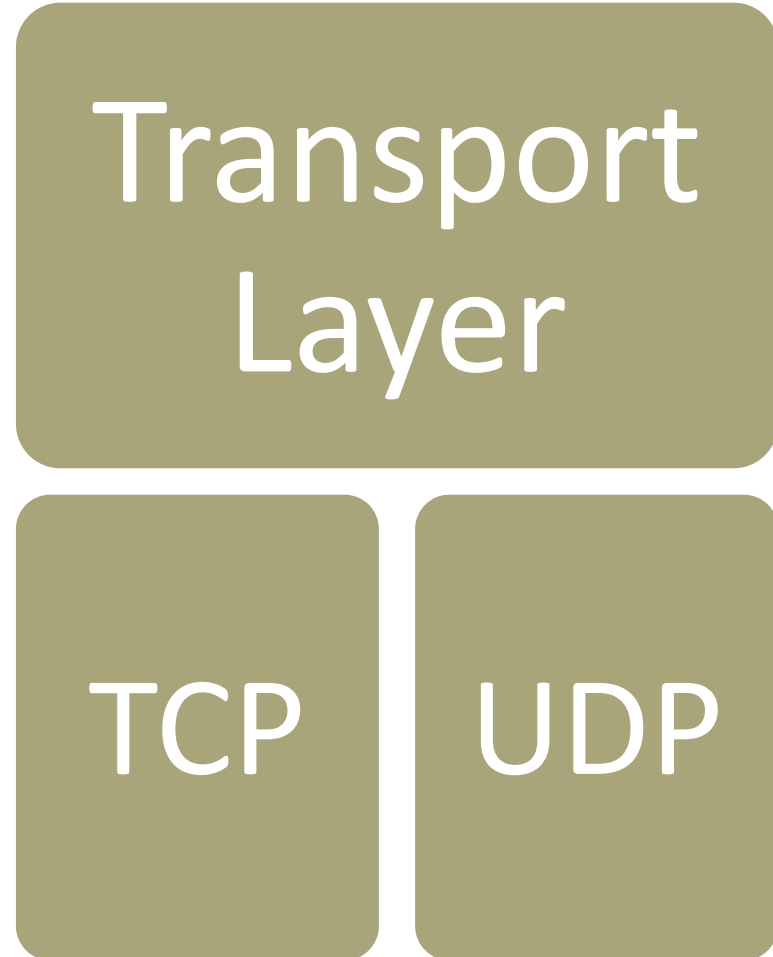
# Επιπεδο Εφαρμογής

- Χειρίζεται τα πρωτόκολλα 'υψηλού επιπέδου' δηλαδή θέματα αναπαράστασης, μορφοποίησης, κωδικοποίησης, αλληλεπίδρασης με το χρήστη.
- Το TCP/IP φροντίζει για τη σωστή μορφοποίηση/σύνταξη των δεδομένων πριν αυτά περάσουν στο επόμενο επίπεδο.
- Παραδείγματα:
  - Telnet: Παρέχει τη δυνατότητα απομακρυσμένης σύνδεσης σε κάποιο υπολογιστικό σύστημα, δίνοντας τις περισσότερες από τις δυνατότητες που έχει ένας τοπικός χρήστης (χρήστης κονσόλας)
  - File Transfer Protocol (FTP): Μεταφορά αρχείων από/προς κάποιος εξυπηρετητή
  - Hypertext Transfer Protocol (HTTP): Μεταφορά υπερκειμένου (αλλά και άλλων αρχείων) μέσα στον Παγκόσμιο Ιστό (WWW)



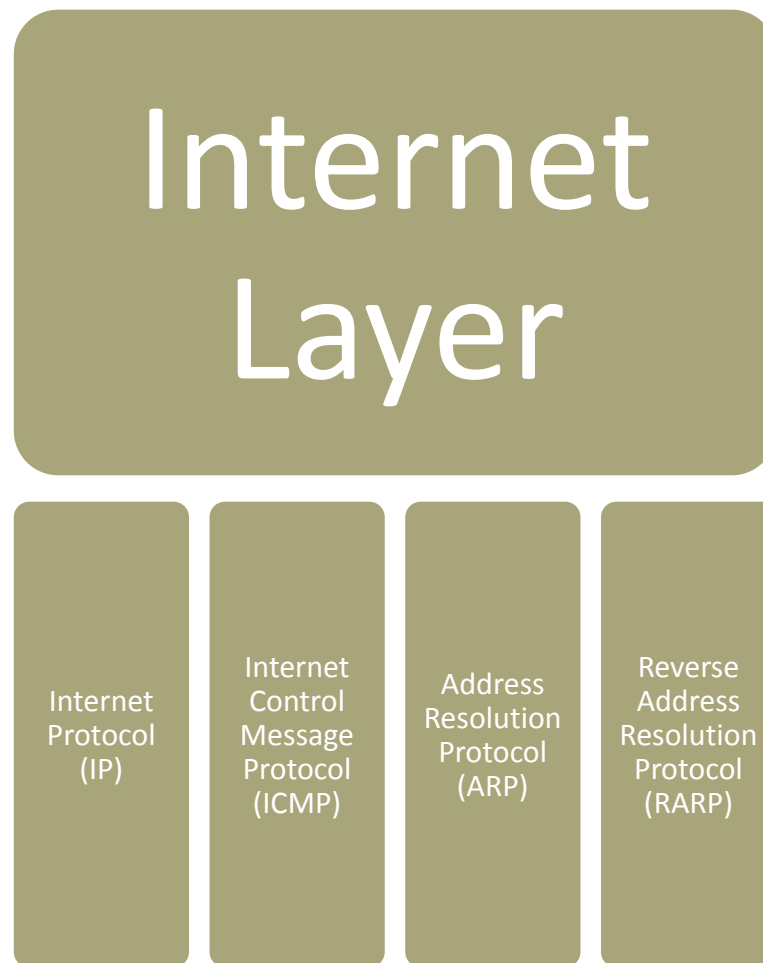
# Πρωτοκόλλα Επιπέδου Μεταφοράς (Transport Layer)

- Ο κύριος στόχος του πρωτοκόλλου TCP είναι να επιβεβαιώνεται η αξιόπιστη αποστολή και λήψη δεδομένων
- Πρωτόκολλα επιπέδου μεταφοράς:
  - TCP: Connection-oriented communication
  - UDP: Connectionless communication. Δεν υπάρχει έλεγχος αν έφτασαν τα δεδομένα στον προορισμό τους.



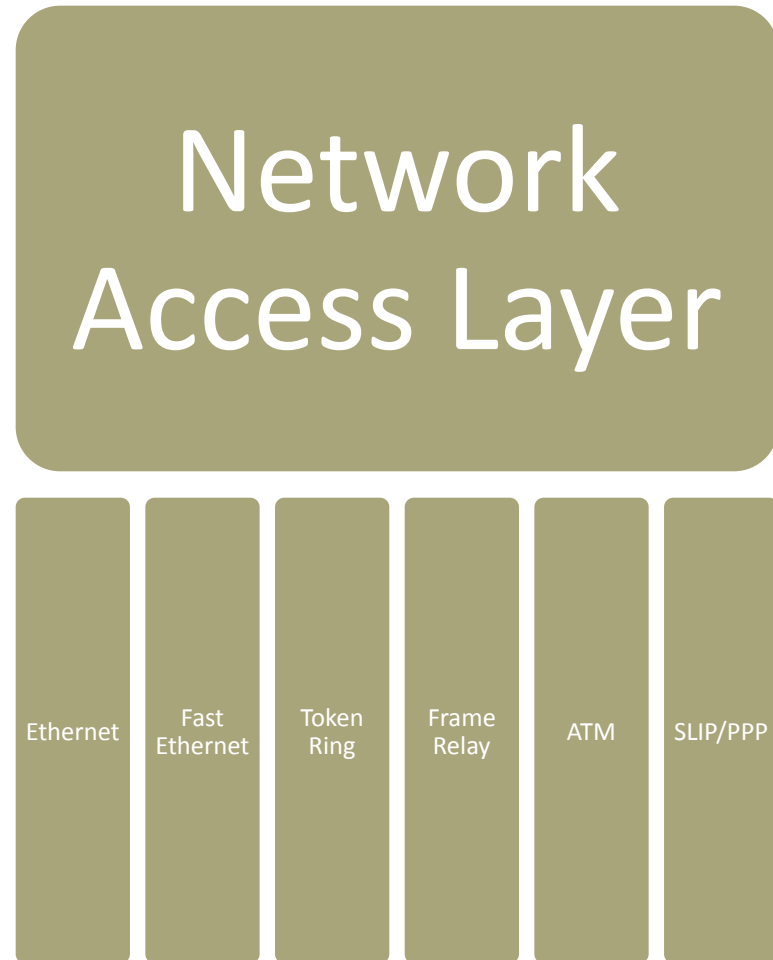
# Επιπεδο Διαδικτυου (Internet Layer)

- Ο σκοπός του επιπέδου Internet είναι η αποστολή και σωστή δρομολόγηση των πακέτων δεδομένων προς τον προορισμό τους, ανεξάρτητα από τη διαδρομή που θα ακολουθήσουν.
- Πρωτόκολλα επιπέδου Internet:
  - Internet Protocol (IP)
  - Internet Control Message Protocol (ICMP)
  - Address Resolution Protocol (ARP)
  - Reverse Address Resolution Protocol (RARP)

