“Trasporto della posta, trasporto della voce umana, trasporto di immagini in movimento - in questo secolo, come in altri, i nostri più grandi progressi hanno sempre l'unico scopo di mettere gli uomini in contatto”

[Antoine de Saint Exupery]

Riduzione dei costi, taglio delle spese di trasferta, continuità della relazione, velocizzazione delle decisioni e possibilità di condividere a distanza file e applicazioni, sono solo alcuni dei benefici introdotti dalla videocomunicazione agli organismi che decidessero di farne uso. In Italia queste prospettive non sembrano aver fatto gola alle istituzioni, le quali secondo un sondaggio condotto nel 2004 in merito alle applicazioni utilizzate, piazzerebbero la videocomunicazione al terzultimo posto, con il 6% sul totale. La domanda da porsi è quindi quali possono essere le motivazioni che hanno portato questo scetticismo attorno alla videoconferenza?

Inizialmente le ragioni potevano essere ricondotte a motivi economici e tecnologici.
La videoconferenza si è affacciata nel mercato italiano agli inizi degli anni novanta, grazie alla diffusione delle linee ISDN. L'installazione di un apparato per videoconferenza comunque, portava all'istituto o azienda una notevole spesa per l'installazione di più linee(per aumentare la velocità erano necessarie più collegamenti in parallelo) e per l'hardware stesso; per questo la videocomunicazione era strettamente legata a sale riunioni di grandi enti, e aziende multinazionali.
La grande occasione della videocomunicazione fu rappresentata dall'IP.
La videoconferenza attraverso IP, ha portato un enorme sviluppo a questa tecnologia, oltre ad un abbassamento dei costi sugli apparati. Questo è stato un enorme passo avanti, dal momento che per realizzare il servizio non era più necessaria una linea dedicata, ma era più che sufficiente una connessione ad Internet. Questo ha aperto la strada ad un nuovo tipo di clientela, quali medie e piccole imprese, i quali avrebbero potuto usufruire di questo potente mezzo di comunicazione. Non era esclusa a nessuno ora la possibilità di creare spazi virtuali in cui incontrare partner, fornitori, clienti o qualsiasi altra persona. Inoltre la videocomunicazione su IP si integra perfettamente con i vecchi apparati ISDN, per cui contemporaneamente si salvaguardarono gli investimenti già effettuati e si garantì l'interoperabilità di tecnologie diverse.

Fu questo il contesto nella quale all'ITU sovvenne l'idea di creare una specifica come tecnologia unica per la standardizzazione delle procedure volte a regolare la videoconferenza.
L'introduzione dell'IP comunque portò a sua volta un problema (se pur minore in quanto legato alle performance) di banda. Nelle reti IP la banda deve essere condivisa tra tutte le applicazioni che gli utenti stanno utilizzando, rendendo così la banda larga una premessa indispensabile allo sviluppo della videocomunicazione. Premessa che tutto sommato qui si verificò a metà del 2005, con un enorme incremento degli accessi ad internet con banda larga(5,6 milioni contro i 300 mila del 2001), con la crescita omogenea dei vari segmenti di utilizzo. Il tasso delle imprese che possedevano una connessione a banda larga salì al 37%, le connessioni domestiche raggiunsero il 20%(la metà delle famiglie che possiede internet), mentre è il 61% il tasso delle istituzioni che ha un collegamento a banda veloce; tale valore comprende il 52% dei comuni, il 73% delle scuole, e l'85% delle strutture sanitarie. Si prevede per metà del 2006, che gli accessi ad internet a banda larga saranno oltre la soglia dei 7 milioni.

Eppure gli enti non riservarono ancora alla videocomunicazione una particolare attenzione, tanto che c'è chi ipotizza che il problema non è mai stato quello tecnologico ma bensì umano:
un primo motivo può essere nella non chiarezza dei possibili benefici che le videoconferenza potrebbero portare all'interno dello sviluppo di un'istituzione (sia essa commerciale o di altro tipo), forse proprio perchè le aziende dell'offerta non riescono a trasmettere i valori di questa tecnologia alla clientela;
la paura, specialmente da parte del personale che non ha una preparazione informatica specifica, che gli apparati siano complessi, ed anziché agevolare, siano un ostacolo;
la diffidenza verso ciò che è nuovo, e la riluttanza a cambiare le proprie abitudini lavorative;

l'assenza di socializzazione è inoltre vista come un freno alla videocomunicazione, anche se essa non si propone di sostituire i rapporti umani, ma solo di integrarli.

Per il fatto che inizialmente la videocomunicazione era accessibile solamente attraverso linee dedicate e apparecchiature costose, per molto tempo è stata confinata nelle sale riunioni appositamente attrezzate. Poi l'evoluzione tecnologica portate dall'IP, le ha consentito di arrivare fino al desktop, togliendogli così la sua posizione di strumento di lusso per "Top Manager". La videocomunicazione al desktop consente di conferire con l'interlocutore mediante un qualsiasi PC provvisto di microfono e webcam. E questa è stata vista da molti come il motore che farà impennare il mercato della videocomunicazione, in quanto ci sono soluzioni che hanno introdotto innovazioni che vanno oltre la singola possibilità di videoconferenza. Tra le nuove potenzialità, l'affiancarsi della possibilità di effettuare lavori collaborativi, tramite la condivisione di dati e applicazioni. Il Data Collaboration come definito dal T.120 non consente agli utenti la mera visualizzazione di documenti in remoto, ma la possibilità di interagirvi in tempo reale, così permettendo un'interazione totale.

Per quanto riguarda gli scenari applicativi che abbracciano questa nuova disciplina ne troviamo diversi.
In primo piano c'è l'azienda, il cui contesto richiede spese sempre minime, così facendole abbracciare ampiamente i vantaggi portati dalla videocomunicazione, con applicazioni che variano dalla videocomunicazione collaborativi al supporto delle relazioni commerciali alla formazione del personale, per la quale la videoconferenza oltre a fornire un valore aggiunto riesce ad abbattere gran parte dei costi.
Un'area in cui la videoconferenza è destinata a produrre cambiamenti è quella del telelavoro, una forma di lavoro che si sta sempre più diffondendo anche grazie all'incentivo di molti Paesi lungimiranti, che mirano alla decongestione del traffico.

Nell'ambio dell'insegnamento a distanza sono sempre più utilizzate soluzioni di streaming e di video on demand per accedere alle sessioni in differita. L'insegnamento attraverso sistemi di videocomunicazione da luogo a modelli di insegnamento/apprendimento caratterizzati da comunicazione bi-direzionale e interattività. Tale possibilità di interagire con il docente ha eliminato il tallone di Achille della "tradizionale" formazione a distanza, cioè la mancanza di interazione.

La videocomunicazione svolge un ruolo importante anche nell'ambito sanitario. Ogni branca della medicina, dalla cardiologia, alla dermatologia, alla radiologia potrebbe avvalersi della telemedicina, che secondo la definizione formulata dalla CEE, consente non solo di assicurare assistenza medica a pazienti lontani ma anche altre attività quali emergenza sanitaria, tele assistenza domiciliare(home care), diagnosi e consultazioni remote , servizi ambulatori remotizzati e servizi per aree disagiate.

L'uso della videocomunicazione nel settore medico è in continua crescita: in Italia tra le strutture mediche che si avvalgono di tale tecnologia, il 40% si avvale del teleconsulto, mentre il 25% dell'home care.

Nell'ambito del processo di modernizzazione della Pubblica Amministrazione, con la creazione di un modello di e-government che posi su infrastrutture abilitanti e su strumenti di coinvolgimento e partecipazione dei cittadini, la videocomunicazione potrebbe essere un importante mezzo di collegamento sia tra il cittadino e la PA sia tra le differenti amministrazioni. I progetti di videocomunicazione all'interno del piano di e-government, consistono in più che una sfida tecnologica, una sfida culturale in quanto si tratta di integrare la "vecchia" cultura amministrativa con il "nuovo" mondo della rete

Inoltre comincia anche a svilupparsi la videocomunicazione senza fili grazie alla diffusione di standard per la connessione wireless a banda larga(ad esempio il Wi-max che consente di coprire grandi distanze e aprirebbe nuovi sbocchi per la connettività internet).

1 Lo scopo di H.323

Per che si abbia effettiva comunicazione, un comune linguaggio di comunicazione tra i partecipanti deve essere scelto, in modo tale da evitare la parziale o totale incomprensione tra di essi, risultante dalla mera trasmissioni di suoni tra le parti. Questo non succede solo alle persone; perchè un computer possa comunicare con un altro, bisogna che essi operino su di un suolo comune, proprio per il contesto libero a cui questi appartengono.

Reti di elaboratori sono formate sopra a protocolli e standard, scelti in modo tale che le applicazioni che operano al si sopra del livello di rete possano coesistere e cooperare al pieno delle loro capacità. Proprio a questo proposito, verso la metà degli anni novanta, la International Telecommunications Union(ITU) ha pubblicato una raccomandazione, l'H.323, come estensione logica dell'H.320, che definisce i protocolli per la comunicazione multimediale attraverso LAN.
Eredita inoltre le caratteristiche da sia i tradizionali protocolli su PSTN, che dagli standard relativi ad Internet. Ed è proprio per questa sua peculiaritá che H.323 si distingue dagli altri protocolli; ha portato con sé l'abilitá di integrarsi ad Internet ed al www come interfaccia per reti eterogenee, così fornendo una serie di applicazioni, che sorpassano i classici servizi, quali la possibilità di sostenere videochiamate da locazioni remote o di modificare in real-time un documento con altri utenti attraverso Internet utilizzando il proprio Pc, o qualsiasi altro servizio derivato dall'utilizzo di un interfaccia HTTP tra un'applicazione H.323 ed un qualsiasi altro terminale.

Lo standard H.323 fornisce una base, in quanto definisce le componenti tecniche necessarie, per la comunicazione audio, video e di dati attraverso reti basate sullo scambio di pacchetti(nato per fornire il servizio nelle LAN, il

protocollo si è successivamente esteso a tutte le reti PSN, trovando il suo campo di sviluppo nelle reti IP), quali LAN, MAN, Enterprise Area Network, Intranet, Extranet e Internet. Conformandosi all'H.323, i prodotti e le applicazioni multimediali di diversi venditori possono interoperare, siano essi applicazioni embedded, applicativi software o dispositivi standalone, permettendo agli utenti di comunicare senza interessarsi della compatibilità. Ed è proprio la compatibilità la chiave che concerne i venditori e gli utenti di applicazioni basate su LAN. Fornendo interoperabilità da dispositivo a dispositivo, da applicazione a applicazione, da venditore a venditore e soluzioni ibride(un dispositivo con un'applicazione, magari di due venditori diversi), H.323 permette ai prodotti di un utente di interoperare con tutti gli altri prodotti basati sul protocollo. Ed è per questo motivo che oggi tale protocollo è leader nel suo settore, portando milioni e milioni di bit di videoconferenza ogni ora.

1.1 Famiglia H.32x

H.323 è solo una parte di una più larga serie di protocolli per la comunicazione che rendono possibile la videoconferenza attraverso reti di diversa tipologia. Conosciuta con il nome di H.32x la famiglia di protocolli comprende:

- H.320 trasmissione attraverso ISDN(Integrated Services Digital Network);
- H.321 e H.310 trasmissione attraverso B-ISDN (ISDN a banda larga);
- H.322 trasmissione attraverso LAN che forniscono una garantita qualità di software(QoS);
- H.324 trasmissione attraverso SCN (Switched Circuit Network) e PSTN (Public Switched Telephone Network).

Uno dei principali obiettivi nello sviluppo dell' H.323 è l'interoperabilità tra reti eterogenee. Tale obiettivo è stato raggiunto, come vedremo più dettagliatamente in seguito, mediante l'utilizzo di un componente chiamato gateway, il quale traduce segnali provenienti da altre reti in segnali H.323 e vice versa.

	H.320	**H.321**	**H.322**	**H.323**	**H.324**
Data	1990	1995	1995	1996	1996
Network	ISDN a banda stretta	ISDN a banda larga	PSN con banda garantita	PSN senza garanzia di banda	PSTN e POTS
Video	H.261 H.263	H.261 H.263	H.261 H.263	H.261 H.263	H.261 H.263
Audio	G.711 G.722 G.728	G.711 G.722 G.728	G.711 G.722 G.728	G.711 .722, .728 .723, .729	G.711 G.722 G.728
Controllo chiamata	H.230 H.242	H.242	H.230 H.242	H.245	H.245
Dati	T.120	T.120	T.120	T.120	T.120
Multipoint	H.231 H.243	H.243	H.231 H.243	H.323	
Multiplex	H.221	H.221	H.221	H.225.0	H.223

1.2 Benefici dell'H.323

Codec Standard

H.323 stabilisce durante la fase di negoziazione, gli standard per la compressione e decompressione di audio e video, assicurando che differenti prodotti possano comunicare tra di loro secondo ad un comune supporto.

Interoperabilità

Nelle conferenze gli utenti di una rete non vogliono preoccuparsi della compatibilità degli altri terminali partecipanti. H.323 oltre ad assicurare che il terminale destinatario possa decomprimere le informazione ricevute, stabilisce i metodi per lo scambio delle capacità trasmissive e i metodi di comunicazione adeguati a tali capacità, gestisce il set-up delle chiamate e, una volta che la connessione è stabilita, le segnalazioni di controllo.

Indipendenza dalla rete

H.323 non dipende dalla rete sul quale il servizio viene eseguito in quanto si trova ad un livello superiore dell'architettura di rete. La sua posizione sulla pila protocollare è singolare, in quanto si trova a cavallo tra il livello di rete e quello applicazione, ma è considerato appartenente a quest'ultimo.
Lo scopo di H.323 non include né la LAN né lo strato di trasporto su cui essa si appoggia per collegare diverse reti. Solo gli elementi richiesti per la comunicazione rientrano nel suo scopo.

Indipendenza dalle piattaforme e applicazioni

H.323 non è legato a nessun hardware o sistema operativo. Il protocollo rivolto alle piattaforme sarà disponibile in una varietà di versioni come ad esempio abilitata al video, dedicate ad una specifica piattaforma, abilitate al telefono tramite IP e dispositivi dedicati, così che un utente può scegliere l'opzione a lui più adeguato in base alle esigenze.

Supporto del Multipoint

Nonostante il protocollo potrebbe supportare conferenze con 3 o più punti terminali(endpoint) senza richiedere uno specifico MCU(multipoint control unit), questo fornirebbe una più appropriata, potente e flessibile architettura volta ad ospitare una conferenza multipoint. Questa capacità potrebbe essere inclusa in altri componenti di un sistema H.323.

Gestione della larghezza di banda

Video e audio utilizzano una consistente quantità della larghezza di banda, che a volte può causare una congestione della rete.

Il "bandwidth managing" consente di limitare il numero di connessioni H.323 simultanee entro la quantità di larghezze di banda disponibile dall'applicazione, in modo da assicurare che il limite di traffico critico non venga infranto.

Supporto del Multicast

In conferenze multipoint H.323 può supportare il trasporto multicast (trasmissione in contemporanea ad un certo numero di stazioni di lavoro). In multicast un pacchetto inviato viene trasmesso ad un set di destinazioni nella rete senza bisogno di replica. Al contrario unicast invia molteplici trasmissioni point-to-point, mentre broadcast a tutte le destinazioni.

In unicast o broadcast la rete è utilizzata inefficientemente in quanto i pacchetti sono replicati a in tutti i punti della rete. Multicast invece usa la larghezza di banda in maniera più parsimoniosa dal momento che tutte le stazioni appartenenti al gruppo leggono un solo stream.

Flessibilità

In una conferenze H.323 possono essere inclusi endpoint con differenti capacità. Per esempio un terminale con sole capacità audio potrebbe partecipare in una conferenza con endpoint aventi capacita trasmissive superiori, quali video e data. Inoltre un terminale multimediale H.323 potrebbe condividere una porzione di data derivante da una videoconferenza

con un terminale T.120, capace della sola trasmissione di data, mentre esso condivide voce, audio e data con gli altri terminali H.323. Questo perché, le capacità delle unità di videoconferenza, le quali possono integrare o meno determinate caratteristiche, conferisce a H.323 una flessibilità sia in termini strutturali che modali.

Inter-Network Conferencing

H.323 usa diversi codec derivanti da altrettante tecnologie di videoconferenza per una comune tecnologia di codifica che consente di minimizzare i ritardi dovuti alla decodifica e di avere le migliori performance. Questo permette a utenti appartenenti a reti eterogenee di comunicare utilizzando lo standard come mezzo comunicativo; per esempio H.323 potrebbe fare da mezzo di collegamento tra un sistema desktop residente su una LAN e sistemi basati su ISDN.

Controlli centralizzati e distribuiti

H.323 spinge la gran parte delle funzionalità di controllo agli endpoint, mentre lui continua a fornire il servizio di trasmissione dati. Nelle tradizionali CSN, tutte le funzionalità di controllo erano attribuite a switch centralizzati. Con H.323 parte o tutte queste funzionalità sono portate ai bordi, perché talvolta è sensato che due terminali comunichino direttamente, senza l'ingombro di unità centrali che gestiscano tutto. Così facendo, H.323 fornisce agli endpoint la possibilità di svolgere compiti che prima erano riservati a server centralizzati. Si potrebbe parlare ora di endpoint intelligenti, capaci di effettuare o accettare chiamate autonomamente senza l'ausilio di unità di rete centralizzate. Ci sono invece casi nella quale il controllo centralizzato è altamente desiderato, per esempio quando un provider vuole monitorare l'utilizzo del servizio, ecc.. Ci sono diverse ragioni per la quale si potrebbe desiderare o l'una o l'altra modalità, e h.323 provvede ad una certa flessibilità per quanto riguarda il controllo.

Integrazione con gli standard di Internet

H.323 si integra perfettamente con le tecnologie esistenti di internet, come RTP/RTCP, URL e DNS. In fatti è stato il primo standard ad adottare RTP/RTCP per la trasmissione multimediale.

Tale protocollo permette di effettuare chiamate, semplicemente ciccando su di un URL e permette agli endpoint di eseguire query DNS in modo tale da localizzare un utente o un servizio analogamente a come un web browser vorrebbe localizzare un sito web.

2 Tipologie di Client

Durante una videoconferenza attraverso IP, gli endpoint partecipanti non vogliono che durante la trasmissione attraverso la rete i dati siano soggetti a degrado e che tale trasmissione utilizzi quanta meno banda possibile. Il compromesso per avere una maggior efficienza di rete è la necessità che gli endpoints codifichino audio, video e dati durante la trasmissione dall'endpoint trasmittente, e che tali media stream vengano poi decodificati una volta ricevuti. In questo modo i flussi di dati saranno in minor modo soggetti a degradi durante la trasmissione ed impiegheranno una minor larghezza di banda. Ma questi processi di compressione e decompressione richiedono risorse quali tempo e potenza processuale per elaborare gli streams. E tali risorse sono inesorabilmente inverse, ciò significa che la minor forza processuale implica l'introduzione di un sempre maggiore ritardo nella comunicazione. Tuttavia compressione e decompressione sono funzione logiche della videoconferenza, ciò significa che queste possono essere implementate o integrate in diversi dispositivi. In base alla configurazione che il sistema di videoconferenza presenterà troviamo diverse tipologie con relative prestazioni.

2.1 Soluzioni Hardware

Uno dei vantaggi introdotti da H.323 è stato quello della flessibilità strutturale. I produttori di unità di videoconferenza dalla loro parte hanno avuto la possibilità di produrre una considerevole serie di dispositivi differenti, in base a quello che il mercato richiedeva, integrando in alcuni di essi caratteristiche aggiuntive, o producendone di dedicati mossi da una politica di interoperabilità tra unità. In questo capitolo parleremo delle categorie che integrano risorse e capacità in hardware.

2.1.1 Componenti Set-top

La prima tipologia di videoconferenza hardware sono le "applicazioni" standalone non basate su PC(standalone non-PC-based). Tale tipologia comprende tutti i dispositivi harware specializzati(comprensivi di sistema, camera, microfono, tipicamente piazzati in cima ad uno schermo) che forniscono videoconferenza ad alta qualità per sale conferenze medie o grandi. Sono indipendenti, ovvero non si appoggiano sopra a nessun altro programma, come ad esempio fanno gli endpoints PC-Based, ed inoltre sono più grandi e costosi dei più semplici dispositivi USB. Esempi di questa tecnologia basata sull'H.323, sono Polycom ViewStation e Aethra Vega. Questa soluzione in genere è la più costosa ma fornisce le prestazioni migliori in quanto tutte le risorse processuali sono integrate su hardware apposito, cosicché indipendenti ad altre applicazioni.

2.1.2 Hardware PCI e USB per desktop

La videoconferenza attraverso H.323 è spesso considerata una tecnologia al desktop, ma i PC potrebbero non essere sufficientemente potenti per fornire performance ad alta qualità e compressione/decompressione ful-motion,

specialmente se consideriamo che il sistema potrebbe eseguire contemporaneamente altre applicazioni. Questo ha indotto i venditori di H.323, ad integrare codec per la compressione e decompressione di audio e video direttamene nei loro prodotti di videoconferenza. Tali codec sono specificatamente disegnati per alleggerire i terminali dal peso della codifica audio e video, permettendo agli endpoint di avere migliori performance. Nel passato e talvolta ancora oggi, i codecs sono inclusi come una scheda addizionale PCI.
Più recentemente la tendenza si è indirizzata verso i dispositivi "Plug & Play". Le risorse processuali richieste per la codifica sono incluse nel dispositivo USB(ad esempio la telecamera del dispositivo per la codifica video) e nella porta USB che fornisce la larghezza di banda necessaria a passare i dati video compressi al PC. Esempi di questa tecnologia sono ViaVideo di Polycon.

2.1.3 PC-Based integrated Codecs

E' una combinazione delle precedenti due. Sono in genere computer con hardware specializzato aggiunto, come per esempio una scheda di acquisizione, che offrono possibilità connettive come la categoria Set-top, ma rendono possibile anche la combinazione della videoconferenza con applicazioni collaborative. Esempio di questa tecnologia è il Polycom iPower

2.2 Soluzioni software

Talvolta le caratteristiche necessarie sono implementate in software, così da richiedere il più basso livello per quanto riguarda le risorse hardware, ma a scapito delle risorse processuali di cui si fa un largo abuso. Per quanto riguarda le soluzioni software si distinguono, se pur concettualmente uguali, due sotto categorie, ovvero i package H.323 e gli applicativi opensource.

2.2.1 Package H.323

Tra le soluzione, i client software sono decisamente i più semplici. Produttori che prima offrivano applicazione desktop H.323 basate su hardware, ora stanno offrendo applicativi H.323 basati interamente su software. Tali soluzioni sono spesso le più economiche da implementare, dovuto anche al fatto dei bassi costi per Webcam USB e microfoni. Ed è per questa sua caratteristica che le soluzioni basate su software sono più tosto attrattive per organizzazioni con scarsi o nessun fondo per la videoconferenza. L'unica avvertenza è che i client software richiedono sistemi più potenti, dal momento che essi utilizzano la gran parte di CPU per la codifica e decodifica del video streaming. In passato era un pesante carico per il sistema, spesso causa di una incostante riproduzione video. Ma con il crescere delle potenzialità hardware, si sono cominciati ad avere buone performance anche dalle tecnologie basate su software. Esempi di questa tecnologia variano dal più generico NetMeeting di Microsoft al più professionale vPoint HD di VICON.

Da considerare che gli endpoints basati sull'hardware costano più rispetto alla loro controparte solo software(i cui prezzi variano dall'economico al gratis), e questo costo supplementare è giustificato dal fatto che soluzioni per videoconferenza basate sull'hardware forniscono sempre prestazioni decisamente migliori, in quanto comprensive di componenti dedicati.

2.2.2 Open Source H.323

L'idea di fondo dietro al movimento open source, è che quando il codice sorgente del software è disponibile, gran parte dei programmatori andrà a leggerlo e valutarlo. A questo punto potrebbero apportare miglioramenti o comunque sia trovare i bug prima di qualsiasi compagnia che sia proprietaria del codice. Tra gli altri vantaggi del codice open source troviamo i costi e la portabilità. Il codice dell'open source, gratis, potrebbe essere alla base di software, quando i fondi sono limitati(ed il tempo ancor più). Il codice open source è portabile, in quanto spesso può essere compilato ed eseguito su diversi computer e diversi sistemi operativi.

3 Tipologie di videoconferenza

Abbiamo appena elencato le tipologie di endpoint che possono instaurare una connessione tra loro per effettuare una videoconferenza. Il numero di terminali che partecipano ad una connessione ed il modo in cui essi operano insieme a livello di streaming e funzionalità di controllo identificano diverse tipologie di conferenza. Si parlerà di videoconferenza point-to-point (punto a punto) quando i flussi dati sono scambiati direttamente (o attraverso un gateker) da un terminale all'altro e vice versa. Se invece ad un point-to-point si uniscono ulteriori utenti, la conferenza si trasforma in una multipunto.

3.1 Point-to-Point

La videoconferenza point-to-point è la tipologia più semplice. Non richiede nessuna tecnologia o unità aggiuntiva per che si abbia effettiva comunicazione. L'unico requisito è una connessione di rete ed un terminale; ed è in base alla natura di questa che si possono rilevare tre diverse tipologie di conferenza da punto a punto.

3.1.2 Desktop

La conferenza point-to-point al desktop e' la tipologia che sicuramente richiede il più basso livello di risorse hardware, software e di rete. Una configurazione minima per un sistema che supporti la condivisione di audio, video e dati con capacità di chat e whiteboard potrebbe essere un 486 o superiore, con una connessione modem a 28.8, comprensivo di scheda audio, microfono, video camera ed un qualsiasi applicativo o package H323 come ad esempio Microsoft NetMeeting o Linux GnomeMeeting. I partecipanti al

meeting vedranno rispettivamente l'immagine dell'interlocutore ed a opzionalmente la propria immagine in un riquadro più piccolo, ma tale configurazione cambia da client a client, in base a come esso sia impostato.

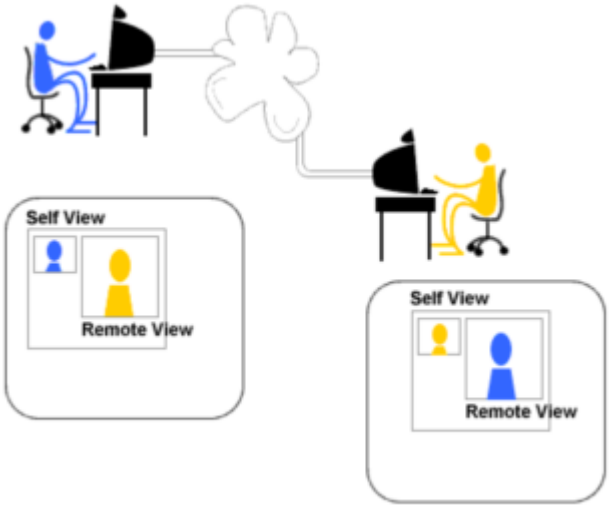

Figura 3.1: Illustrazione di una videoconferenza al desktop

3.1.2 Group to group

La videoconferenza da gruppo a gruppo richiede un passo avanti in termini di dispositivi periferici. Infatti display, videocamere e microfoni tipicamente utilizzati nelle videoconferenze desktop, sono in genere troppo piccoli e deboli, quindi inadatti per una sala conferenza, che di conseguenza necessita di dispositivi dedicati. In oltre questo tipo di conferenza richiede naggiore larghezza di banda per supportare lo streaming di maggiori quantita' di video, a prestazioni migliori. Anche nella videoconferenza da gruppo a gruppo l'output dipenderà dall'impostazione del dispositivo di videoconferenza. L'immagine dell'interlocutore puo' essere proiettata a tutto schermo in un

display e la proprio immagine in un altro, oppure le due immagini potrebbero coesistere nello stesso schermo.

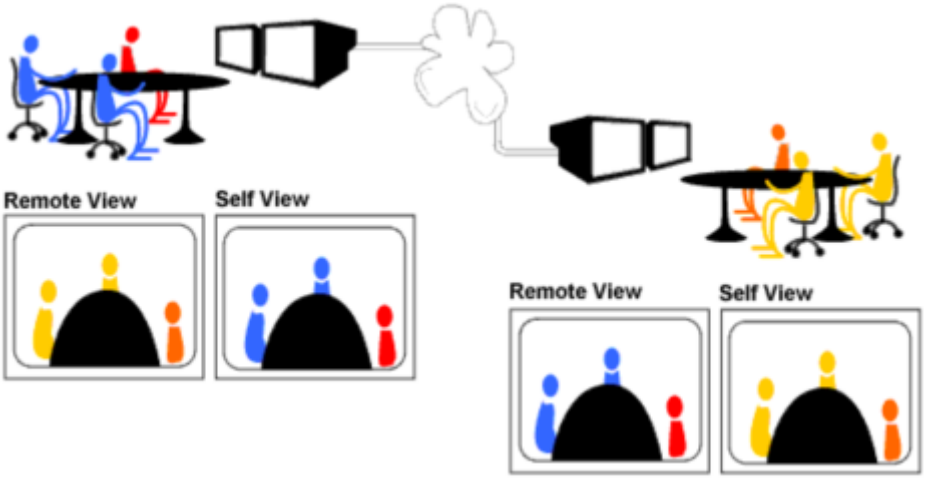

Figura 3.2: tramite apparecchiature dedicati sono possibili conferenze per gruppi di persone

3.1.3 One to group

La videoconferenza da uno a gruppo, e' la soluzione ibrida tra le precedenti due, ovvero quando il meeting è composto da un client al desktop e l'altro da hardware dedicato.

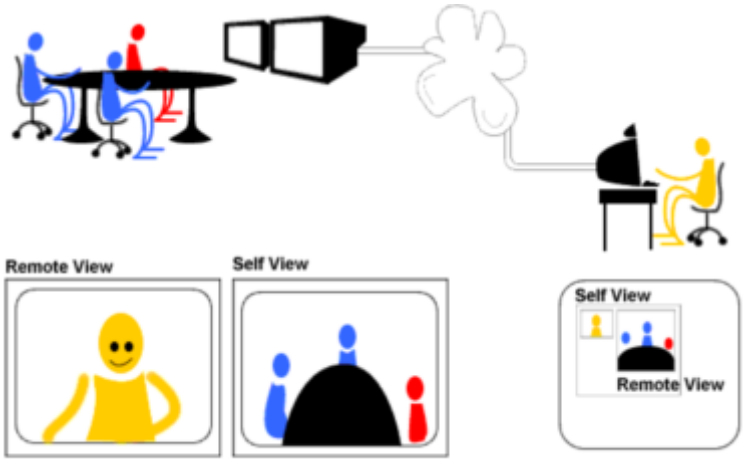

Figura 3.3: Apparecchiature dedicate e per desktop possono interoperare per dar vita a one to group

3.2 Multipunto

Il multipunto oltre che per il numero di partecipanti differisce dal point-to-point per due ulteriori caratteristiche richieste ai fini della videoconferenza. La prima e' la necessita di avere un componente aggiuntivo, una multipoint control unit o reflettore che provveda al collegamento tra i molteplici terminali. Il suo prezzo variando ampiamente da componente a componente abbraccia tutte le esigenze di multiconferenza. La seconda e' che i client che dovranno partecipare alla conferenza debbano poter supportare più connessioni audio e video simultaneamente. Questa necessità potrebbe non essere obbligatoria in una multiconferenza centralizzata(discuteremo della natura della multiconferenza nei seguenti capitoli). Ad esempio NetMeeting nonostante supporti la multiconnessione per la condivisione di applicazioni, chat e whiteboard, non supporta più di una connessione per volta per la condivisione di audio e video. Al contrario altri software H.323 quali CuSeeMe supportano più connessioni audio e video simultaneamente, senza l'ausilio di programmi esterni.

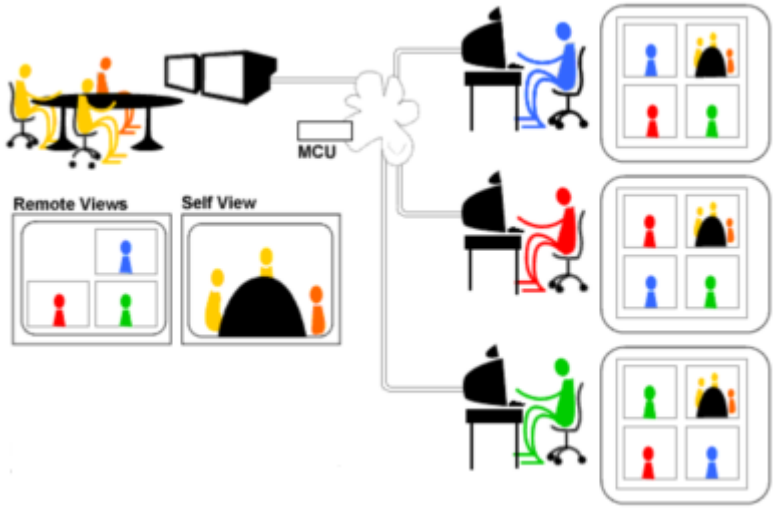

Figura 3.4: H.323 permette di far comunicare insieme diverse tecnologie contemporaneamente

Nel multipunto ciò che vedono i partecipanti può essere guidato da due modalità di videoconferenza:

Voice Activated – dove in automatico il video della persona che più recentemente ha parlato è visualizzato in tutti gli altri siti a pieno schermo. Tutti gli altri flussi video non appariranno nello schermo, ma potrebbero apparire da un momento all'altro prendendo la parola. Ogni terminale partecipante alla videoconferenza vedrà se stesso in una parte del monitor durante tutto il tempo il quale la videoconferenza avrà vita, mentre l'altra parte di schermo sarà occupata dal terminale che sta parlando. Il cambiamento del video da un sito a quello "parlante" potrebbe impiegare diversi secondi da quando ha cominciato a parlare e fondamentale è l'impostazione degli altri microfoni in modalità "mute" o inattivi in modo tale da prevenire che il cambio del controllo video dovuto a insignificanti suoni.

