

ΚΕΦΑΛΑΙΟ 2

ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΤΑΝΕΜΗΜΕΝΩΝ ΣΥΣΤΗΜΑΤΩΝ

Σύνοψη

Στο κεφάλαιο αυτό παρουσιάζονται τα Κατανεμημένα Συστήματα (Κ.Σ.), ως ένα σύνολο κατανεμημένων αντικειμένων που υλοποιούν υπηρεσίες εφαρμογών ή υποκείμενων υπηρεσιών ή πόρων. Ουσιαστικά μιλάμε για μια Κατανεμημένη Υπολογιστική. Η αρχιτεκτονική των Κ.Σ. προσεγγίζεται από δύο απόψεις, την κατακόρυφη (διαστρωματική) προσέγγιση και την οριζόντια. Εξετάζονται λεπτομερώς οι υπηρεσίες των επιπέδων της διαστρωματικής προσέγγισης και γίνεται εστίαση σε βασικά ιδιώματα των Κ.Σ., ήτοι στις υπηρεσίες κατανεμημένης επεξεργασίας (π.χ. επικοινωνίας, ονομάτων, ευρετηρίου, εντοπισμού πόρων, κατανεμημένων αρχείων, συγχρονισμού, κατανεμημένων αντικειμένων, αναπαραγωγής υπηρεσιών και ανοχής σε σφάλματα), στις διαφάνειες (π.χ. τοποθεσίας, πρόσβασης, μετανάστευσης, και αστοχίας.), στην κλιμάκωση (π.χ. γεωγραφική και λειτουργική) και στην ανοικτότητα, η οποία ευνοεί την ανάπτυξη διαλειτουργούντων συστημάτων. παρουσιάζουμε θέματα σχεδίασης των σύγχρονων κατανεμημένων συστημάτων. Τέλος, εξετάζουμε την αρχιτεκτονική και από την άποψη της οριζόντιας και των πολυεπίπεδων (N-tiers) αρχιτεκτονικών, οι οποίες εστιάζουν μεταξύ της επικοινωνίας δύο ή περισσότερων υπολογιστών χωρίς να υπερέχονται στις ασχολούνται με τις υπηρεσίες των υποκείμενων πλατφορμών. Εδώ κυρίαρχο είναι το μοντέλο πελάτη-εξυπηρετητή.

Προαπαιτούμενη γνώση

- 1) Δουληγέρης, Χ., Μητρόπουλος, Σ., 2015. Πληροφοριακά συστήματα στο διαδίκτυο. [ηλεκτρ. βιβλ.] Αθήνα: Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών. Διαθέσιμο στο: <http://hdl.handle.net/11419/3969>
- 2) Savitch Walter (2015), Java, 7η Έκδοση, Εκδόσεις Α. Τζιόλα & Υιοί Α.Ε.
- 3) Shari Lawrence (2011), Τεχνολογία Λογισμικού: Θεωρία και Πράξη, Έκδοση: 2η Αμερικανική, Εκδόσεις Κλειδάριθμος ΕΠΕ.
- 4) Βακάλη Α., Παπαμήτσιου Ζ. (2012), Πληροφοριακά Συστήματα Παγκοσμίου Ιστού, Έκδοση: 1η, Εκδόσεις Νέων Τεχνολογιών.
- 5) Else Lervik, Vegard B. Havdal (2004), Java Με UML: Αντικειμενοστραφής Σχεδίαση και Προγραμματισμός, Έκδοση: 1η, Εκδόσεις Κλειδάριθμος ΕΠΕ.
- 6) Andrew S. Tanenbaum, Herbert Bos (2018), Σύγχρονα Λειτουργικά Συστήματα, Έκδοση: 4η Αμερικανική, Εκδόσεις Κλειδάριθμος ΕΠΕ
- 7) Παπακωνσταντίνου Γ., Τσανάκας Π., Θεοχάρης Θ., Συστήματα Παράλληλης Επεξεργασίας, Εκδόσεις Σ. Αθανασόπουλος & ΣΙΑ Ο.Ε.

2.1. Κατανεμημένη Υπολογιστική

Το ενδιάμεσο λογισμικό είναι μια πλατφόρμα κατανεμημένης επεξεργασίας, η οποία θα πρέπει να παρέχει υπηρεσίες για την ευέλικτη δόμησή της. Ως εκ τούτου, απαιτείται η ανάπτυξη μηχανισμών κατανεμημένης επεξεργασίας και επικοινωνίας μεταξύ των κατανεμημένων αντικειμένων. Τα αντικείμενα εδώ νοούνται ως αναπαραστάσεις υπηρεσιών ή πόρων του συστήματος, και είναι διάσπαρτα σε πολλές υπολογιστικές μηχανές.

Η αρχιτεκτονική των σύγχρονων υπολογιστικών συστημάτων είναι δομημένη σε επίπεδα, το κάθε ένα από τα οποία ικανοποιεί ένα σύνολο λειτουργικών αναγκών. Αυτή η δόμηση της αρχιτεκτονικής σε επίπεδα, ή αλλιώς αυτή η διαστρωμάτωση, περιλαμβάνει το ενδιάμεσο λογισμικό, το οποίο στέκεται μεταξύ του λειτουργικού συστήματος και των κατανεμημένων εφαρμογών. Ισοδύναμος, κάτω από το επίπεδο των κατανεμημένων εφαρμογών στέκεται ένα σύνολο κοινών υπηρεσιών ή αλλιώς οι υπηρεσίες πλατφόρμας, οι οποίες δομούν την λειτουργικότητα της κατανεμημένης επεξεργασίας, της κλήσης των κατανεμημένων αντικειμένων, καθώς και της διαχείρισης.

Ο Διεθνής Οργανισμός Προτυποποίησης (International Organisation for Standardization - ISO), αναγνωρίζοντας την ανάγκη για τη δομημένη ανάπτυξη των κατανεμημένων συστημάτων, συνέστησε μια ομάδα εργασίας για τον καθορισμό του προτύπου ODP-RM (Open Distributed Processing - Reference Model). Επίσης, το ITU Telecommunication Standardization (ITU-T), το πρώην CCITT (International Telegraph and Telephone Consultative Committee), δείχνοντας ενδιαφέρον στην ίδια κατεύθυνση εργάστηκε στην ανάπτυξη ενός πλαισίου κατανεμημένων εφαρμογών, το λεγόμενο πρότυπο DAF (Distributed Application Framework). ISO και ITU-T ένωσαν τις προσπάθειές τους στην ανάπτυξη του προτύπου ODP. Το πρότυπο ODP είναι ιδιαίτερα σημαντικό καθώς εφαρμόζεται σε πολλούς τεχνολογικούς τομείς, όπως στις επικοινωνίες για κατανεμημένη επεξεργασία, στη διαχείριση των κατανεμημένων συστημάτων, στις βάσεις δεδομένων, καθώς και στα γραφικά και τις διαπροσωπείες των εφαρμογών.

Στο ODP ορίζονται επί μέρους διαφορετικές απόψεις (viewpoints) του αυτού συστήματος. Κάθε άποψη βλέπει και περιγράφει το σύστημα από τη δική της μεριά. Εντούτοις, όλες οι απόψεις πρέπει να είναι συνεπείς μεταξύ τους, και αυτό είναι ένα σημαντικό πρόβλημα στην ανάπτυξη των Κατανεμημένων Συστημάτων. Οι απόψεις του ODP είναι οι εξής πέντε: επιχειρησιακή, πληροφορική, υπολογιστική, μηχανική και τεχνολογική.

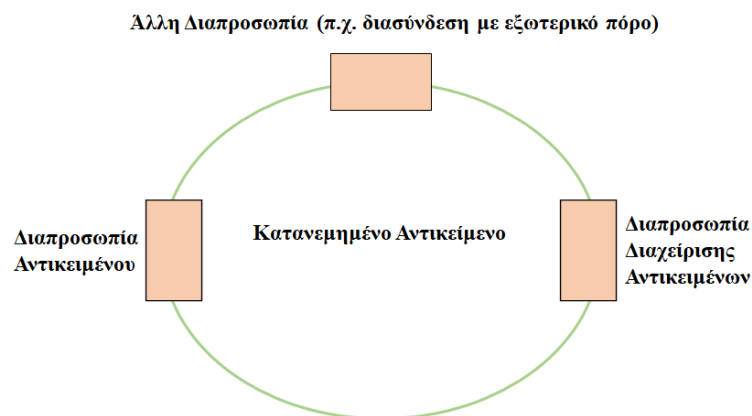
Η *επιχειρησιακή άποψη* αφορά στο επιχειρησιακό περιβάλλον στο οποίο εντάσσεται ένα κατανεμημένο σύστημα, καθώς και στους γενικότερους επιχειρησιακούς στόχους που θέλουμε να ικανοποιεί αυτό. Η *πληροφορική άποψη* έχει να κάνει με τη δομή, τη μοντελοποίηση και τις ροές της πληροφορίας. Η *υπολογιστική άποψη* έχει να κάνει με την περιγραφή του συστήματος ως ένα σύνολο αλληλοεπιδρώντων οντοτήτων, χωρίς να στέκεται στον τρόπο αλληλεπίδρασης, καθώς επίσης έχει να κάνει με τις κλήσεις (invocations), τον συγχρονισμό, τους αλγόριθμους και τις δομές των κατανεμημένων αντικειμένων. Η *μηχανική άποψη* έχει να κάνει με προβλήματα, όπως την ανοχή σε σφάλματα, τη διαφάνεια της φυσικής τοποθέτησης, και την αξιοπιστία της κατανομής. Τέλος, η *τεχνολογική άποψη* αφορά τις συγκεκριμένες τεχνολογίες, με τις οποίες υλοποιούνται τα διάφορα στοιχεία υλικού και λογισμικού του συστήματος.

Το πρότυπο ODP ακολουθεί την προσέγγιση της αντικειμενοστρεφούς αρχιτεκτονικής, καθώς προσφέρει μηχανισμούς ενθυλάκωσης (encapsulation), κληρονομικότητας (inheritance) και δομικότητας (modularity). Αυτοί είναι χρήσιμοι μηχανισμοί για την *επαναχρησιμοποίηση, την επεκτασιμότητα και τη συντήρηση των συστημάτων*, που είναι από τις βασικές απαιτήσεις στα σύγχρονα κατανεμημένα συστήματα.

