

Tesina di maturità

La rete di Istituto

Applicazioni di test
Applicazioni di test

Realizzazione :

Mainer Bansal Classe	5BZ
Vital Ivo Classe	5BZ

Corso sistemi ed elettronica

Anno Scolastico 2005-06

Sommario

Storia Rete.....	3
I fondamenti della Rete	7
LAN & WAN :	8
Ethernet :	8
HUB & SWITCH :.....	9
Componenti della LAN :	10
Bridge :	10
Modem:	10
Router	11
Switches	11
Gateway :.....	11
Firewall.....	11
Scheda di RETE(o NIC) :.....	11
La velocità di connessione.....	12
LE TIPOLOGIE DI RETE :	12
Caratteristiche della topologia a stella:.....	12
Caratteristiche della topologia ad anello :.....	14
Caratteristiche della topologia a Bus :.....	14
Reti Miste :	15
I protocolli di rete.....	17
Protocolli di trasporto:	17
I Protocolli Applicativi:	17
Le reti CLIENT / SERVER.....	18
Componenti fondamentali di una rete	19
Come nasce la rete all'istituto Moretto (1999)	21
1.1.1. Introduzione	21
1.1.2. Architettura di rete	21
1.1.3. Requisiti dei principali nodi di rete.....	23
1.1.4. Requisiti di collegamento delle macchine alla rete	23
1.1.5. Router amministrazione	24
1.1.6. Server Intranet	24
1.1.7. Firewall	25
1.1.8. Vincoli sull'architettura delle applicazioni.....	25
1.1.9. Sintesi delle attività	26
Oggi la struttura della rete si presenta nel seguente modo:.....	27
Delphi.....	28
Il modello TCP/IP	28
Il protocollo IP	29
HEADER IP	29
Indirizzi IP, classi e Subnetting.....	30
Determinare il numero di sottoreti necessarie.....	34
1.1.10. Da Wikipedia, l'enciclopedia libera.....	35
Programmi realizzati	37

Storia Rete

Ai tempi della *guerra fredda*, in Usa era molto forte la preoccupazione per una possibile minaccia nucleare da parte dell'Unione Sovietica, soprattutto in seguito ai successi spaziali di quest'ultima. Nel 1958, quindi, il Congresso approvò la creazione della **DARPA** (*Defence Advanced Research Projects Agency*), un'agenzia di sviluppo di programmi di ricerca scientifici che avessero possibili applicazioni militari. La sede fu posta nel Pentagono, a Washington.

Uno dei primi problemi che gli studiosi iniziarono ad approfondire fu la **vulnerabilità della rete di comunicazione** del controllo strategico dell'esercito. Si doveva evitare che un eventuale bombardamento interrompesse le comunicazioni e bloccasse la difesa del paese. Doveva essere studiata la possibilità che i messaggi non avessero un'unica via ma potessero avere più possibilità. In tal modo la distruzione di una linea non avrebbe impedito alle istruzioni di arrivare a destinazione.

Nel 1967 Donald Davies, ricercatore del *National Physical Laboratory*, ebbe l'idea di inviare i messaggi da un computer all'altro sotto forma di piccoli **pacchetti**, in modo che ogni calcolatore fosse in grado di inviare e ricevere molti dati contemporaneamente. In questo modo si estendeva a molti computer meno potenti la possibilità di accedere alle risorse dei grandi centri di calcolo. Larry Roberts, sovrintendente agli aspetti tecnici della ricerca della **ARPA**, pensò di applicare questa nuova tecnologia alla rete in studio per collegare i grandi centri di calcolo.

Nel 1969 i ricercatori della **ARPA** pensarono di non **collegare i grandi calcolatori** tra di loro, ma di far gestire a un elaboratore specializzato il traffico dei messaggi. In questo modo tale calcolatore sarebbe diventato il nodo tra i vari collegamenti, e l'insieme dei nodi avrebbe costituito una rete, con molti centri, permettendo a ogni singolo pacchetto di percorrere strade diverse, se necessario. Questo tipo di computer, denominato **Interface Message Processor**, fu realizzato da una piccola azienda, la Bolt Beranek and Newman, che si trovava nei pressi di Harvard e del Massachusetts Institute of Technology, e assumeva i cervelli più geniali delle due università. Il primo modello fu installato il 2 settembre 1969 presso l'Università della California a Los Angeles, dove viene tuttora conservato come reperto archeologico.

L'idea della **posta elettronica** venne per caso, nel marzo 1972, a *Ray Tomlinson*, ingegnere della BBN, l'azienda produttrice degli IMP. Egli pensò di adattare alla rete un **sistema di messaggistica** usato per funzionare su un minicomputer utilizzato da diversi utenti, integrandolo con un programma sperimentale di trasferimento dei file. Fu lui a concepire l'idea di separare il nome della persona da quello della macchina attraverso il carattere **@**. Poiché la rete si stava sviluppando rapidamente, tutti i più importanti centri di ricerca avevano il proprio collegamento, costituendo la prima rete, denominata **Arpanet**. Subito tutti iniziarono a usare la posta elettronica, mandandosi

messaggi di ogni tipo, di lavoro come personali. Questo permise la formazione della **prima comunità virtuale**, composta interamente da giovani esperti di informatica. Da lì a poco ci fu la prima **chat**: un malato di mente a Stanford parlò con il suo medico che era alla sede della BBN.

Nel 1972 si tenne **la prima conferenza internazionale sulla rete** e in quella occasione il gruppo di lavoro si aprì agli studiosi degli altri paesi del mondo, trasformandosi in una organizzazione internazionale che prese il nome di **International Network Working Group**. Il gruppo si sarebbe occupato di sviluppare gli standard per la rete. La direzione fu affidata a **Vinton Cerf**, uno dei più brillanti professori di Stanford.

Nel 1973 Vinton Cerf e Bob Kahn si misero a studiare il problema di come fare a **comunicare reti basate su tecnologie diverse**.

