Università degli Studi di Roma "La Sapienza"
Dipartimento INFOCOM

Nicola Blefari-Melazzi, Marco Listanti e Aldo Roveri

RETEMATICA

volume terzo:

infrastrutture per applicazioni telematiche

Versione intermedia
Dicembre 2001

I. INTERNET: STATO ATTUALE

Attualmente viene riconosciuto in modo unanime il ruolo centrale di Internet nell'attuale e futuro sviluppo delle infrastrutture per telecomunicazioni: è avvenuta cioè, da parte di Internet, la sostituzione dei tradizionali paradigmi (in primo luogo di quello telefonico), che hanno finora guidato l'evoluzione delle comunicazioni.

Internet è nata come infrastruttura per comunicazioni di dati (*rete di calcolatori*). Attualmente sta conoscendo un successo a livello mondiale che non ha precedenti per la sua estensione geografica, per la numerosità dei suoi utenti e per la vastità di interessi coinvolti. Al momento (all'inizio del nuovo secolo), se ne è iniziata una utilizzazione anche per *comunicazioni multimediali* a banda stretta o larga.

Il modello architetturale di Internet sarà discusso nel seguito. Qui è importante sottolineare che Internet non è una nuova rete, progettata e costruita ex novo o necessariamente alternativa ad altre soluzioni di reti per dati, ma l'unione di diverse reti, spesso pre-esistenti ed eterogenee tra loro. Le reti componenti Internet sono però omogenee al loro interno e possono operare in accordo ad uno qualsiasi dei paradigmi realizzati per comunicazioni di dati; la loro interconnessione è attuata da dispositivi che consentono a elaboratori connessi a reti diverse di poter comunicare.

Elemento caratterizzante di Internet è il modo in cui questa rete è nata e si è sviluppata. I primi studi sulla interconnessione di calcolatori furono effettuati alla fine degli anni sessanta. Si voleva allora sperimentare una tecnica che permettesse la condivisione di linee di comunicazione da parte di utenti attestati su sistemi diversi, sfruttando la tecnica della commutazione di pacchetto. In quegli anni il Ministero della Difesa degli USA (*Department of Defense*, *DoD*) incaricò l'organizzazione *ARPA* (*Advanced Research Project Agency*) di sviluppare e sperimentare una rete di calcolatori.

Dovendo avere applicazioni militari, uno dei requisiti fondamentali che tale rete doveva soddisfare era la resistenza ad attacchi che ne modificassero la configurazione (ad. es. per interruzione di linee di giunzione o di centri di commutazione). Sue caratteristiche dovevano quindi essere una struttura reticolare, magliata, non gerarchica e con un'elevata capacità di interconnessione e di interlavoro nell'ambito di una grande varietà di nodi di rete. Per lo stesso motivo si scelse di adottare una modalità di trasferimento senza connessione e senza garanzie di qualità del servizio, rimandando queste ultime ai livelli superiori dell'architettura protocollare. La rete nata da questo progetto, e denominata ARPANET, è stata il primo esempio di rete geografica a commutazione di pacchetto. Nel 1983 tale rete fu separata in due parti: una civile (ARPANET) ed una militare (MILNET). Nel 1985, sempre negli USA, la National Science Foundation finanziò lo sviluppo di una rete a lunga distanza (NSFnet) e di reti regionali, che consentirono di interconnettere LAN di università e di altri enti di ricerca a ARPANET.

Internet si è sviluppata a partire da questo nucleo iniziale fino ad estendersi in tutto il mondo e comprendendo non più solo organizzazioni pubbliche o di ricerca e al servizio della comunità scientifica, ma anche organizzazioni commerciali ed utenti privati.

A causa delle particolari esigenze sopra menzionate, e poste dall'organizzazione *ARPA,* si determinò una divergenza tecnica iniziale tra le scelte effettuate per ARPANET e quelle adottate dagli organismi di standardizzazione ISO e ITU-T. Questa divergenza è perdurata nel tempo attraversando gli anni settanta ed ottanta, durante i quali ARPANET cresceva notevolmente negli USA, espandendosi dall'ambito iniziale della difesa a quello dell'istruzione e della ricerca e conquistando sul campo un numero di utenti sempre maggiore. Nel frattempo ISO e ITU-T lavoravano alla definizione di un modello teorico di *interconnessione tra sistemi aperti* (l'Open Systems Interconnection, OSI) ed alla specifica di una serie di protocolli di comunicazione conformi a tale modello.

Fino alla fine degli anni ottanta era opinione diffusa che OSI ed i suoi protocolli sarebbero stati universalmente adottati. Tale previsione si è però dimostrata non corretta: la realtà Internet si è infatti talmente diffusa, sia negli USA che nel resto del mondo, che ha di fatto impedito ad OSI di

svilupparsi significativamente. Internet ha raggiunto gli obiettivi che OSI si prefiggeva, costituendo una base di interconnessione aperta ed indipendente dai costruttori.

Internet e i relativi mezzi di accesso cooperano, già attualmente e con prospettive di rapida crescita, per consentire un'ampia gamma di applicazioni che ne qualificano il carattere multi-valente, da infrastruttura per sostenere comunicazioni multimediali a deposito di informazioni a livello planetario, da piattaforma per lo sviluppo della "new economy" a mezzo per l'automazione di tutte le attività umane (gestione del lavoro, della casa, dell'intrattenimento, degli spostamenti, ecc.). La pervasività di tali valenze è descritta da vari indicatori. Ci si limita qui a presentarne alcuni. Ad esempio, in ambito mondiale il numero di clienti di Internet, che è stato di circa 300 milioni alla fine del 2000, dovrebbe sestuplicarsi entro la fine del 2005; alla fine del 2004 è poi previsto il superamento di questo numero nei confronti di quello riguardante gli utenti telefonici senza filo.

Inoltre, dal punto di vista delle applicazioni, sono menzionabili altri dati significativi:

- a metà 2000 il *fenomeno Web* era descritto da oltre 1,3 miliardi di pagine; già in queste condizioni il relativo volume di informazione superava la possibilità di un essere umano a visitarne lo 0,1% nell'arco della sua vita;
- il *commercio elettronico* è caratterizzato già attualmente da vendite per oltre 400 miliardi di US$; se ne dovrebbe però avere una quintuplicazione entro il 2003.

Altro fenomeno destinato a incidere profondamente sulla organizzazione della nostra vita domestica e lavorativa è rappresentato dalla diffusione di *"utensili informatici"* in aggiunta agli ormai tradizionali PC. Si prevede una esigenza di elaborazione "ovunque" (ricreativi, elettrodomestici, autoveicoli, controllori di traffico, ecc.) realizzata con "processori" orientati agli oggetti e con associati mezzi di comunicazione via Internet. La prospettiva è trasformare "prodotti" in "servizi", consentendo una larga varietà di applicazioni personalizzate. Si parla di un numero di utensili informatici che nel 2010 potrebbe raggiungere la quota di un miliardo.

Con quest'ultima prospettiva gli *accessi ad Internet* sono destinati ad assumere una configurazione unificata per qualunque tipo di applicazione o di servizio (voce, dati, immagini, ecc.).

Nel corso di questo capitolo, dedicato alla presentazione dello stato attuale di Internet, si partirà dalla descrizione dell'ambiente, delle modalità di standardizzazione e dei relativi servizi applicativi (par. I.1). Si passerà poi alle funzionalità preposte al trasferimento dell'informazione; in particolare il par. I.2 presenta qualche ulteriore dettaglio sulla infrastruttura e sulla sua architettura protocollare. Si prosegue con

- il protocollo IP, che qualifica il *servizio di rete* (par. I.3);
- i protocolli TCP e UDP, che curano il *trasporto dell'informazione* e che sono gestiti da estremo a estremo (par. I.4);
- tra questi una specifica attenzione è rivolta a TCP (par. I.5), che contribuisce alla fornitura di un servizio di strato nel modo *con connessione*.

Il capitolo si conclude con la descrizione del protocollo SNMP, preposto a *funzioni di gestione*, (par. I.6) e con le specificità di un accesso tramite un "*Internet Service Provider*" (par. I.7).

In questa trattazione si cercherà di presentare le modalità di funzionamento di Internet secondo opportune strutture logiche, al fine di dare una certa sistematicità all'esposizione. Avvertiamo però che alcuni aspetti di Internet, a causa del modo in cui questa infrastruttura si è evoluta, sfuggono ad un inquadramento in paradigmi ben definiti e, pur nel rispetto dei protocolli, sono possibili numerose eccezioni e soluzioni alternative ad una data modalità di funzionamento.

I.1. Il mondo e le applicazioni di Internet

Il paragrafo presenta l'ambiente in cui Internet si è sviluppata (§ I.1.1) e fornisce elementi sul relativo processo di standardizzazione (§ I.1.2); descrive infine le principali applicazioni attualmente disponibili (§ I.1.3).

I.1.1. L'ambiente

Internet è oggi una infrastruttura di comunicazione le cui risorse sono dislocate spazialmente in tutto il mondo, ma che sono viste dal singolo utente in modo trasparente, senza cioè che sia necessario sapere dove sono fisicamente, ed a cui è agevole accedere con strumenti progettati e realizzati per i suoi utenti.

Il servizio di rete è *senza connessione* e non fornisce alcuna garanzia sulla sua qualità: Internet si impegna a fare del suo meglio, ma non è possibile, in generale, garantire un trasferimento nel rispetto di determinati requisiti prestazionali (integrità informativa, ritardo di trasferimento, trasparenza temporale, etc.). Il compito di rendere la qualità di servizio adeguata alle esigenze degli utenti è demandato ai livelli applicativi residenti nei sistemi terminali.

Fra le principali caratteristiche ed i fattori di successo che hanno portato allo sviluppo esplosivo di Internet, si possono citare i seguenti:

- una *realizzazione più semplice* quale deriva da un modello architetturale decisamente meno complesso e articolato rispetto al modello OSI, almeno negli strati più bassi dell'architettura;
- una naturale *predisposizione a cooperare con le diversità* delle sotto-reti componenti, per le quali non è richiesta una comune architettura protocollare;
- una piena *convivenza di ambienti di comunicazione a gestione pubblic a o privata,* che nel loro insieme contribuiscono in modo significativo al trasporto delle informazioni;
- un offerta di servizi a *costi accessibili*: la relativa tariffazione è prevalentemente di tipo "*flat rate*", cioè non dipende dal volume dell'informazione scambiata o dalla distanza che separa l'origine e la destinazione;
- la completa *gratuità* e l'agevole *installazione* del *software di rete*, che presiede con i suoi protocolli alla operatività di trasporto dell'informazione; inoltre, inizialmente ed ancora oggi, tali protocolli sono stati distribuiti insieme al *sistema operativo Unix*; la grande diffusione di Unix, soprattutto nella comunità scientifica, ha trainato con sè anche i protocolli Internet;
- la ulteriore *gratuità* riguardante spesso il *software applicativo*, che consente di utilizzare a pieno le capacità di comunicazione dell'infrastruttura; gli applicativi più utili e più facili da usare si diffondono in tal modo con grande rapidità, come è avvenuto in passato per applicazioni quali *Gopher* (applicativo per la ricerca di informazione) e, più recentemente, per il *WWW (World Wide Web*), cioè per una applicazione multimediale di banca di dati distribuita;
- la *snellezza* e la *rapidità* del *processo di standardizzazione*, che prevede sempre verifiche in rete: per un protocollo sono richieste almeno due diverse implementazioni interoperanti ed una solida esperienza in campo prima di consolidarlo in uno standard;
- l'ampia e libera *diffusione delle idee di lavoro e delle specifiche dei protocolli*, che è stata ed è tuttora assicurata anche e soprattutto per mezzo della stessa infrastruttura; la disponibilità di informazioni generali e dettagliate sui protocolli, e sulle relative implementazioni, anche nei primissimi stadi di sviluppo, ha svolto un ruolo essenziale nella loro diffusione; un simile impegno verso la documentazione pubblica e gratuita a questo livello di dettaglio è inusuale; i benefici di questo modo di procedere hanno avuto significative conseguenze non solo sulla realizzazione di Internet ma anche, più in generale, sullo sviluppo delle telecomunicazioni; in particolare, il sito *www.ietf.org* fornisce sia informazioni relative all'organizzazione generale di Internet e sui suoi organismi, sia i relativi standard.

Le motivazioni che sono alla base dell'attuale struttura di Internet sono identificabili sostanzialmente nelle *esigenze di inter-lavoro* tra ambienti di comunicazione sorti nel passato, e tuttora in corso di sviluppo, come risposta a richieste di una utenza fisiologicamente differenziata per la diversità delle finalità applicative e delle relative prestazioni. L'obiettivo dell'interlavoro è assicurare comunicazioni su base universale che consentano di porre in corrispondenza utenti comunque dislocati in ambito geografico e comunque dotati in termini di risorse di comunicazione.

A ulteriore commento di quanto ora detto vale la pena sottolineare che nel passato è stato difficile far si che un'unica tipologia di rete potesse rispondere in modo esauriente alla esigenze di comunicazione di ogni possibile utente. Ad esempio, le reti in area locale consentivano elevate velocità di trasferimento, ma erano limitate geograficamente; il viceversa accadeva per le reti in area geografica. Inoltre la fisiologica esistenza di diversi produttori ha portato spesso a sviluppare sistemi di telecomunicazione non compatibili tra loro; da ciò è derivato lo sviluppo di diversi paradigmi di comunicazione e la attuale co-esistenza di diverse reti di telecomunicazioni. Oggi è tecnicamente possibile realizzare un'unica infrastruttura di comunicazione che soddisfi i requisiti di tutti gli utenti in modo efficiente, ma risulta economicamente improponibile sostituire tutti i sistemi già operativi.

Inoltre gli utenti desiderano una connettività universale; l'importanza e l'utilità di una rete di telecomunicazioni sono legate anche al numero dei suoi utenti. Ad esempio, la rete telefonica risulta molto utile proprio perché i suoi utenti sono numerosi e quindi per il suo tramite è possibile comunicare con un grande numero di persone e perché la rete appare ad essi come un solo sistema di comunicazione. La rete telefonica non avrebbe avuto lo sviluppo che conosciamo se fosse stata costituita da diverse sotto-reti con diversi standard e protocolli e non comunicanti tra loro; in tal caso infatti da un dato apparecchio telefonico non sarebbe stato possibile raggiungere tutti gli altri utenti, ma solo un loro sotto-insieme.

In altre parole si desidererebbe una singola rete a cui tutti possano connettersi e tramite la quale sia possibile raggiungere chiunque. E' apparso però sinora difficile realizzare una soluzione unitaria e quindi si è preferito rendere possibile la comunicazione tra diverse tipologie di reti. Tali considerazioni hanno portato al concetto di *inter-rete*, e cioè di una infrastruttura *fisicamente* costituita da:

- un certo numero di reti componenti (*sotto-reti*), cui sono connessi gli apparecchi terminali, chiamati *host*;
- un certo numero di dispositivi necessari per interconnettere le sotto-reti componenti e usualmente denominati *router*.

Gli host possono essere super-computer paralleli, mini-computer, workstation, semplici personal computer o calcolatori portatili. I ritmi binari di trasferimento delle informazioni possono essere molto variabili, sia nel tempo che in funzione delle coppie origine-destinazione considerate (da pochi bit/s fino a centinaia di Mbit/s).

Da un punto di vista *logico*, l'elemento distintivo di una inter-rete è rappresentato dall'insieme di *procedure di inter-lavoro* (o protocolli di comunicazione) necessarie per interconnettere le sotto-reti e per permettere a qualunque host *A* connesso alla inter-rete di colloquiare con qualunque altro host *B* anch'esso connesso alla inter-rete. La comunicazione tra *A* e *B* deve essere possibile indipendentemente dalle sotto-reti a cui *A* e *B* sono direttamente connessi e dal numero e dalla tipologia delle altre sotto-reti eventualmente coinvolte in tale comunicazione.

Internet è l'esempio più significativo di inter-rete oggi operativa su base mondiale. Sono da sottolineare le sue specificità riassumibili in tre aspetti fondamentali:

- la relazione tra l'infrastruttura nel suo insieme e le sotto-reti componenti;
- l'indipendenza dell'interfaccia utente-rete dalle specificità della sotto-rete di accesso;
- la coesistenza tra i protocolli Internet e quelli che appartengono all'architettura di ogni singola sotto-rete componente.

Circa il primo di questi aspetti, la *struttura di Internet* è nascosta ai suoi utenti. Questi ultimi e i programmi applicativi di loro interesse non devono preoccuparsi dei dettagli implementativi della

rete fisica. Non si vuole imporre una pre-determinata *topologia* di rete. L'aggiunta di una nuova sotto-rete a Internet, e quindi alle sotto-reti pre-esistenti in Internet, non si effettua connettendo la nuova sotto-rete ad un nodo centrale o con l'aggiunta di connessioni fisiche dirette tra la nuova sotto-rete e *tutte* le altre pre-esistenti. E' sufficiente connettere la nuova sotto-rete ad *una* qualsiasi altra sotto-rete già connessa ad Internet. Le informazioni sono quindi trasferite da una sotto-rete ad un'altra utilizzando, in generale, altre sotto-reti intermedie e attraversando diversi router. Non vengono invece coinvolte sotto-reti intermedie qualora una sotto-rete che origina informazioni sia *direttamente* connessa alla sotto-rete cui queste informazioni sono dirette, ovvero quando è possibile trasferire informazione dall'una all'altra attraversando un solo router.

Relativamente poi all'*interfaccia utente-rete*, è assicurata indipendenza di questa interfaccia dall'architettura specifica della sotto-rete di accesso. L'insieme di operazioni necessarie per trasferire dati è indipendente dalla tecnologia della sotto-rete di accesso e dal sistema di destinazione. Un programma applicativo di comunicazione non deve tenere in conto la topologia e la struttura di sotto-rete.

Riguardo infine alla *coesistenza tra le pile protocollari* del modello Internet e quelle dei modelli di sotto-rete, i protocolli Internet si vanno ad aggiungere a quelli già esistenti all'interno delle sotto-reti componenti; questi ultimi non devono essere modificati. In tal modo alcune funzioni possono risultare duplicate e quindi si possono verificare inefficienze, ma questo modo di procedere determina notevoli semplificazioni operative. Prima dell'avvento di Internet le reti per dati erano in gran parte diverse e non inter-comunicanti. Le sotto-reti di Internet sono rimaste diverse, nel senso che sono basate su diverse architetture e protocolli (e di per se autonome), ma i protocolli e i router di Internet hanno fatto sì che potessero colloquiare tra loro. Essendo Internet costituita da diverse tipologie di rete, ognuna, in generale, di proprietà di enti diversi, la sua modalità di gestione è necessariamente di tipo distribuito.

Prima di procedere oltre è utile fare una precisazione. La trattazione che segue fa riferimento a host *direttamente* connessi ad Internet e dotati di un proprio, stabile ed univoco indirizzo. Questo non è il caso di host che si connettono ad Internet per il tramite di un *Internet Service Provider* (ISP), usando tipicamente la rete telefonica commutata. Le modalità di funzionamento di Internet sono state infatti definite con riferimento ad host direttamente connessi e quindi è necessario fare riferimento a questa situazione. In particolare, quando un utente si connette ad Internet tramite un ISP, prima di poter iniziare effettivamente uno scambio informativo con altri host, riceve dall'ISP opportune informazioni di configurazione, tra cui un indirizzo che lo identifica nell'ambito di quella sessione, ma che potrà essere diverso in sessioni successive. Ciò significa anche che un utente di questo tipo è raggiungibile dall'esterno solo tramite il suo ISP, dal momento che l'utente in questione, non disponendo di un indirizzo stabile ed univoco in Internet, non è "conosciuto" dagli altri host connessi ad Internet.

I.1.2. La standardizzazione

L'Ente responsabile della definizione dei protocolli e delle regole di funzionamento di Internet è denominato "*Internet Engineering Task Force"* (IETF). Qui citiamo la presentazione dell'IETF reperibile in *www.ietf.org/overview.htm*:

« La Internet Engineering Task Force (IETF) è una comunità internazionale, aperta, di progettisti di rete, di operatori, di costruttori, di venditori e di ricercatori interessati all'evoluzione ed al regolare funzionamento di Internet. E' aperta a tutti gli interessati.

L'attività di ricerca&sviluppo è organizzata in Gruppi di Lavoro (Working Groups). I Gruppi di Lavoro sono organizzati in Aree, ognuna responsabile di una certa tematica (routing, transport, security, etc.). Gran parte del lavoro è svolta mediante "mailing lists", ovvero via posta elettronica. L'IETF organizza tre riunioni ogni anno. Ogni Area è gestita dagli Area Directors (AD). Gli AD sono membri dell'Internet Engineering Steering Group (IESG). La Internet Architecture Board (IAB) si occupa di questioni architetturali di portata più generale e risolve eventuali conflitti che

l'IESG non sia riuscita a risolvere. L'IESG e l'IETF sono presiedute dal General Area Director, che è anche membro di diritto dell'IAB.

L'Internet Assigned Numbers Authority (IANA) è l'autorità centrale che coordina l'assegnazione dei parametri numerici dei protocolli (inclusi gli indirizzi IP ed i numeri di porta TCP). L'Internet Society (ISOC) infine sopraintende all'IANA, all'IESG e all'IAB. ».

Secondo l'IETF, Internet è definita come la cooperazione di reti autonome ed interconnesse, non rigidamente organizzata (loosely organized), che supporta comunicazioni tra host attraverso il *volontario* rispetto di protocolli e procedure aperti, definiti dagli Internet Standards. Il "credo" dell'IETF è "We reject kings, presidents and voting. We believe in rough consensus and running code" (IETF Credo, Dave Clark, 1992).

Come si vede, la comunità Internet tiene a preservare lo spirito potremmo dire pioneristico originale, ma la stessa importanza di Internet sta spingendo sempre più verso una organizzazione più gerarchica e formale, seppure nel rispetto del suo credo e soprattutto nell'apertura verso chiunque voglia collaborare e nella diffusione dei protocolli.

In Internet, gli standard sono denominati *Request For Comments* (RFC). Ciò in accordo alle procedure seguite inizialmente (ed in parte ancora oggi) dalla comunità Internet, in cui di fronte ad un problema si chiedeva a chiunque fosse interessato di proporre soluzioni. Originariamente, quindi, le RFC non erano standard, ma semplicemente comunicazioni tra gli addetti ai lavori. Attualmente esistono due tipi di RFC:

- *For Your Information RFC* (FYI RFC): sono di natura descrittiva, riportano argomenti di natura generale o perfino l'organizzazione delle riunioni della comunità Internet ed il modo di vestirsi alle stesse.
- *Standard RFC* (STD RFC): sono i veri e propri standard adottati in Internet.

La strada che conduce ad una RFC standard parte tipicamente da un cosiddetto *Internet-Draft*, che è un documento di lavoro proponibile da un qualunque individuo o Ente interessato e posto a disposizione di tutti tramite i siti dell'IETF e le mailing lists. Un Internet-Draft ha una validità di sei mesi, trascorsi i quali decade, a meno che non sia ritenuto interessante e quindi promosso a RFC (eventualmente con opportune modifiche/integrazioni). Un Internet-Draft può essere sostituito, migliorato o cancellato in ogni momento durante il suo periodo di validità.

I.1.3. Le applicazioni

Nell'ambiente Internet, i protocolli di strato con rango superiore al 4 (nel senso OSI) sono denominati *Servizi Applicativi* e le loro funzioni (a differenza di quanto previsto nel modello OSI) non sono divise in termini di strato di sessione, di presentazione e di applicazione.

I Servizi Applicativi in Internet operano prevalentemente in accordo al paradigma *client/server*; hanno cioè una componente client ed una server. Tale paradigma è importante anche perché è alla base di molti sistemi di comunicazione ed è fondamentale per comprendere le modalità di funzionamento di algoritmi distribuiti.

Il termine "*server*" si applica a qualunque processo che offre un servizio e che può essere raggiunto attraverso la inter-rete. I server accettano le richieste, le elaborano, effettuano il servizio richiesto e restituiscono il risultato al richiedente. Un processo applicativo è invece un "*client*" quando invia una richiesta ad un server ed aspetta la relativa risposta. Ad esempio, un semplice server "ora esatta" fornisce l'ora attuale ogni volta che un client gli invia una richiesta. I server gestiscono le risorse e i client richiedono servizi alle risorse.

La durata in attività delle componenti client e server è diversa; un server inizia ad essere attivo prima che l'interazione con il client inizi e continua ad accettare richieste e ad inviare risposte senza cessare di essere attivo; la componente client si attiva al momento di inviare una richiesta e si disattiva una volta ricevuta la risposta. Normalmente la componente server inizia ad operare al momento dell'accensione di un host e, quando invocata, fornisce alla componente client i mezzi e le procedure o i dati di cui questa ha bisogno.

Molti dei servizi applicativi richiedono un'autorizzazione all'uso: un utente può usare un servizio applicativo se dispone di una *utenza* (account), concessa dall'amministratore del sistema remoto (server), a cui accede mediante una *parola d'ordine* (password). In tal caso, la componente

client deve comunicare a quella server il nome dell'utenza e la relativa parola d'ordine, prima di poter richiedere uno specifico servizio.

Si descrivono qui di seguito i più importanti e diffusi servizi applicativi che la comunità Internet mette a disposizione dell'utenza. Ognuno di tali servizi opera in accordo ad uno specifico protocollo che governa lo scambio di informazioni tra la componente client e quella server. Inoltre, un dato servizio applicativo può essere implementato nell'ambito di diverse applicazioni, basate su specifiche piattaforme operative e personalizzate o arricchite da interfacce di utente e più o meno sofisticate.

I servizi applicativi qui descritti sono raggruppati in alcune categorie, che hanno l'unico scopo di strutturare l'esposizione e che non corrispondono ad alcun tentativo di classificazione. Le categorie includono servizi di: (i) *emulazione di terminale* per sessioni interattive remote; (ii) *scambio di messaggi* con il possibile corredo di una *gestione di liste di indirizzi* (mailing list); (iii) *gestione di archivi* (files); (iv) *reperimento e acquisizione di informazioni*, normalmente strutturate in *documenti;* (v) *repertorio* (directory); (vi) *interazione conversazionale*.

I.1.3.1. Emulazione di terminale

In questa categoria sono inquadrabili gli applicativi *TELNET* e *X-Window,* che permettono agli utenti di un sistema l'accesso ad applicazioni in altri sistemi remoti, come se fossero direttamente connessi a tali sistemi.

Più in particolare l'applicativo TELNET consente di aprire sessioni interattive su host ovunque ubicati all'interno della inter-rete (*login remoto*) purché si disponga di un'autorizzazione all'uso (utenza + parola d'ordine). Il terminale emulato è di tipo testuale.

La componente client di TELNET consente quindi di utilizzare le funzioni di un calcolatore remoto come se il terminale che si usa fosse uno dei terminali locali del sistema remoto. L'host di destinazione deve contenere la componente server di TELNET. Il procedimento può essere iterativo; ovvero ci si può collegare ad un host remoto e da lì collegarsi ad un altro host e così via.

X-Window è simile a TELNET, ma fornisce anche *funzionalità grafiche*. Consente ad un utente di accedere contemporaneamente a diverse applicazioni in altri sistemi come se fosse direttamente connesso a tali sistemi. La principale differenza rispetto a TELNET è che il terminale emulato è di tipo grafico e quindi consente di usare anche applicazioni remote che fanno uso di visualizzazioni non solo testuali.

I.1.3.2. Scambio di messaggi

Sono qui presentati la *Posta elettronica,* il *servizio News* e l'applicazione *Listserv*.

La *Posta elettronica* (E-mail) è uno dei servizi applicativi più usato in Internet. La sua popolarità deriva dalla facilità di uso che consente di trasferire velocemente sia piccole note che voluminosi documenti. I vantaggi sono ovvii: il messaggio ricevuto può essere registrato e catalogato; inoltre può essere modificato o usato come base per successivi lavori, al contrario di un messaggio ricevuto su carta o a voce che necessita di essere trascritto. Inoltre non richiede che mittente e destinatario siano contemporaneamente presenti. Un utente può inviare un messaggio ed il destinatario può leggerlo in un tempo successivo. Gli applicativi di posta elettronica permettono di scambiare messaggi tra utenti utilizzando il protocollo di comunicazione *SMTP* (Simple Mail Transfer Protocol).

L'obiettivo che SMTP si propone è offrire i meccanismi per trasferire messaggi in forma elettronica secondo modalità affidabili ed efficienti attraverso Internet. A tal fine, SMTP usa TCP come protocollo di strato 4 e quindi adotta una modalità di trasferimento con connessione, che consente di soddisfare l'obiettivo di affidabilità.

A fronte di una richiesta di invio di E-mail da parte dell'utente, il "sender-SMTP" stabilisce una connessione bi-direzionale con il "receiver-SMTP". La connessione è bi-direzionale, in quanto il servizio è di tipo confermato, cioè il destinatario invia al mittente messaggi di riscontro per confermare la corretta ricezione del messaggio inviato dall'utente.

Preliminarmente all'invio del messaggio vero e proprio, il "sender-SMTP" verifica la disponibilità del "receiver-SMTP" ad accettare questo messaggio e, se non vengono riscontrati problemi (indisponibilità del receiver, utente sconosciuto, etc.), si avvia il processo di trasferimento del messaggio generato dall'utente.

Il formato dei messaggi è regolato da uno standard. SMTP prevede anche funzionalità di traduzione tra diversi formati e l'uso di gateway verso sistemi di E-mail non basati su TCP/IP. Ha diverse potenzialità: uso di alias, liste di destinatari, risposte automatiche, etc.

Netnews o Usenet o News è un'applicazione per la *condivisione di messaggi* che permette di scambiarli elettronicamente usando un formato standard.

I messaggi scambiati su Netnews sono organizzati per argomenti all'interno di categorie chiamate "newsgroup". I messaggi possono contenere sia esclusivamente testo che informazioni binarie codificate; possono contenere anche intestazioni informative sul mittente, su quando e dove il messaggio è stato messo in rete, dove è transitato ed altre informazioni amministrative.

Esistono oggi migliaia di *categorie newsgroup* riguardanti i più svariati argomenti. Il contenuto dei messaggi scambiati nei newsgroup non è in genere regolamentato, anche se qualche newsgroup ha un moderatore che esamina i messaggi prima che siano distribuiti e decide quali siano appropriati per la diffusione.

La componente client di News è chiamata "*news reader*" ed utilizza *NNTP* (Network News Transfer Protocol) quale protocollo per ricevere informazioni da opportune basi di dati.

Per concludere questa categoria, *Listserv* è un'applicazione per la *gestione di liste distribuite* che permette a gruppi di utenti con interessi comuni di comunicare tra loro tramite Posta elettronica, realizzando al tempo stesso un uso efficiente delle risorse di rete.

La funzione principale di Listserv è quella di operare su liste di indirizzi, ossia di realizzare la distribuzione di posta elettronica ad un gruppo di utenti stabilendo un vero e proprio forum di utenti su argomenti di interesse comune. Tale servizio è utile per uno scambio di idee e di informazioni, poiché è la stessa applicazione Listserv che si incarica di recapitare a tutti i partecipanti alla mailing list l'informazione inviata da un suo utente, evitando così a quest'ultimo la replica e l'invio dello stesso messaggio a tutti i componenti della lista.

Listserv fornisce inoltre gli strumenti per effettuare monitoraggio ed archiviazione del traffico di posta elettronica, funzioni di file server e ricerca di archivi in basi di dati.

I.1.3.3. Gestione di archivi

La gestione di files, per scopi di archiviazione, di trasferimento, di visualizzazione e di modifica/cancellazione da posizione remota, rientra, seppure con differenti possibilità operative, nei compiti degli applicativi *FTP* (File Transfer Protocol), *TFTP* (Trivial FTP) e *NFS* (Network File System).

FTP è uno dei servizi applicativi più usati ed è responsabile di una rilevante quantità del traffico totale. La principale funzione di FTP è il *trasferimento* efficiente ed affidabile *di archivi* tra host remoti. Consente anche di visualizzare il contenuto di sistemi di archiviazione remoti e di modificare o cancellare files e directories ivi residenti.

Supporta il trasferimento di quattro tipi di files: *Binari*, *ASCII*, *EBCDIC* e "*paged*". Fornisce funzionalità di protezione dei file trasferiti e compressione di dati; non fornisce funzionalità di traduzione o di conversione di files.

Nell'effettuare il trasferimento di un file, la componente client di FTP realizza innanzitutto una connessione di controllo con la componente server. Attraverso questa connessione vengono inviati al server FTP i comandi, opportunamente codificati, che originano da richieste dell'utente. Il server FTP interpreta tali comandi ed effettua il trasferimento dei dati richiesti. Per garantire l'affidabilità dei dati trasferiti, usa TCP come protocollo di trasporto .

TFTP è un protocollo simile a FTP, trasferisce archivi tra host remoti, ma fornisce meno funzionalità di FTP: non consente di proteggere i files trasferiti e di gestire directories. Usa UDP (invece di TCP) come protocollo di trasporto.

NFS consente di utilizzare un *sistema di archiviazione* (file system) *remoto* allo stesso modo di uno locale. Grazie a NFS è possibile vedere, copiare, modificare e cancellare files residenti in un sistema remoto in modo del tutto trasparente per l'utente. A differenza di FTP, con NFS l'utente può anche ordinare l'esecuzione di un qualunque processo ed usare qualsivoglia file (locale o remoto) per l'input e l'output di tale processo. Viene poi consentito all'utente finale di eseguire tutte le azioni appena descritte mediante gli stessi comandi del sistema operativo che l'utente usa nel proprio sistema, per le stesse azioni riferite al sistema locale. Non è necessario quindi che l'utente usi in modo esplicito il tramite di un'applicazione di rete per interagire con un sistema remoto; NFS gestisce tutte le operazioni necessarie per tali interazioni senza che l'utente finale ne abbia coscienza: opera cioè, per usare un termine gergale, "*nello sfondo*" (background). In Fig. I.1.1 le directories ed i files dei sistemi remoti "berkeley" ed "osaka" sono accessibili come se fossero residenti sul sistema locale, a meno solo di eventuali limitazioni nella velocità di accesso. Sempre in gergo, si dice che i file systems remoti sono "*montati*" sul sistema locale.

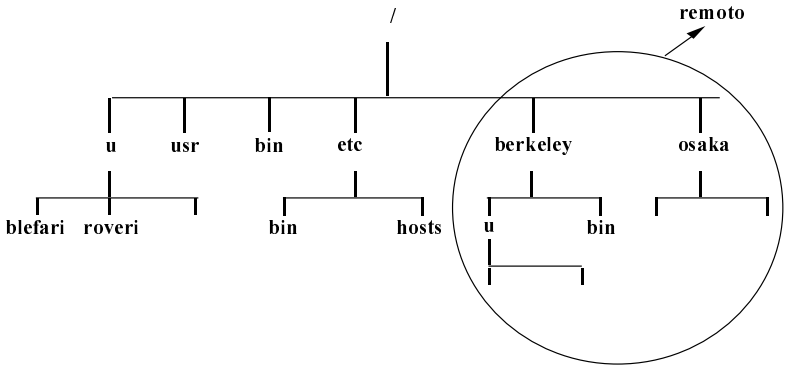


Figura I.1.1 - Esempio di funzionamento di NFS

I.1.3.4. Reperimento e acquisizione di documenti

Sono inclusi gli applicativi *World Wide Web*, *Gopher* e *WAIS* (Wide Area Information Server), tra cui il primo è attualmente il servizio più usato in Internet.

World Wide Web, chiamato anche WWW o W3, è un *sistema informativo multimediale* basato su ipertesti la cui origine risale al 1989. Consente ad un utente di acquisire un documento situato in un sistema remoto, usando la componente client di WWW (denominata "*browser*"). In ogni host in cui siano contenuti documenti deve invece operare la componente server. La particolarità di WWW è la tipologia dei documenti acquisibili.

I documenti WWW sono:

- *multimediali* e cioè contengono informazioni testuali, grafiche e sonore, eventualmente compresse mediante opportuni algoritmi per ridurre il volume di traffico scambiato durante il trasferimento;
- *ipertestuali*: alcune parole o immagini o zone del documento acquisito contengono *collegamenti* (link) ad altri documenti; l'utente, selezionando uno di questi collegamenti, acquisisce il relativo documento.

Quest'ultima caratteristica è quella che ha determinato lo sviluppo attuale del WWW. Un particolare documento può contenere diversi collegamenti, ognuno dei quali è un *puntatore* ad un altro documento; ogni documento può essere situato in un qualunque host connesso ad Internet. L'utente, sfogliando il documento in questione, può scegliere di seguire uno qualsiasi di questi collegamenti. L'aspetto più comodo, che rende il WWW particolarmente semplice da usare, è che ogni collegamento contiene anche l'indirizzo dell'host in cui il documento cercato si trova. In tal modo, l'utente può acquisire diversi documenti, localizzati ovunque nel mondo, senza bisogno di sapere dove essi siano fisicamente memorizzati. Il processo può divenire iterativo per cui da un

documento l’utente può passare, tramite apposito collegamento, ad esaminare un altro documento che a sua volta contiene collegamenti ad altri documenti e così via.

Di fatto l’intero insieme del WWW (e cioè tutti i documenti accessibili di tipo WWW) appare all’utente, nella sua interezza, come se fosse disponibile nel suo host, a meno di eventuali limitazioni nella velocità di accesso. L’utente non è tenuto a sapere dove sia localizzato uno specifico documento e non deve digitare comandi o interrogare basi di dati alla ricerca di questa locazione.

Ad esempio, quando una nuova parola o un nuovo concetto sono introdotti in un testo, WWW offre i mezzi per collegarsi ad un altro documento in cui vengono forniti maggiori dettagli a riguardo, in modo tale che il lettore può aprire il secondo documento selezionando la parola o il concetto sconosciuto; questo documento può a sua volta contenere ulteriori riferimenti ad altra documentazione residente in altri sistemi. E’ ciò che si intende per “navigare” nella rete, seguendo un “albero” di documenti che si può biforcare in ogni momento.

L'accesso ai server WWW avviene tramite *applicativi di tipo client*. Questi applicativi sono disponibili per diverse piattaforme software. Gli esempi più noti e diffusi di tali applicativi sono *Netscape* e *Internet Explorer*. Spesso le applicazioni che implementano la componente client di WWW contengono anche componenti client di altri servizi applicativi, come ad esempio FTP o News. Tali applicazioni contengono inoltre ulteriori moduli software (non servizi di rete, ma applicazioni locali) necessari per:

- convertire il formato di alcuni documenti;
- decomprimere documenti codificati con opportuni algoritmi per ridurre il volume di traffico scambiato;
- presentare in modo adeguato l’informazione (ad es. video o audio) all’utente finale.

Il servizio applicativo WWW si basa su due protocolli:

- *HTML* (Hyper Text Markup Language), che definisce il formato dei documenti;
- *HTTP* (Hyper Text Transfer Protocol), che consente di trasferire i documenti.

HTTP usa sia TCP che UDP, instaurando contemporaneamente anche più di una connessione TCP per il trasporto di diverse parti del documento.

Internet Gopher, o più semplicemente Gopher, è un servizio distribuito per l’*acquisizione di documenti*, che consente di esplorare, cercare e acquisire informazioni residenti in differenti locazioni. E’ simile a WWW, ma a questi precedente e sta diventando obsoleto, poiché offre meno funzionalità. L'interfaccia tipica di questo servizio suddivide le informazioni in una serie di menu nidificati, che rispecchiano l'organizzazione delle informazioni in directory, sub-directory e file.

Dal punto di vista dell'utente, tutte le informazioni sembrano essere dislocate nello stesso posto, sebbene alcune sub-directory e il loro contenuto possano trovarsi fisicamente su sistemi remoti gestiti da altri server Gopher. Tale struttura gerarchica delle informazioni porta alla definizione di uno “*spazio Gopher*”, cioè di una struttura logica unitaria composta da informazioni contenute in server Gopher ubicati spazialmente in luoghi diversi.

La tipologia delle informazioni gestite da questo servizio comprende file di testo, file binari, immagini e suoni.

Infine, *WAIS* è un sistema per il *recupero di informazioni* distribuite nella rete; aiuta l'utente nella ricerca sulle basi di dati presenti in Internet. Le informazioni contenute in tali basi di dati sono generalmente di tipo testo, ma possono contenere anche suoni, immagini e filmati, con un vasto campo di applicazioni.

L'organizzazione di queste basi di dati varia a seconda dell'ente che gestisce un particolare WAIS, mantenendo comunque una semplice interfaccia d'utente, al quale vengono mascherate le differenze. Tale interfaccia fa uso generalmente del linguaggio naturale, mediante il quale è possibile impostare tutti i parametri necessari alla ricerca.

L'architettura del servizio è strutturata in modo tale che, a seguito di una ricerca avviata dall'utente, il centro servizi a cui tale utente è collegato si incarica di attivare le opportune operazioni di ricerca sulle basi di dati selezionate, eventualmente remote. Il risultato di tale ricerca

si concretizza in un insieme di documenti, che contengono le stringhe oggetto della ricerca con i relativi riferimenti necessari per il recupero degli stessi attraverso la rete.

I.1.3.5. Repertorio

Lo scopo degli applicativi qui inquadrati è fornire servizi di *repertorio* (directory) finalizzati a

- *indirizzamento dei siti* presso i quali sono reperibili informazioni di interesse per l'utente;
- *reperimento di indirizzi* degli utenti registrati.

Gli applicativi inseriti in questa categoria sono *ARCHIE* e *WHOIS*, che riguardano il primo e il secondo di questi scopi, rispettivamente.

ARCHIE è un sistema informativo nato per offrire un servizio *di directory elettronico centralizzato*, mediante il quale è possibile localizzare le informazioni nell'ambito di Internet; allo stato attuale si contano un centinaio di siti distinti che offrono il servizio.

L'uso principale che viene fatto di tale sistema informativo consiste nell'interrogare una base di dati di carattere generale, che contiene informazioni su host ad accesso libero presenti nella rete; il risultato dell'interrogazione è una lista di indirizzi, presso i quali è possibile reperire le informazioni oggetto della ricerca.

Il servizio *WHOIS* fornisce un sistema di *directory elettronico* per gli utenti registrati in Internet. Fornisce cioè i mezzi per identificare indirizzi di posta elettronica, indirizzi postali e numeri telefonici.

La base di dati principale con le informazioni generali sulla rete (organizzazioni, siti, reti, persone, etc.) è gestito da un ente denominato "INTERnet Network Information Center" (INTER-NIC). Allo stato attuale, i nomi dei gestori dei domini registrati sono automaticamente introdotti nella base di dati quando le autorità di coordinamento di Internet concedono una numerazione specifica di rete (*indirizzo IP*).

L'accesso ai vari server WHOIS attualmente presenti in rete è reso possibile o attraverso sessioni interattive Telnet con siti opportuni, o facendo uso di appositi programmi applicativi.

In aggiunta a tali modalità di accesso, INTER-NIC offre per le proprie basi di dati un'interfaccia di tipo E-mail, per consentire l'accesso anche a quella fascia di utenza che è dotata soltanto di posta elettronica, o che non appartiene a Internet.

I.1.3.6. Interazione conversazionale

In questa categoria può essere citato il servizio *TALK*, che consente il dialogo in tempo reale tra due utenti. Genera, sullo schermo degli utenti, una finestra divisa in due parti: in una compare ciò che si scrive, mentre nell'altra si presenta ciò che scrive l'utente remoto.

I.2. Architettura e protocolli

Vengono qui trattate la struttura (§ I.2.1) e l'architettura protocollare (§ I.2.2) di Internet. Vengono anche date alcune prime indicazioni sui protocolli dell'ambiente Internet (§ I.2.3).

I.2.1. Struttura

Come già detto, Internet è una particolare inter-rete costituita dall'interconnessione di sotto-reti in generale tra loro eterogenee, ognuna delle quali utilizza protocolli degli strati di trasferimento (cioè degli strati fisico, di collegamento e di rete secondo il modello OSI), che sono in generale diversi da quelli usati da altre sotto-reti, fermo restando il fatto che ogni sotto-rete sia omogenea al suo interno. Grazie a questi soli protocolli gli host connessi a una sotto-rete possono scambiare informazioni con host appartenenti alla stessa sotto-rete, ma non con host connessi ad altre sotto-reti, quando queste non sono omogenee con la precedente. I router, e cioè i già citati sistemi di interconnessione tra le sotto-reti componenti, operano a livello di strato di rete (secondo il modello

OSI). Fino a qualche tempo fa la comunità Internet usava per i sistemi di interconnessione la denominazione “gateway”, che è però ora in disuso.

Gli host e i router connessi a Internet comunicano tra loro mediante un insieme di protocolli, a cui ci si riferisce con il termine “*Pila Internet*” (Internet Protocol Suite) e che, limitando la citazione a quelli più conosciuti e importanti, includono il *TCP* (Transmission Contro Protocol) e l’*IP* (Internet Protocol). I protocolli della Pila Internet rappresentano la base comune per lo scambio di informazioni tra le varie sotto-reti e sono spesso indicati, sempre per brevità e quando ciò non crei equivocazione, con la dizione *TCP/IP*, che è anche utilizzata per identificare una rete o una sotto-rete operanti secondo le regole procedurali della Pila Internet. Questa identificazione vale ovviamente per la stessa Internet.

In alcuni o eventualmente in tutti gli host connessi a una sotto-rete possono essere sovrapposti i protocolli della Pila Internet senza modificare i protocolli di trasferimento pre-esistenti. Tali apparecchiature terminali, qui denominate *host IP*, possono scambiare informazioni con qualsiasi altro host IP connesso alla stessa sotto-rete o alle altre sotto-reti interconnesse. Si identificano inoltre con il termine *host non-IP*quelle apparecchiature terminali che, pur dotate dei protocolli necessari per scambiare informazioni con host a loro omogenei e connessi alla stessa sotto-rete, non sono in grado di operare secondo le regole dei protocolli della Pila Internet.

Dal punto di vista dell’indirizzamento, ogni interfaccia fisica tra un apparecchio terminale o un sistema di interconnessione da un lato e una sotto-rete dall’altro è caratterizzata da una *coppia di indirizzi*: uno *locale* che ha significatività nel piano di numerazione della sotto-rete interfacciata, l’altro *globale* che identifica quell’interfaccia in modo univoco nell’ambito di Internet.

Dato che un host ha in generale una interfaccia fisica, esso è identificato da una sola coppia di indirizzi (quelli locale e globale). Invece un router, che possiede due o più interfacce fisiche tante quante sono le sotto-reti che esso pone in corrispondenza, è individuato da tante coppie di indirizzi (quelli locale e globale) quante sono le sue interfacce fisiche. Inoltre, mentre un host non deve decidere verso quale interfaccia emettere l’informazione in lui stesso originata, un router deve necessariamente scegliere verso quale delle sue interfacce fisiche emettere una unità di dati: esso svolge cioè le stesse funzioni logiche che, in altre reti (ed anche *all’interno* di alcune sotto-reti che costituiscono Internet), sono svolte da *commutatori a pacchetto*.