Στην προσέγγιση του ODP έχει προσδιοριστεί μια σειρά από βασικές λειτουργίες, όπως αυτές της διαχείρισης, των δοσοληπιών, της ομαδοποίησης, της διαφάνειας και της ασφάλειας. Πέρα από αυτή τη λειτουργικότητα είναι πολύ βασικό να προσδιορίζεται ένας κοινός τρόπος *διασυνεργασίας (interworking)* μεταξύ των συστημάτων, τα οποία σε πολλές περιπτώσεις προκύπτουν από την ενοποίηση υπαρχόντων συστημάτων, ενώ από την φύση τους περικλείουν διαφορετικές τεχνολογίες σε επίπεδο υλικού, δικτύων, λειτουργικών συστημάτων και γλωσσών προγραμματισμού. Άρα απαιτείται προδιαγραφή του πώς πρέπει να συνεργάζονται μεταξύ τους τα κατανεμημένα αντικείμενα / υπηρεσίες, ώστε να επιτυγχάνεται ο καθολικός στόχος λειτουργίας τους. Επίσης, θα πρέπει να περιγράφεται *το πώς επιδρά η συμπεριφορά ενός κατανεμημένου αντικείμενου σε ένα άλλο κατανεμημένο αντικείμενο*. Τα ονόματα των αντικειμένων εξαρτώνται από την ιεραρχία περιεκτικότητας (containment hierarchy) και θα πρέπει να είναι ανεξάρτητα από την περιγραφή των αντικειμένων, προκειμένου να αποφευχθούν προβλήματα ονοματοδοσίας, στην περίπτωση αλλαγής της περιγραφής του συστήματος (system configuration). Με άλλα λόγια, απαιτείται η χρήση μιας *ελέκτικης υπηρεσίας ονοματοδοσίας των*

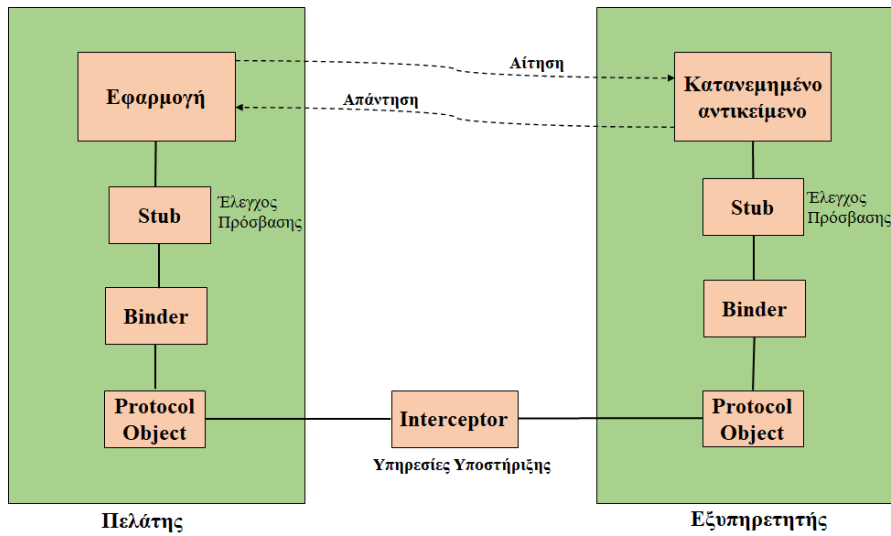
αντικειμένων, η οποία να μην επηρεάζει την λειτουργικότητα του κατανεμημένου συστήματος, όταν αυτό αλλάζει διαμόρφωση. Επίσης, ένα κατανεμημένο διαχειριστικό μοντέλο πρέπει να περιγράφει τις διαφάνειες κατανεμημένης επεξεργασίας, όπως π.χ. διαφάνειες τοποθέτησης, επανατοποθέτησης, μετακίνησης, πρόσβασης, επανάληψης, δοσοληψιών, εξακολουθητικότητας και αποτυχιών.

Τα κατανεμημένα αντικείμενα, τα οποία ονομάζονται και απομακρυσμένα, αποτελούν μια αφηρημένη αναπαράσταση μιας υπηρεσίας, παρέχοντας διαπροσωπικές επικοινωνίες προς άλλα αντικείμενα και εφαρμογές (Σχήμα 2.1). Η κατανεμημένη φύση των αντικειμένων στα κατανεμημένα συστήματα βελτιώνει την επίδοση των συστημάτων και την αποτελεσματικότητά τους. Το ODP-RM, που προδιαγράφει την λειτουργία των κατανεμημένων συστημάτων, προσφέρει το πλαίσιο για τον καθορισμό μιας κατανεμημένης υπηρεσίας. Τα κατανεμημένα αντικείμενα τα διακρίνουμε σε απλά και σύνθετα.



Σχήμα 2.1: Ένα κατανεμημένο αντικείμενο με διαπροσωπίες που δύναται να διαθέτει

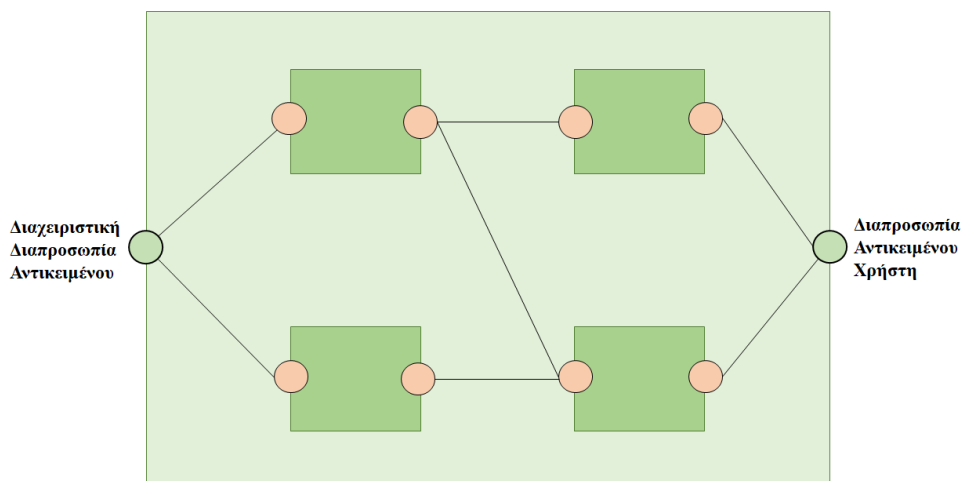
Τα απλά αντικείμενα αποτελούν την ελάχιστη μονάδα κατανομής. Προκειμένου να παρέχονται λειτουργίες ή μέθοδοι από ένα αντικείμενο θεωρούμε τις αντίστοιχες διαπροσωπείες. Οι διαπροσωπείες αυτές λειτουργούν ως σύνδεσμοι μεταξύ αλληλοεπιδρώντων αντικειμένων. Τα κατανεμημένα αντικείμενα υποστηρίζονται από τα αντικείμενα stub και binder, καθώς και από τα αντίστοιχα αντικείμενα πρωτοκόλλων (protocol). Τα κατανεμημένα αντικείμενα έχουν κύκλο ζωής, έλεγχο πρόσβασης, εμβέλεια (scoping), φιλτράρισμα (filtering) και συγχρονισμό με άλλα κατανεμημένα αντικείμενα (Σχήμα 2.2). Εδώ πρέπει να σημειωθεί ότι τα αντικείμενα stub, binder και protocol παρέχουν επιπρόσθετα υπηρεσίες, οι οποίες σχετίζονται με τη διαφανή κατανεμημένη επεξεργασία.



Σχήμα 2.2: Επικοινωνία μιας εφαρμογής με ένα απομακρυσμένο αντικείμενο

Τις καταναμημένες υπηρεσίες τις θεωρούμε ως διαπροσωπείες που προσφέρουν τα καταναμημένα αντικείμενα τα οποία διαμορφώνουν καταναμημένα συστήματα. Οπότε απαιτείται ανοικτή προσέγγιση των καταναμημένων συστημάτων, η οποία συνεπάγεται αλληλεπιδράσεις μεταξύ των διαπροσωπειών, ευέλικτη ονοματοδότηση, επικοινωνία αντικείμενο προς αντικείμενο, συμμετρία στις διαπροσωπείες, ευέλικτες περιγραφές των αντικειμένων, καταγραφή της επίδρασης της συμπεριφοράς ενός αντικειμένου σε ένα άλλο, καθώς και διαφάνειες καταναμημένης επεξεργασίας.

Επιπρόσθετα, ένα καταναμημένο αντικείμενο μπορεί να επικαλύπτει έναν πόρο (resource encapsulation), να είναι σύνθετο ως αποτέλεσμα μίας σύνθεσης άλλων αντικειμένων (Σχήμα 2.3), να διασυνεργάζεται, να έχει ομότιμες (peer-to-peer) αλληλεπιδράσεις και να υλοποιεί λειτουργίες ανάλογα με τον ρόλο και τα δικαιώματα που διαθέτει. Ένα καταναμημένο αντικείμενο μπορεί να παίζει διαφορετικούς ρόλους σε μια δεδομένη χρονική στιγμή ή σε διαφορετικές χρονικές στιγμές, αυξάνοντας έτσι την ευελιξία του καταναμημένου συστήματος.



Σχήμα 2.3: Σύνθετο καταναμημένο αντικείμενο

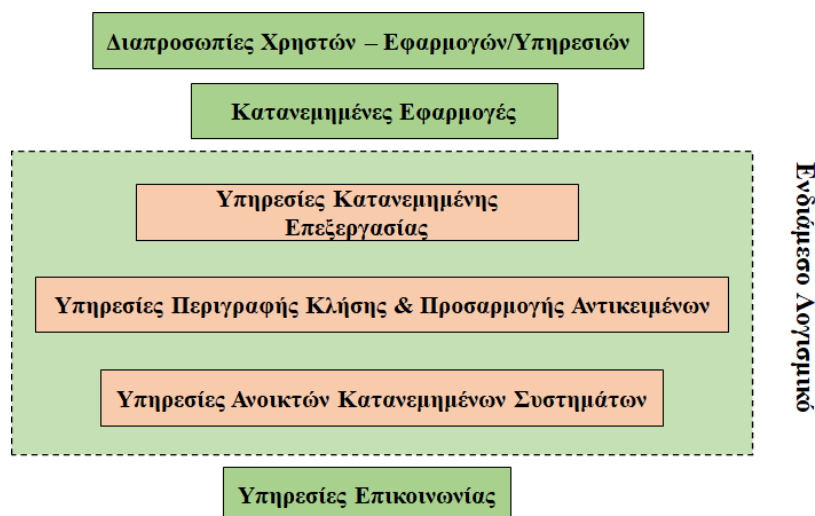
Μια διαπροσωπεία αντικειμένου προσφέρει μια άποψη του αντικειμένου και καθορίζεται από λειτουργίες που παρέχονται από το αντικείμενο. Ένα αντικείμενο μπορεί να εμπεριέχει κάποιον πόρο

που αναπαριστά π.χ. ένα πρόγραμμα εφαρμογής. Από την άλλη, ο πόρος μπορεί να είναι εξωτερικά του αντικειμένου (external resource), οπότε σε αυτή την περίπτωση το αντικείμενο λειτουργεί ως προσαρμοστής (adapter) του πόρου, π.χ. μια διαχειριστική αναπαράσταση ενός δρομολογητή. Τέλος, ένα αντικείμενο μπορεί να είναι σύνθετο, δηλαδή μπορεί να συνίσταται από άλλα αντικείμενα που αλληλοεπιδρούν διαμέσου των διαπροσωπειών τους, που είτε είναι διαπροσωπείες του σύνθετου αντικειμένου προς το εξωτερικό περιβάλλον, είτε είναι κρυφές και εξυπηρετούν στη σύνθεσή του. Παράδειγμα σύνθεσης κατανεμημένων αντικειμένων αποτελούν αποτελεί η σύνθεση ή η ενορχήστρωση των υπηρεσιών ιστού (WS Orchestration). Στο επόμενο κεφάλαιο θα δούμε την δημιουργία μιας σύνθεσης υπηρεσιών ιστού.

2.2 Αρχιτεκτονική

Τα κατανεμημένα πληροφοριακά συστήματα χρησιμοποιούν ένα σύνολο υποκειμένων υπηρεσιών που είναι κοινές για όλα τα συστήματα που εγκαθίστανται πάνω από αυτές. Οι υποκειμένες αυτές υπηρεσίες συνιστούν το ενδιάμεσο λογισμικό και είναι απαραίτητες για να λειτουργήσει ένα κατανεμημένο σύστημα. Ο σκοπός στο να προδιαγράψουμε την αρχιτεκτονική έγκειται στο να προσδιοριστούν ευκρινώς, ποιος είναι ο ρόλος και ποιες οι υπηρεσίες που πρέπει να προσφέρει το κάθε επίπεδο μιας αρχιτεκτονικής κατανεμημένων υπηρεσιών, ώστε να ικανοποιούνται όλες οι απαιτήσεις που έχουμε ήδη αναφέρει.