I due risorsero il problema formulando un protocollo, cioè un insieme di indicazioni da fare eseguire ai computer, che fu denominato **Transmission Control Protocol**. Esso permetteva una comunicazione che era totalmente indipendente dall'hardware delle macchine. Inoltre si pensò di collegare le reti attraverso un *gateway*, cioè una macchina che doveva fare da nodo di congiunzione fra reti diverse.

Negli anni Ottanta cioè nel 1983 fu deciso di dividere **ARPA** in due strutture separate per motivi di sicurezza militare: MILnet e ARPAnet.

MILnet sarebbe diventata la *rete chiusa*, e quindi sicura, riservata all'esercito. Dei 113 nodi esistenti, 68 furono riservati a questa struttura. **ARPAnet**, invece, divenne la rete per la *comunità scientifica*: essa aveva a disposizione i nodi rimanenti ma, soprattutto, **non aveva limiti di accesso**, e questo permise il suo enorme sviluppo. Di fatto ARPAnet fu la base del futuro sviluppo mondiale di Internet.

Agli inizi degli anni Ottanta, soltanto 15 dipartimenti di informatica delle università statunitensi avevano il collegamento in rete. Per questo motivo, la ricerca sulla rete non poteva procedere molto velocemente. Il **National Science Foundation**, un ente governativo preposto al finanziamento delle ricerche scientifiche, decise di dare un **impulso alla rete**: finanziò la costruzione di nuovi collegamenti che permisero che ogni dipartimento di informatica (in tutti gli *States* erano più di cento) avesse la propria connessione. Ma il vero boom dei collegamenti avvenne nel 1986, sempre grazie all'iniziativa del NSF. Quest'ultimo si assunse l'onere di costruire un "**backbone**" - dorsale (cioè un collegamento super veloce) tra i cinque maggiori centri di calcolo, invitando tutte le università a finanziarsi un collegamento al *backbone*. In questo modo, in breve tempo, **tutte le università americane furono dotate di un loro collegamento veloce**. Il risultato fu la crescita di dieci volte del numero di utenti della rete in meno di un anno: passarono infatti da *1000 a 10.000*.

Nel 1989 fu evidente che **ARPAnet** aveva perso la propria funzione: tutti i nuovi accessi passavano attraverso la rete fatta costruire dalla NSF. Si procedette al suo **smantellamento**, trasferendo tutti i

suoi siti alla nuova rete o a reti locali. Nacque così Internet.

Nel 1991 Tim Berners Lee, figlio di due genitori entrambi matematici che partecipò alla progettazione dei primi computer, si mise a studiare insieme a Robert Cailliau, si mise a studiare un metodo per poter comunicare via rete con tutti i ricercatori del CERN, con l'obiettivo di poter condividere molte informazioni e risorse del centro. Il risultato fu l'ideazione del **World Wide Web** nel 1991.

Nel 1994 entrò a far parte del gruppo di scienziati del **Laboratorio di Scienza dei Computer**, presso il Massachusetts Institute of Technology (MIT).

Fu uno dei promotori della prima **WWW Conference**, nel 1994, e uno dei membri più importanti del **WWW Consortium**.

L'interesse per il World Wide Web si fece molto forte, ma la sua diffusione era limitata dalla interfaccia a caratteri, che rendeva la lettura difficoltosa.

A dare la svolta alla diffusione del web, fu **Marc Andreessen**, studente specializzando presso il *National Center for Supercomputing Applications* (NCSA) della University of Illinois. Egli concepì l'idea di sviluppare un **browser web grafico**: fu creato **Mosaic**. La facilità di installazione e la **semplicità dell'interfaccia**, attirò in pochissimo tempo migliaia di utenti sul WWW. Per avvicinarsi alla rete non era più necessario conoscere complicati codici e lunghi elenchi di indirizzi. Fu l'inizio del **boom** di Internet.

Yahoo! fu creato da due studenti dell'Università di Stanford, **Jerry Yang** e **David Filo**.

Nel 1993 iniziarono a classificare e recensire, per divertimento, i siti Internet che più incontravano il loro apprezzamento. Il catalogo, che fu chiamato scherzosamente *Jerry's Guide*, comprendeva le recensioni di 200 siti, suddivisi per categorie. Decisero di renderlo disponibile per tutti i naviganti di Internet, mettendolo sul server dell'università. L'iniziativa ebbe un **grandissimo successo**, perché andava incontro all'esigenza diffusa di avere una guida nel variegato e disordinato mondo di Internet. In breve tempo il catalogo raggiunse qualche migliaia di titoli, causando l'**intasamento della linea dell'università** per gli oltre 700.000 accessi che richiamava al giorno. A quel punto Yang e Filo pensarono di trasformare la scherzosa idea in una opportunità di lavoro, fondando, nell'aprile del 1995, Yahoo! Il nome richiama alla mente l'urlo di gioia per aver trovato qualcosa che si stava cercando, ma ufficialmente sarebbe l'acronimo della curiosa frase "Yet Another Hierarchical Officious Oracle".

Nel 1996 la società assunse 80 persone, di cui la metà *surfers*, cioè giovani appassionati di Internet, il cui compito era di viaggiare nella rete, alla ricerca di nuovi e interessanti siti da segnalare. A metà giugno il sito aveva raggiunto 9 milioni di accessi al giorno.

Al momento Yahoo! è il sito più visitato al mondo e i due ideatori sono diventati, ovviamente, miliardari.

Il 25 maggio 1994 si tenne a Ginevra la prima **World Wide Web Conference**, da alcuni battezzata la "Woodstock del Web". I lavori della conferenza durarono 3 giorni: si tennero 49 presentazioni ufficiali, 11 lavori di gruppo e molti forum di discussione su argomenti specifici. Infine, vennero assegnati, alla cena conclusiva, i primi premi della storia di Internet.