Continuous Presence – Tutti vedono tutti e l'immagine dello schermo è suddivisa in quattro (o più) riquadri. Tutti i partecipanti della videoconferenza possono vedere allo stesso tempo tutti gli altri partecipanti; vedranno loro stessi in una parte dello schermo, mentre gli altri partecipanti saranno rappresentati da più piccole icone. Continuous Presence non è molto adeguata per la condivisione di dati(documenti, lavagne ecc.) proprio per la ridotta dimensione con la quale i partecipanti saranno visti. Nel caso che si voglia cambiare la modalità di videoconferenza da CP a VA, è buona raccomandazione chiudere la conferenza e riconnettersi successivamente onde evitare che la qualità del video sia scarsa, in quanto la riconnessione in modalità voice activated assicura una migliore qualità di immagine. In tutti i terminali i microfoni possono essere indiscriminatamente attivi, ma questo può portare alla produzione di rumori ed eco durante la videoconferenza. Può esserci un Chair Control.

4 Architettura

H.323 non può essere considerato come un protocollo vero e proprio. Esso infatti definisce le collaborazioni tra i diversi elementi, comprendendo componenti, protocolli e procedure, che forniscono servizi volti alla comunicazione multimediale di voce, audio e data attraverso una rete PSN. Essenzialmente è una specifica che lega insieme un numero di protocolli già esistenti, ed è quindi buona norma studiarlo parallelamente con gli altri protocolli. Per questo motivo tale standard viene anche chiamato raccomandazione a ombrello, in quanto volto a regolare le cooperazioni dei protocolli inseriti nella suite.

Spesso H.323 è considerato solo per la sua abilità di segnalazione di chiamata, responsabile del set-up di chiamata e dell'instaurazione della connessione tra gli endpoints. Analizzando individualmente protocolli e componenti vedremo che lo scopo dell'H.323 è ben più ampio. La raccomandazione copre interamente le richieste tecniche per la comunicazione audio e video nelle LAN, senza però garantire il sevizio. Inoltre inglobando nella suite anche il protocollo T.120, H.323 abilità conferenze con la possibilità di data sharing.

5 Componenti

H.323 definisce quattro elementi, chiamati anche entità H.323, richiesti per le trasmissioni multimediali. Alcuni di essi sono obbligatori, mentre altri opzionali. Questi elementi quando connessi insieme forniscono una sorta di sottorete a livello applicativo, interagente con le altre tipologie di sottorete.

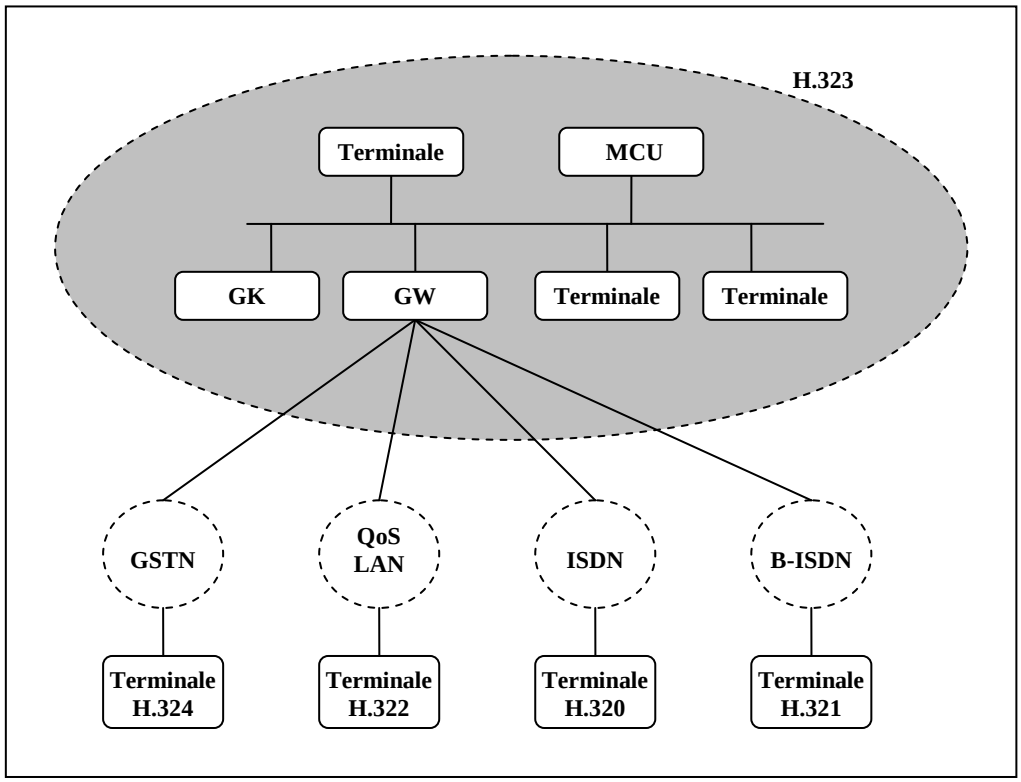

Figura 5.1 traccia di un sistema H.323 con i suoi componenti

Spesso queste unità sono implementate in software ed è possibile che in uno stesso computer siano installate più di una stessa unità, è per questo che in fase di instaurazione di chiamata vengano individuati le entità master e quelle slave. Tali componenti sono i seguenti:

Terminali

Un terminale è un punto finale di una rete(endpoint), il quale può fornire comunicazione bi-direzionale in tempo reale con un altro terminale, un gateway o un MCU. Un terminale potrebbe fornire solo audio, audio e data, audio e video, oppure audio, video e data insieme.

Gateway

Un gateway(GW) provvede alla conversioni di protocolli da un terminale H.323 e un terminale che non supporta l'H.323. Per esempio un GW potrebbe instradare audio proveniente da chiamate IP da un terminale H.323 attraverso PSTN, permettendo di effettuare regolari chiamate telefoniche da applicazioni H.323 come NetMeeting, oppure attraverso ISDN così diventando ponte tra le due tecnologie.

Gatekeeper

Un gatekeeper(GK) è un entità opzionale. Il suo compito principale è il controllo dell'ammissione entro una rete concedendo o rifiutando la comunicazione di due entità H.323 entro la sua zona. Fornisce anche servizi di traduzione di indirizzo(per esempio da numero di telefono a IP).

Multipoint Control Unit

Un MCU fornisce servizi che permettono a tre o più terminali di prendere parte ad una conferenza.

E' composta da un Mupltipoint Controller per la gestione dei controlli di chiamata e da opzionali Mutipoint Processors per lo scambio dei dati nella multiconferenza.

E' inoltre dovuto sottolineare che, mentre terminali, gateway e MCU sono visti come entità di terminazione(da cui viene il nome endpoint), il gatekeeper

è considerato come entità di rete. Questo deriva dal fatto che mentre i primi sono in grado di generare e ricevere chiamate, il gatekeeper non può essere chiamato e non genera alcun flusso multimediale.

L'insieme di tutti i terminali, gateways, e MCU gestiti da un singolo gatekeeper viene definita zona H.323. Una zona quindi, contiene al meno un terminale registrato ad un gatekeeper e potrebbe includere gateway e MCU opzionalmente. Obbligatoria è invece la presenza di un solo gatekeeper, il quale la definisce estendendola a tutti i componenti ad esso registrati.
Una zona H.323 è indipendente dalla topologia della rete o potrebbe essere compresa in diversi segmenti di reti diversi connessi tra loro tramite router o altri dispositivi.

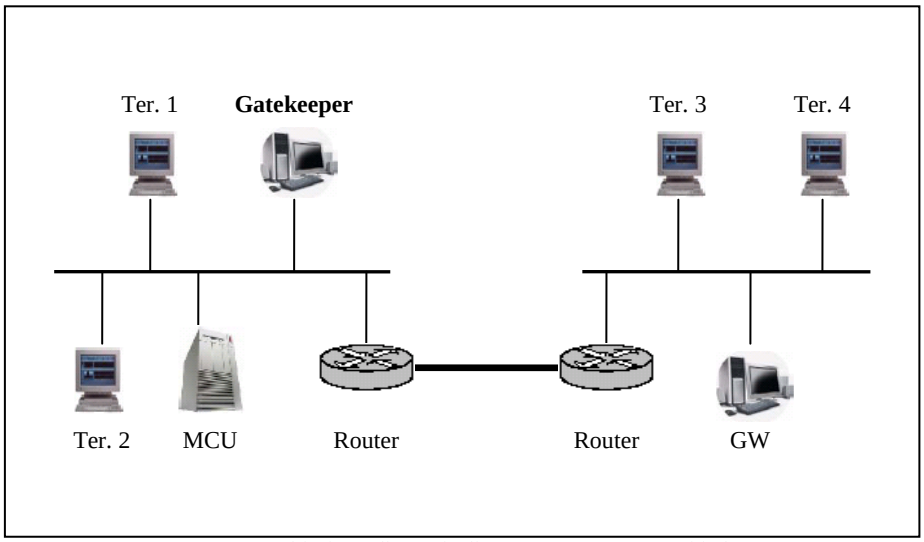

Figura 5.2: l'insieme di Terminali, Gateways, e MCU gestiti da un singolo gatekeeper è conosciuta come zona H.323

Se più zone sono dislocate sotto lo stesso controllo amministrativo, queste formano un dominio amministrativo, e non vi è un numero limite per le zone che possono risiedere sotto lo stesso dominio.

5.1 Il terminale

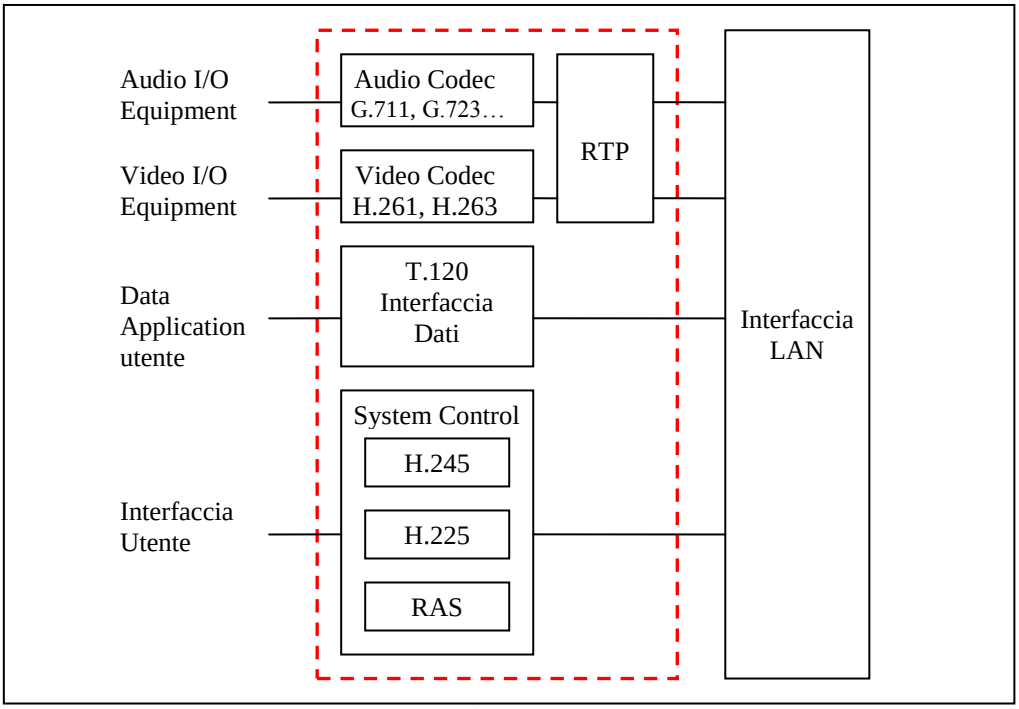

Figura 5. 3: i componenti cha fanno parte dello stack di un terminale

I terminali sono usati nella comunicazione multimediale bi-direzionale in tempo reale, e possono essere sia dispositivi stand-alone che applicazioni multimediali eseguite su personal computer. H.323 specifica le operazioni richieste che consentono a diversi terminali di operare insieme. In quanto il servizio basico dello standard H.323 è l' audio comunicazione, un endpoint deve supportare obbligatoriamente il formato audio. Opzionale è la capacità di supportare video e data streaming. Il principale obiettivo dell'H.323 è quello di riuscire a collaborare con altri terminali multimediali. I terminali H.323 sono compatibili con i protocolli H.324 nelle reti SCN e wireless, H.321 e H.310 nelle B-ISDN, H.320 nelle ISDN e H.322 nelle LANs.

Dal punto di vista strutturale i terminali devono obbligatoriamente supportare:

- H.245, per lo scambio di capacità trasmissive fra terminali e la creazione di un canale di comunicazione;
- H.225.0, per la segnalazione di setup della chiamata;
- RTP/RTCP per la sequenza di pacchetti in real-time;

- il codec audio G.711;
- RAS, per la registrazione ed altre funzioni di controllo con il gatekeeper

Facendo sempre riferimento alla figura, osserviamo che ogni terminale H.323 inoltre devono essere provvisti di un'unità di controllo del sistema, di un unità di rete ed infine di un unità di codifica/decodifica audio.
Componenti opzionali in un terminale H.323 sono il video codec, il protocollo T.120 (UserDataInterface) per la data-conferencing, e MCU.

Un terminale H.323 quindi è un entità capace di fornire, trasmissione e/o ricezione dell' audio, opzionalmente del video e dei dati in uno scenario di utilizzo sia punto-punto che all'interno di una conferenza multipunto.

5.2 Gateway

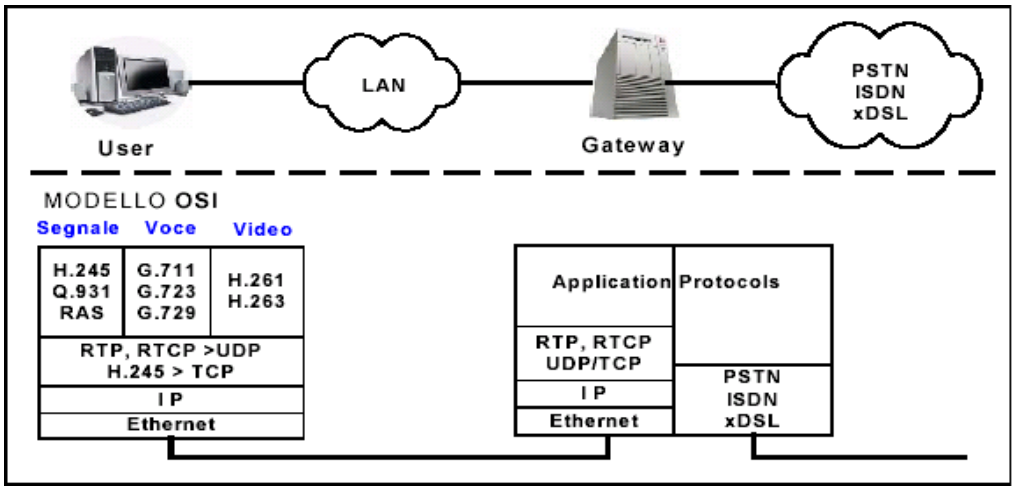

Figura 5.4: un gateway traduce segnali H.323 in segnali per reti diverse

Un gateway, nella conferenza mediante H.323, è un componente opzionale, in quanto non necessariamente richiesto nelle comunicazioni tra terminali appartenenti a reti H.323.

Un gateway fornisce diverse funzionalità, ma la più importante è la funzionalità di traduzione nelle comunicazioni tra due sotto reti diverse. Esso connette due reti eterogenee, stabilendo connessione tra una rete H.323 ed una rete non-H.323. Ad esempio un gateway può connettere e provvedere alla comunicazione tra una LAN e una rete SCN(le reti SCN includono tutte le reti a commutazione di circuito, come ad esempio Public Switched Telephone Network[PSTN]) interfacciando le caratteristiche dell'endpoint della rete H.323 verso l'endpoint della rete SCN. Questa connettività di reti eterogenee include la traduzione tra formati di trasmissione(es. da H.225.0 a H.221) e tra procedure di comunicazione(es. da H.245 a H.242). Inoltre, un gateway potrebbe effettuare i setup delle chiamate e fungere da traduttore tra codecs. La traduzione di audio e video potrebbe non essere richiesta se i terminali trovano un modo di comunicazione comune. Ad esempio nel gateway per un

terminale H.320 di una ISDN, entrambi i tipi di terminali richiedono G.711 per l'audio e H.261 per il video, quindi una comune via esiste sempre.
In generale l'obiettivo di un Gateway è quello di riflettere le caratteristiche di un endpoint appartenente ad una LAN ad un endpoint SCN e vice versa. Tra i principali scopi di un gateway troviamo:

- Stabilire i collegamenti con terminali analogici PSTN;
- Stabilire i collegamenti con i terminali H.320 su una rete commutata basata su ISDN;
- Stabilire i collegamenti con i terminali H.324 attraverso reti PSTN;
- Stabilire collegamenti con terminali H.321-compatibili sulla rete xDSL.

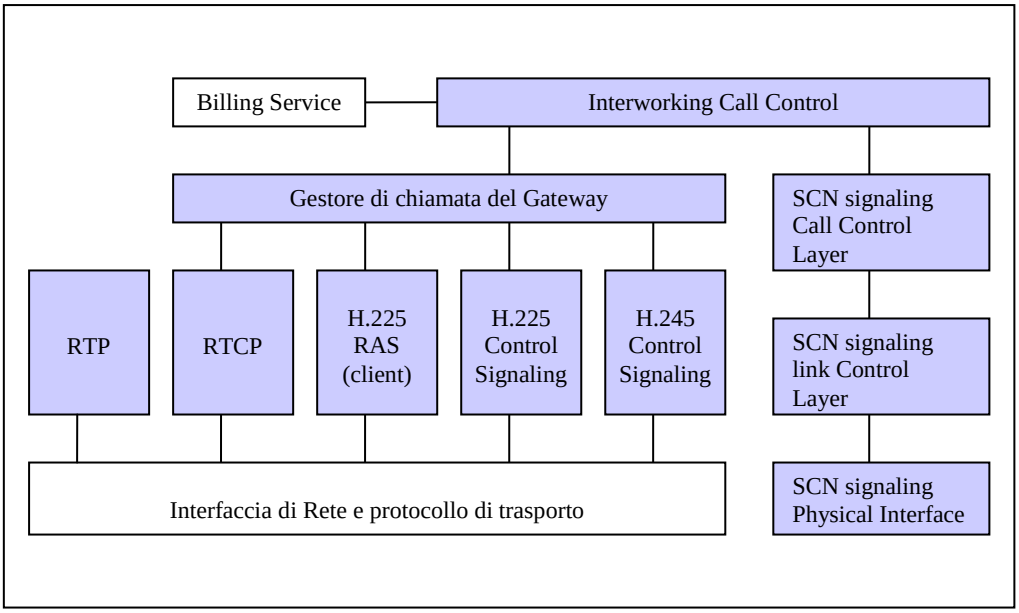

Figura 5. 5: Stack di un gateway

Dal lato H.323, un gateway esegue l'H.245 control signaling per lo scambio di capacità, H.225 call signaling per il setup di chiamata ed il rilascio e H.225 RAS per la registrazione con i gatekeeper. Dal lato SCN invece il gateway esegue i protocolli specifici per la tipologia della rete.(es. ISDN e protocollo SS7).

I terminali quindi comunicano con il gateway attraverso i protocolli H.245 e H.225; è di quest'ultimo il compito di tradurre tali protocolli in un formato trasparente alla rispettiva controparte nella rete non H.323. Un gateway ha quindi le caratteristiche di entrambe le reti di cui fa da ponte. Un gateway è capace di supportare diverse chiamate simultanee tra il terminale H.323 e le reti non H.323.

Un gateway è un componente logico del H.323, è per questo potrebbe essere implementato come parte di un gatekeeper o di un MCU.
E' composto da due elementi, il Media Gateway Control (MGC), il quale gestisce i messaggi di controllo ed altre segnalazioni non relativi ai media, e il Media Gateway (MG) che traduce invece tutti i flussi, se necessario.
Il design dei gateway è lasciata agli implementatori. Per esempio il numero di terminali H.323 che possono comunicare attraverso il gateway, non è soggetto a standardizzazione. Similarmente il numero di connessioni SCN, il numero di conferenze indipendenti simultanee, le funzioni per le conversioni data/video/audio, e le inclusioni di funzioni multipoint, sono ampiamente personalizzabili.

I gateway non sono richiesti se non si incontra la necessità di connettersi ad altre reti, dato che gli endpoint potrebbero comunicare direttamente con altri endpoints della stessa natura.

5.3 Gatekeeper

Un gatekeeper può essere considerato come il cervello della rete; nonostante non sia obbligatoriamente richiesto un gatekeeper funge da punto centrale per tutte le chiamate entro la sua zona e fornisce importanti servizi. Per come agisce, un gatekeeper può essere paragonato ad uno switch virtuale.

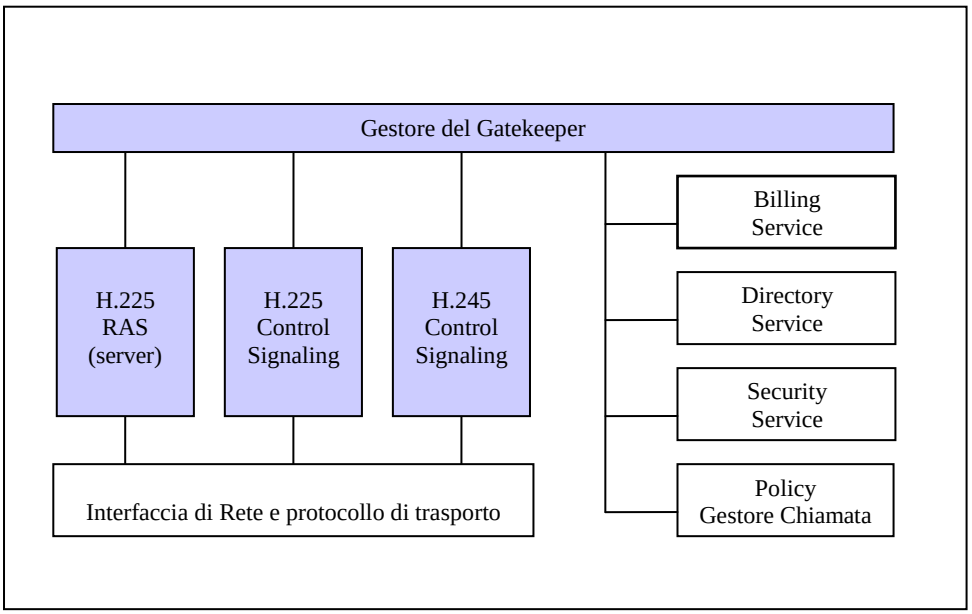

Figura 5.6: Stack di un gatekeeper

I gatekeeper fondamentalmente eseguono due importanti funzioni di controllo sulla chiamata. La prima è la traduzione di indirizzi per i terminali e gateway in indirizzi IP o IPX, come specificato dal RAS. La seconda funzione è la gestione della larghezza di banda, anch'essa disegnata nel RAS. Ad esempio, se un gestore di rete ha specificato un limite per le video o audio conferenze simultanee nella LAN, un gatekeeper può rifiutare ulteriori connessioni qualora il limite sia raggiunto. L'effetto è quello di limitare l'utilizzo della larghezza di banda destinato alle conferenze, in modo tale da permettere ad altre applicazioni, quali e-mail, trasferimento file, o qualsiasi altro protocollo LAN, di utilizzare la rimanente capacità di rete.

Una funzione opzionale, ma largamente utilizzata è la abilità di instradare chiamate H.323. Gli endpoints inviano un messaggio di call-signaling ad un gatekeeper, il quale indirizza successivamente all'endpoint di destinazione. Questa abilità del gatekeeper è anche utile per tener traccia delle chiamate, in quanto un gatekeeper fornisce un miglior controllo sulle chiamate della rete. In oltre è largamente utilizzata per la fatturazione delle chiamate(ad esempio dai fornitori di servizi quali telefonia IP). Il routing fornito da un gatekeeper potrebbe anche essere utilizzato per il trasferimento di chiamata, qualora l'endpoint non fosse disponibile, sottoforma di re-routing.
Con l'ausilio di un gatekeeper, il routing avrà prestazioni decisamente migliori, anche per la sua abilità di inoltrare chiamate ad un altro gatekeeper gerarchicamente inferiore. Questo in quanto un gatekeeper ha visione di alcuni parametri che esulano dallo stato network, per esempio i livelli di QoS richiesti dall'utente.

I gatekeeper potrebbero anche avere un ruolo nella connessione multipoint. Per supportare la multiconferenza, utenti potrebbero impiegare un gatekeeper per ricevere H.245 Control Channel da due terminali in modalità point-to-point. Quando poi la conferenza cambia in multipoint, il gatekeeper può ridirigere la segnalazione H.245 ad un multipoint controller(MC). Il gatekeeper non necessita quindi di elaborare i segnali H.245, ma ha solo bisogno di passarli tra terminarli o tra terminale e MCU.

5.3.1 Funzioni Obbligatorie

Address translation

Le chiamate originate all'interno di una rete H.323 potrebbero utilizzare un alias per l'indirizzo del terminale di destinazione. Chiamate originate fuori dalla rete h.323 e ricevute da un gateway potrebbero utilizzare un numero di telefono E.164 per indirizzare la chiamata al terminale di destinazione. Il gatekeeper traduce tale alias o numero telefonico nel corrispondente indirizzo di rete del terminale di destinazione(ad esempio per una rete basata su IP 192.168.0.16:456) L'endpoint di destinazione può essere raggiunto poi, utilizzando il suo l'indirizzo di rete nella rete H.323

Admission Control

Il gatekeeper può controllare l'ammissione degli endpoints nella rete H.323. Questo avviene tramite i messaggi del RAS che può rispondere ad un messaggio di richiesta di ammissione(ARQ), confermando all'endpoint l'ammissione alla rete(ACF) o rifiutandola(ARJ). L'accesso può essere basato su autorizzazioni di chiamata, parametri di banda richiesta o altri criteri, ma potrebbe anche essere una funzione nulla che ammette tutte le richieste alla LAN.

Bandwidth Control

Il gatekeeper fornisce un supporto per il controllo della larghezza di banda, utilizzando i messaggi RAS bandwidth request(BRQ), bandwidth confirm(BCF) e bandwidth reject(BRJ). Ad esempio, se il gestore di rete ha specificato un limite per il numero di connessioni simultanee alla rete H.323, il gatekeeper può rifiutare la richiesta di accesso ad ulteriori terminali, qualora tale limite sia raggiunto. Il risultato è quello di limitare la larghezza di banda utilizzata ad una frazione di quella effettivamente disponibile, lasciando il resto per altre applicazioni. Il controllo della larghezza di banda potrebbe anche essere una funzione nulla che accetta tutte le richiesta di cambiamento della larghezza di banda.

Zone Management

Il gatekeeper fornisce le appena citate funzionalità per terminali, gateway e MCU che sono registrati entro la sua zona H.323, così gestendola per intero.

5.3.2 Funzioni opzionali

Call-Control Signaling

I gatekeeper possono stabilire il percorso dei messaggi di call-signaling tra endpoints H.323. In una conferenza punto a punto, il gatekeeper potrebbe elaborare i messaggi h.225 per la segnalazione di chiamata. In alternativa un GK potrebbe permettere agli endpoints di inviarsi direttamente messaggi H.225 l'un l'altro.

Call Authorization

Un terminale che volesse chiamare un altro terminale, che però risulterà essere registrato ad un gatekeeper, deve far passare prima il messaggio di segnalazione di chiamata H.225 attraverso di esso. Questo può decidere per conto del terminale se accettare i meno la chiamata, in base a diversi criteri. Le motivazioni di rifiuto includono restrizione di tempo, restrizioni di banda e restrizioni di accesso da e verso determinati termali o gatekeeper.

Call Management

Il gatekeeper potrebbe mantenere informazioni relative tutte le chiamate attive, così che può indicare quali terminali sono occupati, re-indirizzare chiamate e può inoltre controllare la sua zona, fornendo informazioni alla funzione di gestione della larghezza di banda. Il gatekeeper quindi potrebbe svolgere funzionalità di routing, le quali avvengono in due modi:

- Routing della segnalazione di chiamata. Il Gatekeeper si fa carico dell'inoltro agli endpoint della segnalazione di chiamata (figura 5.5). Questo permette, ad esempio, a un service provider di monitorare le chiamate sulla rete a scopi di fatturazione, oppure di inoltrare chiamate ad altri endpoint ove quelli chiamati non siano disponibili.

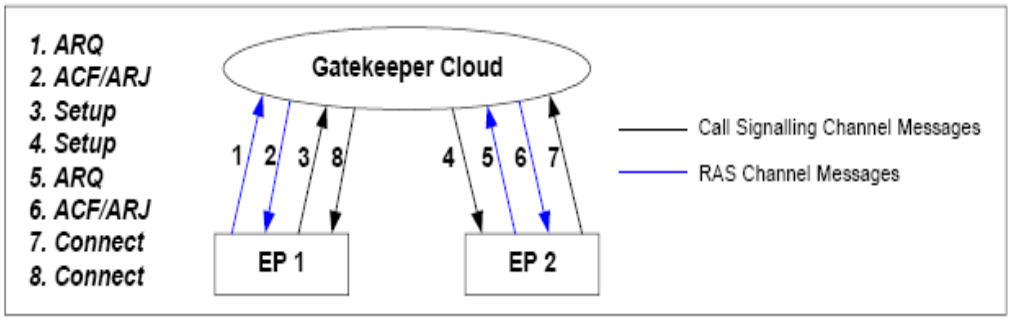

Figura 5.7: segnalazione di chiamata attraverso un GK

- Routing del controllo di chiamata. Il canale di controllo H.245, invece di essere stabilito tra i due endpoint, passa attraverso il gatekeeper. Questo permette al gatekeeper, ad esempio, di ridirigere i canali di controllo H.245 di una conferenza punto-punto ad un Media Controller, nel momento in cui si voglia farla diventare punto-multipunto.

Un gatekeeper all'interno di un sistema H.323 è opzionale. I servizi da lui offerti sono definiti dal RAS e includono traduzione di indirizzo, controllo della larghezza di banda e gestione della zona.

In realtà, non si deve pensare ad Gatekeeper come un singolo apparato, in quanto sue le funzionalità possono essere opzionalmente distribuite nella rete su più apparati: per esempio, ci possono essere più apparati che offrono in maniera distribuita le funzioni legate alla segnalazione RAS; in questo caso ogni apparato viene indicato come Alternate Gatekeeper.

5.4 MCU

Uno dei punti di forza della Raccomandazione H.323 è il supporto che fornisce alle applicazioni multimediali il multipunto. Il componente fondamentale per questo aspetto è la Multipoint Control Unit, la quale permette di costruire architetture di comunicazioni molto flessibili e facilmente scalabili.

Se da una parte può essere considerato come un terminale vero e proprio, in quanto può generare e terminare flussi audio e video, dall'altra la MCU si differenzia dai terminali su come interagisce con tali flussi.

Infatti, la MCU riceve tutti i flussi audio e video dai partecipanti alla conferenza e restituisce, a ciascuno, un unico flusso derivante dal miraggio dei contributi di ognuno.

In questo flusso trasmesso, le varie comunicazioni audio vengono sovrapposte come nella realtà, mentre per quel che riguarda il video, viene spedita un'immagine di formato standard, con al suo interno riquadri più piccoli contenenti il video acquisito dai partecipanti alla videoconferenza.

Inoltre la MCU può accordarsi con ogni utente relativamente alla trasmissione di una tipologia di streaming diverso da quello inviato da altri utenti, poiché talvolta è in grado di convertire in formati diversi, i flussi relativi ai diversi partecipanti.

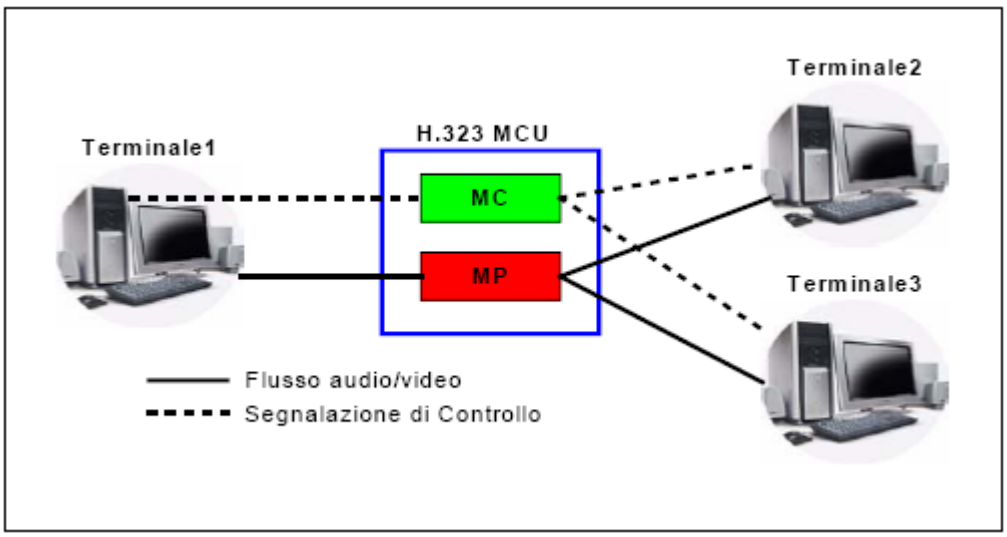

Figura 5.8: struttura logica di un MCU H.323

Nell'ambito delle reti H.323 la MCU è formato da un Multipoint Controller(obbligatorio) e potrebbe avere uno o più Multipoint Processor. E' indispensabile qualora nella multiconferenza non sia presente almeno un altro MC.

5.4.1 Multipoint Controller (MC)

Permette la conferenza con tre o più partecipanti. Gestisce le negoziazioni H.245 tra i terminali e determinare comuni capacità per elaborare audio e video(stabilisce i codecs comuni a tutti partecipanti). Può determinare la capacità di banda sostenibile dai terminali e le loro capacità trasmissive(capabilty exchange). Inoltre un MC ha il compito di controllare le risorse della conferenza determinando quali stream di audio e video sono multicast. Durante una conferenza solo un MC può essere attivo, vedremo in seguito come questo viene determinato.