Οι υποκειμένες αυτές υπηρεσίες είναι κι αυτές κατανεμημένες, οπότε χρησιμοποιούν με την σειρά τους τις υπηρεσίες κατανεμημένης επεξεργασίας. Η χρήση οποιωνδήποτε υπηρεσιών πλατφόρμας απαιτεί την ανάπτυξη κατάλληλων διαπροσωπειών. Η αρχιτεκτονική πρέπει να προσφέρει υπηρεσίες περιγραφής και κλήσης αντικειμένων (object invocation), καθώς και προσαρμοστές (adaptors) για την επικοινωνία ετερογενών πλατφορμών. Όπως έχει ήδη αναφερθεί σε ένα κατανεμημένο περιβάλλον θα πρέπει να προσφέρονται μια σειρά από μηχανισμούς διαφανειών. Τέλος, στο χαμηλότερο επίπεδο πρέπει να προσφέρονται μηχανισμοί επικοινωνίας. Στο Σχήμα 2.4 αποτυπώνεται η διαστρωμένη αρχιτεκτονική που πρέπει να ακολουθεί η ανάπτυξη των Κατανεμημένων Συστημάτων.



Σχήμα 2.4: Επίπεδα Αρχιτεκτονικής των Κατανεμημένων Συστημάτων

2.3 Υπηρεσίες Κατανεμημένης Επεξεργασίας

Μια καταναμημένη εφαρμογή θα πρέπει να διαθέτει την απαραίτητη πληροφορία με τα τεχνικά χαρακτηριστικά της υποκείμενης πλατφόρμας, προκειμένου να μπορεί να διαλειτουργήσει μαζί της. Με άλλα λόγια υπάρχει μια διαπροσωπεία μεταξύ της καταναμημένης εφαρμογής και των υποκείμενων μηχανισμών του ενδιαμέσου λογισμικού. Θα πρέπει να παρέχεται η λειτουργικότητα αποστολής εντολών από και προς τα καταναμημένα αντικείμενα. Οι μηχανισμοί της πλατφόρμας του ενδιαμέσου λογισμικού, είναι κι αυτές υπηρεσίες, αλλά όχι όμως επιπέδου εφαρμογής (δηλ. δεν είναι εφαρμογές). Έτσι, μια καταναμημένη εφαρμογή, όπως άλλωστε και οι ίδιες οι υποκείμενες υπηρεσίες ενδιαμέσου λογισμικού, μπορούν χρησιμοποιώντας τις υπηρεσίες καταναμημένης επεξεργασίας να καταναμηθούν διαφανώς σε ένα σύνολο μηχανών του δικτύου.

Υπολογιστικά περιβάλλοντα που δύνανται να παρέχουν καταναμημένη επεξεργασία είναι ενδεικτικά το OSF/DCE¹ και το ANSA². Τυπικές υπηρεσίες που προσφέρουν αυτά τα περιβάλλοντα καταναμημένης επεξεργασίας είναι μεταξύ άλλων οι ακόλουθες:

- *Επικοινωνία* μεταξύ των καταναμημένων αντικειμένων
- *Χρονικές Υπηρεσίες* που διαχειρίζονται την τρέχουσα ημερομηνία και ώρα, καθώς και τον καθολικό συγχρονισμό των καταναμημένων ρολογιών
- *Ατομικές Δοσοληψίες* ως δυνατότητα διαχείρισης κλήσεων πολλαπλών λειτουργιών ως μία λειτουργία. Οι υπηρεσίες δοσοληψιών ασχολούνται με την εκτέλεση δέσμης ενεργειών με ατομικό τρόπο και με τον συγχρονισμό πολλών παράλληλων συναλλαγών.
- *Ονοματολογία* για μετάφραση των ονομάτων σε διευθύνσεις. Αυτή η υπηρεσία παρέχει έναν ενιαίο χώρο ονομάτων και τη μετάφραση των ονομάτων των πόρων σε διευθύνσεις.
- *Αρχεία και Βάσεις Δεδομένων* για τη διατήρηση των δεδομένων.
- *Υπηρεσίες Ευρετηρίου* με σκοπό την προσπέλαση σε αρχεία απομακρυσμένων μηχανών, εξειδικευμένους εξυπηρετητές ή απλούς σταθμούς εργασίας και ενιαίους ανεξάρτητους χώρους ονομάτων. Οι υπηρεσίες ευρετηρίου συσχετίζονται άμεσα με την αναζήτηση διεύθυνσης με πρόσθετα κριτήρια και τους τύπους, τη γεωγραφική θέση και τον ιδιοκτήτη του εκάστοτε πόρου.
- *Υπηρεσίες εντοπισμού*, οι οποίες εντοπίζουν την τρέχουσα θέση ενός κινούμενου πόρου. Οι υπηρεσίες αυτές συνεργάζονται με υπηρεσίες ονοματολογίας ή ευρετηρίου.
- *Ασφάλεια*, η οποία περιλαμβάνει έλεγχο πρόσβασης, ταυτοποίηση και αυθεντικοποίηση, και αποδοχή ή απόρριψη της λήψης ή της διανομής. Οι υπηρεσίες ασφάλειας έχουν ως κύριο μέλημα την πιστοποίηση ταυτότητας, τη διαφύλαξη απορρήτου και την ασφαλή συνεργασία μεταξύ των αυτόνομων μηχανών.
- *Αντικείμενα* για την αντιγραφή, τη μετανάστευση ή τη μεταβίβασή τους.
- *Εμπορία (Trading)*, η οποία χρησιμοποιείται για την επιλογή της κατάλληλης υπηρεσίας.

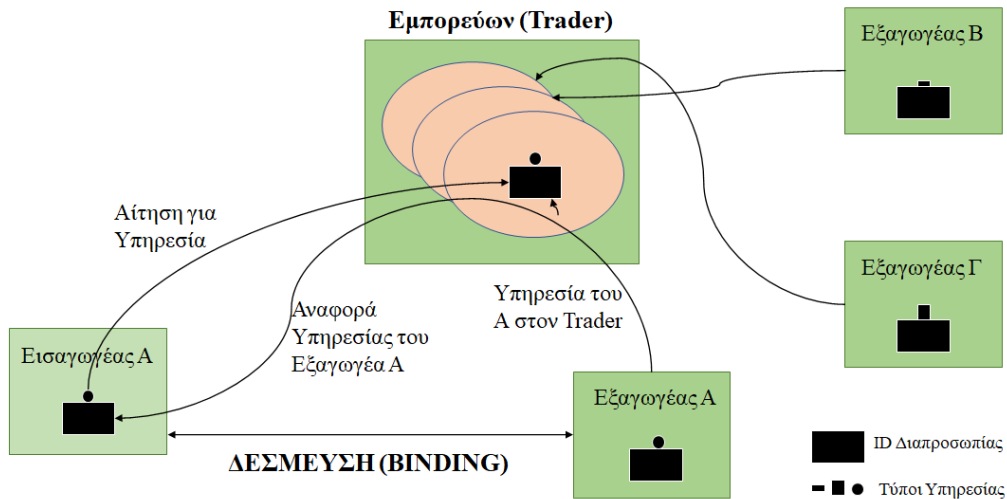
2.3.1 Εμπορία και Σύνθεση Υπηρεσιών

Η Εμπορία (Trading) είναι ένας όρος υιοθετημένος από το πρότυπο ODP. Σύμφωνα με το πρότυπο αυτό, η υπηρεσία είναι ένα σύνολο λειτουργιών, που προσφέρονται σε ένα χρήστη από έναν οργανισμό. Ο Εξαγωγέας (Exporter) είναι ένα αντικείμενο, ο εξυπηρετητής, το οποίο προσφέρει κάποια υπηρεσία στον Εμπορεύοντα (Trader). Ο Εισαγωγέας (Importer) είναι ένα αντικείμενο που ψάχνει για μια υπηρεσία και γι' αυτό τον σκοπό δίνει την περιγραφή της στον Εμπορεύοντα. Ο Εμπορεύοντας ψάχνει στο ευρετήριο υπηρεσιών, εάν υπάρχει τέτοια υπηρεσία και το ποιος εξαγωγέας την πρόσφερε.

¹ <http://www.opengroup.org/dce/> (πρόσβαση 10/9/2021)

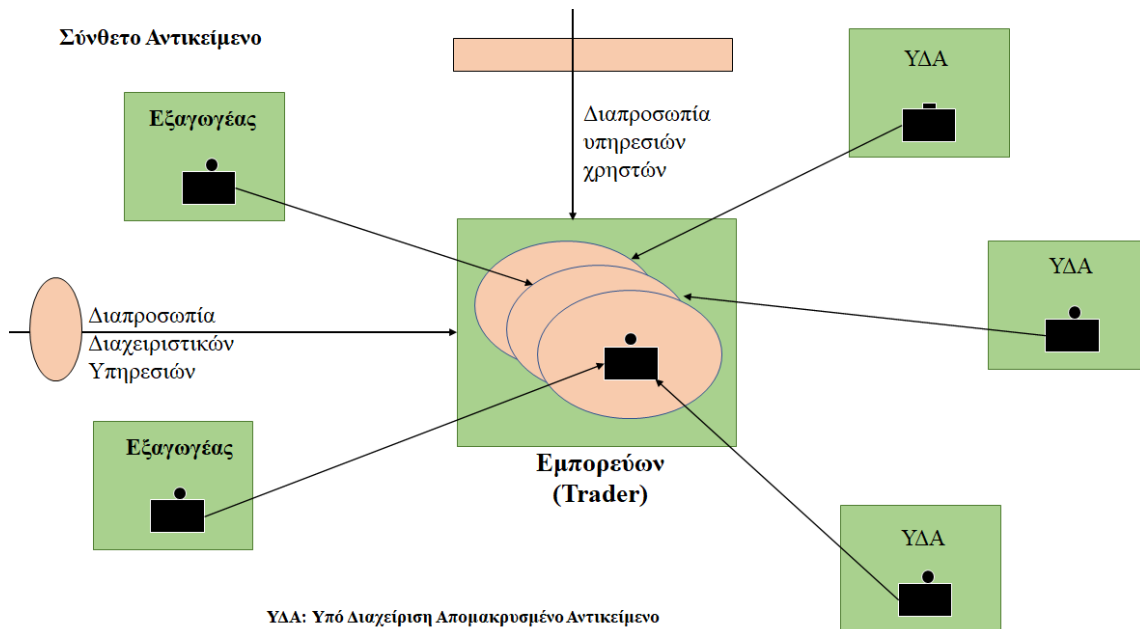
² <https://www.computerconservationsociety.org/ansa/91/RC25300.pdf> (πρόσβαση 10/9/2021)

Εάν υπάρχει υπηρεσία που να ταιριάζει στην αίτηση του εισαγωγέα, τότε ο Εμπορεύοντας αναφέρει την υπηρεσία του εξαγωγέα στον εισαγωγέα. Αυτή είναι η διαδικασία της εμπορίας. Ακολουθεί η διαδικασία της δέσμευσης (binding), ώστε να αναπτυχθεί ο κατάλληλος σύνδεσμος μεταξύ εισαγωγέα και εξαγωγέα. Αυτή η διαδικασία είναι ανάλογη με αυτή που περιγράφηκε στο πρώτο κεφάλαιο για την παροχή και κατανάλωση υπηρεσιών και το μητρώο εγγραφής των υπηρεσιών στις υπηρεσιοστρεφείς αρχιτεκτονικές, παρατηρώντας ότι η κατανεμημένη και δικτυοκεντρική φύση των υπηρεσιών εφαρμογών είναι αυτή που κυριαρχεί στα σύγχρονα πληροφοριακά συστήματα.