La conseguenza più significativa di questo evento fu la nascita del **World Wide Web Consortium**, nell'ottobre dello stesso anno, a opera di Tim Berners-Lee, inventore del World Wide Web. L'organizzazione ha come obiettivo lo sviluppo della rete fino alle sue massime potenzialità. Questa missione si concretizza in tre obiettivi principali: rendere la rete universalmente accessibile, promuovendo tecnologie e investimenti; permettere che ogni utente possa farne il migliore uso possibile, grazie all'introduzione di software sempre più semplici e intuitivi; favorire la sicurezza e la legalità in Internet.

Nel 1995 **Netscape**, l'azienda fondata nel 1994 da *Marc Andreessen* e *Jim Clark*, era leader del mercato mondiale per la produzione di browser. **Navigator**, il browser sviluppato dalle ceneri di Mosaic, veniva utilizzato dall'80% degli utenti di Internet.

In quell'anno, la **Microsoft**, azienda produttrice di Windows, fondata da *Bill Gates*, si rese conto delle grandissime opportunità commerciali fornite dalla rete. Cercò, quindi, di sviluppare, in pochissimo tempo, un prodotto che potesse essere messo sul mercato e contrastare il gigante Navigator. Tra agosto e novembre, gli ingegneri di Microsoft riuscirono a ideare un nuovo browser, chiamato **Explorer**. In esso erano state integrate tutte le potenzialità di Navigator.

Nei mesi successivi iniziò una vera e propria gara tra Netscape e Microsoft per il rilascio di versioni, dei rispettivi browser, sempre più nuove e integrate con le tecnologie nascenti. La fretta con cui vennero svolti i lavori determinò la presenza di moltissimi errori di programmazione (*bach*), che, a loro volta, spinsero a continue revisioni e rilasci.

Microsoft iniziò a legare il proprio browser a Windows, rendendolo parte integrante del sistema operativo, spingendo tutti gli utilizzatori di Windows ad adattarsi, quasi naturalmente, a Explorer.

In meno di un anno, Explorer divenne il *browser più diffuso*, strappando l'80% del mercato a Navigator.

Netscape non si arrese ai risultati e intentò causa a Microsoft, nell'agosto 1996, per **concorrenza sleale**. Il processo fu aperto nel 1997 ma, a tutt'oggi, non è ancora stata emessa una sentenza definitiva.

Per ora il vincitore sembra essere l'azienda di Bill Gates, ma la guerra dei browser non può ancora considerarsi conclusa.

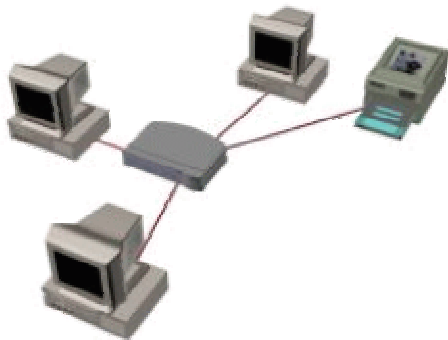
Nel 1996 Internet inizia a diffondersi in modo massiccio anche in Italia. Grazie ai servizi proposti prima da **Video On Line**, poi da **Telecom Italia Network**, la rete comincia a raggiungere anche le case dei cittadini. Nello stesso 1996, gli utenti diventano 405.000.

Ma la **svolta** nella diffusione di Internet è favorita da **Tiscali**, una compagnia fondata nel 1996 da Renato Soru, attuale azionista di maggioranza, che propone la connessione alla Rete gratis. In poco tempo il Web raggiunge il milione di utenti.

A metà del 2000 gli utenti italiani risultano 9 milioni e 400 mila. Ormai Internet può a ragione essere definito un *mass-media* anche in Italia.

I fondamenti della Rete

Mi è capitato spesso di entrare in un laboratorio di informatica e trovare tanti bei computer, magari dell'ultima generazione, ma anche ragazzi vocianti per scambiarsi informazioni su errori di programmazione, mentre l'insegnante correva da una postazione all'altra per rispondere alle nostre esigenze. In tale situazione il problema è facilmente risolto semplicemente connettendo i computer o, come si dice in gergo, installando una rete. Una rete semplifica il lavoro dell'insegnante che, in ogni momento, può controllare dalla propria postazione l'attività degli allievi e può inviare loro messaggi e riceverne. Una rete non è solo questo. Essa permette non solo lo scambio di informazioni, ma anche l'installazione di risorse (hardware o software) a cui potranno accedere più utenti. Questa **condivisione** di risorse è analoga a quanto accade in Internet: unica connessione al provider, ma accesso da parte di tutti gli utenti della rete, con notevole risparmio globale.



In anni recenti, mentre i sistemi operativi e l'hardware si evolvevano, per meglio soddisfare le esigenze dell'utente e renderne più semplice il lavoro, è nata l'esigenza di scambiare dati e informazioni sia all'interno di un'azienda, sia con filiali o più in generale con il mondo esterno.

Il sistema di trasmissione dei dati prende il nome di NETWORK: rete di comunicazione. In pratica una rete informatica è un insieme di PC e di altri dispositivi che sono collegati tra loro tramite cavi. Il sistema consente a questi dispositivi di comunicare tra loro e di condividere informazioni e risorse. Le reti possono avere dimensioni differenti ed è possibile ospitarle in una sede singola oppure dislocarle in tutto il mondo.

LAN & WAN :

Una rete che è collegata su un'area limitata si chiama "Rete Locale" oppure LAN (Local Area Network). Le prime LAN (Local Area Network) si sviluppano negli anni '70, ma si diffondono su larga scala solo nell'intervallo temporale a cavallo tra gli anni '80 e '90. Le LAN sono costituite da gruppi di calcolatori distribuiti su un'area limitata, collegati tra loro mediante cavi e schede di rete.

Grazie all'introduzione delle LAN, i calcolatori sono in grado non solo di comunicare, ma anche di condividere risorse come spazio di memoria o stampanti.

Spesso la LAN è localizzata in una sola sede. Per WAN (Wide Area Network) si intende un gruppo di dispositivi o di LAN collegate nell'ambito di una vasta area geografica, spesso mediante linea telefonica o altro tipo di cablaggio (ad es. linea dedicata, fibre ottiche, collegamento satellitare, ecc.). Uno dei più grandi esempi di WAN è l'Internet stessa.

