**Solutii de comunicare intre sistemele Macintosh: AppleTalk**

1. **Scurt istoric al protocoalelor Apple (Tudor Cristina)**

Dupa lansarea computerului Apple Lisa in anul 1983, Apple a inceput investirea unui efort in dezvoltarea unui sistem de retea local (LAN) pentru masini. Cunoscut sub numele de AppleNet, el s-a bazat pe stiva de protocol Xerox XNS, dar folosea un sistem de cabluri coaxiale de 1Mbit/s, fata de Etherenet-ul Xerox de 4Mbit/s.

De-a lungul evolutiei inginerii au luat decizia de a folosi controller-ul chip serial (SCC) Zilog 8530 in locul variantei mai ieftine UART pentru a oferi conexiuni de port serial. SCC oferea viteza de pana la 250 kilobytes/s si oferea suporta un numar de protocoale de retea asemenatoare cu Bisync de la IBM.

SCC a fost ales pentru ca permitea mai multor dispozitive sa fie atasate la port. Perifericele echipate cu SCC-uri similare puteau comunica folosind protocoalele incorporate, intercalandu-si datele cu alte periferice de pe acelasi bus. Conceptul initial a fost cunoscut sub numele de AppleBus, intruchipand un sistem controlat de device-urile Macintosh de interogare ciclica intr-un mod similar cu modem-ul USB.

Noul AppleBus a fost anuntat la inceputul anului 1984, permitand conexiunea directa de pe Mac sau Lisa printr-o mica cutie conectata la un port serial si conectata prin cabluri la urmtorul computer din nivelul superior sau inferior. In acelasi timp, au fost anuntate adaptoarele pentru Apple II si Apple III.

O retea AppleTalk aparuta in anul 1985 putea suporta pana la 32 de echipamente conectate, schimbul de date se poate face la o viteza maxima de 230.4 Kbps., iar distanta maxima dintre echipamente poate atinge si 0,308 km.

Intreaga stiva de retea avea cerinte doar de 6kB de memorie RAM. Sistemul a fost gandit pentru viitoare expansiuni, astfel ca sistemul de adresare permitea expansiunea la 255 de noduri intr-un LAN, folosindu-se rutere.

Viteza relativ redusa a AppleTalk-ului a permis reduceri in plus asupra costului. Sistemul de cablare al retelei personale AppleTalk folosea o singura impamantare electrica obisnuita, care limita viteza la 500 kbit/s, dar permitea unui conector sa fie indepartat.

Adaptoarele au fost proiectate pentru a avea terminatii proprii, ceea ce inseamna ca nodurile de la capatul retelei puteau lasa ultimul conector neconectat. Astfel, nu mai era nevoie de hub-uri sau alte dispozitive de conectare, ci era doar necesar sa conectezi adaptorul la calculator si sa conectezi un cablu la cea mai apropiata masina cu un port liber.

Un alt adaptor, al carui design a fost realizat de catre BMUG, a fost PhoneNet, lansat in 1987. Acesta a fost un inlocuitor al conectorilor Apple care aveau mufe conventionale de telefoane in locul conectorilor rotunzi Apple. PhoneNet permitea retelelor AppleTalk sa fie conectate impreuna folosind cablu normal de telefon.

La jumatatea anului 1987 Apple a introdus EtherTalk 1.0 pentru noi calculatoare Macintosh II. Pachetul continea atat un card NuBus cu port Ethernet cat si un nou panou de control al retelei care permitea user-ului sa selecteze ce conexiune fizica sa folosesca pentru retea. Noua stiva de retea a extins sistemul pentru a permite 255 de noduri pe LAN. Astfel, reteaua personala AppleTalk s-a numit LocalTalk. [[1]](#footnote-1)

AppleTalk este set de protocoale de comunicatii pentru retele locale (LAN-uri), dezvoltate original de catre Apple Inc., pentru calculcatoarele Macintosh.

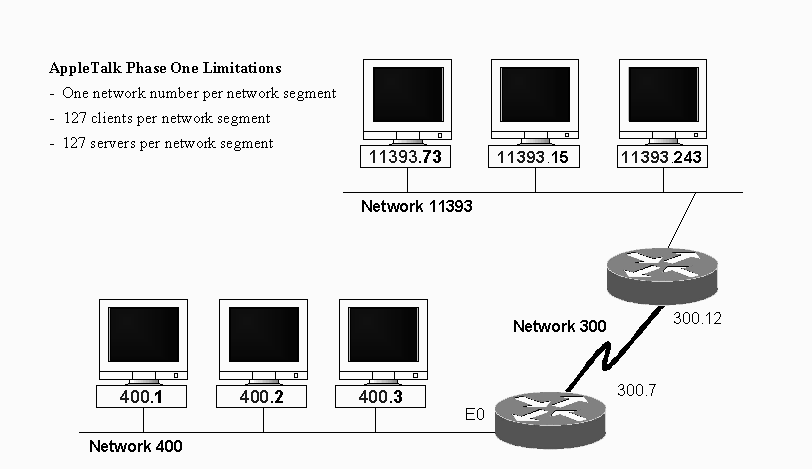
Scopul Appletalk-ului a fost de a permite mai multor useri sa se conecteze fara a fi nevoie de configurari anterioare sau de a folosi un router sau server centralizat, in vederea schimbului de resurse, cum ar fi fisiere sau imprimante. Conectand impreuna sisteme AppleTalk ele vor atribui automat automat adrese, vor updata spatiul distribuit al numelor, si vor configura orice rutare inter-retea ceruta.Astfel, AppleTalk se defineste ca un sistem plug-n-play.

Echipamentele care aprovizioneaza resursele se numesc servere, iar cele care beneficieaza de resurse se numeste clienti. Astfel, AppleTalk este una dintre primele implementari a sistemului de retele distribuite client/server.

A fost primul protocol folsoit de catre echipamentele Apple de-a lungul anilor 1980-1990. Alte versiuni au fost lansate pentru IBM PC si echipamente compatibile, cat si pentru Apple IIGS. Acest tip de protocol a fost valabil si in majoritatea retelelor de printare ( mai ales in cazul imprimantelor laser), serverelor de fisiere si pentru cateva rutere.[[2]](#footnote-2)

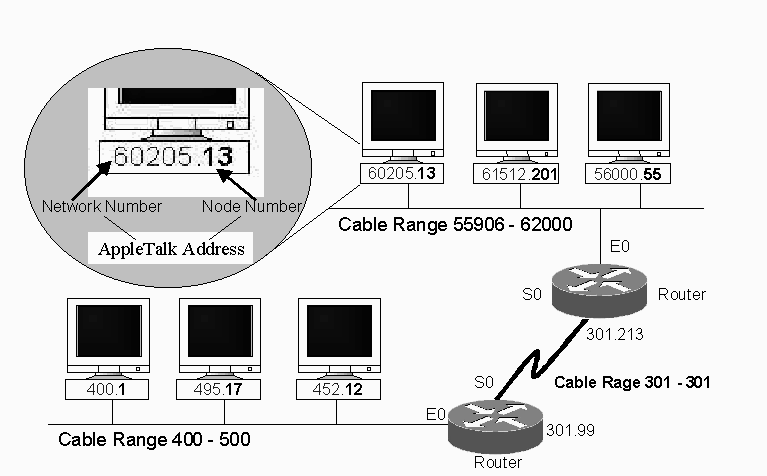
1. **Organizarea retelelor AppleTalk. Implementarea si tehnici de adresare (Tudor Cristina)**
   1. **AppleTalk faza 1 si AppleTalk faza 2**

Abordarea retelei faza 1 este orientata catre interconectarea grupurilor de lucru. Faza 1 sustin un diametru al retelei al unui singur segment al retelei care contine nu mai mult de 127 de clienti si 127 de servere. Retelele din faza 1 folosesc un singur numar de retea ( 0 ) pentru intreaga retea fizica. Retelele din faza 2 sustin mai multe retele logice pe acelasi segment de cablu. Fiecare retea logica suporta pana la 253 de clienti sau servere. [[3]](#footnote-3)



Modul de adresare in AppleTalk faza 1[[4]](#footnote-4)

Pentru a mentine compatibilitatea cu retelele din faza 1, retelele din faza 2 sustin doua configuratii diferite de retea: extinse si neextinse. Prin faza 2, o retea logica AppleTalk este definita prin distanta cablului. Distanta cablului este data de numarul sau numerele folosite de catre nodurile finale conectate la mijloacele de transmisie. Fiecare cablu AppleTalk poate sustine 253 de gazde. Dimesiunea distantei cablului determina numarul de gazde care pot fi conectate simultan la mijloacele de trasmisie. De asemenea, dimensiunea distantei cablului determina daca reteaua este de tip extins sau nu. [[5]](#footnote-5)



Modul de adresare in AppleTalk faza 2[[6]](#footnote-6)

O retea faza 2 neextinsa folsoeste distanta unui singur cablu si poate sustine 253 de useri conectati. 60001-60001 este un exemplu de retea faza 2 neextinsa. Un exemplu de distanta de cablu extinsa este 60001-60011. Astfel, 253 de useri sunt sustinuti pe fiecare distanta, deci teoretic, pot fi conectate 2530 de statii-capat la mijloacele de transmisie ale segementului.

Avanajtul retelelor extinse fata de cele neexstinse consta in numarul de useri pe care ii poate sustine pe un singur segmnet de cablu.