5.4.2 Multipoint Processor (MP)

Ha il compito di processare i flussi multimediali in un unico flusso di output, sotto controllo dell'MC. Non è specificato dallo standard come questo controllo avvenga. Il Multipoint Processor quindi crea lo stream video contenente tutti o meno (in base alla modalità di videoconferenza a cui si partecipa) i flussi video catturati dai terminali e miscela i flussi audio in uno, per poi ridistribuirli ai partecipanti della videoconferenza.
Le capacità fornite dall'MC e dall'MP potrebbero essere integrate in componenti dedicati o essere parte degli altri componenti H.323. Proprio per questa flessibilità ed il fatto che la MCU stessa può essere considerata un terminale, ha reso possibile una varietà di configurazioni.

Detto questo si potrebbe anche dedurre che un comune MP potrebbe non essere obbligatorio, qualora i partecipanti alla conferenza decidano di provvedere loro stessi all'elaborazioni dei flussi video e audio.

5.4.3 Conferenze Multipoint

Lo standard H.323 non prevede alcuna imposizione circa la strutturazione e l'implementazione delle conferenze multipunto. Infatti, le funzionalità di supporto unicast e multicast messe a disposizione dai protocolli di trasporto H.323 sulle reti IP, ha reso possibile l'implementazione di diversi tipi di conferenza:

Centralizzate. Tutti ti terminali inviano i propri stream audio, video, data e di controllo all'MCU in modalità point-to-point. Il multipoint controller gestisce centralmente la conferenza utilizzando le funzioni di controllo del protocollo H.245, così definendo le capacità di ciascun terminale. E' poi del multipoint processor il compito di processare gli streams ricevuti, combinando i flussi audio e video in uno solo, e distribuendo i dati inviando i risultati a tutti i terminali partecipanti della videoconferenza. L'MP potrebbe anche provvedere alla conversione tra differenti codecs e bit rates.
Un vantaggio della conferenza centralizzata è che tutti i terminali H.323 supportano la comunicazione point-to-point. La MCU restituisce l'output in modalità unicast a ciascun partecipante alla conferenza, non richiedendo così speciali caratteristiche dalla rete. In alternativa però, l'MCU potrebbe ricevere i flussi in unicast parallelamente, elaborare i dati, e restituire l'output in multicast, guadagnando così sulla larghezza di banda.

Decentralizzata. Fa uso della tecnologia multicast. I terminali H.323 partecipanti alla conferenza, inviano ad un indirizzo multicast audio e video senza spedirli esplicitamente ad un MCU. Comunque, il controllo del multipoint è ancora processato centralmente da un MCU, e le informazioni sulle capacità sono ancora trasmesse in modalità point-to-point ad un MC.
I terminali saranno di seguito responsabili di processare il mix di audio e video ricevuti, utilizzano il canale di controllo H.245 per indicare ad un MC quanti video e audio streams possono simultaneamente decodificare. Il numero di trasferimenti simultanei entranti non limita il numero di streams video e audio che possono essere inviati in multicast in una conferenza.

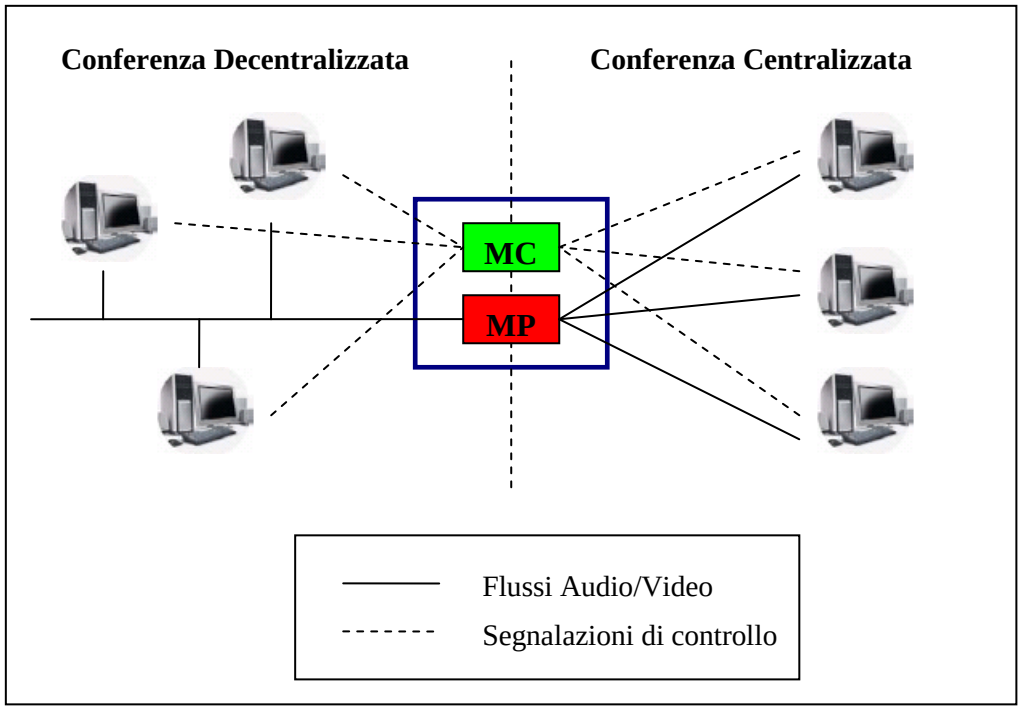

Figura 5.9: differenza tra conferenza centralizzata e decentralizzata

Ibrida. Usa una combinazione di caratteristiche appartenenti sia alla modalità centralizzata che decentralizzata. I segnali di controllo H.245 e uno tra i due streams, audio o video, vengono inviati point-to-point ed elaborati dall'MCU,

mentre il resto dei segnali è trasmesso direttamente ai terminali H.323 partecipanti della conferenza in multicast.

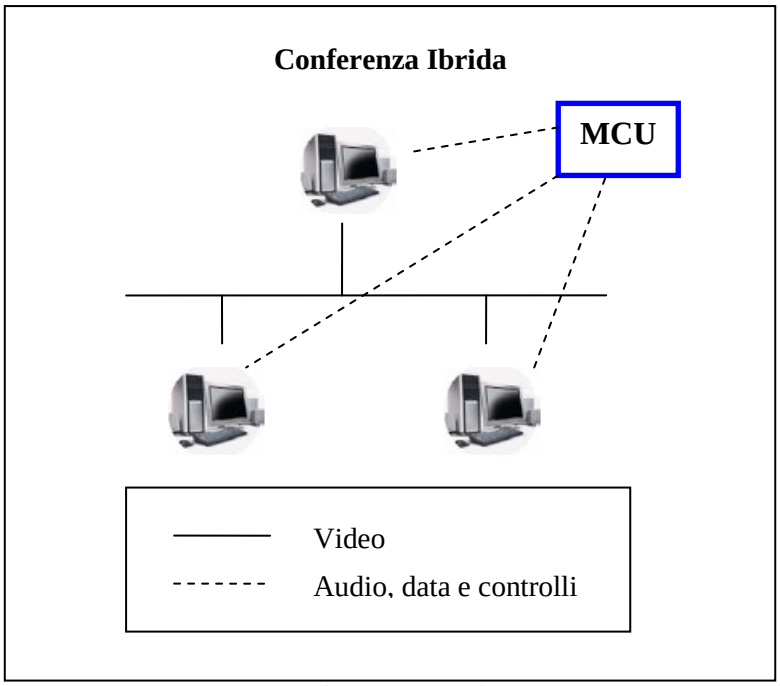

Figura 5.10: una possibile configurazione di conferenza ibrida. Audio, dati e segnalazioni di controllo sono centralizzati, mentre il video decentralizzato

Mista. H.323 in oltre, supporta la conferenza multipoint mista, nella quale alcuni terminali adottano la tipologia di conferenza centralizzata, mentre altri utilizzano la tipologia decentralizzata, mentre l'MCU fa da ponte tra i due tipi. Il terminale non sarà consapevole della natura mista della conferenza, ma solo del modo in cui esso invia e riceve.

Supportando entrambi gli approcci, sia multicast che unicast, l'H.323 abbraccia le correnti generazioni di multimedia e le future tecnologie nelle reti. Il multicasting fa un più efficiente uso della banda di rete, ma impone un più alto carico computazionale ai terminali, che hanno il compito di elaborare i dati, combinando gli stream ricevuti e smistando i propri.

Come esempio consideriamo una conferenza multipoint istituita tra tre client. Il terminale di uno dei client(in questo caso il B) esegue le funzioni MC. Tutti

i terminali potranno utilizzare multicast per partecipare ad una conferenza decentralizzata. Una funzione MP in ciascun nodo combinerà gli stream entranti e fornirà al client audio e video. Questo approccio minimizza il bisogno di specializzate risorse di rete. Comunque la rete deve essere configurata per supportare multicast.

In alternativa, un MCU separato potrebbe essere utilizzato per gestire l'audio, i dati e le funzioni di controllo. Con questa configurazione il video rimarrebbe in multicast, mentre gli altri streams sarebbero inviati in unicast all'MCU, così conservando la larghezza di banda. Tale MCU potrebbe essere un sistema dedicato oppure un terminale. Comunque sia, le conferenze multipunto che attraversano terminali in una LAN e fuori della rete, traggono enormi benefici da una configurazione dove le funzioni dell'MCU sono integrate con il gateway.

6 Protocolli

Come già detto, la raccomandazione H.323 viene detta raccomandazione 'ombrello' in quanto, piuttosto che introdurre nuovi protocolli o nuovi tipi di codifiche, tenta di armonizzare le indicazioni delle precedenti raccomandazioni della serie H., e di prevedere inoltre una famiglia di gateway idonei a garantire l'interoperabilità di queste con le entità di videotelefonia in internet, o comunque con reti e/o sistemi di telecomunicazioni di diverso tipo. La Raccomandazione H.323 comprende quindi un certo numero di documenti correlati, anch'essi pubblicati in forma di raccomandazione, ognuno concerne uno specifico aspetto della comunicazione (vedi figura 5.1).
Tra i protocolli appartenenti alla raccomandazione H.323, quelli che saranno oggetto di una analisi particolarmente attenta saranno i seguenti:

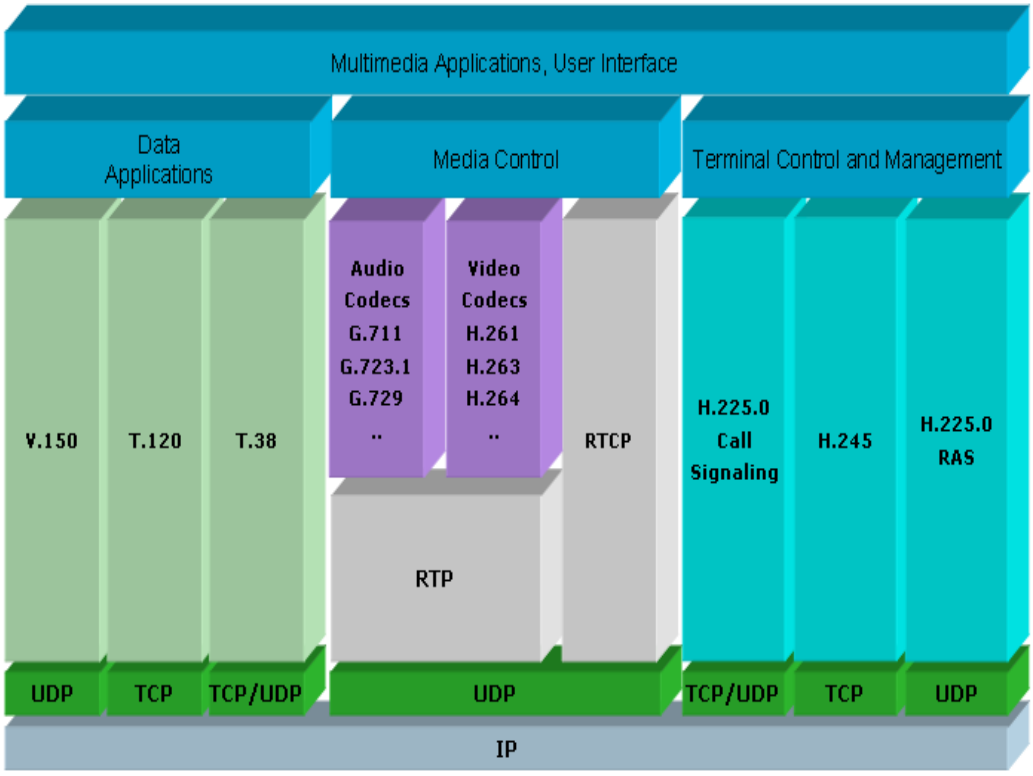

Figura 6.1: lo stack H.323 è indipendente dalla rete a pacchetti e i protocolli di trasporto sopra al quale opera, e non li specifica

Audio Codecs, adibito alla funzione di codifica/decodifica del segnale audio;

Video Codecs, adibito alla funzione di codifica/decodifica del segnale video;
H.225 RAS, protocollo per la registrazione, ammissione ai servizi e controllo di stato;
H.225/Q.931 call signaling, protocolli per la segnalazione di chiamata;
H.245 control signaling, negoziazione dei canali e scambio di capacità;
RTP/RTCP, protocollo per la trasmissione degli streams in tempo reale.

La pila protocollare definita da H.323 deve operare sempre al di sopra dello strato di trasporto della rete sottostante, in modo tale che i protocolli H.323 possano essere usati con una qualsiasi rete a pacchetto.
Per quanto riguarda il livello di trasporto su cui essi si appoggiano dobbiamo fare una distinzione.
H.323 utilizza sia trasmissioni affidabili che trasmissioni inaffidabili. Questo potrebbe essere spiegato dal fatto che mentre segnali di controllo e dati non possono permettersi la perdita di nessun byte e non possono perdere l'ordine di invio dei pacchetti(due fattori che richiedono costanti cicli di controllo che comportano l'aumento del tempo di trasmissione), gli stream multimediali potrebbero perdere valore nel tempo, ovvero un pacchetto in ritardo potrebbe non avere più alcuna rilevanza per l'utente e la perdita di qualche byte talvolta può non essere così grave, a meno che ciò non provochi gap nella riproduzione di una certa dimensione.
Per questo motivo quindi, H.323 utilizza una connessione sicura per la trasmissione di dati e di segnali di controllo e ricorre all'ausilio di una connessione inaffidabile, ma più efficiente, per il trasporto di audio e video.
La trasmissione affidabile di un messaggio utilizza una modalità orientata alla connessione, e nello stack IP, questo tipo di trasmissione e realizzato tramite il protocollo TCP.
Le trasmissioni affidabili garantiscono il controllo degli errori, del flusso e della sequenzalizzazione dei pacchetti inviati, ma a suo discapito potrebbe ritardare la trasmissione e ridurre il throughput, ovvero la quantità di dati

immessi. H.323 si avvale del TCP per i servizi end-to-end segnalazione di chiamata, di H.245 Control Channel e di trasferimento di data (T.120).
Per quanto riguarda la trasmissione inaffidabile, UDP è il protocollo dello stack IP che provvede a tale servizio. Una trasmissione via UDP è una trasmissione connection-less (senza connessione), che non garantisce niente più del "massimo sforzo" di consegna, in quanto non offre nessun servizio di controllo. H.323 utilizza UDP per la trasmissione di audio, video e per il canale RAS.

6.1 Protocolli per il set-up e la gestione della chiamata

I protocolli inerenti il setup di chiamata e la successiva negoziazione e gestione della chiamata comprendono tutti qui messaggi scambiati prima che i canali logici vengano aperti e gli la trasmissione dei flussi abbia inizio. Individuiamo quindi i protocolli Q.931 e H.225 per la segnalazione di chiamata e H.245 per il controllo di chiamata.

6.1.1 H.225 Call signaling

H.225 call signaling è il protocollo che definisce i messaggi scambiati per stabilire le fasi della chiamata tra endpoints H.323 (terminali e gateways). Deriva dal protocollo Q.931(call signaling per le chiamate attraverso ISDN) ed è stato modificato per renderlo adatto su reti a pacchetti.
H.225.0 descrive come audio, video, data e informazioni di controllo debbano essere gestite in una rete a pacchetti, e fornisce servizi di conversazione tra terminali H.323. La seguente illustrazione, descrive la struttura di un pacchetto H.225:

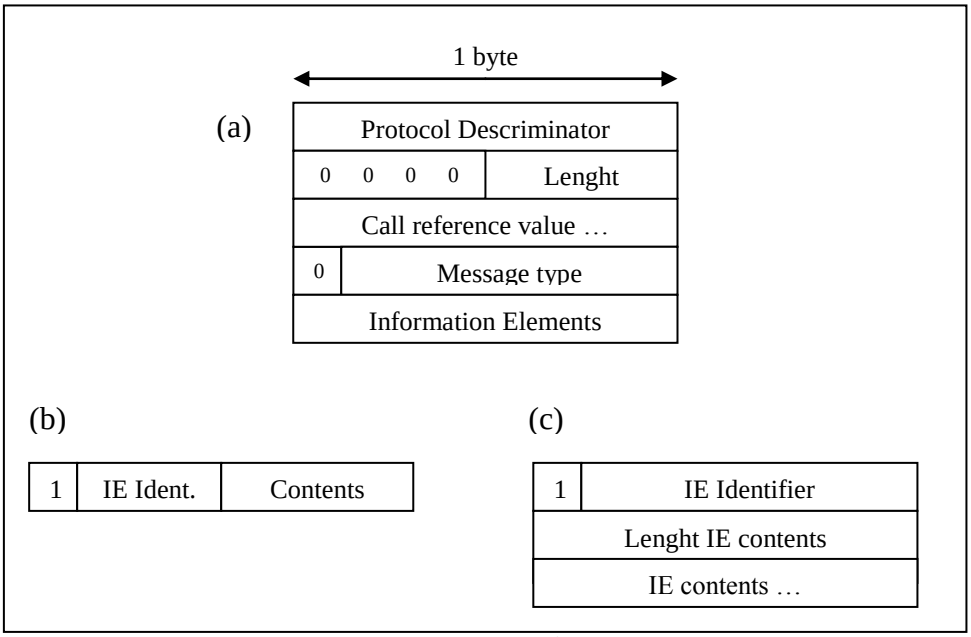

Figura 6.2: (a) Struttura di un pacchetto h.225. (b) Information element su singolo ottetto. (c) Information Element a lunghezza variabile.

Andiamo ora ad analizzare il significato di ciascun campo:

- **Protocol discriminator**: è il discriminatore del protocollo, un identificatore, e distingue i messaggi utente per il controllo di chiamata da altri messaggi.

- **Lenght è la lunghezza**: del valore riferito alla chiamata(call reference value).

- **Call Reference Value**: identifica la chiamata o facility registration/ cancellation richiesti all'interfaccia rete dell'utente. Potrebbe essere più di 2 bytes.

- **Message Type**: identifica la funzione del messaggio H.225 inviato; in tale campo potremmo avere i seguenti valori:

Relativi l'instaurazione della chiamata	Informazioni di chiamata
00000001 Alerting	00100110 Resume
00000010 Call Proceeding	00101110 Resume Aknowledge
00000111 Connect	00100010 Resume Reject
00001111 Connect Aknowloedge	00100101 Suspend
00000011 Progress	00101101 Suspend Aknowledge
00000101 Setup	00100001 Suspend Reject
00001101 Setup aknowledge	00100000 User information
Messaggi di vario genere	**Relativi l'abbattimento di chiamata**
00000000 Segment	01000101 Disconnect
00011001 Congestion Control	01001101 Release
00011011 Information	01011011 Release Complete
00001110 Notify	01000110 Restart
00011101 Status	01001110 Restart Aknowledge
00010101 Status Enquiry	

- **Information elements**: determina la categoria dell'elemento informativo(il carico utile) contenuto nel pacchetto. Sono definite due categorie di elemento (riportate nella figura precedente), quella a singolo pacchetto(fig. b) e quella a lunghezza variabile (fig. c).

I messaggi H.225 vengono trasmessi su di un canale di callsignaling affidabile (la trasmissione si affida a TCP). Tali messaggi possono essere scambiati direttamente tra gli endpoints nel caso che non ci sia un gatekeeper. Nel caso contrario, in cui la rete prevede un gatekeeper, i messaggi possono essere sia scambiati direttamente tra endpoints(direct call signaling) che re-indirizzati agli endpoints in seguito dell'invio al gatekeeper registrato (gatekeeper-routed call signaling). Il metodo è scelto dal gatekeeper durante lo scambio di messaggi RAS con l'endpoint in fase di controllo di ammissione.

Gatekeeper-Routed Call Signaling

I messaggi di ammissione sono scambiati tra endpoints e il gatekeeper in un canale RAS.

Il gatekeeper riceve i messaggi di segnalazioni di chiamata nel canale call-signaling dall'endpoint e li inoltra al canale call-signaling dell' endpoint con la quale si desidera connettersi.

Direct Call Signaling

Durante la conferma di registrazione, il gatekeeper indica che gli endpoints possono scambiarsi direttamente messaggi di call-signaling. Tali messaggi saranno scambiati attraverso il canale di call-signaling aperto tra i due terminali.

I messaggi H.225 più comunemente scambiati tra unità sono:

- Setup: è il messaggio che indica all'endpoint che lo riceve il desiderio da parte di un altro terminale di voler stabilire una connessione; può essere risposto con un messaggio di callProceeding qualora la chiamata viene accettata, mentre da un releaseComplete nel caso opposto.

- Call Proceeding: che mette in attesa il terminale che ha inviato il messaggio di Setup, indicandogli lo stato di avanzamento dell' endpoint in connessione.

- Alerting: viene tipicamente inviato prima che la connessione venga stabilita, per avvertire un terminale dell'apertura di canali logici; a seguito di tale messaggio si troverà connect.

- Information: con il quale vengono scambiate informazioni generiche.

- Release Complete: avverte l'endpoint connesso che si sta per terminare la sessione; a seguito di questo messaggio tutti i canali logici vengono chiusi, e le trasmissioni interrotte.

- Facility: contiene le capacità trasmissive dell'endpoint, che potrebbero servire in un secondo tempo alla negoziazione h.245.

- Progress: tale messaggio potrebbe essere inviato da un endpoint o da un gatekeeper per indicare il progresso di una chiamata nel caso essa sia inoltrata attraverso SCN. Progress potrebbe essere usato per esempio per controllare se un gateway sta interoperando con la rete.

- Acknowledge: si inviano informazioni relative ad un precedente messaggio, per esempio setup aknowledge o connect aknowledge; tipicamente è inviato per far sapere il risultato di una determinata operazione o una scelta.

- Connect: si notifica al terminale con la quale si sta stabilendo la comunicazione che la connessione è stata effettuata.

La maggior parte di questi messaggi sono opzionali durante il processo di connessione. Tali messaggi intermediari, sono generalmente utilizzati per prevenire errori di timeout o fornire informazioni addizionali(per esempio perché una chiamata è fallita). Quelli strettamente necessari al fine di stabilire una chiamata sono Setup e Connect.

6.1.2 Q.931

Q.931 è un protocollo per la segnalazione di chiamata, largamente utilizzato nelle reti ISDN per il set up e la segnalazione delle chiamate. E' anche usato per stabilire chiamate H.323. In questo caso i messaggi Q.931 includono messaggi di controllo della chiamata H.225 per fornire informazioni addizionali, come informazioni relative all'indirizzo IP, altrimenti non disponibili.

6.1.3 H.245 Control Signaling

H.245 control signaling consiste nello scambio end-to-end di segnali di controllo H.245 tra gli endpoints in comunicazione a seguito dell'avvenuta connessione. Tali messaggi sono trasmessi attraverso il canale di controllo di controllo H.245, che gli endpoint aprono appositamente per lo scambio di messaggi di controllo relativi la chiamata. Il H.245 control channel è identificato dal canale logico 0, e diversamente dagli altri canali è perennemente aperto per tutti il tempo della chiamata, in fatti questo viene aperto sin a partire dalla creazione di un'istanza H.245 control channel, fino all'abbattimento della connessione tra i terminali, a patto che il setup non venga effettuato tramite H.245tunneling(l'abilità di aggregare UDP H.245 dentro messaggi H.225 per poi essere inviati attraverso il canale di call signaling).
Tale canale viene utilizzato per portare a ciascuna estremità del collegamento, i messaggi di controllo che governano le operazioni delle entità, incluso lo scambio di capacità, le segnalazioni di canale con la quale si richiede o si porta conoscenza dell'apertura e chiusura dei canali logici, la scelta dei nodi preferenziali di funzionamento, i messaggi di controllo di flusso ed altri comandi.
I messaggi scambiabili tramite il protocollo H.245 possono essere ricondotti nelle quattro categorie sotto citate:

- **Request**, richieste di azioni con risposta(es. masterSlaveDetermination, terminalCapabilitySet);
- **Response**, (es. masterSlaveDeterminationAck, terminalCapabilitySetAck);
- **Command**, richieste senza l'attesa di risposta (es. sendTerminalCapability Set);
- **Indication**, messaggi informativi(es. userInput).

Un endpoint deve aprire un H.245 Control Channel per ogni conferenza a cui partecipa; è per questo motivo che le MCU e i gatekeepers, i quali possono supportare molteplici conessioni contemporanee, sono in grado di aprire, all'occorrenza, più canali di controllo H.245.
La raccomandazione H.245 specifica una serie di entità protocollari che definiscono lo scambio di messaggi finalizzati a scopi diversi. Queste entità protocollari sono:

- Master/Slave determination
- Terminal Capability
- Logical Channel Signaling
- Request Mode
- Round Trip Delay Determination
- Maintenance Loop Signaling
- Communication Mode
- Conference Request
- TerminalID
- Command and Indication

Negoziazione delle capacità

Lo scambio di capacità è il processo nella quale i terminali si scambiano rispettivamente le proprie capacità trasmissive e ricettive (ad esempio di quali codecs sono in possesso). Tali capacità saranno quindi le capacità di un determinato terminale a trasmettere media streams o ricevere ed elaborare quelli ricevuti. Tramite il portare a conoscenza gli altri terminali delle proprie possibilità trasmissive, gli endpoints possono scegliere le modalità con la quale verrà poi svolta la comunicazione, comprendendo le modalità in termini di codecs ed i rate più adatti ad entrambe le parti per la trasmissione audio/video.

Lo scambio delle capacità di sistema avviene tramite l'invio di un messaggio di TerminalCapabilitySet, il quale può essere risposto tramite un messaggio di Acknowledge, Reject o Release.

Messaggio	Campi
TerminalCapabilitySet	sequenceNumber, protocoIdentifier, multiplexCapability, capabilitytable, capabilityDescription
TerminalCapabilitySetAck	sequenceNumber
TerminalCapabilityReject	sequenceNumber, cause
TerminalCapabilityRelease	sequenceNumber, cause

Tramite questi messaggi gli endpoint si scambiano quindi le modalità con la quale vorrebbero la comunicazione avvenisse e le alternative. Le capacità trasmissive di ciascun terminale vengono numerate ed inserite in una tabella; vengono inoltre fornite le capacità simultanee, ovvero quelle trasmissioni che possono avvenire simultaneamente nella stessa conferenza, i quali vengono rappresentati nei Capability Descriptor. Il messaggio terminalCapabilitySet, deve essere obbligatoriamente il primo messaggio da scambiare nel canale H.245.

Qualora due o più terminali vogliano dare vita ad una conferenza, subito dopo lo scambio delle capacità trasmissive, si avrà la determinazione dell'Multipoint Controller attivo. Questo dovuto al fatto che più terminali potrebbero avere al loro interno un MC e in quanto, come definito precedentemente, ogni multiconferenza può essere gestita solamente da un controllore di conferenza per volta, al fine di fuorviare conflitti è necessario determinare quello master. messaggi di Master/Slave Determination sono i seguenti:

Messaggio	Campi
MasterSlaveDetermination	terminalType, StatusDeterminationNumber
MasterSlaveDeterminationAck	decision
MasterSlaveReject	cause

Tramite lo scambio del messaggio MasterSlaveDetermination, gli endpoint si specificano che tipo di terminali essi siano tramite il campo terminalType. Questo campo contiene le caratteristiche processuali del terminale, ovvero se esso possegga un MC o un MP.

6.2 G.7xx Audio Codecs

Un codec audio codifica il segnale audio catturato dal microfono del terminale H.323 trasmittente, e decodifica il codice audio ricevuto dal terminale ricevente. Il procedimento di trasformazione dell'audio è fondamentalmente il seguente:

- prima di tutto il segnale analogico viene digitalizzato utilizzando una campionatura;
- il segnale digitalizzato potrebbe essere poi compresso, in modo tale che parti del segnale alla quale l'orecchio umano non è molto sensibile vengano scartate;
- L'informazione viene poi pacchettizzata in base alle caratteristiche del codec, raccogliendo il segnale digitale corrispondente a 20ms o 30ms in un unico pacchetto.

In quanto nell'H.323 l'audio è il servizio basico fornito, un terminale deve contenere almeno un codec audio. Il codec raccomandato è il G.711 che consente di decodificare ad una velocità di 64kbps. Altri codec audio supportati sono G.722(64, 56 e 48 kbps), G.732.1(5.3 e 6.3 kbps), G.728 (16 kbps) e G.729 (8 kbps). Di seguito riportiamo una rappresentazione dei più comuni audio codecs utilizzati:

Nome	Bit Rate (Kbps)	Frequenza di camp. (KHz)	Grandezza Frame(ms)	Note
G.711 (ITU)	64	8	Campione	Mu-Law(USA) A-Law(UE)
G.721	32	8	Campione	Obsoleto
G.722 (ITU)	64	16	Campione	Divisione della banda in 16KHz in sottobande ADPCM
G.722.1	24/32	16	20	-
G.723	24/40	8	Campione	Obsoleto
G.723.1	5.6/6.3	8	30	Obsoleto
G.726	16/24/32/40	8	Campione	Sostituisce G.721 e G.723
G.727	Variabile		Campione	ADPCM
G.728 (ITU)	16	8		Operazioni a bit rate variabile per DCME
G.729	8	8	10	Basso ritardo
MPEG	32-448	32/44.1/48	Campione	-
GSM	13	8	22.5	Basso ritardo

6.3 H.26x Video Codecs

Analogamente all'audio codec, un video codec codifica il video stream proveniente dalla camera per la trasmissione dal terminale H.323, e lo decodifica una volta inviato al terminale destinatario. Come già detto per la specifica H.323, supportare il video è opzionale e quindi un terminale non necessita obbligatoriamente di un video codec. Ma qualora la conferenza video sia una prerogativa richiesta, la codifica H.261(standard per la trasmissione di immagini in movimento) deve obbligatoriamente essere fra le codifiche supportate. Tale codec è disponibile con due formati di immagine differenti, ovvero CIF per immagini di 352x288 pixel, e QCIF per una profondità di 176x144 pixel. Un'altra importante codifica è l'H.263 che supporta trasmissioni ad un minor bit rate, ma comporta ovviamente un maggior carico di elaborazione da parte degli endpoint, dal momento che è ottenuta integrando la codifica H.261 con tecniche di predizione dei frame.
Comunque sia, i terminali che forniscono video-comunicazione devono obbligatoriamente supportare la codifica e decodifica dello stream audio.

6.4 Protocolli usati nel trasferimento in real-time

Di seguito riportiamo i protocolli che vengono utilizzati per l'invio dei dati ad uno o più endpoint. Tali protocolli sono T.120 per l'invio di dati, RTP e RTCP per l'invio degli streams audio e video. Mentre il primo è trasportato via TCP gli ultimi due si avvalgono di UDP.

6.4.1 T.120

T.120 definisce un mezzo per la condivisione di data, durante una chiamata multimediale. E' uno stack protocollare relativo alla trasmissione in tempo reale(utilizza diversi canali TCP diversamente dagli strema audio e video che utilizzano UDP) dei dati delle applicazioni da condividere. Le applicazioni potrebbero includere trasferimento di file, chat, condivisione di applicazione e lavagna grafica. Come l'H.323, anche il T.120 è una raccomandazione ombrello in quanto racchiude un insieme di standard finalizzati alla trasmissione di dati tra terminali appartenenti a reti diverse.
Vediamo alcuni vantaggi chiave della raccomandazione T.120 rispetto allo scambio di dati tradizionale:

- Supporto per le conferenze multipunto: l'MCU provvede a manipolare i flussi dati così come avviene per i flussi audio/video.
- Indipendenza dalla piattaforma di rete: il protocollo T.120 opera al di sopra dello strato di trasporto.
- Interoperabilità: il T.120 è supportato da tutti gli standard H.32X fornendo un alto grado di interoperabilità.
- Supporto per il multicast.

6.4.2 RTP

Real-time Transfer Protocol è un protocollo usato per fornire sincronismo durante la trasmissione di audio e video nelle reti a pacchetti. RTP è collocato al quinto livello della pila ISO, il livello applicazione e si appoggia al protocollo UDP (situato al livello inferiore sulla stack protocollare) per la trasmissione attraverso IP. Insieme, RTP e UTP provvedono a funzionalità di protocollo di trasporto.

Tra i vantaggi introdotti da RTP, troviamo la maggior omogeneità dei dati utilizzati dalle diverse applicazioni. Prima di tale protocollo, ogni programmatore personalizzava l'header dei datagrammi UDP in base alle proprie esigenze. In questo modo ogni programma aveva pacchetti UDP con intestazione diversa in formato e in dimensione, il che rendeva assai difficile la sua compressione da parte dei router o di altri apparati di rete. Inoltre l'invio e la ricezione di dati poteva avvenire solo tra le stesse applicazioni, il comportava un discreto problema di compatibilità.

Con RTP i dati prodotti da un programma di un vendor possono essere ricevuti senza problemi dal programma di un altro.

Oggi sono sempre più numerosi i software in grado di gestire contemporaneamente voce, musica, video e dati tradizionali sulla stessa rete: RTP si è rivelata una valida tecnologia per il trasporto di tali "mix" multimediali, e lo sarà certo ancora per un bel po' di tempo.

Parleremo in seguito in maggior dettaglio di questo protocollo, descrivendo come esso operi per garantire la trasmissione di audio e video agli utenti connessi.