La Fig. I.2.1 mostra in modo esemplificativo la struttura ora descritta. La "nuvola" più grande rappresenta Internet; gli host IP sono rappresentati al di fuori di tale nuvola per evidenziare il fatto che Internet appare ad essi come un’unica rete. Gli utenti che ad essi fanno capo non “vedono” la complessità sottostante e possono ignorare i dettagli e la topologia di rete. In tal modo è più semplice derivare un modello astratto delle comunicazioni e sviluppare le applicazioni di rete. Il software applicativo è indipendente da quello di rete ed ambedue possono essere modificati in modo indipendente. Gli host non-IP sono invece rappresentati all’interno della nuvola in quanto essi possono comunicare *solo* con host connessi alla loro stessa sotto-rete ed a questi omogenei.

Non è necessario che una data sotto-rete sia connessa *direttamente*, tramite un router, a *tutte* le altre sotto-reti. Il traffico proveniente da una sotto-rete e destinato ad un’altra sotto-rete attraversa, in generale, diverse sotto-reti intermedie. I router non consentono, in generale, connessioni *dirette* tra due *qualsivoglia* sotto-reti che compongono Internet. Ogni entità che costituisce Internet contribuisce quindi al trasferimento dell’informazione, realizzando anche funzioni che in altre infrastrutture di telecomunicazioni sono tipiche dei gestori di rete.

Dal punto di vista di Internet, ogni sistema di comunicazione capace di trasferire informazione appare come una singola sotto-rete in modo *non dipendente* dalle sue caratteristiche tecniche e dalle sue prestazioni (protocolli, dimensione delle unità informative, scala geografica, velocità e ritardo di trasferimento, trasparenza temporale, integrità informativa, portata, etc.). La Fig. I.2.1 rappresenta infatti in modo indifferenziato tutte le sotto-reti componenti, poiché i protocolli della Pila Internet le trattano allo stesso modo, nonostante le loro differenze: in altre parole detti protocolli trattano tutte le sotto-reti in modo uguale; una rete in area locale (LAN), una rete in area

metropolitana (MAN), una rete in area geografica (WAN), una connessione punto-punto dedicata, ognuno di questi sistemi di comunicazione è visto da Internet come una singola sotto-rete.

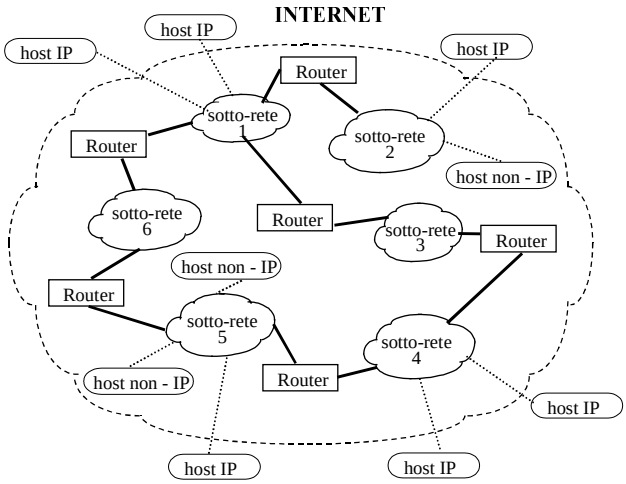


Figura I.2.1- Struttura esemplificativa di Internet

I.2.2. Architettura protocollare

L'architettura protocollare di Internet è rappresentata nella Fig. I.2.2. La numerazione nella prima colonna fa riferimento ai *ranghi degli strati nel modello OSI*. A questi ranghi sono posti in corrispondenza, nella seconda colonna, i protocolli di trasferimento e di utilizzazione del modello Internet; in campo grigio sono anche indicati varii protocolli che caratterizzano l'operatività di alcune sotto-reti che possono essere di supporto alla Pila Internet. Nel seguito le unità informative di ogni protocollo della Pila, in accordo con la nomenclatura propria delle architetture a strati, sono indicate con la sigla *PDU* (Protocol Data Unit) preceduta dalla sigla del protocollo (ad esempio, *IP-PDU* per indicare la PDU di IP).

Lo strato di rango 3 OSI è suddiviso in tre sotto-strati. Nel *sottostrato 3a,* detto di "*accesso alla sotto-rete*", sono incluse le funzioni dello strato di rete che riguardano una specifica sotto-rete. Al *sottostrato 3b,* detto di "*potenziamento della sotto-rete*", sono attribuite le funzioni necessarie per armonizzare le sotto-reti che offrono servizi diversi: un punto chiave è l'armonizzazione dei piani di numerazione relativi ad ambienti di comunicazione tra loro differenti (ad esempio, alla rete telefonica, a una LAN o a una rete dedicata per dati). Infine il *sotto-strato 3c*, detto di "*inter-rete*", comprende le funzioni che sono tipiche di una infrastruttura di interconnessione (com'è Internet) e che sono esemplificabili con l'instradamento da estremo a estremo.

<table>
<tr><td rowspan="2">Strati</td><td colspan="4">Protocolli</td></tr>
<tr><td colspan="4">Applicazioni</td></tr>
<tr><td>5 - 7</td><td>TELNET
SMTP
FTP
HTTP</td><td>GOPHER
ARCHIE
TALK
X-Window</td><td>NFS
RIP
TFTP
SNMP</td><td></td></tr>
<tr><td>4</td><td colspan="2">TCP</td><td>UDP</td><td rowspan="2">ICMP</td></tr>
<tr><td>3c</td><td colspan="3">IP</td></tr>
<tr><td>3b</td><td colspan="2">ARP/RARP</td><td colspan="2"></td></tr>
<tr><td>3a</td><td colspan="4">X.25 liv. 3, SNA, DECnet, ATM+AAL, PPP, LLC, etc</td></tr>
<tr><td>2</td><td colspan="4">X.25 liv. 2, 802.2, 802.3, 802.4, Ethernet etc.</td></tr>
<tr><td>1</td><td colspan="4">Strato fisico</td></tr>
</table>

Figura I.2.2 - Architettura protocollare di Internet

Ai ranghi superiori al 4 corrispondono in modo indifferenziato i protocolli associati ai servizi applicativi che sono stati descritti in § I.1.3. Questi servizi utilizzano, a loro scelta, i protocolli corrispondenti al rango 4, e cioè TCP e UDP. Mentre quest'ultimo è di supporto a un servizio senza connessione, TCP offre un servizio affidabile orientato alla connessione. Questa affidabilità viene raggiunta senza che TCP imponga vincoli ai protocolli su cui si appoggia: infatti TCP è stato progettato per funzionare anche sfruttando un servizio di strato senza connessione e potenzialmente inaffidabile (come IP). TCP può operare su un ampio spettro di piattaforme comunicative, dalla semplice connessione dedicata a una commutata o semi-permanente con modo di trasferimento a circuito o a pacchetto. Trasferisce un flusso informativo continuo e bi-direzionale. TCP può sopperire a problemi di integrità, di perdita, di duplicazione e di consegna fuori sequenza dei dati. TCP implementa anche un controllo di flusso che consente di adeguare il volume dei dati trasferito alle reali capacità di ricezione e di emissione da parte dei processi TCP coinvolti nelle reti attraversate.

Ai ranghi non superiori al 3a sono in corrispondenza protocolli, detti "*di accesso alla rete*", rappresentati in campo grigio nella Fig. I.2.2 e trattati dalla Pila Internet in modo indifferenziato. Tali protocolli sono quelli tipici di una data sotto-rete. Un host non-IP dispone solo di questi protocolli e di quelli applicativi. Un host non-IPdiviene host IP quando ai protocolli appena menzionati si sovrappongono (in senso logico) quelli della Pila Internet. Questa riesce a inter-connettere tutti i tipi di sotto-rete grazie alla sua operatività: infatti, essa assume solo che ognuna delle sotto-reti interconnesse sia capace di trasferire informazione, senza richiedere particolari prestazioni; quindi implementa tutte le funzioni tipiche degli strati di rango 2, 3 e 4: controllo di errore, indirizzamento, instradamento, segmentazione/assemblaggio delle proprie unità informative. Se alcune o tutte queste funzioni non erano state svolte da una particolare sotto-rete, la Pila le realizza; se erano già state svolte le duplica, realizzandole nuovamente; ciò conduce ad eventuali inefficienze, ma consente di non imporre alcun vincolo sulla tecnologia e sui protocolli delle sotto-reti che interconnette. Come ovvia conseguenza la velocità di trasferimento delle informazioni, i gradi di trasparenza temporale e di integrità informativa, nonché tutte le altre prestazioni di rete sono fortemente legate alla tecnologia ed alla tipologia delle sotto-reti su cui la Pila Internet si appoggia.

IP tratta ciascuna IP-PDU come un messaggio indipendente da tutti gli altri; non esistono pertanto, in questo strato, i concetti di connessione e di circuito logico; IP è senza connessione. Il trasferimento delle IP-PDU può richiedere una segmentazione delle stesse laddove le dimensioni delle unità informative gestite dalle sotto-reti siano inferiori alle dimensioni massime consentite alle IP-PDU. A tale scopo IP fornisce un meccanismo specifico per segmentare/assemblare le proprie IP-PDU.

Si noti infine che una data sotto-rete componente Internet può adottare IP e TCP come protocolli *propri,* in corrispondenza con i ranghi 3 e 4, rispettivamente. In tal caso gli host ad essa appartenenti sono già host IP e quindi possono colloquiare con altri host IP connessi ad Internet, una volta che la sotto-rete in esame sia interconnessa ad altre sotto-reti in accordo ai principi sopra espressi.

Nel seguito del paragrafo si esporranno brevemente le caratteristiche principali dei protocolli usati negli strati di trasferimento dell'architettura Internet. Nella collocazione logica di questi protocolli, si fa riferimento alla prima colonna di Fig. I.2.2.

I.2.3. La Pila Internet

ARP e *RARP* sono protocolli di supporto collocabili in corrispondenza con il rango 3b. ARP (Address Resolution Protocol) è usato per determinare quale indirizzo locale (che identifica un sistema nell'ambito della sotto-rete di appartenenza) corrisponde ad un dato indirizzo globale (chiamato anche *indirizzo IP*). RARP (Reverse Address Resolution Protocol) esegue l'operazione inversa (Fig. I.2.3). Le specifiche modalità di funzionamento di ARP e RARP dipendono dalla

particolare sotto-rete in cui operano. In altri termini, ogni tipo di sotto-rete usa particolari e diversi protocolli ARP e RARP. Il protocollo RARP è utilizzato da un host o un router privo di dispositivi di memorizzazione di massa, per determinare, durante la fase di inizializzazione, il *proprio* indirizzo IP a partire dal suo indirizzo locale; in tal caso l'indirizzo locale sarà fisicamente scritto, in hardware, nella sua interfaccia di rete. RARP assume che nella sotto-rete siano presenti uno o più "RARP-server" a cui inviare la richiesta RARP. Il "RARP-server", una volta ricevuta la richiesta, risponde inviando l'indirizzo IP cercato.

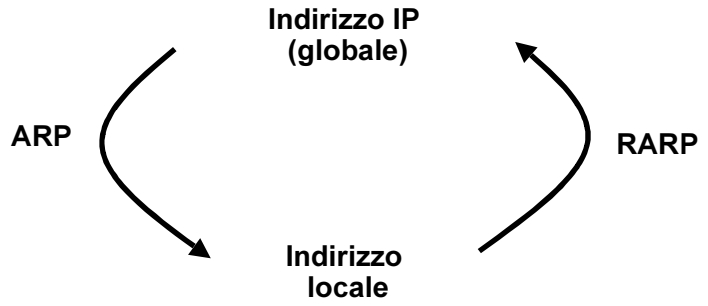


Figura I.2.3 – Traduzione di indirizzi

IP (Internet Protocol) è il principale protocollo di Internet e sarà descritto in dettaglio nel par.I.3. E' logicamente collocabile in corrispondenza con il rango 3c, cioè al di sopra degli altri sottostrati in cui è suddivisibile lo strato 3. Le sue funzioni possono riassumersi nell'indirizzamento di uno specifico host, nell'instradamento, nella segmentazione/assemblaggio delle IP-PDU. In ricezione, lo strato IP consegna le IP-PDU al protocollo di strato superiore a cui esse sono destinate (ad es. a TCP, a UDP o a ICMP).

ICMP (Internet Control Message Protocol) è un protocollo senza connessione utilizzato per risolvere eventuali situazioni anomale, comunicando ai sistemi coinvolti i problemi riscontrati; effettua inoltre semplici funzioni di controllo di flusso. I messaggi di ICMP sono trasportati all'interno delle IP-PDU: da questo primo punto di vista, ICMP è logicamente situato *al di sopra* di IP. In caso di malfunzionamento della rete, ICMP provvede ad uno scambio di messaggi fra i sistemi coinvolti per notificare l'errore o per indicare le circostanze inaspettate che causano il comportamento anomalo. Il destinatario di un messaggio ICMP non è un programma applicativo od un utente, ma è il processo che implementa IP. In altre parole ICMP fornisce i mezzi per uno scambio di informazione tra un'entità IP di un sistema e la corrispondente entità IP di un altro sistema. Da questo secondo punto di vista, ICMP può essere considerato funzionalmente situato *a fianco di IP.* L'insieme dei due punti di vista giustifica la posizione di ICMP nella architettura protocollare rappresentata in Fig. I.2.2, ove ICMP è posto a fianco e al disopra di IP. La funzione di ICMP è solo di *notifica* degli errori al sistema di origine; ICMP non specifica le azioni che devono essere prese per rimediare agli errori ed ai malfunzionamenti; sarà poi il sistema di origine a porre in relazione il particolare errore con il relativo programma applicativo ed a decidere cosa fare per correggere il problema.

TCP (Transmission Control Protocol) sarà descritto nel par. I.5. È un protocollo collocabile in corrispondenza con il rango 4. Offre un servizio con connessione. Svolge controllo di errore con rivelazione e recupero, controllo di flusso, ri-ordinamento dell'informazione trasferita tra entità applicative e indirizzamento di uno specifico processo all'interno di un host. Un host può essere utilizzato da più processi. Dato che IP indirizza un host in modo globale, è necessario anche precisare con quale specifico processo residente in un determinato host si vuole comunicare. Di ciò si occupa TCP.

UDP (User Datagram Protocol) sarà descritto in dettaglio nel par. I.4; è un protocollo collocabile in corrispondenza dello stesso rango di TCP. E' però molto più semplice di TCP ed è senza connessione. Offre agli strati superiori un trasferimento non affidabile; la sua funzione

principale è l'indirizzamento di uno specifico utente all'interno di un host. Opzionalmente fornisce un meccanismo per il controllo di errore, limitato alla rivelazione di errori senza quindi richieste di ritramissione. È utilizzato prevalentemente da TFTP (cfr. § I.1.3.3) e RIP (cfr. § I.3.5.4).

I.3. Il protocollo IP

Il protocollo IP opera nello strato omonimo del modello Internet. Questo strato è *di modo di trasferimento* per il paradigma Internet e svolge le seguenti funzioni:

- definisce lo *schema di indirizzamento*; gli elementi di questo schema, corrispondenti a una numerazione globale, sono gli *indirizzi IP*;
- definisce il formato della *IP-PDU*, che è chiamata *datagramma* (datagram) e che è l'unità- base per il trasferimento dell'informazione attraverso Internet;
- definisce le modalità per *segmentare/assemblare* le IP-PDU; il risultato della segmentazione, solitamente chiamato *frammento* (fragment), ha lo stesso formato della IP-PDU assoggettata a segmentazione;
- svolge la funzione di *instradamento*, e cioè definisce il percorso che una IP-PDU o un suo frammento seguono per arrivare a destinazione;
- specifica la regola secondo cui host e router debbono *processare le* IP-PDU o i loro frammenti; ad esempio precisa come e quando occorre generare messaggi di errore, oltre che le condizioni che rendono necessario scartare un IP-PDU o un suo frammento.

Il servizio offerto dallo strato IP allo strato superiore è il *servizio di rete* nel modello Internet; ha le seguenti caratteristiche distintive

- ♦ tratta in modo indistinto una IP-PDU o un suo frammento; si può quindi parlare di IP-PDU con riferimento a una unità informativa di IP senza specificare se è o meno il risultato di una segmentazione;
- ♦ comprende, come principale elemento, il *trasferimento* delle IP-PDU tra le entità dello strato superiore (cioè tra entità TCP o UDP);
- ♦ effettua questo trasferimento nel modo *senza connessione*;
- ♦ è quindi inaffidabile e basato sul paradigma del "*best-effort*" (cioè cerca di "fare del suo meglio").

Il servizio è definito inaffidabile perché la consegna di una IP-PDU non è garantita. La IP-PDU può essere persa, duplicata o consegnata fuori sequenza. Il servizio non notifica queste condizioni né al mittente né al destinatario. Il servizio è senza connessione e quindi ogni IP-PDU è trattata in modo indipendente dalle altre e ognuna di esse può seguire una strada diversa per arrivare a destinazione. Infine il servizio è detto "best-effort" perché fa del suo meglio nel recapitare le IP-PDU, ma, nel caso di malfunzionamenti, di errori o di congestione, la qualità di servizio offerta può degradarsi anche fortemente. In queste ultime situazioni IP sfrutta le funzioni di ICMP per *cercare* di contenere la degradazione.

Il seguito del paragrafo è diviso in otto parti. Nelle prime due (§ I.3.1 e I.3.2) si descrivono il formato della IP-PDU e le modalità con cui la si segmenta o la si assembla; la terza (§ I.3.3) introduce lo schema di indirizzamento; la quarta (§ I.3.4) spiega come gli indirizzi IP sono posti in corrispondenza con gli indirizzi locali. La quinta parte (§ I.3.5) definisce le procedure per l'instradamento; la sesta (§ I.3.6) si occupa di ICMP. Nella settima parte (§ I.3.7) si introduce il *Domain Name System* (DNS), ovvero un meccanismo che consente di associare ad ogni indirizzo IP un *nome*, più significativo per gli utenti e più semplice da ricordare e da utilizzare. Infine, nell'ultima parte (§ I.3.8), si presentano esempi che illustrano il funzionamento di alcune operazioni tipiche svolte durante uno scambio informativo in Internet.

I.3.1. Formato della IP-PDU

La IP-PDU è composta da una *intestazione* (header) e da un *campo informativo* che contiene i dati di utente (data). Il formato della IP-PDU è illustrato nella Fig. I.3.1, ove ogni riga contiene 32 bit.

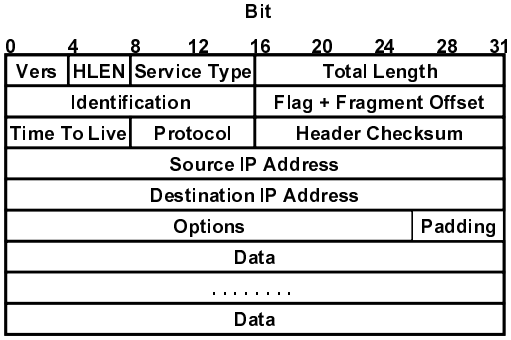


Figura I.3.1 - Formato della IP-PDU

I campi definiti in Fig.I.3.1 hanno significati e funzioni che sono precisati qui di seguito.

- *Vers* (Version): è la *versione del protocollo* usata; grazie a tale campo (di lunghezza 4 bit) è possibile che più versioni di IP operino contemporaneamente; in tal modo si rendono più semplici la modifica ed i miglioramenti del protocollo; attualmente si utilizza la versione 4 (*IPv4*), ma è previsto il passaggio alla versione 6 (*IPv6*).
- *HLEN* (Header Length): è la *lunghezza dell'intestazione* specificata con il numero di *parole di 32 bit* (cioè di righe secondo la rappresentazione di Fig. I.3.1); dato che questo campo ha la lunghezza di 4 bit, l'intestazione ha una lunghezza massima di 16 righe; la lunghezza dell'intestazione in assenza del campo "opzioni" è di 5 righe (corrispondenti a 20 ottetti).
- *Service type*: è il *tipo di servizio* e specifica parametri della qualità di servizio richiesti dall'utente: affidabilità, velocità di trasferimento; nella definizione originale, un sub-campo (di tre bit) con valori da 0 a 7 indicava l'importanza della IP-PDU; un sub-campo successivo era composto da tre bit che potevano essere utilizzati per richiedere particolari caratteristiche: *D-bit* (basso ritardo, "Delay"), *T-bit* (alta portata, "Throughput"), *R-bit* (alta affidabilità, "Reliability"); i restanti due bit erano riservati per usi futuri. In seguito il formato di tale campo è stato modificato: uno dei due bit riservati è stato aggiunto al secondo sub-campo e i quattro bit seguenti i primi tre (i quali continuano ad identificare l'importanza di una IP-PDU in scale di priorità); sono utilizzabili con i seguenti significati:
  - 1000 minimizzare il ritardo (delay);
  - 0100 massimizzare la portata (throughput);
  - 0010 massimizzare l'affidabilità (reliability);
  - 0001 minimizzare il costo;
  - 0000 servizio normale.

  Di fatto l'uso di tale campo è attualmente opzionale e risulta raramente tenuto in conto nelle implementazioni, in favore di una maggiore semplicità. Alcune recenti proposte ne rilanciano l'uso, proponendo di usarlo in modo obbligatorio per lo meno in tutti i router appartenenti ad una sezione di Internet che intenda offrire nuove funzionalità in termini di qualità di servizio.
- *Total length*: specifica con 16 bit la *lunghezza totale* della IP-PDU, misurata in *ottetti*, includendo l'intestazione e il campo informativo; la lunghezza di una IP-PDU è sempre uguale ad un multiplo di 4 ottetti (cioè di 32 bit) e può raggiungere un valore massimo di 2^16=65.536 ottetti.
- *Identification*: è il *numero della IP-PDU* (identificazione), specificato con 16 bit; è un valore identificativo assegnato dal processo -sorgente alla IP-PDU o ai suoi frammenti. Questo valore è fornito da un contatore dell'host sorgente che viene incrementato ogniqualvolta l'host genera una

IP-PDU. Ogni router che segmenta una IP-PDU ricopia questo campo nell'intestazione di ogni frammento della IP-PDU di partenza;

- *Flags*: sono le *bandiere* costituite dai 3 bit *X*, *DF* e *MF*, tra i quali:
  - *X* non è usato ed è posto al valore logico 0;
  - *DF* (Don't Fragment), se uguale a 0, indica che la IP-PDU può essere segmentata; se 1, no;
  - *MF* (More Fragment), se uguale a 0, indica che è l'ultimo frammento di una IP-PDU; se 1, che seguono altri frammenti.
- *Fragment Offset*: indica, in unità di 8 ottetti, la *posizione di un frammento* all'interno della IP-PDU originaria; l'unità utilizzata implica che la lunghezza del campo-dati di un frammento, con l'eccezione dell'ultimo, è sempre uguale a un multiplo di 8 ottetti; inoltre, dato che nel campo Fragment Offset sono disponibili 13 bit , è possibile numerare fino a un massimo di 8.192 frammenti; si nota che questo ultimo numero unito alla lunghezza minima del campo-dati di un frammento (1 ottetto) è coerente con la lunghezza massima di una IP-PDU, che, lo si ricorda, è uguale a 65.536 ottetti;
- *Time to Live*: indica con 8 bit quanto tempo la IP-PDU può rimanere all'interno della inter-rete (*tempo di vita*). Quando un host genera una IP-PDU, inizializza questo campo con il valore del tempo concesso alla IP-PDU stessa per attraversare l'inter-rete. Questo valore viene decrementato da ogni router incontrato nel cammino percorso dalla IP-PDU. Quando il valore di questo campo diventa nullo, la relativa IP-PDU viene scartata. Ciò impedisce ad una IP-PDU di viaggiare indefinitamente nella rete, (se, a causa di errori, il suo instradamento risultasse in un cammino chiuso) o, in ogni caso, di evitare tempi di trasferimento eccessivamente lunghi. Il "Time to Live" (nel seguito indicato con la sigla *TTL*) è decrementato a passi minimi di un secondo; dopo 256 secondi (2^8) la IP-PDU viene scartata. Nelle implementazioni attuali, e sempre in favore di una maggiore semplicità, il valore di questo campo non è definito in secondi ma in "salti" (*hops*). Per salto si intende l'attraversamento di un router e quindi una IP-PDU può attraversare al massimo 256 routers prima di essere scartata.
- *Protocol*: indica con 8 bit a quale *protocollo* dello stato superiore deve essere trasferito il contenuto informativo della IP-PDU.
- *Header Checksum*: contiene la ridondanza (somma di controllo) per controllare gli errori *sulla sola intestazione*; l'obiettivo è evitare che una IP-PDU vada verso siti diversi da quelli voluti. Il contenuto di questo campo, comprendente 16 bit, è ottenuto considerando i bit dell'intestazione a gruppi di 16 alla volta; l'entità emittente ne effettua la somma e memorizza in questo campo il complemento a 1 del risultato. L'entità ricevente effettua a sua volta la stessa somma, includendo però come addendo la parola di 16 bit contenuta nel campo in esame. Se, dopo aver effettuato il complemento a 1, il risultato è costituito da una parola di tutti 1, l'entità ricevente conclude che non si sono verificate variazioni nell'intestazione; in caso contrario (parola che non è costituita da tutti 1), la IP-PDU viene scartata senza alcun ulteriore provvedimento.
- *Source Address*: con 32 bit precisa l'*indirizzo IP di origine* (dell'host, non del processo-sorgente).
- *Destination Address*: con 32 bit precisa l'*indirizzo IP di destinazione* (dell'host, non del processo-collettore).
- *Options*: è un campo di lunghezza variabile (multipli di 8 bit) che include le possibili *opzioni* a scelta dell'utente. Il campo può essere omesso. La lunghezza è funzione delle opzioni implementate. Ad esempio:
  - *Record Route Option* (RRO): consente al mittente di creare una lista vuota di indirizzi IP in modo che ogni host/router attraversato inserisca il suo indirizzo in questa lista;
  - *Source Route Option*: consente al mittente di specificare gli host/router attraverso i quali vuole che transiti la IP-PDU; in tal modo un mittente può far percorrere alla IP-PDU una particolare strada. E' usata per scopi gestionali e di verifica del funzionamento di una determinata parte di Internet.

- *Timestamp Option*: ha la stessa funzione di RRO, ma in aggiunta ogni host/router attraversato comunica, oltre al proprio indirizzo, anche l'istante temporale in cui la IP-PDU lo attraversa.

- *Padding*: è un riempitivo che, mediante introduzione di uno o più 0, rende la lunghezza dell'intestazione un multiplo intero di 32 bit.

È significativo sottolineare che, sulla base della lunghezza dei campi componenti, la lunghezza dell'intestazione di una IP-PDU in assenza del campo Options è di 5 righe, corrispondenti a *20 ottetti*. Questa è anche la *lunghezza minima* dell'intestazione.

I.3.2. Segmentazione e assemblaggio

Le sotto-reti componenti Internet possono avere diverse limitazioni circa la lunghezza massima delle loro unità di dati; ad esempio la lunghezza massima di un'unità di dati in una LAN Ethernet è di 1500 ottetti, mentre nella MAN FDDI è di 4470 ottetti. La dimensione massima dell'unità di dati di una sotto-rete è denominata, in questo contesto, *Maximum Transfer Unit* (MTU). Dovendo scegliere la dimensione di una IP-PDU, una possibile soluzione potrebbe essere quella di adottare un valore uguale al minimo delle MTU delle sotto-reti da attraversare. Ciò richiederebbe però uno scambio di informazioni di controllo per determinare tale valore minimo e causerebbe inefficienze nel trasporto attraverso sotto-reti con dimensioni di MTU maggiori del valore minimo. Come per altre problematiche, si è scelta invece una soluzione che sia la più semplice possibile e che non sia legata a particolari tecnologie delle sotto-reti componenti Internet.

Ogni host che emette una IP-PDU può scegliere per questa qualsivoglia dimensione, purché inferiore alla lunghezza massima di una IP-PDU e non inferiore a quella relativa alla sola intestazione. Tipicamente la dimensione di una IP-PDU viene scelta uguale alla MTU della sotto-rete a cui è connesso l'host/router emittente. Questa MTU è resa nota all'entità IP mittente dal software (*driver*) che interfaccia l'host/router stesso ad una data sotto-rete. Ovviamente, se la stringa di dati da emettere ha una lunghezza inferiore alla MTU prescelta, si attribuisce alla IP-PDU una dimensione commisurata alla lunghezza della stringa.

Quando poi una IP-PDU deve attraversare diverse sotto-reti per giungere a destinazione ed almeno una di queste sotto-reti ha

- una MTU di dimensioni inferiori a quelle scelte per una data IP-PDU; ovvero
- una dimensione inferiore a quella della MTU della sotto-rete di origine,

la IP-PDU, come già detto, viene segmentata.

I frammenti non devono essere necessariamente tutti della stessa dimensione, anche perché nulla assicura che una IP-PDU abbia una lunghezza che sia un multiplo intero di una data MTU; almeno l'ultimo frammento di una data IP-PDU può avere una dimensione diversa dagli altri frammenti della stessa IP-PDU. Può accadere anche che un frammento venga a sua volta frammentato, qualora debba attraversare sotto-reti con MTU di dimensioni ancora inferiori.

Il solo vincolo che IP pone ai sistemi connessi ad Internet è che

- i routers debbano accettare IP-PDU di dimensioni uguali a quelle delle MTU delle sotto-reti a cui sono connessi;
- tutti gli host o i router debbano comunque accettare e gestire IP-PDU di dimensioni almeno uguali a 576 ottetti.

La segmentazione di una IP-PDU si rende necessaria quando anche solo una delle sotto-reti attraversate ha una MTU inferiore alla lunghezza della IP-PDU. Alcune IP-PDU potrebbero però essere contrassegnate come *non segmentabili* (bit DF posto ad uno) e, in questo caso, non vengono suddivise in unità più piccole per attraversare la inter-rete. Ciò comporta la perdita della IP-PDU. In tal caso viene generato un messaggio ICMP.

Le procedure di segmentazione e di assemblaggio devono essere in grado di segmentare la IP-PDU originaria in un numero arbitrario di unità che, giunte a destinazione, devono poter essere ricomposte nella forma originaria. Il destinatario utilizza il campo "Identification" di ogni

frammento per garantire che IP-PDU generate da processi diversi non siano confuse tra loro. Tale campo, che è univoco per tutti i processi operanti in quel momento tra entità agenti come sorgenti e collettori, è attribuito dalla entità sorgente.

Ad ogni frammento è inoltre assegnato il campo “Fragment Offset” che permette al destinatario di risalire alla posizione occupata dal frammento nella IP-PDU *originaria.* Il frammento con il Flag MF posizionato a zero è identificabile come l'ultimo della IP-PDU originaria.

L'informazione presente nell'intestazione della IP-PDU viene copiata nell'intestazione di ognuno dei suoi frammenti, ad eccezione del campo “Total Length” che viene modificato per fornire la lunghezza di uno specifico frammento. In tal modo ogni frammento diventa a sua volta una IP-PDU e può essere quindi ulteriormente segmentato.

La IP-PDU originale viene ricostruita completamente solo a destinazione. Questa riassembla i frammenti caratterizzati dagli stessi valori dei campi “Identification”, “Source IP Address”, “Destination IP Address” e “Protocol”. Nella IP-PDU ricostruita la parte dati di ogni frammento è inserita nella posizione indicata dal campo “Fragment Offset”.

Se uno o più frammenti di una IP-PDU vengono persi, i restanti che arrivano a destinazione vengono scartati. Ciò avviene dopo un certo tempo pre-definito (*time-out*), necessario per evitare che si scartino frammenti non perché persi ma perché in ritardo. Al riguardo si ricorda che IP è senza connessione e quindi non conserva la sequenza originaria di emissione. Se si utilizza TCP come protocollo di trasporto, quest’ultimo chiede al mittente di una IP-PDU scartata di ri-inviarla. Questo modo di procedere porta a eventuali inefficienze, ma l'alternativa sarebbe quella di effettuare un controllo di errore in ogni router, rendendo così più complessi tali sistemi.

I.3.3. Schema di indirizzamento

Per consentire a tutti gli host o i router connessi ad Internet (nel seguito indicati, per brevità, come “*sistemi*”) di comunicare tra loro, è necessario stabilire un metodo globalmente accettato per identificare ed indirizzare in modo *univoco* tutti i sistemi.

Lo schema di indirizzamento di Internet è definito all’interno degli strati IP e TCP (o UDP). Un indirizzo IP identifica *solo un sistema* e non uno specifico processo. L’identificazione di un processo all’interno di un sistema è affidata ai protocolli di strato superiore (TCP o UDP). Un indirizzo completo è quindi costituito da *due parti*, una definita all’interno di IP ed un’altra definita all’interno di TCP (o UDP). In questo paragrafo tratteremo solo *gli indirizzi IP*; si rimanda invece ai parr. I.4 e I.5 per quanto riguarda l’indirizzamento di uno specifico processo all’interno di un sistema.

Lo schema di indirizzamento IP si sovrappone a quello delle sotto-reti che interconnette. Gli indirizzi IP devono essere unici in tutta la inter-rete. E’ possibile attribuire indirizzi arbitrari ad una data rete TCP/IP solo se questa non è connessa con altre reti o con Internet.

Un indirizzo IP è caratterizzato da una *struttura gerarchica*, per facilitare l’operazione di instradamento. Si possono però distinguere una gerarchia *a due livelli* e una a *tre livelli.*

I.3.3.1. Gerarchia a due livelli

Un indirizzo IP identifica prima una *porzione* di inter-rete a cui un sistema è connesso e poi il sistema all’interno di quella porzione; ciò facilità la scelta di una strada per raggiungere un dato sistema.

Un indirizzo IP è costituito da una stringa di 32 bit; possono quindi teoricamente esistere 2^32 (=4.294.967.296) possibili indirizzi; ogni indirizzo IP consta di due parti: *Net_Id* e *Host_Id.* L’indirizzo completo può quindi essere scritto come:

$$IP\_Address = Net\_Id.Host\_Id\ .$$

I 32 bit totali sono divisi tra Net_Id e Host_Id.

E’ importante sottolineare che la componente Net_Id di un indirizzo IP *non* è necessariamente in corrispondenza con una data sotto-rete fisica; lo è invece con quella che si è prima definita genericamente come una *porzione* dell’inter-rete. Tale porzione può coincidere con una data sotto-

rete fisica, ma può anche comprendere più sotto-reti fisiche o essere un sotto-insieme di una data sotto-rete fisica. Nel seguito la porzione di inter-rete, che è caratterizzata da una comune componente Net_Id degli indirizzi IP in essa contenuti viene chiamata "*rete logica*".

Per ciò che riguarda poi la componente Host_Id, questa, almeno in una gerarchia a due livelli, è in corrispondenza con l'interfaccia di un qualunque sistema (host o router) facente capo a Internet nell'ambito della rete logica caratterizzata dalla componente Net_Id. Quindi un host, che è connesso a Internet tramite un'unica interfaccia, può essere individuato con un singolo indirizzo IP; invece un router, che ha tante interfacce quante sono le reti logiche interconnesse, possiede almeno un indirizzo IP per ciascuna rete logica verso cui si interfaccia.

La suddivisione di un indirizzo IP nelle sue due componenti non è fissa. In particolare sono state definite *cinque classi*, in ognuna delle quali si assegna una frazione diversa dei 32 bit totali alla Net_Id e conseguentemente alla Host_Id.

La possibile suddivisione di un indirizzo IP è riportata in Tab. I.3.1, per ognuna delle cinque classi. Il motivo dell'introduzione di queste classi è ancora da ricercarsi nel fatto che Internet si propone di inter-connettere diversi tipi di sotto-rete; si possono perciò avere diverse necessità, circa la definizione delle reti logiche. Le cinque classi sono contraddistinte da una lettera, da A a E; le prime tre, ovvero le classi A, B e C, sono destinate all'utenza normale; la classe D è destinata a comunicazioni di tipo multipunto ("multicast"); la classe E è riservata per usi futuri e per ricerca/sviluppo. Grazie alla distinzione in classi, possono esistere:

- ✓ per la *classe A*, un numero ristretto di reti logiche (=128), ognuna delle quali può contenere milioni di possibili indirizzi diversi (fino a 16.777.216); il numero totale di indirizzi di classe A è (2^32)/2;
- ✓ per la *classe B*, un numero intermedio di reti logiche (=16.384), ognuna delle quali può contenere diverse migliaia di possibili indirizzi diversi (fino a 65.536); il numero totale di indirizzi di classe B è (2^32)/4;
- ✓ per la *classe C,* un elevato numero di reti logiche (=2.097.152), ognuna delle quali può contenere solo un ristretto numero di possibili indirizzi diversi (fino a 256); il numero totale di indirizzi di classe C è (2^32)/8.

Classe	Bit Iniziali	Net_Id	Host_Id	"reti" disponibili	host disponibili
A	0	7 bit	24 bit	128	16.777.216
B	10	14 bit	16 bit	16.384	65.536
C	110	21 bit	8 bit	2.097.152	256
D	1110	indirizzo multi-punto, 28 bit numero di indirizzi possibili: 268.435.456			
E	11110	riservata per usi futuri e ricerca, 27 bit numero di indirizzi possibili: 134.217.728			

Tabella I.3.1 - Formato degli indirizzi

Come già detto, un indirizzo IP di *classe D* è invece utilizzato per *comunicazioni di tipo multipunto*. Se si vuole inviare un messaggio ad un predefinito gruppo di destinatari, si può usare un apposito indirizzo di tipo multipunto, che corrisponde ad una pluralità di sistemi, denominata "*gruppo multicast*". A tal fine si usa anche un protocollo aggiuntivo (*Internet Group Management Protocol*, IGMP). Si noti che una comunicazione di tipo multipunto è diversa da una di tipo diffusiva ("broadcast"); la prima riguarda uno specifico gruppo di destinatari, mentre la seconda fa riferimento a *tutti* gli utenti all'interno di una specificata sotto-rete (o perfino a tutti gli utenti di Internet). Il numero totale di indirizzi di classe D è (2^32)/16. Infine il numero totale di indirizzi di classe E è (2^32)/32.

A causa della divisione in classi, il numero *totale N* di possibili indirizzi IP, di tutte e cinque le classi, è uguale a

$$N = \frac{2^{32}}{2} + \frac{2^{32}}{4} + \frac{2^{32}}{8} + \frac{2^{32}}{16} + \frac{2^{32}}{32} = \frac{31}{32} 2^{32}$$

invece di *N*=2^32; infatti (2^32)/32 indirizzi non risultano assegnati ad alcuna classe e quindi, ad oggi, non sono utilizzabili.

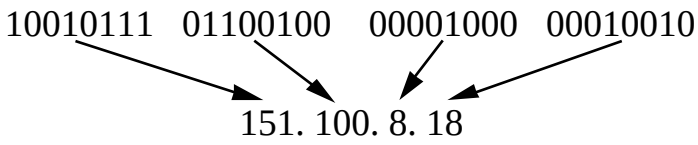
Al fine di assicurare che ogni indirizzo IP sia unico è stata costituita un'autorità centrale, l'"INTERnet Network Information Center" (INTER-NIC), con il compito di assegnare gli indirizzi. L'INTER-NIC è controllato a sua volta da una autorità amministrativa, l'"Internet Assigned Number Authority (IANA). L'INTER-NIC assegna però solo la parte Net_Id, delegando poi l'organizzazione richiedente ad assegnare la restante parte dell'indirizzo, l'Host_Id. Ovviamente una rete logica di classe A è concessa in uso solo ad organizzazioni molto grandi (o ad intere nazioni). Una rete logica di classe B è concessa solo ad organizzazioni che dimostrino di aver bisogno di connettere ad Internet almeno qualche migliaia di host e così via.

L'organizzazione che si vede assegnare una Net_Id può suddividere la parte di Host_Id per creare, all'interno della "sua" rete logica, *sotto-reti logiche*, ognuna delle quali ha la stessa Net_Id. In tal modo, un utente che ha bisogno di un singolo indirizzo IP non deve rivolgersi direttamente all'INTER-NIC, ma può fare riferimento ad un'autorità locale. La scelta di uno schema di indirizzamento gerarchico può dare luogo a indirizzi non utilizzati e quindi a sprechi. Gli indirizzi corrispondenti alla parte Host_Id di una data rete logica, non usati dall'organizzazione responsabile della corrispondente Net_Id, non possono essere usati da nessun altro; ciò spiega l'attuale relativa penuria di indirizzi IP, a fronte dei 3.758.096.384 indirizzi teoricamente disponibili per le prime tre classi.

In particolare, sono le reti logiche di classe B quelle con maggiore rischio di esaurimento. Nel tentativo di alleviare questo problema, nel 1993 si è deciso di consentire una certa violazione della suddivisione in classi di cui sopra, nota come "*supernet addressing*". Questa metodica consente di attribuire più reti logiche ad una stessa organizzazione. In tal modo una data organizzazione, invece di chiedere una rete logica di classe B e non utilizzare tutti i relativi indirizzi disponibili, può chiedere un certo numero di reti logiche di classe C, fino a soddisfare le sue esigenze. Affinché però questo insieme di reti logiche appaia all'esterno come un'unica rete logica, è stato necessario introdurre alcune modifiche a procedure precedentemente utilizzate (cfr. § I.3.5.3).

Nonostante questa tecnica (ed altre, introdotte allo stesso fine, come "Transparent Routers" e "Proxy ARP"), lo spazio degli indirizzi appare non essere più adeguato alle esigenze di una utenza in continua crescita. Per questa ragione, ed anche per migliorare altri punti deboli dell'attuale versione di IP (IPv4), è stata proposta una nuova versione di IP (IPv6). In IPv6 il singolo indirizzo è una stringa binaria di 128 bit (invece di 32); lo spazio degli indirizzi ne risulta significativamente esteso.

Come detto, un indirizzo IP è costituito da una stringa di 32 bit; per le comunicazioni tra persone, in documenti tecnici ed anche in alcuni programmi applicativi si preferisce però usare un'altra notazione, che si ottiene separando i 32 bit in 4 campi di 8 bit ciascuno. Questi campi si esprimono poi in decimale invece che in binario, separandoli con un punto; questa notazione è nota come "*dotted*" ("puntata") o decimale. Ad esempio l'indirizzo IP di un particolare sistema, espresso in una stringa di 32 bit e nella relativa notazione "dotted" è:



È utile sottolineare che l'indirizzo *effettivamente* utilizzato da IP è sempre una stringa di 32 bit e la rappresentazione "*dotted*" deve essere preventivamente tradotta da un opportuno software (molto semplice, in questo caso) prima di essere utilizzata da IP per l'effettivo scambio di informazione.

In aggiunta alle classi definite nella Tab. I.3.1, sono stati definiti alcuni *indirizzi speciali*. Con riferimento alla notazione binaria si è stabilito che un campo costituito da "0", (0 nella notazione "dotted") significa "questo" ed un campo costituito da "1", (255 nella notazione dotted) significa "tutti". Infine un indirizzo che inizia con 01111111 (127 nella notazione dotted) è un indirizzo di "rinvio a se stesso" (loopback), cioè il messaggio inviato ritorna al mittente (usato per prove, verifiche e per comunicazioni tra processi che operano nello stesso sistema). I casi possibili e permessi sono rappresentati nella Tab. I.3.2. Ne segue che gli indirizzi definiti nella Tab. I.3.1 hanno limitazioni ai valori che i loro campi possono assumere; in particolare alcuni campi di un generico indirizzo non possono assumere i valori 0, 127 e 255.

SIGNIFICATO	Net_Id	Host_Id
"questo host"	tutti 0	tutti 0
la rete a cui appartiene l'host con indirizzo "Host"	tutti 0	"Host"
tutti gli host	tutti 1	tutti 1
tutti gli host della rete logica "Net"	"Net"	tutti 1
loopback	127 (in decimale) seguito da qualsiasi cosa	

Tabella I.3.2 - Indirizzi speciali

Gli indirizzi con campi uguale ad "1" sono utilizzati per comunicazioni diffusive ("broadcast") ovvero per comunicare con una pluralità indifferenziata di sistemi. Con riferimento alla notazione "dotted" (si ricorda che una stringa di 8 bit uguale a 11111111 equivale a "255" in decimale), inviando un messaggio all'indirizzo 255.255.255.255 si indirizzano *tutti* i sistemi di Internet (il che in generale non è permesso). L'indirizzo 151.100.255.255 si riferisce a tutti i sistemi della rete logica 151.100. Il campo composto da "0" è invece importante quando si vuole comunicare con un host collegato alla stessa rete logica cui si è collegati e non si conosce l'indirizzo della rete logica stessa (è un problema che sorge per alcune applicazioni). Ad esempio, 0.0.0.0 si riferisce all'host che origina la richiesta; 0.0.0.23 si riferisce alla rete logica a cui è connesso l'host 23.

A titolo di esempio esaminiamo l'indirizzamento di un insieme di sistemi in un ambiente (Fig. I.3.2), ove sono considerate tre sotto-reti fisiche, tutte identificate come altrettante reti logiche. La LAN Ethernet corrisponde alla rete logica con Net_Id=128.10 (in notazione binaria: 10000000.00001010, 16 bit complessivi, di classe B); la LAN Token Ring corrisponde alla rete logica con Net_Id=192.5.48 (in notazione binaria: 11000000.00000101.00011000, 24 bit complessivi, di classe C). Si noti anche che si usa indicare una rete logica aggiungendo alla Net_Id degli zeri, fino ad avere la stessa dimensione di un indirizzo IP (sia in binario che in decimale).

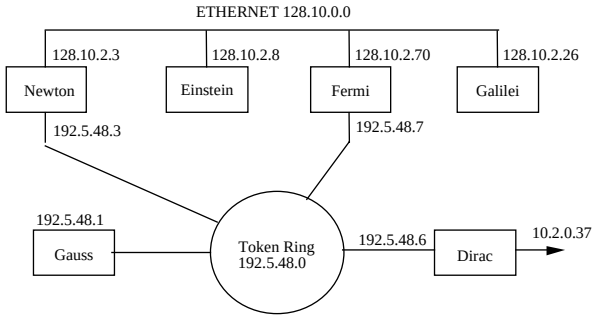


Figura I.3.2 - Esempio di sistemi connessi a più di una sotto-rete

I calcolatori "Newton" e "Fermi" sono router che, sono connessi alla Token Ring e all'Ethernet e hanno quindi due indirizzi IP. Anche "Dirac" è un router, che interconnette la Token Ring verso un'altra sotto-rete fisica e quindi anch'esso ha due indirizzi IP. Gli altri calcolatori hanno un solo indirizzo IP e sono quindi host. Si noti anche che, siccome gli indirizzi sono utilizzati per l'instradamento (cfr. § I.3.5), se ci riferiamo a "Newton" con l'indirizzo 192.5.48.3 il messaggio gli perverrà tramite la Token Ring; se invece usiamo 128.10.2.3, il messaggio gli perverrà tramite l'Ethernet.