Sunt cateva incompatibilitati intre retelele faza 1 si faza 2 , de aceea, este de preferat a se folosi doar faza 2, daca este posibil. Marea problema de incompatibilitate este cu EtherTalk faza 1 si faza 2 (implementarea Ethernet a AppleTalk-ului) si cu folosirea impreuna retelelor extinse faza 1 si faza 2. EtherTalk faza 1 si faza 2 folosesc diferite formate de cadru si nu sunt compatibile. Putem sa rulam faza 1 si faza 2 pe acelasi cablu Ethernet, dar ele nu pot face schimb de date intre ele fara un ruter.

Retelele faza 1 si faza 2 extinse nu pot interopera deoarece faza 1 nu poate intelege distantele extinse a cablului.

In cazul unei folosiri simultane a fazei 1 si fazei 2, se vor folosi retele faza 2 neextinse.

AppleTalk opereaza peste toate protocoalele din nivelul 2, configuratiile punct-la-punct si apelare la comanda. [[7]](#footnote-7)

* 1. **Arhitectura si adresare**

Retelele AppleTalk sunt aranjate ierarhic. Cele 4 componente care formeaza baza unei retele AppleTalk sunt: socket-urile, nodurile, retelele si zonele.[[8]](#footnote-8)

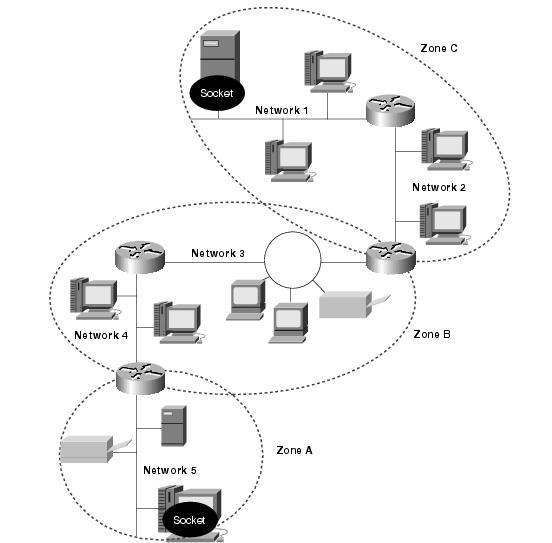
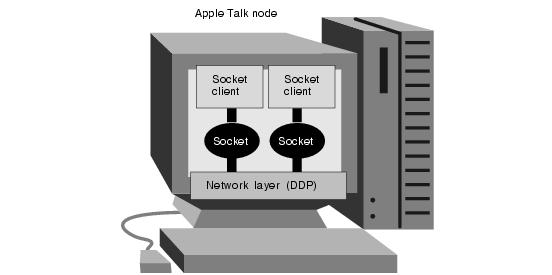


Figura ilustreaza organizarea ierarhica a acestor componente in inter-reteaua AppleTalk.[[9]](#footnote-9)

Un socket AppleTalk este o locatie unica si adresabila din cadrul unui nod. Este un punct logic in care stratul de sus al software-ului AppleTalk proceseaza si stratul de retea Datagram Delivery Protocol (DDP) interactioneaza. [[10]](#footnote-10)



Nodul AppleTalk[[11]](#footnote-11)

Un nod AppleTalk este un dispozitiv conectat la reteaua AppleTalk, si el poate fi un calculator Macintosh, o imprimanta, un IBM PC, un ruter, sau orice alt echipament similar. Fiecare nod apartine unei singure retele si unei zone specifice. [[12]](#footnote-12)

O retea AppleTalk este formata dintr-un singur cablu logic si multiple noduri atasate. Cablul logic cuprinde fie un singur cablu fizic, fie mai multe cabluri fizice interconectate folosind rutere.

O zona AppleTalk este un grup logic de noduri sau retele care este definita cand administatorul de retea configureaza reteaua. Nodurile sau retelele nu trebuiesc sa fie fizic alaturate pentru a apartine aceleasi zone. [[13]](#footnote-13)

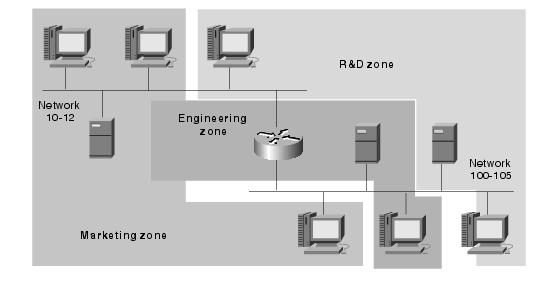


Figura ilustreaza o inter-retea AppleTalk compusa din 3 zone nealaturate.[[14]](#footnote-14)

Design-ul AppleTalk urmeaza foarte indeaproape modelul OSI al nivelelor de protocol[[15]](#footnote-15).

|  |  |
| --- | --- |
| OSI Model | Corresponding AppleTalk Layers |
| Application | Apple Filing Protocol (AFP) |
| Presentation | Apple Filing Protocol (AFP) |
| Session | Zone Information Protocol (ZIP) AppleTalk Session Protocol (ASP) AppleTalk Data Stream Protocol (ADSP) |
| Transport | AppleTalk Transaction Protocol (ATP)  AppleTalk Echo Protocol (AEP)  Name Binding Protocol (NBP)  Routing Table Maintenance Protocol (RTMP) |
| Network | Datagram Delivery Protocol (DDP) |
| Data link | EtherTalk Link Access Protocol (ELAP)  LocalTalk Link Access Protocol (LLAP)  TokenTalk Link Access Protocol (TLAP)  Fiber Distributed Data Interface (FDDI) |
| Physical | LocalTalk driver  Ethernet driver  Token Ring driver  FDDI driver |

Apple a proiectat AppleTalk ca fiind independent de conexiunea dintre nivele. Cu alte cuvinte, el poate rula peste orice implementare a oricarui nivel. AppleTalk suporta o mare varietate de implementari, inclusiv pentru Ethernet, Token Ring, Interfata de date distribuita prin fibra (FDDI), si LocalTalk. In cazul AppleTalk, Ethernet poarte denumirea de EtherTalk, Token Ring de TokenTalk, si FDDI de FDDITalk.

Protocoalele care sustin AppleTalk pe aceste nivele sunt: protocolul de acces la link EtherTalk (ELAP), protocolul de acces la link LocalTalk (LLAP), protocolul de access la link TokenTalk (TLAP), protocolul de acces la link FDDITalk (FLAP).

Factorii care fac diferenta in cazul AppleTalk sunt cele doua protocoale care tintesc sa faca sistemul complet auto-configurabil : AARP (protocolul de rezolutie al adresei AppleTalk) si NBP (protocolul de legatura a numelor).

O adresa AppleTalk este formata din 4 bytes astfel : 2 bytes pentru numarul retelei, 1 byte pentru numar nodului si 1 byte pentru numarul socket-ului. Dintre toate, doar numarul retelei necesita configurare, fiind obtinut de la un ruter. Fiecare nod isi alege in mod dinamic numarul conform unui protocol (la inceput se folosea LLAP, apoi s-a folosit AARP), care rezolva conflictul alegerii aceluiasi numar de catre noduri diferite. In cazul numerelor de socket, cateva numere bine stabilite au fost rezervate in mod special pentru insusi protocolul AppleTalk.

Din cauza dinamismului, useri nu pot accesa servicii specificand adresele lor. In schimb, toate serviciile au *nume*, care pot fi *cu inteles* pentu useri si pot fi suficient de lungi pentru a minimiza sansa de a se produce conflicte.

Serviciile pot fi mutate pe o masina diferita, si atata timp ct pastreaza acelasi nume de serviciu, nu trebie facut nimic in plus pentru ca un user sa acceseze in continuare acel serviciu. Mai mult, aceesi masina poate gazdui orice numar de instante de servicii de acelasi tip, fara a fi conflicte.

Daca useri sunt obisnuiti sa foloseasca un anumit nume de masina pentru a accesa un anumit serviciu, accesul va fi intrerupt cand serviciul va fi mutat pe o alta masina. Acesta intrerupere poate fi micsoarata prin folosirea inregistrarilor CNAME care indica serviciile mai degraba decat numele masinilor care se refera la servicii, dar nu avem garantia ca userii vor folosi aceasta conventie. [[16]](#footnote-16)

* 1. **Regulile de angajament al unui echipament la reteaua AppleTalk**

Retelele AppleTalk folosesc o schema de adresare, in care fiecare calculator, atunci cand este online, face urmatorii pasi:

* cauta o adresa stocata, folosita in sesiunile anterioare
* daca una nu este valabila, atunci alege la intamplare din cele care sunt valabile
* apoi trimite un mesaj catre toate celelalte echipamente din retea pentru a se asigura ca nici un alt calculator nu foloseste acea adresa
* in cazul in care adresa este folosita deja de alt echipament, alege alta adresa
* daca adresa nu este folosita, atunci o stocheaza adresa ca fiind potential buna de folosit pentru urmatoarea sesiune cand va fi online [[17]](#footnote-17)
  1. **Invechirea si abandonul AppleTalk-ului**

Odata cu cumpararea ulterioarelor dezvoltari ale Mac OS X de catre NeXT, AppleTalk a devenit cu siguranta un sistem invechit. A fost adaugat sprijin pentru OS X cu scopul de a sustine un numar foarte mare de echipamente AppleTalk, cu precadere imprimante laser si servere de fisiere, dar solutiile alternative de conectare din perioada aceea, de exemplu USB pentru imprimante, au limitat cererile de dezvoltare. Astfel, odata cu abandonul multor din produsele din aceasta categorie, si fiind noile sistemele se bazau pe IP, AppleTalk a devenit din ce in ce mai putin uzual.