In maniera graduale, le reti utilizzate dalle aziende per condividere e scambiare informazioni cominciano ad espandersi arrivando a coprire aree di dimensioni sempre maggiori con calcolatori distribuiti tra vari uffici in filiali distanti centinaia di chilometri. Grazie allo sviluppo delle tecnologie informatiche pian piano risulta possibile realizzare reti di calcolatori che si estendono su aree di grandi dimensioni: le cosiddette WAN. Attraverso le WAN, costituite da più LAN collegate tra loro in vario modo, le aziende riescono a condividere informazioni con i propri collaboratori a livello globale e scambiare dati in tempo reale tra filiali che si trovano da un capo all'altro del pianeta. Con il passare del tempo le tecnologie WAN si sono sempre più evolute e consolidate permettendo la creazione di reti a livello globale. Nel mondo moderno con il termine Internet Work ci si riferisce solitamente ad una WAN costituita da un insieme di LAN fisicamente distinte collegate fra loro in vario modo. Per accedere ad una WAN, è necessario un modem o un router. Per accedere ad Internet, occorre avere inoltre un account con un provider di servizi Internet (ISP).

ETHERNET :

Esistono diverse tecnologie LAN; le più comuni sono: Ethernet, Fast Ethernet e Gigabit Ethernet. Una rete può essere formata da una o più di queste tecnologie. Le reti Ethernet, Fast Ethernet e Gigabit Ethernet funzionano in modo simile e la differenza principale è data dalla velocità alla quale trasferiscono le informazioni.

Ethernet funziona a 10 Megabit per secondo (o Mbps), Fast Ethernet a 100Mbps e Gigabit Ethernet a 1000 Mbps.

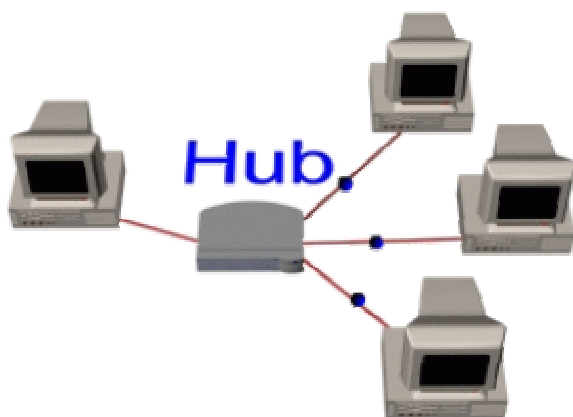
I dispositivi di una rete comunicano trasmettendosi reciprocamente informazioni; le informazioni trasmesse sono gruppi di piccoli impulsi elettrici, detti pacchetti. Ogni pacchetto contiene l'indirizzo del dispositivo che esegue la trasmissione (l'indirizzo di sorgente) e l'indirizzo del dispositivo che riceve i dati (l'indirizzo di destinazione).

Queste informazioni vengono utilizzate dai PC e da altri dispositivi presenti nella rete per aiutare il pacchetto a raggiungere la propria destinazione. Le reti Ethernet e Fast Ethernet impiegano un protocollo chiamato CSMA/CD (Carrier-Sense Multiple Access with Collision Detection). In tal modo può comunicare solo un dispositivo per volta. Quando due dispositivi cercano di comunicare simultaneamente, tra i pacchetti trasmessi si verifica una collisione che viene rilevata dai dispositivi trasmettenti. I dispositivi cessano quindi di trasmettere e attendono prima di inviare nuovamente i loro pacchetti.

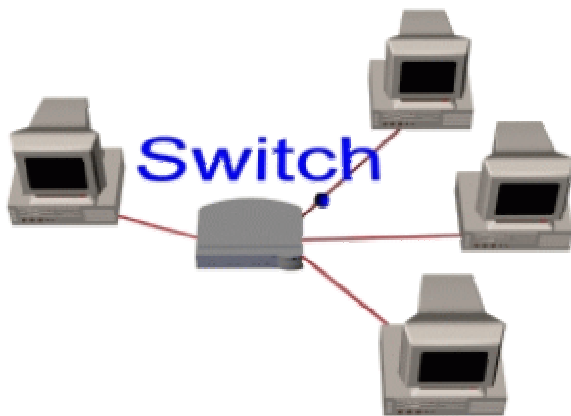
Il meccanismo è paragonabile ad una conversazione tra un gruppo di persone; se due persone parlano contemporaneamente, si fermano entrambe e una di esse inizia a parlare nuovamente.

HUB & SWITCH :

Gli hub e gli switch servono a collegare PC, stampanti ed altri dispositivi di rete. Gli hub si differiscono dagli switch per il modo in cui avviene la trasmissione del traffico di rete. Con il termine "hub" ci si riferisce a volte ad un componente dell'apparecchiatura di rete che collega assieme i PC, ma che in effetti funge da ripetitore. Nel caso, molto diffuso, delle reti Ethernet, un hub è un dispositivo che inoltra i dati in arrivo da una qualsiasi delle sue porte su tutte le altre. Per questa ragione può essere definito anche un "ripetitore multiporta". Questo permette a due dispositivi di comunicare attraverso l'hub come se questo non ci fosse, a parte un piccolo ritardo nella trasmissione. La conseguenza del comportamento dell'hub è che la banda totale disponibile viene ridotta ad una frazione di quella originaria, a causa del moltiplicarsi dei dati inviati. Il ritardo introdotto da uno hub è di generalmente in pochi microsecondi, quindi quasi ininfluenza. La semplicità del comportamento di uno hub ne fa uno dei componenti più economici per costruire una rete.