6.4.3 RTCP

Real-time Transfer Control Protocol è strettamente legato a RTP e può essere visto come la sua controparte, fornendogli servizi di controllo. Tra le sue funzioni principali troviamo quella di fornire ai terminali connessi, feedback relativi alla qualità dei dati ricevuta, così comunicando proprietà di rete quali il jitter, la congestione della rete, la larghezza di banda utilizzata e altro. RTCP controlla la qualità del servizio fornito da RTP, convoglia informazioni relative alle sessioni attive, e periodicamente distribuisce dei pacchetti di controllo contenenti informazioni sulla qualità di tutte le sessioni partecipanti ma non trasporta alcun dato.
Un'altra funzione è quella di portare al livello di trasporto degli identificatori, chiamati nomi canonici(CNAME), che vengono poi utilizzati per sincronizzare audio e video. Tramite questi nomi infatti, RTCP identifica il partecipante-mittente, utile all'organizzazione delle sessioni multiple, nella quale lo stesso mittente invia più flussi.
La comunicazione di messaggi RTCP si basa tramite la trasmissione di 5 diversi tipi di pacchetti, che vengono trasmessi utilizzando UDP tra due socket dinamicamente allocati(vedi sessione RTP):

- **SR** (Sender Report): analogamente ai pacchetti RR portano informazioni relative la qualità, ma in aggiunta contengono anche una sezione relativa al mittente, che invia informazioni relativi alla sincronizzazione.
- **RR** (Receiving Report): generati dai partecipanti alla conferenza, trasportano informazioni relativi alla qualità di ricezione, specificando il numero di pacchetti ricevuti e quelli persi, la valutazione del jitter ed il ritardo di percorrenza dei pacchetti.
- **SDES** (Source Descriptor): contiene elementi di descrizione dei partecipanti, incluso il CNAME.
- **BYE**: indica che un partecipante sta per lasciare la sessione.
- **APP**: indica che un partecipante sta per unirsi alla sessione.

Queste 5 tipologie di pacchetto verranno analizzate in maggior dettaglio nel capitolo otto.

6.5 H.225 Registration, Admission, and Status

H.225 RAS costituisce il protocollo utilizzato tra endpoints H.323(terminali, MCU e gateways) e gatekeeper in modo da effettuarne la registrazione e poter utilizzare i servizi da lui forniti. Tra le sue principali finalità del RAS troviamo:

- permettere ai gatekeeper di gestire gli endpoints;
- permettere ad un endpoint di richiedere l'ammissione per una chiamata;
- permettere gli endpoint richieste relative alla larghezza di banda;
- permettere ai gatekeeper di fornire funzionalità di risoluzione di indirizzo.

Possiamo identificare tre tipologie principali di messaggi RAS:
Request, rappresentato tramite xRQ;
Confirm, rappresentato tramite xCF;
Reject, rappresentato tramite xRJ.

Tali messaggi vengono scambiati tramite un canale RAS che utilizza il protocollo di trasporto UDP. Questo canale è aperto su un endpoint indipendentemente dalla chiamata verso un altro endpoint, quindi potrebbe essere aperto prima dell'instaurazione di qualunque altro canale. I messaggi in questo canale inoltre, potrebbero essere associati a contatori di retry o timeout, per l'inaffidabilità del canale.

Gatekeeper Discovery

Il processo di scoperta di un gatekeeper è attuato dagli endpoint H.323 per determinare il gatekeeper alla quale devono registrarsi. Tale processo può essere manuale o automatico, dinamico o statico.
Nella scoperta statica, l'endpoint sa l'indirizzo di trasporto del suo gatekeeper a priori.
Mentre con il metodo dinamico di gatekeeper discovery, l'endpoint invia in multicast un messaggio di GRQ(gatekeeper request) all'indirizzo multicast per la scoperta di GK. Questo messaggio è spedito utilizzando il well-known Discovery Multicast Address dei Gatekeeper 224.0.1.41 e la porta predefinita 1718 .
Uno o più gatekeeper potrebbero rispondere alla richiesta. Generalmente se un gatekeeper non desidera fornire funzionalità all'endpoint, invierà un messaggio di GRJ. Il metodo dinamico permette costi amministrativi minori in quanto non richiede il settaggio della lista di Gatekeeper nella configurazione dell'applicativo dell'end-point; inoltre, in caso di guasto del Gatekeeper, permette la ricerca di ulteriori Gatekeeper funzionanti presso cui registrarsi.

Per una ragione di sicurezza, i gatekeeper potrebbero ignorare richieste da parte di endpoints sconosciuti. Se invece desidera provvedere funzionalità all'endpoint, il gatekeeper restituirà un messaggio GCF.
La comunicazione avviene attraverso la porta 1718 in multicast, e più comunemente attraverso la porta 1719 in unicast.

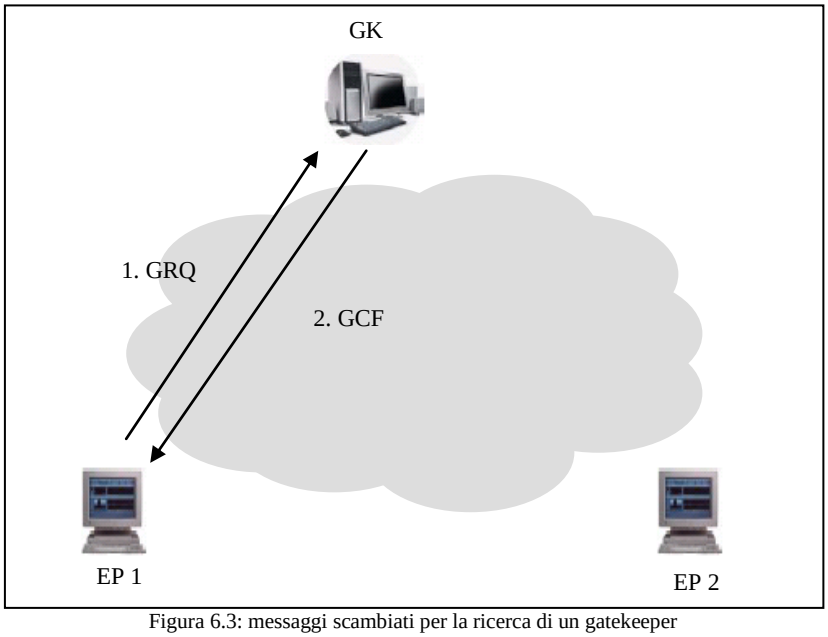

Figura 6.3: messaggi scambiati per la ricerca di un gatekeeper

Alcuni campi comuni ai tre massaggi GRQ/GRJ/GCF sono:

- requestSeqNum: fa riferimento al numero della richiesta GRQ . Ogni messaggio GRJ e GCF deve specificare a quale GRQ fa riferimento.
- protocolIdentifier: identifica il protocollo e la versione del componente trasmittente.
- IntegrityCheckValue: porta al suo interno informazioni utili all'integrità ed all'autenticazione del messaggio.

Alcuni campi specifici per ogni messaggio sono invece, oltre a quelli già citati:
- endpointAlias: nel GRQ l' end-point può informare il Gatekeeper su tutti i suoi indirizzi alternativi.
- enpointType: nel GRQ l' end-point specifica il tipo di componente H.323 esso sia.
- autenticationCapability: nel GRQ l' end-point dichiara le modalità di autenticazione supportate.

- autenticationMode: nel GCF il Gatekeeper specifica il modo di autentificazione da usare.
- rejectReason: nel GRJ il Gatekeeper specifica la ragione del rifiuto.

Endpoint Registration

Una volta che un gatekeeper è stato trovato, per far si che l'endpoint possa beneficiare dei suoi servizi deve registrarsi ad esso. Tramite la registrazione l'endpoint entrerà anche a far parte della zona H.323 definita dal gatekeeper. La comunicazione ora avviene esclusivamente alla porta 1719(unicast). La registrazione avviene tramite il messaggio di richiesta RRQ attraverso il canale RAS Channel Transport Address stabilito dal gatekeeper nella fase di ricerca. Il gatekeeper a sua volta che restituirà o un messaggio di accettazione, RCF, o un messaggio di rifiuto RRJ. Il rifiuto da parte del gatekeeper di registrare l'endpoint significa solamente che quest'ultimo non utilizzerà le sue funzionalità, non che esso non possa comunicare attraverso la rete. Durante la registrazione il gatekeeper fornirà un identificativo all'endpoint, che dovrà utilizzare per le seguenti comunicazioni con lui. E' possibile che il gatekeeper consenta la registrazione a due endpoints con alias uguale ma transport address differente, anche se in genere si tende a rifiutare le registrazioni ambigue.
Nell'eventualità che l'endpoint non fornisca un alias, questo viene attribuito automaticamente dal gatekeeper.
La controparte del RRQ è il messaggio URQ, che permette ad un endpoint da cancellare la registrazione con il gatekeeper, che a sua volta risponde con un UCF o un URJ. Il messaggio RRQ può essere anche essere inviato dal gatekeeper, in questo caso l'endpoint può solo replicare con un messaggio di conferma

Campi caratteristici del RRQ sono:
- discoveryComplete: valore booleano che informa il Gatekeeper se è stata precedentemente effettuata o meno la procedura di discovery.

- terminalType: indica il tipo di end-point che richiede la registrazione (GK,GW,MCU,Terminal)
- terminalAlias: lista di alias tramite i quali l' end-point può essere identificato.
- capacity: indica la massima capacità di chiamata dell' end-point.

Per quanto riguarda RCF, un campo interessante è il preGarantedARQ che elenca gli eventi per i quali è garantita l'ammissione al transito dal Gatekeeper a cui l'end-point è registrato.

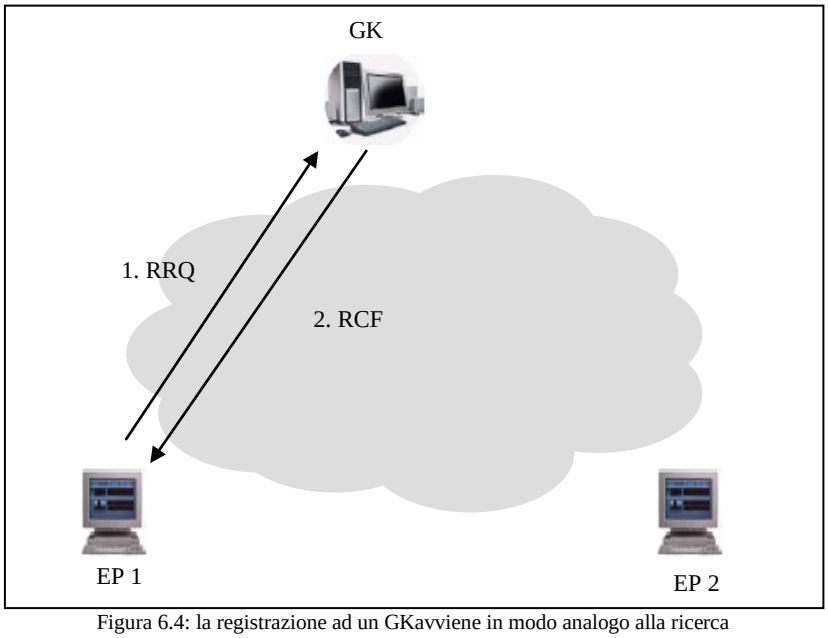

Figura 6.4: la registrazione ad un GKavviene in modo analogo alla ricerca

Admission Control

Una volta effettuata la registrazione, all'endpoint non rimarrà altro che accettare una chiamata o iniziarne una tramite la richiesta di ammissione al gatekeeper, tramite il messaggio ARQ. Il gatekeeper potrebbe rifiutare(ARJ) o accettare(ACF) la richiesta di iniziare una chiamata o riceverla. L'endpoint indicherà poi, l'indirizzo di destinazione, ed il gatekeeper restituirà se possibile(canMasAlias deve essere true), un insieme di indirizzi alternativi. Fornirà inoltre al gatekeeper un identificatore di chiamata(CallID), unico tra

endpoint e gatekeeper, un ID di conferenza se conosciuto(CID è unico nella conferenza punto a punto e condiviso con tutti i partecipanti in quella multipunto), la larghezza di banda desiderata e l'origine della chiamata, ovvero se sta effettuando una chiamata o sta ricevendone una.
Ovviamente la conferma o meno dell'ammissione alla registrazione ed il controllo della chiamata, dipende anche dalla disponibilità del destinatario a riceverla. Quindi il Gatekeeper dovrà assicurarsi che l' end-point ricevente accetti la chiamata. Questa segnalazione avviene tramite protocollo H.225/Q.931.

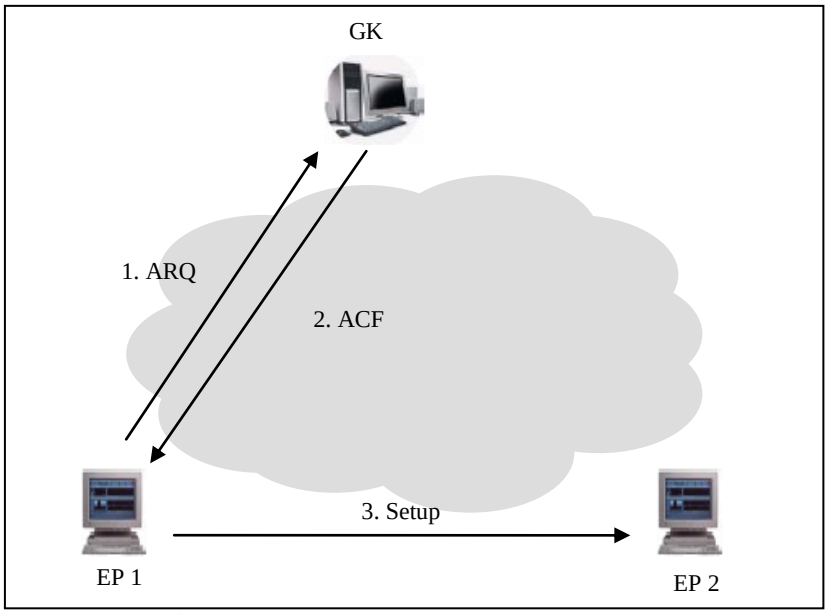

Figura 6.5: per usufruire dei servizi di un GK, un terminale deve registrarvisi tramite un ARQ

I gatekeeper possono inoltre dare ai terminali la possibilità di effettuare chiamate senza la necessità di inviare il messaggio di ARQ. Questa possibilità viene effettuata durante la registrazione RRQ, e denominata "Pre-granted Admission"

Endpoint Location

La locazione di un endpoint è il processo con la quale il si richiedono informazioni relative ad un altro endpoint, a partire dal suo alias o transport address.

Ha luogo tramite messaggi LRQ, che possono essere scambiati tra gatekeeper o tra endpoint e gatekeeper, restituendo un messaggio contenente i dati dell'endpoint richiesto(per esempio l'indirizzo relativo ad un alias che comprende commutare un numero telefonico in un indirizzo IP ecc.).

Il messaggio di LRQ potrebbe essere inviato contemporaneamente a uno o più GK, oppure in modo gerarchico ovvero inoltrato ad altri GK finché non si trova la risposta all'originale messaggio di ARQ. Per esempio un messaggio di LRQ potrebbe essere inviato da un GK1 ad un GK2, il quale non conoscendo la risposta lo inoltrerà al GK3. Quest'ultimo è in grado di definire la locazione del terminale che si vuole chiamare e invierà il messaggio di LCF direttamente a GK1 o a ritroso.

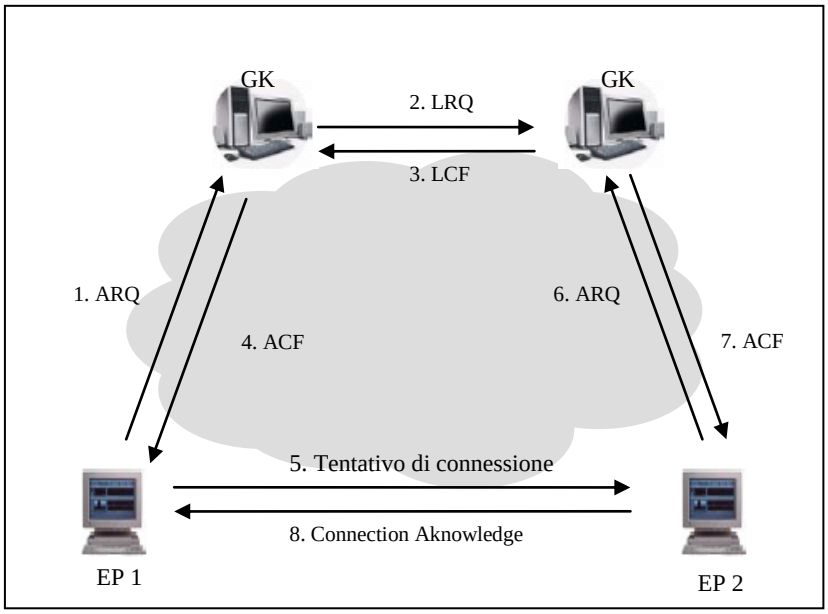

Figura 6.6: stabilimento di connessione attraverso GK

Il servizio di locazione potrebbe inoltre essere fornite tramite l'accesso a database da parte del gatekeeper, come ad esempio LDAP.

Bandwidth Control

Successivamente al setup di chiamata, un endpoint potrebbe manifestare il bisogno di più o meno banda di quella assegnatagli dal gatekeeper. Un endpoint deve inviare un messaggio di BRQ dopo che la connessione tra i terminali sia stata stabilita e qualora l'effettiva larghezza di banda sia meno di quella inizialmente richiesta. La richiesta dell'endpoint, potrebbe essere quella di diminuire la larghezza di banda, ed in questo caso il gatekeeper accetterà sicuramente. Nel caso opposto che il terminale chieda un aumento di larghezza di banda, questo verrà concesso esclusivamente se il gatekeeper è in grado di fornirlo.

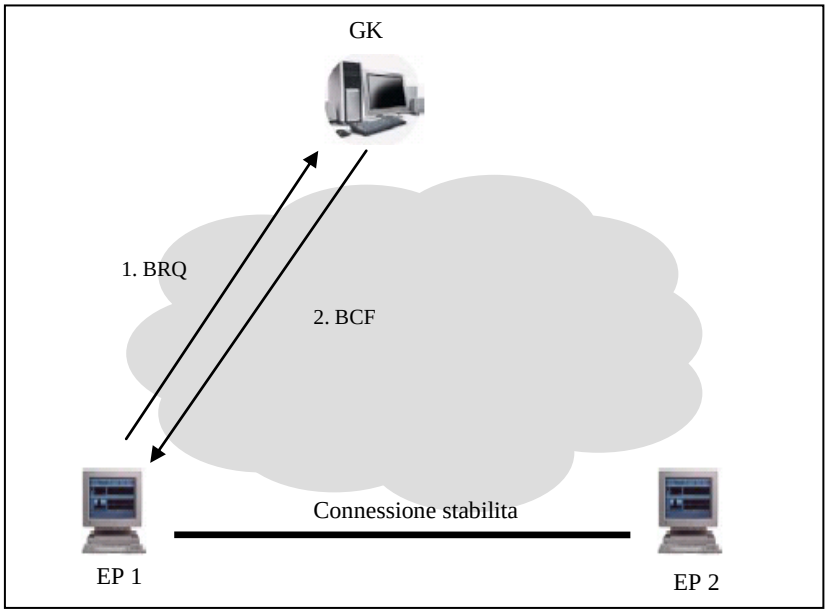

Figura 6.7: gestione della larghezza di banda tramite GK

Desengage Request

Una volta terminata la chiamata, gli endpoint inviano un messaggio DRQ al gatekeeper. A sua volta un gatekeeper replicherà con un messaggio di DCF o DRJ, anche se quest'ultima e fortemente sconsigliata(un endpoint che invia un DRQ indica che la chiamata è chiusa, quindi la richiesta di sconnessione non dovrebbe essere rifiutata). Un messaggio di DRQ potrebbe anche essere inviato da un gatekeeper per forzare l'endpoint a disconnettere la chiamata.

Altri messaggi RAS

IRQ
Inviato dai gatekeeper agli endpoints, per richiedere informazioni relative ad una o più chiamate (per esempio per sapere se si è scollegato).
La risposta a tale messaggio è IRR (information response) è conterrà tutti i dettagli richiesti della chiamata. In H.323 ci sono delle disposizioni che permettono ai gatekeeper di ricevere informazioni relative ad una chiamata periodicamente e senza richiesta.
Un gatekeeper è in grado di richiedere lo stato di qualunque terminale per venire a conoscenza dei suoi parametri di connessione alla conferenza.

RIP
Il messaggio RIP (Request In Progress) può essere inviato sia da un endpoint che da un gatekeeper qualora un messaggio RAS non possa essere risposto entro il normale tempo di elaborazione.

RAI
Il messaggio RAI (Resource Available Indicate) è inviato da un endpoint per indicare quando ha quasi raggiunto il limite di risorsa o comunque sia si sta avvicinando ad esso. Il gatekeeper risponde all'endpoint con un messaggio RAC(Resource Available Confirm)

UMR
Il messaggio UMR (unknown message request) è inviato ad un endpoint come risposta ad un messaggio non riconosciuto

NSM
Il messaggio NSM (Non-Standard Message) è utilizzato per permettere ai gatekeeper e endpoint lo scambio di messaggi che vanno oltre i canoni dei messaggi standard.

Retries e Timers dei messaggi RAS

I messaggi RAS vengono inviati via UDP, che non garantisce la consegna dei pacchetti, e per questo motivo a ciascuno di essi è stato attribuito un valore di time-out, dopo il quale il mancato riscontro dall'invio del messaggio, da parte del terminale destinatario, comporta un ulteriore tentativo, ed il numero massimo di tentativi possibili prima che la procedura fallisca. Qui di seguito riportiamo i valori di time-out e di tentativi per i messaggi RAS:

Messaggio RAS	Valore di Time-out	Tentativi
GRQ	5	2
RRQ	3	2
URQ	3	1
ARQ	5	2
BRQ	3	2
IRQ	3	1
IRR	5	2
DRQ	3	2
LRQ	5	2
RAI	3	2
SCI	3	2

7 Fasi di una comunicazione

Durante il processo di chiamate si possono individuare 5 fasi, che comprendono setup, svolgimento e chiusura. Tali fasi delineano il ciclo di vita di una chiamata e sono:

1) Setup di chiamata;
2) Scambio delle capacità trasmissive;
3) Instaurazione della comunicazione
4) Servizi della chiamata;
5) Abbattimento della chiamata.

7.1 Setup della Chiamata

Questa fase ha luogo mediante messaggi di controllo definiti nella raccomandazione H.225.0 e spediti mediante il canale di segnalazione di chiamata (call signaling).
La procedura di setup ha diversa natura, a seconda che l' endpoint chiamante e quello chiamato siano o no registrati ad un gatekeeper o dalla tipologia della conferenza; per tale flessibilità si possono avere le seguenti configurazioni:

- entrambi non registrati con nessun gatekeeper;
- entrambi registrati allo stesso Gatekeeper
- solo il chiamante (o il chiamato) è registrato ad un gatekeeper
- gli endpoints sono registrati a gatekeeper diversi.
- i due endpoint appartengono a network eterogenee (setup di Chiamata via Gateway)
- gli endpoints partecipano ad una Conferenza Multipoint (setup di chiamata via MCU)

Di seguito studieremo solo due casi, ovvero il caso in cui entrambi gli endpoint sono registrati a gatekeeper diversi ed il caso in cui entrambi gli endpoint non lo siano. Questo perché le altre configurazioni potrebbero essere viste come una via di mezzo tra queste due, e si potrebbe facilmente immaginare l'andamento della procedura.

Setup di base: nessuno dei due end-point è registrato presso un GK

Questa è la situazione più semplice in quanto il Gatekeeper non interviene ed i messaggi di controllo e segnalazione vengono scambiati direttamente tra gli end-point evitando così l'uso dei messaggi RAS.
Questo impone che l'endpoint chiamante conosca a priori l' indirizzo dell'endpoint chiamato.
Praticamente, l' end-point 1 (chiamante) spedisce un messaggio di Setup(1) verso un ben noto Call Signaling Channel (solitamente il 1720) utilizzando il TSAP Identifier dell' end-point 2 (chiamato).
L' end-point 2 a sua volta, può rispondere con un messaggio di Connect(4), il quale conterrà anche l'H.245 Control Channel Transport Address (ovvero l'indirizzo IP e la porta sulla quale il terminale chiamante si metterà in ascolto di messaggi H.245) da usare per la successiva fase di scambio delle capacità, oppure saltare direttamente alla fase di abbattimento di chiamata tramite un messaggio di fine sessione.

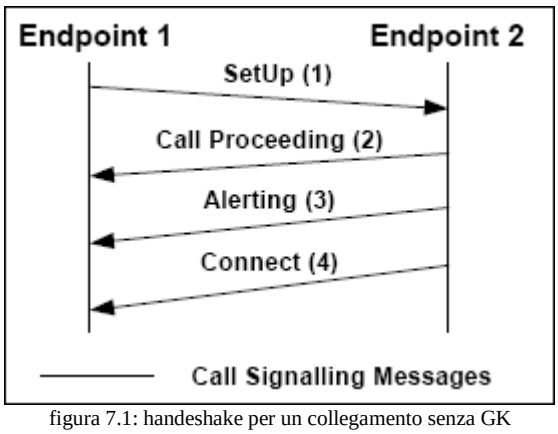

figura 7.1: handeshake per un collegamento senza GK

I messaggi Alerting e Call Proceeding hanno funzionalità trascurabili. Il messaggio di alerting ha una funzione di wait in caso il messaggio di connessione venga inviato dopo 4 secondi, mentre il secondo avverte che l'endpoint sta iniziando la procedura di connessione.

Entrambi gli end-point registrati su differenti Gatekeeper

La configurazione opposta alla precedente si ha con entrambi i terminali registrati a due gatekeeper differenti. La situazione è quella riportata nella seguente figura:

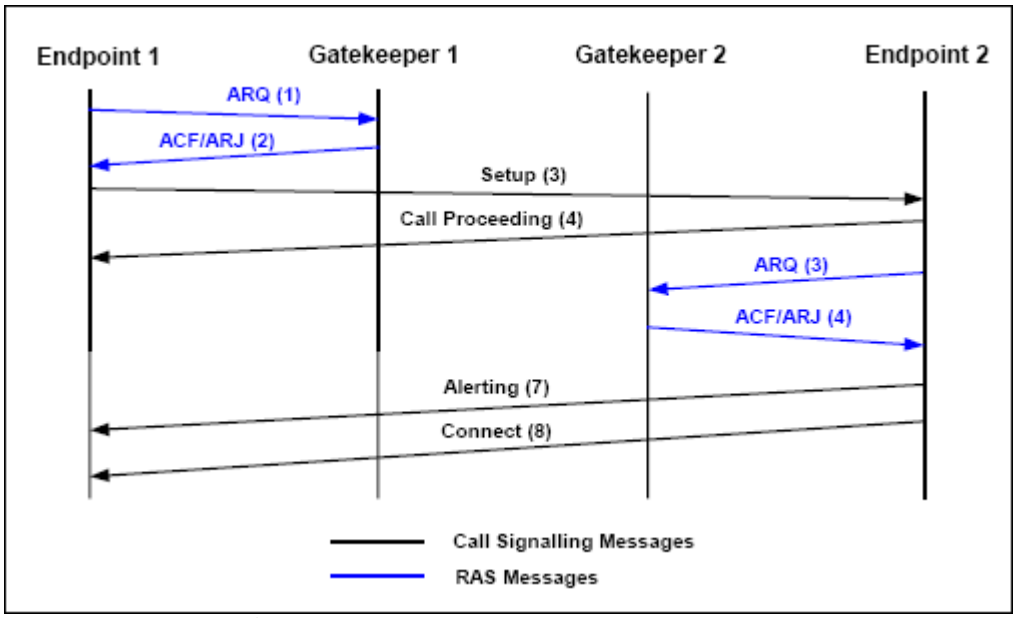

figura 7.2: messaggi scambiati per stabilire una connessione

Premessa per cui si possa avere questa configurazione è che entrambi i terminali, al loro avvio abbiano trovato due gatekeeper a cui registrarsi, e che questi abbiano accettato la richiesta di disporre dei loro servizi. A questo punto il terminale chiamante (endpoint 1) indicherà al gatekeeper la volontà di effettuare la chiamata tramite un messaggio di admission request. Questo in base ai criteri precedentemente citati (vedi capitolo 5.3.2, call authorization),

può assecondare o negare la sua richiesta, oppure offrire i suoi servizi come per esempio la risoluzione dell'indirizzo (endpoint location).
Una volta che l'endpoint 2 riceve la richiesta di connessione da parte dell'endpoint 1, questo richiede l'ammissione alla chiamata al suo gatekeeper, il quale analogamente al gatekeeper 1, può rifiutare o accettare la chiamata.
Il messaggio di Setup viene poi scambiato direttamente tra i terminali. Questo significa che durante l'invio del messaggio di ACF, i terminali hanno specificato il desiderio che la comunicazione si svolgesse in modalità diretta. Se diversamente avessero stabilito che lo scambio dei messaggi H.225 e H.245 sarebbero avvenuti in modalità routed, gli endpoints avrebbero prima inviato le segnalazioni ai rispettivi gatekeeper, i quali avrebbero ridiretto le chiamate all'endpoint destinatario.
Si chiude a questo punto la fase di instaurazione del collegamento, lasciando il posto alla successiva fase di negoziazione della qualità del servizio (QoS) desiderabile durante il collegamento.

Fast Start

Ultimamente il trand per il call signaling si è rivolto verso il fast-start o fast-connection. Questa procedura di segnalazione è stata concepita con l'idea di eliminare la fase di negoziazione H.245 inglobandola direttamente nel messaggio di call signaling H.225. In poche parole un terminale invia un solo messaggio di connessione ad un altro terminale, contenente setup, terminalCapabilitySet e apertura di canali logici. Se l'utente chiamato non supporta il fast start, o non gradisce i parametri scelti si avvierà la procedura di negoziazione, altrimenti inizierà la trasmissione considerando già aperti i canali logici proposti.

7.2 Scambio delle capacità trasmissive

Se il setup della chiamata è andato a buon fine (cioè se il terminale chiamante ha ricevuto il Connect da quello chiamato) si ha che tra i due endpoint viene aperto un canale di controllo H.245, attraverso il quale vengono scambiate segnalazioni di controllo (né viene aperto uno per ogni conferenza a cui il terminale partecipa). Tale canale infatti viene utilizzato per trasportare da un terminale all'altro, i messaggi di controllo che governano le operazioni delle entità, incluse scambio di capacità, apertura e chiusura dei canali logici, le richieste di modi preferenziali di svolgimento, i messaggi di controllo di flusso, la determinazione del master\slave ed altri comandi. Queste comunicazioni avvengono tramite la trasmissione di messaggi H.245, tra due endpoints, tra un endpoint ed un gatekeeper o tra un endpoint ed un MC. Se per esempio, facendo riferimento alla figura precedente, il terminale 1 in seguito ad una connessione avesse voluto informare il terminale 2, delle sue capacità trasmissive, avrebbe inviato un messaggio contenente le capacità di multiplexing e la tabella degli item; ciascun item avrebbe fatto riferimento ad una capacità trasmissiva del terminale in questione. L'endpoint 2 a sua volta avrebbe risposto con un aknowledge per notificare l'avvenuta ricezione del pacchetto.

Le capacità di sistema sono scambiate tramite la trasmissione dei messaggi di terminalCapabilitySet e masterSlaveDetermination descritti nel capitolo 6.1.3
Se una delle due procedure fallisce l'endpoint abbandona ed entra nell'ultima fase della chiamata, altrimenti si passa alla fase di instaurazione della comunicazione per audio e video.

H.245tunneling

Sempre nel capitolo 6.1.3 abbiamo accennato alla tecnica H.245tunneling. Questa tecnica può essere considerata la controparte del fast start, inglobando messaggi di negoziazione o controllo H.245 in pacchetti H.225. Questo permette ai terminali di effettuare lo scambio di capacità o richieste specifiche

senza aprire il canale di controllo H.245 e attraverso UDP. Nella prova di laboratorio sono riportati esempi di fast start e H.245tunneling.

7.3 Instaurazione della comunicazione

In questa fase le procedure H.245 vengono usate per aprire i canali logici atti allo scambio dei vari flussi informativi. I flussi audio e video saranno trasmessi su canali logici attraverso UDP, quindi soggetti ad errori di trasmissione. Il flusso di dati invece è sempre trasmesso su un canale logico mediante TCP. E' anche possibile durante la comunicazione cambiare le specifiche del canale logico qualora una delle due parti lo richieda, sempre tramite meccanismi di segnalazione H.245. Dunque, i due end-point devono ricontrattare le rispettive capacità, modalità di ricezione o trasmissione e così via.

Per l'apertura di un canale logico, un terminale invia un messaggio H.245 openLogicalChannel ed attende come risposta openLogicalChannelAck, il quale contiene il Trasport Address che il primo terminale ha assegnato a quel canale logico.
Continuando il grafico proposto nella figura 7.1, una possibile rappresentazione della fase di scambio di capacità e successiva apertura dei canali logici tra due endpoints potrebbe essere la seguente:

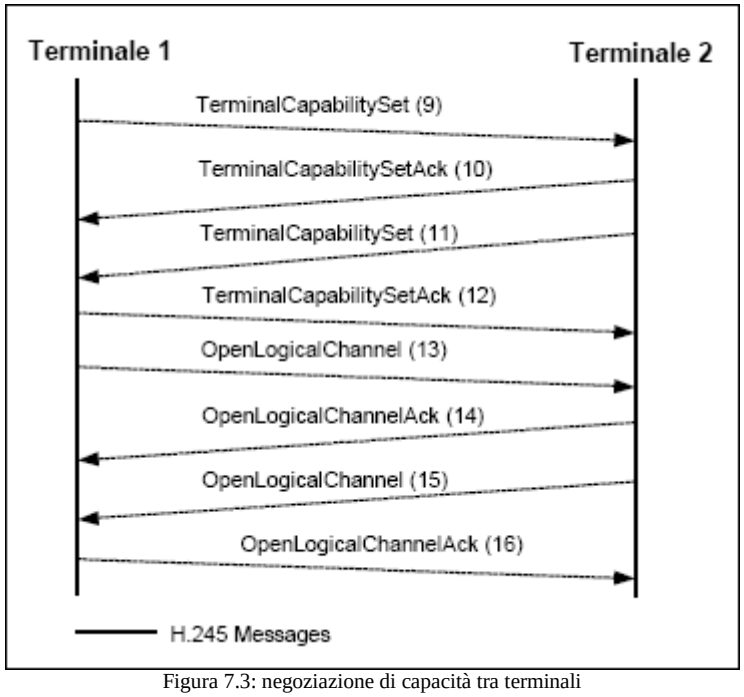

Figura 7.3: negoziazione di capacità tra terminali

I terminali si scambiano rispettivamente il messaggio di TerminalCapabilitySet (9 e 11) per la negoziazione delle capacità trasmissive dei terminali. La controparte di questo messaggio è TerminalCapabilityAck (10 e 12) inviato sul canale di controllo per il riscontro.