Cu toate acestea, pierderea aceasta nu a redus dorinta pentru solutiile de retele care combina modul usor de folosit cu rutarea prin IP. Apple a depus mult efort in astfel de implementari, de la introducerea ruterelor AirPort pana la dezvoltarea sistemului de configuratie de retea Zero si implementarea lui. [[18]](#footnote-18)



Ruter AirPort Apple[[19]](#footnote-19)

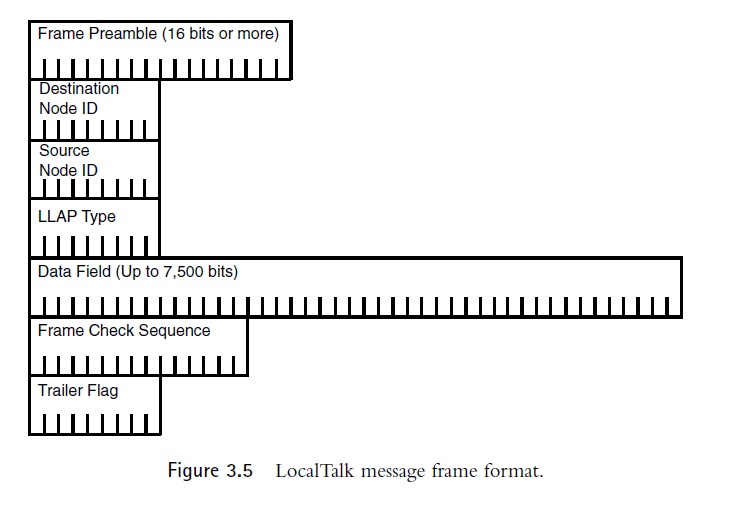
1. **Protocoalele AppleTalk si rolul acestora (Dumitran Tudor)**
   1. **Nivelul fizic si nivelul legatura de date**

Prima problema care se pune in dezvoltarea unei retele este schema de cablaj. In ceea ce priveste Apple, aceasta schema a fost numita  **LocalTalk.** Initial, LocalTalk folosea cabluri torsadate ecranate si o topologie bus. Mai tarziu, topologia a devenit stea, denumita PhoneNet. LocalTalk opera la 230Kbps folosind CSMA/CA (acces multiplu cu ascultarea mediului si evitarea coliziunii) ca metoda de control de acces. Cu CSMA/CA, reteaua trebuia sa aiba o pauza inter-dialog de 400 microsecunde, pe langa o asteptare aleatoare, bazata pe nivelul traficului si coliziuni, inainte ca un nod sa inceapa sa transmita date. Transmisiile foloseau tehnica handshake pentru stabilirea conexiunilor intre noduri.

In ceea ce priveste adresarea acestor noduri si a retelei, s-a folosit  **protocolul de acces LocalTalk Link (LLAP)**. Acesta oferea transport pe nivelul legatura de date, de tip “best effort” si foloseste numerele nodurilor si ale retelei pentru adresare. Adresarea era dinamica.[[20]](#footnote-20)

La aparitia Ethernet-ului, s-a cautat sa se creeze o legatura intre sistemele Apple si aceasta retea, deoarece oferea performante mult mai bune. Astfel, reteaua Apple a putut fi legata la reteaua Ethernet prin **LocalTalk-to-Ethernet Bridge.** Unele dintre aceste „poduri” puteau fi folosite si pt utilizarea de catre Apple a anumitor protocoale. Un astfel de exemplu este sistemul MacIS care permitea Mac-urilor LocalTalk sa schimbe pachete TCP/IP prin intermediul acestor poduri.

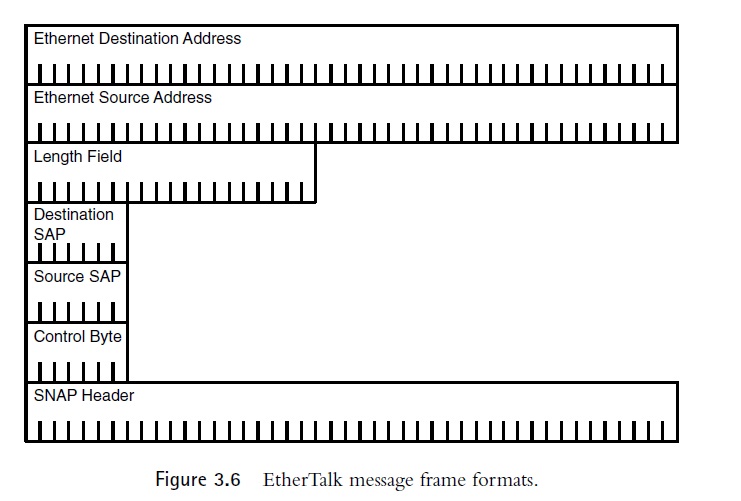
In cele din urma, Apple a realizat suport pentru retelele Ethernet, si LocalTalk a disparut.[[21]](#footnote-21)



(Sura 1 Bibliografie)

Deoarece Ethernet oferea largime de banda mai mare si viteza sporita de transfer, a fost necesara adaptarea protocolului AppleTalk pentru aceasta retea.Astfel a aparut **EtherTalk, ce** folosea tehnica de acces multiplu cu ascultarea mediului si detectarea coliziunii (CSMA/CD) pe 10Mbps si protocolul de acces **EtherTalk Link (ELAP)** pe reteaua Ethernet 100Mbps.Functionalitatile si functiile nivelelor AppleTalk erau in continuare folosite, pe reteaua Ethernet. ELAP gestiona toate interactiunile protocoalelor ULP(Upper Layer) cu mediul de transmisie. Versiunea de AppleTalk folosita (Phase 1 sau Phase 2) determina cum este format cadrul EtherTalk. Versiunea 1 folosea specifiicatiile Ethernet-II, in timp ce versiunea 2 folosea specificatiile IEEE 802.3 SNAP (Subnetwork Access Protocol).

ELAP transmitea date, obtinand adresa destinatie din diagram DDP, folosind un tabel AARP de cautare pentru maparea adresei, apoi construind cadrul Ethernet potrivit pentru retea (in functie de versiunea AppleTalk).[[22]](#footnote-22)

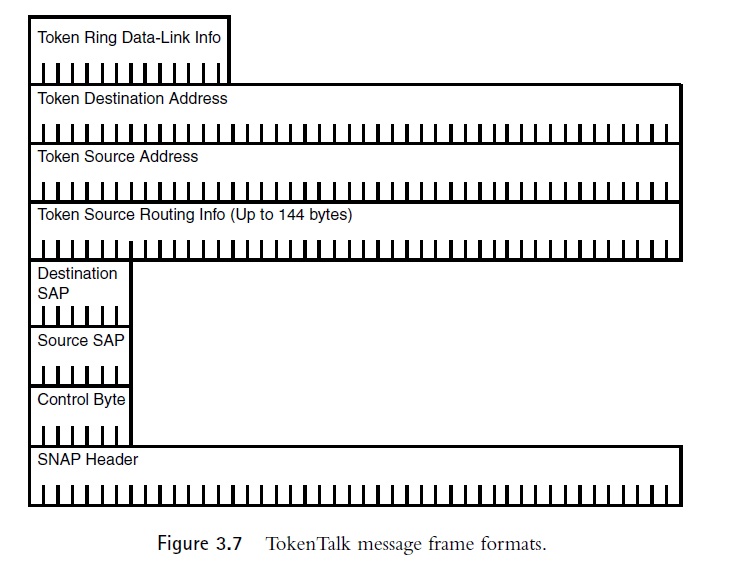


(Sura 1 Bibliografie)

Toate protocoalele nivelului legatura de date foloseau standardul IEEE 802.2 pentru MAC. Acest standard oferea posibilitatea ca diferite protocoale de retea ce ruleaza pe acelasi computer sa poata discerene ce cadre apartin lor. Acest lucru era posibil folosind SAPs (service acces points) pentru a identifica carui protocol ii este destinat un pachet. Header-ul 802.2 continea o valoare destinatie si o sursa SAP. Valoarea folosita pentru a indica un protocol cu standard non-IEEE era $AA. AppleTalk folosea aceasta valoare. Impreuna cu SAP, mai exista un protocol SNAP de 5 octeti discriminator. Acesta era folosit pentru a identifica familia de protocoale de care apartine pachetul respectiv. Doua valori SNAP se foloseau pentru a defini un pachet AppleTalk:

* $080007809B arata ca un cadru contine un pachet de date AppleTalk
* $00000080F3 arata ca un cadru este un pachet AARP

La aparitia retelelor cu raspuns asigurat, in special token ring, care folosea cablu coaxial la inceput si fibra optica mai tarziu, viteza de transfer obtinuta (in special pe fibra optica) era suficient de mare pentru a determina realizarea unei adaptari a AppleTalk la aceste retele. Solutia a fost **TokenTalk,** care oferea control de acces media fara coliziune pe 4 si 16 Mbps IEEE 802.5 Token Ring. TokenTalk era suportat doar in retele AppleTalk Phase 2. Ca si EtherTalk, folosea un protocol de control al accesului, numit **TokenTalk Link Access Protocol (TLAP)** care gestiona interactiunile UPL cu mediul de transport si folosea AARP pentru translatarea adreselor. TLAP extragea adresa destinatie din diagramele DDP, apoi verifica aceasta adresa cu AARP, pentru a obtine adresa de destinatie hardware. Dupa confirmarea acestei adrese, TLAP construia cadrul cu ajutorul datagramei DDP, plus header-ul protocolului 802.5 Token Ring si header-ul SNAP.[[23]](#footnote-23)



(Sura 1 Bibliografie)

Mai tarziu, aparitia retelelor cu topologie inel, bazate pe fibra optica, care nu foloseau protocolul IEEE 802.5 Token Ring**,** a fost motivul pentru care Apple a oferit suport si pentru FFDI(Fiber Distributed Data Interfaces). Interfata folosita era ANSI X3T9.5, simpla sau duala. **Protocolul FDDITalk Link Access (FLAP)** oferea control de acces pentru 100Mbps pe aceste interfete. FLAP folosea, de asemenea, AARP pentru obtinerea adresei hardware. Cadrele era obtinute prin atasarea datagramei DDP la headerele FFDI si SNAP.