Uno [switch](#), che si comporta in modo simile ad un hub ma con una maggiore intelligenza, in modo da non sprecare gran parte della banda, è leggermente più complicato e costoso. Gli switch si avvalgono degli indirizzi di ciascun pacchetto per gestire il flusso del traffico di rete. Monitorando i pacchetti che riceve, uno switch "impara" a riconoscere i dispositivi che sono collegati alle proprie porte per poi inviare i pacchetti solamente alle porte pertinenti. Lo switch riduce la quantità di traffico non necessario (sostanzialmente il broadcast), dato che le informazioni ricevute nella porta vengono trasmesse solo al dispositivo con il giusto indirizzo di destinazione, e non come negli hub, a tutte le porte.



Gli HUB sono adatti per piccole reti di pochi PC, ma se la rete ha un elevato livello di traffico o di postazioni si consiglia un'apparecchiatura supplementare di networking ad esempio uno switch che riduce il traffico sostanzialmente in modo da non utilizzare molta banda .

Componenti della LAN :

REPEATER (RIPETITORI) : Rigenerano i segnali per la ri-trasmissione. Muovono i pacchetti da un mezzo fisico all'altro. Lasciano passare le tempeste di broadcast. Non possono connettere diverse topologie o metodi di accesso.

BRIDGE : Vengono usati per segmentare le reti. I bridge "forwardano" i pacchetti basati su indirizzi MAC al nodo di destinazione. Usano la RAM per generare una "routing table" propria basata su MAC address. Connette differenti topologie di rete insieme. Rigenera il segnale a livello pacchetto.

MODEM: Il modem è un dispositivo che va collegato direttamente al computer e che si avvale della linea telefonica per chiamare le sedi (ad es. un servizio online o un ISP). Il compito essenziale di un modem è di convertire i dati digitali necessari al computer in segnali analogici per la

trasmissione attraverso la linea telefonica, e viceversa. Il modem LAN che si avvale della linea telefonica per collegarsi alle sedi remote.

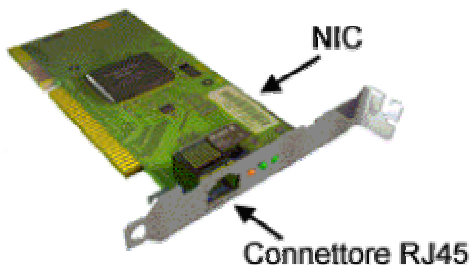
ROUTER :I router instradano i pacchetti attraverso reti multiple o divise in sottoreti. Condividono lo stato e le informazioni di routing a gli altri router della rete per provvedere alla migliore gestione del traffico. Bloccano il broadcast. Sono più lenti dei bridge a causa delle loro complesse funzioni che eseguono. I router possono avere multipli percorsi attivi diversi per instradare i dati tra i segmenti della LAN. Non passano i protocolli non routabili.

SWITCHES : Sono HUB con capacità di "bridging". Switchano il traffico attraverso i MAC addresses. Vengono usati spesso quando si passa (upgrade) ad una rete 1000mb Fast Ethernet.

GATEWAY : Usato per le comunicazioni tra differenti NOS (Network Operating System) es. Windows NT e IBM SNA. I Gateway tolgono dai pacchetti le informazioni di protocollo e li ri-inpacchettano per essere interpretati dalla rete di destinazione. Un Gateway può essere un dispositivo hardware o un software. Converte protocolli diversi / Unisce reti diverse come Windows e UNIX.

FIREWALL Nodo configurato come barriera per impedire l'attraversamento del traffico da un segmento all'altro. I firewall migliorano inoltre la sicurezza della rete e possono fungere da barriera tra le rete pubbliche e private collegate. Possono essere implementati in un router o configurati a tal scopo come dispositivi di rete. Impiegando un firewall è possibile impedire gli accessi indesiderati, monitorare le sedi alle quali si accede più di frequente ed analizzare la quantità di larghezza di banda che la connessione Internet sta utilizzando.

SCHEDA DI RETE(O NIC) : Tutti I PC, per poterli utilizzare in rete, devono essere dotati di schede di rete (NIC).



Alcuni PC sono dotati di NIC preinstallate. Nello scegliere una NIC per un PC, considerare quanto segue:

1. La velocità dell' hub, dello switch o del server di stampa - Ethernet (10Mbps) o Fast Ethernet (100Mbps);
2. Il tipo di collegamento necessario RJ-45 per doppino o BNC per cavo coassiale;
3. Il tipo di connettore NIC disponibile all'interno del PC-ISA o PCI.

LA VELOCITÀ DI CONNESSIONE

Con un hub o uno switch Ethernet, va utilizzata una scheda di rete Ethernet; con un hub o switch Fast Ethernet, invece, va utilizzata una scheda Fast Ethernet.

Collegando il PC ad un dispositivo a velocità duale che supporta sia 10 che 100Mbps, è possibile usare una NIC da 10Mbps o una da 100Mbps. Grazie ad una porta del dispositivo a velocità duale, la velocità è regolata in funzione alla massima velocità supportata dalle due estremità della connessione. Ad esempio, se la NIC supporta solo 10Mbps, la porta del hub a velocità duale, collegata alla NIC, diventa una porta da 10Mbps. Se la NIC supporta 100Mbps, la velocità della porta del hub sarà di 100Mbps. In tal modo, una NIC a velocità duale può essere collegata ad un hub Ethernet da 10Mbps o ad un hub Fast Ethernet da 100Mbps. La NIC a velocità duale regolerà la propria velocità in funzione della massima velocità supportata dalle due estremità della connessione.

LE TIPOLOGIE DI RETE :

I computer sono connessi ad un HUB/Switch centrale.

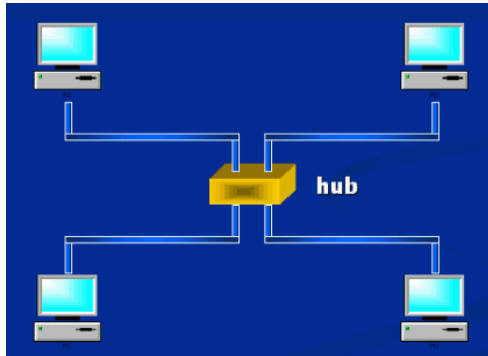
I computer sono connessi ad un componente centrale chiamato Hub. I dati sono inviati dal computer trasmittente attraverso l'Hub a tutti i computer della rete.