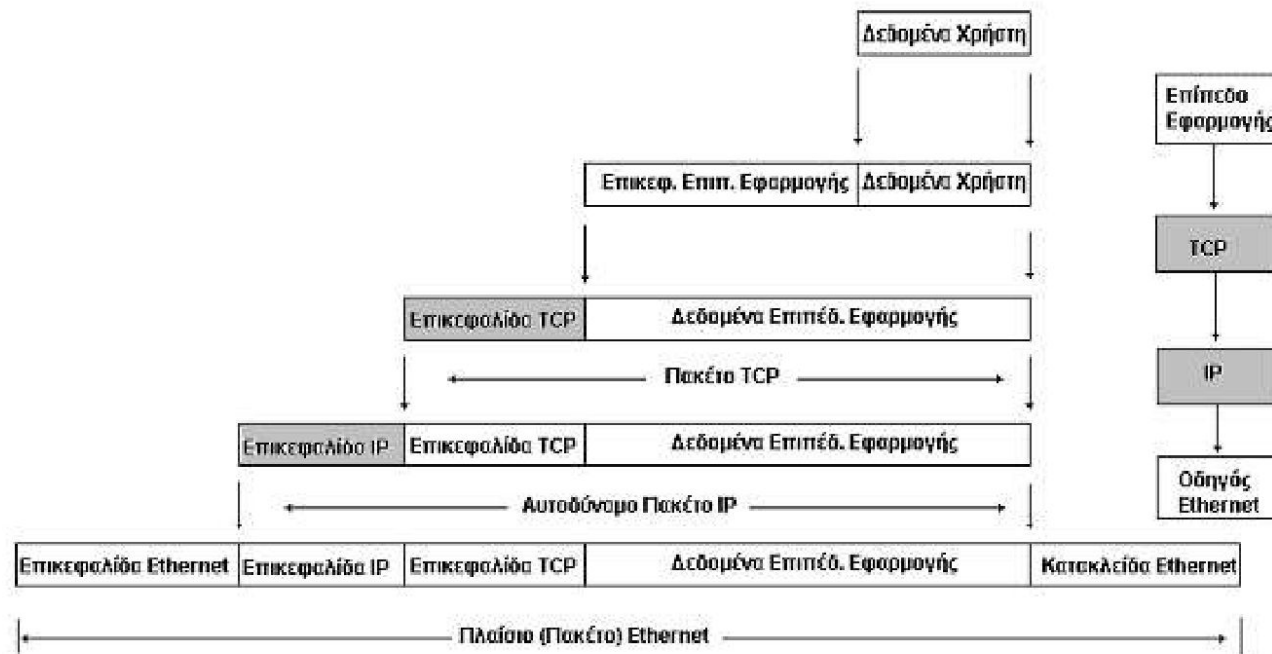


# Επίπεδο Δικτυακής Προσβασης (Network Access Layer)

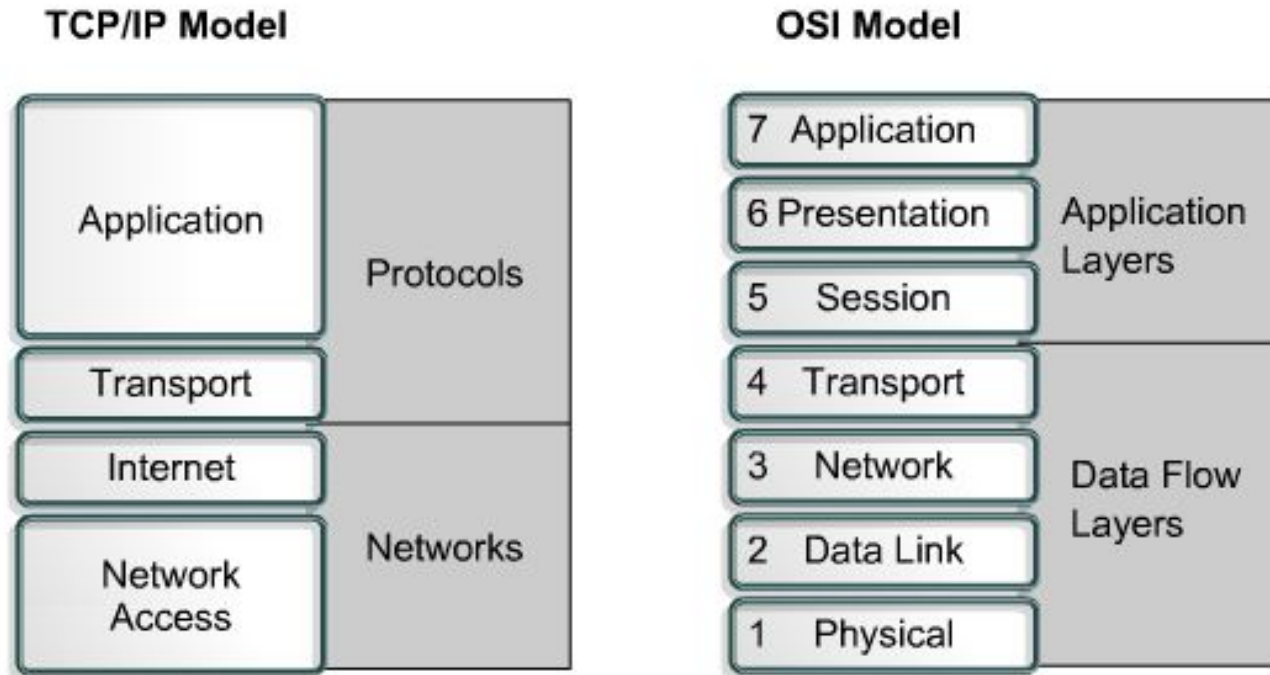
- Το επίπεδο Δικτυακής Πρόσβασης κάνει ότι χρειάζεται έτσι ώστε ένα πακέτο IP να μεταδοθεί πάνω από το φυσικό μέσο. (π.χ. Μετατροπή σε αναλογικό ηλεκτρικό σήμα, κατακερματισμός σε κελιά (cells))
- Αντιστοιχεί σε τεχνολογίες LAN και WAN.



# Μεταφορά και σταδια πακετοποίησης πληροφορίας



# Συγκριση Των Μοντελων TCP/IP Και OSI



# Συγκριση Των Μοντελων TCP/IP και OSI

- Ομοιότητες

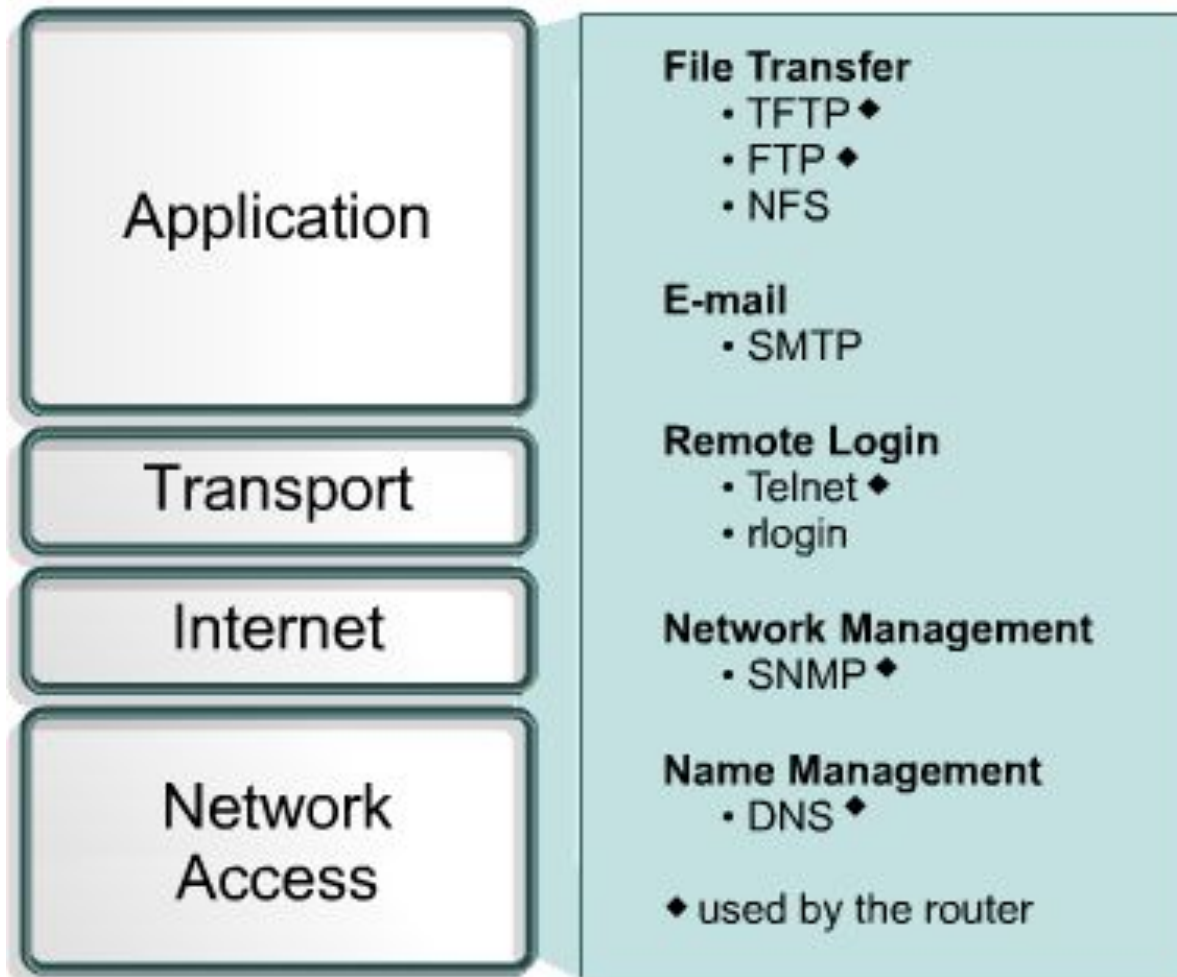
- Βασίζονται στη λογική των επιπέδων
- Έχουν συγκρίσιμα επίπεδα εφαρμογής, μεταφοράς και δικτύου
- Ασχολούνται με το 'πακέτο' και όχι με το 'κύκλωμα'

- Διαφορές

- Το TCP/IP φαίνεται απλούστερο γιατί συνδυάζει πολλές υπηρεσίες σε λιγότερα επίπεδα. (στην πραγματικότητα αυτό δυσκολεύει το σχεδιασμό των πρωτοκόλλων)
- Το TCP/IP (π.χ. Υλοποίηση Transport Layer με UDP) *ΔΕΝ εγγυάται την αξιόπιστη παράδοση* των δεδομένων στον προορισμό τους, αντίθετα με το OSI όπου η αξιόπιστη παράδοση είναι κανόνας.