Existenta mai multor protocoale de retea intr-o retea multipunct a fost posibila prin folosirea LLP, ce oferea mecanisme de multiplexare.De asemenea, oferea mecanisme de control al fluxului de date si mecanisme de gestinoare a erorilor (ARQ- automatic repeat request).

SNAP era un protocol pe 5 octeti care determina familia de protocoale din care face parte un pachet;

Doua valori SNAP se foloseau pentru a defini un pachet AppleTalk:

* + $080007809B arata ca un cadru contine un pachet de date AppleTalk
  + $00000080F3 arata ca un cadru este un pachet AARP[[24]](#footnote-24)
  1. **Nivelul retea**

Toti clientii si toate serverele AppleTalk necesita o adresa unica pentru a participa la retea. Adresa de retea este de 24 biti (16 biti adresa de retea+8 biti adresa nodului). Nodurile nu trebuie sa aiba o adresa preconfigurata, acestea obtinand o adresa la prima accesare a retelei. Adresele nodurilor si retelei sunt asignate dinamic.

Atunci cand un nod acceseaza o retea, obtine o adresa provizorie. Aceasta adresa este selectata aleator din gama 1-254(0 si 255 sunt rezervate).Adresa retelei este selectata de protocoalele de nivel legatura de date(ELAP, LLAP, TTAP, or FDAP) aleator dintr-o clasa rezervata de adrese de la 65,280 (FFF0 hex) pana la 65,534 (FFFE hex). Acestea sunt tot adrese provizorii. Sunt necesare adrese permanente.

Nodul trimite apoi o cerere GetNetInfo. Daca un ruter raspunde, numarul de adresa al acestuia este returna. Nodul foloseste acest numar, genereaza un ID de nod aleator si trimite acest ID in toata reteaua. Daca adresa nu este folosita de niciun alt nod, nodul current o foloseste. Altfel, se repeta procesul de generare si validare pana cand se gaseste o adresa nefolosita.

Daca niciun ruter nu este valabil sau reteaua este de tip Phase 1, adresa de retea este setata 0 si nodul face broadcast pentru a determina daca adresa lui e in conflict cu a altui nod.Daca nu, aceasta devine adresa permanenta a nodului respectiv. Adresa este pastrata, in cazul in care nodul paraseste reteaua si apoi o acceseaza dupa un timp, pentru ca nodul respectiv sa o poate folosi in continuare(daca nu exista conflicte).[[25]](#footnote-25)

Adresele MAC de la nivelul legatura de date erau abstracte si nu putea fi inteles de nivelul retea. Aceasta problema a fost rezolvata implementand protocolul AppleTalk **Address Resolution Protocol (AARP)**. AARP avea doua roluri, primul fiind de a construi un tabel de mapare a adreselor(AMT) care continea adresele hardware (de nivel legatura de date) translatate. Fiecare nod construia un astfel de tabel. Fiecare nod trebuia sa interogheze AARP pentru a accesa tabelul, la un anumit interval de timp. Accesarea tabelului se facea pentru rezolvarea adreselor hardware.

Era posibila actualizarea AMT prin citirea adreselor din pachetele de date ce soseau. Acest process era numit ”spicuirea adreselor”. Era o procedura ce reducea interogarile AARP, dar adauga cost suplimentar procesarii pachetelor.

Existau 3 tipuri de mesaje AARP: Cerere, Raspuns si Proba

Mesajele Cerere si Proba erau trimise prin broadcast ca adrese de nivel hardware si erau procesate de fiecare nod. Mesajele Response erau trimise ca mesaje unicast, catre expeditorul cererii sau probei.

Al doilea rol al AARP era de a asista la procesul de achizitie a adreselor. Atunci cand se facea o cerere a unei adrese de nivel legatura de date, se cauta aceasta adresa in AMT. Daca nu exista, host-ul folosea AARP pentru a o obtine. In acest timp, o adresa provizorie era furnizata host-ului.

Protocolul AARP era echivalentul protocolului ARP din TCP/IP.

Adresarea si transportul de date al protocoalelor de nivel superior intre client si server se realiza prin datagrame. Transportul era de tip „best effort” , dar nu avea suport pentru gestionarea erorilor, pe care le lasa in seama nivelului legatura de date. Datagrama avea maxim 599 octeti (13 octeti header si restul date), incluzand si bitii de control, folositi la verificarea corectitudinii transmisiei. Protocolul care se ocupa de acest aspect s-a numit **Datagram Delivery Protocol (DDP).**

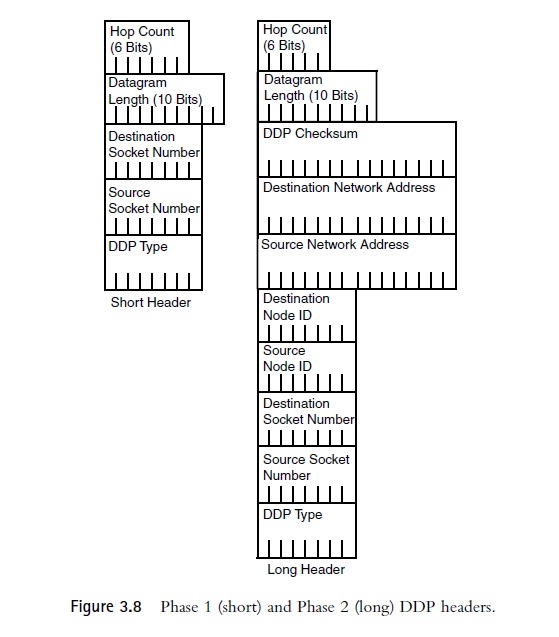
Toate functiile retelelor AppleTalk si serviciile NVE erau furnizate de interfata socket. Socket-urile ascultau pe o anumita adresa pentru a verifica o cerere de servicii si aveau rolul de a trimite datele catre destinatie.

Adresele de socket-uri erau numere pe 8 biti si reflectau daca sarcinile sunt de tip static sau dinamic. Socket-urile asignate sarcinilor statice (SAS) apartineau intervalului 1-127. Numerele intre 1-63 erau foloste pentru servicii de mentenanta :SAS 1 (RTMP), SAS 2 (Names Information Socket), SAS 4 (Apple Echo Protocol), si SAS 6 (ZIP). Numerele intre 64 to 127 erau rezervate pentru folosire experimentala. Socket-urile asignate sarcinilor dinamice foloseau numere de port de la 128 la 254. Acestea erau atribuite aleator.

Cu aceste mentiuni, putem defini DDP ca metoda folosita pentru transportul datagramelor intre socket-uri end-node, locale sau la distanta(client/server). Pentru a furniza acest serviciu, DDP aveau 2 formate diferite de datagrame: DDP Datagram Headers si DDP Broadcasts.

DDP Datagram Headers erau destinate unei singure destinatii si foloseau adrese sursa si destinatie de 32 biti. Adresele de retea erau de 16 biti , cele de nod de 8 biti si socket-ul folosea 8 biti. In cazul retelelor de tip Phase 1, adresa de retea era 0, deci doar adresele de nod si socket erau relevante pentru transmisie.

In functie de tipul retelei, se deosebeau doua tipuri de header-e. Pentru cazul Phase 1, se foloseau short header, deoarece erau necesare doar adresele nodului si socket-ului. Pentru cazul Phase 2, se foloseau long header, aici intervenind si adresa de retea, care nu mai era 0.[[26]](#footnote-26)



(Sura 1 Bibliografie)

DDP Broadcasts erau folosite pentru a trimite datagrame la toate nodruile din retea si erau de 3 tipuri: broadcast pentru o retea specifica, broadcast pentru orice retea si broadcast pentru o zona specifica.

Modul in care broadcast-ul era interpretat de nod era determinat de campul adresei nodului in adresa retelei destinatie din pachetul de date. Daca numarul de retea era diferit de 0, o datagrama cu ID-ul nodului de destinatie $FF (255) era examinata de toate nodurile si era acceptata de nodurile care se potriveau ca numar de retea. Daca numarul de retea era 0, broadcast-ul este destinat segmentului local de retea. Daca pachetul este pentru toate retelele, adresa de retea era 0 si adresa nodului era $FF, deci toate nodurile acceptau pachetul. Daca pachetul era destinat unei zone specifice, avea acceasi adresa ca cel destinat tuturor retelelor. Fiecare nod accepta pachetul, apoi era verificata informatia despre zona si tot ce nu era destinat nodului era eliminat.