A questo punto vengono aperti i canali su cui si baserà la comunicazione tramite il messaggio OpenLogicalChannel(13 e 15) all'interno del quale è presente l'indirizzo IP e la porta che verranno utilizzati per inoltrare e inviare gli stream ricevuti e le segnalazioni RTCP. Questo messaggio sarà inviato per ogni canale logico unidirezionale che verrà aperto durante la chiamata. Il messaggio di riscontro per OpenLogicaloChannel è OpenLogicalChannelAck (14 e 16). Il processo di apertura dei canali logici è simmetrica.

Tutto questo procedimento avviene nel caso di un collegamento point-to-point. Qualora si passi ad un collegamento multicast la procedura si modifica nel seguente modo:

il terminale che decide di aprire un canale logico di comunicazione con tutti gli altri che partecipano alla videoconferenza, invia la sua richiesta al

Multipoint Controller che la gestisce. A questo punto sarà il MC che, con una procedura simile a quella descritta sopra, provvederà ad attivare il canale logico con tutti gli altri partecipanti della videoconferenza.

7.4 Servizi della chiamata.

Qualora gli endpoint che partecipano alla conferenza fossero registrati con due gatekeeper (o con il medesimo) possono richiedere alcuni servizi specifici, durante la chiamata stessa. Per fare questo occorre chiudere e riaprire il canale logico. Per esempio se un endpoint richiedesse ad un gatekeeper un cambiamento di larghezza di banda si avrebbe la chiusura del vecchio canale con closeLogicalChannel e né verrebbe aperto uno nuovo con le recenti impostazioni.
E' in questa fase inoltre che avvengono gli inviti e le inclusioni a una videoconferenza.

Espansione di una conferenza
Il passaggio da conferenza poin-to-point a conferenza multipoint decentralizzata (in quella centralizzata la procedura viene svolta dall'MCU, e non comporta nessun cambiamento agli endpoint già connessi), può essere fatto solo qualora nella conferenza point-to-point iniziale era contenuto un MC. L' estensione della conferenza a multipoint può avvenire in due modi:
o uno dei due endpoint già in conferenza(E1 e E2) né invitano un terzo (E3) oppure se E3 chiama E1 o E2.

Supponiamo di essere nel primo caso e che il Multipoint Controller attivo sia

contenuto in E1; vediamo una descrizione generale di come avviene l' espansione.
E1 invia un messaggio di Setup ad E3. Se E3 accetta la chiamata, si procederà con la procedura di setup sino a l'invio del messaggio Connect. A questo punto E1 apre un canale di controllo H.245 con E3 con il quale vengono scambiati i messaggi di terminalCapibilitySet tra E3 ed MC.
Il MC invia a tutti i partecipanti le nuove impostazioni delle capacità trasmissive ed assegna il numero di identificazione nella conferenza ad E3 mediante terminalNumberAssign e lo notifica agli altri partecipanti.
Se i canali di comunicazione aperti in precedenza da E1 ed E2 non sono compatibili con i nuovi parametri della comunicazione vengono chiusi ed immediatamente riaperti con le impostazioni corrette.

7.5 Disconnessione della chiamata.

L' ultima fase è quella della terminazione di chiamata, in cui qualsiasi end-point possono decidere la terminazione della chiamata. Sono possibili le seguenti procedure:

1) interrompere una trasmissione video chiudendo il canale logico per il video.
2) interrompere la trasmissione dei dati chiudendo il canale logico per i dati.
3) interrompere la trasmissione audio chiudendo il canale logico per l'audio.
4) trasmettere il messaggio H.245 endSessionCommand tramite il canale di controllo H.245.
5) ricevere un messaggio H.245 endSessionCommand da un altro end-point e chiudere l' H.245 Control Channel.
6) se il Call Signaling Channel è aperto, spedire il messaggio H.225 Release Complete e quindi chiudere il canale.

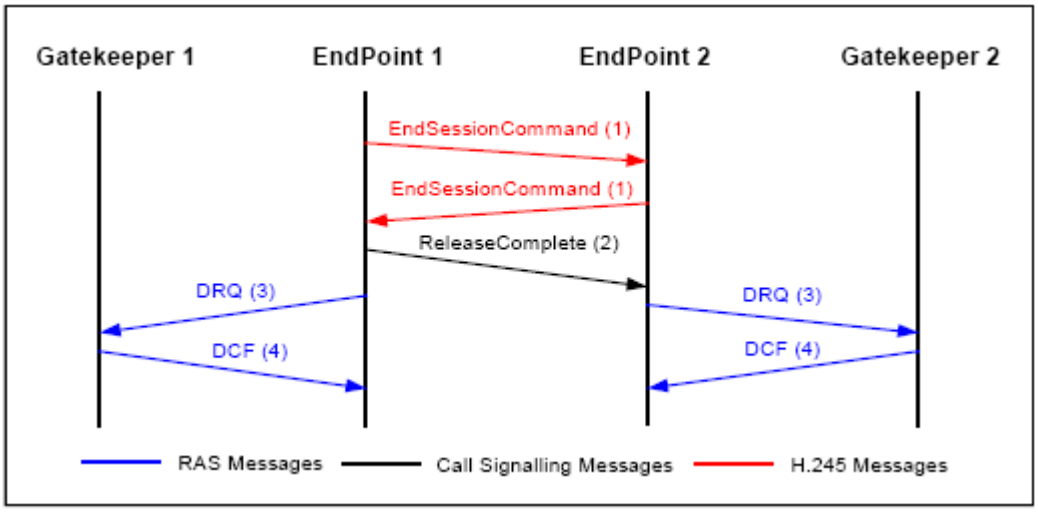

Figura 7.4: handshake per il rilascio di chiamata

Un endpoint che riceve un messaggio di endSessionCommand senza averne inviato uno esegue tutte le procedure eccetto la 5°, poiché non deve aspettare endSessionCommand dal terminale che richiede di essere scollegato dalla chiamata.

Ovviamente la disconnessone di un terminale appartenente ad una conferenza, non comporta l'annullamento di questa finché due o più terminali rimangono attivi.

Se l'EndPoint era registrato ad un Gatekeeper deve comunicargli il rilascio della banda che era a sua disposizione.

Un gatekeeper può inoltre forzare un endpoint a cancellare una conferenza, attraverso il messaggio di DRQ(Disengage request).

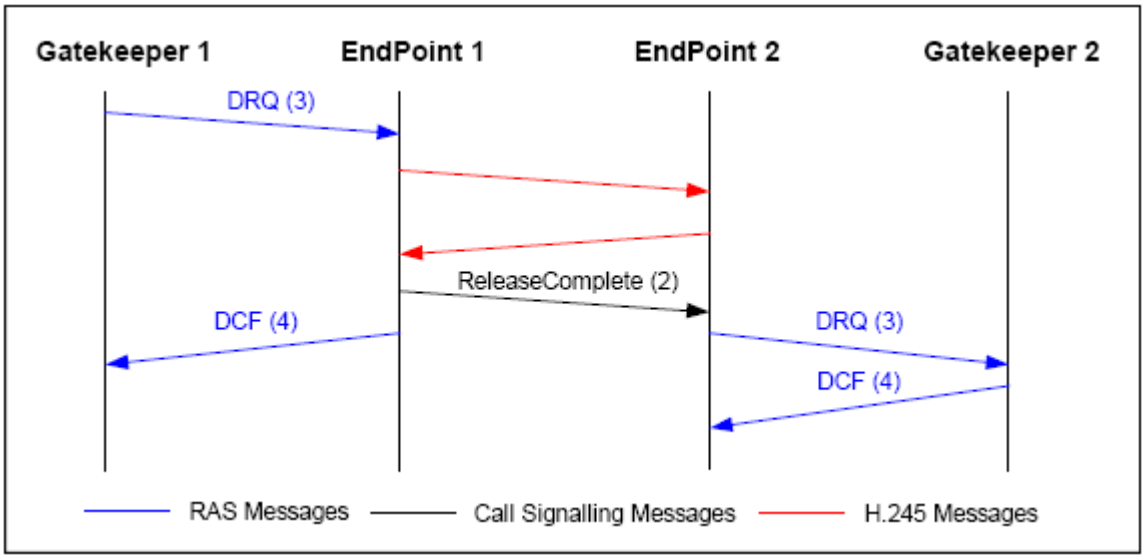

Figura 7.5: handshake per il rilascio di chiamata forzato da un GK

8 Streaming tramite H.323

Come anticipato nel capitolo precedente, H.323 utilizza due tipi di connessioni, una affidabile (TCP) per le segnalazioni di controllo mentre un non affidabile (UDP) per la consegna di dati multimediali. Questo perché in una videoconferenza alcuni dati sono inviati con la necessità di un riscontro che essi siano effettivamente giunti a destinazione, come normalmente avviene per quasi tutti i pacchetti, ma in quanto la videoconferenza è un applicazione interattiva tra due o più utenti, emerge il problema del real-time per i flussi multimediali. Proprio per risolvere tale problema, l'ITU ha inserito il protocollo RTP nella suite H.323 di standard di comunicazione.

RTP (descritto nel documento RFC numero 1889 del IETF) provvede ad una distribuzione di servizi punto-punto per dati che necessitano di trasferimento in tempo reale come l'interattività audio e video. Questi servizi includono l'identificazione del tipo di payload, il sequence numbering, timestamping, e il monitoring. Le applicazioni tipicamente pongono l'RTP in un'insolita posizione. La sua posizione all'interno dello stack è nello spazio utente, sopra a UDP, anche se può essere usato con altri protocolli di rete e trasporto sottostanti. Quest'ultimo lo contiene e trasporta attraverso una rete IP.

UDP è uno standard di comunicazione "connectionless", dove perciò i singoli pacchetti sono del tutto indipendenti tra loro e non seguono sempre la stessa strada.

Va ricordato inoltre che RTP non fa parte dello stack TCP/IP come UDP: in ogni suo datagramma le applicazioni che lo gestiscono aggiungono alla partenza dei dati (e riconoscono all'arrivo) un ulteriore header di 12 byte, contenente:

- il tipo di carico o payload, che descrive quali dati sono trasportati e la loro codifica;
- il numero del pacchetto RTP, grazie a cui si riassembla il messaggio originale e si scoprono pacchetti persi, duplicati o danneggiati;

- il time stamp, con il quale si ricostruisce la sincronizzazione del flusso e si determinano, tra l'altro, eventuali variazioni del tempo di trasferimento, conosciute come jitter;
- l'identificatore della sorgente, che aiuta il destinatario a distinguere tra più flussi contemporanei.

Un pacchetto UDP quindi oltre che a contenere il suo header(8 byte), e quello del protocollo IP(20 byte), conterrà un ulteriore header, quello RTP(12 byte), con il suo relativo payload.

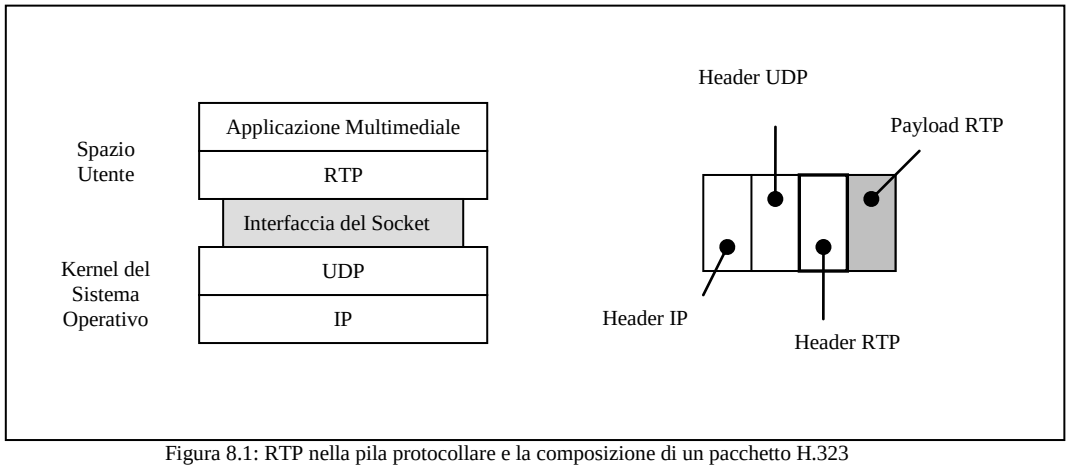

Figura 8.1: RTP nella pila protocollare e la composizione di un pacchetto H.323

Abbiamo già detto che il protocollo RTP non esegue nessuna routine di controllo per accertarsi dell'effettiva consegna dei dati. In mancanza di un pacchetto la miglior azione che il destinatario può compiere è approssimare il valore mancante per interpolazione(questo dovuto al fatto che anche un leggere ritardo potrebbe far perdere valore al pacchetto, in quanto si parla di un contesto real time). Il compito di controllare che i dati inviati giungano a destinazione dovrebbe essere fatto dal sottostante stato di trasporto e come accennato il compito di controllo non rientra tra le responsabilità dell'UDP.

Inoltre RTP non si interessa del mancato ordine dei pacchetti né nei criteri di ricostruzione degli streams. Ed è per questi motivi che la IETF ha affiancato a RTP il protocollo **RTCP**, il quale provvederà alle lacune introdotte da RTP.

Mentre il Real-Time transport Protocol è adibito al multiplexing dei flussi di dati in un unico flusso UDP e il successivo trasporto con proprietà di tempo reale, la sua contro parte Real-Time transport Control Protocol, è in grado di fornire una sorta di riscontro bidirezionale tra i due computer in comunicazione. La sua principale funzione è quella di monitorare la qualità del servizio e fornire informazioni sui partecipanti di una sessione in atto. Quest'ultimo aspetto dell'RTCP può essere sufficiente per sessioni dove non c'è un esplicito controllo dei "dialoganti", ma non necessariamente inteso a sostenere tutti i controlli sulle richieste di comunicazione di un'applicazione.
Per esempio, se vi sono problemi, l'applicazione sul computer ricevente può richiedere di diminuire la velocità di trasmissione del flusso video. In tal modo la qualità delle immagini peggiorerà, diminuirà la risoluzione o vi saranno meno frame al secondo, con un risultato più "sobbalzante" e discontinuo, ma il flusso video non sarà interrotto. Contrariamente, se la rete funziona bene, si potrebbe chiedere l'aumento della velocità dei dati che comporterebbe una più alta qualità.
Inoltre le specifiche dello standard RTCP contengono molte indicazioni per aiutare i programmatori a non consumare troppa banda con i messaggi di controllo.

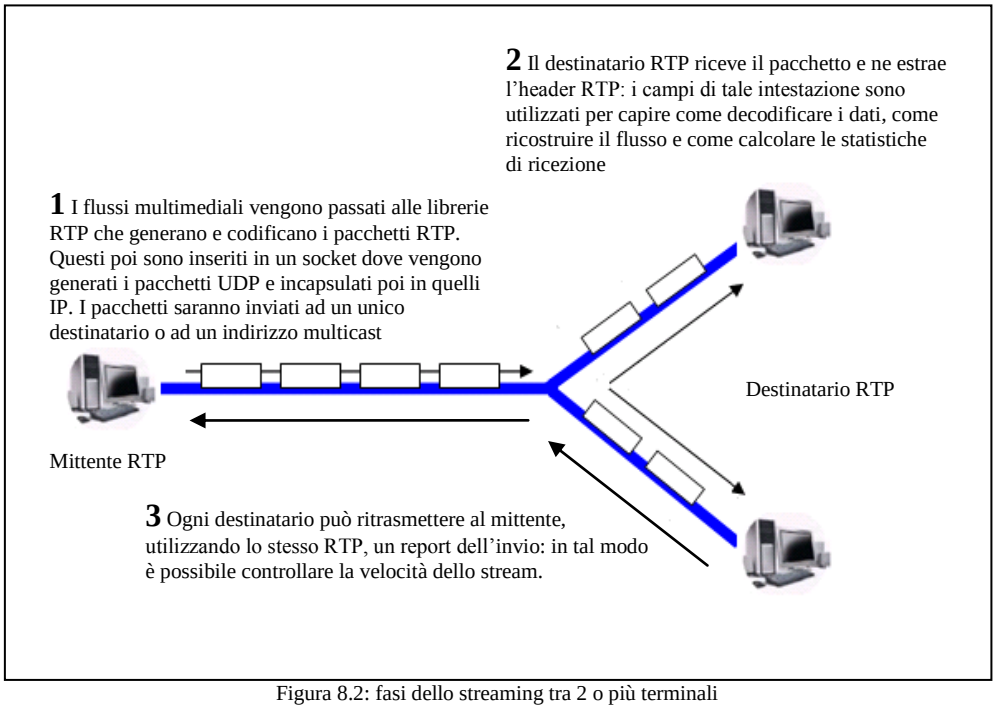

Figura 8.2: fasi dello streaming tra 2 o più terminali

8.1 Sessione RTP

In una conferenza audio-video entrambi i media sono trasmessi come sessioni RTP separate e i relativi pacchetti RTCP usano due differenti coppie di porte UDP e/o indirizzi multicast. Per ciascun flusso dal mittente ai destinatari viene utilizzata una diversa sessione RTP per trasportare i dati. Per quanto detto quindi, se chi invia trasmette più tipi di dati dovrà avere più sessioni RTP, per trattare ciascun flusso separatamente. Non esiste un diretto accoppiamento a livello RTP fra sessioni audio e video, a meno che l'utente che partecipa ad entrambe le sessioni non usi lo stesso nome canonico nel pacchetto RTCP per l'audio e il video, cosicché sia possibile associare le sessioni. Un motivo di questa separazione sta nel fatto di permettere ai partecipanti della conferenza di avere la possibilità di ricevere solo un determinato flusso, in base alle loro esigenze. Nonostante la separazione comunque possibile sincronizzare il

playback della sorgente audio-video usando informazioni di temporizzazione, trasportate nei pacchetti RTCP, per entrambe le sessioni.

8.2 Il protocollo UDP

UDP (acronimo per User Datagram Protocol) descritto in RFC 768 è un modo per inviare datagrammi IP incapsulati senza dover stabilire una connessione. Equivale fondamentalmente al pacchetto IP con l'aggiunta di un header di otto byte contenente quattro campi.

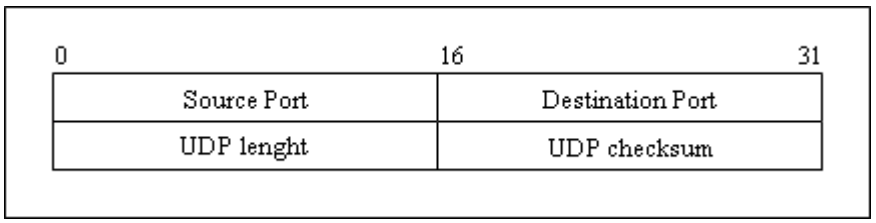

Figura 8.3: header UDP

I quattro campi sono di due byte ciascuno, e contengono quanto segue:

- Source Port: contiene la porta sorgente per quel determinato pacchetto;
- Destination Port: contiene la porta al quale il pacchetto dovrà essere consegnato;
- UDP length: la lunghezza del pacchetto, ed include l'intestazione e i dati trasportati;
- UDP checksum: campo opzionale.

Quando arriva un pacchetto UDP, il suo carico utile viene direttamente consegnato al processo associato alla porta di destinazione, è il protocollo non si occupa del controllo di flusso, del controllo degli errori o della ritrasmissione dopo la trasmissione errata di un frame.

UDP può essere considerato come niente più che un interfaccia del protocollo IP, con la caratteristica aggiunta del demultiplexing di più processi utilizzando le porte.
Per questa sua caratteristica, risulta dunque essere un'ottima tecnologia per le applicazioni che richiedono lo scambio di brevi e veloci messaggi. Un esempio pratico di un'applicazione che utilizza UDP è DNS(Domain Name System), nella quale un messaggio contenente il nome di un host è inviato ad un server DNS. Tale server risponderà con un altro messaggio UDP contenente l'indirizzo IP relativo all'host name inviato. Qualora la richiesta o la risposta sia persa, il client va in timeout e riprova. In questo modo non è necessario stabilire una connessione per poi rilasciarla in seguito, ma è sufficiente lo scambio di due messaggi.

8.3 Struttura di un pacchetto RTP

Andiamo ora a studiare la struttura di un pacchetto RTP. Qui di seguito troviamo la rappresentazione grafica della sua intestazione:

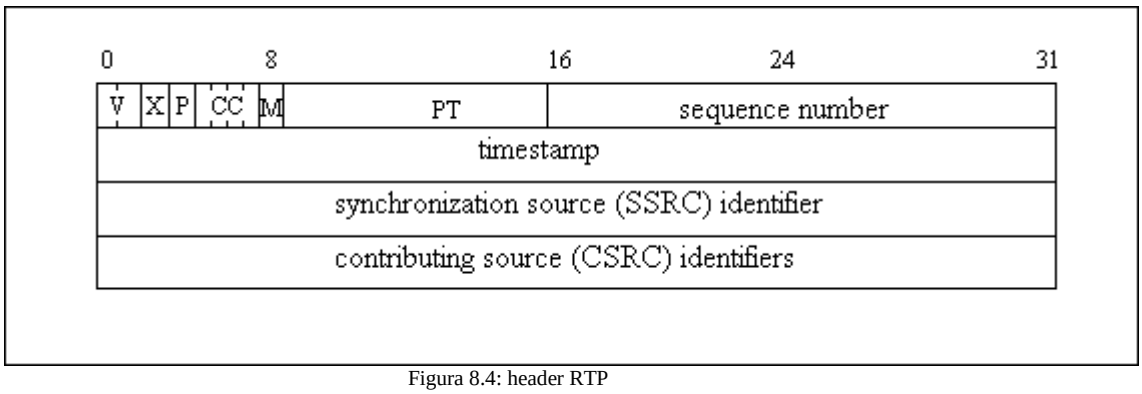

Figura 8.4: header RTP

E' da notare che precedentemente è stato detto che ogni header RTP era formato da solamente dodici byte, mentre in figura è stato riportato un header con quattro byte in eccesso. Questo perché il campo contributing source

identifiers, come si vedrà, è un campo opzionale, ed è presente solo se inserito da un mixer, diversamente dai primi dodici bytes che sono presenti in ogni pacchetto RTP.
Andiamo ora a descrivere ciascuno dei campi citati:

- **V** (version): 2 bits , questo campo identifica la versione di RTP. Il valore 1 è usato nella prima versione RTP e il valore 0 è usato dal protocollo inizialmente implementato nel VAT audio tool, mentre attualmente la versione corrisponde a 2.

- **X** (extension): 1 bit, se il bit di estensione è settato, la parte fissa della testata è seguita esattamente da un'estensione header, con un formato da definire; si tratta di una via di fuga da qualsiasi imprevisto.

- **P** (padding): 1 bit , se il bit padding è attivo, il pacchetto contiene alla fine uno o più padding bytes addizionali non facenti parte del payload. L'ultimo byte padding contiene un valore di quanti bytes padding sono stati aggiunti. Questi particolari bytes possono essere necessari per un'eventuale criptaggio con blocchi di misura fissa o per trasportare pacchetti RTP in un'unità dati per protocolli di strati inferiori.

- **CC** (CSRC Count): 4 bits , contiene il numero degli identificatori contributing source, da 0 a 15, che seguono la testata fissa (vedi sotto).

- **M** (marker): 1 bit , l'interpretazione del marker è definita da un documento (profile). Permette eventi significativi come il delimitare "i confini" di un frame in uno stream di pacchetto. Un profile può definire marker bits aggiuntivi. In pratica il marker definisce l’inizio di un inquadratura in un video, l’inizio di una parola in un canale audio o qualcos’altro che l’applicazione è in grado di comprendere.

- **PT** (Payload type): 7 bits, questo campo identifica il formato del carico di RTP e determina la sua interpretazione dall'applicazione. Una relativa specifica contiene una mappa di codici payload per l'interpretazione del formato(per esempio audio a otto bit non compresso, mp3, ecc.).

- **sequence number**: 16 bits , il sequence number è un contatore incrementato di uno per ogni pacchetto RTP spedito, cosicché il ricevitore può risalire ad un'eventuale perdita di pacchetti e ristabilirne la sequenza. Il valore iniziale del sequence number è casuale (impredicibile) per rendere più difficili eventuali attacchi sul criptaggio, anche se in realtà la sorgente stessa non cripta. Ciò può essere effettuato dal translator. Le tecniche per la scelta dei numeri impredicibili è dettagliata nella versione integrale delle specifiche RTP.

- **timestamp**: 32 bits , il timestamp riflette l'istante di campionamento del primo byte del pacchetto RTP, ed è prodotto da chi genera il flusso. L'istante di campionamento deve essere derivato da un clock che incrementa monotonamente e linearmente nel tempo per permettere i controlli di sincronizzazione e le misure dell'incertezza sugli arrivi dei pacchetti (arrival jitter). Questo valore può aiutare a ridurre il jitter nel ricevitore, disaccoppiando l'instante di riproduzione da quello di arrivo.
Se invece i pacchetti RTP sono generati periodicamente, bisognerà considerare l'istante di campionamento nominale come determinato dal clock di campionamento e non come lettura del clock di sistema. Il valore iniziale del timestamp è casuale, come per il sequence number. Molti pacchetti RTP consecutivi possono avere gli stessi timestamp se essi sono generati nello stesso istante. Pacchetti consecutivi possono contenere timestamp non monotoni se il dato non è trasmesso nell'ordine di campionamento, come nel caso di frame video MPEG interpolati(i sequence numbers dei pacchetti come trasmessi saranno monotoni).

- **SSRC** (synchronization source): 32 bits, specifica a quale flusso appartiene il pacchetto, e supporta il metodo utilizzato per il multiplexing e

demultiplexing di più flussi dati in un singolo pacchetto. Esempi di synchronization source possono essere riferiti a sorgenti di segnali come un microfono, una videocamera, o un mixer RTP. Una synchronization source può cambiare il suo data format, per esempio nell'audio encoding. L'identificatore SSRC è scelto in maniera casuale per essere globalmente unico per una sessione RTP; l'intento è quello di non avere due synchronization source nella stessa sessione RTP con un uguale SSRC. Se un partecipante genera stream multipli in una sessione RTP, per esempio da videocamere separate, ogni sorgente deve essere identificata con un SSRC differente. Non è richiesto infatti che un partecipante abbia lo stesso SSRC identifier per tutte le sessioni RTP in un'unica sessione multimediale. Sebbene la probabilità che sorgenti multiple siano associate allo stesso identifier è bassa, tutte le implementazioni RTP devono essere preparate a evitare e risolvere le collisioni. Il legame fra gli SSRC identifier è regolato attraverso l'RTCP.

contributing source identifier : da 0 a 15 campi, 32 bits ciascuno e sono utilizzati in presenza di un mixer. Il mixer inserisce nell'RTP header di un particolare pacchetto una lista di SSRC identifiers relativi a quelle sorgenti che hanno contribuito con il loro mixaggio alla generazione di quel pacchetto. Questa lista è chiamata CSRC list. Un esempio applicativo è la multi conferenza audio dove un mixer indica tutti i partecipanti del colloquio nel pacchetto, permettendo al ricevente di risalire ai partecipanti in corso, anche se tutti i pacchetti audio contengono lo stesso SSRC identifier (quello del mixer). Il numero di identifiers è dato dal CC field. Se ci sono più di quindici contributing sources, soltanto quindici ne possono essere identificate.

8.4 Struttura pacchetto RTCP

Come già detto al fine di comunicare proprietà di rete e di sessione, durante una videoconferenza vengono scambiati dei pacchetti RTCP, che possono appartenere ad una delle seguenti 5 tipologie:

SR(sender report);
RR(receiver report);
SDES(source description);
BYE e APP.

Ogni pacchetto è composto da una parte fissa e da una variabile. Più pacchetti insieme potrebbero aggregarsi in un unico pacchetto composto inviato da un solo datgramma UDP. Tale pacchetto risulterà essere composto come segue:

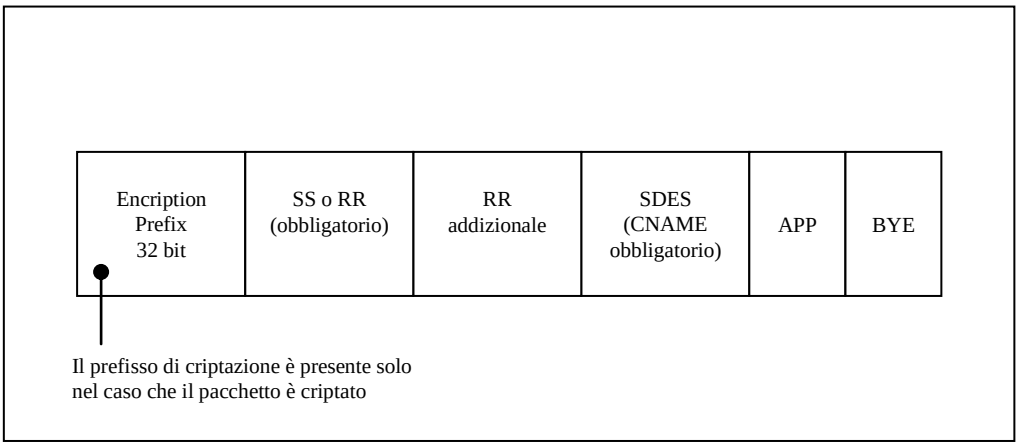

Figura 8.5: un pacchetto RTCP composto

8.4.1 RTCP SR

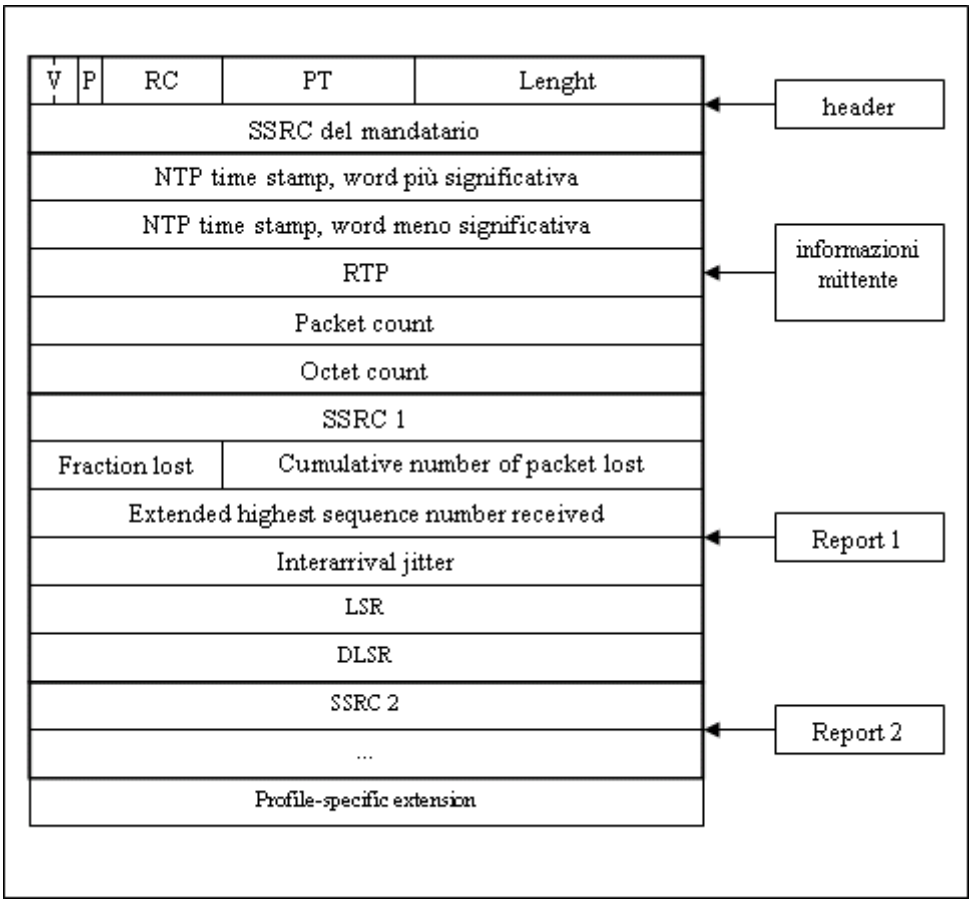

Figura 8.6: struttura di un pacchetto SR

Fondamentalmente il pacchetto SR consiste in 3 sezioni, con la possibilità di una quarta, chiamata profile-specific extension.
La prima sezione è l'header del pacchetto, di otto bytes ed i campi in essa contenuti sono:

RC (reception report count): 5 bits, il numero delle sezioni di report contenute nel pacchetto. Può anche essere 0.

PT (packet type): 8 bits, contiene la costante 200, che lo identifica come pacchetto RTCP tipo SR.

length: 16 bits, la lunghezza del pacchetto in parole da trentadue bit, includendo lo stesso header e tutte le parole di padding.

SSRC (synchronization source): 32 bits, l'identificatore della risorsa che ha generato il pacchetto.