Nelle topologie di rete esistono tre differenti reti cioè:

1. Topologia ad STELLA;
2. Topologie ad ANELLO;
3. Topologie a BUS

CARATTERISTICHE DELLA TOPOLOGIA A STELLA:

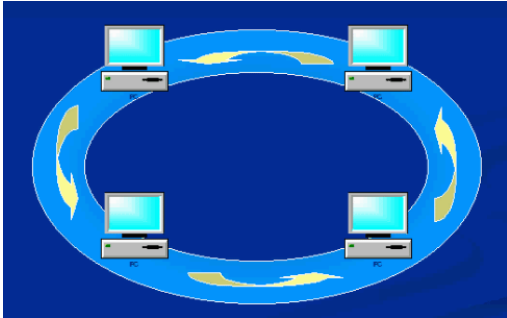
In una rete a Stella i calcolatori sono tutti collegati ad un componente centrale chiamato HUB



Quando un calcolatore deve inviare un messaggio sulla rete, il messaggio giunge all'HUB centrale e quindi a tutti gli altri Pc direttamente collegati. Il principale vantaggio della tipologia a stella consiste nel fatto che quando si interrompe il collegamento tra uno dei Pc e l'HUB centrale, solo il Pc in questione non riesce più a inviare e ricevere dati, tutti gli altri continuano a lavorare senza problemi.

CARATTERISTICHE DELLA TOPOLOGIA AD ANELLO :

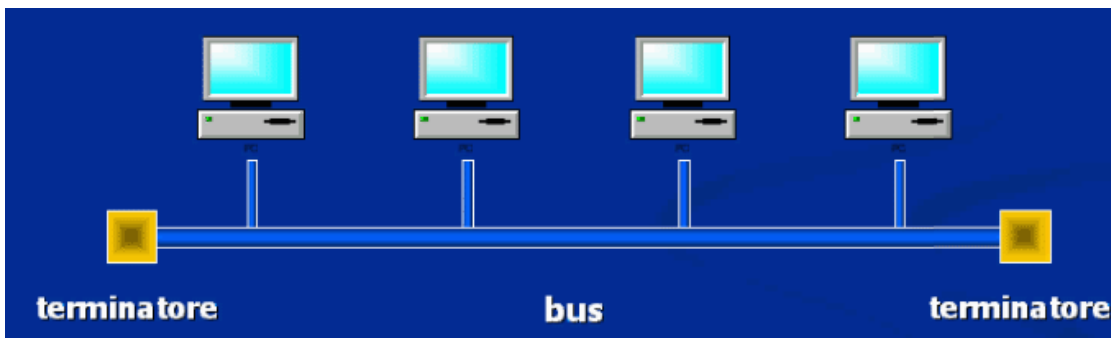
In una rete che utilizza la topologia ad anello tutti i Pc sono collegati tramite un unico cavo che rappresenta un anello logico. Il segnale viaggia attraverso l'anello in una sola direzione attraverso i computer che costituiscono i nodi della rete fino a raggiungere il Pc di destinazione. Ogni nodo funge da ripetitore del segnale che viene amplificato di passaggio in passaggio.



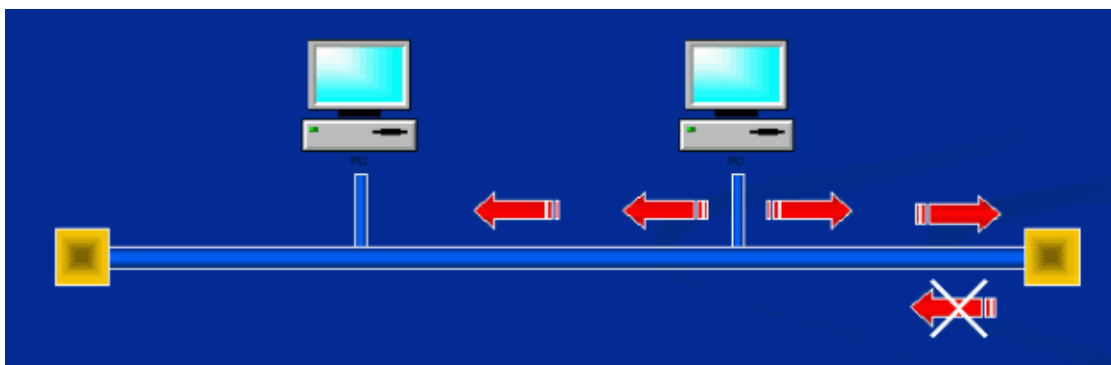
Nelle reti ad anello il metodo utilizzato per la trasmissione dei dati è basato sul concetto di TOKEN. Un **TOKEN** è un insieme di BIT che viaggia sull'anello contenente informazioni di controllo. Cioè aggiunge al dato l'indirizzo del destinatario e quello del mittente rinviandolo lungo l'anello. I dati passano attraverso ciascun computer finché raggiungono quello il cui indirizzo corrisponde a quello indicato sui dati. Questo computer restituisce un messaggio di conferma al computer trasmittente il quale crea un nuovo token e lo immette nella rete. Un token può percorrere un anello di 200m di diametro 10.000 volte al secondo, poiché viaggia alla velocità della luce. Nelle reti **Token Ring**, a differenza di altre, un computer malfunzionante viene automaticamente escluso dall'anello consentendo agli altri di continuare a funzionare regolarmente in rete. In altri tipi di reti ad anello, un computer che non funziona può provocare la caduta di tutta la rete.

CARATTERISTICHE DELLA TOPOLOGIA A BUS :

La topologia a bus, detta anche topologia lineare, rappresenta la struttura più semplice da implementare. E' costituita da un singolo cavo cui sono collegati da tutti i Pc che costituiscono i nodi della rete. Quando un calcolatore deve inviare dati a un altro computer trasmette le informazioni sul cavo servendosi della propria scheda di rete. Le informazioni viaggiano sul supporto fisico fino a raggiungere tutti i computer della rete ma solo il destinatario o i destinatari del messaggio processano e leggono i messaggi inviati.