Retelele AppleTalk sunt limitate la 15 hop-uri. Header-ele DDP extinse au un contor de hop-uri, care este incrementat cu 1 de fiecare data cand o datagrama trece printr-un ruter. Cand contorul ajunge la 15, pachetul este eliminat. [[27]](#footnote-27)

* 1. **Nivelul transport**

Protocoalele nivelului transport contribuie la furnizarea urmatoarelor servicii de transport a datelor ULP(Upper Layer Protocols) intre noduri terminale:

* Crearea si gestionarea tabelei de rutare
* Servicii de transport intre AppleTalk si TCP/IP
* Evaluarea nodurilor terminale
* Adresare binara a serviciilor de translatie a entitatilor de retea
* Transport de date orientat pe socket-uri

Rutarea AppleTalk era un proces dinamc. Desi adresele de retea erau alocate static, adresele de noduri erau alocate dinamic.Nodurile terminale determinau daca o datagrama va fi livrata local sau la distanta. Daca destinatia era la distanta, ruterul prelua procesul. Scopul lui era de a mentine informatii despre diverse segmente de retea la care se poate ajunge. Aceasta informatie includea :

* Lungimea retelei(cablului)
* Distanta pana la retea, in hop-uri
* Interfata ruterului folosit pentru a ajunge la reteaua destinatie
* Adresa retelei in care se afla urmatorul hop[[28]](#footnote-28)

Existau 3 tipuri de rutere folosite in retele AppleTalk:

* Local routers folosite pentru a conecta segmente locale adiacente de retea.Un ruter local era folosit pentru a segmenta o retea fizica mare in diverse segmente de reteal.
* Half routers folosite pentru conexiuni p2p WAN. O jumatate era conectata la un segment local AppleTalk, si cealalta jumatate la link-ul WAN. Natura transportului folosit putea fi un modem, o retea publica , etc.
* Backbone routers erau folosite pentru a transporta traficul dintr-o retea AppleTalk intr-o alta retea oarecare.

In cazul ruterelor locale si “jumatati” ,protocoalele AppleTalk erau folosite prin partile interconectate. Ruterele backbone foloseau AppleTalk in conjunctie cu un alt protocol (de obice TCP/IP). Informatia din tabelele de rutare era independent de tipul ruterului.

Crearea tabelelor de rutare incepea cu un singur ruter, numit seed router. Scopul lui era de a furniza celorlalte rutere informatii legate de adresele de retea. Acest ruter avea gama de adrese de retea setata static la porturi.

Atunci cand un ruter pornea, creea o tabela cu toate segmentele de retea conectate la el. Fiecare retea definita era introdusa ca o retea locala cu o distanta de hop-uri egala cu 0. Un ruter de tip seed construia o tabela ce continea o gama de retele, asociata cu interfata fiecarui ruter. Un ruter non-seed construia o tabela cu toate interfetele ce foloseau o adresa de retea 0. Apoi, ruterul trimitea o actualizare ce continea toate retele la care putea ajunge prin interfetele sale.[[29]](#footnote-29)

Cum rutarea AppleTalk era un proces dinamic, era nevoie de un protocol care sa automatizeze acest proces de actualizare a tabelelor de rutare din nodurile retelei. Acest protocol a fost numit **Routing Table Maintenance Protocol (RTMP),** asemanator cu RIP din TCP/IP, care folosea o metrica de rutare pentru gestionarea rutelor. Metrica aleasa a fost hop count (nr hop-urilor), calculata in baza numarului de rutere ce trebuiau traversate pentru a se ajunge la destinatie. Daca o retea era direct conectata la alta retea, numarul hop-urilor era 0. RTMP folosea tehnica despicarii orizontului(slipt horizon) pentru a preveni buclele de rutare.

Scopul RTMP era de a avea un tabel de rutare cu cea mai buna ruta pentru fiecare retea data. Treaba ruterului seed era de a trimite numere de retea tuturor rutere-lor care intra in retea.

Split horizon impunea o restrictie importanta : informatia de rutare invatata de la o interfata nu poate fi trimisa inapoi ca actualizare. Doar actualizari din interfetele ruter-ului puteau fi trimise inapoi la alte rutere.

Tipurile de mesaje RTMP erau : Cerere, Data, Raspuns, Cerere a rutei unei date. Aceste mesaje erau transportate in datagrame DDP, si formatul varia in functie de tipul retelei (extinsa sau nu). Mesajele de date erau folosite de rutere pentru a schimba informatie de rutare. Actualizarile RTMP aveau loc la fiecare 10 secunde. Toate intrarile rutelor aveau un timp de validare asociat. Daca o ruta nu era verificata dupa 10 secunde, era marcata ca suspecta. Dupa 20 de secunde, era marcata ca defecta si eliminata. Mesajele Cerere, Raspuns, si ruta datelor erau folosite pentru comunicatia intre noduri si rutere, pentru atribuirea de adresele nodurilor (Cerere si Raspuns) si pentru achizitionarea de date din tabela de rutare a unui anumit ruter (Ruta datelor).[[30]](#footnote-30)

Aparitia protocolului TCP/IP a dus la dezvoltarea unei metode de a trimite pachetele Apple folosind tunelul IP. Procesul de incapsulare a pachetelor AppleTalk in pachete IP se numeste AppleTalk tunneling si folosea ruter-e AURP (Update-Based Routing Protocol).

Ruter-ul AURP incapsula pachetele AppleTalk in UDP si le trimitea urmatorului ruter AURP, prin portul UDP 387. Ruter-ul AURP care primeste, elimina header-ele UPD si IP din pachet si trimite pachetul la reteaua destinatie.[[31]](#footnote-31)

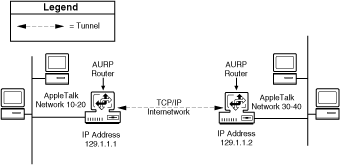
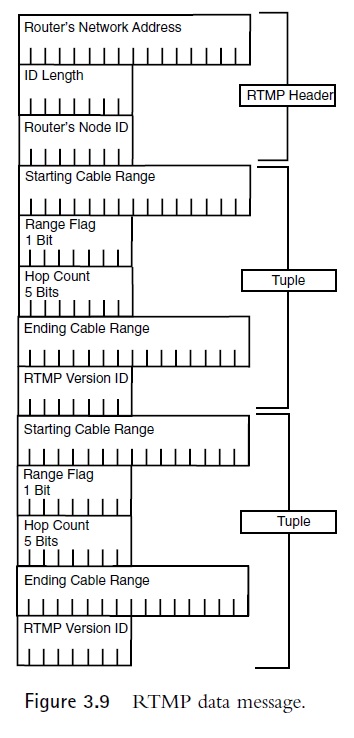
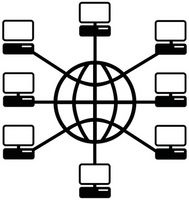


Fig. AppleTalk tunneling (Sursa 2 Bibliografie)



(Sura 1 Bibliografie)

**(Benciu Florin)**

Problema legată de specificarea furnizorului de servicii de internet a fost rezolvată folosind NBP-ul. Protocol Nume de legătură - este legătura dintre un protocol și altul dintr-o rețea pentru a crea un nou flux de date . În rețelele de comunicații , fluxurile de date între protocoale , cum ar fi protocolul de transport și protocolul de rețea , seamană cu modul de transmisie de la o sursă la o destinație. Obligatoriu, protocoalele creează canalul folosit de date ca să se misca de la cererea inițială prin intermediul rețelei .

NBP permite utilizatorilor să specifice serviciului utilizat printr-un port WAN ( Wide Area Network ) . În cazul în care utilizatorul are mai multe ISP-uri , sau furnizori de servicii Internet , disponibili la un VPN sau Virtual Private Network , utilizatorul poate specifica care este furnizorul de servicii de Internet pentru a utiliza portul WAN .

NBP a fost un sistem dinamic distribuit, de gestionare a numelor AppleTalk . Atunci când un serviciu începea pe o mașină , era înregistrat un nume pentru sine, ca fiind ales de către un administrator uman . În acel moment, NBP prevedea un sistem de control asftel încât nici un alt aparat nu putea fi înregistrat deja cu același nume. Mai târziu , atunci când un client dorea să acceseze acest serviciu , folosea NBP pentru a interoga mașini cu scopul de a găsi acest serviciu. NBP-urile furniza browseability ( " care sunt numele tuturor serviciilor disponibile ? " ), precum și capacitatea de a găsi un serviciu cu un anumit nume.

Fiecare terminal de comunicare AppleTalk ( de exemplu, fiecare " priză " ), are un nume de forma " Entitate : Tipul @ Zone " . Un exemplu de nume de NBP ar putea fi :

Stuart Printer : LaserWriter @ Casa de Stuart

Tipul de serviciu " LaserWriter " indica faptul că această comunicare a socket-ului numit trebuia să utilizeze AppleTalk Printer Access Protocol ( PAP ) . Acesta putea sau nu putea fi un veritabil brand LaserWriter Apple. De fapt, aceasta nu putea fi o imprimantă deloc . Entitatea putea fi un software de calculator care rula spoolerul de imprimare , sau putea fi un alt tip de dispozitiv , ca un aparat fax de rețea. Modul în care aceste persoane puteau găsi aceste servicii este scopul NBP .

În practică , pentru a reduce sarcinile inutile pe alte computere din rețea , NBP nu difuzează într-adevăr cererile , ci se foloseste în schimb multicast Ethernet. NBP utilizează o funcție hash pentru a selecta o adresă multicast Ethernet , și apoi trimite cererea multicast numai la acea adresă . Mașini cu care se furnizează servicii de pe rețeaua utiliza aceeași funcție hash , și așteaptă de la adresele corespunzătoare de multicast Ethernet pentru a primi pachetele de căutare NBP.

Există câteva detalii minore , cum ar fi modul în care gateway-uri transmite pachete și răspunsuri cache , dar asta e , practic, tot ce trebuie să știm pentru a înțelege ideea fundamentală . Este foarte simplu , foarte ușor de implementat , și foarte util .