La seconda sezione contiene le informazioni relative a chi invia, ed è composta da 20 bytes presenti in ogni pacchetto SR(sezione obbligatoria). Riassume i dati inviati da questo terminale, specificando i seguenti campi:

NTP timestamp: 64 bits, indica l'istante in cui il pacchetto di report è stato inviato, e potrebbe essere utilizzato in coppia con il timestamp dei report di ritorno ricevuti cosicché il terminale possa misurare il tempo di round-trip per i messaggi a quel determinato destinatario.

RTP timestamp: 32 bits, indica lo stesso istante di NTP timestamp, ma specificato come nei pacchetti RTP(stesse unità e stesso offset casuale dei pacchetti di data).

sender's packet count: 32 bits, il numero totale di pacchetti dati RTP trasmessi dal mandatario del pacchetto RTCP, dal momento in cui la trasmissione è cominciata sino alla creazione del pacchetto SR. Questo contatore viene azzerato se il terminale cambia uno dei suoi identificatori SSRC.

sender's octet count: 32 bits, come sopra ma il numero di ottetti relativi al payload inviati. Questo campo potrebbe essere utilizzato per stimare la tasso medio di payload inviato.

La terza sessione contiene 0 o più blocchi di report di ricezione, in base al numero di risorse "ascoltate" da questo terminale sino all'ultimo report. Ciascun blocco conterrà le statistiche sulla ricezione di pacchetti RTP di una singola SSRC. Tali statistiche saranno le seguenti:

SSRC_n (source identifier): 32 bits, l'identificatore SSRC della risorsa a cui le informazioni del blocco fanno riferimento.

fraction lost: 8 bits, il numero di pacchetti persi con risorsa SSRC_n, dall'invio del precedente messaggio SR o RR

cumulative number of packets lost: 24 bits, il numero totale di pacchetti persi durante la ricezione con SSRC_n, per tutta la sua durata. Il numero di pacchetti persi(sia fraction che cumulative) è dato dal numero di pacchetti aspettato meno il numero di pacchetti arrivati, il che comprende anche pacchetti in ritardo e pacchetti duplicati. Questo significa che tali valori potrebbero anche essere negativi.

extended highest sequence number received: 32 bits, numero di sequenza dell'ultimo pacchetto RTP ricevuto dalla risorsa SSRC_n.

interarrival jitter: 32 bits, una stima della variazione statistica del tempo di arrivo dei pacchetti RTP.

LSR (last SR timestamp): 32 bits, questo campo conterrà 0 se il terminale non ha ancora ricevuto nessun pacchetto SR da SSRC_n, altrimenti conterrà il timestamp del pacchetto RTCP sender report più recentemente ricevuto da SSRC_n. In quanto NTP timestamp è composto da otto ottetti, né saranno presi solamente i 32bit centrali.

DLSR (delay since last SR): 32 bits, rappresenta il ritardo espresso in 1/65536 di secondo, fra l'ultimo pacchetto SR ricevuto da SSRC_n e la spedizione di

questo blocco di report. Questo campo è zero se nessun pacchetto è ancora stato ricevuto da SSRC_n.

Una sorgente SSRC_n potrebbe calcolare il tempo totale di propagazione per l'invio a SSRC_r salvando il tempo A relativo all'arrivo di un suo blocco di report. Potrebbe calcolare il tempo che un pacchetto impiega per giungere la destinazione e il tempo che la risposta impiega a tornare all'originario mandante tramite la formula A-LSR, utilizzando il campo LSR del report appena arrivato, per poi sottrarne il ritardo provocato dall'invio della risposta, avendo così A-LSR-DLSR.

8.4.2 RTCP RR

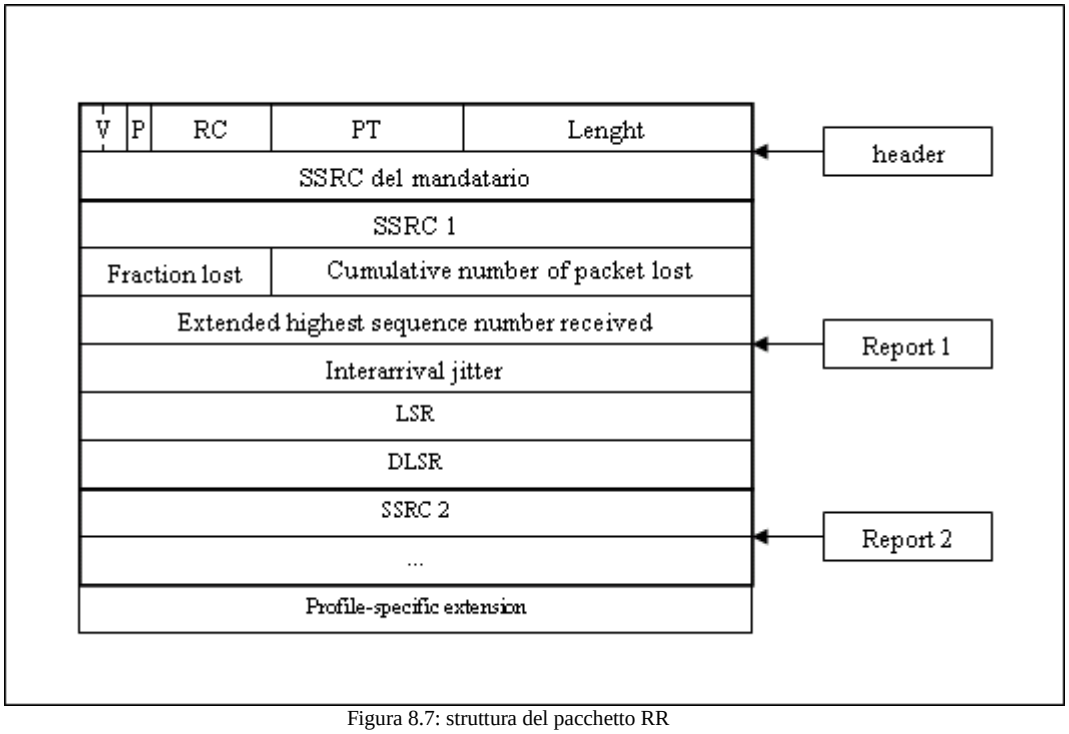

Figura 8.7: struttura del pacchetto RR

Il formato del pacchetto receiver report è sostanzialmente lo stesso di sender report, eccetto per il contenuto del campo packet type (PT) il quale conterrà la

costante 201. Vengono inoltre omesse le cinque parole relative alle informazioni del mittente. I rimanenti pacchetti sono identici al pacchetto SR.

8.4.3 RTCP SDES

Il pacchetto SDES è un pacchetto composto da due livelli, un'intestazione e nessun o più "pezzi" (chunks), ognuno dei quali contiene la descrizione di un item, ovvero una sorgente.

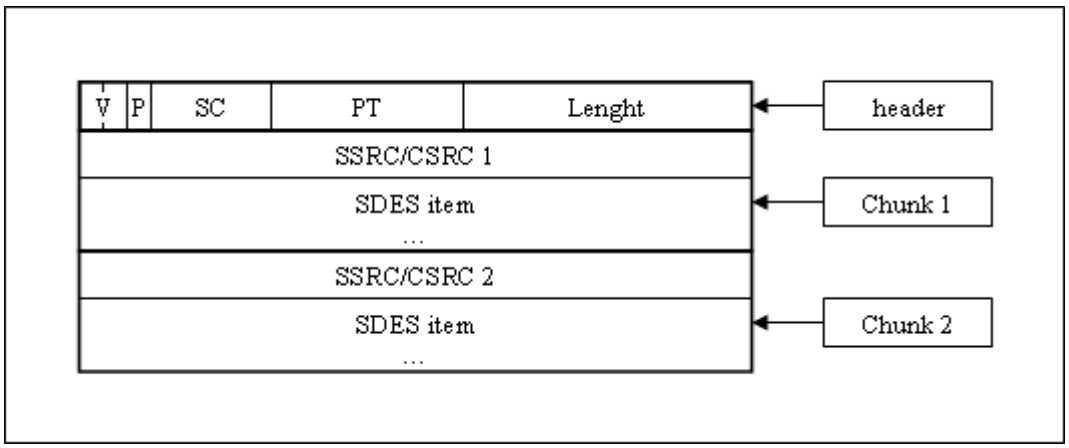

Figura 8.8: struttura di un pacchetto SDES

Per quanto riguarda l'intestazione, le uniche variazioni, rispetto ai precedenti protocolli, si riscontrano nel campo PT, il quale contenuto sarà 202(costante assegnata al pacchetto SDES) ed il campo RC, sostituito con il campo CC.

PT (packet type): 8 bits, contiene la costante 202 per identificare che il pacchetto è di tipo SDES.

SC (source count): 5 bits, il numero di SSRC/CSCR chunks contenuti nel pacchetto. Zero è valido ma rende il pacchetto inutile.

Ciascun chunk consiste in un identificatore SSRC/CSRC seguito da uno o più item, i quali portano informazioni relative la sorgente. Ogni chunk contiene:

SSRC/CSRC: 32 bit, indica la sorgente a cui si riferisce il chunk.

SDES item: è una parola costituita a sua volta da 3 campi contenenti un identificatore di 8 bit per il tipo di item(NAME, CNAME, EMAIL, ecc.), un campo Lenght, anch'esso di 8 bit, per la lunghezza del testo dell'item, ed il campo per il testo stesso, che non può essere più lungo di 255 bytes.

In pratica possiamo dire che un endpoint invia un pacchetto SDES con il suo proprio identificatore(lo stesso SSRC contenuto dal pacchetto RTP). Un mixer invia un pacchetto SDES contenente un chunk per ogni risorsa che contribuisce allo stream composto e dalla quale riceve informazioni SDES.

8.4.4 RTCP APP

The APP packet is intended for experimental use as new applications and new features are developed, without requiring packet type value registration. APP packets with unrecognized names should be ignored. After testing and if wider use is justified, it is recommended that each APP packet be redefined without the subtype and name fields and registered with the Internet Assigned Numbers Authority using an RTCP packet type.

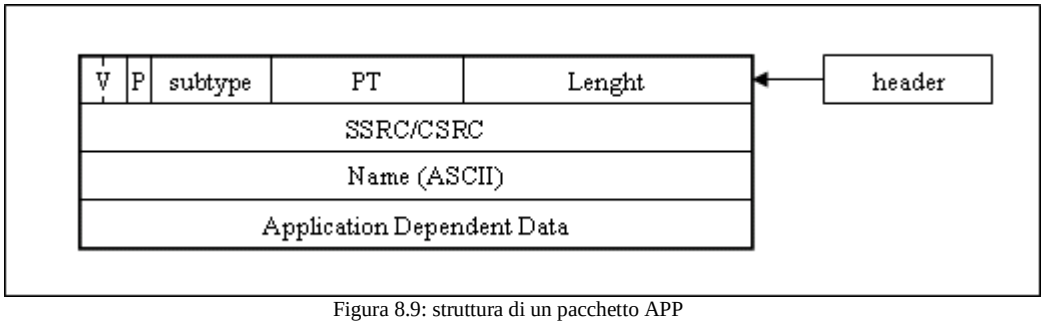

Figura 8.9: struttura di un pacchetto APP

I campi version, padding e length descrivono le stesse funzionalità del pacchetto SR. Per quanto riguarda il campo PT, la costante che identifica un pacchetto APP è 204, mentre il campo subtype descrive quante segue:

subtype: 5 bits, potrebbe essere usato come sotto tipo per permettere ad un insieme di pacchetti APP di essere definiti sotto un unico nome, o per qualsiasi applicazione dipendente.

name: 32 bits, nome associato all'utente

application-dependent data: lunghezza variabile, tale campo non è obbligatorio, potrebbe quindi non apparire. E' un campo particolare anche per il motivo che non è interpretato da RTP, ma la sua interpretazione è lasciata all'applicazione. Deve essere un multiplo.

8.4.5 RTCP BYE

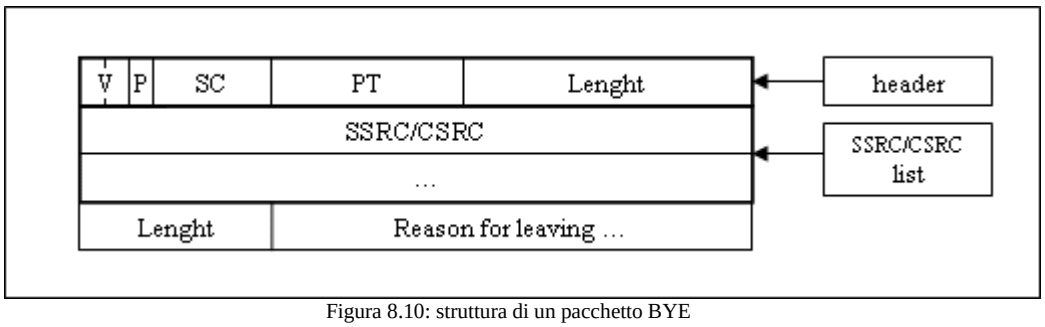

Figura 8.10: struttura di un pacchetto BYE

Il pacchetto BYE indica che uno o più sorgenti stanno uscendo e non sono più attivi.

La sua intestazione è paragonabile a quella del pacchetto SDES, con l'unica differenza che il contenuto del pacchetto PT corrisponde a 203, il quale identifica un pacchetto come pacchetto BYE, ed SC indica quanti SSRC sono listati e non più quanti chunk.

9 Sessione Pratica di Laboratorio

L'obiettivo della sessione di laboratorio è stata quello(come anticipato nell'introduzione) di verificare l'effettiva possibilità di cooperazione tra diversi applicativi H.323.
Le verifiche di videoconferenza sono state divise in diverse sezioni, in modo da evidenziare la cooperazione dei diversi client H.323 su di una stessa piattaforma, verificare l'interoperabilità su multipiattaforma, e una volta verificato il corretto funzionamento del point-to-point, si è passati alla multiconferenza. Per quanto riguarda i componenti utilizzati per stabilire le conferenza, sono stati utilizzati vari applicativi tra cui alcuni rilasciati dal progetto open source chiamato openH323 e due tra i più popolari applicativi per la videoconferenza basati su H.323, NetMeeting e GnomeMeeting. Nel prossimo capitolo tali componenti verranno descritti più dettagliatamente elencando inoltre le caratteristiche trasmissive.

9.1 Descrizione dei Componenti Utilizzati

Durante la fase di laboratorio abbiamo utilizzato diversi componenti per coprire le funzionalità richieste dalla videoconferenza. Alcuni di questi sono stati messi a disposizione dall'Università di Camerino, ma la maggior parte è disponibile gratuitamente su internet o distribuita con i sistemi operativi.

9.1.1 Il progetto OpenH323

Il progetto OpenH323 ha come obiettivo la creazione e sviluppo di un a serie di applicativi tra loro interoperabili basati sul protocollo per videoconferenza H.323, comprensivi di tutte le caratteristiche dei client professionali, utilizzabili sia da sviluppatori per scopi personali che dagli utenti commerciali. Tale progetto, disponibile alla pagina Web http://www.openh323.org/ è coordinato da Quicknet Technologies Inc. e l'insieme degli applicativi rilasciati possono essere ricondotti a tutti i componenti definiti dalla pubblicazione dell'H.323 (Terminali, Gateway, Gatekeeper, MCU) e sono rilasciati sia per piattaforma Linux che per piattaforma Windows, sia codice binario che sorgente(richiede la compilazione del codice e delle librerie).
Di seguito elenchiamo tutti gli applicativi rilasciati dal progetto e disponibili nel sito:

Programma	**Descrizione**	**Eseguibile**
OhPhone	Un client H.323 su linea di comando	Win/Linux
OpenPhone	Un client H.323 funzionalment eequivalente all'OhPhone ma basato su GUI	Win
OpenMcu	Un server per conferenze H.323	Win/Linux
OpenAM	Semplice programma che risponde alle chiamate, senza richiedere alcun tipo di codec	Win/Linux
OpenGK	Un gatekeeper H.323	Win/Linux
OpenIVR	Interactive Voice Responce, un altro programma di risposta	Win/Linux
PSTNGw	Un gateway per instradare chiamate su PSTN	Win/Linux
CallGen323	Un generatore di chiamate H.323	Win/Linux
T.38 Modem	Trasmette da Modem a gateway T.38	Linux

Nel caso che gli esecutivi vengano utilizzati sotto Windows, c'è la necessità di scaricare due librerie dll, a cui tutti i file sono dipendenti, e sono PwLib(Portable Windows Libraries), una classe di librerie originate le la produzioni di applicazioni eseguibili sia sotto i sistemi Microsoft Windows che X-Windows e OpenH323, che ha lo scopo di realizzare l'implementazione open source delle specifiche dei protocolli per la videoconferenza dell'H.323, favorendo l'interoperabilità . Sarà sufficiente salvarle nella stessa folder nella quale sono stati salvati i client H.323.

9.1.2 OhPhone

OhPhone è un'applicazione su linea di comando basato sulle librerie PWlib e openH323, che può essere utilizzata sia per ricevere chiamate H.323 mettendosi in modalità di ascolto, che per effettuarne una a un host remoto. Inizialmente ideato come un software per test, è stato sviluppato nel tempo con l'aggiunta di nuove caratteristiche, tanto che ora un funzionante applicativo h.323. Ed è proprio per questa sua funzione di tester che ohphone ci verrà molto utile a l'analisi di una chiamata. La prima versione rilasciata del software risale al 1998, e le ultime modifiche apportate sono datata ottobre 2002. Andiamo ora a vedere quali sono i comandi che il software mette a disposizione

Opzioni

Tutte le opzioni da linea di comando di ohphone possono essere espresse in formato lungo più pesante ma mnemonicamente più facile da ricordare. Alcuni invece oltre che con forma estesa, potrebbero essere espressi in forma breve, che consente un'esecuzione del software più rapida.

Opzioni generiche

Comando	Descrizione
-a, --auto-answer	Risponde automaticamente alle chiamate ricevute
-d, --autodial host	Composizione automatica del numero se la chiamata cade
-h, --help	Visualizza l'help in linea
-l, --listen	Viene avviato il programma in modalità di attesa di chiamate entranti
-v, --verbose	Vengono visualizzate in formazioni più dettagliate sui processi in corso
--disable-menu	Disabilita il menu interno
--ringfile filename	Imposta il suono del telefono
--save	Salva la configurazione corrente

Opzioni riguardo al gatekeeper

Comando	Descrizione
-g, --gatekeeper host	Imposta il nome del gatekeeper al quale registrarsi
-G, --gatekeeper-id name	Specifica il gatekeeper tramite ID
-n, --no-gatekeeper	Lancia ohphone senza registrarsi presso un gatekeeper
-r, --require-gatekeeper	Esce se no trova un gatekeeper a cui registrarsi

Opzioni riguardo il protocollo

Comando	Descrizione
-i, --interface ipaddr	Seleziona l'interfaccia per le connessioni entranti. Di default tutte le interfacce TCP/IP
--listenport port	Seleziona la porta d'ascolto per le connessioni entranti(default 1720)
--connection port	Seleziona la porta per le connessioni uscenti(default 1720)
-b, --bandwidth bps	Banda limite da usare in bit/s
-f, --fast-disabled	Disabilita il fast start
-T, --h245tunneldisable	Disabilita il tunneling H.245
-u, --user name	Setta un nome per l'utente

--tos n	Specifica un campo TOS per i pacchetti uscenti

Opzioni audio

Comando	**Descrizione**
-e, --silence	Disabilita la modalità “silente deteciton”
-j, --jitter[min-]max	Imposta il minimo(opzionale) e massimo buffer di jitter(ms)
--recvol n	Regola il volume di registrazione
--playvol n	Regola il volume d’ascolto

Opzioni per la trasmissione vide

Comando	**Descrizione**
--videodevice dev	Seleziona il dispositivo video da utilizzare in trasmissione
--videotrasmit	Abilita la trasmissione video
--videolocal	Apre una finestra localmente con ciò che stiamo trasmettendo
--videosize	Dimensione della finestra video. Puo essere small(default) o large
--videoformat type	Pal, ntsc, auto (efault)
--videoinput num	Seleziona l’ingresso di cattura video
--videotxquality n	Seleziona la qualità del video trasmesso. 1(good)-31 (9 di default)
--videobitrate n	Abilita un costante bitrate. 16-2048 kbit/s (256 di default)

Opzioni per la ricezione video

Comando	**Descrizione**
--videoquality n	Seleziona la qualità video ricevuta 0-31
--videoreceive viddev	Seleziona il dispositivo dal quale si riceve
--videopip	In un riquadro del video ricevuto compare il video locale
--videotest	Per alcuni secondi si vede il video locale, dopodichè si chiude l’applicazione

Opzioni per la scheda audio

Comando	**Descrizione**
-s, --sound device	Seleziona il dispositivo audio da usare
--sound-in device	Seleziona il dispositivo audio da usare in ingresso
--sound-out device	Seleziona il dispositivo audio da usare in uscita
--sound-mixer device	Seleziona il dispositivo mixer da usare
--sound-rechan device	Seleziona la pista da usare sul mixer
--sound-recvol n	Volume di registrazione per la scheda audio
--sound-playvol n	Volume di ascolto per la scheda audio

Opzioni di codifica audio

Comando	**Descrizione**
--g711frames count, --gsmframes count, --g731, --gsm, --g711-ulaw, --g711-alaw, --g728, --g7321	Seleziona il tipo di codifica preferita

Opzioni di debugging

Comando	Descrizione
-t, --trace	Traccia su video tutti i messaggi scambiati
-o, --output file	Registrai messaggi scambiati sul file specificato

Una volta in esecuzione, ohphone mette a disposizione dell'utente una seria di operazioni per le chiamate in ingresso o quelle in progresso. Ognuna delle operazioni è identificata da un singolo carattere che deve essere invocato dalla shell di ohphone. I comandi disponibili sono

Comando	**Descrizione**
q, x	Chiude l'applicazione
H	"Hang up", termina la chiamata
C address[gateway]	Effettua una chiamata verso l'host specificato.

L	Mostra informazioni sul codice della chiamata
d code address	Crea una connessone con uno specifico codice con un utente specificato
S	Da le statistiche sulle chiamate in corso
{,}	Alza, abbassa il volume di registrazione
[,]	Alza, abbassa il volume d'ascolto
V	Mostra il volume attuale
E	Abilità/disabilità silente detection
I	Mostra le ultime 16 chiamate effettute
i	Mostra le ultime 16 chiamate ricevute

Esempi

ohphone -l, trova un gatekeeper nella rete locale, ad esso si registra, e aspetta in attesa di chiamate;

ophone -ln, si mette in ascolto di chiamate senza cercare un gatekeeper;

ohphone -ln --q0 -callerid, si mette in ascolto senza registrarsi ad un gatekeeper, utilizzando /dev/phone0 (una scheda Quicknet) come dispositivo audio, a abilitando la trasmissione dell'audio dal host specificato alle cuffie;

ohphone -n ipaddress, inizia una chiamata diretta all'host specificato, senza l'uso di un gatekeeper

9.1.3 OpenPhone

Open Phone è un client realizzato per Windows basato sulle librerie OpenH323 e Pwlib. Offre essenzialmente le stesse funzionalità del OhPhone,

con l'unica differenza che essendo basato su GUI, ha un interfaccia grafica, che permette ai suoi utenti un approccio più diretto e intuitivo.

9.1.4 OpenMCU

OpenMCU è un applicativo per il supporto della multiconferenza e realizza di fatto l'elemento di rete Multipoint Control Unit definito dalla raccomandazione H.323. Può essere eseguito sia su Linux che su Windows, ma è supportato da più piattaforme riportate nel sito del progetto.

Ha le seguenti caratteristiche:

- non richiede codecs su hardware per funzionare
- supporta G.711, GSM MS-GSM e LPC-10 come codecs audio
- supporta codifica video H.261
- può accettare numerose connessioni contemporaneamente
- diverse videoconferenze possono avere luogo allo stesso tempo utilizzando la caratteristiche delle 'stanze'.
- mostra le statistiche delle chiamate correnti
- possibilità di effettuare chiamate verso endpoint remoti
- audio loopback mode: restituisce l'audio stream in una specifica stanza

Quando eseguito, l' OpenMCU inizializza un processo di H.323 listener aspettando segnalazioni di chiamata da parte di altri utenti. Qualora una connessione viene stabilita, si determina a quale conferenza l'endpoint ha richiesto di aggregarsi tramite l'opzione della 'stanza', e la aggiungi a tale conferenza. L'MCU può essere chiamato tramite il formato "nome_stanza@nom_server".

Le nuove stanze vengono automaticamente create e nel caso che nessuna stanza venga specificata, l'utente viene aggiunto alla stanza(quindi alla conferenza) di default, chiamata room101.

L'MCU riceve gli stream degli utenti ad esso connessi, sia audio che video, questi vengono mixati e restituisce infine l'output a tutti i partecipanti alla conversazione. Per quanto riguarda l'audio non ci sono restrizioni, ma l'output del video sarà una finestra contenente quattro riquadri, ciascuna delle quali conterrà il video della persona che più recentemente ha preso la parola, come definito dalla modalità di videoconferenza Voice Activated, è per questo buona norma la modalità "Mute" qualora non si parlasse, per evitare ad un continuo switch di immagine.

Essendo un file eseguibile, OpenMCU può essere lanciate tramite doppio click sulla sua icona, ma sconsigliabile in quanto non consente di specificare le opzioni da linea di commando. Qui a seguito è la lista di tutte le opzioni da console:

Comando	**Descrizione**
-u –username str	Imposta il nome dell'endpoint locale come str
-g --gatekeeper host	Specifica un determinato host per il gatekeeper
-n --no-gatekeeper	Disabilita la ricerca di un gatekeeper
--require-gatekeeper	Termina la sessione se non trova un gatekeeper
-i --interface ip	Seleziona un interfaccia IP da usare
--g711frames count	Imposta nelle capacità il numero di frame g.711 (30 di dafault)
--gsmframes count	Imposta nelle capacità il numero di frame GSM (4 di dafault)
-t –trace	Traccia i messaggi scambiati su schermo
-o –output	Determina il file per il tracing, il file di dafault è stderr
--save	Salva la configurazione corrente
-v --video	Abilità la gestione da parte dell'MCU di flussi video(H.261)
--videolarge	Imposta la grandezza del video da normale(176x144) a grande(352x288)
--videotxquality n	Seleziona la qualità del video, da 1(migliore) a 31. (Default 9)

--videofill n	Numeri di blocchi di background aggiornati per frame 1-99 (Default 2)
--videotxfps n	Il numero massimo di frame video trasmessi per secondo 1-30
--defaultroom name	Le connessioni che non specificano una stanza verranno inserita a quella specificata tramite questo commando (room di default room101)
--no-defaultroom	Rifiuta le connessioni senza un stanza specifica
--disable-menu	Disabilita il menu di comandi interni (da linea di comando)
--audio-loopback name	Gli utenti sentiranno la proprio voce
-h --help	Stampa questo messaggio su console

Generalmente è sufficiente porre solamente i comandi –n e –v per avere un videoconferenza multipoint, ma per maggiori esigenze è necessario utilizzare gli altri comandi.

Es.
openmcu -n, per un MCU solo audio senza la ricerca di un gatekeeper
openmcu -n -v, un MCU con abilità video senza l'ausilio di un gatekeeper

Una volta che l'MCU è in esecuzione, è possible specificare i comandi descritti nella seguente lista, a meno che l'opzione --disable-menu non sia stata esplicitata precedentemente:

Comando	**Descrizione**
?	Visualizza la lista di comandi
V	Visualizza le connessioni video
M	Effettua una chiamata
X	Esce
S	Visualizza le statistiche
Z	Mette i messaggi un file di log

9.1.5 NetMeeting

Microsoft NetMeeting è un client di comunicazione videovocale ricco di funzionalità, quali il supporto degli standard di conferenza internazionali, la condivisione di applicazioni e la conferenza dati.

Grazie alle funzioni di videoconferenza monodirezionale e bidirezionale, gli interlocutori possono ricevere l'immagine di chi chiama anche se non dispongono di una videocamere. Sono anche supportate conversazioni solo audio e chat di testo. Due o più utenti possono collaborare alla creazione di un documento, disegnare su una lavagna o addirittura condividere in tempo reale qualsiasi applicazione di Windows in Internet i nella rete Intranet aziendale. NetMeeting è la distribuzione per videoconferenze di Microsoft ed è integrato al sistema operativo a partire dalla versione Windows 2000. Per le precedenti versioni(95/98/ME e NT) il software è gratuitamente reperibile su internet.

Le versioni di NetMeeting 2.0 e precedenti fanno uso di una particolare procedura di setup di chiamata; questo perché originalmente NetMeeting era un client basato sul protocollo T.120, e per questo motivo imposta prima la chiamata T.120 e dopo la chiamata H.323. E' quindi riscontrabile in tali versioni che la chiamata viene accettata prima che il setup abbia inizio.
Nelle recenti versioni comunque, NetMeeting imposterà prima la chiamata H.323 e poi, se necessario, apre i canali logici per T.120.

L'unica pecca rilevata durante la fase di test è che il software non supporta la funzionalità delle stanze, implementata dall'openMCU precedentemente descritta.

9.1.6 GnomeMeeting

GnomeMeeting è un'applicazione H.323 per VoIP, telefonia IP, e videoconferenza basato sul protocollo H.323(include la libreria OpenH323). Si può connettere con una varietà di altri applicativi H.323, includendo il sopraccitato Microsoft NetMeeting®. Supporta inoltre servers ILS(la versione per microsoft di LDAP). Può funzionare senza una webcam, dando così vita a pure comunicazioni audio, o con una webcam, così lasciando spazio alla possibilità di conferenze video e audio. GnomeMeeting è stato ideato per i desktop Gnome, ma questo non lo preclude dalla possibilità di essere eseguito in altri ambienti quali KDE. Utilizza gconfd-2(1) per salvare i propri dati utente, è offre un pratico wizard per la configurazione del software, e delle GUI attraverso le quali possono essere settate quasi tutte impostazioni. Tra le opzioni di linea di comando troviamo –d per l'attivazione della modalità di debugging durante una chiamata.

Caratteristiche H.323

Conforme alla versione 4 del H.323.
Possibilità di fast start e negoziazione H.245.
Possibilità di chiamate a h.323 e URL.
Supporto del GateKeeper (RAS), con scoperta tramite multicast o broadcast.
Supporto dell'annesso H.235 D.
Supporto di gateway/proxy.
Monitoring di chiamata.
Chiamate da PC a telefono.
Range di porte configurabile.
Chat testuale tramite H.245.
Composizione di chiamata tramite URL, indirizzo IP e E.164.
Supporto di ENUM.
Supporto del NAT.
Chiamate Video e Audio in modalità Mute.
IPv4 e IPv6.

Caratteristiche Codecs

Supporto di codecs audio plugins.

Audio Codecs: iLBC, GSM-06.10, MS-GSM, G.711-Alaw, G.711-uLaw, G.726 e Speex.

Codec audio G.723.1 supportato con schede Quicknet.

Video Codecs: H.261, QCIF e CIF.

Buffer di Jitter dinamico.

Limitazione della larghezza di banda per il video automatica.

Controllo per la trasmissione/ricezione di video.

9.2 Test di verifica

Come abbiamo già anticipato, una volta creato il laboratorio per la videoconferenza, tutte le prove effettuate tra i diversi terminali software possono essere ricondotte essenzialmente a tre tipologie:

- da punto a punto su stessa piattaforma, per verificare l'integrazione, che i diversi client basati o che supportano H.323, manifestano.
- da punto a punto su multipiattaforma, per verificare l'indipendenza dello standard, oltre che dall'applicativo anche dalla piattaforma.
- multipunto multipiattaforma, per vedere come i diversi client si comportano con una MCU, e come questo componente elabora i flussi di videoconferenza.

Si è inoltre effettuato in un secondo tempo, un'ulteriore test, avvalendoci del set-top box (Vega 2 di Aethra) per sala conferenza messo a disposizione dall'Università di Camerino. Inizialmente abbiamo configurato il Vega 2 per interoperare con un applicativo software, e né abbiamo poi testato i comportamenti connesso con l'MCU.

I test sono stati effettuati nella rete LAN (Fast Ethernet a commutazione di frame di 100Mbps) del laboratorio dell'Università di Camerino, e i terminali utilizzati sono i seguenti:

Nome Host	**Indirizzo IP**	**Componente H.323**
PC 08	192.168.1.197	Terminale (Pentium III 2600)
Riccardo	192.168.1.185	Terminale (AMD Athlon 2600)
Docente	192.168.1.179	Terminale (Pentium III 2600)
PC 14	192.168.1.186	Terminale (Pentium III 2600)

9.2.1 Chiamata diretta da punto a punto

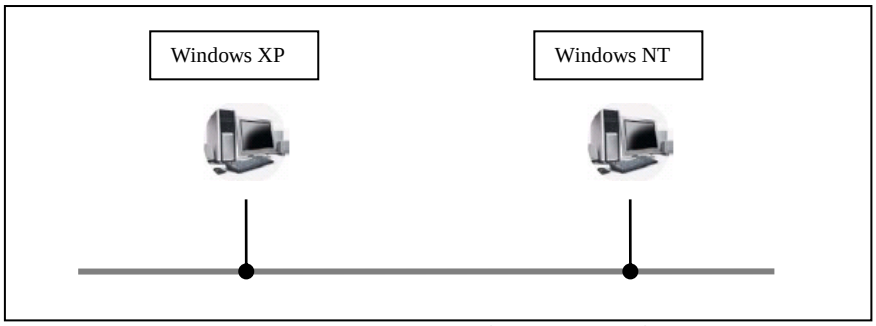

Figura 9.1: struttura del laboratorio di videoconferenza per le verifica punto a punto

Abbiamo effettuate diverse test tra gli applicativi NetMeeting, OhPhone e OpenPhone su Windows e OhPhone e GnomeMeeting per la piattaforma Linux. Inizialmente vediamo cosa succede instaurando una connessione tra due utenti OpenPhone su di un piattaforma Windows e studieremo successivamente cosa succede al livello di trasporto tramite l'ausilio del software Ethereal per vedere quali sono i pacchetti che gli host si scambiano.