# TCP/IP – Επιπεδο Εφαρμογής

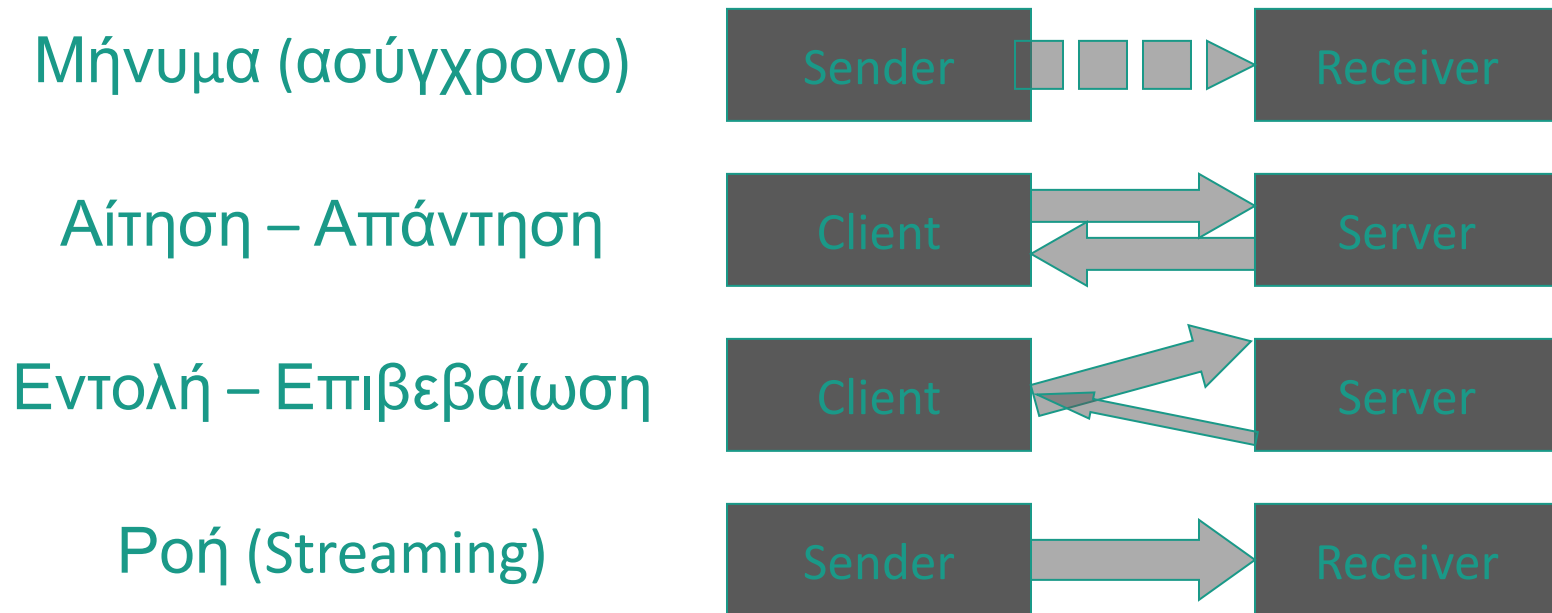


# Παραδείγματα Πρωτοκολλων Στο Επιπεδο Εφαρμογης

- Διαχείριση ονομάτων στο διαδίκτυο
  - Domain Name System
- Μεταφορά αρχείων
  - File Transfer Protocol
- Μεταφορά υπερκειμένου και αρχείων
  - Hypertext Transfer Protocol
- Μεταφορά ηλεκτρονικού ταχυδρομείου
  - Simple Mail Transport Protocol
- Διαχείριση δικτύου
  - Simple Network Management Protocol
- Απομακρυσμένη πρόσβαση σε συστήματα
  - Telnet

# Απαιτήσεις επικοινωνίας εφαρμογών

## Εναλλακτικές αρχιτεκτονικές



# Απαιτήσεις Επικοινωνιας Εφαρμογών

## Χαρακτηριστικά Ποιότητας επικοινωνιας

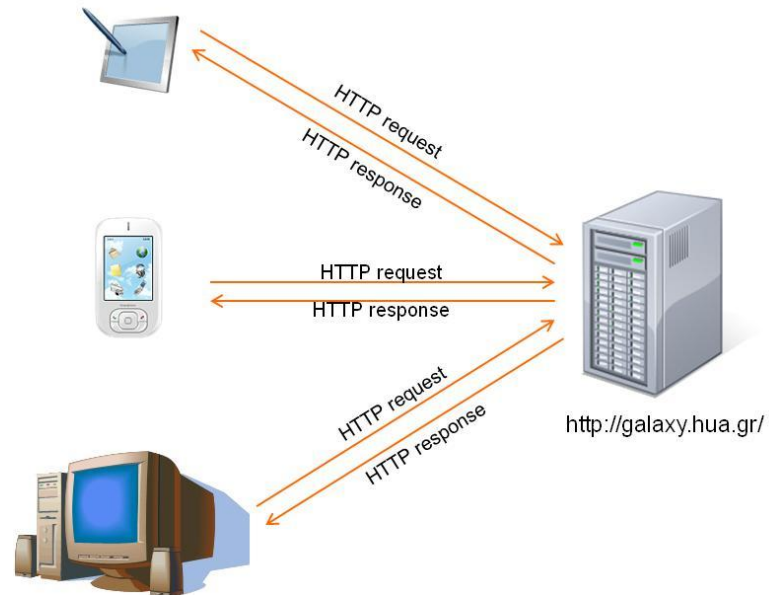
- Απώλεια δεδομένων
  - Σε κάποιες εφαρμογές (π.χ. Video) η απώλεια κάποιων δεδομένων είναι αποδεκτή
  - Άλλες εφαρμογές απαιτούν 100% αξιόπιστη μεταφορά δεδομένων (π.χ. Μεταφορά αρχείων)
- Χρονισμός
  - Σε κάποιες εφαρμογές πραγματικού χρόνου η καθυστέρηση μετάδοσης δεν είναι αποδεκτή (π.χ. Τηλεφωνία Internet, interactive games)
  - Σε άλλες είναι αποδεκτή (π.χ. Streaming με διαφορά κάποιων δευτερολέπτων)
- Χωρητικότητα δικτύου
  - Κάποιες εφαρμογές απαιτούν μεγάλο bandwidth για να λειτουργήσουν (π.χ. Τηλεδιάσκεψη με εικόνα)
  - Για άλλες (ελαστικές εφαρμογές) το bandwidth δεν είναι κρίσιμο (π.χ. Ηλ. Ταχυδρομείο)

# Συνοψη Απαιτησεων Κοινων Εφαρμογων

Εφαρμογή	Απώλεια Δεδομένων	Bandwidth	Χρονισμός
Μεταφορά Αρχείων	Μη Αποδεκτή	Ελαστικό	Όχι
E-mail	Μη Αποδεκτή	Ελαστικό	Όχι
Web Browsing	Αποδεκτή	Ελαστικό	Όχι
Real-time Audio/Video	Αποδεκτή	Audio: 5kb-1mb Video:10kb-5mb	Ναι, Εκατοστά Δευτερολέπτου
Αποθηκ. Audio/Video	Αποδεκτή	>>	Ναι, Δευτερολ.
Interactive Games	Αποδεκτή	>10kb	Ναι, Εκατοστά Δ.
Λογιστική	Μη Αποδεκτή	Ελαστικό	Ναι, Για Πολλούς Χρήστες

# Το Πρωτόκολλο HTTP (HyperText Transfer Protocol)

- Πρωτόκολλο επιπέδου εφαρμογής για τον παγκόσμιο ιστό (WWW)
- Μοντέλο client/server
  - client: ένα πρόγραμμα – πλοηγός που αιτείται, λαμβάνει και παρουσιάζει Υπερκείμενο (HTML) και άλλα αντικείμενα του WWW
  - server: εξυπηρετής παγκόσμιου ιστού που στέλνει απαντήσεις (υπερκείμενο κλπ.) για κάθε αίτημα που λαμβάνει
- http1.0: RFC 1945
- http1.1: RFC 2068 ([www.ietf.org](http://www.ietf.org))
- HTTP/2: RFC 7540



# Το Πρωτόκολλο HTTP (2)

- http ως μια υπηρεσία του TCP:
  1. Ο πελάτης ξεκινά μια σύνδεση TCP (ιδεατό κανάλι - socket) προς τον server, στην πόρτα 80
  2. Ο server δέχεται τη σύνδεση TCP από τον πελάτη
  3. Ανταλλάσσονται μηνύματα του πρωτοκόλλου http (επίπεδο εφαρμογής) με συγκεκριμένο συντακτικό μεταξύ του πλοηγού (http client) και του εξυπηρετή WWW (http server)
  4. Η σύνδεση TCP κλείνει
- Το http είναι “stateless”
  - Κάθε αίτημα είναι ανεξάρτητο από το προηγούμενο και το επόμενο

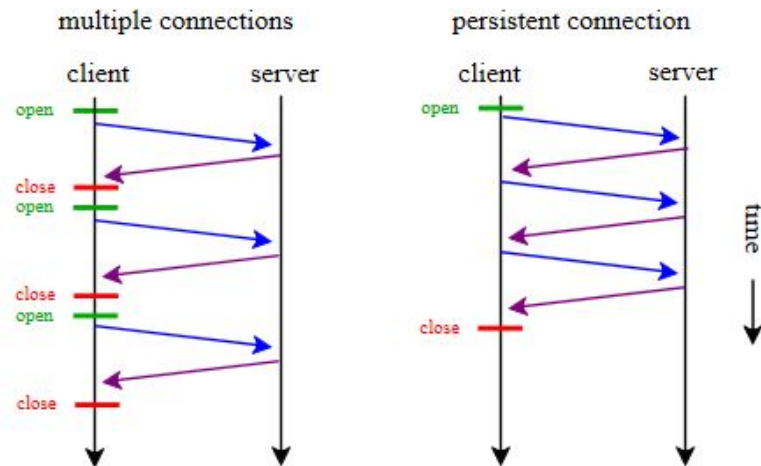
# HTTP - Παραδειγμα

1. Ο χρήστης εισάγει το URL <http://galaxy.hua.gr/~tserpes/index.html> (περιέχει μία σελίδα .html και αναφορές σε 1 εικόνα)
2. ο http πελάτης ξεκινά μία σύνδεση TCP (socket) προς τον http server που βρίσκεται στη διεύθυνση 195.130.90.169 (galaxy.hua.gr). Η σύνδεση γίνεται στην πόρτα TCP 80.
3. ο http server που 'αναμένει' για εισερχόμενες συνδέσεις TCP στην πόρτα 80 “δέχεται” τη σύνδεση και στέλνει επίβεβαίωση προς τον πελάτη
4. ο http πελάτης στέλνει ένα μήνυμα-αίτηση (request) προς την TCP σύνδεση που ανοίξε: GET /index.htm HTTP/1.0
5. ο http server λαμβάνει την αίτηση και ετοιμάζει ένα μήνυμα-απόκριση (response message) που περιέχει μία επικεφαλίδα με πληροφορίες (http header) και το περιεχόμενο (html) του αρχείου index.html. Το στέλνει προς την ανοικτή σύνδεση.
6. ο http πελάτης λαμβάνει το μήνυμα-απόκριση, επεξεργάζεται τον html κώδικα και εμφανίζει το αποτέλεσμα. Μέσα στον κώδικα βρίσκει την αναφορά προς τη μία εικόνα, για την οποία ετοιμάζει νέα αιτήση
7. ο http server κλείνει τη σύνδεση TCP