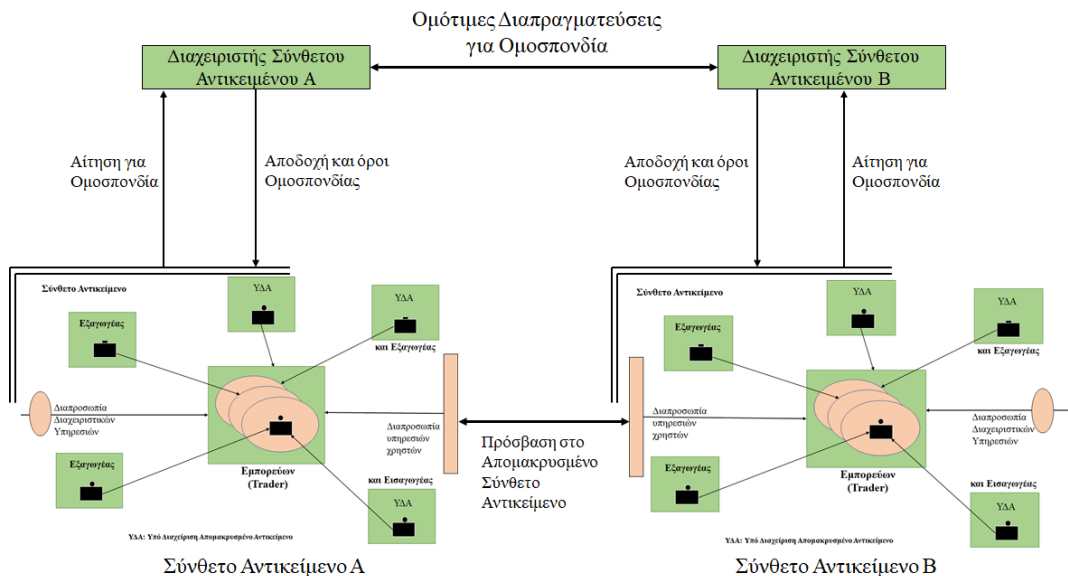
Σχήμα 2.4. Η λειτουργία της Εμπορίας (Trading)

Ας υποθέσουμε ότι έχουμε μια κατανεμημένη εφαρμογή που αποτελείται από ένα σύνολο κατανεμημένων αντικειμένων. Τα κατανεμημένα αντικείμενα παρέχουν την εικόνα ενός ενιαίου συστήματος, οπότε μπορούμε να θεωρήσουμε την εφαρμογή ως ένα σύνθετο αντικείμενο, όπου τα εμπεριεχόμενα αντικείμενα είναι φυσικά κατανεμημένα στους διάφορους υπολογιστικούς κόμβους του δικτύου. Στο Σχήμα 2.5 δείχνουμε έναν τρόπο υλοποίησης ενός κατανεμημένου σύνθετου αντικειμένου.



Σχήμα 2.5: Μια Κατανεμημένη Σύνθετη Υπηρεσία μέσω του Εμπορεύοντα

Εάν θεωρήσουμε ότι το κάθε σύνθετο αντικείμενο (εφαρμογή) διαθέτει έναν δικό του εμπορεύοντα (trader), έχοντας καταγεγραμμένες όχι μόνο τις εσωτερικά προσφερόμενες υπηρεσίες, αλλά και τις υπηρεσίες που προσφέρει το σύνθετο αντικείμενο στο εξωτερικό του περιβάλλον, τότε χρειαζόμαστε μια ομοσπονδία (federation) από traders, ώστε οι υπηρεσίες που είναι στην δικαιοδοσία του ενός, να περνάνε και σε αυτές των άλλων. Αυτό απαιτεί την ύπαρξη κάποιου διαχειριστή - συντονιστή. Οπότε ο διαχειριστής του κάθε εμπορεύοντα ζητά να πραγματοποιήσει μια ομοσπονδία (federation request) με κάποιον άλλο διαχειριστή. Όταν γίνει η ομοσπονδία, τότε μπορεί να γίνονται αναφορές (references) σε εξαγωγείς ενός άλλου trader. Το σημαντικό είναι ότι κάθε εμπορεύοντα προσφέρει μόνο τις παρεχόμενες προς το εξωτερικό περιβάλλον υπηρεσίες, ενώ τις άλλες τις κρύβει. Λόγοι ομοσπονδιακής εμπορίας (federated trading) είναι η επίτευξη μιας κατανεμημένης και ιεραρχικής διαχείρισης.

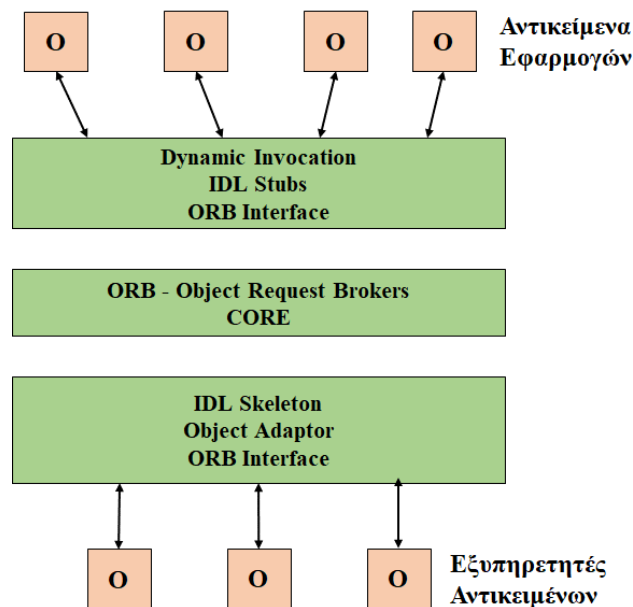


Σχήμα 2.6: Σχήμα Ομοσπονδία Κατανεμημένων Σύνθετων Αντικειμένων

Η δέσμευση (*binding*) είναι η λογική διασύνδεση σε επίπεδο αντικειμένων. Με αυτή την λειτουργία μπορεί να υπάρξει διαμόρφωση (*configuration*) συστημάτων αντικειμένων. Όπως ήδη αναφέραμε μετά την εμπορία (*Trading*) ακολουθεί η δέσμευση μεταξύ εισαγωγέα και εξαγωγέα χρησιμοποιώντας τους υποκείμενους μηχανισμούς επικοινωνίας, π.χ. την απομακρυσμένη κλήση διαδικασιών (*Remote Procedure Call*). Παράλληλα με την δέσμευση υπάρχουν οι λειτουργίες επαναδέσμευσης (*rebinding*) και αποδέσμευσης (*unbinding*) για την αλλαγή της δομής ενός συστήματος.

2.3.2 Υπηρεσίες Περιγραφής, Κλήσης και Προσαρμογής Κατανεμημένων Αντικειμένων

Η αντικειμενοστρεφής προσέγγιση είναι η επικρατούσα τεχνοτροπία σχεδίασης των κατανεμημένων συστημάτων γιατί περιλαμβάνει τις έννοιες της τμηματοποίησης (*modularity*), της επικάλυψης (*encapsulation*) και της κληρονομικότητας (*inheritance*). Αυτά τα χαρακτηριστικά βελτιώνουν την επαναχρησιμοποίηση, την επεκτασιμότητα, τη διαμόρφωση και τη συντήρηση των συστημάτων. Ακολουθώντας την αντικειμενοστρεφή προσέγγιση, αναπτύχθηκε η ανάγκη για Υπηρεσίες Κλήσης Αντικειμένων, για περιγραφή των κλάσεων, για δημιουργία, διαγραφή, εκκίνηση και σταμάτημα αντικειμένων. Οι Μεσίτες Αιτήσεων σε Αντικείμενα (*Object Request Brokers - ORB*), , κάνουν διαφανή ως προς την θέση τη διαδικασία κλήσης τοπικών ή απομακρυσμένων αντικειμένων. Οι προσαρμοστές αντικειμένων (*Object Adaptors*) προσαρμόζουν τις κλήσεις στον απαιτούμενο κάθε φορά τύπο αντικειμένων (Σχήμα 2.8).



Σχήμα 2.8: Μεσίτες Αιτήσεων σε Εξυπηρετητές Αντικειμένων (Object Request Brokers)

Τα αντικείμενα είναι παθητικές οντότητες οι οποίες φιλοξενούνται από έναν εξυπηρετητή. Ο κάθε εξυπηρετητής χρησιμοποιεί ένα ORB για βρει τη φυσική θέση των αντικειμένων. Έτσι, οι λειτουργίες δεν στέλνονται άμεσα στα αντικείμενα, και δεν απασχολείται ο χρήστης τους με την πραγματική διεύθυνσή τους. Η προσέγγιση OMG της CORBA υποστηρίζει αυτόν τον τρόπο, αλλά υποστηρίζει και την δυναμική επίκληση, όπου η ταυτότητα του αντικειμένου περνάει ως παράμετρος. Υπάρχει μια σειρά από αρχιτεκτονικές, οι οποίες υποστηρίζουν κλήση και διαχείριση αντικειμένων. Στην αρχιτεκτονική CORBA, μπορεί να γίνουν υλοποιήσεις στη βάση του πελάτη και του εξυπηρετητή, περιγράφοντας τις διαπροσωπείες (*interfaces*) σε μία γλώσσα όπως είναι η IDL γλώσσα ή C++.

Ως παραδείγματα συστημάτων καταναμημένων αντικειμένων μπορούμε να δούμε, πέρα από τα RM-ODP και CORBA και το Java RMI το οποίο θα εξετάσουμε εκτενώς στο επόμενο κεφάλαιο. Άλλες τεχνολογίες που εμπλέκονται στα καταναμημένα αντικείμενα είναι τα lightweight components (π.χ. Fractal, OpenCOM), οι εξυπηρετητές εφαρμογών - αντικειμένων, όπως τα SUN EJB, JBoss, Tomcat, Glassfish, τα συστήματα εγγραφής και δημοσίευσης αντικειμένων, όπως τα CORBA Event Service, Scribe και JMS (Java Messaging System), τα Web Services (π.χ. Apache Axis) και Grid Services (π.χ. Globus), τα peer-to-peer πρωτόκολλα³, όπως τα Pastry, Ivy, OceanStore, Ivy και Squirrel.

2.3.3 Διαφάνειες των Καταναμημένων Συστημάτων

Σε αυτό το υποκεφάλαιο θα παρουσιάσουμε τις διαφάνειες των καταναμημένων συστημάτων κατά το μοντέλο αναφοράς ODP (ODP-RM). Οι τύποι διαφάνειας χωρίζονται ανάλογα με τις ιδιότητές τους. Παρότι οι διαφάνειες στα καταναμημένα συστήματα, είναι βασικός στόχος τους, αυτές δεν υλοποιούνται σε όλες τις περιπτώσεις στην πλήρη έκτασή τους διότι απαιτούνται για κάθε μία διαφάνεια μηχανισμοί, οι οποίοι μπορεί να μην προσφέρονται από την υποκείμενη υπολογιστική πλατφόρμα.