OpenPhone(Riccardo) <> OpenPhone(PC 08)

E' stato utile utilizzare questo client, in quanto dotato di un display di logging attraverso il quale l'utente può rilevare lo stato attuale dell'applicativo. Eseguo OpenPhone su entrambi gli host tramite doppio click. Appena avviato il programma stampa a display il messaggio "Waiting for incoming call …", il che significa che il listener è stato attivato ed è pronto oltre che a instaurare nuove chiamate, anche a riceverne. Effettuiamo ora una richiesta di connessione dall'host Riccardo all'host PC 08. I due display stampano rispettivamente "calling 192.168.1.197 …" e "Incoming call from Riccardo [192.168.1.185]" ed accettando la chiamata al logging si aggiunge:

(host Riccardo)

Started receiving G.711-uLaw-64{sw} data (30 frames).
Started sending G.711-uLaw-64{sw} data (30 frames).
Started receiving H.261-QCIF data.
Started sending H.261-QCIF data.
Talking to PC 08 [192.168.1.197]

(host PC 08)

Started sending G.711-uLaw-64{sw} data (30 frames).
Started receiving G.711-uLaw-64{sw} data (30 frames).
Started sending H.261-QCIF data.
Started receiving H.261-QCIF data.
Talking to Riccardo [192.168.1.185]

La connessione è stata stabilita, i due terminali hanno negoziato i canali logici da utilizzare e i codecs per la trasmissione di video e audio. Il risultato ottenuto è il seguente:

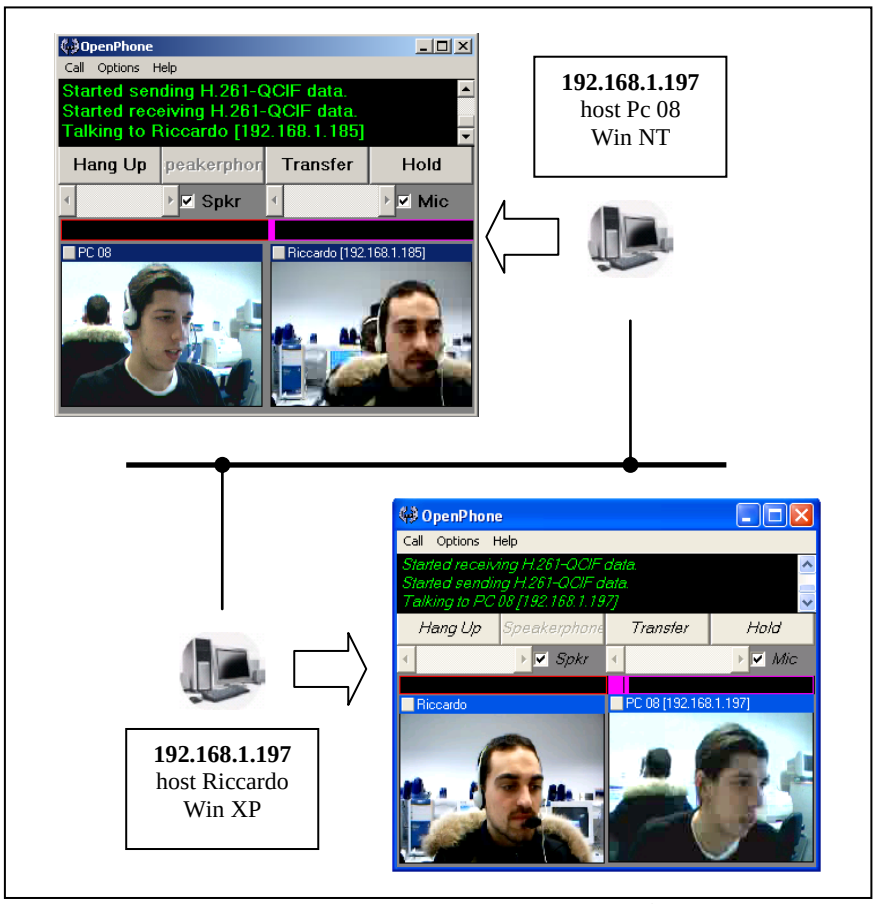

Figura 9.2: due terminali OhPhone connessi in videoconferenza

L'analisi di rete condotta tramite Ethereal ci ha inoltre permesso di vedere l'insieme dei pacchetti scambiati tra i due host, nella loro sequenza temporale. La seguente immagine riporta i pacchetti inviati e ricevuti durante la fase di instaurazione della connessione, puntualizzando che per semplificare la lettura dei diagrammi abbiamo omesso i pacchetti non scambiati tra i due host da tutti i grafici che seguono, escludendo così il messaggio broadcast ARP per la risoluzione dell'IP specificato.

Time	Source	Destination	Protocol	Info
0.000000	192.168.1.185	Broadcast	ARP	Who has 192.168.1.197?
0.000184	192.168.1.197	192.168.1.185	ARP	192.168.1.197 is at 00:

Figura 9.3: messaggio ARP per la risoluzione dell'indirizzo IP

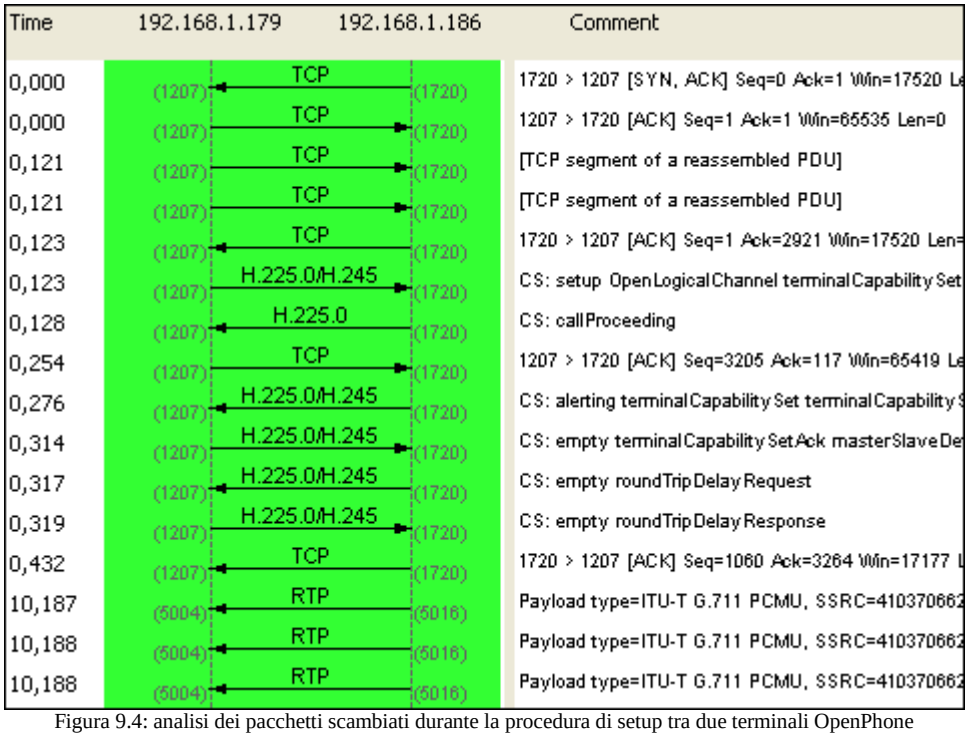

Figura 9.4: analisi dei pacchetti scambiati durante la procedura di setup tra due terminali OpenPhone

Il setup di chiamata viene effettuato tramite lo scambio dei messaggi H.225, con la quale il primo host(Riccardo) manifesta la volontà di connessione. Studiando la rete si è rilevato che il primo terminale invia un solo pacchetto contenente la segnalazione di chiamata H.225, di apertura dei canali logici e le negoziazioni H.245 dei canali, capacità trasmissive; il pacchetto è inviato alla porta 1720, dove il terminale chiamato è in ascolto. Come abbiamo spiegato precedentemente questa procedura viene chiamata fast start. Il secondo utente è concorde alla modalità proposta, risponde alla richiesta di connessione con un callProceeding e invia a sua volta le proprie capacità trasmissive. E' interessante notare come il secondo terminale invii le proprie capacità trasmissive tramite H.245 tunneling. I messaggi di terminalCapabilitySet vengono inviati all'interno di pacchetti H.225, cosicché il canale di call control non viene aperto, tanto che, da come si può notare dalla figura 9.4, vengono utilizzate le stesse porte di call signaling

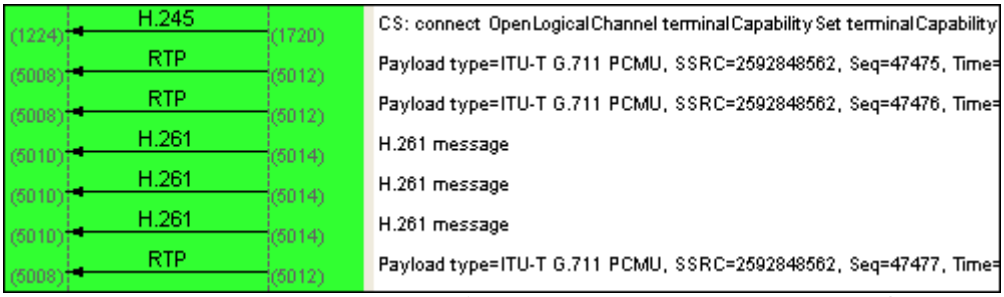

Figura 9.5: pacchetti scambiati per la notifica di connessione avvenuta e scambio di capacità

Nel momento in cui il terminale accetta la chiamata, invia un messaggio H.245 contenente la risposta a setup, connect, e le informazioni relative all'apertura dei canali logici. Qualora l'utente non avesse accettato la chiamata, al posto del messaggio di connect si sarebbe inviato un messaggio di endSession. A questo punto entrambi gli endpoints hanno aperto i canali logici per la ricezione dei dati, e lo streaming ha inizio. Dallo studio dei pacchetti siamo inoltre stati in grado di ottenere quali sessioni sono iniziate e i canali logici correlati. Ciascun terminale ha iniziato sia una sessione RTP per l'audio che una per il video, rispettivamente con Session ID 1 e 2. Ad esse sono attribuiti i canali logici 125 per l'audio e 126 per il video, che trasportano i datagrammi direttamente all'applicazione. I messaggi H.261 infine, sono messaggi contenenti i frame video, mentre l'audio è trasportato nel payload dei messaggi RTP.

Andiamo a vedere ora cosa accade quando uno dei due utenti sceglie di terminate la connessione, tramite il comando di HangUp. Come dovuto i flussi audio e video si interrompono, e il display del logging stampa diversi messaggi:

(host Riccardo)

Hanging up connection
Stopped receiving G.711-uLaw-64{sw} data (30 frames).
Stopped sending G.711-uLaw-64{sw} data (30 frames).
Stopped receiving H.261-QCIF data.
Stopped sending H.261-QCIF data.
Call with PC 08 [192.168.1.197] completed, duration 21:22.

(host PC 08)

Stopped sending G.711-uLaw-64{sw} data (30 frames).
Stopped receiving G.711-uLaw-64{sw} data (30 frames).
Stopped sending H.261-QCIF data.
Stopped receiving H.261-QCIF data.
Riccardo [192.168.1.185] has cleared the call, duration 21:22.

I flussi di video e audio vengono interrotti, e il client torna nella modalità di ascolto, aspettando nuove chiamate o di instaurarne di nuove. I pacchetti catturati da Ethereal sono i seguenti:

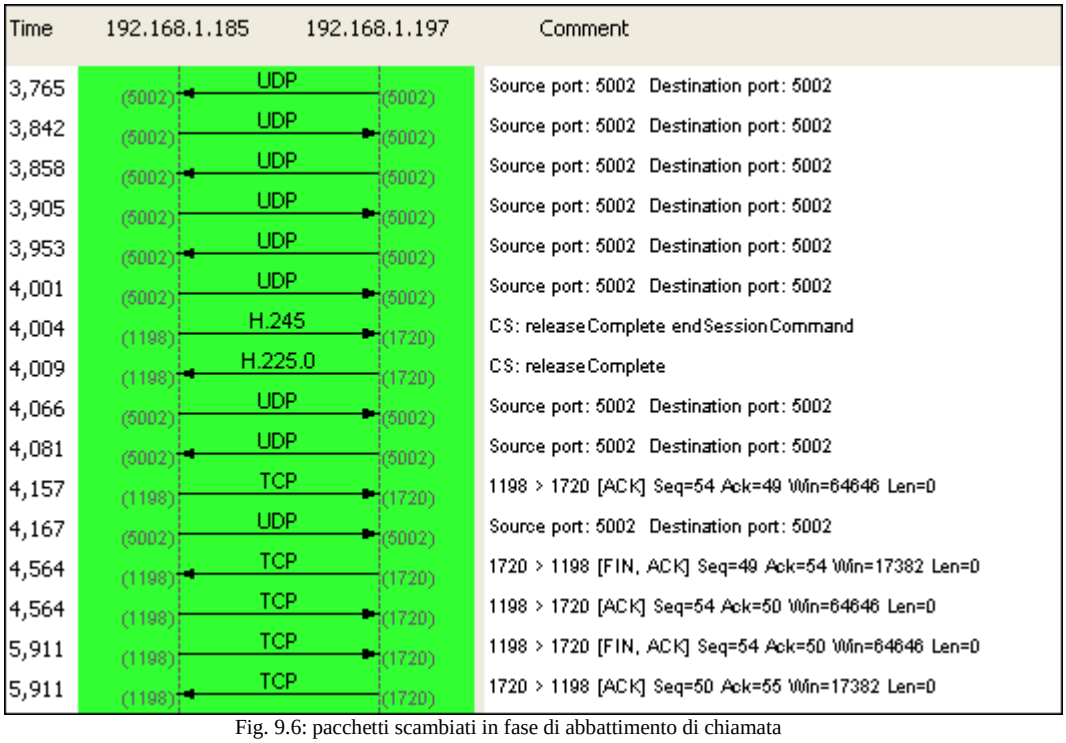

Fig. 9.6: pacchetti scambiati in fase di abbattimento di chiamata

Quando l'utente chiude la chiamata tramite il comando HangUp, il programma invia un segnale al terminale connesso di endSession Command tramite un messaggio H.245, che verrà corrisposto da un H.225 releaseComplete. Tali messaggi implicano la chiusura di tutti i canali logici e la fine della trasmissione per audio e video, come effettivamente avviene.
Ora vediamo l'impatto che una video conversazione ha sulla rete a cui il Pc è collegato. Per far questo utilizziamo il servizio di rete fornito nel Task Manager di Windows.

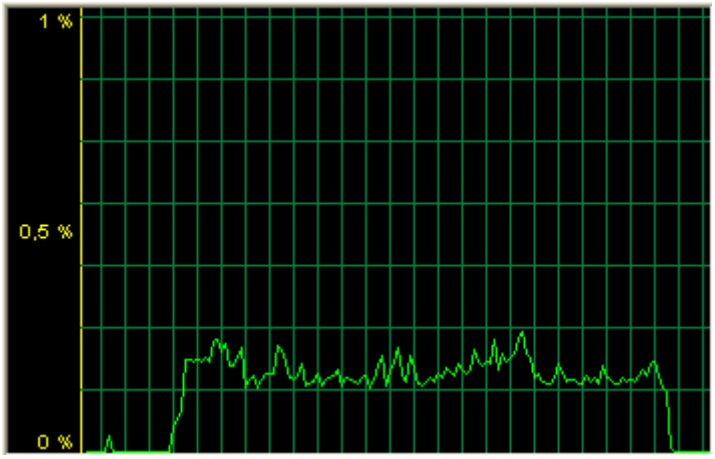

Figura 9.7: banda di rete utilizzata durante la chiamata OpenPhone

Il processo di instaurazione della connessione può arrivare ad utilizza lo 0.7% delle risorse di rete disponibili, dopodichè la quantità di rete utilizzata oscilla mediamente tra il 0.2% e lo 0.3%.

OpenPhone(Docente) <> NetMeeting(PC 14)

Tramite il secondo test, verificheremo la compatibilità di due diversi applicativi basati su H.323. Proviamo quindi a connettere un utente OpenPhone con un utente NetMeeting, per vedere se lo standard H.323 fornisce la compatibilità tra applicativi su di esso basati.
Eseguiamo i due client e proviamo a chiamare l'host PC 14 dall'host Docente e andiamo ad analizzare quali pacchetti sono scambiati tra i due terminali.

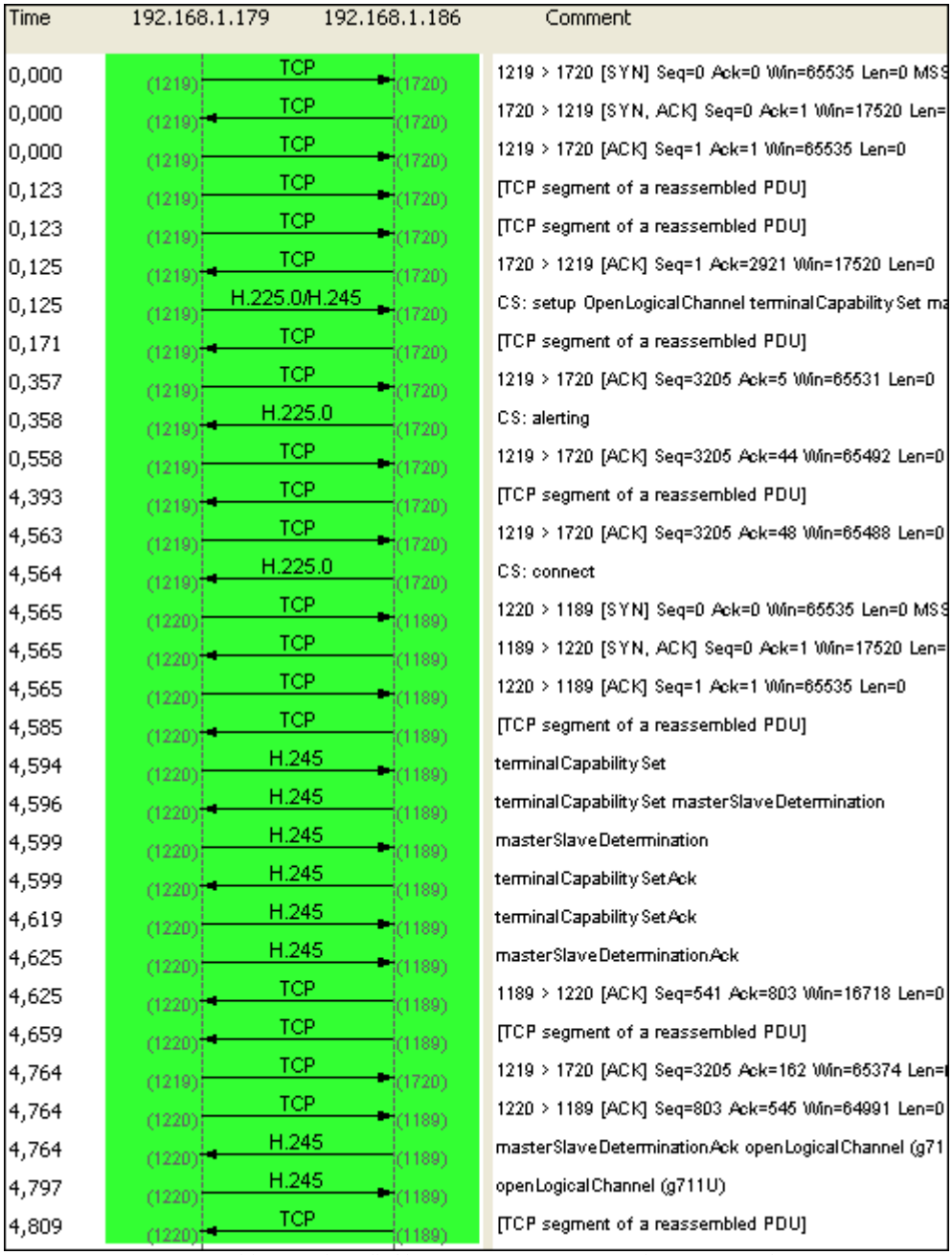

Figura 9.8: l'instaurazione di chiamata tra un utente NetMeeting e un OpenPhone è analoga a quella studiata per i due utenti OpenPhone

Anche in questo caso, come nel precedente, il terminale chiamante invia un messaggio in broadcast per la risoluzione dell'indirizzo IP. Una volta che la locazione del terminale chiamato è nota, si può procedere con il tentativo di instaurazione della chiamata e lo scambio delle capacità trasmissive. Si può notare che la sequenza dei messaggi inviati dal terminali OhPhone per

effettuare il setup è la stessa riportata nel precedente test, ma con la differenza che NetMeeting non accetta la proposta ricevuta, omettendo l'invio di aknowledge. Quest'ultimo inoltre non supporta H.245 tunneling e il canale di call control viene aperto tra i due applicativi(si può notare dal fatto che le porte non sono più le stesse utilizzate per call signaling); i messaggi per lo scambio delle capacità trasmissive vengono scambiati nella maniera classica, tramite messaggi H.245.

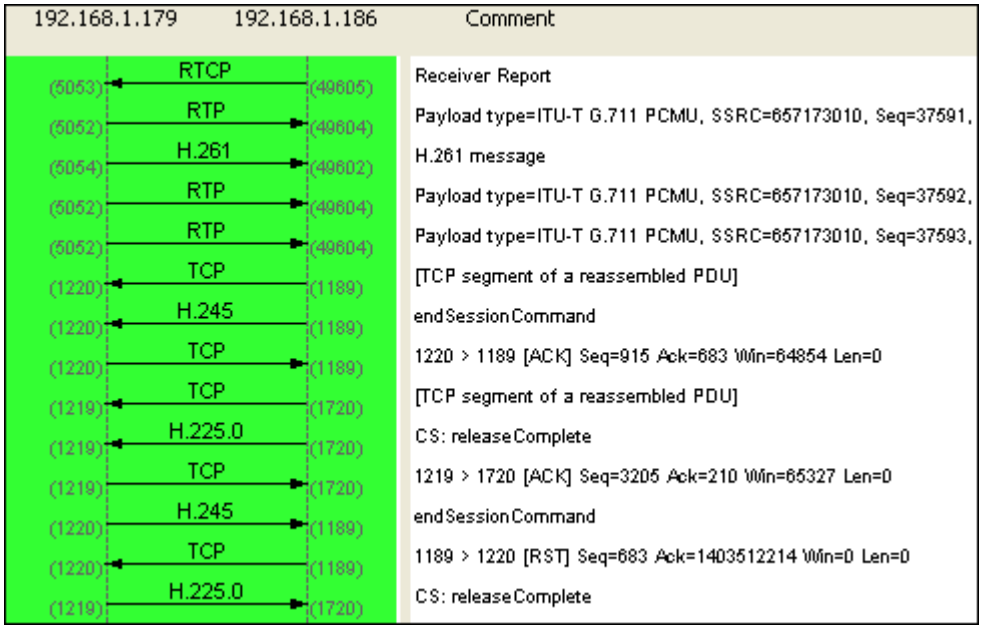

Figura 9. 9: abbattimento della connessione tra NetMeeting e OpenPhone

In maniera analogo è invece l'abbattimento della chiamata, dove i due terminali si scambiano il messaggio di endSessionCommand e chiudono i canali logici. Le sessioni avviate dal OpenPhone sono state le medesime del precedente test, mentre NetMeeting ha dato vita a due sessioni aventi ID 258 e 259 rispettivamente per audio sul canale logico 101 e video su canale logico 102. E' inoltre risultata una leggera variazione nel'utilizzo di rete, che si aggirava tra 0.25 e lo 0.40%. Il test appena descritto ha avuto gli stessi risultati del test in cui si sono connessi OhPhone con NetMeeting.

9.2.2 Chiamate da punto a punto su multipiattaforma

Una volta appurata la compatibilità tra diversi applicativi basati su H.323, andiamo a vedere come lo standard si comporta su due diverse piattaforme per confermare o meno la sua indipendenza dall'architettura su cui opera. Sono stati effettuati due test utilizzando GnomeMeeting e OhPhone per piattaforma Linux, mentre NetMeeting sotto Windows.

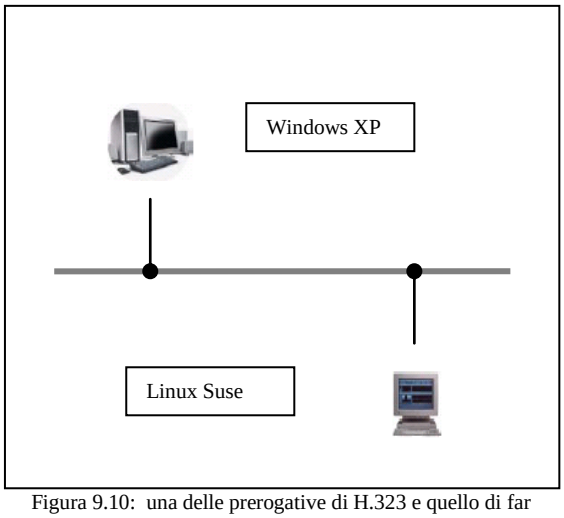

Figura 9.10: una delle prerogative di H.323 e quello di far interoperare applicativi di diversi vendors

OhPhone(Riccardo) <> NetMeeting(PC 08)

Abbiamo eseguito il software OhPhone sull'host Riccardo eseguito sotto Linux, mentre NetMeeting era già in esecuzione in attesa di ricevere chiamate dall'host PC 08. L'ausilio di OhPhone è stato utile per vedere informazioni relative al sistema che in altri applicativi (grafici) non sarebbero altresì espliciti. Infatti questo software, eseguibile da linea di comando, pratica un'attività di logging che porta a conoscenza l'utente di informazioni quali le capacità trasmissive in termini di codec o la modalità con la quale il programma è stato lanciato.

Andiamo a vedere nel pratico. Le opzioni eseguibili in linea di comando sono già state elencate precedentemente, per cui non ci soffermeremo sui comandi dati. Lanciamo il programma tramite il comando ./ohphone -lna --

videotransmit --videoreceive, eseguendo così il client in modalità di ascolto, senza la ricerca di un gatekeeper ed abilitando la trasmissione video. Tutte le altre impostazioni verranno prese per default. Il risultato stampato su console è il seguente:

```
ricca@linux:~> ./ohphone -lna --videotransmit --videoreceive x11
OhPhone Version 1.3.7 by Open H323 Project on Unix Linux (2.6.13-15-default-i686)

Incomming channel port ranges 5000 to 5999
Local username: ricca
TerminateOnHangup is 0
Auto answer is 1
DialAfterHangup is 0
FastStart is 1
H245Tunnelling is 1
SilenceSupression is 1
H245InSetup is 1
Jitter buffer: 50-250 ms
Connect port: 1720

Set video size to be 0
Video receive using device : x11
Video receive quality hint : -1

Video transmit enabled with local video window disabled
Video transmit size is small
Video capture using input 0
Video capture using format PAL
Video picture in picture of local video disabled
Video transmit quality is 0
Video background fill blocks 2
Video transmit frames per sec 0 (default)
Video bitrate 0 bps

Sound output device: "/dev/dsp"
Sound  input device: "/dev/dsp"
Recording using mixer channel mic
Record volume is 100
Play volume is 100
G.711 frame size: 30
GSM frame size: 4
User Input Send Mode: as H.245 string
Codecs (in preference order):
 Table:
   GSM-06.10{sw} <1>
   MS-GSM{sw} <2>
   G.711-uLaw-64k{sw} <3>
   G.711-ALaw-64k{sw} <4>
   SpeexNarrow-5.95k{sw} <5>
   SpeexNarrow-8k{sw} <6>
   SpeexNarrow-11k{sw} <7>
```

```
  SpeexNarrow-15k{sw} <8>
  SpeexNarrow-18.2k{sw} <9>
  G.726-16k{sw} <10>
  G.726-24k{sw} <11>
  G.726-32k{sw} <12>
  G.726-40k{sw} <13>
  LPC-10{sw} <14>
  H.261-QCIF <15>
  H.261-CIF <16>
  UserInput/hookflash <17>
  UserInput/basicString <18>
  UserInput/dtmf <19>
  UserInput/RFC2833 <20>
 Set:
  0:
   0:
    GSM-06.10{sw} <1>
    MS-GSM{sw} <2>
    G.711-uLaw-64k{sw} <3>
    G.711-ALaw-64k{sw} <4>
    SpeexNarrow-5.95k{sw} <5>
    SpeexNarrow-8k{sw} <6>
    SpeexNarrow-11k{sw} <7>
    SpeexNarrow-15k{sw} <8>
    SpeexNarrow-18.2k{sw} <9>
    G.726-16k{sw} <10>
    G.726-24k{sw} <11>
    G.726-32k{sw} <12>
    G.726-40k{sw} <13>
    LPC-10{sw} <14>
   1:
    H.261-QCIF <15>
    H.261-CIF <16>
   2:
    UserInput/hookflash <17>
   3:
    UserInput/basicString <18>
    UserInput/dtmf <19>
    UserInput/RFC2833 <20>


Listening interfaces : ALL:1720
Waiting for incoming calls for "ricca"
Command ?
```

OhPhone è ora in esecuzione e pronto a ricevere o instaurare connessioni. Tramite il logging effettuato su console, è possibile vedere in che modalità l'applicazione è stata eseguita. In questo caso H.245Tunneling e H.245InSetup impostati ad 1 indicano che il terminale può ricevere richieste di connessione e instaurarne una. E' inoltre indicato se la chiamata sarà solo audio, o audio e video e riporta una lista contenente i codec a disposizione.

Ora l'applicazione è in attesa di un comando interno. Alla richiesta "Command ?" apparsa sulla console digitiamo quindi c192.168.1.197, indicando così la volontà di volersi connettere all'host

Command ? c192.168.1.197
ricca is calling host 192.168.1.197
Command ? Ringing phone for "192.168.1.197" ...
Started logical channel: sending G.711-uLaw-64k{sw} <3>
Started logical channel: sending H.261-QCIF <13>
Call with "PC08 Cisco System [192.168.1.197]" established.
Started logical channel: receiving G.711-uLaw-64k{sw} <3>
Started logical channel: receiving H.261-QCIF <15>
"PC08 Cisco System [192.168.1.197]" has cleared the call, duration 0:44

Command ? x
Exiting.
OhPhone Ended.

I canali logici vengono aperti regolarmente, e le sessioni audio e video prendono vita in entrambi i terminali. Sull'host Riccardo si apre una finestra contenente il video catturato da PC 08, mentre il locale non viene mostrato in quanto non richiesto. Analogamente anche PC 08 visualizzerà il video catturato e trasmesso dall'altro utente. Chiudendo la connessione entrambi i client chiudono i canali logici precedentemente aperti e si mettono in ascolto di nuove chiamate.

GnomeMeeting(Riccardo) <> NetMeeting(PC 08)

Anche in questo test di verifica, la comunicazione avviene analogamente alle precedenti. L'immagine dell'interlocutore appare nella finestra del proprio applicativo senza alcun problema e l'audio ha una qualità che non invidia niente ad i suoi "antenati" apparecchi fissi. Forse questo tra tutte le verifiche è il più significativo, non solo perché abbiamo visto come effettivamente la specifica sia indipendente dal contesto operativo sul quale è eseguito, ma perché dimostra come lo standard sia in grado di far interoperare due applicativi appartenenti a due sistemi decisamente conflittuali.

Windows e Linux coprono un'ampia percentuale sul complesso di persone che possiedono un personal computer. Considerando che ognuno di essi ha in dotazione un client H.323 compatibile, in quanto le versioni di sistema operativo uscite da tre anni a questa parte li includono nella loro installazione standard, si può immaginare la posizione che un giorno potrebbe occupare H.323 nella telefonia. E forse è per questo motivo che molti enti incentivano il passaggio alla connessione veloce, prevedendo, in un forse non molto lontano futuro, il passaggio della telefonia su IP.

Fig. 9.11 Linux GnomeMeeting connesso con NetMeeting di Windows

9.2.3 Conferenza multipunto utilizzando l'unità MCU

Come abbiamo visto in precedenza, la MCU è il cuore elaborativo qualora si costituisse una videoconferenza multipunto. Esso è un end-point (in quanto crea degli stream), ma con le capacità aggiuntive di processare i dati entranti, per poi distribuirli ai partecipanti della videoconferenza. Abbiamo inoltre visto che le MCU possono gestire diversi tipi di videoconferenza(centralizza, distribuita, voice activated, ecc…). L'OpenMCU utilizzato in questo progetto è in grado di supportare solamente videoconferenze centralizzate in modalità Voice Activated.
Questo significa che l'unità di multiconferenza riceverà in input tutti gli stream, sia video che audio, dagli endpoint partecipanti. I dati verranno processati e mixati, per poi essere inviati indietro ai mandatari sotto forma di miscela audio/video. Ma mentre lo stream audio restituito conterrà tutti i flussi audio dei partecipanti, quello video conterrà solamente i quattro flussi degli endpoint che più recentemente hanno parlato. Quindi finché la conferenza contiene quattro partecipanti, il video non presenterà nessun cambiamento, ma per un maggior numero si comincerebbe ad avere uno switch di utenti, come definito dalla modalità Voice Activated.

Ma andiamo a vedere nel pratico cosa succede. Creiamo il laboratorio di videoconferenza per la simulazione utilizzando i computer sopra citati come terminali di videoconferenza, ed utilizziamo un ulteriore macchina per svolgere la mansione di server, eseguendovi la MCU.