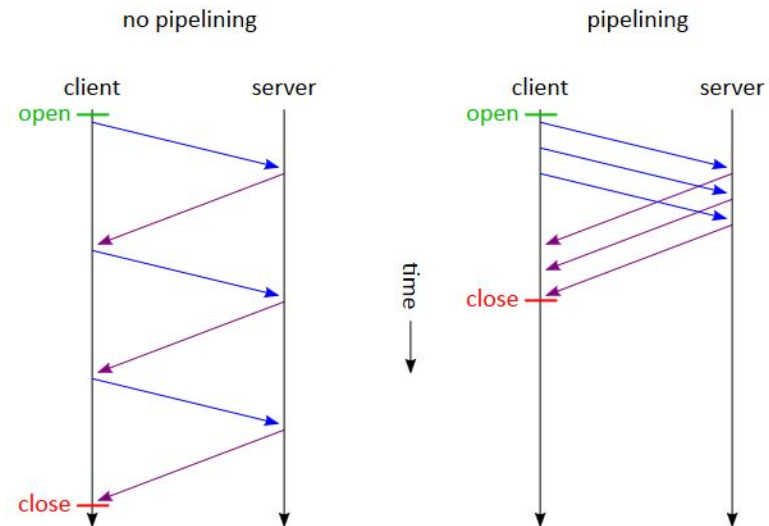
# HTTP – Παραδειγμα (Εναλλακτικές Υλοποιήσεις)

- Στιγμιαίες συνδέσεις (non-persistent): ένα αντικείμενο για κάθε TCP σύνδεση. (HTTP/1.0)
- Παράλληλες στιγμιαίες συνδέσεις (pipelining): Ταυτόχρονες, 'παράλληλες' συνδέσεις (μία για κάθε αντικείμενο) – Υπάρχει όριο στον αριθμό συνδέσεων



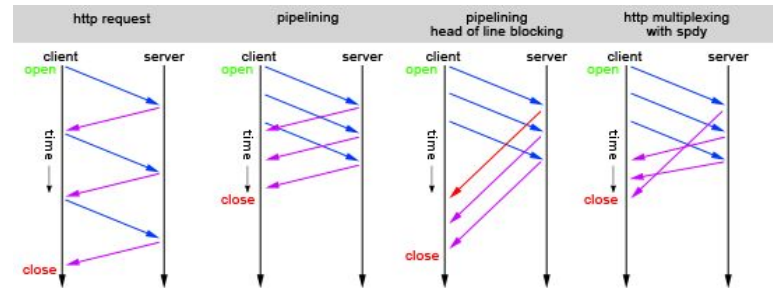
# HTTP – Παραδειγμα (Εναλλακτικές Υλοποιήσεις)

- Στιγμιαίες συνδέσεις (non-persistent): ένα αντικείμενο για κάθε TCP σύνδεση. (HTTP/1.0)
- Παράλληλες στιγμιαίες συνδέσεις (pipelining): Ταυτόχρονες 'παράλληλες' συνδέσεις (μία για κάθε αντικείμενο)
  - Υπάρχει όριο στον αριθμό συνδέσεων
- Διαρκείς (persistent) συνδέσεις: Πολλά αντικείμενα μεταφέρονται μέσα από την ίδια σύνδεση TCP. (HTTP/1.1)