I.3.3.2. Gerarchia a tre livelli

Con il progressivo estendersi delle dimensioni di Internet si è deciso di ampliare lo schema di indirizzamento a due livelli introducendo un *ulteriore livello gerarchico*: la parte Host_Id *può* essere a sua volta suddivisa in due parti (denominate nel seguito come *Sub_Net_Id* e *Sub_Host_Id*), creando così "sotto-reti logiche" e portando a tre i livelli gerarchici; un generico indirizzo IP può quindi essere considerato diviso in tre parti:

$$IP\_Address = Net\_Id.Host\_Id = Net\_Id.Sub\_Net\_Id.Sub\_Host\_Id\ .$$

Grazie a tale terzo livello gerarchico (introdotto nei primi anni '80) si possono semplificare le operazioni di instradamento e ridurre gli sprechi dovuti ad indirizzi non utilizzati. Si noti anche che,

in una data rete logica, la componente Host_Id può anche *non* essere suddivisa in due parti; questa situazione può sempre essere vista come un caso particolare della gerarchia a tre livelli: è il caso in cui il campo Sub_Net_Id ha dimensione nulla, mentre rete logica e sotto-rete logica coincidono.

La divisione della parte Host_Id è di competenza dell'amministratore di una data rete logica, che può scegliere qualsiasi tipo di suddivisione: le dimensioni e le configurazioni delle parti Sub_Net_Id e Sub_Host_Id possono essere qualunque, purché, ovviamente, la loro somma sia uguale alla dimensione originaria dell'Host_Id.

Da un punto di vista operativo, occorre rendere noto ad ogni sistema in che modo il suo indirizzo IP è suddiviso, ovvero quali sono le dimensioni delle parti componenti il suo indirizzo. Questa informazione è codificata mediante la cosiddetta "*maschera di sotto-rete*". Questa è una stringa binaria che ha la stessa dimensione di un indirizzo IP e nella quale i campi corrispondenti alle parti Net_Id e Sub_Net_Id sono costituiti da "1" binari, mentre la restante parte, corrispondente alla parte Sub_Host_Id, è costituita da "0".

Ad esempio, in un indirizzo di classe B le parti Net_Id e Host_Id hanno entrambe dimensione uguale a 16 bit. Se la parte Host_Id non è suddivisa, la corrispondente maschera di sotto-rete è:

11111111.11111111.00000000.00000000;

ovvero in notazione decimale:

255.255.000.000.

Se invece la parte Host_Id fosse suddivisa a metà in Sub_Net_Id e Sub_Host_Id, la relativa maschera di sotto-rete sarebbe:

11111111.11111111.11111111.00000000;

ovvero in notazione decimale:

255.255.255.000.

Come si vede esaminando tale maschera, un host può dedurre la modalità di suddivisione del suo indirizzo.

I.3.4. Traduzione tra indirizzi IP e indirizzi locali

Abbiamo visto come ad ogni sistema connesso ad Internet sia assegnato un unico indirizzo IP che lo identifica nell'ambito dell'intera inter-rete (indirizzo globale). Come si vedrà in § I.3.5, quando una IP-PDU arriva alla sotto-rete di destinazione, l'ultimo router attraversato ha il compito di rilanciare questa PDU verso l'host di destinazione: ciò prendendo in considerazione anche la componente Host_Id dell'indirizzo e sfruttando i meccanismi protocollari propri della sotto-rete di destinazione. L'indirizzo IP non è però sufficiente per far arrivare la IP-PDU all'host in questione. Infatti, la sotto-rete di destinazione ha un suo schema di indirizzamento (locale) e l'interfaccia di rete di un host è indirizzata, nell'ambito della sotto-rete di appartenenza, da un indirizzo locale. L'ultimo router deve quindi compiere operazioni di ricerca per determinare l'indirizzo locale di sotto-rete dell'host di destinazione che corrisponde all'indirizzo IP contenuto nella IP-PDU in arrivo.

La traduzione di un indirizzo globale in uno locale interessa però anche gli host e non solo i router; infatti ogni sistema (host o router), quando invia una IP-PDU ad un altro sistema connesso alla stessa sotto-rete, ha bisogno delle funzionalità di traduzione; ciò accade quando:

- un host invia una IP-PDU direttamente ad un altro host, senza passare attraverso un router: cosa possibile quando sia l'host di origine che quello di destinazione sono connessi alla stessa sotto-rete;
- un host invia una IP-PDU ad un router affinché questi lo rilanci poi verso la destinazione finale: in tal caso sicuramente l'host di origine ed il router appartengono alla stessa sotto-rete (cfr. § I.3.5).

Il problema di tradurre indirizzi IP in indirizzi locali è risolto mediante protocolli di tipo ARP (cfr par. I.2.2). Tale procedura di traduzione è nota come *risoluzione di un indirizzo*; questa operazione avviene secondo diverse modalità, ognuna specifica di una data tipologia di sotto-rete. Le modalità più usate sono le seguenti:

- alcuni protocolli ARP utilizzano *tabelle*, che contengono coppie di indirizzi IP e dei corrispondenti indirizzi locali; tali tabelle, predisposte dall'amministratore della sotto-rete in questione, sono contenute in un server (o in più di un server); quando un host (o un router) ha bisogno di una corrispondenza tra un indirizzo IP ed un indirizzo locale, interroga tali server (denominati *ARP-server*); in tal caso l'host (o il router) deve conoscere l'indirizzo locale di almeno un ARP-server, per potergli inviare una richiesta di risoluzione; nulla vieta che un ARP-server sia fisicamente implementato in un calcolatore che svolge anche funzioni di router;
- altri protocolli ARP utilizzano un *algoritmo*; ciò è possibile quando lo schema di indirizzamento della sotto-rete in questione gode di particolari proprietà e gli indirizzi IP sono assegnati agli host (o ai router), in relazione ai rispettivi indirizzi locali;
- altri protocolli ARP ottengono dinamicamente questa corrispondenza usando *messaggi di interrogazione*: quando un host (o un router) ha bisogno di risolvere un indirizzo IP, interroga *tutti* gli host della sotto-rete a cui è collegato, chiedendo a quale host corrisponde quell'indirizzo IP.

Quest'ultima modalità è vantaggiosa quando usata in sotto-reti caratterizzate da capacità di trasferimenti diffusivi ("broadcast") "intrinseci" come, ad esempio, in una LAN Ethernet. Vista la numerosità di queste sotto-reti in Internet, i corrispondenti protocolli ARP sono tra i più usati. In tal caso un messaggio inviato sul mezzo fisico relativo ad una sotto-rete perviene sempre a *tutti* gli host collegati a quel mezzo. Nel messaggio di interrogazione si chiede quindi: "quale host ha l'indirizzo IP X.X.X.X?"; il messaggio perviene a tutti gli host della sotto-rete e *solo* l'host con questo indirizzo IP risponde all'interrogazione con un messaggio in cui comunica: "l'indirizzo IP X.X.X.X corrisponde al (mio) indirizzo locale Y".

Consideriamo, ad esempio, una sotto-rete Ethernet in cui operi un protocollo ARP del tipo appena descritto. Una Ethernet è caratterizzata da indirizzi locali costituiti da stringhe di 48 bit, spesso rappresentati con 12 caratteri esadecimali divisi, a coppie, dal simbolo ":". Assumiamo che in questa Ethernet esista un host, denominato "Plinius", con indirizzo IP "128.6.18.45" e indirizzo Ethernet "08:0:20:9:3e:be" (espresso in caratteri esadecimali). Quando un host (o un router), connesso alla stessa Ethernet, emette un'interrogazione ARP ("ARP REQUEST"), chiedendo a chi corrisponde l'indirizzo IP 128.6.18.45 ottiene la risposta:

"Plinius (128.6.18.45) at 08:0:20:9:3e:be"

Ovvero l'host "Plinius", che è a conoscenza di essere caratterizzato dall'indirizzo IP in questione, risponde comunicando il proprio indirizzo locale Ethernet.

Quando il protocollo ARP ha risolto un indirizzo IP, memorizza la corrispondenza *indirizzo IP→ indirizzo locale* in un'apposita tabella che svolge una *funzione di "cache"* (magazzino nascosto) e che cioè permette di ottenere rapidamente quelle corrispondenze che vengano richieste ripetutamente in brevi intervalli di tempo, senza bisogno di invocare ogni volta il protocollo ARP. Periodicamente le informazioni contenute nella "cache" vengono cancellate in modo da garantire la consistenza con le (eventualmente) mutate condizioni della topologia di rete. In tal modo, un router (o un host) deve emettere un messaggio ARP REQUEST solo quando deve trattare una IP-PDU il cui indirizzo di destinazione non sia contenuto nella sua tabella di corrispondenze.

Risulta molto complesso implementare un protocollo ARP di questo tipo in una sotto-rete priva di capacità di trasferimenti diffusivi intrinseci e legati alla sua topologia fisica. Se, ad esempio, la sotto-rete adottasse una modalità di trasferimento con connessione, il router dovrebbe instaurare connessioni verso *tutti* gli host della sotto-rete ed inviare ad *ognuno* di essi il messaggio ARP REQUEST. Si pensi a cosa ciò significherebbe in una sotto-rete di rilevanti dimensioni. Per questo ambiente (*Non Broadcast Multi-access Network* - NBMA), quando lo si voglia interconnettere a Internet, il problema della risoluzione degli indirizzi risulta essere una delle tematiche più complesse.

In questi casi si preferisce adottare la prima delle soluzioni precedentemente descritte e cioè l'uso di ARP-server. Anche questa soluzione presenta però aspetti critici. Infatti, se si usa un solo ARP-server, questo dovrà gestire numerose richieste e, nel caso di un suo guasto, l'intera sotto-rete non sarebbe in condizioni di operare. Se si usano più ARP-server sorge il problema di coordinarne il funzionamento e di comunicare ad ogni host a quale (o a quali) ARP-server rivolgere le sue richieste.

Infine, consideriamo il caso di risoluzione di indirizzi inversa; ovvero quello in cui è noto un indirizzo locale, ma non il corrispondente indirizzo IP. Questo caso si presenta quando un sistema, all'accensione, pur conoscendo il proprio indirizzo locale, non conosce il proprio indirizzo IP. Per risolvere questo problema si usa il protocollo *RARP*. Il sistema emette un messaggio in cui chiede "qual'è il mio indirizzo IP?", comunicando il proprio indirizzo locale; un RARP-server, residente nella stessa sotto-rete ed opportunamente configurato dall'amministratore di rete, risponde alla

richiesta comunicandogli il relativo indirizzo IP. Altri esempi di protocolli che svolgono funzioni simili a quelle di RARP sono *BootP* e *DHCP*.

Questa situazione è tipica per sistemi privi di dispositivi di memorizzazione di massa, che non possono quindi “ricordare” il proprio indirizzo IP o per sistemi a cui *non* è stato assegnato un indirizzo IP e che si connettono ad Internet per il tramite di un ISP . In quest’ultima eventualità l’ISP dispone di un insieme limitato di indirizzi IP che assegna dinamicamente a quegli host che in un dato momento gli richiedono di connettersi ad Internet. Ognuno di questi host è caratterizzato da un dato indirizzo IP solo per il tempo durante il quale è connesso ad Internet; quando l’host termina di usufruire dei servizi Internet, quell’indirizzo IP sarà assegnato ad un altro host che, in un tempo successivo, chiederà di connettersi ad Internet. In un'altra alternativa, l’ISP assegna ai suoi utenti indirizzi IP che hanno solo un significato locale e, non essendo visibili all'esterno, possono essere scelti in modo arbitrario. Sarà poi compito dell’ISP porre in corrispondenza tali indirizzi con specifiche sessioni di comunicazione.

Questo modo di operare è molto usato per host che accedono ad Internet tramite la rete telefonica e consente sia un risparmio di indirizzi IP, essendo questi condivisi tra più utenti, sia una semplificazione d’uso per gli utenti, i quali non devono richiedere un proprio personale indirizzo IP e configurare conseguentemente il proprio calcolatore. Oltre a tale indirizzo, un utente di questo tipo riceve dal suo ISP anche altre informazioni di configurazione, tra cui l'indirizzo del “*default router*” (cfr. § I.3.5.3) e quello del “*name-server*” (cfr. § I.3.7). Quando un host si connette ad Internet secondo questa modalità, è necessario usare anche un ulteriore protocollo che governa lo scambio di informazioni e di comandi tra utente e ISP. Un esempio di tale protocollo è *PPP* (Point to Point Protocol).

I.3.5. Instradamento

Molti degli algoritmi di instradamento in Internet sono abbastanza semplici ed usano regole basate sul *cammino più breve*.

In Internet, ogni IP-PDU attraversa un cammino composto da router e da sotto-reti. Quando un router consegna una IP-PDU ad una sotto-rete, questo diventa il "corpo" (SDU) della PDU propria dello strato immediatamente inferiore ad IP in questa sotto-rete.

La sotto-rete consegna tale PDU al prossimo router o a destinazione, se la destinazione è all'interno della sotto-rete stessa. Le modalità sono le stesse con cui la sotto-rete tratta le PDU ad essa "appartenenti". Il relativo *incapsulamento* della IP-PDU in unità di dati caratteristiche delle sotto-reti attraversate è esemplificato in Fig. I.3.3. La parte in grigio rappresenta la IP-PDU, in cui si è evidenziato il campo Net_Id contenuto nell’intestazione.

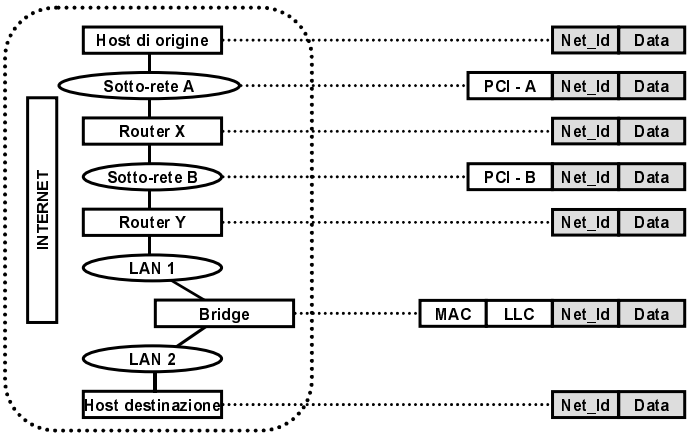


Figura I.3.3 - Esempio di incapsulamento della IP-PDU in unità di dati caratteristiche delle sotto-reti attraversate

Più in dettaglio si può dire che i router instradano le IP-PDU solo verso la rete logica di destinazione e non verso il singolo host a questa connesso. L'algoritmo di instradamento operante nei router determina solo la sequenza dei router da attraversare e non quella di *tutti* i sistemi di ogni sotto-rete e si basa *solo* sulla componente Net_Id dell'indirizzo IP di destinazione. Nel caso di indirizzamento a due livelli, la componente Host_Id viene presa in considerazione *solo* quando una IP-PDU arriva alla rete logica di destinazione.

Analogo discorso vale nel caso di indirizzamento a tre livelli. La componente Sub_Net_Id è presa in esame *solo* quando una IP-PDU arriva alla rete logica di destinazione; sono solo i router locali connessi ad una rete logica in grado di conoscere e di gestire tale suddivisione (tipicamente un solo router). I router esterni ad una data rete logica continuano a basarsi *solo* sulla parte Net_Id dell'indirizzo per instradare le IP-PDU e non prendono mai in esame la componente Host_Id; quindi, dal loro punto di vista, la suddivisione della parte Host_Id non ha nessun significato.

L'instradamento in Internet può essere diviso in due classi: *diretto* ed *indiretto*. L'instradamento diretto è possibile solo se gli host di origine e di destinazione sono connessi alla stessa sotto-rete. L'instradamento indiretto si applica invece quando una IP-PDU deve attraversare almeno un router per giungere a destinazione; ciò avviene sicuramente quando l'host di destinazione è connesso ad una sotto-rete diversa da quella relativa all'host di origine. L'instradamento indiretto può però essere applicato anche tra due host appartenenti alla stessa sotto-rete fisica; l'amministratore di rete può cioè scegliere di far instradare in modo indiretto le IP-PDU emesse da un dato host e dirette ad un host appartenente alla stessa sotto-rete, anche se questo non è strettamente necessario.

I.3.5.1. Instradamento diretto

Un host connesso ad una sotto-rete ha i mezzi per inviare informazioni a qualsiasi altro host connesso alla medesima sotto-rete, grazie alle risorse e ai protocolli di quella sotto-rete. Lo scambio di IP-PDU tra host connessi alla medesima sotto-rete *non* coinvolge, in generale, i router. Quando un host IP deve inviare una IP-PDU in modo diretto, la incapsula nella PDU specifica di quella sotto-rete, traduce l'indirizzo IP di destinazione nel corrispondente indirizzo locale (grazie ai protocolli ARP) ed utilizza i meccanismi propri della sotto-rete in questione per inviare la IP-PDU. Vediamo ora come un host di origine determina se usare un instradamento diretto oppure uno indiretto.

Per prima cosa l'host di origine deve stabilire se l'host di destinazione appartiene alla stessa sotto-rete fisica o meno. Nel secondo caso deve sicuramente usare un instradamento indiretto. Nel primo caso può invece usare entrambi i tipi di instradamento e la decisione tra le due alternative è conseguenza delle scelte dell'amministratore di rete.

Ad un host di origine, nella sua scelta tra instradamento diretto o indiretto, si presentano due alternative principali:

- se la sotto-rete fisica coincide con la sotto-rete logica, l'host di origine deve solo confrontare la parte del proprio indirizzo corrispondente ai campi Net_Id e Sub_Net_Id con la parte dell'indirizzo del destinatario di dimensioni corrispondenti; se tali parti sono uguali allora i due host risiedono nella stessa sotto-rete e il mittente usa l'instradamento diretto; il confronto viene effettuato usando la maschera di sotto-rete (cfr. § I.3.3.2): le parti da confrontare sono quelle caratterizzate da "1" binari nella maschera;
- se la sotto-rete fisica a cui è collegato l'host di origine contiene diverse sotto-reti logiche, allora sono possibili diverse sub-alternative, conseguenze delle scelte dell'amministratore di rete.

Una prima sub-alternativa è che l'host di origine usi l'instradamento diretto solo per destinazioni contenute nella stessa sotto-rete logica a cui esso appartiene; in tal caso per tali comunicazioni si procede come al caso precedente. Per comunicazioni indirizzate ad host appartenenti alla stessa sotto-rete fisica, ma a diverse sotto-reti logiche, l'host di origine adotta un instradamento indiretto, coinvolgendo quindi un router.

Una seconda sub-alternativa è che l'amministratore di rete configuri opportunamente ogni host, comunicandogli in modo esplicito (cfr. § I.3.5.3) quali sotto-reti logiche, oltre a quella a cui esso appartiene, sono raggiungibili direttamente e quindi appartengono alla stessa sotto-rete fisica. Tale comunicazione può riguardare un numero qualunque di sotto-reti logiche contenute nella stessa sotto-rete fisica; se le riguarda tutte, allora l'host di origine può usare l'instradamento diretto per tutte le comunicazioni indirizzate alla sotto-rete fisica a cui appartiene.

Si noti che, se la parte Host_Id non è suddivisa, le decisioni di cui sopra si possono prendere basandosi solo sulla Net_Id: per determinare se un host di destinazione appartiene alla stessa rete logica di quello mittente è sufficiente confrontare la parte Net_Id. In tal caso, a meno che non si usino altre soluzioni alternative a quella della suddivisione in sotto-reti logiche, *non* è possibile che una rete logica comprenda più di una sotto-rete fisica; altrimenti, un host di origine tenterebbe di usare un instradamento diretto anche verso host di destinazione non appartenenti alla propria sotto-rete fisica.

Per concludere citiamo brevemente due soluzioni, alternative alla suddivisione in sotto-reti logiche, per consentire che una rete logica comprenda più sotto-reti fisiche; le soluzioni in parola sono chiamate "*Transparent Routers*" e "*Proxy ARP*". In entrambe queste soluzioni, e come accade anche per la suddivisione in sotto-reti logiche, si partiziona l'insieme degli host appartenenti ad una data rete logica in più sotto-insiemi. Tale partizione *deve* sicuramente essere nota al router o ai router locali che pongono in corrispondenza la rete logica con il resto di Internet; viceversa ciò non è sempre necessario per *tutti* i sistemi in questione. La suddivisione *non deve* però essere visibile dal resto di Internet. Ne segue che l'amministratore di rete può scegliere *qualsiasi* metodologia per definire e rendere nota ai sistemi interessati la partizione in sotto-insiemi della rete logica.

La soluzione "Transparent Routers" si basa su router che svolgono compiti diversi da quelli tradizionali; in particolare essi multiplano (e demultiplano) tutto il traffico "diretto verso" e "proveniente da" una data rete logica in una sola connessione fisica. Essi realizzano, in modo concentrato, la partizione di cui sopra, distinguendo così le IP-PDU di pertinenza di ognuno dei sotto-insiemi in cui la rete logica è divisa.

La soluzione "Proxy ARP" attua la suddivisione a livello degli indirizzi *locali*, invece che al livello degli indirizzi IP. La suddivisione è nota solo ad un opportuno router che, grazie ad una apposita tabella, pone in corrispondenza sotto-insiemi di indirizzi locali con una data sotto-rete fisica; esso può quindi sapere se un host con un dato indirizzo *locale* appartiene o meno ad una data sotto-rete fisica. Questa soluzione è applicata quando le sotto-reti fisiche sono LAN e sono in uso opportuni protocolli ARP. Gli host non sono a conoscenza di questa suddivisione e quando emettono una richiesta ARP, al fine di raggiungere un host che appartiene alla stessa rete logica, e che essi credono appartenga anche alla stessa sotto-rete fisica, il router intercetta tale richiesta e la gestisce in proprio. Il router "fa credere" all'host di origine di essere lui il destinatario e quindi rilancia la IP-PDU in questione verso il vero destinatario.

I.3.5.2. Instradamento indiretto

L'instradamento indiretto è più complesso. Il mittente deve identificare un router a cui inviare una IP-PDU e tale router deve inviare la IP-PDU verso la rete logica di destinazione; si ricorda ancora una volta che i router instradano le IP-PDU solo verso *la rete logica* di destinazione e non verso *il singolo host* a questa connesso. I router non si occupano dell'instradamento *all'interno* delle sotto-reti. Nell'instradamento indiretto viene utilizzata solo la componente Net_Id dell'indirizzo IP. Si immagini una situazione come quella rappresentata in Fig. I.3.3: due host agli estremi di una inter-rete costituita da diverse sotto-reti.

Supponiamo che esista almeno un router connesso alla sotto-rete fisica di origine e che esista almeno un cammino attraverso un certo numero di router che porti alla sotto-rete di destinazione. Il mittente (host o router) invia la IP-PDU al router più vicino (ad esempio, al router A), utilizzando la sotto-rete a cui è direttamente connesso; a tal fine può quindi usare un instradamento *diretto* in cui il router è visto come un destinatario intermedio a cui il mittente invia la IP-PDU, chiedendogli di rilanciarla verso la destinazione finale. Il router A esamina la IP-PDU ricevuta e decide verso quale altro router indirizzarla; una volta presa tale decisione usa un instradamento *diretto* per far pervenire la IP-PDU al router successivo, attraverso la sotto-rete a cui sono entrambi collegati. Il processo si ripete di router in router finché si arriva alla sotto-rete di destinazione; qui tramite un instradamento *diretto* la IP-PDU viene inviata allo specifico host di destinazione.

I router formano una struttura interconnessa e cooperativa. Le IP-PDU passano di router in router finché ne raggiungono uno che può consegnare la IP-PDU tramite un instradamento diretto.

Si può quindi dire che l'instradamento *indiretto* consiste di una successione di instradamenti *diretti*, opportunamente coordinata dai router.

Rimane ora da vedere:

- ✓ come un host di origine individua il primo router a cui inviare una IP-PDU e come tale router decide verso quale altro router rilanciare la IP-PDU stessa, ovvero come si determina la sequenza di router che una IP-PDU deve attraversare;
- ✓ quali sono le procedure operative seguite dagli host per emettere le IP-PDU e dai router per rilanciarle.

Nel seguito ci occuperemo di queste due questioni. In particolare si illustreranno dapprima le procedure operative di inoltro, assumendo che ogni sistema (host o router) *conosca già* quale percorso far seguire ad ogni IP-PDU, ovvero verso quale router emettere (nel caso di un host) o rilanciare (nel caso di un router) le IP-PDU, in funzione dell'indirizzo di destinazione. Successivamente si spiegherà come host e router vengono a conoscenza del percorso da far seguire ad una generica IP-PDU.

I.3.5.3. Tabelle di instradamento

Il meccanismo operativo usato per attuare l'instradamento in IP è basato su una *tabella di instradamento* che ogni host/router mantiene allo scopo di conoscere le possibili destinazioni e i percorsi per raggiungerle. Si noti che non solo i router, ma anche gli host devono avere una tabella di instradamento. Questo perché ogni host ha la necessità di emettere IP-PDU, se non quelle altrui, almeno quelle che esso stesso origina.

Una tabella di instradamento è costituita da coppie (*DN,Ro*) dove *DN* è l'indirizzo della *rete logica di destinazione* e *Ro* è l'indirizzo IP completo del *prossimo router* lungo la strada che porta alla sotto-rete di destinazione. La tabella di instradamento specifica quindi *solo un passo* lungo il cammino verso la destinazione e *non* contiene il cammino *completo* che la IP-PDU dovrà compiere: punta quindi solo a router che possono essere raggiunti attraversando una singola sotto-rete. Quando una sotto-rete di destinazione è raggiungibile direttamente, ovvero si è arrivati all'ultimo router, nella sequenza dei router da attraversare, nella coppia (*DN,Ro*) si sostituisce *Ro* con l'indicazione di instradare in modo diretto.

La tabella, oltre alla coppia (*DN,Ro*), può contenere anche una *metrica* che definisce la distanza dalla destinazione espressa secondo diverse unità di misura: distanza fisica, numero di sotto-reti da attraversare, costo del percorso (espresso in affidabilità o tempo di attraversamento o grado di integrità informativa etc.). Tale informazione può, *opzionalmente*, essere tenuta in conto per la decisione di instradamento (soddisfacendo così la richiesta dell'utente espressa nel campo "Service Type") o per ottimizzare le decisioni prese mediante algoritmi statici o dinamici o adattativi.

La Fig. I.3.4 si riferisce ad una ipotetica sezione di Internet costituita da quattro sotto-reti fisiche e da tre router (Ro1, Ro2 e Ro3), e riporta anche la tabella di instradamento di Ro2. Tale tabella illustra le decisioni che deve prendere Ro2 quando deve instradare una IP-PDU. In questo esempio si assume anche che le sotto-reti fisiche coincidano con reti logiche e che queste ultime non siano suddivise in sotto-reti logiche. Si noti che ogni router dispone di due indirizzi IP, uno per ognuno delle sotto-reti fisiche (e delle relative reti logiche) a cui è direttamente collegato. Ad esempio, Ro1 è noto, all'interno della sotto-rete coincidente con la rete logica con Net_Id=10, mediante l'indirizzo 10.0.0.5 e, all'interno della sotto-rete coincidente con la rete logica con Net_Id=20, mediante l'indirizzo 20.0.0.5.

Una tabella di instradamento ha una dimensione che dipende dal numero di reti logiche interconnesse e che cresce quando nuove reti logiche vi sono aggiunte; *non* dipende invece dal numero di host. Al fine di nascondere il più possibile i dettagli inerenti la inter-rete, di mantenere piccole le tabelle di instradamento e di consentire un instradamento efficiente, le tabelle contengono infatti solo *informazioni sulle reti logiche di destinazione* e non sui singoli host. Al riguardo occorre però aggiungere alcune osservazioni.

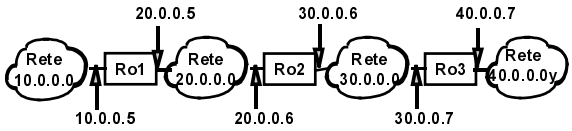


Tabella d'instradamento di Ro2	
Net_Id	Router_Id
20.0.0.0	Instradamento diretto
30.0.0.0	Instradamento diretto
10.0.0.0	20.0.0.5
40.0.0.0	30.0.0.7

Figura I.3.4 - Esempio di interconnessione di quattro sotto-reti e tabella di instradamento del router Ro2.

In primo luogo, siccome ogni router prende le sue decisioni in modo indipendente dagli altri, una comunicazione da A e B può seguire una strada diversa da una comunicazione da B ad A. È quindi necessario prevedere opportuni meccanismi per assicurare una comunicazione bi-direzionale con cammini appaiati per i due versi da A a B e da B a A.

In secondo luogo, sebbene ogni generico host debba avere una tabella di instradamento, ci si sforza di mantenere il più piccole possibile le tabelle degli host forzando questi ultimi a far affidamento sui router nella maggior parte dei casi. Ad esempio, con riferimento alla situazione presentata in § I.3.5.1, la scelta di usare l'instradamento diretto solo per comunicazioni dirette ad host appartenenti alla stessa sotto-rete logica del mittente (nel caso di coesistenza di diverse sotto-reti logiche in una stessa sotto-rete fisica), va in questa direzione.

Questa criterio è applicato anche più in generale. Se un router non trova una strada nella sua tabella, allora indirizza le sue IP-PDU verso un "*default-router*". Il meccanismo di default (mancanza) consiste nel fornire una scelta obbligata, in "*mancanza"* di informazioni che possano contribuire a prendere scelte alternative. Un default-router è generalmente un elaboratore abbastanza potente e destinato principalmente ad operazioni di instradamento o un sistema progettato ad hoc, tenendo conto del carico di lavoro richiestogli. Il meccanismo del default-router è usato quindi:

- da piccoli host, che possono anche inviare ad un default-router *tutte* le IP-PDU non dirette alla rete/sotto-rete logica cui sono collegati; in tal caso la loro tabella di instradamento conterrà una sola strada per tutte le destinazioni diverse da quelle appartenenti alla loro stessa rete/sotto-rete logica;
- da router con una tabella di instradamento di medie dimensioni, che non copre tutte le possibili destinazioni; quando a tali sistemi perviene una IP-PDU con una destinazione non contemplata dalla tabella, essi rilanciano la IP-PDU verso un default-router.

Con riferimento alla tabella del router Ro2, mostrata nella Fig. I.3.4, qualora esistessero altre reti logiche interconnesse, oltre a quelle mostrate in figura, la tabella stessa dovrebbe contenere:

- altre righe, una per ogni altra rete logica interconnessa;
- una sola altra riga strutturata come in Fig. I.3.5, ove il default-router RoDx dovrebbe essere opportunamente identificato.

La tabella di instradamento di RoDx dovrebbe contenere le informazioni necessarie per raggiungere tutte le altre destinazioni non contemplate nella tabella di Ro2. Eventualmente tale meccanismo può essere usato in modo ricorsivo e quindi RoDx può instradare IP-PDU dirette verso destinazioni non conosciute da RoDx stesso verso un altro router RoDy che svolgerebbe quindi funzioni di default-router di RoDx.

Rimane da citare un caso speciale. Come visto in § I.3.1, è possibile chiedere che una IP-PDU, per arrivare a destinazione, segua un pre-determinato percorso, contenuto nel campo "Source Route Option". In altre parole al momento dell'invio di una IP-PDU è possibile richiedere che essa passi attraverso una pre-definita serie di host e/o router. Ciò serve per scopi di controllo e di gestione.

Tutte le *altre* destinazioni	Rilanciare verso l'indirizzo IP del default- router RoDx

Figura I.3.5 - Riga da aggiungere alla tabella di instradamento di Ro2

I.3.5.4. Algoritmo di instradamento

Con queste premesse si può definire l'algoritmo di instradamento che un qualunque sistema (host o router), attua per ogni IP-PDU *ricevuta.* Sia *X* l'indirizzo IP di questo sistema (che per brevità chiameremo *sistema X*). Quando il sistema *X* riceve una IP-PDU possono verificarsi due eventualità a seconda che la IP-PDU abbia come destinazione finale lo stesso sistema *X* o un altro sistema con indirizzo *Y* (*sistema Y*). Nel primo caso la IP-PDU è arrivata a destinazione, mentre nel secondo il sistema *X* deve rilanciarlo verso il sistema *Y* .Ambedue queste eventualità debbono essere contemplate nell'algoritmo che può essere così formulato:

1. estrai l'indirizzo IP di destinazione *Y* dalla IP-PDU;
2. se è stata richiesta una strada specifica (nel campo "Source Route Option"), invialo verso tale strada;
3. se l'indirizzo di destinazione *Y* coincide con *X*, estraine il contenuto informativo e consegnalo agli strati superiori per l'ulteriore elaborazione;
4. decrementa il Time to Live della IP-PDU (cfr § I.3.1); se il Time to Live è arrivato a zero scarta la IP-PDU e danne comunicazione all'host di origine (utilizzando il protocollo ICMP, cfr. §I.3.6);
5. altrimenti, determina la componente Net_Id.Sub_Net_Id dell'indirizzo *Y* (usando la maschera di sotto-rete);
6. se la componente Net_Id.Sub_Net_Id di *Y* coincide con la corrispondente componente di *X*, invia la IP-PDU direttamente (cioè con instradamento diretto); ciò implica tradurre l'indirizzo IP *Y* in indirizzo locale ed incapsulare la IP-PDU nell'unità di dati della sotto-rete in questione; in tal caso viene presa in considerazione anche la componente Sub_Host_Id);
7. altrimenti, consulta la tabella di instradamento; se la componente Net_Id.Sub_Net_Id di *Y* è compresa nella tabella di instradamento, instrada la IP-PDU come specificato nella tabella;
8. altrimenti verifica se almeno la componente Net_Id è compresa nella tabella di instradamento; in caso positivo instrada la IP-PDU come specificato nella tabella; (nel caso in cui la componente Sub_Net_Id ha dimensioni nulle questo passo e il precedente si fondono nella stessa operazione;
9. altrimenti, se è stata specificata una strada di default, invia la IP-PDU al default-router;
10. altrimenti dichiara un errore di instradamento (invocando eventualmente ICMP, § I.3.6) e scarta la IP-PDU.

La definizione completa dell'algoritmo richiede precisazioni:

- il *passo 2* (e cioè il caso di richiesta di una strada specifica) implica che X deve instradare la IP-PDU seguendo le indicazioni contenute nel campo Source Route Option e non quelle che la sua tabella di instradamento suggerisce per raggiungere la destinazione finale. Ciò nonostante deve tener sempre conto della sua tabella per instradare la IP-PDU verso il primo host o router specificato nella sequenza richiesta nel campo "Source Route Option"; ad esempio un utente di X potrebbe voler inviare un messaggio a se stesso (per scopi di prova), facendolo però passare per un router Z; in tal caso X deve inviare la IP-PDU non alla destinazione finale (e cioè a X stesso) ma a Z, il quale poi lo ri-invierà a X;
- il *passo 3* (e cioè determinare se una IP-PDU è diretta a X stesso) va ulteriormente specificato; infatti, in primo luogo, un generico sistema può avere più indirizzi (almeno due nel caso di un router), in secondo luogo X può ricevere anche IP-PDU di tipo "broadcast" e "multicast"; in entrambi questi casi l'indirizzo di destinazione non coincide con X, ma X deve comunque

accettare tali IP-PDU, estrarne il contenuto informativo e consegnarlo agli strati superiori per l'ulteriore elaborazione.

Un'ultima questione è la seguente: l'algoritmo esposto riguarda sia un host che un router e teoricamente è valido per entrambi. In pratica però possono esserci differenze, dovute al fatto che in un host non sono implementati, in generale, tutti i protocolli che invece sono presenti in un router. Ad esempio, la tabella di instradamento di un host è tipicamente configurata dall'amministratore di rete e non viene aggiornata dinamicamente. Nei router, invece, opportuni protocolli contribuiscono a creare e a mantenere aggiornate le tabelle di instradamento (cfr. § I.3.5.4). D'altra parte ciò è naturale dal momento che il router è preposto a svolgere funzioni che un generico host non è tenuto ad eseguire. Ciò premesso, ci si può chiedere se è opportuno che un host svolga comunque tutte le funzioni previste nell'algoritmo di instradamento. In particolare bisogna decidere se un generico host (non un router, ovviamente) deve rilanciare una IP-PDU non diretta a se stesso, contribuendo quindi in modo attivo a gestire comunicazioni altrui. In tal caso l'host svolgerebbe di fatto alcune funzioni di un router. La risposta è che si deve cercare di evitarlo. Questo per le seguenti ragioni:

- ✓ se un host riceve una IP-PDU non inviata a se stesso vuol dire che si è verificato un errore di instradamento; se l'host vi pone rimedio è difficile poi per il gestore di rete individuare l'errore;
- ✓ instradamenti addizionali rispetto a quelli previsti aumentano il traffico globale e sottraggono risorse di calcolo agli utenti degli host;
- ✓ i router si scambiano continuamente messaggi di errore e messaggi per creare e mantenere consistenti le tabelle di instradamento, gli host, in generale, no; se un host instrada IP-PDU senza partecipare a questo scambio di informazioni, e quindi in sostanza ignorando lo stato della inter-rete in un dato momento, si possono verificare situazioni di malfunzionamento anche gravi; ad esempio, se un host accidentalmente manda in "broadcast" un messaggio originariamente destinato ad un host X e tutti gli altri host (e non solo i router) rilanciano le IP-PDU che gli pervengono, allora tutti gli host cui arriva tale IP-PDU ne invieranno una copia a X che si vedrà quindi recapitare un elevato numero di copie dello stesso messaggio.

In conclusione un host *X*, all'atto della *ricezione* di una IP-PDU, applica tipicamente solo i seguenti passi:

1. estrai l'indirizzo IP di destinazione *Y* dalla IP-PDU (=*Y*);
2. se l'indirizzo di destinazione *Y* coincide con *X*, estraine il contenuto informativo e consegnalo agli strati superiori per l'ulteriore elaborazione;
3. altrimenti dichiara un errore di instradamento e scarta la IP-PDU.

L'algoritmo di instradamento che un generico sistema (host o router *X)* attua per ogni IP-PDU che lui stesso *origina* con indirizzo di destinazione *Y* è simile a quello applicato per le IP-PDU ricevute; cioè:

1. se è stata richiesta una strada specifica (nel campo "Source Route Option") invialo verso tale strada;
2. se l'indirizzo di destinazione *Y* coincide con *X*, estraine il contenuto informativo e consegnalo agli strati superiori per l'ulteriore elaborazione (può accadere per ragioni particolari, ad es. di controllo);
3. altrimenti determina la componente Net_Id.Sub_Net_Id dell'indirizzo *Y*, usando la maschera di sotto-rete;
4. se la componente Net_Id.Sub_Net_Id di *Y* coincide con la corrispondente componente di *X*, invia la IP-PDU direttamente (cioè con instradamento diretto); ciò implica tradurre l'indirizzo IP *Y* in indirizzo locale ed incapsulare la IP-PDU nell'unità di dati della sotto-rete in questione; in tal caso viene presa in considerazione anche la componente Sub_Host_Id;
5. altrimenti, consulta la tabella di instradamento; se la componente Net_Id.Sub_Net_Id di *Y* è compresa nella tabella di instradamento, instrada la IP-PDU come specificato nella tabella;
6. altrimenti verifica se almeno la componente Net_Id è compresa nella tabella di instradamento; in caso positivo instrada la IP-PDU come specificato nella tabella; nel caso in cui la componente Sub_Net_Id ha dimensioni nulle questo passo e il precedente si fondono nella stessa operazione;

7. altrimenti, se è stata specificata una strada di default, invia la IP-PDU al default-router;
8. altrimenti dichiara un errore di instradamento, notificandolo agli strati superiori (di *X* stesso) e scarta la IP-PDU.

I.3.5.5. Limitazione della complessità di instradamento

In conclusione, è utile sottolineare quanto segue. In IP, il disaccoppiamento tra la fase di determinazione delle tabelle di instradamento (cfr. § I.3.5.4) e la relativa consultazione fa si che, di fatto, non e' necessario attuare la funzione decisionale per ogni IP-PDU. In altri termini, supponendo che in un certo intervallo di tempo la tabella di instradamento non muti, allora il processamento di una IP-PDU non è poi operativamente molto diverso da quello di un modo con connessione. In entrambi i casi, un opportuno campo dell'intestazione funge da puntatore ad una tabella la quale fornisce l'identificativo della linea di uscita su cui instradare l'unità informativa. La differenza sta nelle dimensione del campo e nelle modalità di ricerca; queste ultime possono essere più efficienti nel caso di modo con connessione rispetto al caso IP.

In particolare, in IP, la scansione della tabella per ricercare l'informazione voluta avviene mediante la tecnica del "*longest prefix matching*". Bisogna cioè cercare un campo di dimensioni *variabili*; infatti l'algoritmo di instradamento parte dell'indirizzo IP di destinazione e cerca nella tabella la più lunga stringa in comune con l'indirizzo di destinazione, a partire dai bit più significativi. La coincidenza tra indirizzo di destinazione e stringa trovata nella tabella può essere completa, riguardare solo la parte Net_ID, o essere nulla. Invece, nei protocolli con connessione l'operazione eseguita è denominata "*table lookup*" e riguarda un campo di dimensioni costanti. E' evidente che quest'ultima operazione è alquanto più semplice del "*longest prefix matching*".

Per mostrare come sia di fondamentale importanza limitare la complessità dei meccanismi di instradamento si ritiene opportuno illustrare come la tecnica del “supernet addressing” sia implementata nelle tabelle di instradamento. Si è detto che per limitare l’uso di reti logiche di classe B si è deciso di consentire una certa violazione della suddivisione in classi degli indirizzi IP (cfr. § I.3.3). Ovvero si è permesso di attribuire più reti logiche ad una stessa organizzazione. In tal modo una data organizzazione, invece di chiedere una rete logica di classe B e non utilizzare tutti i relativi indirizzi disponibili, può chiedere un certo numero di reti logiche di classe C, fino a soddisfare le sue esigenze. Affinché però questo insieme di reti logiche appaia all’esterno come un’unica rete logica è stato necessario introdurre alcune modifiche ai protocolli di instradamento; questo modo di operare è stato denominato “*classless interdomain addressing* (o routing)”.

In teoria non sarebbe però necessaria alcuna modifica a quanto esposto sinora; tutte le reti logiche di classe C, relative alla stessa organizzazione, sarebbero contemplate nelle tabelle di instradamento e tutte avrebbero la stessa informazione di instradamento; ovvero si avrebbero tante coppie (*DN*,*Ro*) con diversi valori di *DN*, ma con lo stesso valore di *Ro*. Siccome si vuole però limitare la dimensione delle tabelle, al fine di diminuire il tempo di ricerca nelle stesse, è stato introdotto il “*classless interdomain addressing”*, il quale consente di inglobare tutte le reti logiche di una stessa organizzazione in una singola informazione di instradamento. Se ad esempio, tutti gli indirizzi IP da 194.0.0.0 a 194.255.255.255 sono assegnati all'Europa, per tutti i router fuori dall'Europa è sufficiente un'unica riga nella tabella di instradamento (contenente l'indirizzo 194.0.0.0 e la maschera 254.0.0.0) per indirizzare tutte le 65536 reti logiche di classe C attribuite all'Europa. La maschera identifica quali sono i bit da considerare significativi. All'interno dell'Europa, questi indirizzi possono poi essere assegnati in blocchi contigui a diverse regioni geografiche o a specifici Internet Service Provider, diminuendo così le dimensioni delle tabelle di instradamento anche per i router interni all'Europa.

I relativi dettagli non sono qui esposti, ma il risultato è quello voluto: inserire nella tabella una sola riga per tutte le reti logiche coinvolte nella “supernet”. Se si adotta tale metodologia gli algoritmi di instradamento devono però essere opportunamente modificati. Infine si fa presente che il supernet addressing è applicabile non solo per aggregare reti logiche di classe C, ma anche per aggregare reti di classe B (e sostituire quindi una rete di classe A).

Si sottolinea che, a seguito dell'introduzione di tale tecnica, la classificazione degli indirizzi in classi precedentemente esposta tende a divenire obsoleta. Gli operatori di rete tendono a fare riferimento direttamente alle informazioni di instradamento contenute nelle tabelle mediante la tecnica del "longest prefix matching" e quindi identificano le reti logiche semplicemente mediante le maschere di rete, indicando il prefisso comune e quindi evitando un esplicito riferimento alla classe. Tutte le classi C dell'esempio appena riportato appaiono ai router esterni all'Europa come un'unica rete logica, di una classe che non è tra quelle precedentemente definite. Ciò giustifica anche il nome "classless". Infine, si nota che tale modo di operare equivale ad introdurre un (parziale) legame tra indirizzi IP e posizione geografica, contrariamente alle scelte originali (cfr. § I.3.5.4); è evidente che tale legame facilita l'instradamento, ma pone vincoli sull'assegnazione degli indirizzi stessi.

I.3.5.6. Determinazione delle tabelle di instradamento

Una volta precisate le *procedure operative* con cui host e router inoltrano le IP-PDU verso le destinazioni volute, utilizzando le tabelle di instradamento, occorre stabilire come queste vengano definite.

Le tabelle di instradamento sono, in generale, *dinamiche*; all'atto dell'accensione, ogni sistema (host o router) inizializza la sua tabella di instradamento (o per mezzo di informazioni conservate in un dispositivo di memoria di massa o interrogando opportuni server). In seguito, i sistemi che dispongono di opportuni *protocolli di instradamento*, aggiornano o incrementano le loro tabelle con nuove informazioni di instradamento. Una singola informazione di instradamento consiste di una coppia (*DN,Ro*).

In aggiunta all'informazione contenuta in una generica coppia (*DN,Ro*), un router ha anche bisogno di conoscere quale interfaccia di rete usare per raggiungere il sistema *Ro.* Ovvero, verso quale sotto-rete, tra quelle a cui è direttamente collegato, inviare una generica IP-PDU per fargli raggiungere il sistema *Ro.* L'informazione di instradamento completa sarà perciò:

- per raggiungere la rete logica con Net_Id DN, invia la IP-PDU verso il sistema Ro, usando l'interfaccia di rete NI".