Κατά το πρότυπο ODP-RM, λοιπόν, οι διαφάνειες είναι οι εξής:

- *Τοποθέτησης (Location)*, η οποία αφαιρεί την ανάγκη για γνώση της θέσης των διαπροσωπειών τους, ήτοι αναφέρεται σε προσπέλαση των αντικειμένων - πόρων δίχως τη γνώση των θέσεων τους
- *Επανατοποθέτησης (Relocation)*, η οποία αποκρύπτει την επανατοποθέτηση μιας διαπροσωπείας αντικειμένου από αντικείμενα που είναι συνδεδεμένα με αυτή (την χρησιμοποιούν).
- *Μετακίνησης (Migration)*, η οποία αποκρύπτει τις μετακινήσεις αντικειμένων σε διάφορες θέσεις. Να σημειωθεί ότι η μετακίνηση ενός πόρου χωρίς αλλαγή του τρόπου προσπέλασης του καλείται και μετανάστευση, ενώ η μετακίνηση ενός πόρου κατά τη διάρκεια της χρήσης του καλείται μετάθεση.
- *Προσπέλασης (access)*, η οποία συσχετίζεται με τον τρόπο προσπέλασης των πόρων και υπηρεσιών, προκειμένου αυτή να πραγματοποιείται με ενοποιημένο τρόπο. Πιο πρακτικά, αποκρύπτει τις διαφορές στην αναπαράσταση των δεδομένων, καθώς και τους μηχανισμούς κλήση των αντικειμένων. Υποστηρίζεται από τα αντικείμενα stub.
- *Αποτυχίας (Failure)*, η οποία αποκρύπτει από κάποιο αντικείμενο ή κάποια υπηρεσία τις διαδικασίες που εξασφαλίζουν ανοχή σφαλμάτων για αυτό το αντικείμενο-υπηρεσία. Η διαφάνεια αυτή αποκρύπτει τις αποτυχίες από ένα σύστημα, καθώς και την πιθανή ανάκτηση (recovery) των αντικειμένων – υπηρεσιών.
- *Διατήρησης (Persistence)*, η οποία αποκρύπτει από ένα αντικείμενο τη χρήση της λειτουργίας απενεργοποίησης και επανεργοποίησης, η οποία μεταβάλλει τους πόρους επεξεργασίας, αποθήκευσης και επικοινωνίας που παρέχονται σε ένα αντικείμενο.
- *Επανάληψης (Replication)*, η οποία αποκρύπτει την υποστήριξη μιας διαπροσωπείας από ένα σύνολο αντικειμένων (και όχι μόνο από ένα). Αυτός ο τύπος διαφάνειας αναφέρεται και ως αναπαραγωγή διότι βασίζεται στην ύπαρξη πολλαπλών αντίγραφων των πόρων. Η αναπαραγωγή κρύβει το γεγονός ότι ένα αντικείμενο ή η κατάσταση του μπορεί να αναπαραχθεί και ότι τα αντίγραφα του βρίσκονται σε διαφορετικές τοποθεσίες.

³ <https://www.csie.nuk.edu.tw/~wuch/course/csd511/csd511-10.pdf> (πρόσβαση 10/9/2021)

- *Δοσοληψίας (Transaction)*, η οποία αποκρύπτει τις ενέργειες μεταξύ αντικειμένων, προκειμένου να επιτευχθεί συνέπεια στην εκτέλεση μιας δοσοληψίας.
- *Ταυτοχρονισμού (parallelism)*, η οποία αναφέρεται στην απόκρυψη της παράλληλης χρήσης του ίδιου πόρου. Επίσης, αυτή η διαφάνεια αποκρύπτει το απαιτούμενο συντονισμό (*synchronisation*) των δραστηριοτήτων μεταξύ αντικειμένων για την επίτευξη συνέπειας.

Όπως αναφέραμε παραπάνω, ίσως αποτελεί υπερβολή ο στόχος επίτευξης μιας πλήρους κατανεμημένης διαφάνειας. Παραδείγματος χάριν, το κρύψιμο των αποτυχιών των κόμβων ενός δικτύου είναι πρακτικά και θεωρητικά αδύνατο. Είναι πολύ δύσκολο κάποιος να διακρίνει έναν αργό υπολογιστή από έναν υπολογιστή που έχει αστοχία υλικού. Επίσης, δεν μπορεί ο χρήστης να είναι σίγουρος ότι ένας εξυπηρετητής απέτυχε αφότου εκτέλεσε το καθήκον που του ανάθεσε. Η πλήρης διαφάνεια, επιπλέον, θα κόστιζε στην επίδοση του συστήματος, π.χ. λόγω της συνεχούς ενημέρωσης των αντιγράφων για λόγους συνέπειας.

2.3.4 Πρόσθετες Λειτουργίες επί των Κατανεμημένων Αντικειμένων

Τα κατανεμημένα αντικείμενα απαιτούν ένα σύνολο άλλων υπηρεσιών απαραίτητων για την λειτουργία ενός κατανεμημένου συστήματος. Ορισμένες από τις λειτουργίες αυτές μπορεί να προσφέρονται ως αυτόνομες υπηρεσίες ενώ άλλες από τα ίδια τα αντικείμενα. Οι λειτουργίες αυτές είναι:

- Διαχείριση αντικειμένων τύπου stub
- Διαχείριση αντικειμένων τύπου binder
- Διαχείριση αντικειμένων τύπου protocol
- Δημιουργία και διαγραφή αντικειμένων
- Διαχείριση αιτήσεων λειτουργιών (operations requests)
- Διαχείριση Πρόσβασης ελέγχου (access control)
- Διαχείριση εμβέλειας (scoping)
- Διαχείριση φιλτραρίσματος (filtering)
- Διαχείριση συγχρονισμού (synchronization)
- Διαχείριση διάδοσης των ειδοποιήσεων (notification dissemination)
- Διαχείριση αποστολών από τον διανεμητή (dispatcher)

2.3.5 Υπηρεσίες Επικοινωνίας

Στο κατώτατο επίπεδο της αρχιτεκτονικής των κατανεμημένων συστημάτων στέκονται οι υπηρεσίες και οι μηχανισμοί επικοινωνίας. Παραδείγματα τέτοια αποτελούν ο μηχανισμός της Απομακρυσμένης Κλήσης Διαδικασιών (Remote Procedure Call - RPC), οι μηχανισμοί που αναφέρονται στο ISO/OSI stack και το TCP/IP. Θα πρέπει να υποστηρίζονται και μηχανισμοί για πολλαπλή επικοινωνία, δηλαδή μηχανισμοί ταυτόχρονης επικοινωνίας με ένα σύνολο αντικειμένων (groupcast).

Αναλυτικότερα, οι υπηρεσίες επικοινωνίας προσφέρουν υψηλού επιπέδου επικοινωνία, διεργασίες μεταξύ ομάδων, ασύγχρονη επικοινωνία με ουρές μηνυμάτων και επικοινωνία μέσω ηλεκτρονικού ταχυδρομείου (email). Οι υπηρεσίες επικοινωνίας έπονται των υπηρεσιών εκτέλεσης δίνοντας τη δυνατότητα πρόσβασης σε απομακρυσμένα συστήματα, περιλαμβάνοντας εκτέλεση εντολών, χρήση τερματικού, κλήση διαδικασιών και μεθόδων αντικειμένων, καθώς και εκτέλεση διεργασιών.

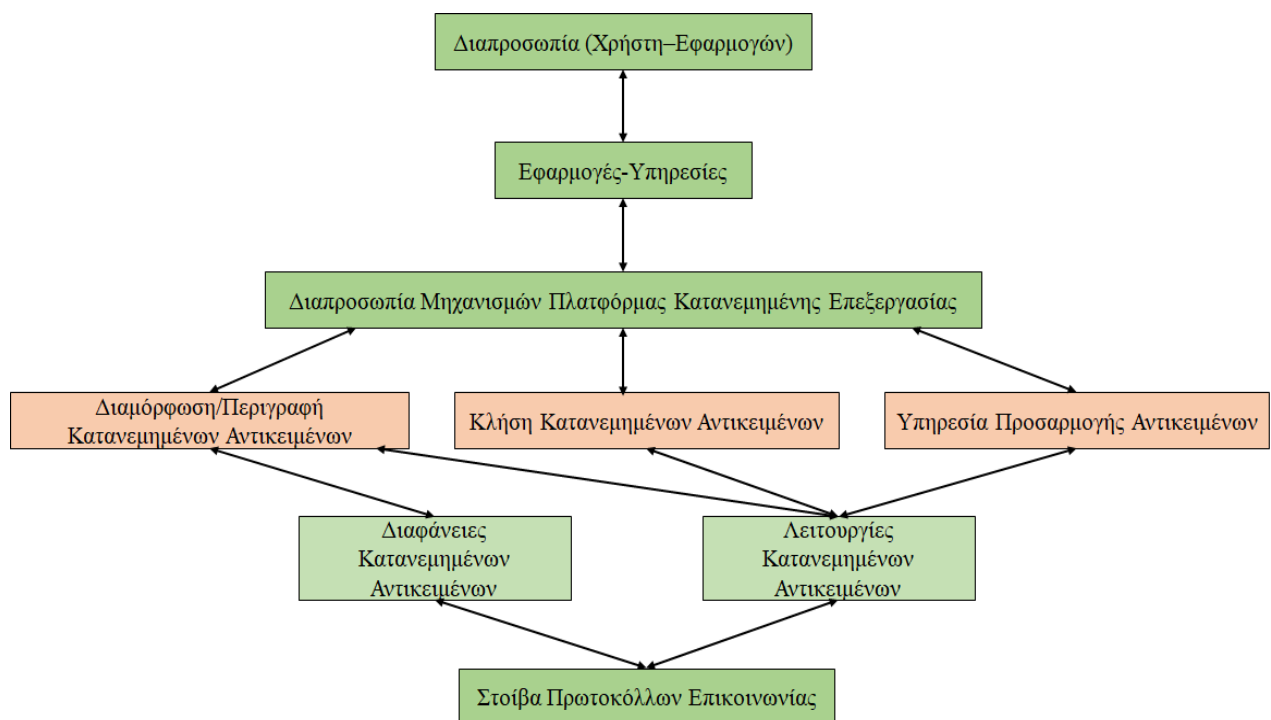
Στον Πίνακα 2.1. αναφέρονται κατηγορίες και παραδείγματα επικοινωνιών.

Οντότητες Επικοινωνίας	Μηχανισμοί/Τεχνολογίες
Υπολογιστικοί Κόμβοι (Nodes)	Πέρασμα Μηνυμάτων (Message Passing)
Διεργασίες (Processes)	Υποδοχείς (Sockets)
Κατανεμημένα Αντικείμενα (Distributed Objects)	Κλήση Μεθόδων (RMI)
Συστατικά Στοιχεία (Components/Elements)	Multicasting
Υπηρεσίες Ιστού (Web Services)	Request-Reply Protocol
	Remote Procedure Call (RPC)
	Group Communication/ Enqueued
	Ουρές Μηνυμάτων (Message queues)
	Publish (Post)

Πίνακας 2.1: Παραδείγματα επικοινωνούντων οντοτήτων και μηχανισμών/τεχνολογιών επικοινωνίας.

2.4 Ενοποιημένη Αρχιτεκτονική των Κατανεμημένων Πληροφοριακών Συστημάτων

Στο Σχήμα 2.9 βλέπουμε την συνολική αρχιτεκτονική των Κατανεμημένων Πληροφοριακών Συστημάτων σε μεγαλύτερη λεπτομέρεια, ως ένα σύνολο διασυνδεδεμένων λειτουργικών κομματιών (functional blocks).