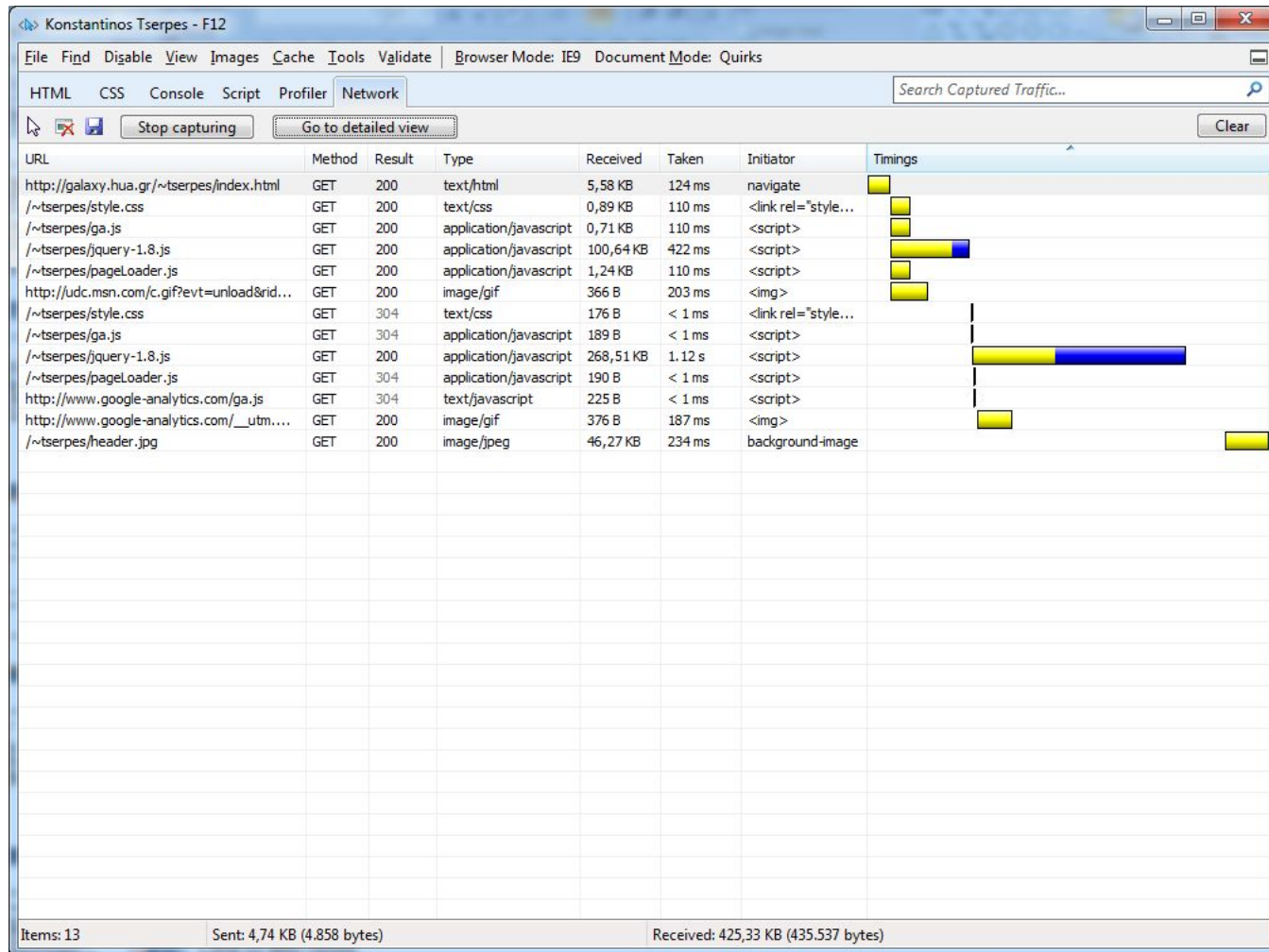


# HTTP – Παραδειγμα (Εναλλακτικές Υλοποιήσεις)

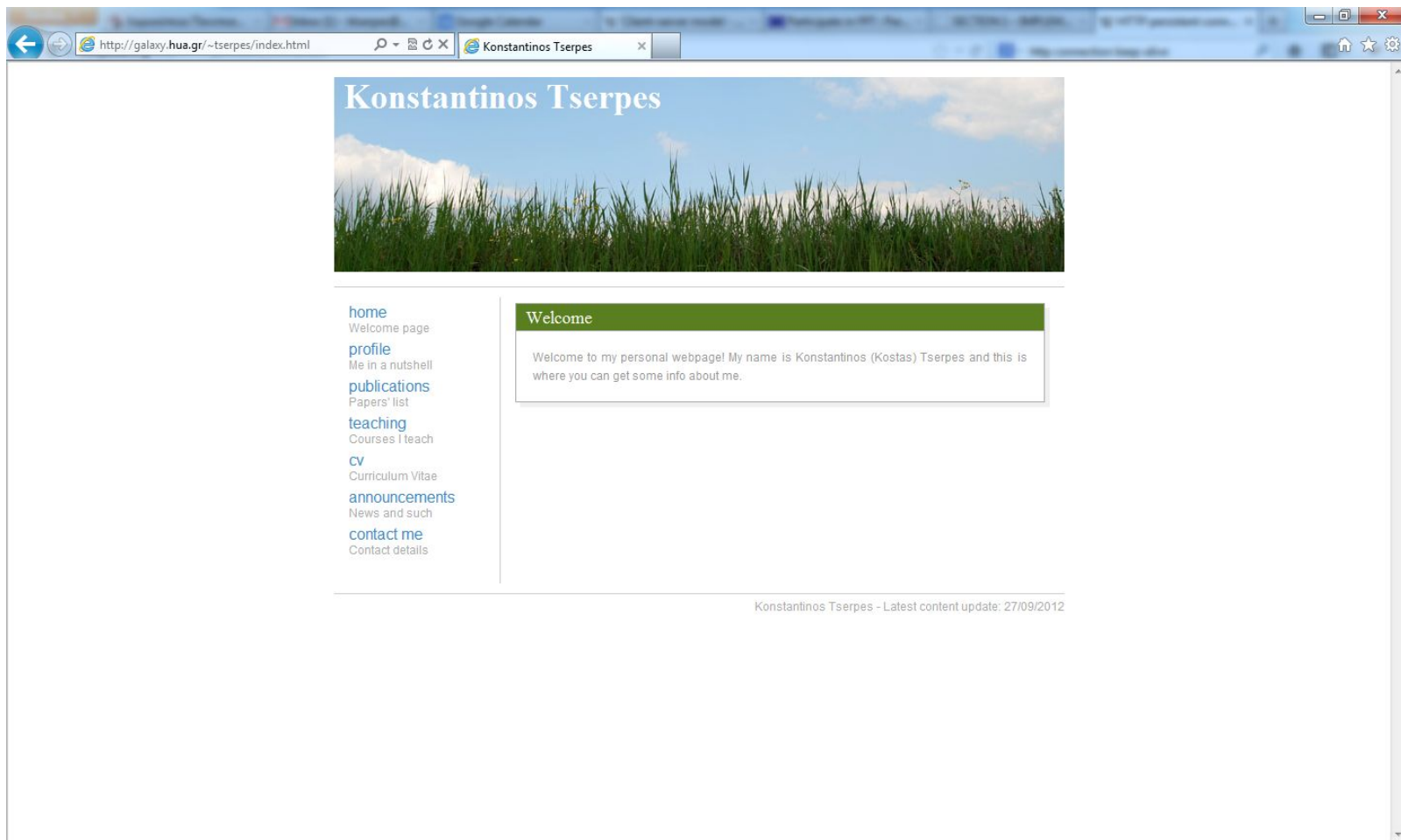
- Στιγμιαίες συνδέσεις (non-persistent): ένα αντικείμενο για κάθε TCP σύνδεση. (HTTP/1.0)
- Παράλληλες στιγμιαίες συνδέσεις (pipelining): Ταυτόχρονες 'παράλληλες' συνδέσεις (μία για κάθε αντικείμενο) – Υπάρχει όριο στον αριθμό συνδέσεων
- Διαρκείς (persistent) συνδέσεις: Πολλά αντικείμενα μεταφέρονται μέσα από την ίδια σύνδεση TCP. (HTTP/1.1)
- Πολυπλεξία (multiplexing): Οι απαντήσεις δίνονται με σειρά ανεξάρτητα από τη σειρά που παράγονται τα αιτήματα. (HTTP/2)



# Διαρκής Συνδεση (With Pipelining)



# Αποτέλεσμα (Response Body)



# HTTP Conversation

- Αίτημα Πελάτη
  - Request Line: {METHOD} {Resource ID} {HTTP Version}
  - Client Generated Headers
  - Request Body
- Απάντηση Εξυπηρετητή
  - Status Line
  - Server Generated Headers
  - Data (Response Body)

# Request Line



Request Line

Client Generated Headers

Request Body

Server Generated Headers

Data (Response Body)

# Μεθοδοι

- Εντολές που αποστέλλει ο client στο server
- Η λειτουργία τους έχει μοντελοποιηθεί στις προδιαγραφές του HTTP
- Η υλοποίηση της λειτουργικότητας εξαρτάται από το server
- Ένας server δε χρειάζεται να υποστηρίζει όλες τις πιθανές μεθόδους



# Λιστα Μεθόδων

- GET
  - Αίτημα για παραλαβή ενός resource
- POST
  - Αποστολή δεδομένων σε ένα resource
- HEAD
  - Αίτημα για την παραλαβή του response header
- PUT
  - Αίτημα για δημιουργία ενός resource
- DELETE
  - Αίτημα για διαγραφή ενός resource
- TRACE
  - Αίτημα για αποστολή από το server στοιχείων του client (για debugging)
- OPTIONS
  - Αίτημα για παραλαβή της λίστας των υποστηριζόμενων μεθόδων από τον εξυπηρετητή

# Εκδόσεις HTTP

- HTTP 1.0 (1996)
- HTTP 1.1 (1999)
- HTTP/2 (02/2015)
  - As of 02/2016, 6.7% of world's top 10M websites are supporting it

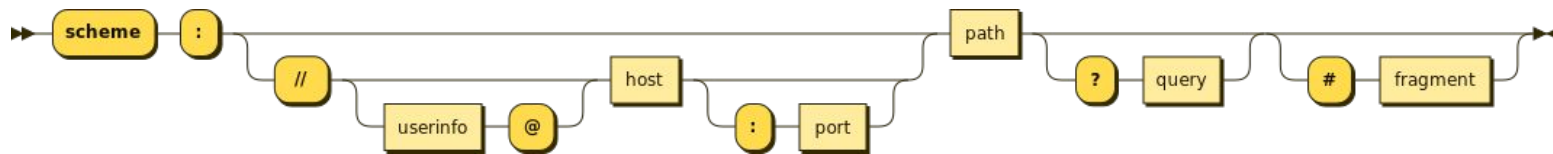
# Αναγνώριση Πόρων (Resources)

- Πρόκληση
  - Αναγνώριση πόρων μοναδικά μέσα σε όλο το δίκτυο (Internet)
- Δύο μέρη
  - Αναγνώριση των πόρων στο host
  - Αναγνώριση του host στο δίκτυο

# Uniform Resource Identification

- Αλφαριθμητικό που επιτρέπει το μοναδικό ορισμό των πόρων
- Ορισμός σύνταξης στο RFC 3986, January 2005
- Δύο μορφές
  - Uniform Resource Name (URN): Το URI αναγνωρίζει τους πόρους με βάση το όνομά τους μέσα σε ένα namespace. Π.χ. urn:isbn:0-486-27557-4
  - Uniform Resource Locator (URL): Το URI ορίζει τόσο το μηχανισμό πρόσβασης όσο και την τοποθεσία των πόρων στο δίκτυο. Π.χ. <http://www.dit.hua.gr/>
- ASCII encoded
- Άλλοι χαρακτήρες αναπαρίστανται με τη χρήση του '%' ( percent-encoded) (π.χ. Ο χαρακτήρας '#' είναι %23 ή το '/' είναι %2F ή το 'ω' είναι %CF%89)

# Σύνταξη



# Δομή του URI

- Scheme
  - Ορίζει το τύπο των πόρων
  - Παραδείγματα: http, https, ftp, mailto, file, data, irc
- Authority (προαιρετικά):
  - Userinfo: όνομα χρήστη και password με το χαρακτήρα ':' ως διαχωριστικό, ακολουθούμενα από το χαρακτήρα '@'
  - Host: Είτε ένα καταχωρημένο όνομα είτε μία διεύθυνση IP
  - port: Αριθμός μεταξύ 1 και 65535 ο οποίος δείχνει μία λογική πύλη σε μία συσκευή ακολουθούμενη από το χαρακτήρα ':'
- Path: ακολουθία από τμήματα μονοπατιών (paths) διαχωριζόμενων από το χαρακτήρα '/'.
  - Query: Το ερώτημα (query string) μη ιεραρχημένων δεδομένων (attribute-value pairs κατά σύμβαση διαχωριζόμενα από ένα χαρακτήρα. συχνά αποκαλούνται: query string parameters)
  - Fragment: προσδιοριστικό ενός τμήματος (fragment) που παρέχει κατευθύνσεις προς ένα δεύτερο resource, όπως μία ενότητα μέσα σε ένα άρθρο

# Παραδειγμα URL

The diagram illustrates the components of several Uniform Resource Identifiers (URIs) using brackets to group them into standard parts:

- https://john.doe@www.example.com:123/forum/questions/?tag=networking&order=newest#top**
  - scheme: https
  - authority: john.doe@www.example.com:123 (sub-components: userinfo: john.doe, host: www.example.com, port: 123)
  - path: /forum/questions/
  - query: ?tag=networking&order=newest
  - fragment: #top
- ldap://[2001:db8::7]/c=GB?objectClass=one**
  - scheme: ldap
  - authority: [2001:db8::7]
  - path: /c=GB
  - query: ?objectClass=one
- mailto:John.Doe@example.com**
  - scheme: mailto
  - path: John.Doe@example.com
- news:comp.infosystems.www.servers.unix**
  - scheme: news
  - path: comp.infosystems.www.servers.unix
- tel:+1-816-555-1212**
  - scheme: tel
  - path: +1-816-555-1212
- telnet://192.0.2.16:80/**
  - scheme: telnet
  - authority: 192.0.2.16:80 (sub-components: host: 192.0.2.16, port: 80)
  - path: /
- urn:oasis:names:specification:docbook:dtd:xml:4.1.2**
  - scheme: urn
  - path: oasis:names:specification:docbook:dtd:xml:4.1.2

# Σχετικά Vs Απολυτά URLs

- Σχετικές / Απόλυτες διαδρομές (Absolute/relative paths)
  - Απόλυτα: Πλήρες path προς το συνδεόμενο έγγραφο π.χ. `http://www.dit.hua.gr/index.php`
  - Σχετικά: Αναφέρονται σε έγγραφα των οποίων η θέση καθορίζεται σχετικά με το τρέχον έγγραφο π.χ. :  
`../index.html`, `./images/im1.jpg`,
    - Η χρήση σχετικών αναφορών καθιστά τα έγγραφα μεταφέρσιμα (portable) σε διαφορετικά περιβάλλοντα



# URLs Με Ονοματα Ή Με IP Διευθυνσεις;

- Γιατί δεν πρέπει να χρησιμοποιούμε μια IP διεύθυνση αντί για το όνομα του server σε ένα URL?
  - το `http://195.251.168.1` είναι URL με IP διεύθυνση
  - το `http://195.com` τι είναι?
  - το `http://3288049665` τι είναι?
  - το `195.251.168.1.gr` τι είναι?