Da un punto di vista operativo, la coppia (DN,Ro) diventa quindi una terna (DN,Ro, NI). Il terzo elemento NI può essere contenuto nella tabella di instradamento stessa, oppure può essere tenuto in conto mediante altre soluzioni implementative."

Un router può implementare o meno i protocolli di instradamento. Nel secondo caso la tabella di instradamento sarà definita manualmente dall'amministratore di rete e resterà invariata fino a successive operazioni di gestione, eseguite sempre da un operatore. Nel primo caso invece i protocolli di instradamento si occupano automaticamente di acquisire nuove informazioni di instradamento e di aggiornare di conseguenza le tabelle.

La seconda soluzione è tipicamente adottata in router di piccole dimensioni ed in zone dell'inter-rete che possano essere considerate relativamente stabili, avendo come punti di vista la configurazione fisica e la tipologia con la quantità del traffico smaltito. L'amministratore di rete configura le tabelle in base ad informazioni in suo possesso.

Nel seguito ci occuperemo quindi della prima soluzione, in cui le tabelle sono create ed aggiornate per mezzo dei protocolli di instradamento. Osserviamo però che sono comunque possibili soluzioni miste, in cui una prima configurazione è manuale ed i successivi aggiornamenti sono governati dai protocolli di instradamento.

I.3.5.7. Aggiornamento dinamico

I protocolli di instradamento esaminano la configurazione della inter-rete, ed eventualmente lo stato di occupazione delle risorse dei sistemi componenti, ad intervalli di tempo pre-definiti. In seguito a tali operazioni, definiscono le informazioni di instradamento, che quindi sono aggiornate nel tempo, e le comunicano ai router coinvolti. Fisicamente, i protocolli di instradamento sono tipicamente implementati negli stessi sistemi che svolgono funzioni di router.

Quando un protocollo di instradamento determina una strada per una data destinazione, e la comunica ad un router, il router stesso la memorizza nella sua tabella e continua ad utilizzarla per un tempo pre-definito. Trascorso tale tempo, a meno che la stessa strada non gli sia comunicata di nuovo, il router potrà non ritenerla più valida. Se la strada cambia (ad esempio a causa di un guasto o di una modifica dell'inter-rete), i protocolli di instradamento se ne accorgeranno e comunicheranno al router la relativa nuova informazione di instradamento. Se la strada per raggiungere una data destinazione cambia prima che il router ne riceva comunicazione, può succedere che una IP-PDU venga inviata verso una strada non corretta. In tal caso, un opportuno messaggio di errore, generato dal protocollo ICMP (cfr § I.3.6), notificherà al router questo evento. Il processo di aggiornamento delle tabelle continua finché i router rimangono in attività. In seguito ad un eventuale disattivazione il processo ricomincia.

La necessità di un aggiornamento dinamico è dovuta al fatto che Internet non può essere considerata stabile, nuovi host e sotto-reti vengono aggiunti od eliminati frequentemente e molti percorsi che prima erano disponibili possono non esserlo più o viceversa. Inoltre, in caso di guasti, alcune strade non sono utilizzabili. Infine, se i protocolli di instradamento usano un algoritmo adattativo, allora l'aggiornamento deve anche tener conto dello stato di occupazione delle risorse di rete. Le tabelle di instradamento devono quindi essere aggiornate continuamente, anche ad intervalli di pochi secondi. In conclusione, i protocolli di instradamento possono quindi essere definiti come quei protocolli che consentono ai router di comunicare tra loro al fine di definire le strade da usare per rilanciare le IP-PDU verso le destinazioni volute.

La questione di creare e mantenere consistenti le tabelle di instradamento è di per se molto complessa; se inoltre si vuole che le decisioni di instradamento siano sempre ottimizzate in funzione di distanza, affidabilità, efficienza e qualità di servizio allora il problema può apparire insolubile in termini economicamente accettabili. Una descrizione completa di tali meccanismi è molto al di là degli scopi di questa trattazione. Nel seguito si forniranno perciò solo alcuni concetti di base.

I.3.5.8. Sistema autonomo

ARPANET, nucleo originario di Internet, era di dimensioni limitate; era amministrata da un'unica autorità, che poteva decidere autonomamente come gestire la rete e garantire che tutti i possibili percorsi fossero mantenuti consistenti e percorribili. Con il passare del tempo Internet si è estesa e comprende non solo diverse autorità amministrative, ma anche diverse nazioni. Finora abbiamo considerato gli host e i router di Internet solo in termini tecnici; siamo ora costretti a introdurre anche entità amministrative. Al fine di definire con più dettaglio i protocolli di instradamento dobbiamo quindi introdurre una nuova definizione:

- un *sistema autonomo* è un insieme di host, router e sotto-reti controllate da un'unica autorità amministrativa; si distinguono quindi router interni ad un sistema autonomo e router esterni a questo.

Un'autorità amministrativa è libera di scegliere qualsiasi protocollo di instradamento, all'interno del suo sistema autonomo; i sistemi autonomi usano quindi, in generale, diversi protocolli di instradamento. Ogni sistema autonomo deve però affidare in modo specifico ad uno o più dei suoi router il compito di informare il mondo esterno della sua topologia, cioè comunicare quest'ultima agli altri sistemi autonomi e nella fattispecie ad uno o più router all'uopo dedicati da ognuno degli altri sistemi autonomi. Questi *edge router* (router di frontiera) hanno perciò il compito di consentire il colloquio tra sistemi autonomi diversi e quindi tra diversi protocolli di instradamento. Si può fare un paragone tra i router generici sinora considerati e gli edge-router.

Cosi' come i router generici consentono il colloquio tra sotto-reti diverse, così gli edge-router consentono il colloquio tra sistemi autonomi diversi, con riferimento però solo alle informazioni di instradamento. Un edge-router continua a svolgere anche le normali funzioni di un router generico.

I.3.6. Protocolli di instradamento

La nozione di sistema autonomo ha portato a definire due classi di protocolli di instradamento usati per creare ed aggiornare le tabelle di instradamento:

- *Interior Gateway Protocols* (IGP), usati all'interno di un sistema autonomo (questa dizione risente del precedente nome "gateway" usato per indicare i router);
- *Exterior Gateway Protocols* (EGP) usati per le comunicazioni tra router appartenenti a diversi sistemi autonomi e deputati allo scambio di informazioni per informare i rispettivi sistemi autonomi della topologia di altri sistemi autonomi e viceversa.

Nel seguito si discuteranno queste due classi di protocolli.

Per meglio comprendere quanto segue è utile definire la nozione di instradamento basato su *informazioni parziali*. Il concetto alla base di questo instradamento è che non è necessario disporre in ogni router di informazioni di instradamento riguardanti *tutte* le possibili destinazioni. L'importante è che tali informazioni siano "consistenti" e che tutte le destinazioni siano prima o poi contemplate in almeno uno dei router coinvolti. Ad esempio in un incrocio stradale non sono

riportate le indicazioni per andare ovunque, ma solo un insieme ristretto di destinazioni e bisognerà, in generale, attraversare diversi incroci prima di vedere apparire l'indicazione per una data destinazione finale.

L'instradamento con informazioni parziali funziona dunque se prima o poi conduce ad un router dove è contemplata la destinazione finale di interesse, ovvero se le informazioni fornite sono consistenti.

In Internet esistono *core-router* (router di nucleo), che contengono informazioni di instradamento sufficienti per inviare una IP-PDU verso qualunque destinazione. Ogni sistema autonomo deve comunicare ai core-router quali reti logiche fanno parte del sistema autonomo stesso. In tal modo quando ad un core-router si chiede un'informazione di instradamento relativa ad una data rete logica *X*, il core-router è a conoscenza che quella rete logica si trova in un dato sistema autonomo *Y*; esso può quindi comunicare al richiedente che la rete logica *X* si trova nel sistema autonomo *Y*. Una IP-PDU, che ha come destinazione la rete logica *X*, potrà quindi essere inviata ad un edge router del sistema autonomo *Y*; qui arrivato, i protocolli di instradamento interni al sistema autonomo *Y* potranno suggerire la strada per arrivare alla rete logica di destinazione.

Il vantaggio principale dell'instradamento con informazioni parziali è che evita di dover fornire a *tutti* i router le informazioni per raggiungere *tutte* le destinazioni. Il suo svantaggio è che a volte può succedere di dover transitare per un dato router, dove sono citate tutte le informazioni di instradamento, anche se esiste una strada più breve per arrivare a destinazione.

Ciononostante, la questione di assicurare la consistenza di un sistema complesso come Internet rimane complessa, anche perché la Net_Id di un indirizzo non è strutturata in modo gerarchico, né esiste *necessariamente* una corrispondenza, pre-fissata e deducibile dall'esame dell'indirizzo, tra una Net_Id e la posizione geografica della relativa sotto-rete. Ad esempio, le reti logiche 151.100 e 153.100 possono trovarsi in Italia, mentre la rete logica 152.100 potrebbe trovarsi ovunque nel mondo.

Le comunicazioni tra sistemi autonomi e core-router sono effettuate mediante gli Exterior Gateway Protocol; l'instradamento *all'interno* di un sistema autonomo e la raccolta di dati da inviare ai core-router avviene per mezzo degli Interior Gateway Protocols. I messaggi di tutti protocolli di instradamento sono sempre trasportati all'interno di IP-PDU.

I.3.6.1. Protocolli "Interior Gateway"

Alcuni tra i più importanti *Interior Gateway Protocols* sono:

- *GGP* (Gateway to Gateway Protocol);
- *SPREAD;*
- *SPF*(Shortest Path First) *e OSPF* (Open Shortest Path First);
- *RIP* (Routing Information Protocol);
- *HELLO.*

Il modo in cui questi protocolli determinano le tabelle di instradamento può essere vario; si possono utilizzare diversi algoritmi, ognuno dei quali ha i suoi pro e contro.

Il *GGP* era usato originariamente in ARPANET; misura la distanza tra origine e destinazione in "salti", ovvero in numero di sotto-reti da attraversare, ed usa l'*algoritmo di Bellman-Ford*, che determina la strada da seguire minimizzando il numero di salti; uno dei suoi svantaggi è che non è detto che un numero inferiore di salti equivalga ad una strada più veloce e/o più breve; ricordiamo che le sotto-reti non sono tutte uguali; ve ne sono di lente e di veloci, di grandi e di piccole, di congestionate e di non congestionate. Per alleviare questo problema, molti router GGP attribuiscono artificialmente ad una rete con un elevato ritardo di trasferimento un numero di "salti" maggiore di uno. Il ritardo di trasferimento qui considerato è uno di riferimento, costante; non sono tenute in considerazioni situazioni di congestione; in altre parole l'algoritmo non è adattativo. Un altro svantaggio di GGP è che non è facilmente scalabile e cioè, all'aumentare delle dimensioni del sistema autonomo, le sue prestazioni peggiorano rapidamente. I tempi di risposta sono elevati ed è richiesta una rilevante mole di scambi di messaggi.

SPF richiede che ogni router conosca tutta la topologia di rete, rappresentata con un grafo; dal momento che la topologia cambia continuamente (per guasti ed aggiunte od eliminazioni di collegamenti e di host), un continuo scambio di messaggi mantiene aggiornata la topologia memorizzata da ogni router. La strada migliore viene poi determinata localmente (dato che ogni router conosce l'intera topologia) grazie all'*algoritmo di Dijkstra*, che calcola la più breve distanza verso la destinazione, secondo diversi criteri. Il principale vantaggio di questo algoritmo è la velocità; al momento di determinare una strada l'algoritmo risponde immediatamente senza bisogno di interrogare altri router. Lo svantaggio principale è che per mantenere aggiornata la topologia di rete, è necessario un continuo scambio di messaggi, anche quando non ve ne è bisogno.

OSPF è una estensione di SPF. I suoi principali vantaggi sono: a) è disponibile gratuitamente, al contrario di SPF che è proprietario; b) include l'instradamento basato sul Service Type e quindi tiene conto delle richieste degli utenti in termini di prestazioni (cfr § I.3.1); c) consente di bilanciare il carico in rete; d) consente di dividere gerarchicamente un sistema autonomo; e) le sue comunicazioni prevedono un'autenticazione per garantire la sicurezza.

RIP è simile a GGP, minimizza la distanza in salti tra origine e destinazione e, come GGP, aumenta artificialmente il numero di salti di un cammino attraverso reti "lente". RIP mantiene una tabella di instradamento in ogni sistema. Questa tabella è aggiornata ogni 30 secondi con informazioni ricevute dai sistemi vicini. È uno dei protocolli più usati, non perché abbia particolari vantaggi rispetto agli altri, ma perché è stato distribuito insieme al sistema operativo Unix, godendone la popolarità.

HELLO è un esempio di protocollo in cui la metrica usata per valutare la distanza è basata sul ritardo di attraversamento piuttosto che sul numero di salti. Per ritardo di attraversamento si intende quello attualmente subito in un dato momento e quindi HELLO tiene conto di eventuali condizioni di congestione, ovvero è adattativo. Per fare questo HELLO deve innanzi tutto sincronizzare i clock dei diversi router e poi far si che i router si comunichino non solo informazioni topologiche ma anche informazioni relative al ritardo di attraversamento; ciò viene realizzato mediante l'uso di indicazioni temporali, dette "time-stamps". Come anche altri protocolli, HELLO può soffrire di instabilità, se cambia troppo velocemente le strade che consiglia di usare. Se una strada è libera, allora un router vi indirizza molto traffico; allora tale strada diviene congestionata e il router indirizza il traffico verso un'altra strada; ma allora la strada di prima torna ad essere libera e così via. Per evitare queste oscillazioni, i cambiamenti delle strade consigliate non sono molto veloci e il cambiamento avviene solo se esiste una rilevante differenza tra le due strade.

I.3.6.2. Protocolli "Exterior Gateway "

Uno dei più comuni *Exterior Gateway Protocols,* che ha lo stesso nome *EGP,* svolge tre funzioni:

- individuazione dei router adiacenti con cui scambiare le informazioni di instradamento; per adiacenza si intende quella logica; i router delegati allo scambio di informazioni possono essere separati anche da altri router;
- verifica continua della funzionalità dei router interlocutori;
- scambio periodico delle informazioni di instradamento, contenute in appositi messaggi; queste riguardano la sola raggiungibilità delle reti, non la distanza.

EGP non consente ad un router "non-core" di comunicare strade al di fuori del suo sistema autonomo; un router "non-core" può comunicare solo strade appartenenti al sistema autonomo di cui fa parte. EGP non tiene conto di una metrica di distanza, comunica solo una strada da seguire. Dal momento che non può sapere se una strada conviene rispetto ad un'altra, ne comunica una sola. Da ciò deriva anche la ragione per cui un router "non-core" può comunicare solo strade appartenenti al sistema autonomo di cui fa parte. Infatti, dal momento che ogni IGP può usare una sua metrica, diversa da quella usata in altri sistemi autonomi, non avrebbe senso per un router "non-core" comunicare strade attraverso altri sistemi autonomi; visto che non ne conosce le relative prestazioni in termini di prestazioni del trasferimento di dati, ad esempio del ritardo.

Il protocollo *BGP* (Border Gateway Protocol) è una evoluzione dell'EGP. Per lo scambio di messaggi tra le entità vengono utilizzati i servizi di trasporto offerti da TCP. Il protocollo effettua la verifica dello stato di un collegamento o di un host inviando periodicamente *messaggi keep alive* (il periodo raccomandato è di 30 sec.). Rispetto a EGP, fornisce funzionalità aggiuntive: le informazioni sulla raggiungibilità delle sotto-reti di destinazione includono il percorso *completo* che il traffico deve seguire *attraverso* un sistema autonomo per raggiungere determinate sotto-reti di destinazione. Queste informazioni sono sufficienti a costruire il grafo di connettività dei sistemi autonomi; eventuali percorsi chiusi (loop) possono essere poi rivelati ed eliminati con opportune politiche di gestione. L'informazione sulla raggiungibilità delle reti ha ovviamente maggior impatto nel caso in cui sia presente, lungo il percorso tra un sistema autonomo sorgente e uno di destinazione, un certo numero di altri sistemi autonomi con funzioni di transito; in caso contrario l'impiego di BGP od EGP è pressoché equivalente.

Il protocollo BGP consente di utilizzare opportune politiche di restrizione per il traffico in transito, stabilite dal gestore del sistema e codificate in tabelle di configurazione. Ciò consente al protocollo una scelta tra i diversi instradamenti eventualmente disponibili e la possibilità di effettuare una opportuna ri-distribuzione del traffico.

Infine, al processo di instradamento prende parte anche l'ICMP. Infatti, oltre a notificare al mittente l'eventuale mancato recapito di una IP-PDU, il protocollo ICMP prevede un messaggio per la modifica delle informazioni contenute nelle tabelle di instradamento. Qualora una IP-PDU venga instradata erroneamente verso un router, quest'ultimo provvede ad inviare alla sorgente un opportuno messaggio ICMP, che modifica le informazioni della tabella di instradamento del mittente (cfr § I.3.6).

I.3.7. Messaggi di errore e di controllo

Come già detto, IP non è affidabile. Se un router non riesce ad instradare o a consegnare una IP-PDU o se riscontra situazioni anomale (tra cui una congestione di rete) deve poter notificare tali eventi al mittente della IP-PDU, affinché siano attuate operazioni correttive. Possibili situazioni anomale possono essere:

- un dispositivo di rete (incluse le linee di collegamento) non funziona correttamente o non funziona del tutto;
- l'host di destinazione è temporaneamente o permanentemente disconnesso dalla rete;
- il contatore del "Time to Live" arriva a zero;
- i router o le linee intermedie sono talmente congestionate da non poter gestire il traffico in transito.

Per consentire ai sistemi (host o router) di potersi scambiare informazioni circa tali situazioni, è stato definito l'*Internet Control Message Protocol* (ICMP), che è una parte integrante di IP e che deve essere incluso in ogni implementazione di IP.

I messaggi ICMP sono trasportati in rete per mezzo delle IP-PDU, ove vengono incapsulati nel relativo campo informativo: sono costituiti da una parte dati (ICMP Data) e da una intestazione (Fig. I.3.6).

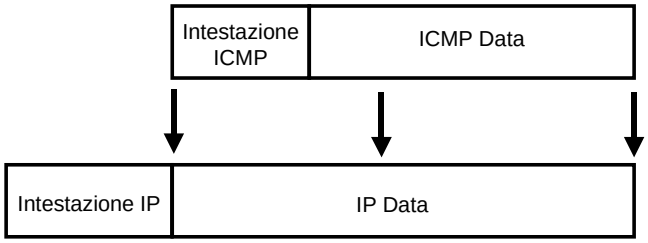


Figura I.3.6 - Trasporto dei messaggi ICMP all'interno di IP-PDU

In caso di malfunzionamento della rete, ICMP provvede ad uno scambio di messaggi fra i sistemi coinvolti per notificare l'errore o per indicare le circostanze inaspettate che causano un comportamento anomalo. I messaggi ICMP viaggiano in rete secondo le stesse modalità seguite dal traffico di utente. Il destinatario di un messaggio ICMP non è un programma applicativo od un utente, ma è una entità dello strato IP. In altre parole ICMP fornisce i mezzi per uno scambio di informazione tra il software IP di un sistema ed il corrispondente software IP di un altro sistema.

ICMP può essere usato sia da router che da host (anche se per questi ultimi sono previste alcune limitazioni). Il suo principale vantaggio è che fornisce una piattaforma unica per lo scambio di qualsiasi informazione di controllo. La funzione di ICMP è solo di notifica degli errori al sistema di origine e non specifica le azioni che devono essere prese per rimediare agli errori ed ai malfunzionamenti; sarà poi il sistema di origine a porre in relazione il particolare errore con il relativo programma applicativo (ad esempio con un protocollo di instradamento, o con IP) ed a decidere cosa fare per correggere il problema.

ICMP notifica eventuali errori solo al sistema che ha originato una IP-PDU e non a sistemi intermedi lungo la strada attraversata dalla IP-PDU stessa. Questo perché ogni IP-PDU contiene solo l'indirizzo del mittente e quello della destinazione e quindi non è possibile, esaminando una IP-PDU, scoprire che strada ha percorso; questa informazione non può nemmeno essere ricavata dalle tabelle di instradamento, lette "a ritroso", sia perché queste ultime cambiano continuamente, sia perché le tabelle riportano solo un passo del cammino seguito e non il cammino completo. Ne segue che, quando una IP-PDU arriva ad un determinato sistema, quest'ultimo non ha modo di sapere che strada ha percorso la IP-PDU, ma solo da dove viene e quindi può notificare un errore solo al sistema di origine della IP-PDU. La conclusione è che, se causa del problema è un sistema intermedio, ICMP non lo può identificare; può solo notificare che esiste un problema ma non quale è.

Un altro punto importante è che, siccome i messaggi ICMP viaggiano come comuni IP-PDU, anch'essi possono essere soggetti ad errore e contribuire alla congestione di rete. La procedura di gestione delle IP-PDU prevede un'unica differenza tra le IP-PDU che trasportano i messaggi ICMP e gli altri: non vengono generati messaggi ICMP in seguito ad errori causati dalle IP-PDU che trasportano messaggi ICMP; ciò serve ad evitare messaggi di errore relativi a messaggi di errore.

Si tenga presente che ogni messaggio ICMP è in relazione ad una specifica IP-PDU e non potrebbe essere altrimenti visto che IP è un protocollo senza connessione; un messaggio ICMP contiene quindi anche un identificativo della particolare IP-PDU che ha generato l'errore o la situazione anomala.

Alcuni dei messaggi previsti da ICMP e le loro funzioni sono i seguenti:

- "*Echo Request*" e "*Echo Reply*"; un sistema, che vuole verificare se un altro sistema è raggiungibile (e quindi se la strada è percorribile) ed è attivo, invia un messaggio "Echo Request"; il messaggio contiene anche una parte dati che può essere un insieme di caratteri casuali o predeterminati; possibili opzioni sono la lunghezza della parte dati ed il numero di messaggi di prova inviati; il sistema a cui perviene un "Echo Request" risponde con un "Echo Reply", ri-inviando i dati che ha ricevuto; il sistema richiedente può così anche valutare eventuali statistiche sul trasferimento, come ritardo minimo, medio e massimo e grado di integrità informativa;
- "*Destination Unreachable*": viene emesso da un router per notificare al sistema mittente che non è in grado di instradare una IP-PDU; contiene un campo che specifica perché la destinazione non è raggiungibile: rete o sistema non raggiungibile, rete o sistema sconosciuti, segmentazione necessaria ma non attuata perché il bit DF era posto ad 1, rete o sistema non raggiungibile con il tipo di servizio richiesto (riportato nel campo Service Type), proibizione amministrativa per la comunicazione con una rete o un sistema, etc.;
- "*Source Quench*": il sistema di destinazione informa il sistema sorgente che il traffico generato è superiore alle sue capacità ricettive; usualmente i sistemi generano un "Source Quench" ogni volta che sono costretti a scartare una IP-PDU; il messaggio contiene anche un identificativo della IP-PDU che è stata scartata; la sorgente provvede a ridurre il numero di IP-PDU inoltrate in rete finché smette di ricevere messaggi "Source Quench"; quando ciò avviene la sorgente ritorna gradualmente verso il ritmo di emissione originario;
- "*Redirect*": un router può informare un sistema sorgente che l'instradamento prescelto non è il migliore o è sbagliato e ne notifica uno nuovo; questo meccanismo integra i meccanismi di inizializzazione ed aggiornamento delle tabelle di instradamento descritti in § I.3.5; se una strada cambia ed un sistema invia erroneamente una IP-PDU lungo quella la strada, un messaggio di "Redirect" risolve il problema; inoltre grazie ai messaggi "Redirect" i

sistemi possono imparare nuove strade; si noti che, normalmente, solo gli host e non i router ricevono messaggi "Redirect"; i router usano tipicamente i protocolli descritti in § I.3.5;

- "*Time Exceeded for a Datagram*": serve ad informare l'host sorgente che una IP-PDU è stata eliminata dalla rete per aver superato il limite temporale di esistenza nella rete (campo Time to Live); si ricorda che lo scopo di questo campo è evitare che una IP-PDU possa percorrere percorsi chiusi (loop) nella rete senza giungere a destinazione, o essere trasferita in un tempo eccessivamente lungo;
- "*Parameter Problem on Datagram*": serve a comunicare ad un host sorgente che sono stati rivelati errori nell'intestazione di una IP-PDU;
- "*Timestamp Request*" e "*Timestamp Reply*": il primo viene inviato da una sorgente per richiedere alla destinazione un'informazione temporale (tempo locale), il secondo viene utilizzato per la risposta; opportuni campi all'interno di questi messaggi consentono una sincronizzazione degli orologi dei due host con un'accettabile precisione;
- "*Information Request*" e "*Information Reply*": originariamente erano utilizzati dagli host per ottenere l'indirizzo IP Internet delle reti logiche a cui erano collegate; attualmente si usano invece i protocolli RARP o BOOTP o DHCP;
- "*Address Mask Request*" e "*Address Mask Reply*": vengono utilizzati dagli host per richiedere ad opportuni server (tipicamente svolgenti anche funzioni di router) la "maschera di sotto-rete" (cfr. § I.3.5.1).

I.3.8. Domain Name System

Gli indirizzi IP in notazione binaria sono quelli *effettivamente* usati da IP. In alternativa a questi indirizzi abbiamo già descritto la notazione "*decimale"*. Anche se quest'ultimo formato consente una più agevole rappresentazione degli indirizzi, gli utenti preferiscono usare indirizzi che siano ancora più facilmente pronunciabili o memorizzabili e che, possibilmente, forniscano un'idea di dove un sistema si trova e a quale organizzazione appartiene: sono gli indirizzi nella notazione *mnemonica*, che nel seguito chiameremo *nome*. Un esempio di corrispondenza tra le notazioni binaria, decimale e mnemonica è illustrato in Fig. I.3.7, ove il nome rende chiaro che si tratta di un Dipartimento della Facoltà di Ingegneria nell'Università di Roma 1 in Italia.

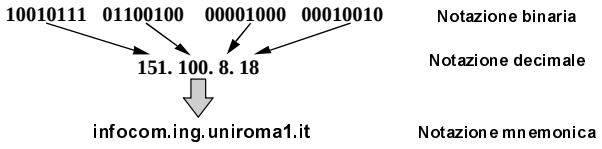


Figura I.3.7 - Notazione mnemonica di un indirizzo IP

Dato che l'indirizzo IP utilizzato nelle IP-PDU è nella notazione binaria, le altre due notazioni a disposizione dell'utente debbono essere preventivamente tradotte (o "risolte") da appositi protocolli.

Il passaggio dalla notazione decimale a quella binaria (e viceversa) è banale in quanto implica una semplice conversione decimale-binario (o binario-decimale). La traduzione tra indirizzo IP binario (o decimale) e nome può essere invece alquanto complessa. Questa traduzione è attuata da un protocollo di alto livello, implementato in un meccanismo denominato *Domain Name System* (DNS).

Il DNS può essere visto come un'agenda telefonica o come un elenco telefonico; siccome può essere difficile ricordare numeri telefonici, risulta utile porli in corrispondenza con i nomi delle persone (o di enti) a cui sono associati. L'agenda telefonica di ognuno di noi, e anche un singolo elenco telefonico, contengono però solo un sotto-insieme dei numeri telefonici della rete telefonica mondiale. Il DNS deve invece consentire di porre in corrispondenza *qualunque* nome con il relativo indirizzo IP. In definitiva il DNS è simile ad un unico elenco telefonico, comprendente però *tutti* coloro che desiderano esservi inseriti, a livello mondiale, consultabile in modo automatico e senza che l'utente finale debba compiere operazioni specifiche. Infatti, gli utenti di Internet usano *direttamente* i nomi, senza avere nemmeno visibilità dei veri indirizzi IP che a questi corrispondono. Sono le applicazioni usate dagli utenti ad invocare il DNS e a chiedere a quest'ultimo le traduzioni nomi-indirizzi.

Il nucleo originale di Internet usava un insieme di nomi non suddiviso gerarchicamente; i nomi erano assegnati da un'unica autorità centrale e la corrispondenza nomi-indirizzi poteva essere contenuta in un unico archivio, memorizzato in un calcolatore che tutti potevano interrogare, o che poteva essere distribuito periodicamente. Come per altre problematiche affrontate sinora, la crescita di Internet ha costretto ha cambiare sistema.

L'analisi dell'attuale DNS è interessante, anche perché fornisce un esempio di una banca di dati distribuita, organizzata gerarchicamente e secondo il paradigma client-server. DNS contiene infatti componenti client e componenti server. Quelle client sono tipicamente implementate nelle applicazioni usate dagli utenti (ad. es. nei browser); quelle server sono residenti in sistemi che svolgono il servizio di traduzione a favore dei client.

L'attuale DNS ha due aspetti; il primo è astratto, specifica la sintassi dei nomi e le regole per decidere chi li attribuisce; il secondo specifica l'implementazione di un algoritmo distribuito per tradurre o "risolvere" un nome in un indirizzo IP e viceversa. Corrispondentemente, nel seguito, la trattazione è divisa in due parti: *attribuzione dei nomi* e *traduzione dei nomi in indirizzi* e viceversa. Ribadiamo ancora che, quando un utente chiede una comunicazione verso un sistema, identificando quest'ultimo con un *nome*, è *necessario* che il DNS traduca il nome nel relativo indirizzo IP, *prima* che inizino tutte le operazioni poste in essere dai protocolli di comunicazione, incluso IP. E' sempre possibile, tuttavia, che un utente usi direttamente l'indirizzo IP di destinazione; in tal modo il DNS non sarà invocato e saranno possibili comunicazioni anche in occasioni di malfunzionamento del DNS stesso. Capita infatti che un utente riceva un messaggio di errore che gli comunica che un dato host non esiste, anche quando ciò non è vero; in questi casi la responsabilità può essere del DNS ed il problema è superabile utilizzando direttamente l'indirizzo IP dell'host in questione, se lo si conosce, ovviamente.

I.3.8.1. Attribuzione dei nomi

L'attuale DNS ha una struttura gerarchica; tale struttura sembra essere la più conveniente, se non l'unica praticabile, per gestire un insieme di nomi di numerosità elevata e variabile nel tempo. L'insieme dei nomi è prima partizionato in un certo numero di sotto-insiemi dall'*INTERnet Network Information Center* (INTER-NIC); il compito di assegnare i nomi all'interno di uno di questi sotto-insiemi è delegato ad un'autorità di livello inferiore che può partizionare il suo sotto-insieme in altri sotto-insiemi e così via. Un nome è quindi composto da una serie di *sotto-nomi* separati da un punto. Ogni punto può separare un'autorità da quella che gli è gerarchicamente inferiore. Ad esempio, INTER-NIC ha assegnato all'Italia il sub-nome "it" delegando un'altra autorità ad assegnare tutti i nomi il cui ultimo sub-nome è "it". Con riferimento all'esempio di Fig. I.3.7, all'Università di Roma "La Sapienza" è stato assegnato il sub-nome "uniroma1", alla Facoltà di Ingegneria è stato assegnato il sub-nome "ing". Infine ad un particolare host del Dipartimento Infocom è stato assegnato il sub-nome "infocom". Il nome risultante per quest'ultimo host è quindi "infocom.ing.uniroma1.it". Come si vede il principio gerarchico è utile e garantisce nel contempo l'unicità dei nomi. Infatti, come per gli indirizzi IP, anche i nomi devono essere unici in tutta l'inter-rete.

È necessario sottolineare che la gerarchia con cui sono assegnati i nomi in Internet segue la struttura delle organizzazioni che sono responsabili di parti del dominio dei nomi e non è necessariamente in relazione con la struttura delle interconnessioni fisiche. Questo sia perché le sotto-reti fisiche non sono necessariamente gerarchiche, sia perché la struttura fisica delle sotto-reti e la struttura gerarchica delle autorità che assegnano i nomi non necessariamente coincidono. Ad esempio un'unica rete in area locale di un'università potrebbe connettere sia gli host del Dipartimento di Informatica che quelli del Dipartimento di Elettronica. Ciononostante i relativi nomi possono essere organizzati in modo gerarchicamente diverso.

Al primo livello gerarchico, l'insieme di tutti i nomi è stato diviso in un primo numero di sotto-insiemi che consentono due tipi di classificazione, una geografica ed una basata sulla tipologia delle organizzazioni che ne fanno parte. Ognuno di tali sotto-insiemi è denominato *dominio di livello massimo* (top level domain); i possibili domini di

livello massimo sono riportati nella Tab. I.3.3. Quelli al di sotto della linea in grassetto, nella classificazione per tipologia, sono stati introdotti recentemente e ancora poco diffusi.

Ognuno dei domini di livello massimo è poi partizionato in un certo numero di *sub-domini*, ognuno dei quali può a sua volta essere partizionato in sub-sub-domini e così via.

Una generica organizzazione può scegliere di registrarsi secondo la classificazione per tipologia o secondo la classificazione geografica; ad esempio la classificazione geografica degli USA è ulteriormente divisa per gli stati della federazione: New York=NY, California=CA, Indiana=IN, etc. Per motivi storici, le organizzazioni americane hanno scelto in maggioranza la classificazione per tipologia (anche perché in un certo periodo questa era l'unica esistente), mentre quelle degli altri paesi hanno seguito prevalentemente la classificazione geografica. Ad esempio le Università di Columbia (New York), Berkeley (California) e Purdue (Indiana) hanno scelto come sub-domini "columbia.edu", "berkeley.edu" e "purdue.edu" piuttosto che "ny.usa", "ca.usa" e "in.usa" rispettivamente, anche perché è più significativa la prima scelta che ci fa capire che si tratta di organizzazioni accademiche e ce ne dice il nome.

La Fig. I.3.8 riporta un esempio di una parte dell'*albero dei nomi* (l'attuale albero conta diversi milioni di nomi). Ad esempio, "uniroma1" è un sub-dominio del dominio di livello massimo "it"; "ing" è un sub-dominio di "uniroma1" e quindi un sub-sub-dominio di "it".

Un nome nel DNS può essere in corrispondenza con diversi *tipi* di indirizzi. In altri termini, ad ogni nome è assegnato un tipo che specifica se quel nome si riferisce ad un host, ad un mailer (cioè ad un sistema con compiti di gestione della posta elettronica), etc. Quando la componente client di DNS chiede di risolvere un nome deve anche specificare che tipo di nome è. Ad esempio, quando un applicativo di posta elettronica chiede a DNS di risolvere un nome, deve specificare che l'indirizzo che DNS fornisce in risposta deve essere quello di un "mailer" ovvero di un sistema che gestisce per gli utenti la posta elettronica; quando un applicativo TELNET chiede a DNS di risolvere un nome deve specificare che si desidera l'indirizzo di un host con cui stabilire una sessione telnet. Inoltre un dato nome può riferirsi a più di un tipo di indirizzo. L'applicazione client di DNS, che richiede una traduzione nome-indirizzo, specifica quindi anche il *tipo* del nome e la componente server risponde con l'indirizzo corrispondente a quel nome *e* di quel tipo.

CLASSIFICAZIONE PER TIPOLOGIA	
nome del dominio	tipo di organizzazione
COM	Commerciali
EDU	Accademiche e didattiche
GOV	Statali
MIL	Militari
NET	Centri di gestione
ARPA	ARPANET (obsoleto)
INT	Organizzazioni Internazionali
ORG	Altre organizzazioni
FIRM	Aziende, affari
STORE	Merce in vendita
WEB	Enfatizzante il WWW
ARTS	Enfatizzante arte e cultura
REC	Enfatizzante intrattenimento
INFO	Enfatizzante fornitori di informazione
NOM	Enfatizzante nomenclature personali

CLASSIFICAZIONE GEOGRAFICA						
nome del dominio	USA	IT	DE	FR	UK	JP
nazione	USA	Italia	Germania	Francia	Gr. Bret.	Giappone

Tabella I.3.3 – Classificazione dei nomi in domini

Infine la sintassi del nome non né specifica il tipo e il numero di sub-nomi che costituisce un nome non è fisso. Ad esempio può esistere un nome composto da due o tre o cinque sub-nomi

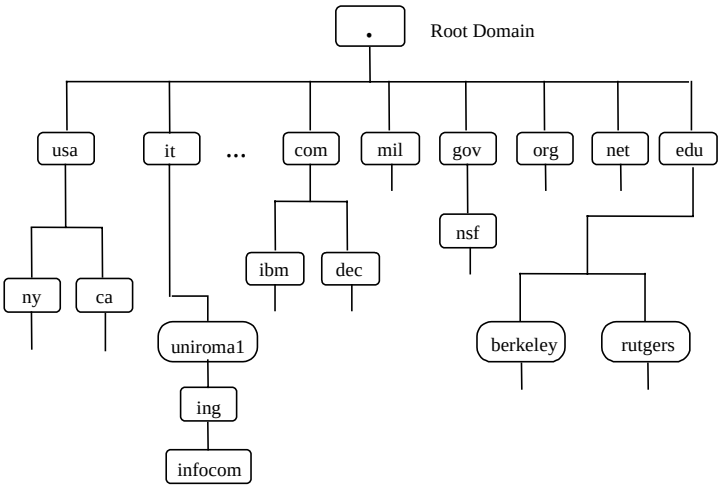


Figura I.3.8 - Esempio di parte dell'albero di nomi

I.3.8.2. Traduzione dei nomi in indirizzi e viceversa

DNS include un *affidabile* ed *efficiente* algoritmo *distribuito* per tradurre nomi in indirizzi e viceversa. È distribuito in quanto è costituito da una molteplicità di sistemi che co-operano tra loro. È efficiente in quanto molti nomi possono essere tradotti localmente senza generare traffico in Internet; è affidabile in quanto il guasto di una singola macchina non pregiudica il funzionamento dell'intero sistema. Qui ci occuperemo della traduzione (o risoluzione) di un nome in un indirizzo IP. L'operazione inversa procede secondo modalità analoghe.

La parte server di DNS è costituita da un certo numero di sistemi indipendenti e co-operanti chiamati *name-server*. Un "name-server" non è altro che una componente server dell'algoritmo, eseguito da un generico elaboratore. Spesso tale programma è eseguito su un elaboratore dedicato. La componente client dell'algoritmo, denominato *name resolver*, usa uno o più name-server per tradurre un nome. I name-server sono organizzati logicamente secondo una struttura gerarchica ad albero. La Fig. I.3.9 mostra un esempio, avente scopo esclusivamente didattico, di una parte di tale struttura. I messaggi di interrogazione e di risposta di DNS sono trasportati, come tutto il traffico in Internet, da IP e usano UDP come protocollo da estremo a estremo.

La radice dell'albero è un name-server (denominato *root-server*) responsabile dei domini di livello massimo; ciascuno di questi domini è a sua volta servito da un name-server ed il root-server è in grado di identificare quale name-server è responsabile per ognuno di essi. Dato un nome da tradurre, il root-server può quindi rivolgersi all'opportuno name-server del dominio di livello massimo. Al terzo livello troviamo i name-server responsabili di un sub-dominio. I server di secondo livello conoscono l'indirizzo IP dello specifico server di terzo livello a cui rivolgersi. Ognuno dei server di terzo livello risponde solo della traduzione di nomi del proprio sub-dominio. Ogni name-server ha in memoria informazioni relative agli host del dominio immediatamente inferiore.

L'albero continua così ad estendersi fino al livello dell'ultimo sub-sub-dominio esistente. Si noti che il name-server di un certo livello gerarchico non deve conoscere *tutti* i nomi degli host ad esso appartenenti; dato un nome da tradurre, deve solo sapere a quale server di livello gerarchico inferiore rivolgersi, se ne esistono. Data la struttura gerarchica dei nomi ciò è abbastanza semplice. Se, ad esempio si richiede al root-server di risolvere il nome "infocom.ing.uniroma.it", tale server indirizzerà la richiesta al server responsabile per il dominio di livello massimo "it"; questo indirizzerà la richiesta al server responsabile del sub-dominio "uniroma1.it" e cosi via finché si incontra un server che conosce l'indirizzo del nome completo.

Le linee che uniscono i server nella Fig. I.3.9 non corrispondono necessariamente a linee fisiche; esse mostrano solo le relazioni logiche tra i name-server e quali altri server un dato name-

server può interrogare. Nella pratica, i singoli server possono essere dislocati ovunque all'interno di Internet e spesso una stessa macchina svolge funzioni di name-server per diversi livelli gerarchici. Inoltre non è raro che un'organizzazione riunisca in un unico server le informazioni relative ai nomi di tutti i suoi sub-domini. In altre parole, la struttura logica rappresentata in Fig. I.3.9, non è necessariamente quella reale; possono esserci eccezioni a qualunque livello. Ritroviamo qui un elemento comune ad altre caratteristiche di Internet, che spesso, come già sottolineato, sfugge ad un inquadramento in paradigmi ben definiti.

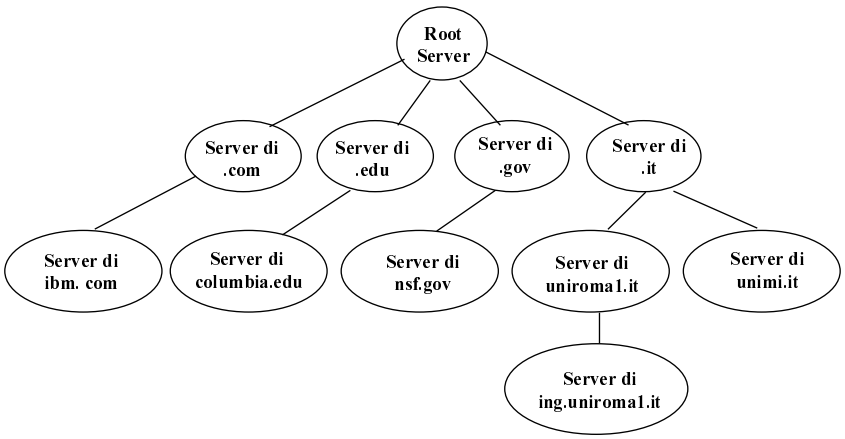


Figura I.3.9 - Struttura gerarchica dei "name-server"

La risoluzione di un indirizzo avviene dunque "dall'alto verso il basso", iniziando dalla radice dell'albero e procedendo attraverso i server di livello gerarchico inferiore. Esistono due modalità per interrogare un server: la *modalità ricorsiva* e quella *non-ricorsiva.*

Nella modalità ricorsiva, quando si interroga un name-server, è responsabilità di quest'ultimo interrogare altri server fino a risolvere completamente il nome; a traduzione ultimata, il primo server interrogato risponderà al sistema che ha inoltrato la richiesta, comunicandogli l'indirizzo richiesto.

Nella modalità non-ricorsiva, il primo name-server interrogato risponderà indicando al sistema richiedente solo a quale altro server rivolgersi (ovvero comunicandogli il relativo indirizzo IP, *non* il nome, altrimenti sarebbe necessario risolvere anche quest'ultimo). Nell'esempio riportato prima, il root-server risponderà dicendo di rivolgersi al server responsabile del top level domain "it"; in seguito, il sistema che ha inoltrato la richiesta si rivolgerà al server "it"; quest'ultimo, se la richiesta che gli perviene è di tipo non ricorsiva, comunicherà di rivolgersi al server "uniroma1.it" e così via. Viceversa se la richiesta fosse stata ricorsiva, sarebbe stato il root-server ad interrogare tutti gli altri e, ultimato il processo, a rispondere al client.

La parte client dell'algoritmo deve quindi conoscere necessariamente l'indirizzo di un solo, qualsiasi, name-server. Da lì potrà poi continuare la sua ricerca. Questo significa che la configurazione iniziale di un generico host deve contenere anche l'indirizzo IP di *almeno* un name-server. Ovvero, un generico host deve avere le seguenti informazioni per poter scambiare dati attraverso Internet:

- il proprio indirizzo IP;
- la maschera di sotto-rete;
- una tabella di instradamento, includente *almeno* l'indirizzo IP di un router di default;
- l'indirizzo IP di un name-server.

Tali informazioni possono essere fornite manualmente o ottenute mediante opportuni protocolli aggiuntivi, alcuni dei quali già citati (ad esempio, RARP, BootP, DHCP, ICMP). Si noti che, in mancanza dell'indirizzo IP di un name-server, l'host in questione potrebbe teoricamente scambiare dati, ma dovrebbe usare e quindi conoscere l'indirizzo IP di ogni destinazione.

Eventualmente si può fornire ad un host anche l'indirizzo di un secondo name-server (backup name-server), per ragioni affidabilistiche.

Per assicurare poi che ogni name-server possa raggiungere gli altri, è sufficiente che esso conosca l'indirizzo del root-server. In aggiunta, un name-server può avere l'indirizzo del name-server ad esso immediatamente superiore o di altri name-server gerarchicamente superiori, inferiori, uguali o trasversali (come giudicato opportuno dall'amministratore di rete).

L'algoritmo appena descritto ha tre svantaggi:

- la gran parte delle richieste degli utenti fa riferimento a nomi locali, risalire ogni volta fino al root-server è inefficiente;

- il root-server è sottoposto ad un carico di elaborazione molto rilevante (anche se più calcolatori possono funzionare in parallelo per svolgere tale compito);
- un guasto del root-server o di un server di alto livello pregiudicherebbe il funzionamento dell'intero DNS.

Per ovviare a questi problemi l'algoritmo è stato integrato con funzionalità di "cache". Ogni server conserva in una memoria ("cache-memory") i nomi che è riuscito a risolvere insieme all'indirizzo del name-server che ha operato la traduzione. Se gli viene richiesta di nuovo la stessa traduzione non ha bisogno di rivolgersi nuovamente al root-server. Tale meccanismo funziona a tutti i livelli gerarchici. Ad esempio, se un name-server negli USA ha risolto il nome infocom.ing.uniroma1.it nel relativo indirizzo IP, e un sistema gli chiede di nuovo di risolvere lo stesso nome, tale name-server conoscerà già la risposta. Inoltre, se allo stesso server viene chiesto di risolvere il nome "dis.fisica.uniroma1.it", non si rivolgerà al root-server ma al server responsabile del dominio "uniroma1.it", di cui "ha imparato" l'indirizzo IP durante la precedente richiesta.

Le informazioni memorizzate nella "cache" hanno un "*tempo di vita*" trascorso il quale vengono cancellate; ciò poiché le associazioni nomi-indirizzi possono cambiare nel tempo (ed alcuni nomi od indirizzi possono cessare di esistere). Quando si memorizza una nuova informazione nella "cache-memory", si scrive anche il suo tempo di vita; quest'ultimo può essere comunicato da chi fornisce l'informazione; il name-server interrogato potrebbe infatti essere in grado di conoscere la "stabilità" dell'informazione comunicata.