Σχήμα 2.9: Αρχιτεκτονική Κατανεμημένων Συστημάτων και Υπηρεσιών

Παρατηρούμε τα παρακάτω λειτουργικά μπλοκ:

- Καταναμημένες Εφαρμογές
- Διαπροσωπείες Υποκείμενης Πλατφόρμας (Underlying Platform Interfaces)
- Μηχανισμοί Πλατφόρμας Καταναμημένης Επεξεργασίας (Distributed Processing Platform Mechanisms): Διαχείριση αρχείων και ευρετηρίων, ασφάλεια, trading κτλ
- Περιγραφή Καταναμημένων Αντικειμένων (Distributed Object Configuration)
- Υπηρεσία Κλήσης Αντικειμένων (Object Invocation Service), π.χ. CORBA, ANSA, Java RMI
- Υπηρεσία Προσαρμογής Αντικειμένων (Object Adaptation Service)
- Διαφάνειες Καταναμημένων Αντικειμένων (Distributed Object Transarencies), π.χ. διαφάνεια τοποθέτησης και μετακίνησης
- Λειτουργίες (Διαχείρισης) Καταναμημένων Αντικειμένων (Distributed Object Management Operations), π.χ. συγχρονισμός και φιλτράρισμα
- Επικοινωνία με Ομάδες Αντικειμένων (Object Group Communication)
- Στοιβά Επικοινωνιακού Πρωτοκόλλου (Communication Protocol Stack), π.χ. στοιβά OSI, στοιβά TCP/IP, ή RPC

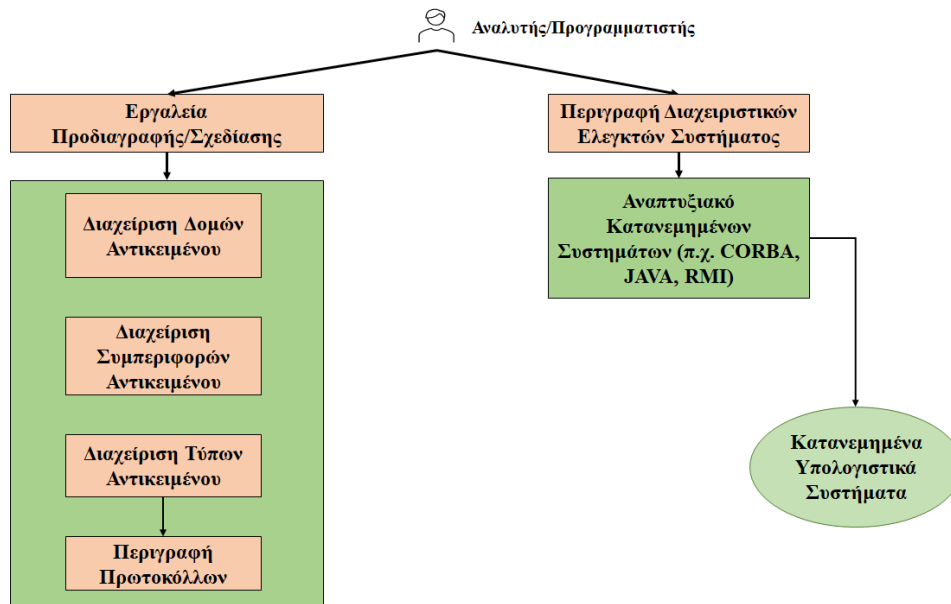
2.5 Σχεδίαση ενός Καταναμημένου Συστήματος: Περιγραφή, Ανοικτότητα και Κλιμάκωση

Έχοντας πλέον εξετάσει ενδελεχώς την αρχιτεκτονική των καταναμημένων συστημάτων μαζί με τους υποκείμενους μηχανισμούς – υπηρεσίες που θα πρέπει αυτή να υποστηρίζει, θα μιλήσουμε για την σχεδιαστική προσέγγιση των καταναμημένων συστημάτων. Το ερώτημα που τίθεται είναι τι υπηρεσίες πρέπει να διατίθενται ώστε να προδιαγραφεί ένα τέτοιο αντικειμενοστρεφές σύστημα, το οποίο περιέχει αντικείμενα εφαρμογών και σχέσεις (relations) μεταξύ τους.

Για τη σχεδίαση ενός *Καταναμημένου Πληροφοριακού Συστήματος* θα πρέπει να προσφέρονται κατ' ελάχιστον οι επόμενες υπηρεσίες:

- Υπηρεσία Διαχείρισης Δομών Αντικειμένων (object structures)
- Υπηρεσία Καθορισμού της Συμπεριφοράς των Αντικειμένων (behaviour of objects)
- Υπηρεσία Καθορισμού των Δεδομένων Αντικειμένων (data objects)
- Υπηρεσία Περιγραφής των Πρωτοκόλλων (protocol configuration), τα οποία καθορίζονται από τα δεδομένα
- Υπηρεσία Περιγραφής (λογικές συνδέσεις) του Συστήματος των Καταναμημένων Αντικειμένων

Κατά τη σχεδίαση ενός καταναμημένου συστήματος θα πρέπει να προκύπτουν συγκεκριμένα πακέτα (packages) υπηρεσιών προς τον χρήστη, όπως βλέπουμε στο Σχήμα 2.10. Χαρακτηριστικά πακέτα π.χ. είναι αυτά της περιγραφής των αντικειμένων και της διαχείρισής τους.



Σχήμα 2.10: Σχήμα Αρχιτεκτονική Υποστήριξης Σχεδίασης Κατανεμημένου Συστήματος

Επίσης, οι γενικές απαιτήσεις σχεδίασης ενός κατανεμημένου συστήματος απαιτούν ευελιξία, δηλαδή την ευκολία αλλαγής ή επέκτασής του, την ανοικτότητα, δηλαδή την υλοποίηση με δυνατότητα διαλειτουργούντων συστημάτων από διαφορετικούς προμηθευτές, η αποδοτικότητα, δηλαδή οι υψηλές ταχύτητες διεκπεραίωσης διεργασιών και, τέλος, η διαθεσιμότητα και η αξιοπιστία. Τα προηγούμενα καθορίζουν και την Ποιότητα Υπηρεσίας (Quality of Service).

Ένα ιδανικά υλοποιημένο ανοιχτό κατανεμημένο σύστημα προσφέρει μηχανισμούς οι οποίοι δίνουν τη δυνατότητα δυναμικής περιγραφής (configuration) του συστήματος, αλλά και των διαφόρων πολιτικών του, και υποστήριξης διαφορετικών επιπέδων εμπιστοσύνης. Επίσης, θα πρέπει να προσφέρει προσαρμοστικότητα στις τιμές των παραμέτρων ποιότητας υπηρεσίας και ασφάλειας.

Η ανοικτότητα κατά την σχεδίαση ενός σύγχρονου κατανεμημένου συστήματος είναι εκ των ουκ άνευ διότι μόνο έτσι είναι ικανό να αλληλεπιδρά με υπηρεσίες από ένα άλλο επίσης ανοιχτό σύστημα άσχετα από το περιβάλλον που λειτουργούν. Τα ανοικτά συστήματα έχουν συγκεκριμένα χαρακτηριστικά και θα πρέπει να υποστηρίζουν αποτελεσματικά τις καθορισμένες διεπαφές (διαπροσωπείες) διαλειτουργικότητας, καθώς και τη φορητότητα των εφαρμογών. Για να φτάσει το κατανεμημένο σύστημα σε αυτού του επιπέδου την ανοικτότητα ως ελάχιστη βάση θα πρέπει να είναι σχεδιασμένο ανεξάρτητα από την ετερογένεια του υποκείμενου περιβάλλοντος, δηλαδή το υλισμικό, τις πλατφόρμες και τις γλώσσες προγραμματισμού.

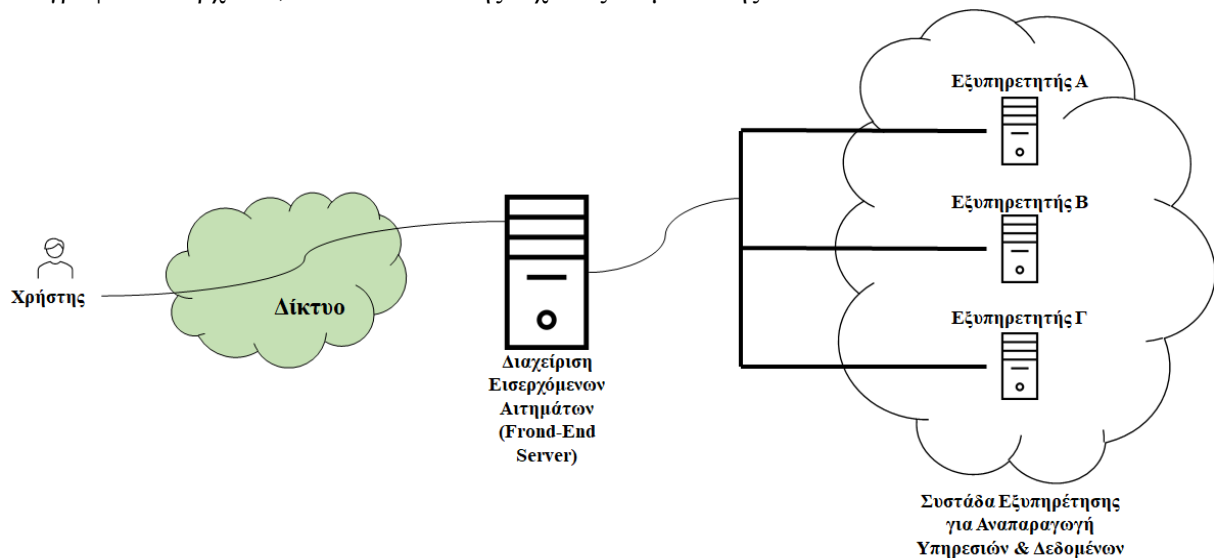
Η κλιμάκωση χωρίζεται σε δύο βασικές κατηγορίες, (α) την κλιμάκωση μεγέθους, δηλαδή την αύξηση υπολογιστικών μηχανών και χρηστών και (β) την γεωγραφική κλιμάκωση, δηλαδή την αύξηση των αποστάσεων.

Η κλιμάκωση μεγέθους αναφέρεται στις υπηρεσίες, τα δεδομένα και τους αλγόριθμους. Η γεωγραφική κλιμάκωση σχετίζεται άμεσα με την αύξηση των αποστάσεων, οπότε έχουμε καθυστέρηση διάδοσης των σημάτων, αντικατάσταση της σύγχρονης επικοινωνίας με ασύγχρονη, διάσπαρτη κατανομή των πόρων στο δίκτυο, και πιθανή αναπαραγωγή, αντιγραφή και ενταμίευση δεδομένων σε μνήμες.

Πολλοί προγραμματιστές των συγχρόνων καταναμημένων συστημάτων παρότι χρησιμοποιούν τον όρο κλιμακούμενο σύστημα εντούτοις δεν είναι ξεκάθαρο το τι ακριβώς εννοούν και προς ποια κατεύθυνση το σύστημα κλιμακώνεται. Συνήθως, η κλιμάκωση αυτή αναφέρεται σε τρεις κατευθύνσεις: (α) τον αριθμό των χρηστών, των διαδικασιών και των κόμβων (size scalability), (β) τη μέγιστη απόσταση μεταξύ των υπολογιστικών κόμβων (geographical scalability) και (γ) τον αριθμό των τομέων διαχείρισης (administrative scalability). Και οι τρεις αυτές κατευθύνσεις είναι σημαντικές στα καταναμημένα πληροφοριακά συστήματα.

Η κλιμάκωση απαιτεί την δυνατότητα διαχωρισμού των δεδομένων και των υπολογισμών σε πολλαπλές μηχανές (Σχήμα 2.11). Αυτό συνεπάγεται ότι θα πρέπει να υπάρχει εύκολη διασπορά των υπολογισμών σε εξυπηρετητές ή ακόμα και σε πελάτες, αποκέντρωση των ονομάτων υπηρεσιών (π.χ. με τη χρήση των υπηρεσιών του DNS (Domain Name System)) και αποκέντρωση των πληροφοριακών συστημάτων (π.χ. μέσω του διαδικτύου). Τα παραπάνω αποτελούν τεχνικές στην κατεύθυνση της κλιμάκωσης.