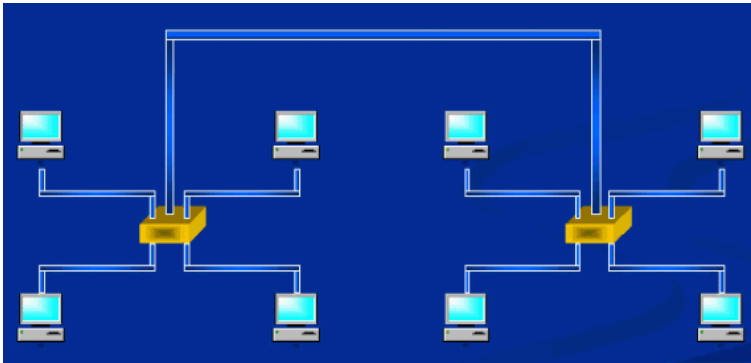
Poiché un solo computer alla volta può inviare dati, maggiore è il numero di computer connessi alla rete, più saranno i computer in attesa di trasmettere dati, rallentando le prestazioni dell'intera rete. Quella a bus è una tipologia di rete passiva: i computer ascoltano i dati trasmessi sulla rete, ma non intervengono nello spostamento di dati da un computer a quello successivo.



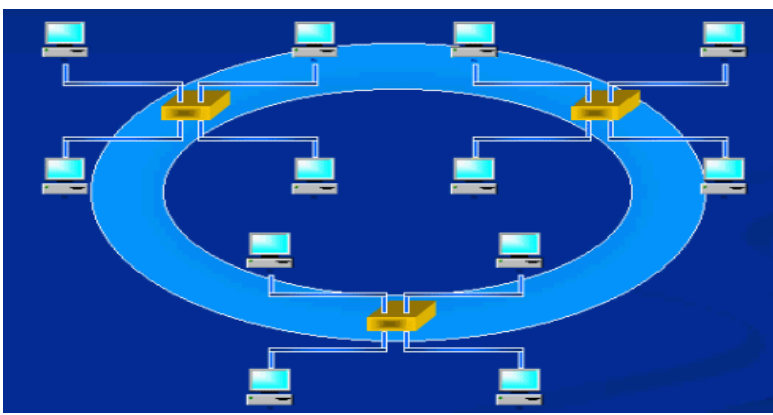
I dati trasmessi da un computer, se non vengono interrotti, viaggiano da un capo all'altro del cavo, rimbalzano e tornano indietro impedendo ad altri computer di inviare segnali. A ciascuna estremità del cavo viene applicato un componente chiamato terminatore che assorbe i dati liberi rendendo disponibile il cavo per l'invio di altri dati. Se un cavo viene tagliato o se uno dei capi viene scollegato, e quindi uno o più capi sono privi di terminatore, i dati rimbalzeranno interrompendo l'attività su tutta la rete (rete inattiva).

RETI MISTE :

Caratteristiche della topologia a bus a stella: in cui due o più reti che utilizzano topologie a stella vengono collegate attraverso un BUS. In caso di mancato funzionamento di un hub, tutti i computer connessi a quell'HUB saranno esclusi dalla rete. Se l'HUB a sua volta è collegato ad altri HUB, anche queste connessioni saranno interrotte.



Caratteristiche della tipologia a stella ad anello : in cui due o più reti che utilizzano la tipologia a stella vengono collegate in modo da formare un anello.



I protocolli di rete

Il Protocollo non è altro che il "Linguaggio" con cui vengono scambiati i dati e le informazioni tra computer in rete.

PROTOCOLLI DI TRASPORTO:

IP - Internet Protocol: Responsabile del trasporto di pacchetti di dati da una sorgente (identificata da un indirizzo IP) ad una destinazione (identificata da un altro indirizzo IP). Se necessario questo livello del protocollo si occupa di spezzettare i pacchetti troppo grandi in pacchetti di dimensione adatta alla rete da utilizzare.

ICMP - Internet Control Message Protocol: Partner di IP con la funzione specifica di inviare, anziché dati, messaggi di controllo e diagnostici (ad esempio pacchetti ECHO).

UDP - User Datagram Protocol: Questo protocollo si trova ad un livello superiore rispetto ad IP, ed aggiunge alla semplice funzionalità di trasporto di IP la possibilità di "smistare" i pacchetti nella macchina di destinazione sulla base di un numero di porta aggiunto all'indirizzo. Viene controllata l'integrità dei dati attraverso una checksum, ma i pacchetti corrotti vengono semplicemente buttati via. **TCP - Transmission Control Protocol:** Questo è il protocollo di livello superiore ad IP che viene utilizzato più di frequente. La sua caratteristica è quella di stabilire una connessione fra due applicazioni identificate, come in UDP, da un numero di porta, e di garantire la trasmissione senza errori di un flusso di dati. Se vengono ricevuti pacchetti corrotti, il protocollo richiede la ritrasmissione dei dati a partire dal primo pacchetto corrotto identificato. TCP implementa anche un timeout per la chiusura delle connessioni interrotte o non stabilite.

PPP - Point to Point Protocol: Permette di trasferire traffico IP su una linea seriale. Creato in particolare per gestire i collegamenti transitori via modem, comprende meccanismi di auto-configurazione delle estremità del collegamento e di autenticazione.

I PROTOCOLLI APPLICATIVI:

FTP - File Transfer Protocol: Anche questo fra i primissimi protocolli applicativi ad essere sviluppati. Consente di trasferire file fra macchine di architettura diversa. I file vengono trattati come file di testo (7 bit per carattere) oppure come file binari (8 bit per carattere). Non viene modificato o "tradotto" il contenuto dei file.