# Δομή URL: Παραμετροί

- `http:// ... .. /myfile.php ? parameter1 = value1 & parameter2 = value2 & ... & parameterN = valueN`
- Γιατί χρειάζονται;
  - Αποστολή στοιχείων αναζήτησης
  - Ενημέρωση μιας βάσης δεδομένων
  - Αποστολή e-mail από μια εφαρμογή webmail (π.χ. `webmail.aegean.gr`)
- π.χ. αναζήτηση όρου “information theory” στο Google
  - URL: `http://www.google.com/search?hl=en&q=information theory &num=50&oe=UTF-8`
  - hl: γλώσσα, q: όρος αναζήτησης, num: αριθμός αποτελεσμάτων, oe: κωδικοποίηση χαρακτήρων  
(δίνουν πληροφορίες προς τον server του Google για το πως θα κάνει την αναζήτηση και τι θα επιστρέψει ως αποτέλεσμα)
- Θυμηθείτε τις παραμέτρους σε Functions και Procedures στον προγραμματισμό

# Δομη URL: Παραμετροι (2)

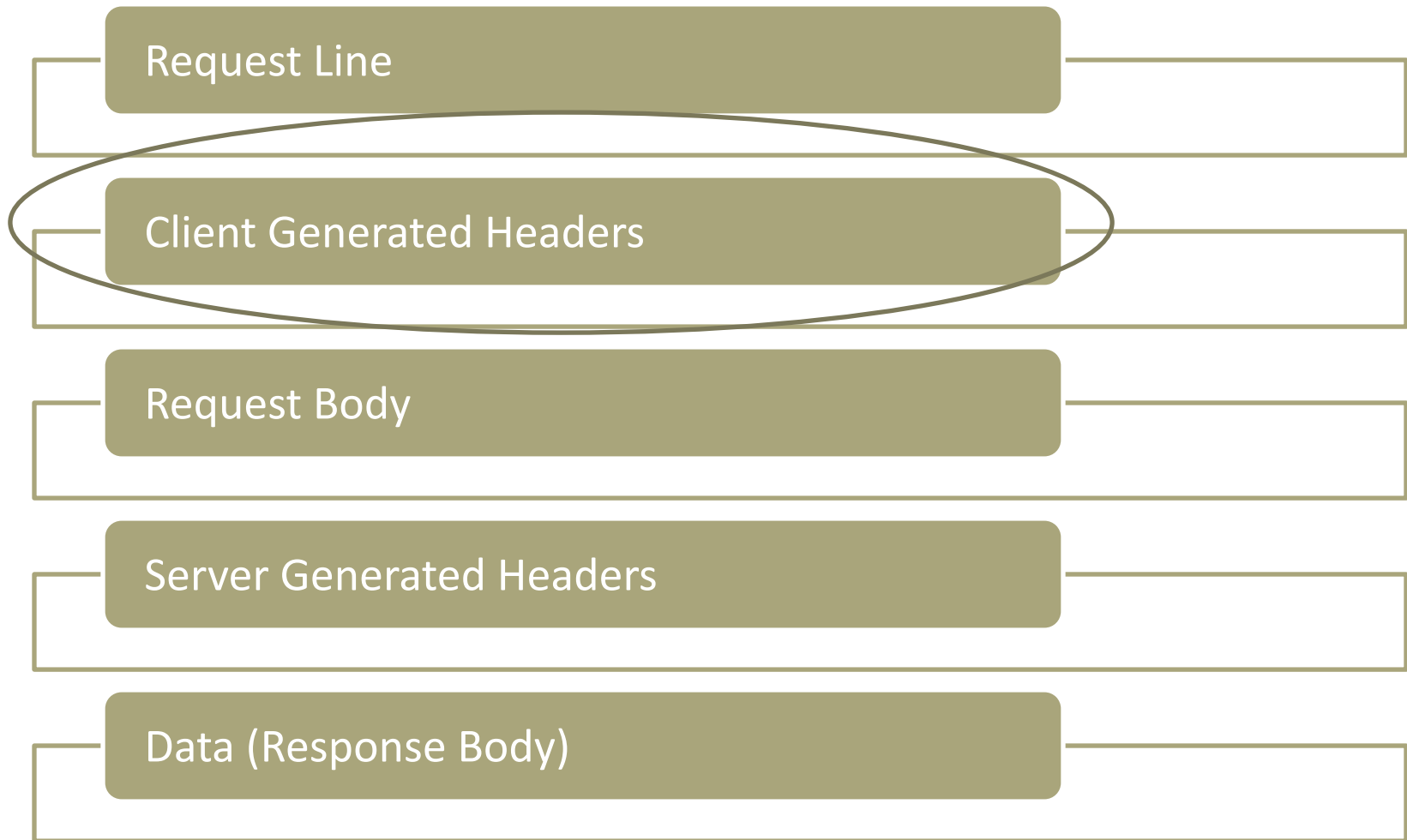
- Πως χρησιμοποιούνται από τον web-server;
- Ο πελάτης στέλνει τις παραμέτρους με τις τιμές τους προς τον server
  - είτε μέσα στο URL, αν η μέθοδος είναι GET
  - είτε ξεχωριστά αν η μέθοδος είναι POST
  - Π.χ. `http://mysite.gr/php?par1=20&par2=10`
- Ένα πρόγραμμα στον web-server αναγνωρίζει τις παραμέτρους και τις τιμές τους και τις χρησιμοποιεί (ως μεταβλητές)
- Π.χ κώδικας σε PHP:

```
<?php
    $x = $_GET["par1"];
    $y = $_GET["par2"];
    print $x-$y;
?>
```

# Κωδικοποίηση URL

- Κάποιοι χαρακτήρες χρησιμοποιούνται (είναι δεσμευμένοι) για ειδικούς λόγους (π.χ. Διαχωρισμός διαδρομής) στα URLs.
  - :, /, ?, &, κενό, ...
- Τι γίνεται όμως αν αυτοί οι χαρακτήρες περιέχονται μέσα σε τιμές παραμέτρων;
  - Π.χ. <http://www.google.com/search?q=ταχεία προτυποποίηση:υπέρ/κατά>
- Το παραπάνω δημιουργεί σύγχυση στον web-server που επεξεργάζεται το URL
- Οι ειδικοί χαρακτήρες κωδικοποιούνται και αντικαθίστανται με το σύμβολο % ακολουθούμενο από την τιμή τους ASCII σε δεκαεξαδική μορφή
- Π.χ. Google
  - Το παραπάνω θα γίνει [q=ταχεία%1Eπροτυποποίηση%3Aυπέρ%2Fκατά](http://www.google.com/search?q=ταχεία%1Eπροτυποποίηση%3Aυπέρ%2Fκατά)
- Π.χ. Αναζήτηση για σελίδες που περιέχουν το <http://www.syros.aegean.gr>
  - <http://www.google.com/search?q=http%3A%2F%2Fwww.syros.aegean.gr>
- %3A είναι : (κωδικός ASCII 58)
- %2F είναι / (κωδικός ASCII 47)
- Ο server επαναφέρει την τιμή σε [q=http://www.syros.aegean.gr](http://www.syros.aegean.gr)

# Client Generated Headers



# Common Request Headers

## Header keys

- Accept
- Accept-Language
- Authorization
- Cookie
- Host
- If-Modified-Since
- User-Agent
- Referrer

## Παράδειγμα (IE9)

Key	Value
Request	GET /~tserpes/index.html HTTP/1.1
Accept	text/html, application/xhtml+xml, */*
Accept-Language	el-GR
User-Agent	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)
Accept-Encoding	gzip, deflate
Host	galaxy.hua.gr
Connection	Keep-Alive

# Multipurpose Internet Mail Extensions (MIME) Types

- Τύποι πολυμέσων που ανταλλάσσονται μεταξύ πελάτη και server
  - text/html
  - text/css
  - image/jpeg
  - image/png
  - image/gif
  - application/pdf
  - application/msword
  - application/vnd.ms-excel

# Επιλογή Τυπων Πολυμεσων

URL: http://galaxy.hua.gr/~tserpes/header.jpg

Request headers	Request body	Response headers	Response body	Cookies	Initiator	Timings
Key		Value				
Request		GET /~tserpes/header.jpg HTTP/1.1				
Accept		image/png, image/svg+xml, image/*;q=0.8, */*;q=0.5				
Referer		http://galaxy.hua.gr/~tserpes/index.html				
Accept-Language		en-GR				
User-Agent		Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)				
Accept-Encoding		gzip, deflate				
Host		galaxy.hua.gr				
If-Modified-Since		Wed, 26 Sep 2012 08:37:35 GMT				
If-None-Match		"16007fb7fc-4ca96bc0211c0"				
Connection		Keep-Alive				
Cookie		_utma=99810312.301503687.1362922273.1362922273.1362922273.1; ...				

URL: http://galaxy.hua.gr/~tserpes/header.jpg

Request headers	Request body	Response headers	Response body	Cookies	Initiator	Timings
Key		Value				
Response		HTTP/1.1 200 OK				
Date		Sun, 10 Mar 2013 15:14:49 GMT				
Server		Apache				
Last-Modified		Wed, 26 Sep 2012 08:37:35 GMT				
ETag		"16007fb7fc-4ca96bc0211c0"				
Accept-Ranges		bytes				
Content-Length		47100				
Keep-Alive		timeout=3, max=100				
Connection		Keep-Alive				
Content-Type		image/jpeg				



# Cookies

- Παρέχουν στο server ένα μηχανισμό να αποθηκεύει και να αντλεί πληροφορίες για τον client
  - Προσωπικές πληροφορίες
  - Ρυθμίσεις
  - Usernames-Passwords
  - Ημερομηνία λήξης
- Χρήση:
  - Login μόνο μία φορά (ή και καμία φορά) κατά την πλοήγησή μας σε διαφορετικές σελίδες του ίδιου web site (Θυμηθείτε: Το http είναι 'stateless')
  - Καταγραφή προσωπικών επιλογών που κάναμε σε ένα site, ώστε να μην χρειάζεται να τις ξανακάνουμε στην επόμενη επίσκεψη (π.χ. Παλέτα χρωμάτων, αγαπημένα links, ...)
- Τα cookies αποθηκεύονται στον client και το περιεχόμενό τους αποστέλλεται μαζί με το request