Combinația dintre toate tabelele nume din toate nodurile Internetwork este cunoscut sub numele de directorul de nume , care este o bază de date distribuită de toate mapările nume - la - adresa . Numele obligatoriu poate aparea atunci cand un nod este primul pornit sau dinamic, imediat înainte de entitatea numită, dacă este accesat .

NBP îndeplinește următoarele patru funcții : nume de căutare , de recunoaștere nume , nume de confirmare , și nume de ștergere . Nume de ștergere este utilizat pentru a elimina o intrare din tabelul de nume, ca atunci când nodul este oprit .

Există două motive principale pentru folosirea numelor de entități , mai degrabă decât adrese straturilor superioare . În primul rând , adresele de rețea sunt atribuite nodurilor dinamic și , prin urmare , schimbate periodic . Numele de entitate oferă un mod consistent pentru ca utilizatorii să se refere la resurse și servicii de rețea , cum ar fi un server de fișiere . În al doilea rând , folosind nume în loc de adrese pentru a se referi la resurse și servicii păstrează transparența operațiunilor din stratul inferior pentru utilizatorii finali .[[32]](#footnote-32)

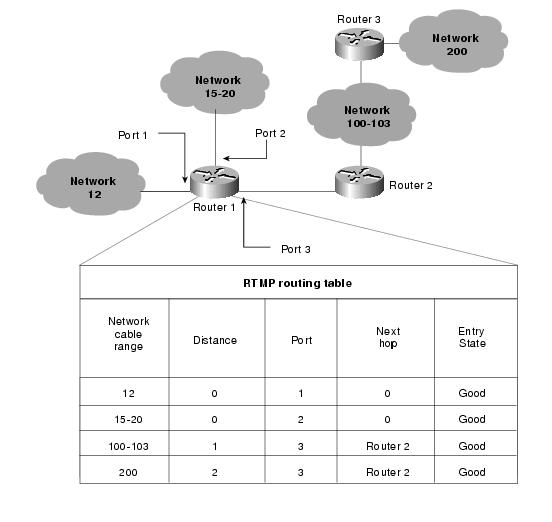
[](http://docwiki.cisco.com/wiki/File:CT843508.jpg)[[33]](#footnote-33)

Figura :Tabelul de rutare conține informații despre fiecare rețea destinație

Problema de testarea accesibilității la nodurile de rețea s-a rezolvat cu integrarea protocolului AEP, oferind un serviciu de ecou la gazdele AppleTalk .

Protocolul Echo ( AEP ) este un protocol de strat de transport care generează pachete care testează accesibilitate la nodurile de rețea . AEP poate fi implementat în orice nod AppleTalk și are ca priză de acces numărul atribuit static 4 ( priza Echoer ) .[[34]](#footnote-34)

Pentru a testa accesibilitate unui nod dat , un pachet de solicitare AEP este trecut de la DDP la sursă . DDP adresează pachetul corespunzător , indicând în câmpul Type că pachetul este o cerere de AEP . Atunci când pachetul este primit de destinație , DDP examinează câmpul Tip și vede că este o cerere de AEP . In acest proces, pachetul este copiat, schimbat la un răspuns de tip AEP ( prin schimbarea unui câmp din pachetul AEP ) , și e returnat la nodul sursă .[[35]](#footnote-35)

Soluția, în legătură cu problema întreruperii de conexiunii atunci când se dorea utilizarea protocoalelor de nivel mai înalt, s-a găsit prin a crea un protocol de control al fluxului de date.

Nivelul Tranzacțional folosește un token bitmap care se ocupe de confirmare și controlul fluxului și o secvență de octeți rezervată pentru utilizarea de protocoale de nivel superior . Cu alte cuvinte, Nivelul Tranzacțional apela doar la un simplu schimb de cerere / răspuns , fără a fi nevoie de a crea sau de a opri conexiuni .

Un pachet cerere ATP putea să răspundă cu până la opt pachete de răspuns . Solicitantul apoi trimitea un pachet de confirmare care conținea o mască de biți, indicând care dintre pachetele de răspuns le-a primit, astfel încât respondentul putea retransmite restul .

ATP -ar putea opera în oricare dintre modurile " at – least - once " sau " cel - puțin - o dată ". Modul exact - o dată a fost esențial pentru operațiunile care nu au fost idempotente , în acest mod , respondentul păstrat o copie a tampoanelor de răspuns în memorie până la primirea cu succes a unui pachet comunicat de solicitant , sau până când un timeout a trecut . În acest fel , s-ar putea răspunde la duplicat, prin retrimiterea aceleași date de răspuns , fără a efectua din nou funcționarea reală .

ASP a fost un protocol intermediar , construit mai sus de ATP , care la rândul său a fost temelia AFP-ului . Aceasta a oferit servicii de bază pentru a solicita răspunsuri la comenzi arbitrare și efectuarea de interogări de stare de out-of -band. De asemenea, ea a permis server-uli pentru a trimite mesaje asincrone în atenția clientului . ASP emite un identificator unic pentru fiecare sesiune legătură logică și continuu monitorizează starea fiecărei conexiuni . Acesta susține sesiuni de mers în gol prin schimbul periodic ține de cadre în viață , în scopul de a verifica starea de sesiune .[[36]](#footnote-36)

**3.4 Nivelul sesiune**

În comunicarea imprimantelor cu serverul s-a ivit problema legată de faptul că nu utilizau canalul de transmisie corect, provocând dereglaje de debit, iar acest lucru îngreuna utilizarea benzii de transfer de date când venea vorba de solicitarea canalului de către un număr major de utilizatori; soluționarea situației a stat în construirea unui protocol care oferea informații legate de bandă ATP-ului.

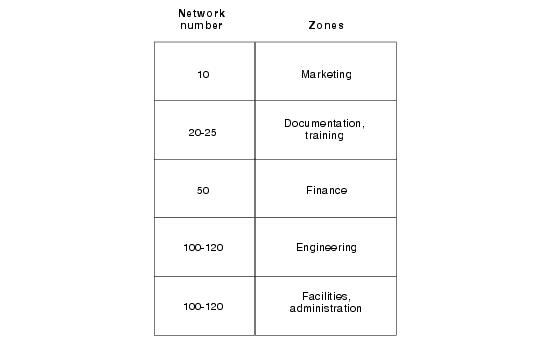
Nivelul AP a fost construit pe nivelul de sus al ATP-ului. [[37]](#footnote-37) Atunci când o conexiune la imprimantă se deschidea , fiecare capăt trimitea la celălalt o cerere de ATP. Răspunsul clientului la server a fost de a trimite un bloc de cod PostScript , în timp ce serverul putea răspunde cu orice mesaje de diagnostic care ar putea fi generate ca rezultat , după care o altă cerere de " *send - more - data* ", era trimisă. Această utilizare a ATP a oferit un control automat de debit. Ca rezultat fiecare capăt putea trimite numai date la celălalt capăt în cazul în care existat o cerere remarcată de ATP, pentru a răspunde .

PAP prevedea , de asemenea, interogări de stare out-of -band, manipulate de tranzacții ATP separate. Chiar în timp ce o imprimanta printa unui client , un server de PAP putea continua pentru a răspunde la solicitările de stare de la alți clienți . Acest lucru a permis alte Macintosh-uri de pe LAN , care au fost puse pe lista de așteptare, pentru a imprima, afișarea mesajele de stare care indica faptul că imprimanta era ocupată.

Printer Access Protocol ( PAP ) gestionează conexiunile virtuale la imprimante și alte servere . PAP este folosit și pentru a coordona transferul de date .[[38]](#footnote-38)

Protocolul Zona de Informații ( ZIP ) este un protocol de nivel sesiune cu care sa căutat o modalitate de a rezolva denumirea zonele de identificare în cadrul rețelei cu denumiri specifice. ZIP este utilizat în principal de către routere AppleTalk. Alte noduri de rețea , cu toate acestea , folosesc servicii poștale la pornire pentru a-și alege zona lor. *Protocolul Zona de Informații* menține un tabel de informații-zonă ( ZIT ), în fiecare router. [[39]](#footnote-39)ZIT-urile sunt menținute, ca liste, de ZIP care mapează numere de rețea specifice pentru una sau mai multe nume de zonă.

O zonă era o subdiviziune a rețelei care a făcut sens pentru oameni ( de exemplu , " Contabilitate " ) , dar în timp ce un număr de rețea a trebuit să fie atribuit o secțiune topologic - contiguă a rețelei , o zonă ar putea include mai multe porțiuni discontinue diferite ale rețelei.[[40]](#footnote-40)

[](http://docwiki.cisco.com/wiki/File:CT843511.jpg)[[41]](#footnote-41)Figura : Asistență ZIT în Zona de identificare ilustrează un ZIT .

Tipul de comunicare la nivel de prize AppleTalk terbuia să implice menținerea unor comunicații full-duplex. Acest protocol a fost adăugat mai târziu în stiva de protocoale.

ADSP garantează că datele sunt ordonate în mod corect și că pachetele nu sunt dublate. ADSP implementează de asemenea un mecanism de flux - control care permite o destinație pentru a încetini transmisii sursă prin reducerea dimensiunii ferestrei publicitate primi. ADSP ruleaza direct deasupra DDP-ului .