Η δημιουργία αντιγράφων δεδομένων σε διαφορετικές μηχανές είναι επίσης ένα κομμάτι της κλιμάκωσης. Η δημιουργία αντιγράφων σε εξυπηρετητές αρχείων, τα οποία χρησιμοποιούνται π.χ. στην περίπτωση ανάκαμψης από σφάλμα, η αντιγραφή βάσεων δεδομένων, το mirroring των ιστοσελίδων και η κατανομή μνήμης σε μεγάλης κλίμακας συστήματα και η προσωρινή αποθήκευση σε τοπικά αντίγραφα των αρχείων, αποτελούν επίσης τεχνικές κλιμάκωσης.



Σχήμα 2.11: Συστάδα εξυπηρετητών

Το πρόβλημα που υφίσταται στην κλιμάκωση σχετίζεται με τη σχεδίαση, η οποία δεν είναι τόσο εύκολη στην επίτευξη της. Έχοντας πολλαπλά αντίγραφα δημιουργείται μία ασυνέπεια όταν αντιγράφεται ένα αντίγραφο διαφορετικά από ό,τι τα υπόλοιπα. Οπότε δημιουργείται το πρόβλημα της διατήρησης αντιγράφων με συνέπεια και προς τούτο απαιτείται καθολικός συγχρονισμός για κάθε μετατροπή. Σχεδιαστικά και ανάλογα με την περίπτωση για λόγους ευελιξίας μπορεί να πρέπει να υπάρχει κάποια ανοχή σε τέτοιου είδους ασυνέπειες.

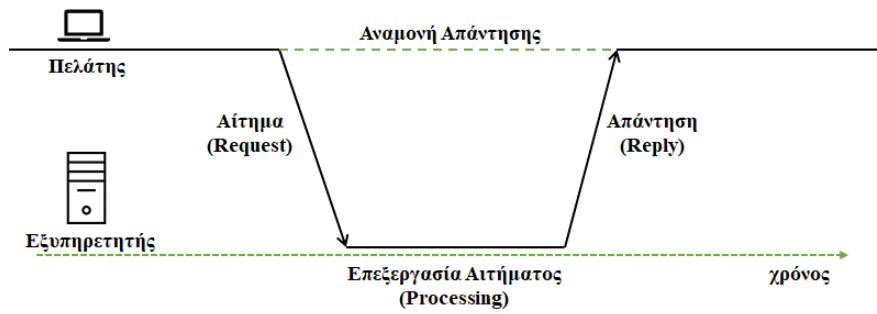
2.5 Οριζόντια Προσέγγιση της Αρχιτεκτονικής

Η οριζόντια αρχιτεκτονική προσέγγιση κινείται μεταξύ της επικοινωνίας δύο ή περισσότερων υπολογιστών χωρίς να υπεισέρχεται στις υπηρεσίες των υποκείμενων πλατφορμών και μηχανισμών και εστιάζει στο μοντέλο πελάτη-εξυπηρετητή. Το μοντέλο αυτό έχει ως κύριο σκοπό την επικοινωνία των διεργασιών διαμέσου ενός δικτύου με την αποστολή μηνυμάτων ή κλήσεων send and receive. Γι αυτό, το μοντέλο πελάτη - εξυπηρετητή διαθέτει λογική αφαίρεση για την επικοινωνία των διεργασιών. Ο ρόλος του υπολογιστή εξυπηρετητή είναι να παρέχει υπηρεσίες προς τους υπολογιστές πελάτες ενώ του πελάτη να ζητάει υπηρεσίες από τους εξυπηρετητές.

Για την υλοποίηση αυτής της αλληλεπίδρασης πελάτη-εξυπηρετητή απαιτείται η ανάπτυξη ενός σχετικού πρωτοκόλλου. Το πρωτόκολλο αυτό πρέπει να προβλέπει την αποστολή αίτησης από τον πελάτη στον εξυπηρετητή, την επεξεργασία της αίτησης από τον εξυπηρετητή και την απόκριση του τελευταίου προς τον πελάτη. Η επικοινωνία σε τοπικά δίκτυα πελάτη-εξυπηρετητή είναι αρκετά αξιόπιστη και δεν χρειάζονται περαιτέρω συνδέσεις. Η επικοινωνία, όμως, πελάτη-εξυπηρετητή δεν περιορίζεται μόνο στα τοπικά δίκτυα, αλλά μπορεί να εφαρμοστεί και σε δίκτυα ευρείας περιοχής. Ωστόσο αυτή η επικοινωνία είναι αναξιόπιστη και υπάρχει μεγάλη πιθανότητα απώλειας αιτήσεων του πελάτη ή αποκρίσεων του εξυπηρετητή. Το καλό με το μοντέλο πελάτη-εξυπηρετητή είναι ότι δίνει τη δυνατότητα αναμετάδοσης αιτήσεων σε περίπτωση μη εύλογης καθυστέρησης τους, οδηγώντας σε πιθανή επανάληψη της επεξεργασίας. Προς αποφυγή μιας τέτοιας επανάληψης απαιτείται η χρήση αξιόπιστων πρωτοκόλλων και αξιόπιστων ζεύξεων.

Στο μοντέλο πελάτη-εξυπηρετητή, όπου όπως είπαμε, μπορεί να συμμετάσχουν περισσότερες των δύο υπολογιστικών οντοτήτων, επικρατεί το μοντέλο των τριών επιπέδων, το οποίο διαχωρίζει τη διεπαφή, την επεξεργασία και την αποθήκευση των δεδομένων. Η Διεπαφή είναι η αλληλεπίδραση του χρήστη με μια εφαρμογή, όπως παραδείγματος χάρη ένα τερματικό κειμένου ή μια πλήρως γραφική διεπαφή. Η επεξεργασία αποτελεί τη γέφυρα μεταξύ των επιπέδων στο μοντέλο αυτό και εξαρτάται από τις απαιτήσεις της εκάστοτε εφαρμογής, ενώ αντλεί στοιχεία και δεδομένα, τα οποία επεξεργάζεται κατάλληλα. Η αποθήκευση αφορά στη διαχείριση των δεδομένων μιας εφαρμογής και είναι ανεξάρτητη από το επίπεδο επεξεργασίας. Αυτό το επίπεδο μπορεί να χρησιμοποιηθεί από πολλές εφαρμογές ταυτόχρονα, όπως π.χ. συμβαίνει με τα συστήματα αρχείων και τις βάσεις δεδομένων. Μια διεπαφή μπορεί να χωρίζεται σε δύο μέρη, σε αυτήν που βρίσκεται στον πελάτη και σε αυτήν που βρίσκεται στον εξυπηρετητή. Όταν η διεπαφή εγκαθίσταται στην μεριά του πελάτη έχουμε την απλούστερη μέθοδο κατανομής ενώ όταν η διεπαφή βρίσκεται στον εξυπηρετητή υπάρχει έλεγχος της διεπαφής κεντρικά. Όταν μέρος της επεξεργασίας λαμβάνει χώρα στον πελάτη τότε λέμε ότι έχουμε τοπική επεξεργασία των δεδομένων της εφαρμογής και παραγωγή αποτελεσμάτων. Μέρος της αποθήκευσης στο πελάτη αποτελεί η ενταμίευση σε κρυφή μνήμη.

Το μοντέλο τριών επιπέδων αποτελεί μια οριζόντια κατανομή της αρχιτεκτονικής. Μιλάμε για οριζόντια κατανομή διότι εκτελείται ένα λογικό καθήκον (task) σε πολλαπλές μηχανές. Το βασικό μοντέλο πελάτη - εξυπηρετητή έχει συγκεκριμένα χαρακτηριστικά, όπως: τις διαδικασίες που προσφέρουν υπηρεσίες (εξυπηρετητές), και εκείνες που χρησιμοποιούν υπηρεσίες (πελάτες). Πελάτες και εξυπηρετητές μπορεί να κατανεμηθούν σε διαφορετικές μηχανές, χωρίς αυτό να είναι υποχρεωτικό. Ουσιαστικά οι πελάτες κάνουν αιτήσεις και περιμένουν απαντήσεις χρησιμοποιώντας υπηρεσίες που προσφέρουν οι εξυπηρετητές (Σχήμα 2.12).

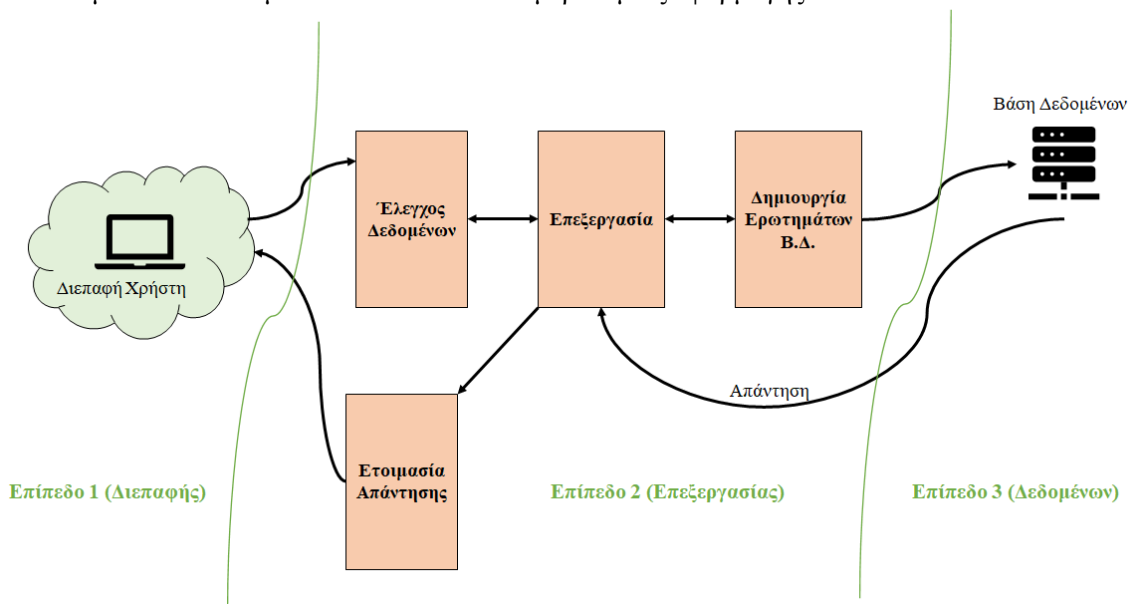


Σχήμα 2.12: Πελάτης - Εξυπηρετητής

Οι εξυπηρετητές προσφέρουν υπηρεσίες που σχετίζονται με ένα σύνολο διαμοιραζόμενων πόρων, όπως είναι τα συστήματα αρχείων, οι βάσεις δεδομένων, οι υλοποιήσεις αποθετηρίων, οι διαμοιρασμένες εφαρμογές και οι ιστοσελίδες, καθώς και διαμοιρασμένα καταναμημένα αντικείμενα. Από την άλλη μεριά, ένας πελάτης θα πρέπει να διαθέτει κατάλληλους μηχανισμούς για την απομακρυσμένη πρόσβαση σε υπηρεσίες, όπως π.χ. ένα προγραμματιστικό περιβάλλον για την μεταφορά τοπικών κλήσεων υπηρεσιών του πελάτη σε μηνύματα αιτήσεων και απαντήσεων, ή υπολογιστικό περιβάλλον που προσφέρει ανεξάρτητες διεπαφές στους χρήστες του για τις διάφορες υπηρεσίες.