Quando un name-server risolve un nome avvalendosi di informazioni registrate nella "cache", comunica anche *da chi* ha avuto quelle informazioni. Inoltre attribuisce a tali informazioni un'etichetta denominata "non di autorità". In tal modo il richiedente sa che l'informazione ricevuta non è "sicura" e, se l'associazione nome-indirizzo non risulta più valida, quando usata, il richiedente stesso sa a chi rivolgersi per avere una traduzione "sicura", senza far ricorso al root-server. Un sistema che inoltra una richiesta può specificare se si accontenta di una risoluzione "non di autorità", ovvero ottenuta grazie al meccanismo di "cache", oppure se ha bisogno di una "risposta di autorità". In quest'ultimo caso il DNS *non* sfrutterà il meccanismo di "cache" e si rivolgerà direttamente a chi è in grado di dare un'informazione sicura ed aggiornata circa un'associazione nome-indirizzo. Non solo i name-server ma anche tutti gli host e tutti i router memorizzano, in genere, nella loro "cache" i nomi precedentemente tradotti.

Concludiamo con esempio, illustrante una possibile procedura di risoluzione di un nome. Se un host vuole risolvere l'indirizzo "st.ryukoku.ac.jp", per prima cosa controlla se la corrispondenza nome-indirizzo IP è contenuta nella sua "cache"; se non c'è, cerca, sempre nella sua "cache", un server di uno dei sub-domini dell'host cercato, "ryukoku.ac.jp", "ac.jp" o "jp"; se non lo trova interroga il name-server della sua zona, che ripeterà la stessa procedura; qualora anch'esso non trovi una corrispondenza nella sua "cache" si rivolgerà (tramite il root-server) ad un name-server del dominio "jp"; quest'ultimo fornirà l'indirizzo di un server nel domain "ac.jp" e così via fino ad arrivare alla traduzione. L'host locale memorizzerà nella sua "cache" l'indirizzo ottenuto e tutti quelli incontrati in questo iter per future eventualità.

Grazie a tale procedura DNS risulta essere molto efficiente ed affidabile. Inoltre, per nomi già conosciuti, la procedura è estremamente rapida. A titolo di verifica si può provare a chiedere per due volte consecutive la risoluzione dello stesso nome. La seconda richiesta sarà esaudita in tempi più rapidi, a meno che, all'atto della prima richiesta, l'informazione in questione non fosse già contenuta nella "cache-memory" del sistema richiedente.

Infine, per assicurare il funzionamento del DNS per ogni nome di Internet, si richiede che un'organizzazione cui sia assegnato un dominio o un sub-dominio di nomi si impegni a gestire i suoi nomi tramite un name-server con relativo backup.

I.3.9. Esempi

Per concludere il capitolo si ritiene utile presentare esempi che illustrano il funzionamento di alcune procedure sinora introdotte.

Un primo esempio è mostrato in Fig. I.3.10, ove si considera un host (detto *di origine*) connesso a una LAN Ethernet. Un utente di questo host vuole stabilire una *sessione ftp* (cfr. § I.1.3) con un host (detto *di destinazione*) il cui nome è "nic.switch.ch"; l'obiettivo è eseguire un trasferimento di file, che deve avvenire nell'ambito di un servizio con connessione, come previsto dal protocollo ftp. L'utente digita quindi il comando "ftp nic.switch.ch", che viene trasferito a una entità dello strato applicativo (*entità ftp di origine*) preposta a trattare il protocollo ftp. I successivi passi sono sintetizzati in Fig. I.3.10 e, per maggiore chiarezza, illustrati qui di seguito.

- In primo luogo, l'entità di origine invoca il DNS per risolvere il nome dell'host di destinazione. La componente client del DNS, residente nell'host di origine, manda un messaggio di

interrogazione alla componente server: a tale scopo utilizza i protocolli UDP e IP; ottenuta la risposta comunica all'entità ftp di origine l'indirizzo IP di destinazione.

- Solo dopo questo passo, l'entità ftp di origine può chiedere di *instaurare una connessione* con l'entità ftp di destinazione; per questo scopo invoca i servizi di TCP e di IP. E' però necessario definire contestualmente la via che deve seguire la connessione tra l'origine e la destinazione; più in particolare, affinchè IP possa trasferire le IP-PDU in cui sono contenuti i dati originati dall'entità ftp, IP stesso deve stabilire verso quale sistema (host o router) inoltrarli. Con questa finalità l'entità IP nell'host di origine (entità IP di origine) deve svolgere la funzione di instradamento, consultando la relativa tabella.
- Un passo obbligato dell'instradamento è determinare se l'indirizzo IP di destinazione appartenga alla stessa rete o sotto-rete logica dell'host di origine.
  - Se vi appartiene, si ricorre all'instradamento diretto. Bisogna però determinare quale indirizzo locale corrisponde all'indirizzo IP di destinazione. A tal fine l'entità IP di origine chiede l'intervento di ARP, che risponde comunicandole l'indirizzo locale Ethernet di destinazione.
  - Sempre nel caso di instradamento diretto, ogni IP-PDU può essere incapsulata nella PDU del sotto-strato LLC, che include la IP-PDU o un suo frammento nel proprio testo e la (LLC)-PCI nella propria intestazione. Il risultato di questa operazione viene poi incapsulato nella PDU del sotto-strato MAC; quest'ultima è poi consegnata allo strato fisico per essere emessa sul bus di Ethernet. In tal modo ogni IP-PDU perviene all'host di destinazione, ove viene decapsulato utilizzandone così l'informazione trasportata.
  - Se l'indirizzo IP di destinazione non appartiene alla stessa LAN Ethernet dell'host di origine, questo ricorre ad un instradamento indiretto, inviando le proprie IP-PDU verso un "default-router". Per questo scopo deve però, anche in questo caso, invocare l'intervento di ARP per determinare l'indirizzo locale Ethernet corrispondente a quello IP del "default-router"; l'intervento non è ovviamente necessario se questa traduzione è già disponibile nella "cache-memory" dell'host di origine.

E' da sottolineare che tutti i passi in cui sono coinvolti IP e gli ulteriori protocolli degli strati inferiori vanno eseguiti per ogni IP-PDU da inoltrare, indipendentemente da quale sia la fase della sessione ftp che determina l'inoltro di quella IP-PDU.

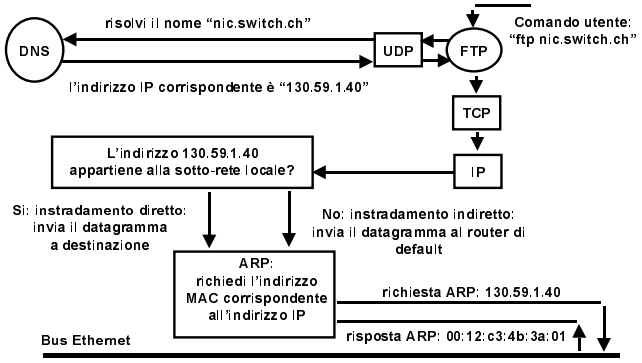


Figura I.3.10 – Risoluzione di nomi e instradamento

Il secondo esempio considera un utente che, usando un applicativo *browser www* (cfr. § I.1.4) residente nell'host di origine denominato "infocom.ing.uniroma1.it", voglia stabilire una sessione http con un host di destinazione con nome www.ibm.com. In Fig. I.3.11 è mostrata l'architettura protocollare di questa comunicazione con riferimento al caso in cui origine e destinazione non appartengano alla stessa sotto-rete fisica: più precisamente l'host di origine faccia capo a una LAN 802.3 (sostanzialmente è Ethernet), mentre l'host di destinazione sia connesso ad una LAN 802.5 (nota come Token Ring). Si suppone inoltre che la componente client di www contenga una analoga componente ftp (cfr. § I.1.4).

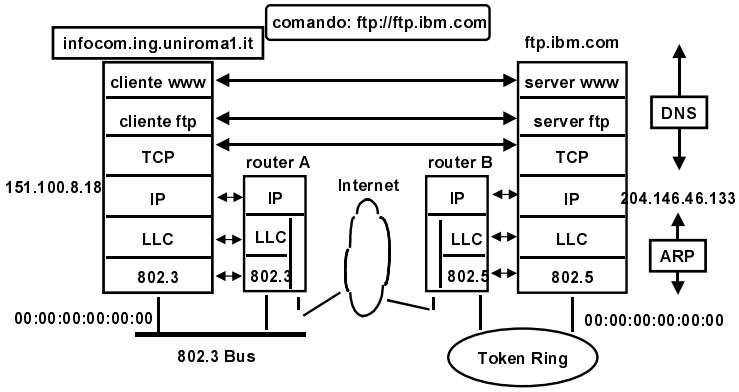


Figura I.3.11– Esecuzione di un comando di utente

La comunicazione considerata in questo secondo esempio si manifesta quindi nella relazione che si stabilisce tra la componente client di www residente nell'host di origine e la corrispondente componente server collocata nell'host di destinazione. Da un punto di vista logico le rispettive entità comunicano direttamente tra loro. Perché ciò accada debbono essere attivate, tra le altre, tutte le procedure richiamate nel primo esempio. Per tener conto che origine e destinazione appartengono a sotto-reti fisiche distinte, in Fig. I.3.11 si assume che siano coinvolti due o più router, di cui due (indicati con A e B) sono rappresentati in modo esplicito. Sono anche rappresentate le relazioni tra i sottosistemi di ugual rango nei due host di origine o di destinazione e nei due router, mettendo così in evidenza che i protocolli di strato non superiore a IP sono gestiti "sezione per sezione", mentre quelli di strato superiore agiscono da "estremo a estremo". Si nota infine che, siccome l'host di destinazione deve rispondere alle richieste dell'host di origine, anche il sistema con nome "www.ibm.com" deve invocare il DNS e l'ARP per fare pervenire le sue risposte al sistema che ha originato il comando.

I.4. I protocolli di strato 4

L'indirizzo IP di destinazione o di origine fa riferimento ad un sistema (host o router), ma non distingue a quale processo applicativo residente nel sistema è diretta una IP-PDU. Nell'architettura di Internet, uno dei compiti dello strato corrispondente al rango 4 (nel seguito, per brevità, indicato come *strato 4*) è quindi distinguere tra i diversi processi applicativi che sono contemporaneamente svolti in un sistema e provvedere a indirizzare le IP-PDU consegnategli dallo strato IP per essere recapitate a uno specifico processo.

In questo indirizzamento occorre però tenere presente che i processi sono creati, modificati o arrestati dinamicamente senza dover rendere noti questi eventi ai possibili mittenti: non sono quindi entità indirizzabili direttamente. Una destinazione (così come una origine) deve invece essere identificata in base alla funzione svolta e non al processo che la realizza. Per questo motivo ogni sistema contiene un insieme di punti di destinazione o di origine che il sistema operativo provvede a porre in corrispondenza con il processo chiamato a svolgere una specifica funzione. Questi punti, chiamati *porte*, sono posti alla frontiera tra lo strato 4 e quello applicativo: costituiscono cioè, secondo la nomenclatura dei modelli architetturali, i *punti di accesso al servizio* (SAP) dello strato 4. Ciascuno di questi SAP identifica quindi in modo univoco una specifica entità dello strato applicativo che è destinazione o origine dell'informazione trasferita.

Spesso una porta dispone anche di una *memoria*, che realizza una fila di attesa per tutte le IP-PDU inviate a quella porta, qualora siano possibili situazioni di congestione. I processi applicativi possono specificare, ed in seguito modificare, la dimensione della fila di attesa. Quando la fila è piena, eventuali PDU entranti sono scartate.

Per porre allora in corrispondenza un processo di origine e uno di destinazione, ciascuno di questi è identificabile con un indirizzo IP e con una porta: quest'ultima è caratterizzabile con un *intero positivo*.

Per assegnare i numeri di porta sono stati definiti due metodi, che sono entrambi utilizzati in alternativa:

- l'*assegnazione universale* (universal assignment): la Internet Assigned Number Authority (IANA) ha definito un insieme di numeri di porta in modo che ad un dato processo applicativo sia associato uno specifico numero di porta; ad es. in UDP la porta numero 53 è associata al processo che implementa il DNS; i numeri di porta così predefiniti sono denominati "*well known ports*" (porte ben conosciute) e tutti coloro che implementano software applicativo per Internet tengono conto di queste assegnazioni; in tal modo, quando si vuole indirizzare una unità di dati ad uno specifico processo si sa quale numero di porta usare; le "well known ports" sono definite in modo specifico nell'ambito dei protocolli UDP e TCP; in Tab. I.4.1 sono riportate alcune "well known ports" di UDP;
- l'*assegnazione dinamica* (dynamic binding): alcuni indirizzi di porta non sono stati assegnati a nessun specifico processo; quando un processo applicativo ha bisogno di inviare dati ad un altro processo applicativo, negozia con quest'ultimo uno specifico indirizzo di porta, tra quelli non già assegnati; l'indirizzo di porta prescelto viene usato solo durante il trasferimento di dati in questione e quindi rilasciato al termine della sessione di trasferimento; tale indirizzo di porta potrà quindi, in seguito, essere usato da altri processi.

I concetti generali ora introdotti sono applicati nei protocolli UDP e TCP. Cominciamo dal primo di questi, rinviando al par. I.5 la trattazione del secondo protocollo.

UDP (User Datagram Protocol) è un protocollo estremamente semplice. La sua funzione principale è indirizzare una porta specifica. Il trasferimento è *senza connessione* e quindi senza garanzie sulla qualità di servizio. Non esegue recupero d'errore e sequenzializzazione delle unità informative. La UDP-PDU, chiamata *datagramma-utente,* ha lunghezza variabile. La *intestazione*, in appositi campi, contiene (cfr. Fig. I.4.1):

- i numeri delle porte di origine e di destinazione;
- la lunghezza dell'intera UDP-PDU;
- la "checksum", per il controllo di errore sull'intera UDP-PDU.

Intero positivo che identifica la porta	Parola chiave	Descrizione del processo associato
11	USERS	Elenca gli utenti attivi in un sistema
17	QUOTE	Citazione del giorno
42	NAMESERVER	Name server di un sistema
53	DOMAIN	DNS
67	BOOTPS	Server del protocollo Bootstrap
68	BOOTPC	Client del protocollo Bootstrap
69	TFTP	Trivial File Transfer Protocol
123	NTP	Network Time Protocol
...	...	...

Tabella I.4.1- Alcune "well known ports" in UDP

All'intestazione segue un campo contenente i dati consegnati da un processo applicativo di origine per essere trasferiti ad uno di destinazione. Alla consegna dei dati corrisponde una accettazione senza vincoli sulla loro lunghezza. UDP eventualmente frammenta le stringhe di dati, inoltrando i dati in IP-PDU distinte (Fig. I.4.2).

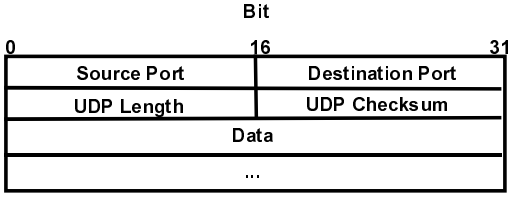


Figura I.4.1 – Formato dell'unità di dati di UDP

La “checksum” della UDP-PDU è *opzionale*; si può fare a meno di usarla riducendo quindi il carico elaborativo per la emissione e la ricezione di ogni UDP-PDU. Infatti, se si usano trasferimenti altamente affidabili, il controllo di errore non è strettamente necessario.

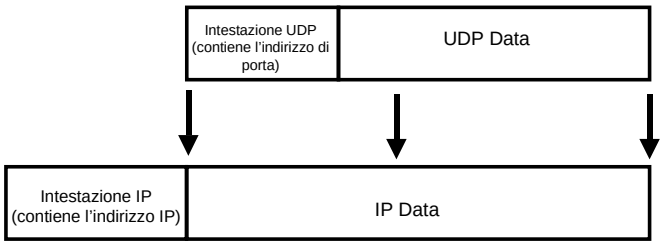


Figura I.4.2 – Incapsulamento della UDP-PDU nella IP-PDU

E' però da osservare che IP non effettua alcun controllo di errore sul campo informativo del IP-PDU IP, per cui, quando si usa UDP come protocollo di strato 4, la checksum di UDP costituisce l'unico strumento per verificare che i dati siano giunti a destinazione correttamente. Qualora tale controllo venga impiegato, esso riguarda non solo *tutta* la UDP-PDU, ma anche un cosiddetto *pseudo-header*, aggiuntivo a quello di UDP e mostrato in Fig. I.4.3. Lo pseudo-header viene considerato al solo fine del calcolo della checksum e non viene trasferito come tale alla destinazione; risulta costituito da:

- gli indirizzi IP della sorgente e della destinazione (contenuti nell'intestazione della IP-PDU);
- il codice IP che identifica UDP (cfr. campo “Protocol type”, in Fig. I.3.1);
- la lunghezza della UDP-PDU;
- un ottetto di *padding* (riempitivo) per fare in modo che la lunghezza complessiva sia multipla di 16 bit.

In altri termini la checksum controlla non solo l’intera UDP-PDU, ma anche gli indirizzi IP contenuti nella IP-PDU che ha trasportato la UDP-PDU. Il motivo per cui viene considerato tale pseudo-header è quello di verificare che la UDP-PDU abbia raggiunto la destinazione corretta. Il solo esame della UDP-PDU non fornirebbe tale garanzia in quanto non contiene l’indirizzo del sistema di destinazione.

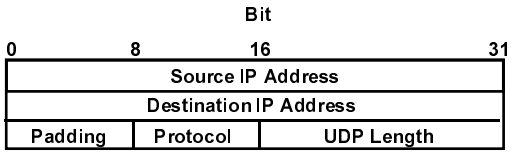


Figura I.4.3 - Formato dello pseudo-header in UDP

I.5. Il Protocollo TCP

TCP (Transmission Control Protocol) è un protocollo di strato 4, operante nel modo con connessione; svolge

- controllo e recupero di errore;
- controllo di flusso;
- ri-ordinamento delle unità informative;
- indirizzamento di uno specifico utente all’interno di un host.

Trasferisce un flusso informativo continuo e bi-direzionale, ma *non strutturato*. Effettua operazioni di multiplazione e de-multiplazione. Sebbene la sua notorietà derivi principalmente dal suo uso in Internet, TCP è sufficientemente generale da poter essere usato anche in altri ambienti (rete telefonica commutata, LAN, MAN, X.25, etc.).

Come avviene per UDP, ogni sistema presenta le *porte* come insieme di punti di destinazione o di origine. Anche in TCP, ogni porta è identificata da un intero positivo. L'indirizzo di un utente di strato 4 è denominato "*port*"; invece l'indirizzo *completo* nell'insieme dei protocolli TCP e IP è denominato "*socket*" ed è costituito da:

port@IP_Address = port@Host_Id.Net_Id.

La componente "port" è contenuta nell'intestazione della TCP-PDU, mentre la componente "IP_Address" è contenuta nell'intestazione della IP-PDU. Questo significa che tutte le sessioni di comunicazione in atto tra due specifici sistemi useranno gli stessi indirizzi IP di sorgente e di destinazione; saranno perciò distinte solo nello strato 4. Ne segue che queste sessioni possono essere viste come "multiplate" su un unico indirizzo IP ovvero su un unico "canale" IP. Qui il termine multiplazione va usato come estensione rispetto al suo significato nell'ambito di una architettura protocollare, ove fa riferimento all'affasciamento di due o più connessioni di un certo strato su un'unica connessione di strato inferiore; in questo caso infatti lo strato inferiore è senza connessione.

Anche in TCP, come in UDP, gli indirizzi delle porte possono essere fissati a priori per uno specifico processo, ovvero essere decisi dinamicamente. In altri termini si possono usare sia l'assegnazione universale che quella dinamica. La gestione delle porte in TCP è però più complessa di quella in UDP poiché alla stessa porta può corrispondere più di un processo. TCP è un protocollo con connessione, a differenza di UDP, e quindi ciò che ha maggiore significato è la definizione di una connessione. In TCP una connessione è identificata da una *coppia* di socket, relativa ai due processi che hanno stabilito la connessione. Ad esempio una connessione tra la porta 1069 dell'host 151.100.8.18 e la porta 25 dell'host 160.80.4.1 è identificata dalla coppia:

"1069@151.100.8.18", "25@160.80.4.1".

Grazie a tale meccanismo, un indirizzo di porta di un sistema può supportare connessioni multiple; la porta 1069 dell'host 151.100.8.18 potrebbe gestire contemporaneamente le seguenti connessioni (ed anche altre):

"1069@151.100.8.18", "25@160.80.4.1"

"1069@151.100.8.18", 25@128.10.2.3 ;

questo perché, lo ribadiamo, ogni connessione è identificata dalla *coppia* socket di origine/socket di destinazione e non *solo* dal socket di destinazione.

La Tab. I.5.1 riporta *alcune* delle "*well known ports*" di TCP. Si noti che alcune di queste sono uguali a quelle definite in UDP, mentre altre sono invece specifiche dell'ambiente TCP.

Intero positivo che identifica la porta	Parola chiave	Descrizione del processo associato
11	USERS	Elenca gli utenti attivi in un sistema
17	QUOTE	Citazione del giorno
20	FTP-DATA	File Transfer Protocol (dati)
21	FTP	File Transfer Protocol
23	TELNET	Protocollo TELNET
25	SMTP	Simple Mail Transfer Protocol
42	NAMESERVER	Name server di un sistema
53	DOMAIN	DNS
79	FINGER	Processo che dà informazioni sugli utenti residenti in un sistema
119	NNTP	USENET News Transfer Protocol
...	...	...

Tabella I.5.1- Alcune "well known ports" in TCP

E' tipicamente responsabilità dei processi applicativi presentare agli utenti umani un'interfaccia che faciliti l'operazione di indirizzamento delle porte. Si vuole cioè evitare che gli utenti debbano conoscere ed usare direttamente gli indirizzi di porta. Internet vuole essere un ambiente in cui le operazioni che gli utenti devono eseguire per comunicare tra loro siano le più semplici possibili. Un esempio di funzionalità introdotte a tal fine, e non *strettamente* necessarie per il funzionamento di Internet, è stato dato con il Domain Name System. Un altro esempio che qui si introduce è l'*indirizzamento automatico* delle porte, eseguito da alcuni processi applicativi:

- se si vuole stabilire una *sessione telnet*, sarà l'applicativo omonimo ad indirizzare automaticamente le informazioni verso la porta 23;
- se si vuole trasferire un file, *usando ftp*, l'applicazione che implementa tale protocollo indirizzerà opportunamente i dati verso le porte 20 e 21;
- se si vuole inviare un messaggio di *posta elettronica*, si userà un indirizzo del tipo: tizio@infocom.ing.uniroma1.it; il protocollo SMTP, implementato negli applicativi di posta elettronica, porrà in corrispondenza il nome "tizio" con un'opportuna porta e farà arrivare il messaggio all'utente voluto.

Poiché IP offre un servizio di consegna non garantito, TCP deve verificare la corretta ricezione delle IP-PDU ed, eventualmente, attuare le procedure necessarie per la loro riemissione. Inoltre è compito di TCP verificare che

✓ le IP-PDU giungano a destinazione nella stessa sequenza con cui sono state emesse;
✓ non vi siano IP-PDU duplicate o mancanti.

Queste funzionalità vengono garantite mediante la numerazione delle TCP-PDU e mediante l'invio di *riscontri* (acknowledgement) da parte della destinazione. TCP offre anche meccanismi di controllo di flusso con lo scopo di impedire il sovraccarico della rete e quindi situazioni di congestione.

Evidentemente tutte le funzionalità fornite da TCP hanno un costo in termini di aumento del ritardo di trasferimento e della quantità di informazioni aggiuntive (overhead) che devono essere trasferite; ciononostante TCP può essere impiegato anche in reti ad alta velocità. In alcuni esperimenti si sono raggiunte portate utili di 8 Mbit/s su una LAN Ethernet con capacità di trasferimento uguale a 10 Mbit/s; è stato dimostrato che in canali opportuni è possibile raggiungere portate utili dell'ordine di 1 Gbit/s. Nel seguito vengono descritti con più dettaglio i meccanismi sin qui elencati e vengono illustrate alcune possibili estensioni di TCP per migliorarne le prestazioni in reti ad alta velocità.

Infine si ricorda che, essendo TCP un protocollo con connessione, nella sua evoluzione sono presenti le fasi di *instaurazione*, di *trasferimento dei dati* e di *abbattimento*.

I.5.1. Formato della TCP-PDU

La TCP-PDU è chiamata *segmento*: il suo formato è illustrato nella Fig. I.5.1. Ogni riga contiene 32 bit.

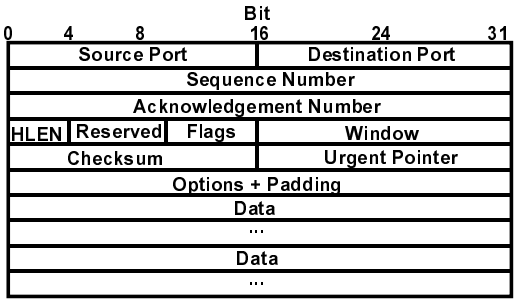


Figura I.5.1- Formato della TCP-PDU

Nel campo informativo sono inseriti i dati consegnati dal processo applicativo di origine. Compito di TCP, se necessario, è frammentare queste stringhe di dati e inoltrarle in IP-PDU distinte (Fig. I.5.2).

I campi dell'intestazione di una TCP-PDU hanno i significati e le funzioni che sono precisati qui di seguito.

- *Source Port:* definisce con 16 bit l'indirizzo logico del processo sorgente dei dati, cioè la *porta di origine*.
- *Destination Port:* definisce, sempre con 16 bit, l'indirizzo logico del processo destinatario dei dati, cioè la *porta di destinazione*.

- *Sequence Number*: è il *numero di sequenza in emissione*, che nel seguito verrà indicato con *SN*; è espresso con un campo di 32 bit ; contiene il numero di sequenza del *primo ottetto* di dati contenuti nella TCP-PDU a partire dall'inizio della sessione TCP: se *SN* = *m* e se la TCP-PDU contiene *n* ottetti il prossimo *SN* è uguale a *m+n.* La numerazione delle TCP-PDU è quindi effettuata non numerando le TCP–PDU stesse, ma gli ottetti in esse contenuti.

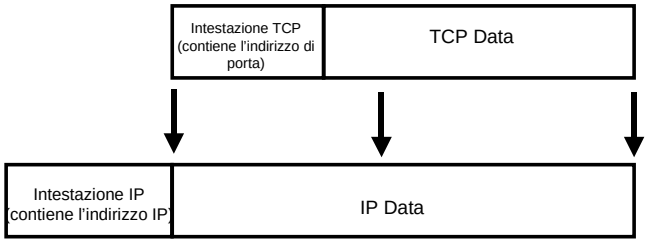


Figura I.5.2- Incapsulamento della TCP-PDU in una IP-PDU

- *Acknowledgement Number:* è il *numero di sequenza in ricezione*, che nel seguito verrà indicato con *AN*; è anch'esso espresso con un campo di 32 bit; nelle TCP-PDU in cui il bit *ACK* (presentato più avanti) è posto al valore binario "1", questo campo contiene il numero di sequenza del prossimo *ottetto* che il sistema emittente si aspetta di ricevere. Nel caso di connessioni interattive bi-direzionali, si usa l'*addossamento* (piggybacking) dei riscontri: si utilizzano cioè TCP-PDU contenenti dati di utente per inviare i riscontri all'emettitore senza dovere, a tal fine, inviare TCP-PDU apposite.
- *Header Length (HLEN)*: esprime il numero di parole di 32 bit contenute nell'intestazione della TCP-PDU; il numero è rappresentato con 4 bit; l'intestazione è sempre costituita da un numero di bit multiplo di 32. La lunghezza HLEN è necessaria in quanto il campo "*Options"* (anch'esso appartenente all'intestazione) è di dimensioni variabili.
- *Reserved*: è riservato per usi futuri; per ora i suoi 6 bit sono posti a "0".
- *Control bit:* i bit di controllo, in numero di 6, sono
  - *URG*: viene posto uguale al valore binario "1" quando il campo "*Urgent Pointer"* (definito in seguito) contiene un valore significativo;
  - *ACK*: viene posto uguale al valore binario "1" quando il campo *AN* contiene un valore significativo;
  - *PSH*: viene posto uguale al valore binario "1" quando l'utente dello strato 4 (applicazione) esige che i dati forniti vengano emessi dall'entità TCP (o consegnati da questa all'applicazione) prescindendo dal riempimento delle memorie allocate fra applicazione e TCP o viceversa; solitamente infatti è il riempimento delle suddette memorie che scandisce la emissione e la consegna dei dati;
  - *RST*: viene posto uguale al valore binario "1" quando un malfunzionamento impone la *re-inizializzazione* (reset) della connessione;
  - *SYN*: viene posto uguale al valore binario "1" solo nella prima TCP-PDU inviata durante l'instaurazione di una connessione, nel qual caso è necessaria una sincronizzazione fra le entità TCP (cfr. § I.5.2);
  - *FIN*: viene posto uguale al valore binario "1" quando la sorgente ha esaurito i dati da emettere.
- *Window:* esprime con 16 bit la *larghezza della finestra*; contiene il *numero di ottetti* che, a cominciare dal numero contenuto nel campo *AN*, il destinatario della TCP-PDU può inviare al mittente senza ricevere riscontri (cfr. § I.5.4). Questo campo consente quindi, per ognuna delle due estremità della connessione, di gestire finestre scorrevoli in emissione e in ricezione, aventi uguale larghezza che varia in relazione alle indicazioni dell'estremità ricevente.
- *Checksum*: contiene, con un campo di 16 bit, gli extra-bit che permettono all'entità TCP ricevente di verificare la correttezza della TCP-PDU ricevuta;

- *Urgent Pointer*: è un *puntatore urgente* che contiene, con un campo di 16 bit, il numero di sequenza dell'ottetto delimitante superiormente i dati da consegnare urgentemente al processo ricevente; il limite inferiore è fornito dal numero di sequenza corrente. Tipicamente questi dati sono messaggi di controllo che esulano dalla comunicazione in senso stretto; a tale traffico ci si riferisce di solito con il termine di "*out-of-band* " (fuori banda): si tratta quindi di messaggi aventi il ruolo di "*interrupt*".
- *Options*: è un campo di lunghezza variabile contenente le *opzioni* di utente; è presente solo raramente; le più note opzioni sono *End of Option List*, *No-operation* e *Maximum Segment Size (MSS)*; ci si soffermerà, in seguito, solo sulla MSS.
- *Padding*: contiene sempre zeri; serve come riempitivo che è aggiunto per far sì che l'intestazione abbia una lunghezza multipla di 32 bit.

Le TCP-PDU possono trasportare solo messaggi di controllo (ad es. per instaurare o abbattere una connessione) o solo dati di utente o entrambi. TCP usa il campo "Control bit" per specificare la funzione ed il contenuto di una TCP-PDU.

Come avviene in UDP, la checksum di TCP controlla non solo l'intera TCP-PDU, ma anche gli indirizzi IP contenuti nella IP-PDU che ha trasportato la TCP-PDU in questione. Il controllo riguarda quindi anche uno *pseudo-header*, mostrato in Fig. I.5.3, che viene considerato al solo fine del calcolo della checksum e che risulta costituito dagli indirizzi IP della sorgente e della destinazione (contenuti nell'intestazione della IP-PDU), dal codice IP che identifica TCP (cfr. campo "Protocol type", Fig. I.3.1), dalla lunghezza della TCP-PDU e da un ottetto di "padding" per fare in modo che la lunghezza complessiva sia multipla di 16 bit. Il motivo per cui viene considerato tale pseudo-header, come nel caso di UDP, è poter verificare che la TCP-PDU abbia raggiunto la destinazione corretta.

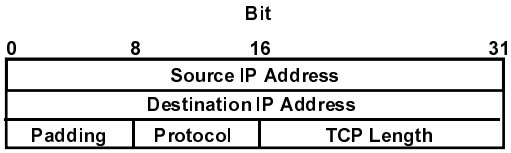


Figura I.5.3 - Formato dello pseudo-header

I.5.2. Instaurazione e rilascio di una connessione

TCP è un protocollo orientato alla connessione. Questo significa che l'entità TCP residente nel sistema mittente deve instaurare una connessione con l'entità TCP residente nel sistema di destinazione, prima che la fase di trasferimento delle informazioni possa avere inizio.

Le due entità TCP da connettere si sincronizzano scambiandosi i loro *SN* iniziali. Con riferimento a una delle due entità TCP, l'*SN* iniziale, indicato con *ISN* (Initial *SN*), rappresenta il numero a partire dal quale tutti gli ottetti emessi sono sequenzialmente numerati, una volta che la connessione sia stata instaurata. Inoltre ciascuno dei due *ISN* non può essere scelto uguale a un valore fisso: ogni volta che si instaura una nuova connessione tra due entità TCP, ciascuna di queste deve scegliere il proprio *ISN* in modo indipendente dall'altra entità nell'ambito di una opportuna *sincronizzazione* reciproca.

Tale sincronizzazione è necessaria per risolvere potenziali situazioni anomale; si ricorda che IP non è affidabile e quindi le IP-PDU possono essere perse, ritardate, duplicate o consegnate fuori sequenza. Inoltre, come si vedrà nel seguito, TCP riemette le proprie PDU che *considera* perse dopo un tempo predefinito (*tempo di riemissione*). Se una TCP-PDU subisce un ritardo di trasferimento superiore al tempo di riemissione, l'entità TCP emittente considera persa questa PDU e quindi la riemette. Al destinatario possono quindi pervenire più copie della stessa TCP-PDU. Se una TCP-

PDU e le sue possibili copie non sono numerate in modo inequivoco, l'entità TCP ricevente non ha modo di interpretare correttamente le TCP-PDU ricevute.

Ad esempio, se la prima TCP-PDU, emessa per chiedere di instaurare una connessione, fosse numerata a partire da un valore fisso (poniamo uguale ad "1"), e se tale TCP-PDU fosse molto ritardata e quindi riemessa (sempre con *ISN* uguale ad 1), all'entità ricevente arriverebbero *due* distinte richieste di instaurazione. Poiché l'entità ricevente non avrebbe modo di sapere che in realtà si tratta *della stessa* richiesta, tenterebbe di instaurare *due* connessioni e risponderebbe *due* volte all'entità emittente. Ma quest'ultima non saprebbe come interpretare la seconda risposta relativa ad una richiesta di instaurazione che tale entità non ha mai effettuato. D'altra parte non si può nemmeno pensare di risolvere questa ambiguità scartando quella che sembra essere una copia della stessa TCP-PDU, poiché potrebbe accadere che un'entità voglia effettivamente instaurare due o più connessioni. Ad esempio, un browser WWW può instaurare più connessioni TCP per trasferire diverse parti di uno stesso documento (immagini, suoni, etc.), così come un utente potrebbe desiderare di stabilire due connessioni TCP per trasferire contemporaneamente due files usando FTP, etc.

Possono quindi sorgere ambiguità se TCP-PDU originarie e riemesse arrivano mentre una connessione sta per essere instaurata, ma anche se TCP-PDU riemesse arrivano dopo che una connessione è stata instaurata, o dopo che sia stata rilasciata. In quest'ultimo caso, ad esempio, l'entità emittente vedrebbe arrivare una TCP-PDU relativa ad una connessione che essa considerava già terminata e quindi non saprebbe come gestire tale PDU.

Esiste anche la possibilità che una TCP-PDU appartenente ad una "vecchia" connessione arrivi ad un sistema dopo che tra gli stessi processi relativi a quella TCP-PDU sia stata instaurata una "nuova" connessione; in tal caso una "vecchia" TCP-PDU si inserirebbe tra i dati relativi ad una diversa connessione. Infine c'è possibilità di confusione quando un sistema cessa di funzionare e perde traccia delle connessioni che altri sistemi considerano ancora in atto. Il progetto di un buon protocollo deve poter tener in conto *qualunque* tipo di evento, per improbabile che possa sembrare. Per risolvere queste situazioni di ambiguità, TCP ricorre alla sincronizzazione dei numeri di sequenza in emissione e in ricezione.

L'idea alla base di questa operazione è garantire che due TCP-PDU con il medesimo *SN* non siano mai in sospeso contemporaneamente. Ciò si può ottenere, con elevatissima probabilità di successo, attraverso:

- una scelta possibilmente casuale dell'*ISN* ;
- una limitazione sulla durata di vita di una TCP-PDU, unitamente a tutti i suoi riscontri nel trasferimento dall'entità emittente a quella ricevente.

Con il primo obiettivo (casualità dell'*ISN*), ogni sistema contiene un orologio con la forma di un contatore binario, che viene incrementato a intervalli uguali e che non è sincronizzato con gli analoghi orologi di altri sistemi. Il numero di cifre binarie del contatore deve essere maggiore o uguale al numero di bit nei campi *SN* e *AN*. Nell'ipotesi di uguaglianza, il contatore può assumere un valore iniziale tra 1 e 2^32 (4.294.967.296). Poiché poi il contatore è incrementato a passi di 4 μs, il ciclo di numerazione ha una durata di circa 4.77 ore (4μs*2^32).

Dal momento che il campo *SN* impiega circa 4.77 ore per compiere un ciclo completo, e poiché il contatore viene incrementato ad una velocità molto superiore a quella relativa all'*SN* di una qualsiasi connessione (almeno alle attuali velocità di trasferimento disponibili in Internet), questo meccanismo risolve i problemi visti sopra. L'*ISN* si sceglie quindi in modo pseudo-casuale tra 1 e 4.294.967.296; in tal modo è estremamente improbabile che due TCP-PDU portino informazioni non coerenti; ad esempio, se due TCP-PDU sono relative a connessioni diverse, esse hanno numeri di sequenza abbastanza "lontani" tra loro (cfr. anche § I.5.3 e I.5.4).

Circa l'obiettivo della limitazione sulla durata di vita di una TCP-PDU e quindi per evitare di confondere gli *SN* delle TCP-PDU, allo scopo provvede il campo TTL contenuto nell'intestazione dell'IP-PDU: questo campo svolge quindi un servizio anche a favore di TCP, oltre a quello relativo ad IP stesso, che, lo si ricorda, risolve eventuali effetti di un instradamento indiretto non corretto.

Ovviamente l'effetto del campo TTL si combina con la durata (oltre 4 ore) del ciclo di numerazione dei campi *SN* e *AN*.

I.5.2.1. Procedura di instaurazione

La Fig. I.5.4 mostra in dettaglio la procedura di instaurazione, che è denominata "3-way handshaking " (stretta di mano a 3 fasi). Quando deve essere instaurata una connessione a livello di strato di applicazione fra un dato processo applicativo, denominato ULP A (ULP=Upper Layer Protocol), residente nel sistema A, ed un ULP B, residente nel sistema remoto B, il primo passo che si deve compiere è l'invio di una "*active open*" (primitiva di Richiesta di Servizio), da parte dell'ULP A all'entità TCP A. L'entità TCP A risponde ad ULP A tramite la primitiva *open id* (primitiva di Risposta di Servizio) ed avvia la procedura di instaurazione.

In questo ambito

- ♦ l'entità TCP A invia all'entità TCP B una TCP-PDU, che è denominata *SYN*; in questa
  - il bit *SYN* è posto al valore logico "1";
  - il campo *SN* è riempito con un *ISN* che è posto uguale al valore *x* assunto *in quel momento* dal contatore residente nel sistema A;
- ♦ l'entità TCP B, nel caso si trovi nelle condizioni di poter accettare la connessione (cioè nel caso abbia ricevuto precedentemente un *passive open* dall'ULP B), risponde all'iniziativa delle entità TCP A con una TCP-PDU, che è ancora denominata *SYN*; in questa
  - i bit *SYN* e *ACK* sono entrambi posti al valore logico "1";
  - il campo *SN* è riempito con un *ISN* che è posto uguale al valore *y assunto in quel momento* dal contatore residente nel sistema B;
  - il campo *AN* è riempito con un valore *x* + "1", che è uguale all'*ISN* di A aumentato di una unità e che quindi riscontra positivamente la TCP-PDU *SYN* emessa da A.
- ♦ l'entità TCP A, alla ricezione della replica dell'entità TCP B, chiude la procedura inviando a questa seconda entità una TCP-PDU, che è chiamata *ACK*; in questa
  - il bit *ACK* è posto al valore logico "1";
  - il campo *SN* è riempito con un valore *y* + 1, che riscontra positivamente la TCP-PDU *SYN* emessa da B.

La ricezione della TCP-PDU *SYN*, emessa dall'entità TCP B e pervenuta all'entità TCP A, permette a quest'ultima di informare il proprio ULP A, mediante una primitiva *open success*, che la connessione da A a B è stata instaurata. L'entità TCP B completa la procedura nel momento in cui riceve la TCP-PDU *ACK*: informa allora il proprio ULP B, con una primitiva open success, dell'avvenuta instaurazione della connessione da B a A.

I.5.2.2. Procedura di rilascio

Il rilascio di una connessione avviene mediante un meccanismo "3-way handshaking" modificato. Si ricorda che una connessione TCP è bi-direzionale e può essere vista come due flussi di dati indipendenti, uno per ciascuna direzione.

Quando un programma applicativo non ha più dati da inviare, ne informa la propria entità TCP, ordinandole di chiudere la connessione *in una direzione.*

Questa entità TCP, che chiamiamo A,

- finisce di mettere i dati restanti, eventualmente rimasti nella memoria di emissione, e attende il relativi riscontri;
- invia, all'entità TCP che è all'altra estremità della connessione e che chiamiamo B, una TCP-PDU, che è chiamata TCP-PDU *FIN*; in questa
  - il bit *FIN* è posto al valore logico "1";
  - il campo *SN* è riempito con un valore x, che segue la sequenzialità di numerazione sulla connessione in corso di abbattimento.

Alla ricezione della TCP-PDU *FIN*, l'entità TCP B

- ▪ informa il "suo" programma applicativo che non ha più dati da ricevere e quindi da trasferirgli;

- invia all'entità TCP A una TCP-PDU, che è di riscontro all'iniziativa di rilascio; in questa TCP-PDU
  - il bit *ACK* è posto al valore logico "1";
  - il campo *AN* è riempito con un valore x + "1", che riscontra la TCP-PDU *FIN* emessa da A.

Una volta che la connessione è stata rilasciata in una direzione,

- ♦ l'entità TCP, a cui la richiesta di rilascio è stata inoltrata, non accetta più dati di utente da quella direzione;
- ♦ i dati continuano a fluire nell'altra direzione, finché anche quest'altra connessione non viene chiusa;
- ♦ fin tanto che almeno una direzione è attiva, *i riscontri* continuano a
  - essere inviati anche da chi ha chiuso la connessione nella prima direzione;
  - essere ricevuti dall'altra entità TCP

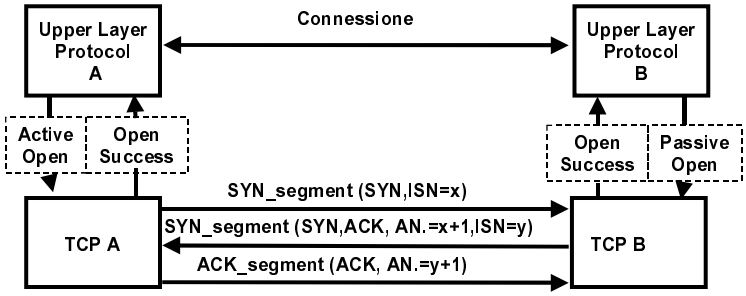


Figura I.5.4- Procedura di instaurazione di una connessione TCP

I dettagli di questa procedura sono più complessi di quella di instaurazione. Questo perché, l'azione richiesta è informare un'applicazione che è pervenuta una richiesta di chiudere una direzione della connessione ed ottenerne una risposta: ciò può determinare un ritardo elevato (implicante ad esempio un'interazione umana).

La procedura di rilascio (o di abbattimento) di una connessione TCP è illustrata nella Fig. I.5.5 Si notino, oltre ai valori assunti dai campi *SN* , *AN*, *FIN* e *ACK*, le differenze rispetto alla procedura di instaurazione. Quando l'entità TCP B riceve una TCP-PDU *FIN*, invece di rispondere immediatamente con un'altra TCP-PDU *FIN*, comunica prima a TCP A di avere ricevuto tale PDU, con una TCP-PDU di riscontro (ed evita così che TCP A continui ad inviare altre TCP-PDU *FIN*); quindi informa l'applicazione in B della richiesta di chiusura ed aspetta che quest'ultima risponda (questa è la fase che potrebbe richiedere una certa attesa, dovuta ad un'interazione con un utente umano). Quando finalmente l'applicazione risponde, TCP B invia una TCP-PDU *FIN* a TCP A; quest'ultima entità riscontra tale TCP-PDU e le procedura si completa.

Infine se una connessione non può essere chiusa secondo la procedura normale, a causa di situazioni anomale, o se un programma applicativo è forzato a chiudere immediatamente una connessione, TCP prevede una *procedura di "reset*". Questa consiste nell'inviare una TCP-PDU con il bit *RST* posto al valore logico "1". Alla ricezione di tale TCP-PDU la connessione è immediatamente terminata senza scambio di ulteriori messaggi e TCP ne informa i programmi applicativi.

I.5.2.3. Dimensione massima di una TCP-PDU

Quando l'entità TCP emittente invia la prima TCP-PDU (*SYN*) per instaurare una connessione con un'entità TCP remota, essa può inserire in tale TCP-PDU un'informazione che rappresenta la massima dimensione del campo dei dati di utente di una TCP-PDU (Maximum Segment Size - MSS) che è in grado di trattare. L'entità ricevente risponde comunicando la propria MSS. Con lo scambio di queste informazioni le due entità TCP interagenti stabiliscono la massima lunghezza

delle TCP-PDU che si scambieranno. Nel caso di uno scambio bi-direzionale di informazione, la dimensione della MSS è scelta in modo indipendente nei due versi e può quindi essere diversa nelle due direzioni .

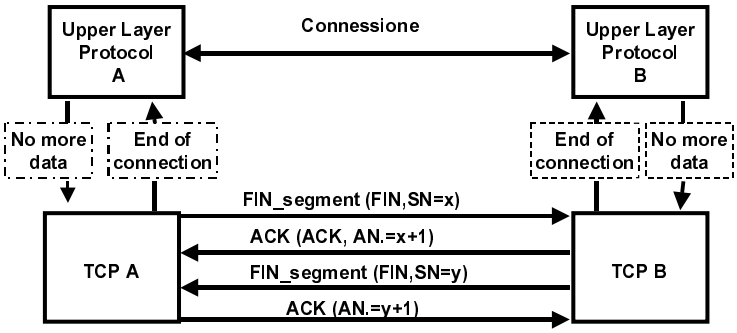


Figura I.5.5 - Procedura di abbattimento di una connessione TCP

La scelta della MSS da parte di ciascuna delle due entità dipende da due fattori: la dimensione della memoria a disposizione delle entità TCP e la dimensione della "Maximum Transfer Unit" (MTU, cfr § I.3.2); ricordiamo che la MTU è la dimensione massima dell'unità di dati di una sotto-rete (tipicamente quella a cui è direttamente connesso il sistema relativo ad una data entità). La MTU è resa nota, sia all'entità IP che all'entità TCP, dal software che interfaccia TCP/IP ad una data sotto-rete (tale software è denominato "driver" di rete).