# Cookies (2)

- Που βρίσκονται;
  - Δείτε στον υπολογιστή σας τον φάκελο 'Cookies' μέσα στο profile σας  
π.χ. C:\documents and settings\user\Cookies
- Υπάρχει κίνδυνος; ΝΑΙ!
  - Καταγραφή προσωπικών προτιμήσεων και μετάδοση προς τρίτους
  - Χρήση username-password χωρίς να το γνωρίζει ο χρήστης
  - Ανάγνωση των αρχείων cookies από τρίτους (εκτός από πιθανά προσωπικά δεδομένα που περιέχουν, βγαίνει αμέσως συμπέρασμα για το ποια sites έχω επισκεφθεί)
- Πως προστατεύομαι;
  - Διαγραφή Cookies
  - Κρυπτογράφηση περιεχομένων (το φροντίζει το ίδιο το web site και όχι εμείς)

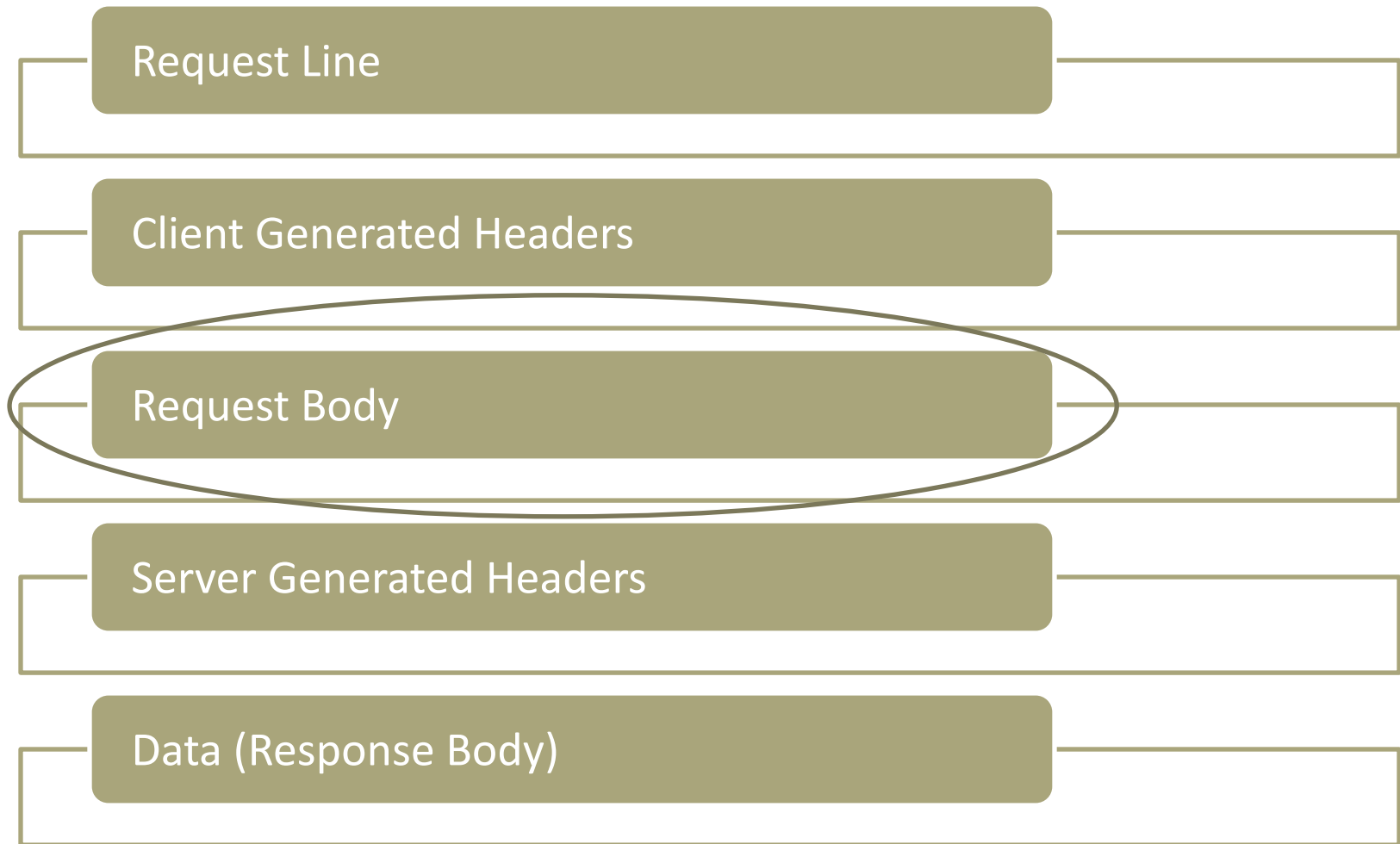
# Cookie Header

- Τοποθετούνται στο request header
- Είναι της μορφής:
  - `<name>=<value>`  
`[;<name>=<value>]...`

URL: <http://www.dit.hua.gr/index.php/el/>

Request headers	Request body	Response headers	Response body	Cookies	Initiator	Timings
Key	Value					
Request	GET /index.php/el/ HTTP/1.1					
Accept	text/html, application/xhtml+xml, */*					
Accept-Language	el-GR					
User-Agent	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)					
Accept-Encoding	gzip, deflate					
Host	www.dit.hua.gr					
Connection	Keep-Alive					
Cookie	bc7ed0866f896c31ba4a3ac501aaff5a=vugg2t4mvk6s5dbda9evopsi20					

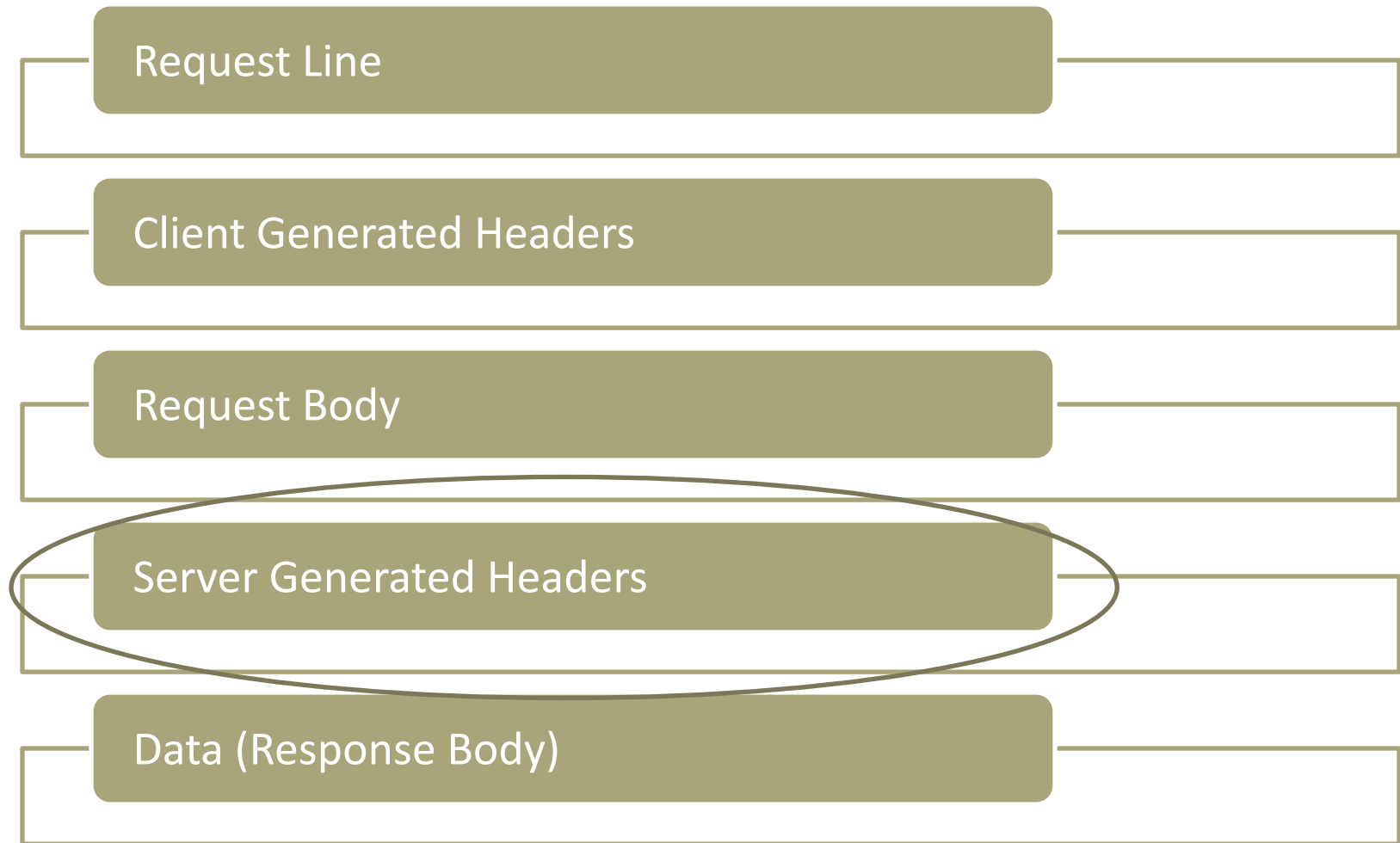
# Request Body



# Request Body

- Δεδομένα που μπορεί να αξιοποιήσει το resource
- Π.χ. το resource είναι ένα πρόγραμμα το οποίο προσθέτει 2 αριθμούς
  - Το request body πρέπει να περιέχει τους αριθμούς αυτούς
- Ελεύθερη σύνταξη
  - Client και Server πρέπει να συμφωνούν

# Server Generated Headers



# Common Response Headers

## Header keys

- Response
- E(ntity)tag
- Content-Length
- Keep-Alive
- Content-Type
- Last-Modified
- Set-Cookie

## Παράδειγμα (IE9)

Key	Value
Response	HTTP/1.1 304 Not Modified
ETag	160074-1540-4cd6eb1260c40
Content-Length	5440
Keep-Alive	timeout=3, max=100
Content-Type	text/html
Last-Modified	Thu, 01 Nov 2012 13:06:49 GMT

# HTTP Response Codes

## Κατηγοριοποίηση κωδικών

Code Range	Meaning
100's	Informational
200's	Success
300's	Redirected
400's	Request Incomplete
500's	Server Error

## Συνήθεις κωδικοί

200	OK
301	Moved permanently
302	Moved temporarily
304	Not modified
403	Forbidden
404	Not found
500	Internal server error