Diferențe semnificative de la TCP sunt o încercarea de conectare care putea fi respinsă, inexistența conexiuni de " semi- deschis "; o dată un capăt al conexiunii iniția o întrerupere , întreaga conexiunea se închidea (ADSP este full - duplex , nu dublu simplex )

Oferă un canal de date pentru gazde. Acesta este un protocol orientat pe conexiune , care garantează livrarea de date în functie de controlul de debit.[[42]](#footnote-42)

3.5. **Nivelul prezentare si nivelul aplicatie**

Legat de siguranța datelor utilizatorilor de pe rețea s-a inițiat un protocol, numit Protocolul Filling si servește al comunicarea cu serverele de fișiere APPLESHARE. Oferă servicii pentru autentificarea utilizatorilor ( extensibile la diferite metode de autentificare , inclusiv două sensuri de schimb de numere aleatoare ) și pentru efectuarea operațiuni specifice pentru sistemul de fișiere MacintoshHFS . AFP este încă în uz în Mac OS X , chiar dacă cele mai multe alte protocoale AppleTalk au fost depreciate.

APPLESHARE a fost un produs de la Apple Computer , care a implementat diverse servicii de rețea. Scopul său principal a fost să acționeze ca un server de fișiere , folosind protocolul AFP . Mai pe scurt, AFP este protocolul de file sharing al arhitecturii AppleTalk.[[43]](#footnote-43)

1. **Solutii de portabilitate(Benciu Florin)**

Când AppleTalk fost lansat, în primul rând , platforma dominantă a fost PC-ul compatibil să ruleaze MS - DOS . Apple a introdus PC Card AppleTalk la începutul anului 1987, care permitea PC-uri să se alăture rețelelor *AppleTalk* și imprima pe imprimante LaserWriter.

Un an mai târziu, PC-ul APPLESHARE a fost lansat, și permitea PC-uri accesul la serverele de fișiere APPLESHARE. **AppleTalk - PC**, a permis comunicarea între PC-uri și Mac-uri cu software-ul TOPS instalat ( Mac-urile fără TOPS instalate puteau folosi aceeași rețea, dar numai pentru a comunica cu alte mașinile Apple). Software-ul Mac TOPS nu se potrivea cu calitatea Apple, dar software-ul DOS a fost relativ simplu de utilizat din punct de vedere DOS , și a fost robust .

Sistemele de operare BSD și Linux suportă *AppleTalk* printr- un proiect open source denumit **Netatalk** , care pune în aplicare suita completă de protocol și le permite să acționeze atât ca fișier nativ sau servere de imprimare pentru calculatoare Macintosh , cât și ca imprimante în rețeaua LocalTalk. Sistemele de operare Windows pentru servere suportă *AppleTalk* începând cu Windows NT până la Windows Server 2003.

Miramar a inclus *AppleTalk* în produsul MacLAN PC-ului , care a fost întrerupt de către CA în 2007 . GroupLogic continuă să conecteze protocolul *AppleTalk* cu software-ul de server ExtremeZ - IP pentru integrare Macintosh - Windows, care suportă Windows 2008 Server și Windows Vista, versiuni de asemenea anterioare. HELIOS Software GmbH oferă o implementare de proprietate a stivei de protocoale AppleTalk , ca de server HELIOS UB2. [[44]](#footnote-44)Aceasta este , în esență, o suită de fișiere și Print Server care ruleaza pe o gamă largă de platforme diferite. În plus , la Universitatea Columbia a lansat AppleTalk Pachetul Columbia ( PAC ), care a implementat suita de protocoale pentru diverse siteme de operare Unix , inclusiv Ultrix, SunOS, BSD și IRIX.[[45]](#footnote-45)

Un obstacol major legat de imprimarea cross-platform este descrierea proprietăților paginii utilizate de computere și imprimante pentru a comunica. Macintosh și multe sisteme Windows folosesc descrierea proprietăților paginii **PostScript** . Mașinile sub Macintosh și Windows pot imprima atât pentru imprimante **PostScript**, utilizând același protocol de rețea cât și cazul în care sunt ambele conectate la un server de imprimantă.

Cu toate acestea , multe imprimante de PC-uri folosesc Printer-Command Language Hewlett Packard (HP PCL). Mac-urile nu se pot folosi în mod obișnuit pentru HP PCL , dar există produse pe care le puteți utiliza, care permit Mac-urilor să comunice cu imprimantele PCL.

Ca și în cazul serviciilor de fișiere , imprimarea în rețea poate fi peer-to -peer cu calculatoare comunica direct cu imprimante de rețea - sau de a folosi un server de imprimare . Imprimantele de rețea stand-alone sunt disponibile și se conectează direct în LocalTalk , Ethernet, sau rețele 100BaseT. În cazul în care computerele comunica direct cu imprimanta de rețea, computerele Mac și Windows trebuie să utilizeze aceeași interfață de rețea. Multe imprimante în rețea conțin un slot care poate accepta un card de interfață de rețea, inclusiv LocalTalk și Ethernet 10/100BaseT, nivele OSI 1. Imprimante de rețea utilizează de asemenea un protocol de rețea , cum ar fi AppleTalk sau TCP / IP , pentru a comunica.[[46]](#footnote-46)

1. **Avantaje si Dezavantaje(**Tudor Cristina**).Concluzii finale** (Dumitran Tudor)

Cateva dintre avantajele oferite de AppleTalk sunt:

* Applet include automat AppleTalk in sistemele de operare din Macintosh
* este usor de implementat si configurat
* setarea unui mic spatiu de lucru este foarte simpla si deloc costisitoare
* userii nu trebuie sa stie numele dispozitivelor dintr-o retea

Cateva dintre cele mai importante dezavanaje sunt :

* nu se preteaza pentru retele de demensiuni foarte mari
* are o viteza redusa fata de alte retele LAN
* nu este rezonabila pentru aplicatii cu o largime de banda mare
* un echipament poate suporta doar o singura adresa AppleTalk
* traficul multicast scade performantele retelei.[[47]](#footnote-47)

In concluzie, AppleTalk este un sistem de retele foarte usor de instalat si de operat.

Modelul arhitectural este foarte apropiat de standardul OSI.AppleTalk suporta o mare varietate de implementari, inclusiv pentru Ethernet si Token Ring. In cazul AppleTalk, Ethernet poarte denumirea de EtherTalk, Token Ring de TokenTalk, si FDDI de FDDITalk.

Cele mai importante protocoale, folosite pentru adrese si identificare in retea si care tintesc sa faca sistemul complet auto-configurabil sunt AARP (protocolul de rezolutie al adresei AppleTalk) si NBP (protocolul de legatura a numelor).

AppleTalk dispune de doua variante de implementare.Astfel se deosebesc AppleTalk Phase 1, pentru retele de dimensiuni mici si AppleTalk Phase 2, pentru retele mai mari. Phase 2 este mai avansat ca performante de viteza si este suportat de mai multe protocoale decat predecesorul sau.

Adresarea este dinamica, cu fiecare calculator,atunci cand este deschis, alegandu-si ultima adresa folosita sau o noua adresa aleatoare. Computer-ul face broadcast pentru a verifica daca acea adresa este folosita. Daca este folosita, va face broadcast cu o alta adresa aleatoare pana cand gaseste una care nu este folosita.

Topologia AppleTalk este bus, tipul de cablu este STP(shielded twisted pair), iar conectorii sunt specializati.Accesul la mediu se face cu metoda CSMA/CA(collision avoidance) . Lungimea maxima de segment si de retea este de 300 metri. Numarul maxim de segmente interconectate este 8, si pot fi pana la 32 de noduri pe segment si 254 noduri pe retea. Viteza este de 230 Kbps. Sistemul de cablaj folosit se numeste LocalTalk. Este, in general, folosit la retele de dimensiuni mici.

In ceea ce priveste portabilitatea, AppleTalk Remote Access, sau ARA, a fost o grupare de protocoale , care permiteau AppleTalk-ului sa ruleze pe modem-uri. A devenit un produs major al Apple la jumatatea anilor 90, cand primele computere portabile si laptopuri au fost scoase pe piata. ARA a disparut usor la sfarsitul anilor 90, cand TCP/IP a preluat asupra majoritatii cerintelor de retele. In Mac OS X, acesta nu mai era necesar, deoarece Apple migrase software-ul de retea spre IP, si include un software de remote acces gratuit.

AppleTalk - PC , a permis comunicarea între PC-uri si Mac-uri cu software-ul TOPS instalat ( Mac-urile fără TOPS instalate puteau folosi aceeași rețea , dar numai pentru a comunica cu alte mașinile Apple). Sistemele de operare Windows pentru servere suportă AppleTalk începând cu Windows NT până la Windows Server 2003.

Avansarea tehnologiei TCP/IP in anii 1990 a condus la o reimplementare a suportului pentru acest protocol, iar AppleTalk nu a mai fost acceptat odata cu lansarea lui Mac OS X v10.6 in anul 2009.