Il calcolo della Maximum Segment Size (MSS) viene effettuato sottraendo alla MTU la dimensione delle intestazioni delle IP-PDU e delle TCP-PDU. Nel calcolo, un problema potrebbe essere rappresentato dalla variabilità della dimensione delle intestazioni di IP e di TCP, ma in realtà, fatta eccezione per i rari casi in cui si usano le opzioni, si avrà sempre a che fare con entrambe le intestazioni aventi dimensione uguale a 20 ottetti.

Una volta che le due entità hanno determinato le *proprie* MSS e si sono comunicati a vicenda i relativi valori, esse si accordano su una MSS *comune*, da usare per le TCP-PDU che si scambieranno. Di solito, si sceglie il valore minimo tra la MSS offerta dall'entità TCP di destinazione e quella relativa al mittente.

Esistono però casi in cui questo non succede, e cioè quando IP può dover operare una frammentazione delle sue IP-PDU alla sorgente. Nel caso in cui l'opzione di scelta della MSS non venga utilizzata, si impone l'uso, per default, di una MSS uguale a 536 byte. Una tale MSS, aggiunta ai 40 ottetti delle intestazioni dà luogo ad una IP-PDU di dimensione uguale a 576 ottetti, ovvero ad una IP-PDU che tutti i sistemi di Internet devono necessariamente accettare e gestire (cfr. § I.3.2).

In passato si raccomandava, nel caso di sistemi appartenenti a sotto-reti differenti, ed interconnesse quindi da almeno un router, di usare una MSS di 536 ottetti. Attualmente si usano valori anche molto più grandi e sono state avanzate proposte che, nel contesto di una MAN come FDDI, porterebbero alla possibilità di usare un MSS uguale a 4096 ottetti. Questo al fine di aumentare l'efficienza e la velocità del collegamento.

I.5.2.4. Trasmissione di dati urgenti

A volte è necessario inviare dati urgenti (denominati "out of band", fuori banda) senza aspettare che l'entità ricevente finisca di elaborare i dati precedentemente emessi. Si fa notare che una rilevante quantità di dati potrebbe essere "in viaggio" verso l'entità ricevente, memorizzata nei router lungo il cammino della connessione e nella coda di entrata dell'host remoto.

Ad esempio, quando TCP è usato per stabilire una sessione Telnet (emulazione di terminale), un utente potrebbe decidere di inviare un segnale di "interrupt" che termini immediatamente l'applicazione remota. Ciò potrebbe avvenire sia in occasione di malfunzionamenti

dell'applicazione sull'host remoto, o semplicemente perché l'utente non vuole aspettare che un processo termini la sua esecuzione.

Il segnale di "interrupt" (tipicamente "control C") deve poter essere inviato senza aspettare che l'host remoto elabori tutti i dati già inviati, altrimenti un utente non sarebbe in grado di far cessare l'esecuzione di un programma al momento voluto.

A tal fine TCP prevede l'invio di "dati urgenti" che hanno priorità su tutti gli altri dati già inviati e che vengono trasferiti immediatamente al processo remoto; quando i "dati urgenti" sono stati elaborati, il processo remoto riprende in esame i "dati normali". Il meccanismo usato da TCP per inviare dati urgenti consiste nel

- porre il bit *URG* di una TCP-PDU al valore logico "1";
- inserire nel campo "Urgent Pointer" il numero che deve essere sommato a *SN* per ottenere il numero dell'ultimo ottetto del campo-dati del segmento che deve essere consegnato urgentemente al processo ricevente.

I.5.3. Controllo e recupero di errore

La strategia utilizzata per il controllo di errore in TCP è simile a quella usata nei protocolli orientati al bit (ad esempio nel LAP-B) ed è basata sull'uso di finestre in emissione ed in ricezione. Nel seguito si suppone noto tale meccanismo.

TCP considera il flusso di dati in emissione come una sequenza di ottetti. Gli ottetti sono numerati sequenzialmente a partire dal numero pseudo-casuale (ISN) scelto durante la fase di instaurazione: cioè ogni ottetto emesso ha un suo numero d'ordine.

In TCP la dimensione della finestra in emissione coincide con quella della finestra in ricezione. La larghezza della *finestra in emissione* specifica quale sia il numero di ottetti che un'entità TCP può emettere senza ricevere riscontri; la larghezza della *finestra in ricezione* specifica invece il numero di ottetti che un'entità TCP ricevente può accettare anche fuori sequenza.

I.5.3.1. Meccanismo dei riscontri

Per analizzare più in dettaglio questa procedura, consideriamo una connessione a strato TCP e prendiamo in esame *una sola* direzione del trasferimento di dati bidirezionale in atto tra due entità TCP; nell'altro verso di trasferimento si avrà una situazione speculare. Nell'ambito del prescelto verso di trasferimento, una entità TCP avrà il ruolo di mittente mentre l'altra assumerà quello di destinatario. In TCP la dimensione della finestra in emissione non è scelta dal mittente, ma è comunicata al mittente dal destinatario.

Ogni volta che l'entità destinataria emette una TCP-PDU verso quella mittente, comunica nel campo "Window" (16 bit) la larghezza (in ottetti) della finestra di emissione che l'entità mittente deve usare (coincidente con la dimensione della finestra in ricezione). Ciò spiega anche perché le dimensioni delle finestre in emissione ed in ricezione coincidano: sarebbe infatti strano che il destinatario comunichi al mittente che questi può emettere un certo numero di ottetti e quindi non li accetti perché arrivano fuori sequenza (il che non è responsabilità del mittente). Sempre per questo motivo la finestra è chiamata in TCP "Advertised Window", ovvero finestra "comunicata" o "pubblicizzata" (dal destinatario al mittente). L'entità mittente deve usare tale valore della finestra in emissione fino a che non riceve dal destinatario una successiva TCP-PDU con una diversa dimensione di finestra. La dimensione della finestra varia quindi *dinamicamente* nel tempo. Nel seguito esaminiamo il funzionamento dello schema assumendo che tale dimensione sia uguale ad un valore fisso *W*.

Ad esempio, all'inizio di una connessione e quando l'entità mittente non ha ancora emesso nessun dato di utente e quindi non ha ricevuto nessun riscontro relativo a questi dati, essa può emettere solo *W* ottetti; emessi questi deve interrompere la emissione di dati di utente fino a che non riceva almeno un riscontro.

I riscontri sono di tipo *cumulativo*; ovvero il destinatario conferma la ricezione dell'ultimo ottetto di una sequenza di dati ricevuta *completamente* in modo corretto, ovvero senza errori e senza elementi mancanti. In particolare l'entità destinataria comunica a quella mittente il *prossimo* ottetto che si aspetta di ricevere (nel campo *AN*), significando così che *tutti* i precedenti ottetti sono stati ricevuti.

Alla ricezione di un riscontro il mittente "sposta in avanti" la finestra e potrà inviare ancora, senza ulteriori riscontri, ottetti con un numero di ordine compreso tra x e $x+W$, dove x è il numero contenuto nel campo *AN* dell'ultimo riscontro ricevuto. In altre parole, man mano che il mittente riceve riscontri, che gli assicurano l'arrivo a buon fine di parte dei dati emessi, può emettere altri dati.

I.5.3.2. Recupero di errore

Il meccanismo di recupero di errore è basato su un meccanismo di "*time-out*" (fuori tempo massimo). L'entità mittente, dopo avere inviato una TCP-PDU, aspetta un tempo pre-definito (TO) e, se non riceve un riscontro, assume che la TCP-PDU si sia persa.

Questo modo di operare, basato su riscontri cumulativi, presenta sia vantaggi che svantaggi.

Costituisce vantaggio il fatto che :

- il destinatario possa determinare in modo semplice quali riscontri inviare;
- la perdita di un riscontro non causi necessariamente una riemissione.

E' invece uno svantaggio che:

- il mittente non riceva riscontri riguardanti *tutti* i dati che ha emesso, ma solo il numero d'ordine dell'*ultimo* ottetto del flusso di dati ricevuto correttamente dal destinatario.

Quest'ultima caratteristica può condurre a potenziali inefficienze. Supponiamo che:

- la dimensione della finestra in emissione sia di 5.000 ottetti
- l'ISN, scelto durante la fase di instaurazione, sia uguale a 1.000.000
- il mittente abbia ricevuto un riscontro relativo all'ottetto di numero di ordine 1.001.000; in altre parole il destinatario abbia ricevuto tutti i primi 1.000 ottetti, ed invii un *AN* uguale a 1.001.001 (cioè aspetta di ricevere un ottetto con questo numero d'ordine).

Il mittente può emettere 5.000 ottetti; assumiamo che effettivamente li emetta, in 5 TCP-PDU contenenti ciascuna 1.000 ottetti, arrivando così al numero d'ordine 1.006.000. Supponiamo però che al destinatario siano pervenuti gli ultimi 4.000 ottetti, ma non i primi 1.000 dei 5.000 emessi (ovvero che riceva 4 delle 5 TCP-PDU inviate). Il destinatario non può confermare la ricezione degli ultimi 4.000 ottetti, poiché i riscontri sono di tipo cumulativo, e quindi continua a riscontrare solo i primi 1.000, originariamente pervenuti, inviando ancora un *AN* uguale a 1.001.001.

Supponiamo che scada il time-out relativo alla prima TCP-PDU emessa. A questo punto il mittente potrebbe avere due scelte:

- ✓ riemettere tutte le 5 TCP-PDU;
- ✓ riemettere solo la prima TCP-PDU.

Nel primo caso si riemettono inutilmente 4 TCP-PDU, nel secondo il mittente riemette solo una TCP-PDU, ma deve però aspettare un riscontro prima di poter emettere ancora e quindi le prestazioni sono simili, quando si verifica l'evento appena descritto, a quelle di un meccanismo di tipo "Stop and Wait".

Lo standard comunemente accettato di TCP prevede di seguire il secondo tipo di comportamento.

L'unico modo di superare queste inefficienze sarebbe quello di introdurre una modifica al protocollo che consenta al destinatario di informare il mittente della eventuale corretta ricezione di TCP-PDU che siano giunte a destinazione fuori sequenza. In tal modo si potrebbe attuare un meccanismo di riemissione selettiva.

I.5.3.3. Ruolo del tempo di riemissione

Torniamo ora al meccanismo del time-out ed in particolare a come viene determinato il valore dell'intervallo di tempo (denominato TO), trascorso il quale un'entità TCP mittente considera una TCP-PDU persa e quindi la ritrasmette.

Definiamo come *Round-Trip Delay* (RTD) (ritardo di andata e ritorno) il tempo impiegato da un'unità di dati per andare dal mittente al destinatario e tornare indietro. Il RTD è quindi uguale alla somma dei seguenti tempi:

- ritardo di trasferimento in un verso della comunicazione;
- ritardo di trasferimento nell'altro verso;
- tempo necessario al destinatario per rispondere (tempo di risposta);

in cui il ritardo di trasferimento può essere diverso da un verso all'altro della comunicazione.

Normalmente, l'intervallo di tempo TO è scelto uguale a (o maggiore di) RTD. Per poter operare, TCP deve conoscere il TO e quindi è necessario determinare il RTD. Ciò implica valutare sia il tempo di risposta che il ritardo di trasferimento nei due versi della comunicazione.

La determinazione del tempo di risposta è semplice. La determinazione del ritardo di trasferimento, invece, se può essere abbastanza agevole in una specifica sotto-rete, non lo è in Internet: una data connessione può attraversare una sola LAN ad alta velocità o seguire un percorso attraverso numerosi router, attraversando sotto-reti con modeste capacità di trasferimento, in diversi continenti. Inoltre il tempo di trasferimento dipende significativamente da quanto è carica la porzione di inter-rete attraversata. Se si attraversa una sotto-rete congestionata, una TCP-PDU impiegherà molto tempo per arrivare a destinazione. Ne segue che il RTD può variare significativamente anche all'interno della stessa connessione. Ad esempio, la Fig. I.5.6 mostra il ritardo di trasferimento subito da 100 successive TCP-PDU in *una* specifica sotto-rete.

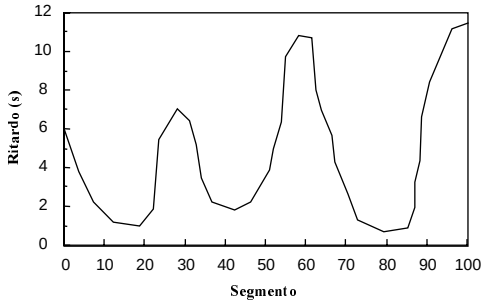


Figura I.5.6- Esempio dell'andamento del ritardo di attraversamento di una sotto-rete

Come si vede il ritardo di trasferimento cambia in modo rilevante da un istante all'altro, all'interno della stessa connessione. Inoltre è facilmente intuibile che esso possa variare in modo ancora più rilevante da connessione a connessione. Ciò significa che non è consigliabile scegliere un valore di TO fisso, a priori; se venisse scelto un valore di TO piccolo, l'entità trasmittente ri-invierebbe senza scopo una TCP-PDU che non è andata perduta ma che è ancora "in viaggio"; se venisse scelto un valore di TO grande, l'entità trasmittente dovrebbe aspettare inutilmente prima di poter assumere che una TCP-PDU sia andata persa.

Per tale motivo, TCP prevede di usare un valore di TO che varia *adattativamente* nel tempo. Un'entità TCP mittente misura, a cadenza pre-stabilita o unità per unità, il RTD e varia il suo TO di conseguenza; se sta trattando una connessione caratterizzata da un alto valore del ritardo di trasferimento sceglierà un alto valore per il TO e viceversa. E' evidente che la misura del RTD e la conseguente scelta del TO sono un aspetto abbastanza critico di TCP. Una scelta errata porterebbe a molti inconvenienti. Ad esempio, si consideri la seguente successione di eventi:

- ✓ se la rete è congestionata, i ritardi di trasferimento aumentano;
- ✓ a causa di una scelta errata del valore di TO, alcune TCP-PDU vengono considerate perse, anche se in realtà non lo sono, e quindi vengono riemesse;

- ✓ ciò aumenta la congestione che causa ancora ri-emissioni finché la portata tende a zero.

D'altra parte, anche in assenza di situazioni così critiche, una scelta errata del TO implica una perdita di efficienza e una minore velocità di trasferimento.

Si noti infine che, se si effettuasse il recupero di errore a strati inferiori, sotto-rete per sotto-rete, sarebbe possibile avere stime più precise sul ritardo di trasferimento; ma in Internet ciò richiederebbe sia la cooperazione tra sotto-reti diverse, sia che un controllo di errore effettuato anche da ogni sistema di interconnessione. Ricordiamo infatti che TCP/IP assume solo che ognuna delle sotto-reti interconnesse sia capace di trasferire informazione, senza richiedere particolari prestazioni; quindi implementa tutte le funzioni che ritiene necessarie, tra cui il controllo di errore. Se alcune o tutte queste funzioni non erano state svolte da una particolare sotto-rete, TCP/IP le realizza; se erano già state svolte le duplica, realizzandole nuovamente.

Come per altre problematiche, si è scelta invece una soluzione che sia la più semplice possibile e che non sia legata a particolari tecnologie delle sotto-reti componenti Internet.

I.5.3.4. Determinazione del tempo di riemissione

Il seguito è dedicato alle procedure di misura del RTD ed alla scelta *adattativa* di un opportuno valore di TO.

Il valore di RTD è più precisamente definito come l'intervallo di tempo che intercorre fra l'istante in cui si inizia la emissione di una TCP-PDU e l'istante in cui se ne riceve il riscontro.

Un'entità TCP mittente inizializza un temporizzatore (timer) nell'istante in cui inizia la emissione di una TCP-PDU; trascorso un tempo uguale a TO, se non riceve un riscontro, riemette la TCP-PDU. Stimare il RTD di una TCP-PDU è in teoria abbastanza semplice; ogni qualvolta TCP emette una TCP-PDU ne memorizza il suo istante di partenza in una memoria dedicata a contenere le informazioni di gestione della connessione; alla ricezione di un riscontro, che informi della corretta ricezione della TCP-PDU in questione, si può agevolmente calcolare il RTD.

Purtroppo la situazione è più complessa poiché un riscontro si può riferire ai dati *complessivamente* ricevuti fino ad un certo momento e non al ricevimento di una *specifica* TCP-PDU. Inoltre si pone il problema di come calcolare il RTD quando vi sono riemissioni, ovvero se calcolare il RTD a partire dall'invio della TCP-PDU originale o di quella riemessa; in realtà nessuna di queste due opzioni fornisce una stima soddisfacente del RTD.

La soluzione scelta è la seguente: si definisce un RTD *medio* (RTDM), calcolato con un'operazione di media tra successive misure di RTD; in particolare l'operazione di media utilizzata è di tipo "a media mobile pesata" (weighted running average), calcolata con la seguente formula:

$$RTDM=\alpha*precedente\_valore\_del\_RTDM+(1-\alpha)*corrente\_valore\_del\_RTD,$$

ove il valore del peso α deve essere scelto opportunamente; minori valori di questo parametro corrispondono ad un veloce aggiornamento del RTDM, maggiori valori rendono il RTDM insensibile a brevi variazioni del ritardo di trasferimento.

Il valore di TO viene quindi scelto uguale a:

$$TO=\beta*RTDM$$

dove β è un altro parametro da scegliere opportunamente.

Per quanto riguarda il problema delle riemissioni, nel calcolo del RTDM, TCP ignora i riscontri di TCP-PDU riemesse, ed aggiorna il RTDM solo con riferimento a TCP-PDU che sono state emesse una sola volta.

Infine, per assicurare la stabilità dell'algoritmo, TCP usa anche una tecnica detta di "back-off"; ogni volta che deve riemettere una TCP-PDU, moltiplica il valore di TO precedentemente calcolato per un opportuno valore, finché la TCP-PDU non va a buon fine. Normalmente il valore di TO viene raddoppiato ad ogni riemissione fino al raggiungimento di un fattore moltiplicativo uguale a

64, ottenuto alla settima riemissione; oltre questo valore la connessione viene re-inizializzata (con una procedura di reset). Il valore di TO tornerà al valore precedente solo dopo la ricezione di un riscontro relativo ad una TCP-PDU che è stata emessa una sola volta.

I.5.4. Controllo di flusso

Il *controllo di flusso*, in questo contesto, è definito come una procedura attuata in modo coordinato da due entità TCP, una emittente ed una ricevente. Tale procedura è intesa a limitare il flusso dei dati trasferiti, in funzione delle risorse a disposizione *nei sistemi terminali* e *prescindendo* dal traffico presente nella inter-rete. Lo scopo del controllo di flusso è evitare che un mittente invii TCP-PDU ad un destinatario che, *in quel momento*, non è in grado di riceverli, a causa di indisponibilità di risorse di elaborazione e di memorizzazione. Tale meccanismo è indispensabile in Internet dove sistemi di dimensioni e di capacità di calcolo molto diverse comunicano tra loro: il più lento dei due deve poter rallentare l'emissione di informazione dell'altro.

Il controllo di flusso in TCP è implementato sfruttando lo stesso meccanismo usato per il controllo di errore e denominato "a finestra scorrevole" (sliding window), orientata all'ottetto. Consideriamo anche in questo caso una connessione di strato 4 e prendiamo in esame *una sola* direzione del trasferimento di dati bidirezionale in atto tra due entità TCP; nell'altro verso di trasferimento si avrà una situazione speculare.

Si è già detto che ogni volta che l'entità destinataria emette una TCP-PDU verso quella mittente, comunica nel campo "Window" la larghezza W (in ottetti) della finestra in emissione che l'entità mittente deve usare. L'entità mittente dovrà usare tale valore di larghezza W fino a che non riceve dal destinatario una successiva TCP-PDU con una diversa dimensione di finestra.

Alla ricezione di un riscontro il mittente potrà inviare ancora, senza ulteriori riscontri, ottetti con un numero di ordine compreso tra x e $x+W$, dove x è il numero contenuto nel campo AN dell'ultimo riscontro ricevuto. Trasmessi questi il mittente *deve interrompere* la emissione di dati di utente fino a che non riceva almeno un riscontro. L'entità mittente può però inviare i riscontri, relativi all'altro verso di trasferimento, purché non usi la tecnica dell'addossamento che comporterebbe anche l'inoltro di dati di utente.

D'altra parte la dimensione della finestra rappresenta anche la quantità di dati che l'entità destinataria è disposta a ricevere in funzione della sua capacità di calcolo e della dimensione della sua memoria in ricezione. L'entità destinataria può quindi variare *dinamicamente* la larghezza della finestra in funzione delle sue esigenze e limitare il ritmo di trasferimento qualora non sia in grado di gestirlo. Ad esempio, se il destinatario comunica al mittente una larghezza di finestra uguale a zero, interromperà completamente l'invio di dati. Il mittente ha però il diritto di usare tutte le opportunità di emissione concesse dalle finestre precedentemente comunicate.

Si noti infine che tale meccanismo giustifica, in parte, la tecnica usata in TCP per la numerazione in ottetti dei campi SN e AN, in quanto consente una maggiore flessibilità nel modulare la larghezza della finestra.

I.5.5. Controllo di congestione

Il *controllo di congestione* ha lo scopo di evitare o di risolvere eventuali situazioni di sovraccarico all'interno della inter-rete. Il meccanismo a finestra scorrevole di TCP funziona però da estremo ad estremo e quindi, in linea di principio, non può essere usato in modo efficiente per il controllo di congestione.

Tuttavia seppure in modo implicito e con alcune limitazioni, lo schema a finestra scorrevole di TCP può proteggere sia il destinatario che, in caso di congestione, la inter-rete. Se la porzione di inter-rete attraversata dalle TCP-PDU di una data connessione è congestionata, al mittente arriveranno, per una data larghezza di finestra, meno riscontri; ciò forza il mittente ad emettere meno informazione. Inoltre, siccome TCP effettua misurazioni sul RTD, il valore del TO è stimato in modo opportuno e si eviteranno riemissioni inutili che porterebbero ad un aumento della

congestione invece che ad una sua diminuzione. Inoltre un'entità TCP destinataria può utilizzare il RTD come misura di congestione e quindi per decidere opportunamente la larghezza della finestra da comunicare all'entità mittente.

Infine il meccanismo di riemissione a intervalli crescenti descritto in precedenza (back-off) coopera in modo significativo a ridurre la congestione.

Tutto questo significa che, *calibrando opportunamente i parametri del protocollo*, si può effettuare non solo un controllo di flusso ma anche un controllo di congestione.

Le prime implementazioni di TCP utilizzavano per il controllo della congestione, in aggiunta ai meccanismi sopra descritti, anche il protocollo ICMP. Ricordiamo infatti che ICMP può rallentare il ritmo di emissione del sistema mittente, mediante l'invio di messaggi opportuni (Source Quence), nel momento in cui il sistema destinatario si trovi a dover rifiutare IP-PDU a causa della mancanza di risorse di memoria (cfr. § I.3.7).

Questo meccanismo di ICMP, nel caso di rapide variazioni del traffico, sembrò però del tutto insufficiente nel contesto di reti ad alta velocità (ad es. nelle LAN). Si propose, quindi, fin dalla seconda metà degli anni '80, di implementare un controllo della congestione basato solo sui TO e che prescinda da ICMP. Il modulo ICMP, tuttora implementato nei sistemi terminali e nei sistemi di interconnessione, continua comunque a svolgere le altre sue funzioni.

Nelle implementazioni attuali si considera quindi lo scadere di un TO come un sintomo di congestione delle risorse di interconnessione e si usano nuovi algoritmi per porre rimedio a tali situazioni. Ad esempio, l'algoritmo CUTE (Congestion control Using Time-outs of the End-to-end layer) fissa il valore della finestra a disposizione del mittente, non uguale al valore comunicato dal destinatario (advertised window), ma variabile tra un minimo e un massimo. Tale algoritmo utilizza i seguenti parametri e modalità di funzionamento:

- *massimo*: è il valore massimo della finestra in trasmissione: in generale è uguale al valore della larghezza di finestra offerta dal destinatario;
- *minimo*: è il valore minimo della finestra: tipicamente uguale alla dimensione di una MSS (cfr. § VI.5.2);
- *inizializzazione*: rappresenta il valore iniziale della finestra: su reti molto cariche è preferibile partire dal valore minimo;
- *incremento*: si incrementa la dimensione della finestra, di un valore uguale ad una TCP-PDU, ogni *N* TCP-PDU ricevute correttamente dall'entità TCP di destinazione (senza però mai superare il valore massimo);
- *decremento*: si decrementa la dimensione della finestra ogni volta che scade un TO; il decremento può essere di diversi tipi:
  - *sudden* (improvviso) la larghezza della finestra è posta uguale al valore minimo;
  - *gradual* (graduale) la larghezza della finestra viene ridotta di una TCP-PDU;
  - *binary* (binario) la larghezza della finestra viene divisa a metà.

Normalmente i parametri vengono scelti in modo che l'incremento della larghezza della finestra sia lento ed il suo decremento veloce, per evitare problemi di instabilità.

I.5.6. Estensioni di TCP per le applicazioni in reti ad alta velocità

Le estensioni che sono state proposte per TCP sono funzionali ad un suo migliore comportamento in reti ad alta velocità. TCP, infatti, così come si presenta nella sua implementazione tradizionale, non offre prestazioni ottimali. In quest'ottica sono state studiate alcune modifiche orientate a migliorare l'efficienza del protocollo in questo nuovo scenario.

Innanzitutto viene osservato che in Internet il prodotto della capacità di trasferimento per il ritardo di propagazione tende ad aumentare. Quindi un primo passo per migliorare le prestazioni è quello di offrire al protocollo la possibilità di operare con una finestra massima più ampia di quella che ha ora a disposizione; i 16 bit del campo "Window" attualmente limitano l'ampiezza della finestra a 65.536 ottetti mentre, per aumentare la portata dei dati, sarebbe opportuno che tale campo

avesse una dimensione di 32 bit. A questo scopo è stata introdotta la "*Window Scale Option*", che permette a due entità TCP che vogliano instaurare una connessione, di accordarsi, durante la fase di instaurazione, sulla dimensione massima del campo “Window”.

Il più rilevante limite delle implementazioni tradizionali di TCP risiede, però, nella scarsa accuratezza con cui viene stimato il RTD delle TCP-PDU emesse. In certi casi l'aggiornamento della stima si basa su una sola TCP-PDU per finestra, creando quindi notevoli problemi di "aliasing" dovuti alla stima ed alla incapacità di adattarsi a rapidi cambiamenti nell'andamento del traffico. L'inadeguatezza di questo metodo cresce inoltre al crescere delle dimensioni della larghezza della finestra disponibile. D'altra parte, anche se si aumenta la frequenza di campionamento, rimane sempre il problema della incapacità a stimare il RTD, nel caso in cui avvengano riemissioni di una TCP-PDU. Questo perché quando si riceve il riscontro di una TCP-PDU riemessa non si è in grado di capire se tale riscontro si riferisca alla TCP-PDU originaria o a quella riemessa.

Una soluzione a questo problema viene offerta dalla "Timestamp Option". Usando tale opzione il mittente scrive l'istante di partenza su *ogni* TCP-PDU emessa in modo tale che il destinatario, quando ne invia il riscontro, possa scrivere sulla TCP-PDU di risposta l'istante di partenza della TCP-PDU a cui il riscontro stesso si riferisce. In emissione basterà operare una semplice differenza per avere un’accurata misurazione del RTD.

Un'ulteriore limitazione alle prestazioni del TCP risiede nel fatto che il destinatario non ha la possibilità di informare il mittente della eventuale corretta ricezione di TCP-PDU che siano giunte a destinazione fuori sequenza. Per superare tale problema è stata definita la opzione SACK (Selective Acknowledgement), che consente di confermare la corretta ricezione di un *particolare* TCP-PDU (invece che procedere in modo cumulativo) e quindi consente di attuare un meccanismo di ri-emissione selettiva.

Il problema principale che le nuove implementazioni di TCP incontreranno sarà quello relativo alla compatibilità nelle interazioni con le versioni tradizionali. Tutte le implementazioni dovrebbero ignorare le opzioni sconosciute che vengono inserite nelle TCP-PDU *SYN* utilizzate durante l’instaurazione della connessione. D'altra parte è possibile che talune implementazioni di TCP siano luogo a malfunzionamenti nel momento in cui ricevono una TCP-PDU, diversa dal *SYN*, contenente opzioni sconosciute. La soluzione prospettata è quindi quella di usufruire delle estensioni al protocollo, in TCP-PDU diverse dal *SYN*, solo se lo scambio di opzioni, durante la fase di instaurazione, ha indicato che entrambe le implementazioni sono in grado di comprendere le estensioni stesse.

I.6. Un protocollo di gestione

SNMP (Simple Network Management Protocol) è un protocollo di gestione; fornisce informazioni relative a configurazioni, errori e allarmi di sistemi remoti. Come il nome suggerisce, è uno strumento abbastanza semplice per la gestione di reti, che si appoggia su un limitato "*Management Information Base*" (MIB) standard per la descrizione delle risorse gestite. Il MIB specifica quali sono le informazioni che un sistema deve contenere per fini gestionali e l’uso che se ne può fare. Ad esempio MIB specifica che si deve mantenere un contatore degli ottetti ricevuti su ogni interfaccia di rete; altre variabili previste da MIB possono essere il tempo dal quale un sistema è in funzione, il numero delle IP-PDU ricevute o emesse, i diversi TO usati dai protocolli etc. MIB classifica queste variabili in 8 categorie (sistema, interfacce, indirizzi, ip, icmp, tcp, udp, egp). SNMP è facilmente implementabile e consente la gestione delle variabili descritte nel MIB, e quindi delle risorse, e la gestione di situazioni anomale mediante segnalazioni non sollecitate.

Il maggior vantaggio di SNMP risiede proprio nella sua semplicità che permette una facile ingegnerizzazione di rete con modesti consumi di risorse di elaborazione e di memorizzazione.

Inoltre, la struttura del protocollo e del MIB sono sufficientemente semplici per garantire l'inter-operabilità tra stazioni di gestione e sistemi controllati provenienti da diversi costruttori.

A fronte di tale semplicità, però, la piattaforma gestionale contrappone un limitato insieme di primitive ed una limitata sicurezza delle informazioni scambiate tra stazione di controllo e sistemi controllati.

I.7. Accesso ad Internet tramite un ISP

A conclusione della descrizione delle modalità di funzionamento di Internet, riteniamo utile riprendere quanto detto in § I.1.1, circa l'accesso ad Internet tramite un ISP (Internet Service Provider).

La trattazione fin qui svolta ha fatto riferimento a host *direttamente* connessi ad Internet e dotati di un proprio, stabile ed univoco indirizzo. Questo non è il caso di host che si connettono ad Internet per il tramite di un ISP, usando tipicamente la rete telefonica commutata. A tali host *non* è stato assegnato un indirizzo IP stabile ed univoco secondo le regole descritte in precedenza. In particolare, quando uno di questi host si connette ad Internet tramite un ISP, prima di poter iniziare effettivamente uno scambio informativo con altri host, riceve dall'ISP opportune informazioni di configurazione, tra cui un indirizzo IP, l'indirizzo del router di default (cfr. § I.3.5.3) e quello del name-server (cfr. § I.3.7).

L'indirizzo IP ricevuto può essere assegnato dall'ISP secondo due modalità:

- ✓ l'ISP dispone di un certo insieme di indirizzi IP univoci e con significato globale che assegna dinamicamente a quegli host che in un dato momento gli richiedono di connettersi ad Internet; ognuno di questi host è caratterizzato da un dato indirizzo IP solo per il tempo durante il quale è connesso ad Internet; quando l'host termina di usufruire dei servizi Internet, quell'indirizzo IP sarà assegnato ad un altro host che, in un tempo successivo, chiederà di connettersi ad Internet;
- ✓ l'ISP assegna ai suoi utenti indirizzi IP che hanno solo un significato locale e, non essendo visibili all'esterno, possono essere scelti in modo arbitrario; sarà poi compito dell'ISP porre in corrispondenza tali indirizzi con specifiche sessioni di comunicazione.

Ciò significa che un utente di questo tipo è raggiungibile dall'esterno solo tramite il suo ISP, dal momento che l'utente in questione, non disponendo tipicamente di un indirizzo stabile ed univoco in Internet non è "conosciuto" dagli altri sistemi connessi ad Internet (cfr. § I.3.4). Quando un host si connette ad Internet secondo questa modalità è necessario usare anche un ulteriore protocollo che governa lo scambio di informazioni e di comandi tra utente e ISP. Un esempio di tale protocollo è *PPP* (Point to Point Protocol).

II. RETI IN AREA LOCALE

Negli ambienti di lavoro si è già da tempo affermata una capillare diffusione dei PC (Personal Computer) per coadiuvare l'esecuzione del lavoro individuale. A tale distribuzione delle risorse elaborative spesso però non corrisponde un'equivalente distribuzione di risorse di altro tipo, ugualmente necessarie all'esecuzione dei singoli compiti. Ad esempio, archivi di dati (data base) e periferiche costose (stampanti di qualità, dispositivi grafici, grosse memorie di massa, ecc.) rimangono centralizzate per ragioni di efficienza di gestione e/o di costo. Ciò rende assolutamente necessario un mezzo rapido di comunicazione che metta in grado il PC del singolo utente di reperire, da un lato, le informazioni necessarie al proprio lavoro e di utilizzare, dall'altro, i dispositivi che rimangono centralizzati.

Per soddisfare tali esigenze sono state sviluppate le *reti in area locale* (LAN) con il compito di interconnettere apparecchiature terminali che sono sorgenti o collettori di dati negli ambienti di ufficio. Queste apparecchiature terminali sono nel seguito indicate come "*stazioni*" in conformità alla terminologia utilizzata per le LAN. Come loro elementi distintivi, queste reti:

- hanno una *limitata estensione geografica*, con uno sviluppo lineare non superiore orientativamente ad una decina di chilometri e in generale racchiuso entro un edificio o un gruppo di edifici tra loro vicini;
- sono normalmente *infrastrutture private*, al servizio di una singola organizzazione, che normalmente ne cura anche la gestione;
- interconnettono le stazioni di loro competenza con un *unico* canale trasmissivo che è *condiviso* dalle sorgenti di informazione quando queste si trovano nello stato di trasferimento;
- utilizzano modi di trasferimento *orientati al pacchetto*;
- come conseguenza della loro limitata estensione geografica, consentono una *capacità di trasferimento anche molto elevata* (nell'intervallo orientativo tra 1 e 1.000 Mbit/s) con un *basso tasso di errore* (minore orientativamente di $10^{-6} \div 10^{-7}$).

Quest'ultima caratteristica consente rapidi scambi informativi tra macchine cooperanti e quindi non introduce limitazioni nella velocità elaborativa del sistema distribuito.

Da un punto di vista logico, il canale trasmissivo comune supporta un unico *mezzo di comunicazione* che costituisce la via di percorrenza dell'informazione per il trasporto da una stazione all'altra. Nel seguito questa via sarà chiamata il *mezzo* della LAN. Questo mezzo supporta un trasferimento di tipo *multiplato dinamicamente*: cioè ad una stazione che, in un certo istante, abbia una *unità informativa* (UI) da emettere, viene assegnata l'intera capacità di trasferimento del mezzo per una durata corrispondente allo svolgimento dell'operazione di trasporto.

Nelle LAN oggi disponibili sul mercato il mezzo è normalmente condiviso con una *multiplazione a divisione di tempo*. Non sono tuttavia da escludere, in prospettiva, soluzioni alternative che però, per brevità, non verranno ulteriormente considerate nel seguito di questa trattazione.

La multiplazione dinamica a divisione di tempo implica che tra le stazioni possano determinarsi *situazioni di contesa*: ciò si verifica quando la richiesta di accesso al mezzo presentata da due o più stazioni comporterebbe, se soddisfatta immediatamente, l'impegno della risorsa comune in intervalli di tempo sovrapposti. Per risolvere queste situazioni di contesa, considerazioni di economia di sistema hanno guidato verso l'adozione di un *controllo distribuito*, e cioè verso una cooperazione tra le stesse stazioni e senza la necessità di una unità di controllo centralizzato.

Questa soluzione deriva da quella adottata per l'interconnessione tra microprocessori in un ambiente multiprocessore. In tali sistemi è spesso presente un mezzo di comunicazione ad alta velocità (bus) che interconnette i vari processori, a cui questi accedono per lo scambio dei dati. È evidente tuttavia che il grado di integrazione delle comunicazioni raggiungibile in un ambiente multiprocessore non è pensabile in una LAN che, anzi, ha lo scopo di permettere la comunicazione anche tra dispositivi eterogenei.

La soluzione a controllo distribuito, se da un lato aumenta la complessità dell'interfaccia tra ogni stazione e il mezzo, dall'altro consente un *elevato grado di flessibilità* e un *aumento dell'affidabilità* di tutto il sistema per l'eliminazione di elementi centralizzati. Si osserva che tali requisiti sono di fondamentale importanza in un ambiente, quale quello di un ufficio, per sua natura estremamente variabile, sia per il numero che per la dislocazione fisica delle stazioni.

Infatti un architettura a controllo distribuito è intrinsecamente più affidabile rispetto ad una soluzione analoga a controllo centralizzato. In questo caso, un guasto nell'unità centrale di controllo determina, nella maggior parte dei casi, un periodo di fuori servizio dell'intera rete. Ciò costringe spesso a replicare tale unità per ridurre la probabilità di fuori servizio. Invece, nelle architetture a controllo distribuito, il guasto di una o più stazioni non compromette il funzionamento della rete, che può continuare ad offrire il proprio servizio alle stazioni ancora funzionanti.

La modalità di condivisione ora introdotta, basata su una multiplazione in cui l'accesso al mezzo è regolato tramite un controllo distribuito, è denominata *accesso multiplo*, mentre le procedure di cooperazione atte a risolvere le situazioni di contesa in questo ambiente di comunicazione prendono il nome di protocolli di *controllo di accesso al mezzo* (MAC, Medium Access Control).

Come già sottolineato, le LAN sono nate e si sono sviluppate negli ambienti di lavoro come reti dedicate a servizi di comunicazione di dati. In questo quadro sono inseribili le *LAN di PC* a cui si è già accennato: normalmente la loro capacità di trasferimento è medio-bassa (orientativamente nell'intorno dei 10 Mbit/s) e l'interconnessione con una WAN è oggi principalmente richiesta per l'accesso a Internet. Le applicazioni più frequentemente utilizzate sono basate su architetture client-server.

Rispetto a questa prima fase di sviluppo, tuttora riscontrabile in molti ambienti di ufficio, il ventaglio di impieghi delle LAN si è successivamente allargato, ad esempio con :

- strutture più complesse, in cui più LAN a basso costo e a bassa capacità (*LAN di piano*) sono interconnesse da una LAN ad alta capacità (*LAN dorsale*);
- il sostegno di *comunicazioni multimediali.*

In entrambi questi casi e in altri di cui è sufficiente menzionare l'esistenza ("backend network"," storage area network"), è richiesto un aumento della capacità di trasferimento con velocità dell'ordine dei 100 Mbit/s. In alcuni casi sono anche richieste :

- una estensione della copertura del territorio con un più elevato numero di stazioni aventi accesso alla LAN;
- una più elevata affidabilità, dato il danno che deriverebbe da un'eventuale guasto nei confronti di una vasta popolazione di utenti.

Una risposta a questo insieme di requisiti si è avuta con le *reti in area metropolitana* (MAN, Metropolitan Area Network), nate per offrire *integrazione dei servizi a larga banda* , con qualità garantita anche per le applicazioni conversazionali che richiedano trasparenza temporale. Come lo stesso nome suggerisce, una MAN è una rete che è concepita per interconnettere stazioni distribuite su un area geografica corrispondente ad un quartiere cittadino o ad un'intera città. Inoltre una MAN usa una tecnologia simile a quella delle LAN (ad es. un canale trasmissivo condiviso, un mezzo multiaccesso, ecc.). È inoltre caratterizzata da un aumento delle capacità di trasferimento; si raggiungono infatti, mediante l'uso delle fibre ottiche, capacità superiori a 100 Mbit/s.

Un esempio di MAN è costituito dalle reti via cavo per la distribuzione del segnale televisivo (Cable Television - CATV). Tali reti interconnettono una centrale locale, posta a livello di quartiere o di intera città, con le singole postazioni d'utente. Data la particolarità del servizio supportato, la trasmissione è analogica ed è di tipo essenzialmente diffusivo e unidirezionale, mentre la topologia è ad albero.

Una MAN può essere di tipo sia privato che pubblico. Nel primo caso, la sua funzionalità è del tutto simile ad una LAN, in cui si ha la possibilità di estendere la connettività a sedi della stessa organizzazione geograficamente distanti. In questo caso, spesso la MAN viene indicata con il termine *LAN ad alta velocità* (HSLAN, High Speed LAN).

Se la MAN è invece di tipo pubblico, essa è gestita da un'unica organizzazione, ma è utilizzata da utenti di tipo diverso. In questo caso la MAN si configura come una rete di interconnessione per un utenza localizzata di tipo particolare, avente esigenze di comunicazione sofisticate, quali, ad esempio, il trasferimento di dati ad alta velocità o l'interconnessione di LAN. Essa inoltre può funzionare come rete di raccolta di traffico da inoltrare verso una WAN per trasferimenti a lunga distanza.

Alle prestazioni delle MAN si ricollega l'ultima fase di sviluppo delle LAN, in cui:

- la capacità di trasferimento è stata ulteriormente aumentata fino a orientativamente 1 Gbit/s per *applicazioni fisse*, con portata media scalabile in termini sia di capacità per singola stazione, sia di capacità aggregata; ciò deve consentire la crescita del numero di stazioni servibili da alcune decine ad alcune centinaia di apparecchiature terminali con elevati ritmi binari di emissione;
- è consentita anche la fruizione di *applicazioni mobili* con la utilizzazione di *LAN senza fili* (WLAN, Wireless LAN), per le quali l'attuale stato della tecnologia offre capacità di trasferimento dell'ordine di una decina di Mbit/s.

Infine, rimanendo nel campo delle comunicazioni di dati, è significativo citare lo standard "*Fibre Channel*", che consente trasferimenti fino a centinaia di Mbit/s, con copertura di distanze dell'ordine di una decina di chilometri utilizzando una fibra monomodale e di qualche decina di metri con mezzi in rame (coppia coassiale e doppino schermato).

II.1. Topologia delle LAN

La topologia logica di una LAN, e cioè la configurazione geometrica del suo mezzo di comunicazione, dal punto di vista delle relazioni di traffico consentite, è scelta tra varie possibili soluzioni, che ben rispondono alle esigenze di una infrastruttura a basso costo. Le topologie più largamente utilizzate sono:

- il bus bidirezionale ;
- il bus unidirezionale ;
- il doppio bus unidirezionale;
- l'anello;
- la stella.

Per ciascuna di queste verranno fornite qui di seguito alcune informazioni essenziali.

II.1.1. Bus bidirezionale.

In una LAN con topologia a bus bidirezionale (Fig. II.1.1), il mezzo ha una configurazione lineare e le stazioni sono connesse ad esso in derivazione. Un esempio di prodotto commerciale con questo tipo di topologia è fornito dalla *rete Ethernet.*

L'informazione viaggia dalla stazione d'origine, in entrambe le direzioni, verso le estremità del mezzo, raggiungendo quindi tutte le stazioni connesse alla rete. La stazione di destinazione riconosce, in base all'indirizzo trasportato nell'etichetta, le UI ad essa indirizzate e ne legge il contenuto durante il loro transito attraverso l'interfaccia tra stazione e mezzo.

Da un punto di vista fisico, la topologia a bus bidirezionale ha la caratteristica fondamentale di avere le interfacce tra le stazioni ed il mezzo completamente *passive*. Questo elemento aumenta considerevolmente l'affidabilità della struttura poiché, da un lato, diminuisce la possibilità di guasti dell'interfaccia e, dall'altro, anche in presenza di guasti, il funzionamento della rete non è compromesso. È tuttavia evidente che l'assenza di unità di rigenerazione limita la distanza percorribile dal segnale a causa dell'attenuazione sul canale trasmissivo e dei relativi disturbi.

Nella LAN Ethernet, la sezione di bus senza dispositivi che operino la rigenerazione del segnale numerico è chiamata segmento. Ogni segmento ha una lunghezza massima che dipende dalle caratteristiche del mezzo trasmissivo e dal ritmo binario di trasmissione: se questo ritmo è

uguale a 10 Mbit/s e se il mezzo è la coppia coassiale “spessa” (thick), questa lunghezza massima è di 500 m, mentre se si adotta una coppia coassiale “sottile” (thin) a parità di ritmo di trasmissione, la lunghezza massima dei segmenti è di 185 m.

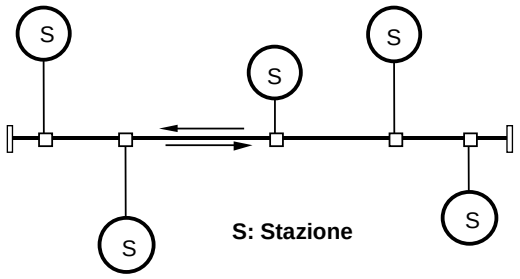


Figura II.1.1 - Topologia a bus bidirezionale.

Allo scopo di aumentare l’estensione delle LAN a bus bidirezionale, si impiegano *ripetitori*, e cioè unità in grado di rigenerare il segnale, assicurando anche il recupero della temporizzazione e la restituzione della forma agli elementi di segnale. Un ripetitore consente di connettere due segmenti e quindi di porre in comunicazione stazioni facenti capo a segmenti diversi. Caratteristica distintiva di questi dispositivi, rispetto a quelli utilizzati nei collegamenti numerici punto-punto, è la loro *bidirezionalità*. Un ripetitore è temporalmente trasparente da un punto di vista logico. Per evitare le interferenze che potrebbero derivare dalla presenza di percorsi multipli, è ammessa solo una via composta di segmenti e di ripetitori per ogni coppia di stazioni. Il grafo che ne risulta, escludendo percorsi chiusi, è un *albero* (Fig.II.1.2), in cui i rami sono in corrispondenza ai segmenti e i nodi sono rappresentativi dei ripetitori.