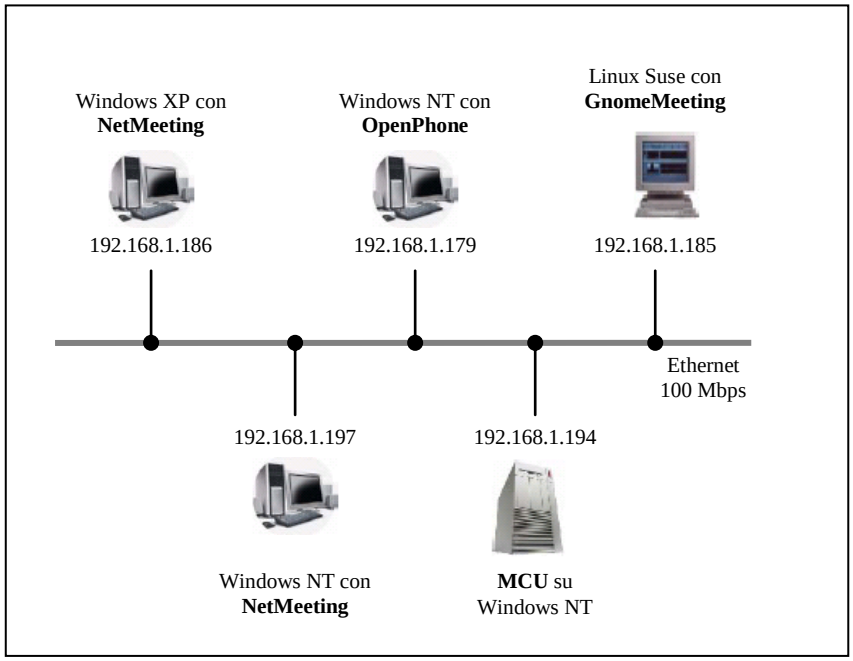

Figura 9.12: layout della struttura del multipunto

Eseguiamo quindi la MCU tramite il comando "openmcu -n -v videotxquality 10", che esegue la multipoint conference unit senza la registrazione ad un gatekeeper(-n), in modalità audio e video(-v) ed utilizzando la qualità 1 (ottima) per il video. Tutte le altre opzioni di videoconferenza verranno prese per default. Analogamente a OhPhone, OpenMCU svolge un'attività di logging su prompt che rende possibile vedere lo status del server e dei suoi partecipanti.

C:\MCU>openmcu -n -v –videotxquality 7
OpenMCU Version 1.1.7 by OpenH323 Project on Windows 2000 (v5.0.2195-i586)
Listening on port 1720
Codecs (in preference order):
Table:
SpeexNarrow-8k{sw} <1>
SpeexNarrow-15k{sw} <2>
GSM-06.10{sw} <3>
MS-GSM{sw} <4>
G.711-uLaw-64k{sw} <5>
G.711-ALaw-64k{sw} <6>
LPC-10{sw} <7>
H.261-CIF <8>
H.261-QCIF <9>

```
Set:
  0:
    0:
      SpeexNarrow-8k{sw} <1>
      SpeexNarrow-15k{sw} <2>
      GSM-06.10{sw} <3>
      MS-GSM{sw} <4>
      G.711-uLaw-64k{sw} <5>
      G.711-ALaw-64k{sw} <6>
      LPC-10{sw} <7>
    1:
      H.261-CIF <8>
      H.261-QCIF <9>

Waiting for incoming calls for "OpenH323 MCU v1.1.7"
Command ?
```

Il server ora è in uno stato di attesa, pronto a stabilire nuove connessioni o ad accettare quelle in ingresso. Possiamo paragonarlo ad un demone, che attende chiamate sulla porta 1720. Tutte le volte che questo riceve un segnalazione di chiamate H.225 avvia il processo di negoziazione per i canali logici e capacità mentre lui si pone nuovamente in ascolto di chiamate sulla porta 1720.
I comandi disponibili sul terminale dell'MCU sono elencati sopra, nella parte relative all'MCU. Vediamo cosa succede quando un terminale manifesta la volontà di connettersi al server:

```
Opening connection
Incoming H.323 call from PC14 Cisco [192.168.1.186] has not selected a room.
Using room room101 as the default.
Accepting call from PC14 Cisco [192.168.1.186] using Microsoft« NetMeeting«/3.0
with room id room101
Member ip$192.168.1.186:1082/5642 will not hear their own voice
Started logical channel: sending G.711-uLaw-64k{sw} <3>
Started logical channel: sending H.261-QCIF <13>
Started logical channel: receiving G.711-uLaw-64k{sw} <5>
Started logical channel: receiving H.261-QCIF <8>
```

In questa prima fase client e server negoziano i canali trasmessivi e definiscono i tipi di algoritmi di codifica/decodifica, dopodichè i canali vengono aperti e la comunicazione ha inizio. In questo caso l'utente, in quanto il solo presente nella stanza "room101", vedrà solo la sua immagine e i restanti tre riquadri grigi. Tramite l'opzioni "s" (statistics) si possono vedere

alcune informazioni relative agli utenti partecipanti alla conferenza e a chi sarà inviato il vide(da cui prende il nome "video corner"). Aggiungiamo altri tre utenti alla videoconferenza e vediamo cosa succede:

Command ? s

Statistics for 4 connected parties in room 1 of 1 with room id room101
Connected to : PC14 Cisco [192.168.1.186] for 2:11 mins
: 4339/1041360 audio packets/bytes sent (H323_muLawCodec)
: 0/0 audio packets/bytes received (H323_muLawCodec)
: Sending video with H323_H261Codec
: Receiving video with H323_H261Codec
Connected to : Pc 08 Cisco [192.168.1.197] for 1:40 mins
: 3299/791760 audio packets/bytes sent (H323_muLawCodec)
: 0/0 audio packets/bytes received (H323_muLawCodec)
: Sending video with H323_H261Codec
: Receiving video with H323_H261Codec
Connected to : Docente System [192.168.1.179] for 0:57 mins
: 1869/448560 audio packets/bytes sent (H323_muLawCodec)
: 0/0 audio packets/bytes received (H323_muLawCodec)
: Sending video with H323_H261Codec
: Receiving video with H323_H261Codec
Connected to : Riccardo [192.168.1.185] for 0:30 mins
: 1026/246240 audio packets/bytes sent (H323_muLawCodec)
: 34/8160 audio packets/bytes received (H323_muLawCodec)
: Sending video with H323_H261Codec
: Receiving video with H323_H261Codec
video for position 0 is "192.168.1.186"
video for position 1 is "192.168.1.197"
video for position 2 is "192.168.1.179"
video for position 3 is "192.168.1.185"
Command ?

MCU mostra inizialmente tutti gli utenti connessi e né da alcune informazioni, quali locazione IP, tempo di connessione e la modalità in cui l'utente è connesso. Se l'utente è connesso in modalità solo audio, le due voci "*Sending video with*" e "*Receiving video with*" sarebbero seguiti da un "*none*", mentre per l'audio sarebbero riportati i pacchetti e il numero di bytes inviati e ricevuti. Se invece, la modalità di videoconferenza è audio e video, il none sarebbe sostituito dal codec utilizzato, come succede in questo caso. Alla lista degli utenti connessi, segue la lista dei quattro terminali che il MCU prenderà in considerazione per elaborare lo stream video da distribuire, i cosiddetti "video corner".

Andiamo ora a vedere che impatto una multiconferenza ha sulla rete dal lato server(al lato client varranno gli stessi discorsi del punto a punto, in quanto openMCU supporta solo la conferenza centralizzata, quindi le modalità di trasmissione dell'MCU saranno le stesse di un comune applicativo H.323).

La qualità del video e la grandezza della finestra sono i fattori che in maniera più rilevante incidono sulla larghezza di banda utilizzata. Questo perché una maggiore qualità delle immagine trasmesse, comporta un aumento per il bit rate del carico da trasportare. Inoltre la qualità del moto di riproduzione incide sul numero di frame trasmessi; più si vuole il moto continuo maggiore saranno i frame per secondo catturati ed inviati agli utenti.
Il prossimo grafico rappresenta la banda utilizzata dal server dal momento della prima connessione:

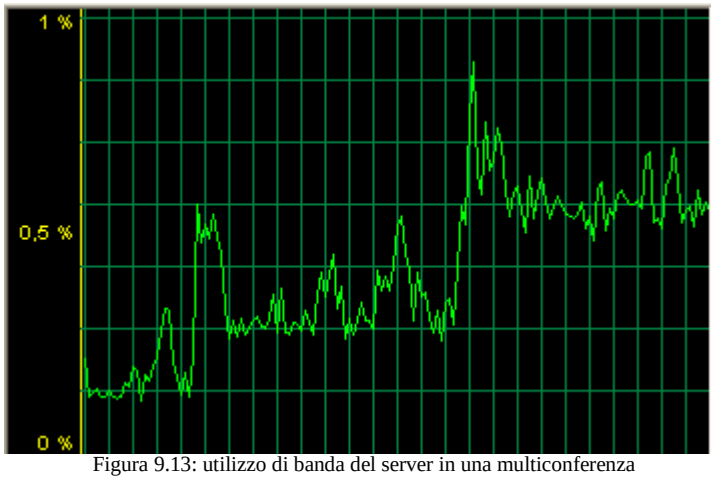

Figura 9.13: utilizzo di banda del server in una multiconferenza

Come si nota dal precedente grafico, la larghezza di banda utilizzata è in continua crescita. Ad ogni aumento notevole corrisponde una nuova connessione accettata dal MCU, e di conseguenza un maggior carico di dati sulla sua rete in quanto ogni terminale entrato a far parte della conferenza invierà i propri stream, sia esso audio, video o entrambi, all'unità centrale per la procedura di mixaggio in un unico stream.

Nel pratico, dopo l'instaurazione di connessione da parte del primo terminale, utilizzando OpenPhone, lo stato d'utilizzo di rete misurata, risulta essere tra lo 0.15% e lo 0.2% del totale.

Successivamente un utente NetMeeting è entrato a far parte della conferenza; il risultato, come è logico pensare, è stato un ulteriore aumento sulla banda utilizzata, fino ad un utilizzo che generalmente variava sullo 0.3% - 0.4%. Ad ogni utente aggiunto alla conferenza corrisponde uno "scalino" sul grafico soprastante, fino all'utilizzo percentuale, in una videoconferenza di quattro utenti, di circa 0.8 – 0.9%. Questo potrebbe anche essere considerato come una limitazione dell'unità MCU; non supportando il multicast deve essere sempre situato in un nodo della rete con banda sufficiente alle stazioni che si desiderano collegare per evitare il degrado della qualità del servizio fornito.

Figura 9.14 Screenshot relativo alla multiconferenza

9.2.4 Conferenza su tecnologia mista

Il punto chiave dello standard H.323 è che non solo riesce a far interoperare applicativi software basati su di esso, ma anche fornire la possibilità trasmissive tra apparecchiature di diversa natura. Questo è anche stato l'ultimo punto che abbiamo voluto toccare in questa tesi, cercando di instaurare una connessione tra un applicativo H.323 basato su pc e un dispositivo dedicato basato su ISDN. Il dispositivo in questione è il Vega di Aethra, utilizzato dall'Università di Camerino per la videoconferenza durante le lezioni con la sede distaccata di Ascoli. Tale apparecchio, connesso ad una televisione consente di allestire sale conferenze di piccola o media dimensione.

Purché si abbia comunicazione tra terminali di differente natura, essenziale è l'utilizzo di un gateway. Per tale proposito è stato utilizzato il Gateway integrato del Vega per la traduzione dei segnali H.320 in segnali H.323 e viceversa. Lo abbiamo poi configurato per la connessione alla LAN del "laboratorio di videoconferenza". Per fare questo abbiamo dovuto configurare la sua interfaccia. Aethra permette di impostare sia la configurazione dell'interfaccia ISDN, che quella IP. Per la nostra sessione di laboratorio è stata rilevante solo quest'ultima, attraverso la quale è possibile impostare sia la configurazione IP, che le impostazioni per H.323. Abbiamo attribuito al Vega 2 un indirizzo IP statico (192.168.1.210), l'indirizzo IP del gateway della rete e l'utilizzo senza un gatekeeper. Tramite la voce "Caratterizzazione del Terminale" possiamo inoltre selezionare la preferenza per il codec audio e video da utilizzare(G.711 per l'audio e H.261 QCIF per il video). Una volta fatto ciò il terminale è operativo per ricevere ed effettuare chiamate su di una LAN, utilizzando il suo Gateway per la conversione dei segnali. Il primo test effettuato è stato la connessione con un terminale NetMeeting.

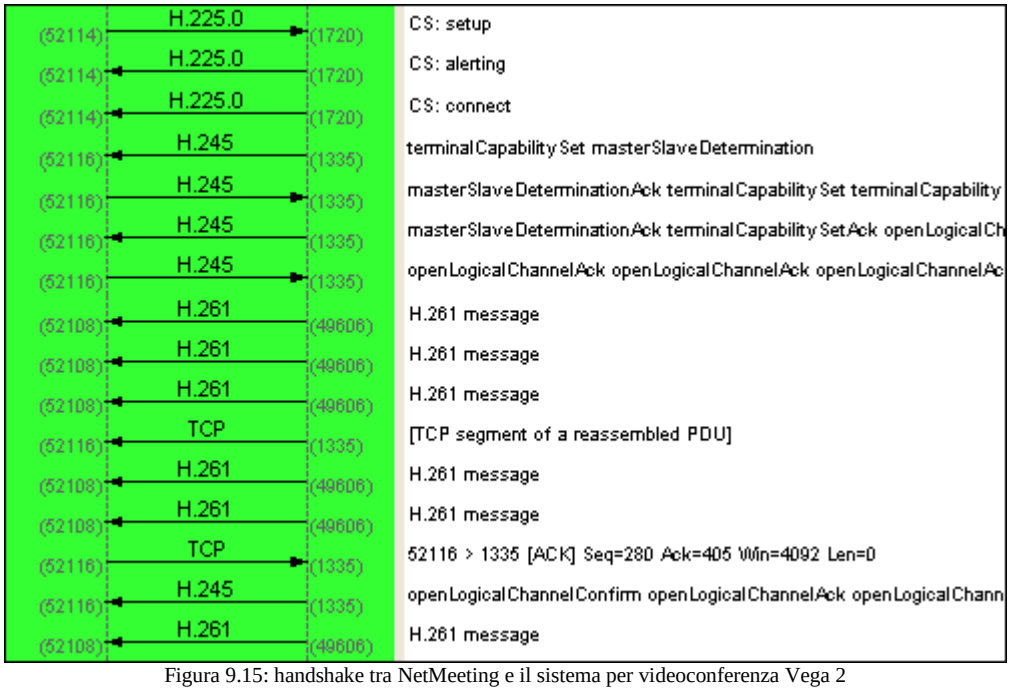

Figura 9.15: handshake tra NetMeeting e il sistema per videoconferenza Vega 2

Le modalità di negoziazione e segnalazione della chiamato sono le stesse riscontrate negli altri test:

il sistema chiamante invia un messaggio di Setup, risposto da un messaggio di Connect quando l'utente accetta la chiamata, per poi passare alla definizione delle capacità trasmissive e l'apertura dei canali logici con lo scambio dei pacchetti terminalCapability e openLogicalChannel. Una cosa da notare sta nel fatto che entrambi i terminali supportano il file sharing e application sharing, viene stabilita un'ulteriore sessione per i dati. Questo dovuto al fatto che durante lo scambio delle capacità trasmissive entrambi i terminali hanno riscontrato la capacità data-trasmission ed hanno ritenuto opportuno instaurare una sessione anche per la trasmissione di dati. I canali logici aperti e le sessioni associati quindi risulteranno:

Tipo di dato	**NetMeeting**		**Vega 2**	
	Session ID	Canale	Session ID	Canale
Dati	0	257	0	0
Audio	1	258	1	1
Video	2	259	2	2

Audio e video vengono trasmessi correttamente e non si riscontra nessuna notevole differenza per quanto riguarda l'utilizzo di rete e le prestazioni della videoconferenza.
Una struttura di questo tipo potrebbe essere adatta a gruppi che dispongono di apparati per videoconferenza ISDN ma vogliono avere la possibilità di partecipare alle riunioni anche dai PC qualora non fosse possibile trovarsi nelle sale opportunamente attrezzate.
Gli stessi risultati si sono ottenuti collegando al Vega 2 l'applicativo per Linux GnomeMeeting e successivamente alla MCU. Quest'utlima è risultata essere funzionante ma talvolta instabile connessa con il Vega, con il risultato che l'unità di multiconferenza va in stallo al seguito di una connessione del Vega. Questo problema non dovrebbe persistere nelle versioni successive a quella dell'apparato da noi utilizzato(la versione 2 risale all'anno 2000). Inoltre non siamo riusciti ad effettuare la connessione con l'OpenPhone; i due sistemi per videoconferenza si scambiano messaggi, ma sembrano non riconoscersi l'un l'altro, probabilmente dovuto a fatto che l'applicativo opensource utilizza la modalità di fast-start la quale potrebbe non essere riconosciuta come segnalazione di chiamata dal Vega.
Riportiamo qui di seguito le tabelle riassuntive riguardanti la connettività ottenuta tra i sistemi di videoconferenza utilizzati durante i test.

1) Tabella comparativa per gli applicativi utilizzati su Windows (prima riga)

Applicativo	**NetMeeting**	**OpenPhone**	**OhPhone**
NetMeeting	Ottima	Ottima	Ottima
GnomeMeeting	Ottima	Ottima	Ottima
OpenPhone(Win)	Ottima	Buono	Buono
OpenPhone(Linux)	Ottima	Buono	Buono
OhPhone(Win)	Ottima	Buono	Buono
OhPhone(Linux)	Ottima	Buono	Buono
Vega	Ottima	No Connessione	No Connessione
MCU	Buono	Ottima	Ottima

2) Tabella comparativa per gli applicativi utilizzati sotto Linux (prima riga)

Applicativo	GnomeMeeting	OpenPhone	OhPhone
GnomeMeeting	Ottima	Ottima	Ottima
NetMeeting	Ottima	Ottima	Ottima
OpenPhone(Win)	Ottima	Buono	Buono
OpenPhone(Linux)	Ottima	Buono	Buono
OhPhone(Win)	Ottima	Buono	Buono
OhPhone(Linux)	Ottima	Buono	Buono
Vega	Buono	No Connessione	No Connessione
MCU	Buono	Ottima	Ottima

9.3 Considerazioni e porte utilizzate

I componenti utilizzati nel laboratorio di videoconferenza si comportano come previsto. Gli applicativi forniscono connettività sia audio che video con terminali di altri produttori e altre tipologie.

La MCU elabora correttamente i flussi, ogni partecipante vede i quattro terminali che per ultimi hanno parlato ed ascolto ogni partecipante alla videoconferenza. E' importante parlare uno alla volta per evitare continui switch del video da parte della MCU. E' consigliato l'uso del "MUTE" per chi ascolta in modo da evitare che rumori accidentali facciano commutare il video involontariamente.

In alcuni casi non è apprezzabile, dal lato dell'applicativo software, la differenza di qualità grafica fra l'apparato ISDN e quelli desktop, ma sono determinanti gli strumenti che si utilizzano per la connessione e per l'acquisizione delle immagini. Al desktop, il miglior risultato, sia in termini di

risoluzione che di full-motion video, è stato ottenuto da GnomeMeeting. Diversamente, per la fluidità del video, è notevole il divario di performance tra gli applicativi software e hardware dedicato.
Per quanto riguarda l'impatto sulla rete, l'utilizzo di banda oscillava mediamente tra lo 0.2% e lo 0,4% (circa tra 200Kb/s e 400Kb/s) nella conferenza point-to-point. Risultava essere leggermente più alta se uno dei due terminali era il Vega 2. La media misurata sulla MCU, con quattro partecipanti connessi, invece risultava essere di circa 0,9%(900Kb/s), con picchi che talvolta raggiungevano l' 1,5% della banda disponibile(1,5Mb/s). Questo conferma la necessità di una connessione a banda larga per le attività che comportano la coesistenza di streaming sia audio che video.

E' possibile collegare più MCU fra di loro in modo da "splittare" il traffico di una stessa videoconferenza per ridurre l'occupazione di banda sui nodi critici (come si fa nelle strutture basate su reflector), ma questo porta con sé l'inconveniente che si creerà una sorta di riflesso video, per la quale il riquadro contenente l'immagine elaborata da un MCU conterrà l'immagine dell'altro MCU, che a sua volta conterrà l'immagine del primo MCU in un riquadro, così entrando in un loop che comporta la visualizzazione di un frame contenente i quattro, ma sempre più piccolo. Questo problema è risolto a livello implementativi dalle MCU che supportano la connessione a cascata.

La sessione di laboratorio inoltre, ci ha messo nella condizione di poter determinare in maniera più o meno approssimata, quali sono le porte utilizzate dai terminali H.323 cha hanno fatto parte dei test. Tali porte devono essere aperte qualora si voglia far funzionare un applicativo dietro un firewall. La prossima tabella riporta le principali porta ai fini di una videocomunicazione:

Protocollo		Tipo	Porta srg	Porta di dst
NetMeeting				
H.225 Call signaling		TCP	Casuale	1720
H.245 Tunneling		TCP	1040-1080	Negoziato*
Stream RTP	Audio	UDP	49590-49610	Negoziato
	Video	UDP	49590-49610	Negoziato
OpenPhone e OhPhone**				
H.225 Call signaling		TCP	Casuale	1720
H.245 Tunneling		TCP	1300-1499	Negoziato
Stream RTP	Audio	UDP	5000-5999	Negoziato
	Video	UDP	5000-5999	Negoziato
GnomeMeeting				
H.225 Call signaling		TCP	Casuale	1720
H.245 Tunneling		TCP	30000-30010	Negoziato
Stream	Audio	UDP	5000-5016	Negoziato
	Video	UDP	5000-5016	Negoziato
MCU				
H.225 Call signaling		TCP	Casuale	1720
H.245 Tunneling		TCP	Casuale	Negoziato
Stream RTP	Audio	UDP	5000-5020	Negoziato
	Video	UDP	5000-5020	Negoziato
Vega				
H.225 Call signaling		TCP	Casuale	1720
H.245 Tunneling		TCP	1024-65535	Negoziato
Stream RTP	Audio	UDP	52000-52300	Negoziato
	Video	UDP	52000-52300	Negoziato

* Durante la negoziazione dei canali H.225, i terminali determinano le porte sulla quale la rispettiva controparte dovrà inviare gli streams. La porta di destinazione quindi potrebbe essere considerata come il range delle porte sorgenti dell'applicativo connesso, per lo stesso protocollo. Tale informazione è in genere contenuta in un messaggio di aknowledge.

** Con OpenPhone e GnomeMeeting tramite l'opzione networking, è possibile impostare quale range di porte dovranno essere utilizzate per pacchetti RTP, TCP e UDP. I valori riportati in tabella sono quelli di default.

Comunque sia la specifica H.323 mette a disposizione la seguente lista di porte "well known", alla quale qualsiasi implementazione basato sullo standard deve adattarsi per avere connettività con altri prodotti. Se per esempio il messaggio di call signaling non venisse inviato alla porta 1720, nessun terminale sarebbe in grado di accogliere la richiesta, in quanto standard è l'utilizzo di questa porta per l'ascolto di chiamate entranti.

Porte e Protocolli usati dai dispositivi H.323					
Porta	**Tipo**	**Protocollo**	**H.323 Client**	**H323 MCU**	**H.323 GK**
1503	TCP	T.120	x		
1718	TCP	GK Discovery	x	x	x
1719	TCP	RAS	x	x	x
1720	TCP	H.323 Call Setup	x	x	
1731	TCP	Data Call Control	x	x	
1024-65535	TCP	H.245	x	x	
1024-65535	UDP	RTP Video Stream	x	x	
1024-65535	UDP	RTP Audio Stream	x	x	
1024-65535	UDP	RTCP	x	x	

10 Alternative all'H.323

Diversamente per il VoIP, nel quale l'H.323 sta vedendo la posizione di leadership, il quale ricopriva, pian piano confermandosi ad un altro protocollo chiamato SIP(Session Initialitazion Protocol), nel contesto della videoconferenza su IP, non è emerso il fenomeno della nascita di nuovi protocolli che hanno provato a difformarsi dalla suite H.323. Tutte le più recenti implementazioni di software sono basate sullo standard, siano esse opensource che applicativi a pagamento. L'unica alternativa per quanto riguarda la videoconferenza al desktop è data dai protocolli proprietari, che offrono la possibilità del video su IP e talvolta anche l'integrazione con H.323 effettuando videochiamate a utenti H.323, ma non il contrario. Popolari esempi di applicativi per videoconferenza basato su protocolli proprietari sono MSN Messenger, IRC ed il noto Skype™, il quale oltre a consentire di effettuare gratuitamente chiamate e videochiamate ad altri utenti Skype™, rilascia regolari chiamate su PSTN a modici prezzi.

11 Conclusioni

Con il termine interoperabilità si vuole indicare la capacità che ha un determinato componente prodotto da un certo produttore di comunicare con altri applicativi indistintamente dalla loro tipologia o marchio; il compito di un protocollo è quello di stendere un comune suolo affinché tale interoperabilità sia possibile, e H.323 svolge pienamente questo compito abbracciando i vari standard per la trasmissione multimediale attraverso varie reti tra cui internet.

Tramite questa tesi si sono confermati i benefici introdotti dalla specifica:

H.323 si accerta che i terminali che vogliono entrare in connessione abbiano le capacità ricettive per la riproduzione dei dati in ingresso, questo perché i terminali sono diventati "intelligenti", ossia capaci di contrattare le modalità che guideranno la conferenza, in base alle loro preferenze così introducendo una notevole flessibilità sul contesto ove lo standard potrebbe operare.

Le funzionalità dell'H.323 inoltre non risentono di limiti tecnologici; la specifica risulta essere indipendente dalla rete e in quanto non la include nella sua suite risulta essere adatta a tutte le tipologie di rete.

Indipendenza anche per quanto riguarda la piattaforma su cui l'applicazione è eseguita: H.323 si basa su comuni librerie e non richiede l'ausilio di alcun hardware dedicato, cosicché le implementazioni possano essere dedicata ad una specifica piattaforma, pur creando un applicativo capace di interoperare con la completa gamma H.323. Sono questi i principali motivi che hanno indotto i maggiori produttori di tool per il multimediale ad integrare nei loro prodotti, H.323 ed è facile capire le ragioni per cui la videoconferenza sia passata dall'immagine di strumento di lusso a popolare applicazione al desktop.

Glossario H.323

Application Sharing.
E' l'abilità implementata da alcuni applicativi che permette a due o più utenti di lavorare insieme nonostante tutti gli utenti non dispongano dell'applicazione. Un utente lancia l'applicazione da condividere e gli altri sono in grado di controllarla da remoto, e gli output ottenuti possono essere facilmente trasferiti, cosicché il risultato ottenuto possa essere disponibile simultaneamente a tutti i partecipanti. Vi è inoltre la possibilità per chi lancia l'applicazione di impedire l'interazione da parte degli altri utenti.

Audio.
Segnale che trasporta i suoni catturati da un terminale.

B-ISDN.
Broadband ISDN. Raccomandazione dell'ITU che estende la suite per il trasporto attraverso ISDN(comprendendo switching, signaling, multiplexing e trasmissione) in una specifica a velocità maggiore.

Bandwidth. Indica la capacità trasmissiva di un canale. Per sistemi digitali corrisponde a bit per secondo, mentre nei sistemi analogici è la differenza tra la frequenza più alta trasportabile e quella più bassa, misurata in Hertz.

Bit. Digit binario. Unità di segnale per tutti i sistemi digitali trasmissive.

Bit rate. Tipicamente espresso in bps, corrisponde al numero di bit di informazioni trasmessi in un dato secondo.

Bps. Bit per secondo, unità di misura per la velocità di trasmissione.

Bridge. Nel dialetto della videoconferenza, un bridge connette tre o più terminali, in modo che questi possano comunicare simultaneamente. Una MCU è quindi spesso chiamato bridge. Il termine bridge si riferisce anche ai dispositivi che interconnettono molteplici chiamate audio.

Broadcasting. Nelle reti PSN, il termine indica l'invio di un pacchetto a tutte le stazioni connesse a quella specifica rete.

Call Signaling Channel. Canale affidabile utilizzato per la trasmissione messaggi di call setup secondo Q.931 o H.225.

CODEC. L'algoritmo che prende in input un segnale analogico e lo converte in un segnale digitale, per poi inviarlo. Dal lato del destinatario un altro codec fa l'opposto, ovvero trasforma il segnale digitale ricevuto in analogico dopodichè questo viene riprodotto. Codec sta per COdec/DECodec.

Document Sharing. Vedi Whiteboard

Endpoint. Un qualsiasi terminale, gateway o MCU.

G.7xx. Famiglia di raccomandazioni dell'ITU, per la codifica/decodifica dei segnali audio.

GateWay. I gateway permettono a sistemi H.323 di interoperare con altri sistemi appartenenti alla famiglia H.32x. E' visto per esempio come il collegamento tra sessioni H.323 e sessioni H.320(basate coè su ISDN).

GateKeeper. E' un componente opzionale che fornisce funzionalità aggiuntive quali controllo di accesso alla rete, gestione della banda e risoluzione di indirizzo. Richiede che l'endpoint si registri ad esso.

Jitter. Differenza di tempo di tragitto tra due pacchetti causato dal fatto che il traffico su IP non da garanzie né sul tempo di arrivo, né sul tragitto.

H.245. Componente della suite H.323 e H.324 che definisce i controlli relativi ad una comunicazione tra due terminali.

H.261. Raccomandazione dell'ITU che permette a due differenti video codec di interpretare come il segnale video deve essere codificato, compresso, decompresso e poi decodificato. Definisce due formati immagine: CIF e QCIF

H.323. Estensione dell'H.320 su Internet, Intranet e Extranet attraverso PSN. Supporta sia il pont-to-point che il multipoint.

H.324. Raccomandazione dell'ITU che provvede data, video e audio in modalità point-to-point attraverso linee telefoniche analogiche. Potrebbe incorporare H.261, ma genericamente usa H.263.

IP. Internet Protocol. E' il più popolare protocollo di rete. Potrebbe essere utilizzato dagli endpoint H.323 per trasmettere audio, video e pacchetti di dati.

ISDN. Integrated Services Digital Network. ISDN è un servizio telefonico completamente digitale che può essere installato a posto della vecchia connessione analogica, con una linea digitale.

ITU. International Telecommunications Union, una degli enti specializzati degli Stai Uniti nello sviluppo di standard per l'interoperabilità di accessory di telecomunicazione attraverso la rete.

Kbps. Kilobytes per second, circa mille bit per secondo.

LAN. Local Area Network. Una rete di computer e altri dispositivi per la comunicazione entro una area ristretta come un edificio o un laboratorio.

Mbps. Megabit per secondo, approssimativamente un milione di bit.

Mixer. Può essere utilizzato in quei casi in cui in un partecipante di una videoconferenza abbia un connessione a bassa velocità. Invece che forzare tutti partecipanti ad adeguarsi alla bassa velocità si potrebbe ricorrere all'ausilio di un mixer, il quale provvederà alla ricostruzione dei pacchetti e sincronizzazione per l'utente.

Multicasting. Modalità con la quale un pacchetto inviato viene ricevuto da più destinatari, i quali sono registrati ad un indirizzo multicast.

Multipoint. Configurazione di comunicazione nel quale tre o più stazioni si connettono contemporaneamente.

Multipoint Controller (MC). Provvede al controllo di tre o più terminali in una conferenza multipunto.

Multipoint Control Unit (MCU). E' l'unità che fa da bridge tra molteplici input, così che tre o più utenti possono partecipare ad una multiconferenza.

Multipoint Processor (MP). Componente che provvede al mixaggio degli stream audio, video e/o data in una multiconferenza.

POTS. Plain Old Telephone Service. Linea telefonica analogica convenzionale basata su banda stretta.

PSN. Public Switched Network, qualsiasi rete basato sullo scambio di pacchetti come LAN o Internet.

Q.931. Protocollo di call signaling per il setup della chiamata. A differenza del H.225 call signaling contiene informazioni addizionali, come ad esempio l'indirizzo IP del destinatario e del mittente.

Quality of Service (QoS). Garantisce le risorse di rete per le richieste di una specifica applicazione.

RAS Channel. Un canale inaffidabile volto alla trasmissione di messaggi di registrazione, ammissione e status tra un entità H.323 ed un gatekeeper.

Trasmissione affidabile. Trasmissione orientate alla connessione che garantisce la sequenzalizzazione, l'assenza di errori ed il controllo dei flussi trasmessi all'utente.

RTP/RTCP (Real-Time Protocol/Real-Time Control Protocol). Le specifiche di IETF per la gestione di segnali audio e video. Permette alle applicazioni di sincronizzare e ottenere dati video e audio in tempo reale.

SCN(Switched Circuit Network). Tutte le reti a commutazione di circuito; include le ISDN e le PSTN.

T.120. Una raccomandazione a ombrello che definisce i protocolli per la condivisione di data durante una qualsiasi videoconferenza H.32x.

TCP. Transmission control protocol. A reliable transport layer on top of IP.

Translator. Viene generalmente usato quando l'applicazione è in un host no raggiungibile via IP, per esempio se dietro firewall, così impedendo a pacchetti IP di entrare. Due translator possono essere locati in ciascun lato del firewall; quello installato al lato esterno filtra tutti i pacchetti ricevuti tramite e tramite una connessione sicura li inoltra al translator interno al firewall, che poi li rinvierà come pacchetto multicast alla rete interna.

Trasmissione inaffidabile. Trasmissione senza connessione, che non offre niente più che il massimo sforzo di consegna per i dati inviati. I pacchetti potrebbero essere duplicati, persi o consegnati fuori ordine.

UDP. User Datagram Protocol. Un inaffidabile livello di trasporto in cima allo strato IP.

Unicast. Trasmissione attraverso una PSN, dove solo un determinato utente riceve i dati. Differisce per questo motivo dal multicast, dove i dati possono essere ricevuti da più di un utente.

Videoconferencing. L'insieme delle tecnologie che integrano video con audio, dati o entrambi per essere inviati in real-time per meeting tra siti dislocati.

WAN. Wide Area Network. Una rete che serve un'area geografica ben più ampia di quella servita dalla LAN o MAN.

Whiteboarding. Il termine è usato per descrivere la funzionalità di alcuni applicative di condividere su schermo documenti. Detta anche "shared notebook" molteplici utenti possono vedere e interagire con il documento.

xDSL. La famiglia di tecnologie the forniscono trasmissione digitale attraverso la cablatura della rete telefonica locale.

Zona. Per la specifica H.323, l'insieme di terminali, gateway e MCU gestiti da un singolo gatekeeper è definita come zona H.323, e include per lo meno un terminale e potrebbe includere segmenti di LAN diversi.