# Bibliography

1. **Martin J. Michael , "Understanding the Network: A Practical Guide to Internetworking" ,Sams, 07.2000, chapter 3**
2. [**http://www.novell.com/documentation/nias41/?page=/documentation/nias41/iptuneun/data/h5bu6qn6.html**](http://www.novell.com/documentation/nias41/?page=/documentation/nias41/iptuneun/data/h5bu6qn6.html)
3. [**http://docwiki.cisco.com/wiki/AppleTalk**](http://docwiki.cisco.com/wiki/AppleTalk)
4. [**http://en.wikipedia.org/wiki/AppleTalk**](http://en.wikipedia.org/wiki/AppleTalk)
5. [**http://www.mactech.com/articles/mactech/Vol.03/03.04/AppleTalkFundamentals/index.html**](http://www.mactech.com/articles/mactech/Vol.03/03.04/AppleTalkFundamentals/index.html)
6. [**http://www.webclasses.net/3comu/intro/units/unit05/sec05d.html**](http://www.webclasses.net/3comu/intro/units/unit05/sec05d.html)
7. [**http://www.protocols.com/pbook/appletalk.htm**](http://www.protocols.com/pbook/appletalk.htm)
8. **http://www.macwindows.com/netbasic.html#AppleTalk**
9. [**http://netatalk.sourceforge.net/2.1/htmldocs/configuration.html#id2256608**](http://netatalk.sourceforge.net/2.1/htmldocs/configuration.html#id2256608)
10. [**http://netatalk.sourceforge.net/2.1/htmldocs/pap.1.html**](http://netatalk.sourceforge.net/2.1/htmldocs/pap.1.html)
11. [**http://www.macwindows.com/Network.html#Peer2Peer**](http://www.macwindows.com/Network.html#Peer2Peer)
12. [**http://en.wikipedia.org/wiki/AppleShare**](http://en.wikipedia.org/wiki/AppleShare)
13. [**http://www.linktionary.com/a/appletalk.html**](http://www.linktionary.com/a/appletalk.html)
14. [**http://en.wikipedia.org/wiki/LocalTalk-to-Ethernet\_bridge**](http://en.wikipedia.org/wiki/LocalTalk-to-Ethernet_bridge)
15. [**http://searchnetworking.techtarget.com/definition/AppleTalk**](http://searchnetworking.techtarget.com/definition/AppleTalk)
16. [**http://www.webopedia.com/TERM/A/AppleTalk.html**](http://www.webopedia.com/TERM/A/AppleTalk.html)
17. [**http://wiki.answers.com/Q/What\_is\_AppleTalk?#slide=1**](http://wiki.answers.com/Q/What_is_AppleTalk?#slide=1)
18. [**http://www.techopedia.com/definition/2631/appletalk**](http://www.techopedia.com/definition/2631/appletalk)
19. [**http://www.princeton.edu/~achaney/tmve/wiki100k/docs/AppleTalk.html**](http://www.princeton.edu/~achaney/tmve/wiki100k/docs/AppleTalk.html)
20. [**http://www.corecom.com/html/appletalk.html**](http://www.corecom.com/html/appletalk.html)
21. [**http://mirror.eu.oneandone.net/projects/media.ccc.de/documentation/vortraege/Felix%20-%20IP%20Routing/support/Appletalk.pdf**](http://mirror.eu.oneandone.net/projects/media.ccc.de/documentation/vortraege/Felix%20-%20IP%20Routing/support/Appletalk.pdf)
22. [**http://www.macworld.com/article/1043641/appletalk.html**](http://www.macworld.com/article/1043641/appletalk.html)

Cuprins

1. Scurt istoric al protocoalelor Apple1
2. Organizarea retelelor AppleTalk. Implementarea si tehnici de adresare 2
   1. AppleTalk faza 1 si AppleTalk faza 2 2
   2. Arhitectura si adresare3
   3. Regulile de angajament al unui echipament la reteaua AppleTalk6
   4. Invechirea si abandonul AppleTalk-ului6
3. Protocoalele AppleTalk si rolul acestora8
   1. Nivelul fizic si nivelul legatura de date8
   2. Nivelul retea11
   3. Nivelul transport13
   4. Nivelul sesiune18
   5. Nivelul prezentare si nivelul aplicatie19
4. Solutii de portabilitate20
5. Avantaje si concluzii finale 21
6. Bibliografie23

1. Informatie luata de pe pagina http://en.wikipedia.org/wiki/AppleTalk, http://www.princeton.edu/~achaney/tmve/wiki100k/docs/AppleTalk.html http://www.webopedia.com/TERM/A/AppleTalk.html, http://www.techopedia.com/definition/2631/appletalk [↑](#footnote-ref-1)
2. Informatie luata de pe pagina http://en.wikipedia.org/wiki/AppleTalk, http://www.princeton.edu/~achaney/tmve/wiki100k/docs/AppleTalk.html http://www.webopedia.com/TERM/A/AppleTalk.html, http://www.techopedia.com/definition/2631/appletalk [↑](#footnote-ref-2)
3. Informatie luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-3)
4. Poza luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-4)
5. Informatie luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-5)
6. Poza luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-6)
7. Informatie luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-7)
8. Informatie luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk, http://mirror.selfnet.de/CCC/documentation/vortraege/Felix%20-%20IP%20Routing/support/Appletalk.pdf [↑](#footnote-ref-8)
9. Poza luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-9)
10. Informatie luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk, http://www.corecom.com [↑](#footnote-ref-10)
11. Poza luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-11)
12. Informatie luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-12)
13. Informatie luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-13)
14. Poza luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk [↑](#footnote-ref-14)
15. Informatie luata de pe pagina http://en.wikipedia.org/wiki/AppleTalk, http://mirror.selfnet.de/CCC/documentation/vortraege/Felix%20-%20IP%20Routing/support/Appletalk.pdf [↑](#footnote-ref-15)
16. Informatii luate de pe http://en.wikipedia.org/wiki/AppleTalk [↑](#footnote-ref-16)
17. Informatii luate de pe http://en.wikipedia.org/wiki/AppleTalk [↑](#footnote-ref-17)
18. Informatii luate de pe http://en.wikipedia.org/wiki/AppleTalk, [↑](#footnote-ref-18)
19. Poza luata de pe www.eprcreations.com [↑](#footnote-ref-19)
20. Martin J. Michael , "Understanding the Network: A Practical Guide to Internetworking" ,Sams, 07.2000, chapter 3 [↑](#footnote-ref-20)
21. Informatie luata de pe pagina http://en.wikipedia.org/wiki/LocalTalk-to-Ethernet\_bridge [↑](#footnote-ref-21)
22. Martin J. Michael , "Understanding the Network: A Practical Guide to Internetworking" ,Sams, 07.2000, chapter 3 [↑](#footnote-ref-22)
23. Martin J. Michael , "Understanding the Network: A Practical Guide to Internetworking" ,Sams, 07.2000, chapter 3 [↑](#footnote-ref-23)
24. Martin J. Michael , "Understanding the Network: A Practical Guide to Internetworking" ,Sams, 07.2000, chapter 3 [↑](#footnote-ref-24)
25. Informatie luata de la pagina http://docwiki.cisco.com/wiki/AppleTalk#AppleTalk\_Physical\_and\_Data\_Link\_Layers [↑](#footnote-ref-25)
26. Martin J. Michael , "Understanding the Network: A Practical Guide to Internetworking" ,Sams, 07.2000, chapter 3 [↑](#footnote-ref-26)
27. Martin J. Michael , "Understanding the Network: A Practical Guide to Internetworking" ,Sams, 07.2000, chapter 3 [↑](#footnote-ref-27)
28. Informatie luata de pe pagina http://docwiki.cisco.com/wiki/AppleTalk#AppleTalk\_Physical\_and\_Data\_Link\_Layers [↑](#footnote-ref-28)
29. Martin J. Michael , "Understanding the Network: A Practical Guide to Internetworking" ,Sams, 07.2000, chapter 3 [↑](#footnote-ref-29)
30. Informatie luata de pe pagina http://www.protocols.com/pbook/appletalk.htm#RTMP [↑](#footnote-ref-30)
31. Informatie luata de pe pagina http://www.novell.com/documentation/nias41/?page=/documentation/nias41/iptuneun/data/h5bu6qn6.html [↑](#footnote-ref-31)
32. Informații luată de pe http://www.protocols.com/pbook/appletalk.htm [↑](#footnote-ref-32)
33. Poză luată de pe http://docwiki.cisco.com/wiki/AppleTalk#AppleTalk\_Echo\_Protocol [↑](#footnote-ref-33)
34. Sursă: http://www.linktionary.com/a/appletalk.html [↑](#footnote-ref-34)
35. Informații luate de pe : http://en.wikipedia.org/wiki/AppleTalk#AppleTalk\_Echo\_Protocol [↑](#footnote-ref-35)
36. Informații luate de pe : http://en.wikipedia.org/wiki/AppleTalk#AppleTalk\_Transaction\_Protocol [↑](#footnote-ref-36)
37. Informații luate de pe : http://en.wikipedia.org/wiki/AppleTalk#Printer\_Access\_Protocol [↑](#footnote-ref-37)
38. Informații luate de pe : http://netatalk.sourceforge.net/2.1/htmldocs/pap.1.html [↑](#footnote-ref-38)
39. Informații luate de pe : http://www.protocols.com/pbook/appletalk.htm#ZIP [↑](#footnote-ref-39)
40. Informații luate de pe : http://en.wikipedia.org/wiki/AppleTalk#Zone\_Information\_Protocol [↑](#footnote-ref-40)
41. Sursă foto : http://docwiki.cisco.com/wiki/AppleTalk#Zone\_Information\_Protocol [↑](#footnote-ref-41)
42. Informații luate de pe : http://en.wikipedia.org/wiki/AppleTalk#AppleTalk\_Data\_Stream\_Protocol [↑](#footnote-ref-42)
43. Informații luate de pe : http://en.wikipedia.org/wiki/AppleShare [↑](#footnote-ref-43)
44. [↑](#footnote-ref-44)
45. Informații luate de pe : http://en.wikipedia.org/wiki/AppleTalk#Cross-platform\_solutions [↑](#footnote-ref-45)
46. Informații luate de pe : http://www.macwindows.com/netbasc2.html [↑](#footnote-ref-46)
47. Informatii luate de pe paginile <http://www.macworld.com/article/1043641/appletalk.html> si http://zaielacademic.net/networking/apple\_talk.htm [↑](#footnote-ref-47)