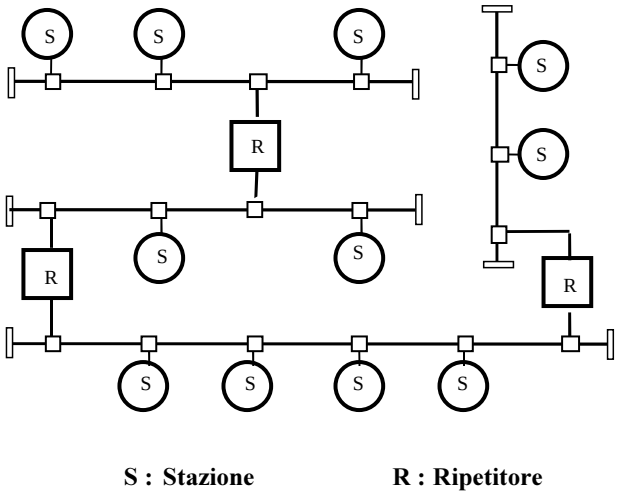


Figura II.1.2 - Topologia ad albero costituita da segmenti di bus bidirezionali interconnessi da ripetitori.

Un’ulteriore modalità di estensione delle LAN a bus bidirezionale è fornita dall’uso dei “*bridge*”, le cui caratteristiche come unità di interconnessione saranno chiarite nel seguito (cfr. §II.2.3).

È infine da osservare che, a ciascuna delle estremità di ogni segmento di una LAN con topologia a bus o a albero, deve essere previsto un *terminatore* che, chiudendo il mezzo trasmissivo in modo adattato, evita la riflessione dei segnali che viaggiano nelle due direzioni del segmento rispetto alla stazione di origine. Da un punto di vista logico, ciò consente di estrarre dal bus

l’informazione che ha raggiunto tutte le stazioni connesse al segmento e, in particolare, quella (o quelle) di destinazione.

II.1.2. Bus unidirezionale.

La topologia a bus unidirezionale si differenzia dalla precedente per il fatto che l’informazione fluisce sul bus esclusivamente in un verso. Affinché sia assicurata la completa connettività tra le stazioni, occorre quindi suddividere il bus in due sezioni, una *di immissione* e l'altra *di estrazione*. Ogni interfaccia tra stazione e mezzo deve riguardare ambedue le sezioni e risulta quindi funzionalmente composta da due unità distinte: una *di scrittura* e l'altra *di lettura*.

Le due alternative possibili per la creazione delle sezioni di immissione e di estrazione su un unico bus sono illustrate in Fig. II.1.3. La prima alternativa (Fig. II.1.3.a) è un *bus unidirezionale ripiegato*, mentre la seconda (Fig. II.1.3.b) è un *bus unidirezionale doppiamente ripiegato*. La differenza sostanziale tra le due alternative è che i versi di attraversamento delle stazioni da parte delle sezioni di immissione e di estrazione sono diversi nel caso del bus ripiegato, mentre sono identici nel caso del bus doppiamente ripiegato.

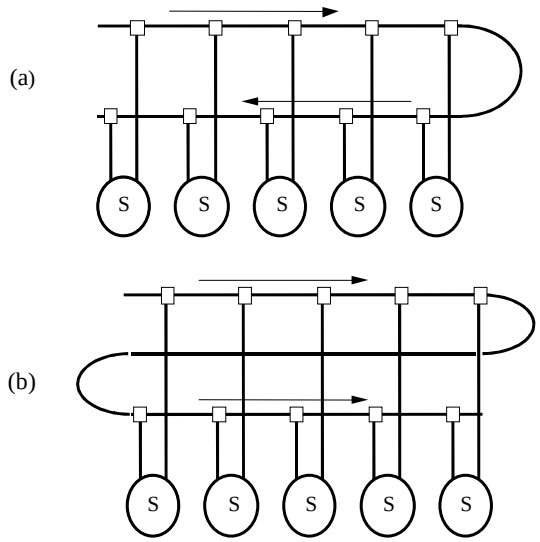


Figura V.1.3 – Topologie a bus unidirezionale: (a) bus unidirezionale ripiegato; (b) bus unidirezionale doppiamente ripiegato

In ogni caso l’ordine di collocazione delle stazioni sulle due sezioni del bus nel loro verso di percorrenza individua i ruoli di:

- *stazioni di testa*, che è a monte di tutte le altre stazioni sulla sezione di immissione e che ha il compito di governare la riconfigurazione della LAN;
- *stazione di terminazione del bus*, che è a valle di tutte le altre sulla sezione di estrazione e che ha il compito di “assorbire” l’informazione dal bus dopo che questa ha completato il suo percorso sulle due sezioni della rete.

Inoltre il passaggio dalla sezione di immissione a quella di estrazione è assicurato da una stazione che, collocandosi a valle delle altre sulla sezione di immissione e a monte sulla sezione di estrazione, svolge la funzione di ripiegatura del bus.

È da osservare che, considerando il verso dei flussi informativi sul mezzo, in entrambe queste topologie una generica stazione è in grado di ricevere l’informazione emessa dalle stazioni "a monte" su entrambe le sezioni del bus, mentre le informazioni emesse dalle stazioni "a valle" sono ricevute solo sulla sezione di estrazione.

Da un punto di vista fisico, l'unidirezionalità della propagazione dei segnali consente di superare i problemi di attenuazione del mezzo trasmissivo. È infatti possibile utilizzare dispositivi

di amplificazione e/o di rigenerazione che, a causa della loro intrinseca unidirezionalità, non possono essere utilizzati nel caso della topologia a bus bidirezionale. In questo modo l'estensione massima di ogni sezione di rete può aumentare considerevolmente.

Un ulteriore elemento a favore della topologia a bus unidirezionale consiste nella sua affidabilità. Infatti, tramite un'opportuna *procedura di riconfigurazione*, le reti con questa topologia possono mantenere la piena connettività anche in presenza di un guasto su una tratta del bus.

In condizioni normali, la rete ha la configurazione ad anello fisico mostrato in Fig. II.1.4a, in cui la stazione di testa assolve anche le funzioni di terminazioni e di ripiegatura del bus. Se avviene un guasto, interviene una procedura di riconfigurazione, che isola la tratta di bus guasta (Fig. II.1.4b), attribuendo diversamente le funzioni di terminazioni e di ripiegatura del bus.

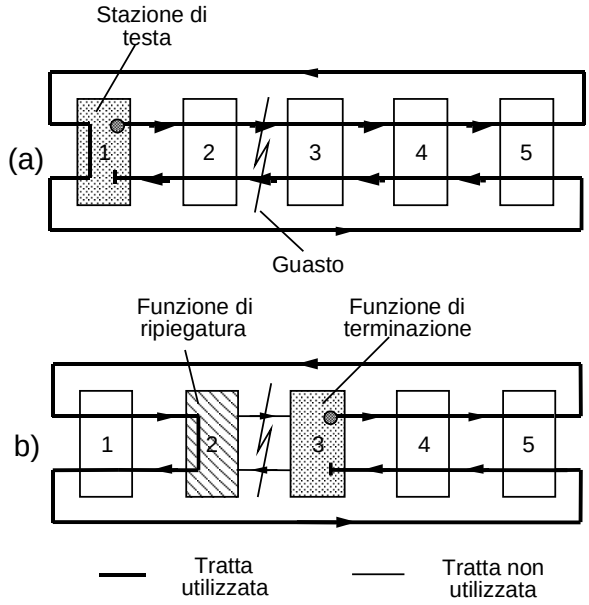


Figura II.2.4 - Riconfigurazione, in caso di guasto, di una rete a bus unidirezionale: (a) funzionamento normale; (b) rete riconfigurata.

La procedura può essere così schematizzata. Un guasto su una tratta del bus viene rivelato dalla stazione a valle della tratta guastatasi: ciò a causa della mancanza di segnale ricevuto. Quando ciò avviene, tale stazione assume il ruolo di testa, svolgendo anche la funzione di termnazione; in questa sua nuova veste emette un segnale di controllo per avvertire le altre stazioni dell'avvio della procedura di riconfigurazione.

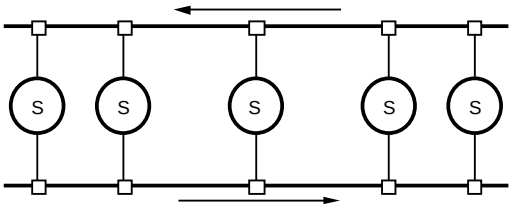
Alla ricezione di questo segnale, la stazione che precedentemente era di testa assume la configurazione normale, mentre la stazione a monte della tratta guasta svolge la funzione di ripiegatura. Al termine della procedura, la rete assicura nuovamente la piena connettività tra le stazioni di rete. Quando il guasto sarà stato riparato, si potrà, per iniziativa della stazione di testa, tornare alla configurazione fisica iniziale.

Come è evidente, condizione necessaria per realizzare questa riconfigurazione è che tutte le stazioni di rete siano in grado di assumere, se richiesto, il ruolo di testa e di svolgere le funzioni di terminazione e di ripiegatura del bus. Ciò aumenta la complessità delle stazioni e quindi il loro costo.

II.1.3. Doppio bus unidirezionale.

La topologia a doppio bus unidirezionale (Fig. II.1.5) è un'evoluzione di quella a bus unidirezionale ripiegato. In questo caso le due sezioni corrispondenti ai due versi di trasferimento sono completamente separate. Una sezione attraversa così le stazioni in un verso, mentre l'altra le

attraversa in verso opposto. Ovviamente, per utilizzare la rete al meglio delle sue capacità, una stazione d'origine dovrà effettuare l'emissione di una UI solo su quella sezione che consente il trasferimento verso la stazione di destinazione. A tale scopo è necessario che la stazione d'origine conosca la posizione di quella di destinazione.



S: Stazione

Figura II.1.5 - Topologia a doppio bus unidirezionale.

Per ciascuna delle due sezioni, esiste una stazione che è a monte di tutte le altre nel verso di trasferimento del segnale: questa è la stazione di testa di quella sezione; esiste però anche una stazione che, sempre nello stesso verso, è a valle di tutte le altre e, come tale, svolge la funzione di terminazione.

Nel caso in cui non si ritenga necessario ottenere la massima efficienza dalla rete, il modo più semplice di funzionamento è, naturalmente, quello di consentire l'emissione di una UI su entrambi i bus. Per evitare l'uso di entrambi i bus, occorre invece che ogni stazione abbia a disposizione una tabella nella quale siano memorizzate le posizioni relative di tutte le altre stazioni connesse alla rete. La gestione di queste tabelle è un elemento molto importante per il corretto funzionamento della rete. In particolare, queste tabelle devono essere aggiornate quando una nuova stazione è connessa alla rete, ma anche quando una stazione viene spostata all'interno della rete stessa. A tale scopo deve essere prevista una procedura di controllo specializzata.

Le interfacce di una stazione su entrambe le sezioni sono del tutto identiche e devono consentire operazioni sia di scrittura che di lettura sul mezzo di comunicazione. Inoltre la gestione dell'accesso alle due sezioni è effettuata in modo completamente indipendente. L'intera rete è quindi costituita da due bus unidirezionali controllati in modo indipendente. Ciò evidentemente comporta una complessità doppia dell'interfaccia tra stazione e mezzo di comunicazione rispetto a quella che si avrebbe utilizzando una rete a sezione singola.

I vantaggi della topologia a doppio bus consistono essenzialmente nella sua affidabilità, nella possibilità di utilizzare pienamente le potenzialità delle fibre ottiche e nel fatto che l'effettiva capacità a disposizione delle stazioni è doppia rispetto a quella raggiungibile con reti a sezione singola. Ciò è evidentemente dovuto al fatto che si ha una sezione dedicata per ogni verso di trasferimento. Per tali ragioni questa topologia ha trovato larga applicazione specialmente nelle reti locali ad alta velocità e nelle MAN.

Anche la topologia a doppio bus può essere riconfigurata in modo da garantirne la piena connettività anche in presenza di un guasto. La configurazione normale è detta a *bus chiuso* (looped bus) (Fig. II.1.6a) ed è caratterizzata dal fatto che le funzioni di stazione di testa e di terminazione per entrambi i bus sono concentrate in un'unica stazione.

In caso di guasto, la procedura di riconfigurazione effettua sulla rete i seguenti interventi modificativi (Fig. II.1.6b) rispetto al funzionamento normale:

- i ruoli di stazione di testa per ciascuno dei due bus sono attribuiti a due stazioni diverse, quelle a valle della tratta guasta sul bus pertinente;
- la stazione, che così assume il ruolo di testa per un bus, diventa anche di terminazione per l’altro bus;
- la rete assume una configurazione a *bus aperto*, pur conservando, come nel caso di funzionamento a bus chiuso, la piena connettività tra le stazioni della LAN.

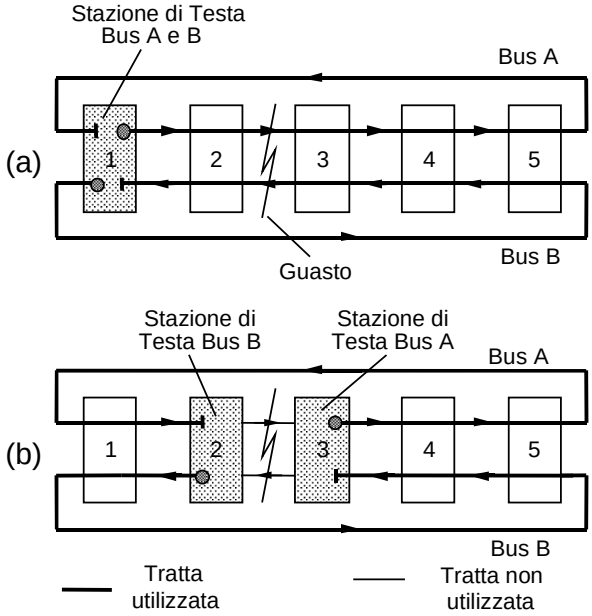


Figura II.1.6 - Riconfigurazione, in caso di guasto, di una rete a doppio bus unidirezionale: (a) funzionamento normale; (b) rete riconfigurata.

II.1.4. Anello

Nella topologia ad anello, ogni stazione o più precisamente la sua *interfaccia* verso la rete è connessa a due sole altre interfacce, una "a monte" e l'altra "a valle", tramite legamenti unidirezionali, che complessivamente formano un percorso chiuso (Fig. II.1.7). Le informazioni sono trasferite in modo sequenziale da una interfaccia alla successiva. Ogni stazione dell'anello può essere, oltre che origine e destinazione, anche punto di transito delle UI; in quest'ultimo caso, da un punto di vista trasmissivo, l'interfaccia della stazione agisce come elemento attivo della rete operando una rigenerazione del segnale.

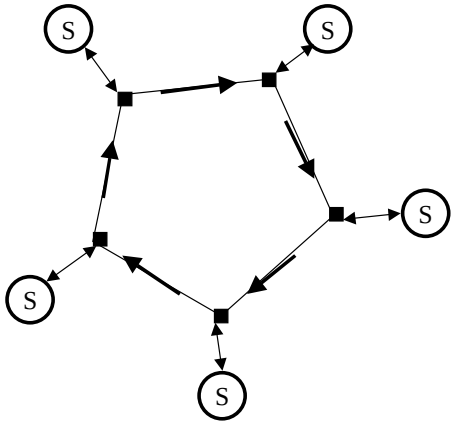


Figura II.1.7 - Topologia ad anello.

Oltre ad agire fisicamente come rigeneratore, l'interfaccia di una stazione deve svolgere l'*inserimento* delle UI nell'anello (quando la stazione è di origine), la loro *ricezione* (quando la stazione è di destinazione) e infine la loro *rimozione.* Mentre le prime due funzioni sono analoghe a quelle svolte in una LAN con topologia a bus, la rimozione richiede qualche maggiore attenzione.

Infatti, come si è già visto, nelle LAN con topologia a bus le nformazioni che sono trasferite sul mezzo vengono assorbite dai terminatori posti alle estremità del bus. Invece nelle LAN ad

anello, in mancanza di uno specifico provvedimento, le UI circolerebbero indefinitivamente sul mezzo. È quindi necessario che una delle stazioni dell'anello, qualla di origine o quella di destinazione, provveda a rimuovere ogni UI quando si raggiunga la ragionevole aspettativa di sua consegna a destinazione.

L'operazione di rimozione di una UI richiede che la stazione a questa preposta sia in grado di leggere il proprio indirizzo tra quelli contenuti nella UI: cioè, se la stazione rimuovente è quella di origine o quella di destinazione, l'indirizzo da leggere è quello di origine o di destinazione, rispettivamente.

Per consentire queste operazioni di lettura e per svolgere altre funzioni che verranno descritte nel seguito, l'interfaccia di ogni stazione deve introdurre, sul flusso di cifre binarie che la attraversa verso l'interfaccia successiva lungo l'anello, un ritardo almeno uguale a un tempo di bit. Questo ritardo (*latenza della stazione*) va aggiunto a quello di propagazione fisica dei segnali lungo l'anello quando si voglia valutare il ritardo che una cifra binaria impiega per percorrere l'anello nella sua interezza.

Rispetto a una topologia a bus, bidirezionale o unidirezionale, una tipologia ad anello, tramite la presenza di interfacce attive, elimina le limitazioni all'estensione massima della rete; diminuisce però le prestazioni in termini di affidabilità, comportando anche un aggravio per ciò che riguarda il costo di installazione.

Per contenere lo svantaggio sul livello di affidabilità , spesso si adottano strutture ad *anello duplicato,* che consentono di riconfigurare la rete in modo da isolare l'eventuale segmento guasto.

In conclusione, la topologia ad anello può essere vantaggiosamente utilizzata laddove occorra connettere stazioni localizzate in un'area estesa che richiedono elevate capacità. Ciò si ottiene però al prezzo di una struttura più complessa per ciò che riguarda la gestione dei guasti e delle riconfigurazioni.

La struttura ad anello duplicato più largamente utilizzata è quella mostrata in Fig. II.1.8a, che prevede due anelli con versi di trasferimento opposti (*counter-rotate ring*). In condizioni normali di funzionamento la presenza di due anelli distinti raddoppia la capacità della rete. In presenza di un guasto, il segmento guasto viene isolato poiché le stazioni a destra e a sinistra di questo provvedono alla connessione tra i due anelli e quindi a formare un unico anello (Fig. II.1.8b). In questo caso però, a differenza delle riconfigurazioni descritte nelle Figg. II.1.4b e II.1.6b, la capacità di trasferimento della rete diventa uguale a quella di un unico anello.

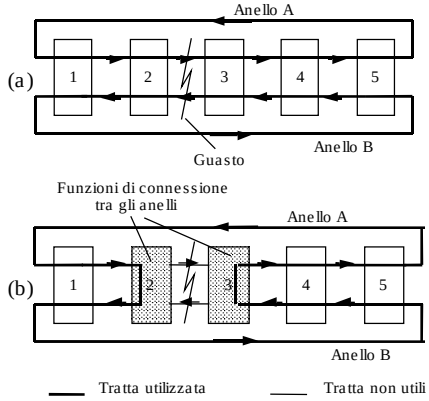


Figura II.1.8 - Riconfigurazione, in caso di guasto, di una rete ad anello: (a) funzionamento normale; (b) rete riconfigurata.

Un ulteriore per migliorare l'affidabilità consiste nel fare assumere al mezzo trasmissivo una configurazione a stella, come mostrato in Fig. II.1.9. In tal modo ogni interfaccia è connessa a un sito comune, che è collocato in posizione centrale rispetto alla localizzazione delle stazioni . È allora più facile individuare una interfaccia guasta e provvedere alla sua eliminazione agendo direttamente dalla posizione centrale.

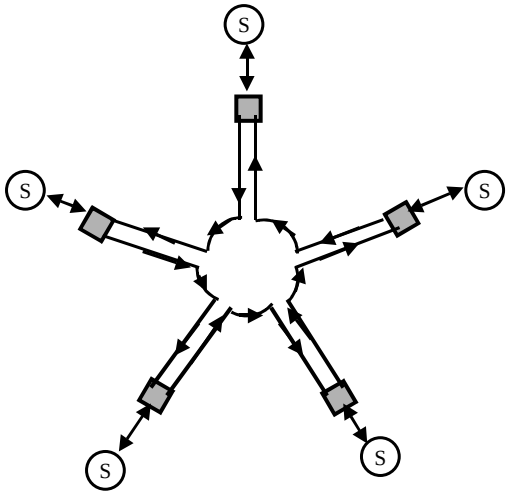


Figura II.1.9 - Topologia logica ad anello realizzata con topologia fisica a stella

II.1.5. Stella

Nella topologia a stella tutte le stazioni sono connesse con un ramo individuale a un nodo comune (Fig. II.1.10). Ogni ramo è rappresentativo di due canali trasmissivi punto-punto: uno per la trasmissione sostiene un *collegamento uscente* dal nodo, l'altro per la ricezione è in corrispondenza con un *collegamento entrante* nel nodo. Circa la funzione svolta dal nodo comune (hub) si distinguono due alternative.

Nella prima di queste il nodo è rappresentativo di un dispositivo che opera la rigenerazione del segnale e che mantiene la condivisione del mezzo: ricevendo una UI su uno dei collegamenti entranti, il nodo la riemette senza ritardo in modalità diffusiva su tutti i collegamenti uscenti. Inoltre una emissione contemporanea di UI da parte di due o più stazioni determina interferenza all'interno del nodo. In tal modo

- la UI emessa da una stazione viene ricevuta da tutte le altre stazioni come avviene nella topologia a bus bidirezionale ;
- volendo evitare interferenza tra le UI emesse da stazioni diverse, solo una stazione alla volta può riuscire a emettere.

Si conclude che, con questo tipo di nodo ripetitore-diffusore, la rete logica è topologicamente un bus bidirezionale anche se la rete fisica ha topologia a stella.

Nella seconda alternativa, il nodo comune è rappresentativo di un *commutatore*, che opera con attraversamento adimmagazinamento e rilancio. In tal modo:

- una UI, che è emessa da una stazione e che è ricevuta dal nodo su un collegamento entrante, viene memorizzata e poi rilanciata su un singolo collegamento uscente verso la sua destinazione;
- gli altri collegamenti uscenti non impegnati in questo trasferimento possono essere utilizzati per differenti relazioni di traffico (ad es. al servizio di un'altra coppia di stazioni, se la relativa comunicazione è punto-punto).

Con la topologia a stella sono anche modellabili le reti locali che utilizzano il nodo come comutatore a circuito (PABX, Private Automatic Branch Exchange) e che operano secondo il modo di trasferimento a circuito. Normalmente i PABX, appartenenti alla generazioni di apparati concepite nell'ultimo quarto del secolo scorso, consentono, oltre all'espletamento del servizio telefonico, anche l'effettuazione di trasferimenti di dati tra le varie terminazioni. Tuttavia, poiché il principio di funzionamento di tali sistemi è quello a circuito, la loro utilizzazione come supporto di comunicazione in un ambiente di informatica distribuita non è, in generale, conveniente. Infatti, occorre considerare che la complessità delle procedure di instaurazione di una connessione tra due

terminazioni è notevolmente maggiore rispetto a quella prevista in una LAN. Inoltre il tipo di traffico generato da un terminale di dati è tale da condurre ad una sensibile sotto-utilizzazione dei canali di comunicazione.

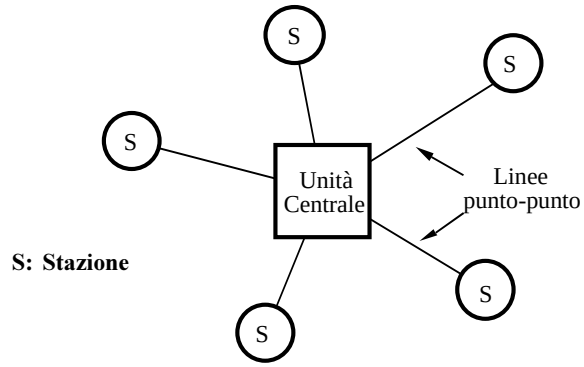


Figura II.1.30 - Topologia a stella.

II.2. Architetture di accesso a una LAN

In Fig. II.2.1 è mostrato il modello architetturale di una stazione connessa ad una LAN. Tale modello viene posto a confronto con quello di un apparecchio terminale che sia conforme al modello OSI. Da questo confronto appare che lo strato di collegamento del modello OSI è sdoppiato, nel caso di stazione connessa ad una LAN, in due sotto-strati: quello *di controllo del collegamento logico* (Logical Link Control-LLC) e quello di *controllo di accesso al mezzo* (Medium Access Control-MAC). Nel seguito, per brevità, questi due sotto-strati saranno indicati come *strato LLC* e *strato MAC*.

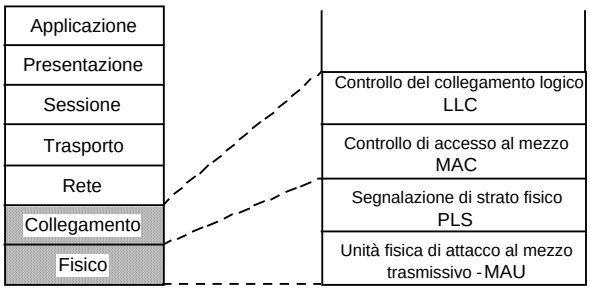


Figura II.2.1 – Architettura stratificata di una stazione connessa ad una LAN e confronto con quella di un apparecchio terminale conforme al modello OSI

Lo strato LLC ha il compito di controllare lo scambio delle unità di dati tra i terminali connessi ad una LAN. In particolare ha lo scopo di rendere lo scambio informativo efficiente ed esente da errori.

Lo strato MAC ha invece la funzione di regolare l'accesso al mezzo, risolvendo eventuali contese di utilizzazione tra le stazioni. Lo svolgimento di questa funzione è affidato al già citato protocollo di strato, e cioè al *protocollo MAC*, che è uno degli elementi caratteristici di una LAN.

Lo strato fisico è a sua volta suddiviso in due sotto-strati, uno logico e l'altro fisico. Il primo è il sotto-strato di *segnalazione di strato fisico* (Physical Layer Signaling - PLS), mentre il secondo è l'*unità di attacco al mezzo trasmissivo* (Medium Attachment Unit - MAU). Compito del sotto-strato PLS è mettere in grado le entità MAC di inviare le loro unità di dati sul canale trasmissivo usufruendo del supporto di opportuni segnali. La MAU consente invece l'accoppiamento tra il sotto-

strato PLS ed il mezzo di comunicazione: quindi emette e riceve questi segnali sotto il comando del sotto-strato PLS.

Sebbene in gran parte le LAN siano state originariamente definite e realizzate in ambienti privati e siano quindi basate su architetture e protocolli di tipo proprietario, varie organizzazioni internazionali hanno provveduto a emettere norme per una definizione univoca dei protocolli relativi ai vari strati funzionali di una LAN. Attualmente, il riferimento in questo campo è costituito dalla famiglia di norme emesse dall'IEEE (Institution of Electrical and Electronics Engineers), conosciute sotto la sigla IEEE 802; queste norme sono poi state fatte proprie dall'ISO nella serie 8802.

Le norme della famiglia IEEE 802 trattano esclusivamente gli strati fisico, MAC e LLC. In Fig. II.2.2 è illustrata la relazione fra queste norme, i cui contenuti sono qui di seguito riassunti brevemente:

- *IEEE 802.1 (ISO 8802.1)*: introduce la struttura e le architetture dei singoli elementi della famiglia IEEE 802;
- *IEEE 802.2*: *(ISO 8802.2)*: descrive le funzioni associate allo strato LLC, specificando la sua interfaccia (SAP, Service Access Point) con lo strato superiore e con quello MAC;
- *IEEE 802.3 (ISO 8802.3)* : descrive il protocollo di accesso *CSMA/CD*, con riferimento a diversi mezzi trasmissivi ed ai ritmi binari utilizzabili, ed il relativo strato fisico;
- *IEEE 802.4: (ISO 8802.4)* specifica gli elementi riguardanti il protocollo di *accesso a testimone su bus* (token bus) ed il relativo strato fisico;
- *IEEE 802.5 (ISO 8802.5) :* specifica gli elementi riguardanti il protocollo di accesso *a testimone su anello* (token ring) ed il relativo strato fisico;
- *IEEE 802.6 (ISO 8802.6)* : descrive la MAN *DQDB*, che utilizza il protocollo d'accesso *a coda distribuita*;
- *IEEE 802.11 (ISO 8802.11)* : riguarda una LAN su supporto radio.

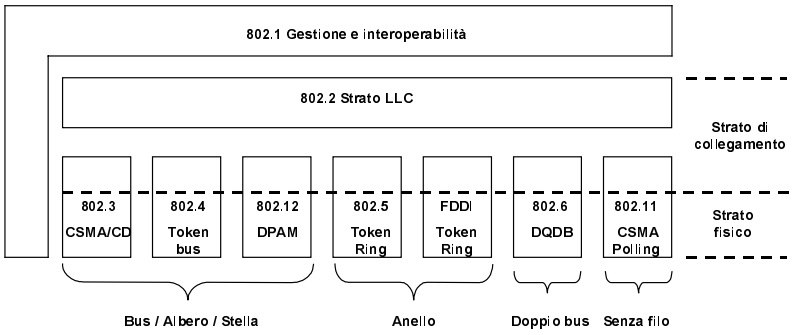


Figura II.2.2- Norme IEEE 802 e loro relazione con il modello OSI

Relativamente allo strato fisico, le due metodologie utilizzate per la trasmissione delle informazioni su un mezzo trasmissivo in rame costituente una LAN erano almeno inizialmente:

- *la trasmissione numerica in banda base*;
- *la trasmissione analogica a banda estesa.*

Nel caso di *trasmissione in banda base*, il segnale numerico in linea occupa l'intera banda del canale trasmissivo. Quindi solo una stazione alla volta può emettere le proprie UI. Si devono quindi prevedere, come già accennato, opportuni schemi di regolazione dell'accesso al mezzo per evitare la sovrapposizione dei segnali.

I vantaggi delle LAN in banda base sono la loro semplicità realizzativa ed il loro basso costo. L'installazione è rapida e semplice ed i sistemi impiegati hanno ormai una tecnologia acquisita. Occorre però tenere conto delle limitazioni riguardanti le loro possibilità di copertura nel caso in cui la rete debba gestire un traffico elevato oppure coprire superfici estese.

Eventuali limitazioni in estensione possono essere superate utilizzando dispositivi di interconnessione come i *ripetitori*, che provvedono a rigenerare il segnale in transito sul mezzo trasmissivo, oppure come i *bridge*, che hanno lo scopo di interconnettere segmenti diversi della stessa rete. Nel primo caso, le varie tratte di ripetizione formano un'unica rete; nel secondo, i vari segmenti hanno un funzionamento indipendente e spetta ai bridge di gestire il traffico che deve passare da un segmento all'altro. È evidente che i bridge, se non sono accuratamente progettati ed utilizzati, possano divenire colli di bottiglia per il traffico in transito sulla LAN, riducendo sensibilmente la portata di rete.

Nella *trasmissione a banda estesa,* che oggi è ormai abbandonata, veniva utilizzata la tecnica di multiplazione a divisione di frequenza, in modo da consentire la trasmissione simultanea da parte di più stazioni. La larghezza di banda del mezzo trasmissivo veniva suddivisa in un certo numero di sottobande, ognuna delle quali era destinata ad uno specifico servizio di comunicazione (ad es. alla telemetria, alla voce, ai dati, al video, ecc.). In questo modo, ogni sottobanda costituiva un canale indipendente, a cui una pluralità di utenti poteva accedere con le modalità precedentemente descritte.

II.3. Standard IEEE 802.3

Con riferimento allo standard IEEE 802.3, la *procedura di emissione* dello strato MAC nel protocollo CSMA/CD, è la seguente:

- accettare i dati dallo strato LLC e formare la MAC-PDU;
- presentare un flusso di dati seriale allo strato fisico per la codifica e per la successiva emissione;
- se il canale è libero, procedere *subito* all'emissione;
- se il canale è occupato, ritardare l'emissione secondo quanto indicato da una delle procedure di persistenza;
- se non si rivelano collisioni, portare a termine l'emissione;
- se è rivelata una collisione, interrompere subito l'emissione e svolgere una procedura di "*imposizione di collisione*" (collision enforcement) per segnalare l'evento alle altre stazioni;
- eseguire poi l'*algoritmo di subentro* per decidere quando deve essere riemessa la PDU andata in collisione;
- assicurare che
  - due PDU consecutive siano separate da un intervallo di durata non inferiore a un valore specificato (*tempo di intertrama*);
  - le PDU abbiano lunghezza non inferiore al valore minimo.

Per quanto riguarda la *procedura di ricezione*, questa richiede di:

- ricevere un flusso seriale di dati dallo strato fisico;
- presentare allo strato LLC le PDU indirizzate alla stazione locale.

L'algoritmo di subentro definito nello standard IEEE 802.3 è di tipo *esponenziale binario troncato* (truncated binary exponential backoff) ed ha lo scopo di determinare l'intervallo di tempo al termine del quale avviene un nuovo tentativo di emissione dopo una collisione. Assumendo come unità di tempo l'intervallo temporale corrispondente al massimo ritardo di propagazione D l'algoritmo di subentro fornisce, per la n-ma ri-emissione, un numero casuale intero uniformemente distribuito nell'intervallo $(0, 2^k)$, dove $k = [0, \min(n,10)]$.

La procedura di imposizione di collisione ha lo scopo di garantire che la durata della collisione sia sufficiente a che tutti i terminali coinvolti in essa siano in grado di rivelare la collisione. La configurazione di bit da emettere per segnalare l'avvenuta collisione non è specificata e può essere qualunque; la sua lunghezza è invece fissata in 32 bit.

In Fig. II.3.1 è riportata la struttura della MAC-PDU del protocollo CSMA/CD. In particolare il significato dei singoli campi è il seguente:

- *Preambolo:* ha lunghezza uguale a sette ottetti ed assicura la sincronizzazione della stazione con l'inizio della MAC PDU;
- *Delimitatore di inizio trama:* è una sequenza di otto bit (10101011) che indica l'inizio della MAC PDU;
- *Indirizzi di destinazione e di sorgente:* indicano rispettivamente la stazione di destinazione e di sorgente; possono avere lunghezza uguale a 16 o 48 bit;
- *Lunghezza:* indica la lunghezza (in ottetti) del campo dati di strato LLC e comprende anche la lunghezza dell'eventuale campo PAD;
- *Dati di strato LLC:* questo campo contiene i dati dello strato LLC;
- *PAD:* è inserito nel caso che la lunghezza del campo dati di strato LLC non sia sufficiente per raggiungere la lunghezza minima di PDU necessaria per assicurare la correttezza delle operazioni di riconoscimento delle collisioni;
- *FCS:* ha una lunghezza di 32 bit con l'usuale significato nelle procedure di controllo di errore (cfr. vol. I, par. V.2); è calcolato sulla base dei bit della trama compresi tra il campo delimitatore di inizio trama ed il campo FCS.

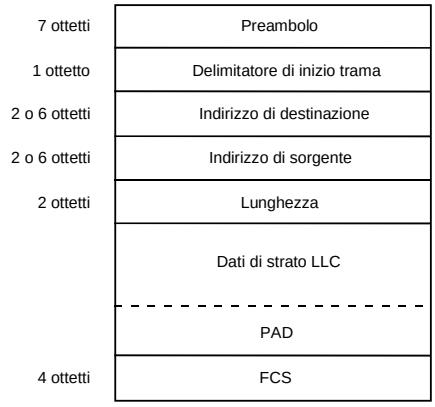


Figura II.3.1 – Formato della MAC-PDU del protocollo CSMA/CD

II.4. Standard IEEE 802.4

Il formato generale delle PDU impiegate nel protocollo a testimone sul bus è riportato in Fig. II.4.1. Più in particolare la Fig. II.4.1A presenta la MAC PDU, mentre la Fig. II.4.1B è relativa al PDU token. I campi contenuti nella MAC PDU hanno il seguente significato:

- *preambolo:* è una sequenza trasmessa prima di qualsiasi MAC PDU per sincronizzare la stazione remota; deve avere lunghezza tale da assicurare una durata di almeno 2 µs (ad esempio, per ritmi binari di 10 Mbit/s deve essere lunga almeno 3 ottetti);
- *delimitatore di inizio trama:* è un ottetto avente struttura fissa che indica l'inizio della trama e che è codificato come *NN0NN000*, dove *N* è un simbolo MAC non utilizzato per i dati e ottenuto con la tecnica della violazione del codice di Manchester utilizzato nella trasmissione;
- *controllo:* è un ottetto utilizzato per distinguere tra PDU di controllo MAC, PDU di dati di strato LLC e PDU di gestione delle stazioni;
- *indirizzi di destinazione e di sorgente:* identificano rispettivamente la stazione di destinazione e quella di sorgente; possono avere lunghezza uguale a 16 o 48 bit.
- *dati di strato LLC:* contiene i dati dello strato LLC;
- *FCS:* è lungo quattro ottetti per la rivelazione degli errori di trasmissione;

- *delimitatore di fine trama:* è un ottetto, che indica la fine della PDU e che è codificato come *NN1NN1IE*, dove *N* è lo stesso simbolo utilizzato nel delimitatore d'inizio di trama, 1 è un bit uguale a "1", *I* è il *bit intermedio* ed *E* è il *bit d'errore*.

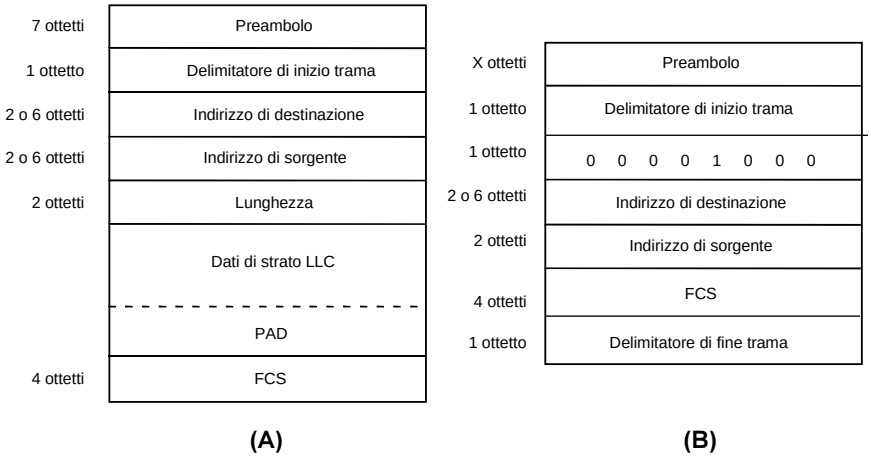


Figura II.4.1 - Strutture delle PDU del protocollo a testimone su bus: (A) MAC PDU; (B) PDU token

I bit *I* ed *E* sono gestiti dai ripetitori di rete ed hanno il seguente significato:

- il *bit I,* se posto ad 1, indica che la stazione in trasmissione deve emettere altre PDU (questo bit aiuta il ripetitore nella determinazione di che cosa segue il campo delimitatore di fine trama);
- il *bit E* è posto ad 1 dal primo ripetitore che riveli un errore nella PDU.

A differenza del protocollo CSMA/CD, nel protocollo a testimone su bus non è necessario che la PDU abbia una lunghezza minima. In questo caso, infatti, non essendo presenti collisioni, una emissione ha sempre esito positivo. La lunghezza del bus non deve essere quindi tenuta in conto. Ciò, evidentemente, aumenta l'efficienza del protocollo.

Il formato del MAC token (Fig. II.4.1B) differisce da quello di una generica MAC PDU per la mancanza del campo di dati di strato LLC.

Lo standard IEEE 802.4 stabilisce in dettaglio la procedura di trasferimento del testimone. In ogni istante una sola stazione è in possesso del testimone ed ha quindi il diritto di emettere. Alla fine della emissione dei dati, il testimone è trasferito alla stazione successiva. Tale operazione è ripetuta se, entro un opportuno intervallo di tempo, la stazione non rivela nessuna PDU valida sul bus. Se anche il secondo tentativo non ha successo, la stazione in questione inizia un'opportuna *procedura di recupero*.

La procedura di recupero del testimone è basata sulla emissione di una PDU di controllo, detta *Who_Follows*, che contiene l'indirizzo della stazione che non ha fornito risposta. Tutti i terminali della rete confrontano il valore del campo di dati della PDU Who_Follows ricevuta con l'indirizzo del proprio predecessore (cioè della stazione che normalmente gli spedisce il testimone). La stazione il cui predecessore ha indirizzo uguale a quello letto diviene il nuovo successore della stazione in possesso del testimone. Questa stazione trattiene il testimone e risponde alla sorgente della PDU Who_Follows con la PDU *Set_Successor* , nel cui campo di dati è scritto il suo indirizzo. Risultato di questa procedura di recupero è l'esclusione della stazione in errore o guasto dalla rete.

II.5. Standard IEEE 802.5

L'unità di dati nell'anello è, come per gli altri protocolli, la MAC PDU, la cui struttura è rappresentata in Fig. II.5.1. In tale figura sono inoltre mostrati i formati del testimone e della

sequenza di arresto. Quest'ultima è utilizzata per segnalare l'interruzione prematura della emissione di una trama, e può essere introdotta in qualsiasi momento durante la emissione.

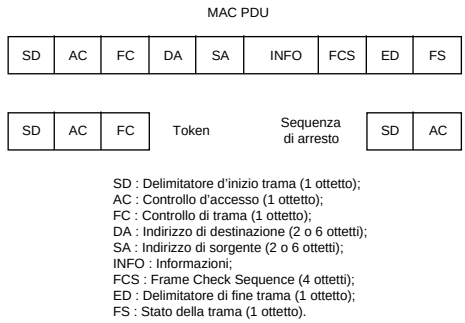


Figura II.5.1 – Struttura della MAC PDU, del token e della sequenza di arresto del protocollo a testimone su anello.

Il formato e la funzione di ciascun campo della MAC PDU è riportato in Fig. II.5.2. In particolare si possono distinguere:

- *delimitatore di inizio trama* (SD, Starting Delimiter): è una sequenza di otto simboli; quattro simboli di questo campo sono codificati attraverso una opportuna violazione del codice di Manchester;
- *controllo di accesso* (AC, Access Control): contiene le informazioni per la gestione delle priorità e l'identificatore del testimone; in particolare il significato dei bit è il seguente:
  - i primi tre bit indicano la *priorità* del testimone e pertanto quale classe di utenti è autorizzata ad utilizzarlo;
  - il quarto bit è il *bit di testimone*, ed il suo valore è "0" in un testimone ed "1" nelle altre MAC PDU; quando una stazione con una PDU in attesa di emissione rivela un testimone circolante sull'anello, con priorità inferiore od uguale a quella della PDU in attesa, pone ad "1" il bit di testimone, trasformando così questo nella sequenza di inizio della MAC PDU;

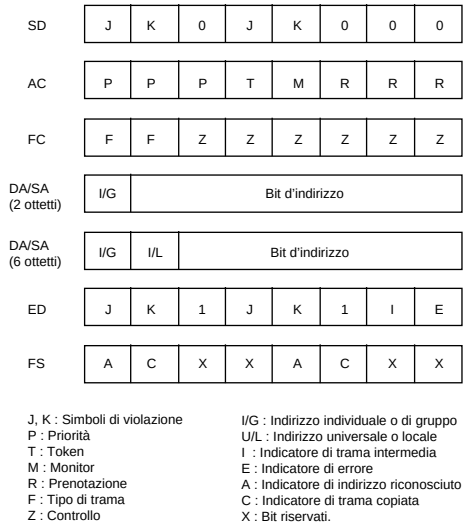


Figura II.5.2 - Formato dei singoli campi della PDU del protocollo a testimone su anello.

- il bit *M* è detto *bit di monitoraggio*, ed è utilizzato dalla *stazione di controllo della rete* (che è l'unica autorizzata a modificare tale campo) per rimuovere quelle PDU che, per effetto di errori di trasmissione, si trovassero a circolare indefinitamente sull'anello; la stazione di controllo legge il valore del bit *M* di ogni MAC PDU che l'attraversa; se tale

bit è uguale a "0", lo sostituisce con il valore "1", altrimenti rimuove tale MAC PDU dalla rete;
- gli ultimi tre simboli di tale campo rappresentano i *bit di prenotazione* e permettono ad una stazione di richiedere il livello di priorità con cui deve essere emesso il successivo testimone.

♦ *controllo di trama* (FC, Frame Control): i primi due bit permettono di distinguere tra due tipi di PDU:
- *00 PDU di controllo* (contiene unità di dati relative al controllo di accesso);
- *01 PDU dati* (contiene una LLC PDU);
- nel caso di MAC PDU, i bit 3-8 di tale campo verranno interpretati come bit di controllo, mentre nel caso di PDU di dati essi indicano la priorità dei dati contenuti nella PDU;

♦ *indirizzi di destinazione e di sorgente* (DA, Destination Address; SA, Source Address): contengono due indirizzi: quello della stazione di destinazione e quello della stazione di origine; ognuno di questi campi di indirizzo può essere lungo 2 o 6 ottetti; in ogni caso tutte i terminali di una specifica LAN devono avere gli indirizzi di uguale lunghezza;

♦ *informazione* (I, Information): può essere lungo 0,1 o più ottetti e trasporta informazioni relative allo strato MAC, allo strato LLC o alla gestione della rete;

♦ *FCS*: è una sequenza di 32 bit ricavata da un polinomio generatore standard di grado 32 e utilizzata per rivelare eventuali errori che abbiano interessato i campi FC, DA, SA ed I della MAC PDU;

♦ *delimitatore di fine* trama (ED, Ending Delimiter): come per il delimitatore di inizio, una PDU valida deve presentare un'opportuna configurazione di simboli nei primi sei sottocampi del campo ED; il sottocampo *I* indica, quando è posto ad "1", che la PDU attuale è una frazione (eventualmente la prima) di una sequenza di PDU; infine se una generica stazione, mediante il controllo di sequenza, rivela un errore sulla PDU, deve porre ad "1" l'ultimo bit del campo ED (bit E).

♦ *stato della trama* (FS, Frame Status): i bit *A* e *C* di questo campo sono posti a zero dalla stazione sorgente e vengono modificati dalla stazione di destinazione per indicare:
- A = 1 PDU ricevuta correttamente;
- C = 1 PDU copiata correttamente;
- gli altri bit di questo campo sono riservati per ulteriori sviluppi del protocollo.

Per la rimozione delle PDU dall'anello è, in linea di principio, possibile operare secondo due differenti schemi: *rimozione alla sorgente* e *rimozione a destinazione*.

Nel primo caso, che è quello previsto dallo standard, il nodo sorgente rimuove la PDU dopo che questa ha percorso completamente l'anello; occorre osservare che tale operazione, oltre a consentire un primo scambio di informazioni di riscontro tra sorgente e destinazione, non richiede la decodifica dell'indirizzo di sorgente, e pertanto può avvenire senza introdurre ulteriori ritardi.