# Cookies

- Set-Cookie:

```
<name>=<value>[;  
<name>=<value>]...  
[; expires=<date>][;  
domain=<domain_name>][;  
path=<some_path>]  
[; secure][; httponly]
```

URL: http://www.dit.hua.gr/index.php/el/

Request headers	Request body	Response headers	Response body	Cookies	Initiator	Timings
Key	Value					
Response	HTTP/1.1 200 OK					
Date	Sun, 10 Mar 2013 15:41:28 GMT					
Server	Apache/2.2.11 (Unix) PHP/5.2.9 mod_ssl/2.2.11 OpenSSL/0.9.7d DAV/2 mod_jk/1.2.27 mod_perl/2.0.2 Perl/v5.8.4					
X-Powered-By	PHP/5.2.9					
Set-Cookie	40e65f9a47a38177e797a72d40070736=el-GR; expires=Mon, 10-Mar-2014 15:41:28 GMT; path=/					
Set-Cookie	ja_university_tpl=ja_university; expires=Fri, 28-Feb-2014 15:41:28 GMT; path=/					
P3P	CP="NOI ADM DEV PSAi COM NAV OUR OTRo STP IND DEM"					
Last-Modified	Sun, 10 Mar 2013 13:42:13 GMT					
ETag	44fdee6edb194f40b0f505d38fbc812f					
Keep-Alive	timeout=5, max=100					
Connection	Keep-Alive					
Transfer-Encoding	chunked					
Content-Type	text/html					

# Caching Related Headers

- If-Modified-Since
- Age
- Expires
- Last-Modified
- Cache-Control
- ETag

Client loads content received earlier

URL	Method	Result	Type	Received	Taken	Initiator
http://galaxy.hua.gr/~tserpes/index.html	GET	304	text/html	179 B	172 ms	refresh
/~tserpes/style.css	GET	304	text/css	176 B	62 ms	<link rel="style...
/~tserpes/ga.js	GET	304	application/javascript	190 B	31 ms	<script>
/~tserpes/jquery-1.8.js	GET	304	application/javascript	195 B	62 ms	<script>
/~tserpes/pageLoader.js	GET	304	application/javascript	190 B	78 ms	<script>
http://www.google-analytics.com/ga.js	GET	304	text/javascript	225 B	156 ms	<script>
/~tserpes/header.jpg	GET	304	image/jpeg	181 B	46 ms	background-image
http://www.google-analytics.com/__utm....	GET	200	image/gif	376 B	172 ms	<img>

# Response Body

Request Line

Client Generated Headers


Request Body

Server Generated Headers

Data (Response Body)

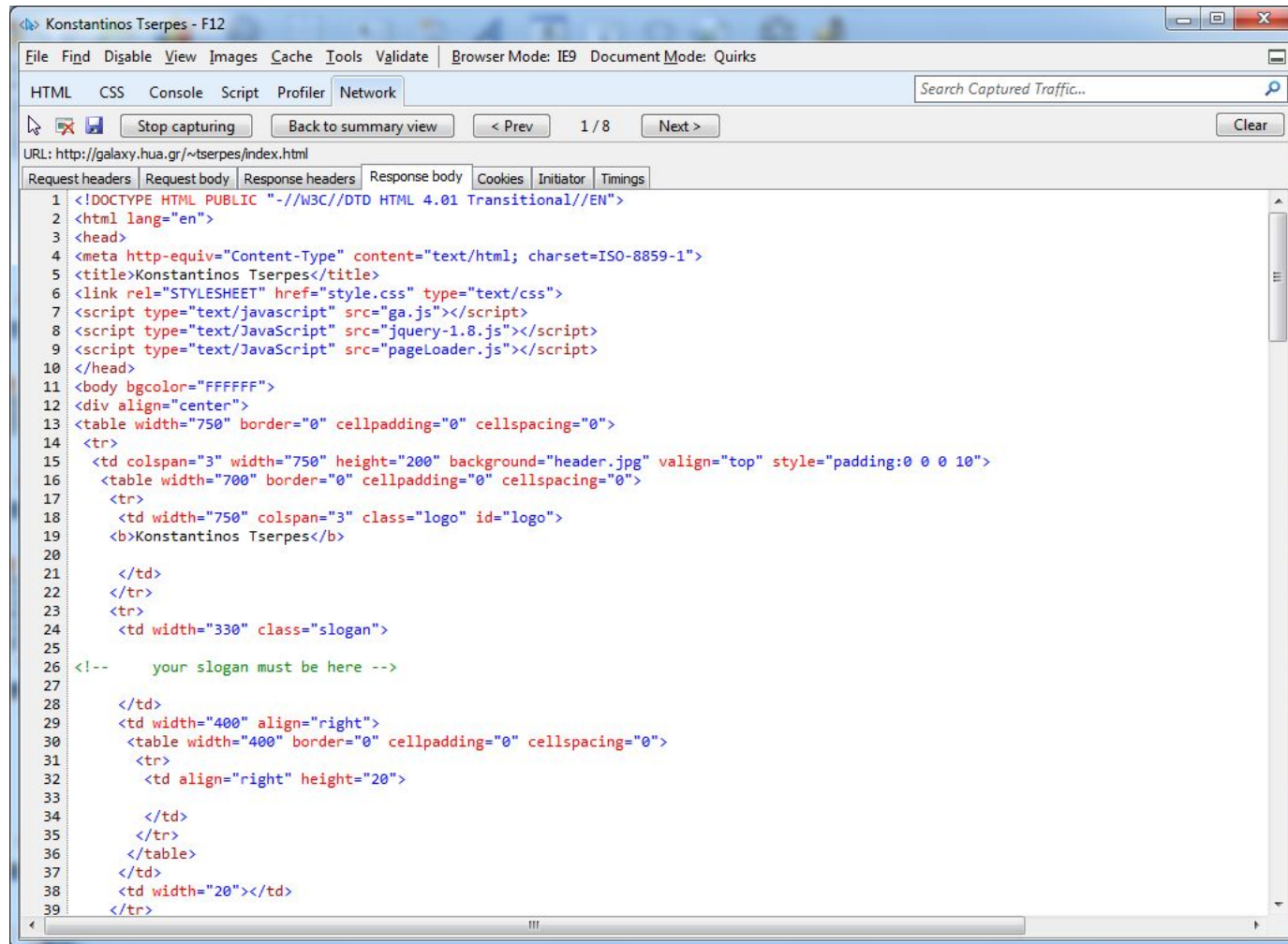
# Στην Πραξη

URL: http://galaxy.hua.gr/~tserpes/index.html						
Request headers	Request body	Response headers	Response body	Cookies	Initiator	Timings
Key	Value					
Request	GET /~tserpes/index.html HTTP/1.1					
Accept	text/html, application/xhtml+xml, */*					
Accept-Language	el-GR					
User-Agent	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)					
Accept-Encoding	gzip, deflate					
Host	galaxy.hua.gr					
Connection	Keep-Alive					



URL: http://galaxy.hua.gr/~tserpes/index.html						
Request headers	Request body	Response headers	Response body	Cookies	Initiator	Timings
Key	Value					
Response	HTTP/1.1 200 OK					
Date	Sun, 10 Mar 2013 10:44:47 GMT					
Server	Apache					
Last-Modified	Thu, 01 Nov 2012 13:06:49 GMT					
ETag	"160074-1540-4cd6eb1260c40"					
Accept-Ranges	bytes					
Content-Length	5440					
Keep-Alive	timeout=3, max=100					
Connection	Keep-Alive					
Content-Type	text/html					

# Response Body



The screenshot shows a web browser window titled "Konstantinos Tserpes - F12". The "Network" tab is selected, and the "Response body" sub-tab is active. The URL is "http://galaxy.hua.gr/~tserpess/index.html". The response body is HTML code, displayed with line numbers from 1 to 39. The code includes a DOCTYPE declaration, HTML and head tags, a meta tag for content type, a title tag, and several link and script tags. The body contains a table with a logo and a slogan.

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
2 <html lang="en">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
5 <title>Konstantinos Tserpes</title>
6 <link rel="stylesheet" href="style.css" type="text/css">
7 <script type="text/javascript" src="ga.js"></script>
8 <script type="text/JavaScript" src="jquery-1.8.js"></script>
9 <script type="text/JavaScript" src="pageLoader.js"></script>
10 </head>
11 <body bgcolor="FFFFFF">
12 <div align="center">
13 <table width="750" border="0" cellpadding="0" cellspacing="0">
14 <tr>
15 <td colspan="3" width="750" height="200" background="header.jpg" valign="top" style="padding:0 0 0 10">
16 <table width="700" border="0" cellpadding="0" cellspacing="0">
17 <tr>
18 <td width="750" colspan="3" class="logo" id="logo">
19 <b>Konstantinos Tserpes</b>
20
21 </td>
22 </tr>
23 <tr>
24 <td width="330" class="slogan">
25
26 <!-- your slogan must be here -->
27
28 </td>
29 <td width="400" align="right">
30 <table width="400" border="0" cellpadding="0" cellspacing="0">
31 <tr>
32 <td align="right" height="20">
33
34 </td>
35 </tr>
36 </table>
37 </td>
38 <td width="20"></td>
39 </tr>
```

# Αιτηση GET Ή POST?

- Ο πελάτης μπορεί να δημιουργήσει δυο διαφορετικούς τύπους αίτησης:
  - GET για την ανάκτηση αρχείων ή δεδομένων από τον server
  - POST για την αποστολή δεδομένων προς τον server
  - Υπάρχουν ακόμη PUT, DELETE, TRACE
- Όταν αιτούμαστε ένα αρχείο, χρησιμοποιούμε GET
- Όταν υποβάλλουμε (submit) μια HTML φόρμα, μπορούμε να χρησιμοποιήσουμε είτε GET είτε POST.
  - GET: Οι τιμές που στέλνω ενσωματώνονται στο URL (καλύτερο όταν κάνω αναζήτηση στοιχείων)  
π.χ. GET /city\_distance.php?from=Athens&to=Paris
  - POST: Οι τιμές δεν φαίνονται στο URL (καλύτερο όταν κάνω ενημέρωση στοιχείων)  
π.χ. POST /city\_distance.php  
from = Athens  
to = Paris