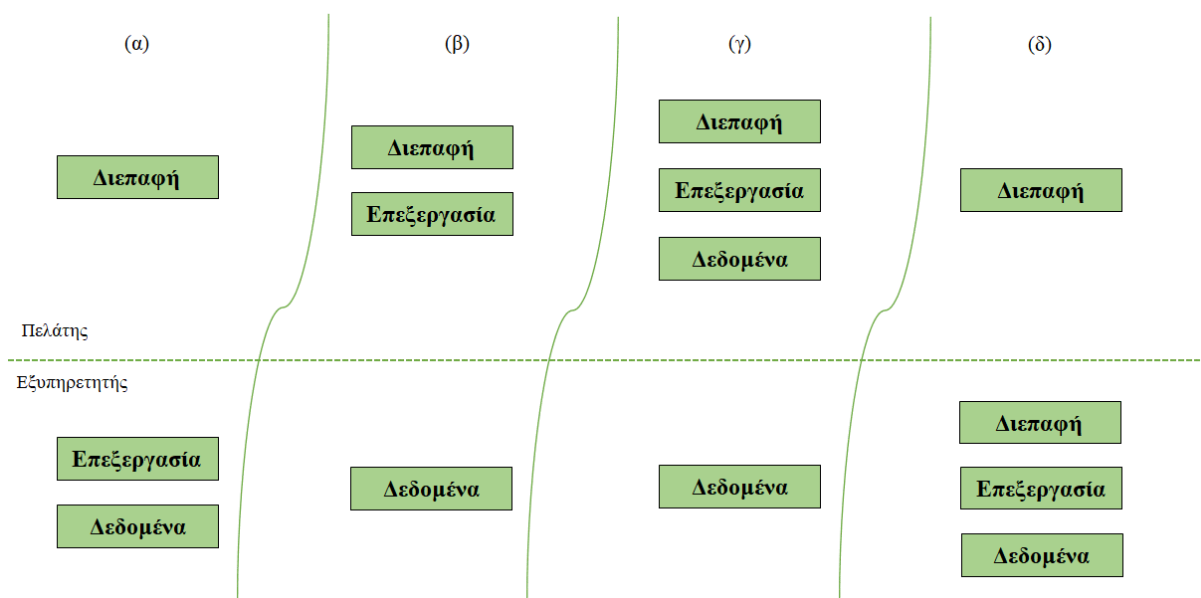
Η αρχιτεκτονική προσέγγιση των τριών στρωμάτων (3-tier architecture), η οποία έχει εδραιωθεί σχεδόν στο σύνολο των σύγχρονων πληροφοριακών συστημάτων, συνίσταται από (Σχήμα 2.13):

1. τη διεπαφή χρήστη, η οποία ως επίπεδο (στρώμα) περιέχει τις λειτουργικές μονάδες εκείνες που απαιτούνται για την δόμηση της διεπαφής χρήστη – εφαρμογής.
2. το επίπεδο επεξεργασίας το οποίο περιέχει της διαδικασίες - μεθόδους της εφαρμογής για την επεξεργασία των δεδομένων εισόδου χωρίς όμως κάποια συγκεκριμένα δεδομένα.
3. το επίπεδο δεδομένων, το οποίο περιέχει τα δεδομένα τα οποία οι πελάτες θέλουν να εκμεταλλευτούν μέσω των συστατικών μερών μιας εφαρμογής.



Σχήμα 2.13: Η αρχιτεκτονική τριών επιπέδων (3-Tier)

Αρχικά η αρχιτεκτονική πελάτη-εξυπηρετητή ήταν ενός επιπέδου αποτελούμενη από ένα τερματικό dummy σε μια διαμόρφωση mainframe. Κατόπιν εξελίχθηκε σε αρχιτεκτονική δύο επιπέδων, όπου έχουμε την διαμόρφωση πελάτη με έναν μεμονωμένο διακομιστή, και τέλος φθάσαμε στην αρχιτεκτονική τριών επιπέδων, όπου τα συστατικά μέρη του κάθε επιπέδου εγκαθίσταται σε ξεχωριστό μηχάνημα. Βέβαια τώρα μιλάμε για επέκταση σε πολυ-επίπεδες (N-tier) αρχιτεκτονικές, όπου το κάθε επίπεδο μπορεί να βρίσκεται σε διαφορετική μηχανή. Οι πολυ-επίπεδες αρχιτεκτονικές φαίνεται να αποτελούν λέξεις-κλειδιά που δεν αποτυγχάνουν να υλοποιήσουν πολλές σύγχρονες οργανώσεις πελατών-διακομιστών. Μια σύγχρονη υπηρεσία θα πρέπει να δύναται να κατανέμεται σε ένα σύνολο εξυπηρετητών προκειμένου να έχουμε μια πολυ-επίπεδη αρχιτεκτονική. Παράδειγμα αποτελούν και τα κατανεμημένα συστήματα αρχείων, τα κατανεμημένα συστήματα ονομασίας, όπως το DNS, τα συστήματα ροής εργασιών και συνεργατικότητας, τα οποία μπορεί να εμπλέκουν πολλούς υπολογιστικούς κόμβους. Στην κατεύθυνση των τελευταίων συστημάτων κινούνται και τα υπηρεσιοστρεφή συστήματα, όπως είναι το Enterprise Service Bus (ESB), το οποίο είναι πολυ-επίπεδο.



Σχήμα 2.14: Διάφορες κατανομές των επιπέδων μεταξύ πελάτη και εξυπηρετητή.

Βιβλιογραφικές αναφορές

- [1] Andrew S. Tanenbaum, Maarten Van Steen (2006), Κατανεμημένα Συστήματα: Αρχές και Υποδείγματα, Έκδοση: 1η/2006, Εκδόσεις Κλειδάριθμος
- [2] Coulouris, J. Dollimore, T. Kindberg, G. Blair (2020), Κατανεμημένα Συστήματα, Έκδοση: 2η, DA VINCI Μ.Ε.Π.Ε.
- [3] Κάβουρας Ι.Κ., Μήλης Ι.Ζ., Ρουκουνάκη Α.Α., Ξηλωμένος Γ.Β. (2011), Κατανεμημένα συστήματα σε Java, 3η έκδοση, Εκδόσεις Κλειδάριθμος ΕΠΕ.
- [4] Kilov, H., Linington, P.F., Romero, J.R., Tanaka, A., Vallecillo, A. (2013), The reference model of open distributed processing: foundations, experience, and applications. Comput. Stand. Interfaces 35, 247–256
- [5] Schill, Alexander (Ed.) (1993), DCE - The OSF Distributed Computing Environment, Client/Server Model and Beyond, Proceedings of International DCE Workshop, Karlsruhe, Germany.

- [6] J.F. Buford, H. Lu, E.K. Lua, (2009) “P2P Networking and Applications”, Morgan Kaufman.
- [7] R. Steinmetz, K. Wehrle (eds), (2005) “Peer-to-Peer Systems and Applications”, LNCS 3485, Springer.
- [8] N. A. Lynch, (1996) Distributed Algorithms, Morgan Kaufmann Publishers, Inc., ISBN 1558603484.
- [9] V. C. Barbosa, (2003) An Introduction to Distributed Algorithms, The MIT Press, ISBN 0262024128.
- [10] G. Tel, (2000) Introduction to Distributed Algorithms, 2nd Edition, Cambridge University Press, ISBN 0521794838.
- [11] H. Attiya and J. Welch, (2004) Distributed Computing Fundamentals, Simulations, and Advanced Topics, 2nd Edition, Wiley-Interscience, ISBN 0471453242.
- [12] M. Raynal, (2013) Distributed Algorithms for Message-passing systems, Springer-Verlag.
- [13] D. Comer, D. Stevens, (2005) Δικτυακός Προγραμματισμός - Ανάπτυξη Εφαρμογών Πελάτη-Εξυπηρετητή στην Οικογένεια Πρωτοκόλλων TCP/IP, ελληνική μετάφραση, Ιων, ISBN: 978-960-411-537-2, Κωδικός Βιβλίου στον Εύδοξο: 14504

Κριτήρια αξιολόγησης

Ερώτηση 1

Τι προσφέρει το πρότυπο ISO/OSI;

α. Πρωτόκολλα επικοινωνίας

β. Πρωτόκολλα υγείας

γ. Πρότυπα επεξεργασίας δεδομένων

δ. Όλα τα παραπάνω

Ερώτηση 2

Ποιο από τα παρακάτω υιοθετεί τις ιδιότητες του αντικειμενοστρεφούς προγραμματισμού;

α. Το μοντέλο πελάτη-εξυπηρετητή

β. Οι κλήσεις RPC

γ. Η διαστρωμάτωση

δ. Το πρότυπο ODP

Ερώτηση 3

Τι είναι ένα πόρος (resource) στη σχεδίαση των καταναμημένων συστημάτων;

α. Ένα μοντέλο υλοποίησης των Κ.Σ.

β. Ένα αρχιτεκτονικό επίπεδο

γ. Μια τεχνική κλιμάκωσης

δ. Ένα συστατικό ενός Κ.Σ. το οποίο προσφέρει διαπροσωπείες στις ιδιαίτερες λειτουργίες του

Ερώτηση 4

Μπορεί ένα καταναμημένο αντικείμενο να προσφέρει τις παρακάτω διαπροσωπείες ταυτόχρονα;

- α. Ναι
- β. Όχι
- γ. Και Ναι και Όχι
- δ. Απροσδιόριστο

Ερώτηση 5

Ποιες από τις παρακάτω υπηρεσίες πρέπει να προσφέρει ένα Κ.Σ;

- α. Υπηρεσία ετερογένειας
- β. Υπηρεσία αποκλιμάκωσης
- γ. Υπηρεσία διαφάνειας τοποθεσίας**
- δ. Όλα τα παραπάνω

Ερώτηση 6

Ποιο από τα παρακάτω είναι βασικό χαρακτηριστικό της κλιμάκωσης σε ένα Κ.Σ;

- α. η δυνατότητα αύξησης του αριθμού των χρηστών του Κ.Σ.
- β. η δυνατότητα αύξησης της απόστασης μεταξύ των υπολογιστικών κόμβων
- γ. η αύξηση του αριθμού πεδίων διαχείρισης
- δ. Όλα τα παραπάνω**

Ερώτηση 7

Πώς επιτυγχάνεται η επικοινωνία στο μοντέλο πελάτη-εξυπηρετητή;

- α. Μέσω ολοκληρωμένων κυκλωμάτων
- β. Μέσω αποστολής μηνυμάτων και κλήσεων send και receive**
- γ. Μέσω ταχυδρομικού περιστεριού
- δ. Όλα τα παραπάνω

Ερώτηση 8

Η τριών επιπέδων αρχιτεκτονική σε ποια επίπεδα αφορά;

- α. Οριζόντιο, κάθετο, κεκλιμένο
- β. Διεπαφής, επεξεργασίας, πληροφορικής
- γ. Διεπαφής, επεξεργασίας, δεδομένων**
- δ. Όλα τα παραπάνω

Ερώτηση 9

Οι υπηρεσίες εντοπισμού με ποια από την παρακάτω υπηρεσίες συνεργάζονται;

- α. Υπηρεσίες ερευτηρίου
- β. Υπηρεσίες ονοματολογίας
- γ. Υπηρεσίες Κινητικότητας
- δ. Όλες τις παραπάνω**

Ερώτηση 10

Ποια από τις παρακάτω δεν είναι υπηρεσία διαφάνειας;

- α. Μετακίνησης πόρου
- β. Προσπέλασης πόρου
- γ. **Κατασκευής πόρου**
- δ. Αποτυχίας πόρου