Viceversa, nella rimozione operata dal terminale di destinazione, occorre decodificare l'indirizzo di destinazione, e pertanto la ri-emissione della trama dovrà essere ritardata fintantoché non è stato ricevuto il campo DA di questa. Tale schema consente tuttavia una migliore utilizzazione della banda del canale se la lunghezza elettrica dell'anello è maggiore della lunghezza massima di una PDU.

Lo standard IEEE 802.5 prevede due valori del ritmo binario in linea : 1 Mbit/s e 4 Mbit/s. Il numero massimo di terminali connettibili alla rete è uguale a 72 o 260 in relazione al tipo di mezzo trasmissivo utilizzato. Il primo valore si riferisce ad una coppia simmetrica non schermata, mentre il secondo è valido per coppie schermate.

II.6. Rete Ethernet

Un esempio particolarmente significativo di applicazione delle tecniche protocollari appartenenti alla famiglia CSMA è fornito dalla rete Ethernet, che è la LAN più diffusa a livello mondiale. I suoi principi architetturali sono stati definiti inizialmente nei primi anni ’70 e sono giunti ad un primo livello di definizione normativa nel 1983 a cura del gruppo 802 dell’IEEE.

La topologia logica di questa LAN è il *bus bidirezionale* a cui le stazioni accedono con un opportuno punto di attacco (nodo): cioè più stazioni trasmettono su un unico mezzo di comunicazione e tutte le stazioni ricevono contemporaneamente tutto quello che si trasferisce sul mezzo.

Le unità informative trasferite vengono chiamate “*trame*”, ognuna delle quali reca al proprio interno l’indirizzo di origine e quello di destinazione. Ogni scheda di rete disponibile in commercio è caratterizzata da un indirizzo permanente unico costituito da 6 byte. I primi 3 byte sono assegnati dalla norma ad ogni singolo costruttore, i restanti 3 sono assegnati dal costruttore alla singola scheda.

Nella normativa definita inizialmente la velocità di trasmissione era di 10 Mbit/s in banda base con codifica di tipo Manchester, in modo da facilitare l’estrazione della temporizzazione da parte delle stazioni. Sono però oggi ormai largamente impiegate velocità di trasmissione maggiori: ad es. 100 Mbit/s. E’ stata anche utilizzata la tramissione in banda estesa.

Il protocollo di accesso segue lo schema *CSMA/CD 1-persistente*. Per esso valgono quindi tutte le considerazioni svolte nella prima parte di questa sezione.

I mezzi trasmissivi impiegati possono essere il *cavo coassiale spesso* (thick coax), il *cavo coassiale sottile* (thin coax), il *doppino ritorto* (twisted pair) e la *fibra ottica.*

La struttura realizzativa di una LAN Ethernet comprende *segmenti* e *domini di collisione*. Un segmento è la sezione di mezzo trasmissivo compresa fra due ripetitori; la sua lunghezza massima è determinata dal mezzo trasmissivo impiegato, con specifico riferimento alle sue caratteristiche di attenuazione. Ogni segmento può essere utilizzato per connettere stazioni alla rete oppure può essere solo di collegamento tra due ripetitori: nel primo caso si parla di segmenti “*popolati*”. Relativamente a questi ultimi viene anche precisato il numero massimo di nodi inseribili su quel segmento: questo numero è anch’esso dipendente dal tipo di mezzo trasmissivo impiegato.

Un dominio di collisione è l’area di rete in cui tutte le stazioni condividono il medesimo traffico e quindi anche le medesime collisioni. In un dominio di collisione sono inclusi più segmenti tra loro connessi mediante *ripetitori*.

Per interconnettere due o più domini di collisione si impiegano i *bridge*, che operano a livello di strato 2, con il compito di filtrare le trame in transito facendo passare solo quelle dirette da un dominio ad un altro. La tecnologia attuale permette di miniaturizzare i bridge ed integrarli in un unico elemento di rete detto “*switch*”: questo è nella sostanza un bridge ad alte prestazioni, costituito da una molteplicità di interfacce anche a differenti velocità di trasmissione.

Una LAN Ethernet è spesso indicata con 3 dati:

- un primo dato specifica la velocità di trasmissione sul canale trasmissivo espressa in Mbit/s;
- un secondo dato precisa se la trasmissione è in banda base o in banda estesa;
- un terzo dato indica la massima lunghezza di un segmento di rete espressa in centinaia di metri.

Ad esempio con 10Base5 si indica una LAN Ethernet che opera con una velocità di trasmissione di 10 Mbit/s in banda base e con segmenti aventi una lunghezza massima di 500 m. Se non sono forniti ulteriori dati, si deve intendere che il mezzo trasmissivo impiegato è il coassiale spesso o sottile: questi due casi si distinguono in quanto, ad es., con una velocità di trasmissione di 10 Mbit/s, la lunghezza massima di un segmento è di 500 m nel caso di coassiale spesso e di circa 200 m nel caso di coassiale sottile. Se invece si adotta il doppino ritorto, ai 3 dati sopra precisati va aggiunta la lettera *T* ; nel caso poi di utilizzazione della fibra ottica, la lettera che specifica questo impiego è la *F*.

Ogni dominio di collisione include un numero massimo di segmenti, solo alcuni dei quali sono "popolati". La lunghezza massima di ogni segmento e la composizione di un dominio di collisione in segmenti determinano l'*estensione massima* della LAN. Ad esempio una Ethernet 10Base5 comprende domini di collisione costituiti da 5 segmenti e 4 ripetitori (con solo 3 segmenti popolati); conseguentemente, dato che con il coassiale spesso ogni segmento ha una lunghezza massima di 500 m, ne segue che l'estensione massima della LAN è di 2.500 m. Invece il dominio di collisione di una Ethernet 10Base2, che ha la stessa composizione precedente, ha un'estensione massima di circa 900 m, come si deduce dal fatto che la lunghezza massima di un segmento è in questo caso poco meno di 200 m. L'estensione massima della LAN, unitamente alla dimensione minima della trama e alla velocità di trasmissione, sono fissati in modo da assicurare la rivelazione di una avvenuta collisione da parte di tutte le stazioni: ad es. per un'estensione massima di 2500 m per una velocità di trasmissione di 10 Mbit/s la dimensione minima della trama deve essere di 64 byte.

Circa il numero massimo di nodi inseribili in un segmento, la LAN 10Base5 consente un massimo di 100 nodi per segmento, con una distanza minima di 2,5 m, mentre la LAN 10Base2 consente di utilizzare un massimo di 30 nodi per segmento, con una distanza minima uguale a 50 cm.

Quando si impiega il doppino ritorto (ad es. in una Ethernet 10BaseT), ogni stazione è connessa, tramite il doppino ad un dispositivo "*hub*" che provvede ad agire come centro stella. Viene così realizzata una topologia che è fisicamente a stella e logicamente a bus. L'hub provvede ad amplificare i segnali ricevuti da tutte le stazioni e a ritrasmetterli in uscita in modo diffusivo.

Come già detto, una LAN Ethernet può operare anche ad alta velocità; si parla allora di "*Ethernet veloce*" (Fast Ethernet) e le velocità di trasmissione sono dell'ordine di 100 Mbit/s. I mezzi trasmissivi impiegati in questo caso sono il doppino ritorto e la fibra ottica. La Ethernet 100BaseT utilizza un massimo di 3 segmenti e 2 hub. Dato che ogni segmento ha una lunghezza massima di 100 m e tenendo presente che i 2 hub possono essere alla distanza massima di 5 m, ne deriva che l'estensione massima della struttura è di poco maggiore ai 200 m. ad es. la Ethernet 100BaseTX usa 2 doppini di categoria 5 per ogni stazione ed è in grado di ricevere e di trasmettere contemporaneamente a 100 Mbit/s. La Fast Ethernet è realizzata anche mediante fibre multimodo: in questo caso si utilizzano due fibre in grado di ricevere e trasmettere a 100 Mbit/s in modo pienamente duplice. La dimensione del segmento è in questo caso di 2000 m.

II.7. Lo strato LLC nelle LAN

Il protocollo di controllo del collegamento logico, definito nello standard IEEE 802.2, ha il compito di:

- realizzare lo scambio delle unità informative tra i terminali connessi alla LAN;
- organizzare il flusso delle LLC PDU;
- gestire e interpretare i comandi e le risposte;
- eseguire le funzioni di rivelazione e di recupero degli errori.

Lo strato LLC è stato definito in modo indipendente rispetto alle diverse alternative di protocolli d'accesso della serie IEEE 802 e può essere quindi applicato a qualsiasi tipo di rete locale.

Per soddisfare i requisiti di un ampia gamma di applicazioni sono stati definiti due tipi di servizio di trasferimento tra le entità dello strato LLC:

- il servizio *senza connessione* (connectionless);
- il servizio *con connessione* (connection-oriented).

Il servizio senza connessione utilizza un sotto-insieme delle funzionalità del servizio dello strato LLC ed è impiegato quando i protocolli di più alto livello dispongono delle funzionalità necessarie per il controllo degli errori e della sequenzialità delle unità informative scambiate. Questo tipo di servizio non garantisce la consegna delle LLC PDU.

Il servizio con connessione è comparabile con quello offerto da altri standard internazionali come, ad esempio, dall'HDLC o dal protocollo X.25 livello 2. Tale servizio garantisce quindi il trasferimento affidabile ed in sequenza delle LLC PDU su di una connessione logica denominata *connessione di strato LLC* (Data Link Connection).

È possibile che un terminale gestisca contemporaneamente varie connessioni logiche con terminali diversi. Ogni connessione logica è instaurata e gestita indipendentemente dalle altre, sia per quanto riguarda la definizione dei suoi parametri, sia dal punto di vista dello scambio informativo.

La possibilità per un terminale di gestire una pluralità di connessioni di strato LLC con terminali diversi è una diretta conseguenza dell'ambiente multiaccesso tipico delle LAN. Nelle WAN un ramo è, nella maggioranza dei casi, di tipo punto-punto e pone in corrispondenza due nodi della rete; le connessioni di strato di collegamento, se sono multiple, hanno tutte uguale origine e destinazione. Nel caso delle LAN, un unico mezzo trasmissivo collega una molteplicità di terminali e quindi possono coesistere sulla stessa connessione fisica più connessioni logiche distinte con origine e destinazione diverse. È quindi necessario prevedere i mezzi per distinguere tali connessioni.

Nel seguito del paragrafo verranno descritte dapprima (§ II.7.1) le principali caratteristiche del servizio di strato LLC; successivamente (§ II.7.2) si parlerà del relativo protocollo di strato nelle sue due modalità di funzionamento.

II.7.1. Servizio di strato LLC.

In accordo con quanto specificato dal modello OSI, nel descrivere il servizio di strato LLC, facciamo riferimento alle interazioni tra le entità dello strato LLC e quello dello strato superiore: le seconde sono indicate come *utenti del servizio LLC* (o più brevemente come utenti, quando tale termine non consenta equivocazioni), mentre le prime sono i *fornitori del servizio*.

Nel servizio di strato LLC le interazioni tra utente e fornitore sono descritte attraverso *primitive*, che sono raggruppate in *classi*. Ogni classe descrive un elemento del servizio di strato LLC. Le classi di primitive definite nell'ambito degli standard della famiglia IEEE 802 comprendono (Fig. II.7.1):

- la primitiva di *richiesta* (REQUEST), che consente all'utente locale di richiedere un servizio al fornitore;
- la primitiva di *indicazione* (INDICATION), che consentono al fornitore di indicare all'utente un evento, che si è presentato all'interno allo strato e che è significativo per l'utente stesso;
- la primitiva di *conferma* (CONFIRM), che permette al fornitore di confermare richiedente l'esito del servizio offerto.

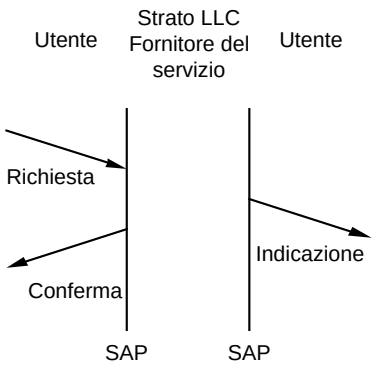


Figura II.7.4 - Primitive di servizio definite per lo strato LLC.

Rispetto alla struttura generale del modello OSI, si può notare la mancanza della primitiva di *risposta* (RESPONSE), utilizzata per consentire all'utente di rispondere alla primitiva di

indicazione. La primitiva di risposta non è stata introdotta perché i servizi previsti negli standard IEEE 802 sono strutturati in modo che, attraverso il servizio di strato LLC, un utente non possa ottenere informazioni sull'utente remoto.

Come precedentemente accennato, lo strato LLC fornisce due tipi di servizio di trasferimento e cioè quello senza connessione e quello con connessione. Il servizio senza connessione prevede lo scambio di *unità di dati del servizio di collegamento* (Link Service Data Unit - LSDU) senza la preventiva instaurazione di una connessione di strato LLC. Il servizio con connessione fornisce i mezzi per instaurare, utilizzare, gestire e abbattere una connessione di strato LLC. Nelle Tabb. II.7.1 e II.7.2 sono riportate, per i due tipi di servizio di trasferimento le relative primitive con i parametri definiti dal colloquio tra due strati omologhi.

Primitiva di servizio	Parametri
L_DATA Request	Indirizzo locale Indirizzo remoto LSDU Classe di servizio
L_DATA Indication	Indirizzo locale Indirizzo remoto LSDU Classe di servizio

Tabella. II.7.1 - Primitive della classe di servizio senza connessione.

Occorre considerare anche le primitive di servizio che descrivono le interazioni tra le entità dello strato LLC e quelle dello strato MAC nella fase di trasferimento dei dati. Queste primitive consentono la definizione del servizio che lo strato MAC deve fornire ad un'entità locale dello strato LLC per consentire, a quest'ultima, di scambiare unità di dati con un'entità remota (*servizio di strato MAC*). Tali primitive sono descritte nella Tab. II.7.3.

Nel caso di servizio di trasferimento senza connessione, la emissione di una LLC PDU può avvenire in un qualsiasi istante senza la preventiva instaurazione della connessione logica. L'unica classe di primitive necessarie è quindi quella relativa al trasferimento dei dati (L_DATA). La primitiva di richiesta L_DATA è emessa dallo strato superiore per richiedere il trasferimento di una LSDU. La primitiva di indicazione L_DATA è invece utilizzata dallo strato LLC per avvisare l'utente della ricezione di una LSDU.

La procedura di instaurazione di una connessione logica da parte di un utente dello strato LLC è effettuata per mezzo della classe di primitive L_CONNECT. L'utente che vuole instaurare una connessione trasferisce allo strato LLC una primitiva di richiesta L_CONNECT, indicante l'indirizzo dell'utente remoto con cui si vuole realizzare una connessione. Nella stessa primitiva è anche indicato il valore della priorità che si richiede per lo scambio dei dati su quella connessione. L'entità di strato LLC che riceve tale primitiva inizia la procedura di instaurazione della connessione, mediante il protocollo di strato LLC. L'LLC entità remota, quando riceve la richiesta di instaurazione di una connessione, trasferisce al proprio utente la primitiva di indicazione L_CONNECT. Al termine della procedura, all'utente che l'ha iniziato viene trasferita la primitiva di risposta L_CONNECT. Entrambe le primitive di indicazione e di risposta contengono un campo che informa se l'instaurazione della connessione ha avuto successo o, eventualmente, le ragioni della mancata instaurazione.

II.7.2. Protocollo di strato LLC.

Il protocollo LLC definito nello standard IEEE 802.2 è di tipo orientato al bit ed è basato sullo scambio di PDU aventi struttura simile a quella delle trame HDLC (Fig. II.7.2).

I due campi di *indirizzo* (DSAP e SSAP) hanno ambedue lunghezza uguale a un ottetto ed identificano i punti di accesso al servizio (SAP) di destinazione e di sorgente della LLC PDU. La struttura dei due campi è mostrata in Fig. II.7.3.

Fase della comunicazione	Primitiva di servizio	Parametri
Instaurazione della Connessione	L_CONNECT Request	Indirizzo locale Indirizzo remoto Classe di servizio
	L_CONNECT Indication	Indirizzo locale Indirizzo remoto Classe di servizio Stato
	L_CONNECT Confirm	Indirizzo locale Indirizzo remoto Stato
Trasferimento dati	L_DATA_CONNECT Request	Indirizzo locale Indirizzo remoto LSDU
	L_DATA_CONNECT Indication	Indirizzo locale Indirizzo remoto LSDU
	L_DATA_CONNECT Confirm	Indirizzo locale Indirizzo remoto Stato
Riinizializzazione della Connessione	L_RESET Request	Indirizzo locale Indirizzo remoto
	L_RESET Indication	Indirizzo locale Indirizzo remoto Motivo
	L_RESET Confirm	Indirizzo locale Indirizzo remoto Stato
Controllo di flusso verso lo strato 3	L_CONNECTION_FLOW CONTROL Request	Indirizzo locale Indirizzo remoto Quantità di dati permessa
	L_CONNECTION_FLOW CONTROL Indication	Indirizzo locale Indirizzo remoto Quantità di dati permessa
Disconnessione della Connessione	L_DISCONNECT Request	Indirizzo locale Indirizzo remoto
	L_DISCONNECT Indication	Indirizzo locale Indirizzo remoto Motivo
	L_DISCONNECT Confirm	Indirizzo locale Indirizzo remoto Stato

Tabella II.7.2 - Primitive della classe di servizio con connessione.

Fase della comunicazione	Primitiva di servizio	Parametri
Trasferimento dei dati verso lo strato MAC	MA_DATA Request	Indirizzo di destinazione MAC_SDU
	MA_DATA Indication	Indirizzo di destinazione Indirizzo di sorgente MAC_SDU Stato Classe di servizio richiesta
	MA_DATA Confirm	Stato Classe di servizio fornita

Tabella II.7.3 - Primitiva di scambio dati con lo strato MAC.

Il primo bit del campo DSAP indica se l'indirizzo di destinazione è di tipo individuale o di gruppo, mentre il primo bit del campo SSAP stabilisce se la LLC PDU è un comando o una risposta. I restanti sette bit dei due campi sono i veri e propri bit di indirizzo. La configurazione di sette "1" consecutivi del campo DSAP indica un indirizzamento globale; in questo caso la LSDU deve essere consegnata a tutti i SAP attualmente attivi. La configurazione di sette bit "0" rappresenta invece un indirizzamento nullo e non identifica alcun SAP.

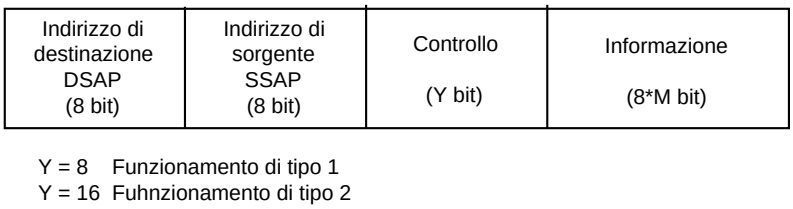


Figura II.7.2 - Struttura della LLC PDU.

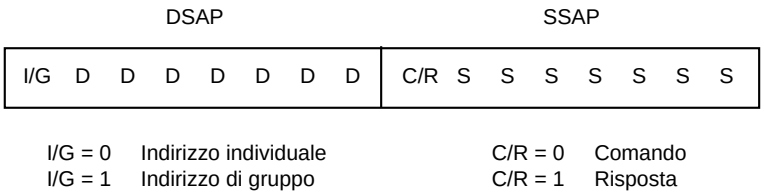


Figura II.7.3 - Struttura del campo di indirizzo di una LLC PDU.

Il campo di *controllo* consiste di uno o due ottetti che identificano il tipo ed i numeri di sequenza delle LLC PDU. Lo scambio delle LLC PDU può avvenire secondo due tipi di funzionamento:

- *funzionamento di tipo 1* : le unità di dati sono scambiate senza instaurare una connessione di strato LLC; non sono previsti meccanismi di recupero degli errori o di riscontro e non è effettuato alcun controllo di flusso.
- *funzionamento di tipo 2*: le unità di dati sono trasferite su una connessione di strato LLC, che deve essere opportunamente instaurata, mantenuta ed abbattuta; sono previste procedure di controllo di flusso, di recupero di errori e di riscontro delle trame ricevute correttamente.

Il funzionamento di tipo 2 è applicato nel caso di connessioni bilanciate. In questo caso le LLC PDU sono numerate e viene attivata una numerazione indipendente per ogni coppia di SAP di sorgente e di destinazione

Sono definite tre tipi di unità di dati: a) le LLC PDU *informative numerate*; b) le LLC PDU *di supervisione*; c) le LLC PDU *non numerate*. Il funzionamento di tipo 1 prevede solo lo scambio di LLC PDU non numerate, mentre il tipo 2 utilizza anche LLC PDU di supervisione e informative numerate.

Oltre alla classificazione dei due tipi di funzionamento, è prevista anche la classificazione delle entità di strato LLC. Una LLC entità di *classe 1* può fornire solo un funzionamento di tipo 1, con indirizzamento globale, individuale o nullo, senza riscontri e senza controllo di flusso. Una LLC entità di *classe 2* può fornire funzionamenti sia di tipo 1 che di tipo 2.

L'insieme dei comandi e delle risposte supportate dalle entità di classe 1 e 2 è indicato in Tab. II.7.4.

II.7.2.1. Funzionamento di tipo 1.

Le LLC PDU utilizzate nel funzionamento di tipo 1 sono quelle necessarie per lo scambio di unità informative non numerate, senza riscontro e senza controllo di flusso. Queste LLC PDU possono essere di tre tipi:

- *informazione non numerata* (Unnumbered Information - UI) per gestire la trasmissione di informazioni non numerate ad una o più entità di destinazione;
- *prova* (TEST) per verificare lo stato della corrispondenza fra le LLC-entità;
- identificazione dello scambio (Exchange Identification - XID) per l'esecuzione di procedure di gestione dello strato; la LLC-entità remota deve rispondere al comando XID con una risposta XID che identifica l'entità stessa e che contiene un campo informativo uguale a quello ricevuto.

Tipo di LLC-entità	Funzionamento	Comandi	Risposte
Classe 1	Tipo 1	UI XID TEST	 XID TEST
Classe 2	Tipo 1	XID TEST UI	XID TEST
	Tipo 2	I RR RNR REJ SABME DISC	I RR RNR REJ UA DM DISC FRMR

Tab. II.7.4 - LLC PDU gestite dalle due classi di LLC-entità.

La codifica del campo di controllo contenuto nelle PDU definite per le operazioni di tipo 1 è riportata in Fig. II.7.4. Il *bit Poll/Final* (P/F) è utilizzato nelle PDU di comando come bit Poll e consente, se posto ad 1, di sollecitare il terminale remoto a emettere una PDU di risposta con il bit Final posto anch'esso ad 1.

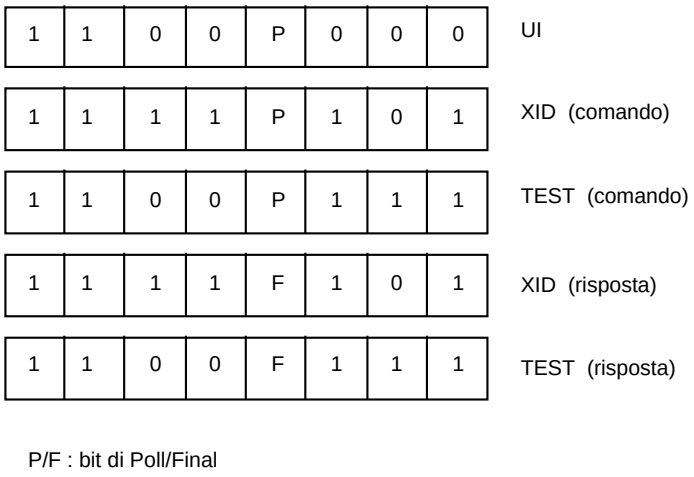


Figura II.7.4 - Codifica del campo di controllo delle PDU definite per il funzionamento di tipo 1.

Il bit P/F è utilizzato, ad esempio, nel caso di situazioni di recupero, rese necessarie dal silenzio prolungato del terminale lontano. In questo caso, il terminale che non riceve risposta dal terminale remoto emette una PDU TEST con il bit P=1. Tale PDU richiede all'entità remota di emettere una risposta TEST con il bit F allo stesso valore del bit P ricevuto e con lo stesso campo informativo. Se la procedura non ottiene risposta, il terminale remoto deve essere considerato fuori servizio.

La PDU XID può avere varie applicazioni. Ad esempio, una XID con un indirizzo nullo sollecita la risposta da parte di tutti i terminali ed è utilizzata per localizzare i terminali stessi; una XID recante un particolare indirizzo di gruppo può essere utilizzata per verificare quali terminali appartengono a quel gruppo.

II.7.2.2. Funzionamento di tipo 2.

Le trame definite nel caso di funzionamento di tipo 2 contengono un campo di controllo analogo a quello definito per le trame del protocollo X.25 livello 2, ma a differenza di queste, l'unica numerazione prevista è modulo 128. Di conseguenza il massimo numero di trame emesse e non riscontrate è uguale a 127.

Le LLC PDU definite nel funzionamento di tipo 2 differiscono fra loro per la struttura del campo di controllo e per la presenza o meno del campo informativo. La struttura del campo di controllo è mostrata in Fig. II.7.5.

Queste LLC PDU sono di tre tipi:

- *Informative* (I): consentono il trasferimento numerato dei dati; il campo di controllo comprende quattro sottocampi:
  - *identificatore* di LLC PDU di tipo I; il bit è posto a 0;
  - *numero di sequenza in trasmissione N(S)*; questo è una variabile modulo 128 che indica il numero d'ordine dell'ultima LLC PDU di tipo I emessa; è aggiornato ad ogni emissione di PDU di tipo I sulla base del valore della *variabile di stato in trasmissione V(S)*;
  - bit *Poll/Final* (P/F); nelle LLC PDU di comando è utilizzato come bit Poll posto ad 1 per sollecitare una risposta da parte dell'entità remota; a sua volta l'entità remota deve rispondere a questa interrogazione con una PDU di risposta che avrà il bit Final posto ad 0;
  - *numero di sequenza in ricezione N(R)*; questa una variabile modulo 128 che indica il numero d'ordine della LLC PDU di tipo I attesa; il suo valore deve essere uguale a quello della *variabile di stato in ricezione V(R)* ed indica che sono state ricevute correttamente tutte le LLC PDU numerate da 0 a *N(R)-1*.
- *Supervisione* (S): gestiscono le funzioni di supervisione delle connessioni consentendo:
  - il *riscontro positivo* (comandi e risposte RR, Receive Ready);
  - la *richiesta di ritrasmissione* (comandi e risposte REJ, Reject);
  - la richiesta al terminale lontano di sospendere temporaneamente la emissione di LLC PDU di tipo I (comandi e risposte RNR, Receive Not Ready);

  il campo di controllo delle LLC PDU di tipo S differisce da quello delle PDU di tipo I per la mancanza del sottocampo *N(S)* e contiene due bit che consentono di distinguere le tre PDU di tipo S.
- *Non Numerate* (U): consentono la emissione di informazioni in operazioni senza riscontro e senza connessione; oltre alle LLC PDU già descritte nel caso di funzionamento di tipo 1 (comandi e risposte XID, comandi e risposte TEST, comando UI), sono previsti:
  - il comando di connessione bilanciata di due LLC-entità (SABME, Set Asynchronous Balanced Mode Extended);
  - il riscontro non numerato (UA, Unnumbered Acknowledgment);
  - il comando di disconnessione (DISC, Disconnect);
  - la risposta DM (Disconnect Mode), impiegata, da parte dell'entità remota, per indicare che non è connessa;
  - la risposta FRMR (Frame Reject), che indica, in operazioni bilanciate, un errore non recuperabile con la ritrasmissione, con conseguente necessità di reinizializzare la connessione di strato LLC.

Le procedure previste per il funzionamento di tipo 2 sono analoghe a quelle del protocollo X.25 livello 2. Per brevità esse non saranno qui commentate, per i dettagli si rimanda il lettore all'unità 7 e allo standard IEEE 802.2.

L'instaurazione di una connessione può essere iniziata da qualsiasi entità di strato LLC. L'entità iniziante emette una SABME PDU e fa partire un temporizzatore. Alla ricezione della UA PDU di risposta, l'entità porrà a zero le variabili V(S) e V(R), fermerà il temporizzatore ed entrerà nella fase dati. Se in risposta ad una SABME PDU, l'entità riceve una DM PDU, l'instaurazione della connessione non è andata a buon fine e l'entità segnalerà l'evento allo strato superiore. Se il temporizzatore si esaurisce prima della ricezione di una risposta, l'entità emette nuovamente una SABME PDU e fa ripartire il temporizzatore. In mancanza di risposte la procedura viene ripetuta per un numero fissato di volte, dopo di che l'instaurazione è considerata fallita.

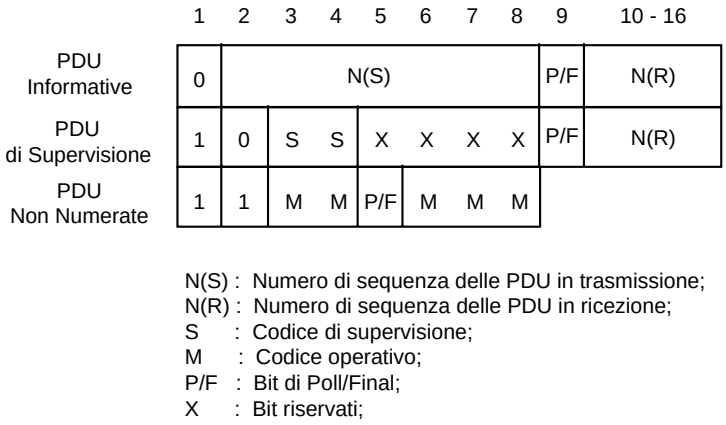


Figura II.7.5 - Codifica del campo di controllo delle PDU definite per il funzionamento di tipo 2.

II.8. La MAN DQDB

Lo standard IEEE 802.6, a differenza dei precedenti che trattano esclusivamente protocolli per LAN, è orientato alla definizione dell'architettura di una MAN.

Nell'accezione dello standard, una MAN è una rete in grado di fornire in modo integrato un'ampia gamma di servizi, che vanno da quelli per dati a quelli vocali e video, su di un'area geografica sufficientemente estesa. In generale, una MAN è vista come composta da un insieme di sottoreti connesse tra loro da opportuni dispositivi di interconnessione e di instradamento (bridge, router). Lo standard IEEE 802.6 descrive le funzionalità di una sottorete, denominata Distributed Queue Dual Bus (DQDB), che può essere utilizzata come parte componente di una MAN.

Una rete DQDB è di tipo multiaccesso ed è in grado di fornire un ambiente di comunicazioni integrate. Essa infatti può essere supporto sia di comunicazioni di dati, con trasferimento senza connessione e/o con connessione, sia di comunicazioni di tipo isocrono, come ad esempio quelle vocali. Queste funzionalità sono riassunte nella Fig. II.8.1, che mostra l'architettura di una rete DQDB. Come si può notare, lo strato DQDB deve comprendere sia le funzionalità necessarie a regolare l'accesso al canale trasmissivo, sia quelle necessarie all'adattamento dei servizi al tipo di trasferimento adottato in rete. Nel seguito ci occuperemo esclusivamente degli aspetti riguardanti le procedure d'accesso al canale trasmissivo.

La particolare struttura protocollare della rete DQDB nasce allo scopo di fornire un servizio di trasferimento integrato per diverse tipologie di servizi. Nel caso di interconnessione tra LAN, lo strato DQDB accetta in ingresso LLC PDU che provvede a trasferire tra le due LAN a colloquio. In questo caso le funzionalità richieste allo strato DQDB sono esclusivamente quelle di strato MAC. Invece, nel caso di trasferimento di servizi isocroni, lo strato deve anche eseguire tutte quelle funzioni che rendano possibile la ricostruzione del messaggio originale a destinazione.

La rete DQDB utilizza una topologia a doppio bus unidirezionale. IN questo caso, per aderenza alla terminologia dello standard, i terminali sono chiamati *stazioni di rete*.

La capacità di ogni bus è suddivisa in intervalli temporali (IT) di lunghezza fissa che sono generati dalla *stazione di testa* di ogni bus (head bus). Tali IT possono essere usati per la emissione, da parte stazioni connesse alla rete, in accordo alle regole del protocollo d'accesso. Il flusso di dati è terminato da un opportuno dispositivo posto alla fine di ogni bus.

La struttura di una stazione DQDB è mostrata in Fig. II.8.2. Ognuna di queste è costituita da un'*unità d'accesso,* che svolge le funzioni del protocollo DQDB, e da due *interfacce* identiche, una per ogni bus, che eseguono le operazioni di lettura e di scrittura. L'operazione di lettura su un bus precede quella di scrittura ed è completamente passiva, mentre l'operazione di scrittura è ovviamente attiva ed è eseguita semplicemente tramite una funzione "OR" tra il flusso di bit del bus e quello emesso dalla stazione.

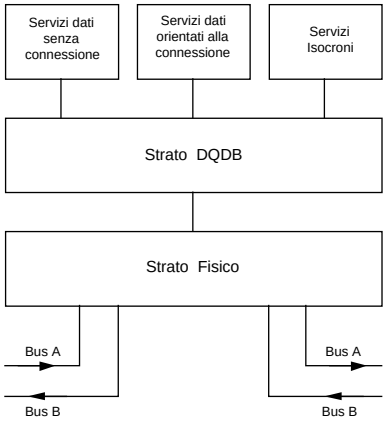


Figura II.8.5 - Architettura logica dei protocolli IEEE 802.6.

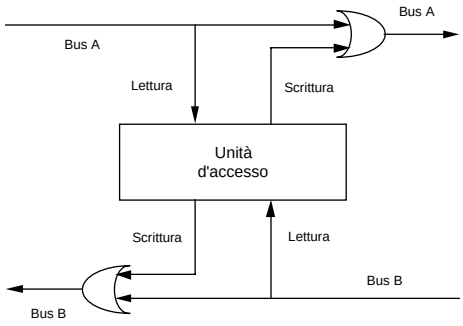


Figura II.8.2 - Struttura di una stazione DQDB.

Il protocollo DQDB prevede due modalità di accesso al bus: 1) un accesso con *arbitraggio a coda* (Queued Arbitrated - QA); 2) un accesso con *pre-arbitraggio* (Pre Arbitrated - PA).

Ognuna di queste modalità utilizza IT di tipo diverso, chiamati rispettivamente *QA* e *PA*. Ogni IT (Fig. II.8.3), sia di tipo QA che PA, ha lunghezza fissa, uguale a 53 ottetti, ed comprende: 1) *campo di controllo accesso* (Access Control Field - ACF), di lunghezza uguale ad un ottetto, che contiene le informazioni per l'utilizzazione della sua capacità; 2) un *segmento informativo* (segment), che contiene le informazioni da trasferire.

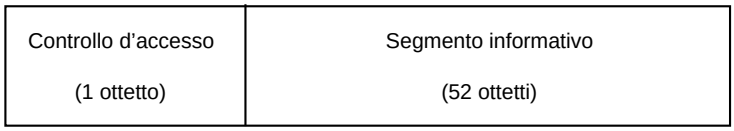


Figura II.8.3 - Struttura di uno slot nel protocollo DQDB.

La struttura del campo ACF è mostrata in Fig. II.8.4; i bit che lo compongono hanno il seguente significato:

- *bit di occupato* (B): indica se l'IT è attualmente occupato;
- *bit di tipo* (SL_TYPE): indica se l'IT è di tipo QA o PA;
- *bit di ricezione del segmento precedente* (PSR): indica se il segmento precedente è stato ricevuto oppure no;
- *quattro bit di richiesta* (R): indicano che un segmento informativo è stato inserito nella coda distribuita in accordo alla sua priorità (nel seguito supporremo, per semplicità che sia prevista una sola classe di priorità e quindi sia sufficiente un solo bit R);
- l'ulteriore bit del campo ACF è riservato per usi futuri.

L'accesso con arbitraggio a coda è regolato dal *protocollo a coda distribuita* ed è orientato alle comunicazioni di dati, mentre l'accesso pre-arbitrato è usato per le comunicazioni isocrone e riproduce uno schema di trasferimento dell'informazione analogo a quello a circuito.

Nel seguito saranno analizzate entrambe le procedure di accesso.

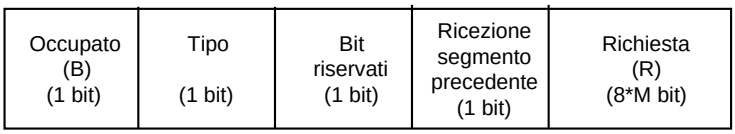


Figura II.8.4 - Struttura del campo ACF.

Il protocollo a coda distribuita è ad accesso controllato. Esso ha lo scopo di riprodurre, in modo distribuito, il funzionamento di un sistema a coda con disciplina primo arrivato-primo servito. In altre parole, il protocollo, tramite un opportuno scambio di informazioni, assegna il diritto di emissione alle stazioni rispettando l'ordine temporale di arrivo dei segmenti informativi a queste.

Il protocollo è basato sull'utilizzazione di due bit del campo ACF: il *bit di occupato B* (busy) che indica se l'IT è attualmente occupato e il *bit di richiesta R* (request) che indica che un segmento informativo è stato inserito nella coda distribuita.

Ogni nodo, contando il numero di richieste che riceve ed il numero di IT vuoti che passano, è in grado di determinare, istante per istante, il numero totale dei segmenti accodati nelle stazioni a valle su ogni bus. Disponendo di tale informazione, una stazione, quando deve trasmettere un segmento, ha la possibilità di determinare immediatamente la sua posizione nella coda distribuita. Se non ci sono segmenti in coda l'accesso può essere immediato, altrimenti occorrerà attendere che tutti i segmenti in coda siano trasmessi.

È il caso di sottolineare che, rispetto ai protocolli MAC sino ad ora descritti, nel protocollo a coda distribuita, le stazioni memorizzano le informazioni relative allo stato della rete che sono utilizzate al momento dell'accesso. Al contrario, nei protocolli precedenti, tutta l'informazione relativa allo stato della rete era contenuta nella rete stessa ed i terminali avevano il compito di attendere il verificarsi di opportuni eventi (canale libero, ricezione del testimone, ecc.).

La Fig. II.8.5 descrive in dettaglio l'algoritmo di base del protocollo a coda distribuita per l'accesso ad uno dei due bus, ad esempio il bus A. Un algoritmo identico presiede l'accesso al bus B. Nel caso in esame, l'algoritmo si basa sull'elaborazione del bit B, trasportato dagli IT in transito sul bus A, e del bit R, trasportato in senso opposto dagli IT in transito sul bus B.

Ogni stazione tiene memoria del numero di segmenti collocati nella coda distribuita da parte delle stazioni a valle sul bus A, aggiornando il proprio *contatore RQ* (Request Counter). Tale contatore viene incrementato ogniqualvolta la stazione rivela, sul bus B, il transito di un bit R=1 e viene decrementato ogniqualvolta, sul bus A, transita un IT vuoto (bit B=0).

Quando una stazione deve emettere un segmento informativo, avverte le stazioni a monte sul bus A ponendo, alla prima opportunità, un bit R=1 su di un IT del bus B. Contemporaneamente salva il contenuto corrente del contatore RQ nel *contatore CD* (Count Down). Con tale operazione il segmento è collocato nella coda distribuita. Una stazione può immettere nella coda distribuita solo un segmento alla volta; il segmento successivo può essere inserito solo quando la emissione del precedente è stata completata.

Il valore del contatore CD indica la posizione del segmento nella coda distribuita. La stazione guadagnerà quindi l'accesso al canale solo dopo che le stazioni a valle avranno esaurito le emissioni precedentemente richieste. Il contatore CD viene decrementato nello stesso modo del contatore RQ, in tal modo la stazione ha il diritto alla emissione nel primo IT vuoto successivo a quello che causa l'azzeramento del contatore CD.

Ovviamente durante il tempo in cui una stazione è in attesa di una emissione, deve comunque continuare l'aggiornamento del contatore RQ.

È evidente che queste operazioni garantiscono normalmente la emissione dei segmenti rispettando l'ordine di presentazione delle richieste.

Un esempio della procedura di accodamento del protocollo d'accesso a coda distribuita è mostrato nella Fig. II.8.6 nel caso di rete composta da cinque stazioni. Inizialmente supponiamo che il valore di tutti i contatori RQ sia posto a zero e tutti gli IT che passano sul bus A siano occupati in modo da non consentire emissioni e comportare decrementi dei contatori RQ. Supponiamo inoltre che le stazioni 5, 2 e 3 presentino nell'ordine una richiesta. Al termine delle operazioni, il contatore RQ della stazione 1 avrà valore 3 avendo visto passare tre richieste sul bus B. Analogamente il valore del contatore RQ della stazione 4 avrà valore 1, poiché l'unica stazione a valle che ha effettuato una richiesta è stata la stazione 5. La stazione 5 avrà i contatori CD e RQ a 0 poiché è stata la prima ad effettuare la richiesta e non ha nessun altra stazione a valle. La stazione 3 avrà il contatore CD ad 1 ed il contatore RQ=0. Il primo indica che la stazione 3 dovrà attendere la emissione di una stazione posta a valle, e cioè la stazione 5, prima di poter effettuare la propria. Il secondo indica che, dopo la richiesta della stazione 3 non è stata presentata nessun altra richiesta da parte delle stazioni a valle. Per ragioni analoghe la stazione 2 è caratterizzata da CD=1 e RQ=1.

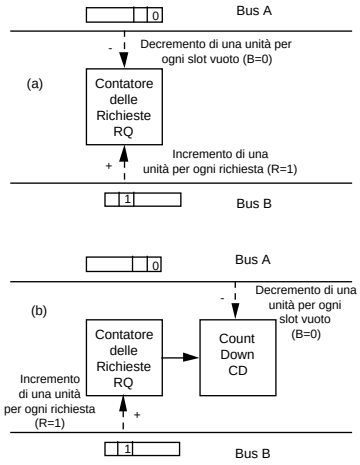


Figura II.8.5 - Algoritmo di coda distribuita: a) aggiornamento del contatore RQ; b)

Un esempio della procedura di trasmissione del protocollo d'accesso a coda distribuita a partire dallo stato ora è mostrato nella Fig. II.8.7. Supponiamo che nessuna altra richiesta venga presentata dalle stazioni e che gli IT che transitano sul bus A siano tutti vuoti. In questo caso, il primo IT vuoto verrà utilizzato dalla stazione 5 che eliminerà quindi il contatore CD e sarà disponibile, eventualmente, a richiedere un'altra emissione. Al transito dello slot vuoto le altre stazioni hanno provveduto a decrementare i propri contatori (CD e RQ). Il secondo IT vuoto verrà utilizzato dalla stazione 2 perché è la prima stazione attraversata che ha il contatore CD posto a 0. Il terzo IT vuoto sarà utilizzato infine dalla stazione 3. A questo punto lo stato della rete è tornato quello iniziale con tutti i contatori posti a 0.

È il caso di sottolineare che, a causa dei tempi di propagazione finiti, può accadere che l'ordine di presentazione delle richieste non corrisponda esattamente a quello di emissione. Ciò può accadere se l'intervallo di tempo tra le due richieste è inferiore a quello di propagazione tra le due stazioni. Conseguenza del fenomeno precedente e che non è più rispettato il principio di equità nell'assegnazione del diritto d'accesso alla rete. Infatti, specialmente se la rete è molto estesa, alcune stazioni, quelle più vicine alla stazione di testa, possono essere avvantaggiate nel guadagnare il diritto di emissione.

Per ovviare a ciò, lo standard IEEE 802.6 prevede l'applicazione di un ulteriore meccanismo, detto di *bilanciamento di banda* (bandwidth balancing). Questo modifica parzialmente le modalità di utilizzazione degli IT vuoti, costringendo una stazione a non utilizzarne alcuni, anche se ha segmenti informativi in coda.

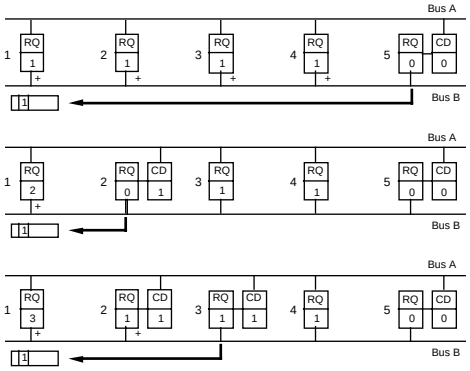


Figura II.8.6 - Inserimento dei segmenti nella coda distribuita.

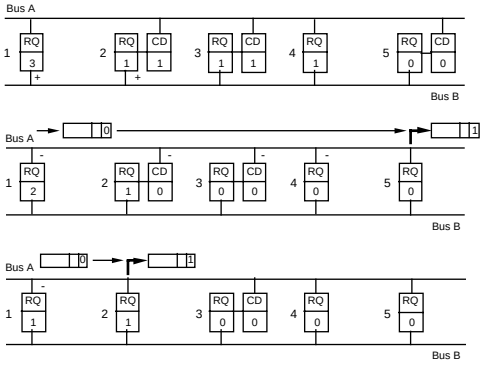


Figura II.8.7 - Trasmissione dei segmenti in coda distribuita.

Lo standard definisce un parametro di sistema, detto *modulo di bilanciamento di banda* (bandwidth balancing module - BWB_MOD), ed un *contatore di emissioni* (bandwidth balancing counter). Tale contatore ha modulo uguale BWB_MOD ed è incrementato quando la stazione effettua una emissione. Dopo che la stazione ha effettuato BWB_MOD emissioni consecutive, il contatore è azzerato, ciò indica che l'IT vuoto successivo, che dovrebbe essere utilizzato per la trasmissione, deve essere lasciato transitare inalterato.

In questo modo alcuni IT vuoti sono lasciati a disposizione delle stazioni a valle. È possibile dimostrare che ciò è sufficiente ad eliminare qualsiasi fenomeno di disequità nell'accesso al canale trasmissivo.

La seconda modalità d'accesso prevista nello standard IEEE 802.6 è quella relativa agli IT PA. Tale tecnica di accesso è riservata alle comunicazioni isocrone ed è realizzata con modalità analoghe al trasferimento a circuito.

Quando una stazione deve eseguire una comunicazione isocrona, la stazione di testa del bus provvederà ad emettere, al ritmo sufficiente per il servizio previsto, gli IT del tipo PA. Tali IT potranno essere utilizzati solo dalla stazione che ne ha fatto richiesta. Il riconoscimento di tali IT avviene mediante la lettura di un apposito campo di indirizzo, denominato *identificatore di canale virtuale* (Virtual Channel Identifier - VCI). Questo identifica univocamente una chiamata.