HTTP - HyperText Transfer Protocol : E' il protocollo che interconnette quella vastissima collezione di siti Internet generalmente nota come World Wide Web (WWW). Non ha molta funzionalità in più rispetto a FTP: permette in più di richiedere l'esecuzione di procedure via rete. E' però forse oggi il protocollo di alto livello di IP più utilizzato in assoluto, perché viene utilizzato per veicolare i documenti codificati in HTML (HyperText Markup Language). E' la funzionalità di questo linguaggio, unita all'interfaccia grafica fornita dai browser, la vera ragione della praticità d'uso, e quindi del successo di WWW.

SMTP - Simple Mail Transfer Protocol : E' il protocollo utilizzato per trasferire (fra host che "parlano" TCP/IP) i messaggi di posta elettronica.

POP - Post Office Protocol: Protocollo utilizzato per recuperare i messaggi di posta elettronica conservati su un host remoto. Nato per permettere l'accesso ai servizi di posta alle macchine non collegate direttamente ad Internet, viene recentemente sempre più spesso utilizzato anche su LAN a causa dei problemi legati alla configurazione di un server di posta "sicuro".

IMAP - Internet Message Access Protocol: Protocollo speculare rispetto a POP: permette di esaminare una casella remota di posta elettronica senza trasferire i messaggi. L'uso e la sua ragione d'essere sono sostanzialmente gli stessi di POP.

Le reti CLIENT / SERVER

Le reti locali si estendono su di un'area geografica piuttosto limitata; si tratta per lo più di reti aziendali che si sviluppano su uno o più uffici o laboratori e che al massimo occupano lo spazio fisico di un edificio. Secondo una prima classificazione, le reti locali possono essere suddivise in due grosse categorie, quelle paritetiche e quelle client/server. Noi tratteremo più in generale sulle reti Client/Server.

UN SERVER è un computer che mette a disposizione le proprie risorse (memoria, potenza di elaborazione,periferiche) a disposizione per gli altri Pc della rete.

I CLIENT sono computer dotati di memoria e capacità elaborativi locale che utilizzano le risorse che i server mettono a loro disposizione. La gestione di un server di questo tipo richiede necessariamente l'implementazione di un sistema operativo di tipo server, come ad esempio Windows 2000 server o Linux. Nel primo tipo, detto anche Peer To Peer non esiste una gerarchia ben precisa all'interno della rete ed ogni computer si può indifferentemente comportare sia da server che da client; nel secondo tipo, invece, il server ha la precisa funzione di gestire ed amministrare la rete ed il client, assume una posizione subordinata accedendo alle

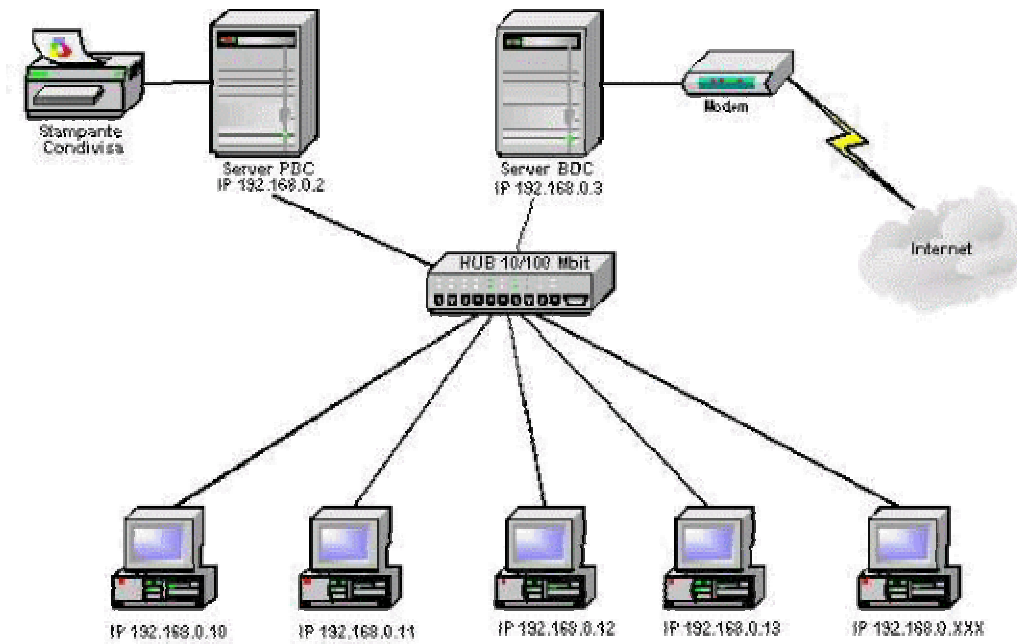
risorse condivise dal server. Dal punto di vista amministrativo, le reti client server, tipicamente basano il loro funzionamento sul concetto di dominio. Un dominio è un insieme di calcolatori che viene amministrato in maniera centralizzata in cui un utente superpartes ha il controllo completo sull'intera rete. Questo utente, detto amministratore del dominio, è in grado di creare account per gli altri utenti, gestirne le password, configurarne l'ambiente di lavoro, distribuire software ed impostare permessi. In questo caso il server ha la precisa funzione di gestire ed amministrare la rete ed il client, assume una posizione subordinata accedendo alle risorse condivise dal server e per ciò necessitano di una gestione più complessa, che, essendo centralizzata, conferisce un alto grado di sicurezza alla rete; ovviamente questo tipo di rete trova applicazione nelle aziende di una certa dimensione dove occorre un livello di sicurezza più elevato.

Componenti fondamentali di una rete

Prima di proseguire, si rende necessario, a questo punto, fare una panoramica sui componenti fondamentali di una rete:

1. un mezzo trasmissivo (cavi di rete RJ-45 Cat.5) per ogni postazione;
2. una scheda di rete Ethernet (10Mbit) o Fast Ethernet (10/100Mbit) per ogni postazione;
3. un protocollo di trasmissione;
4. un dispositivo ripetitore (Hub o Switch);
5. uno o più server;
6. logicamente anche uno o più client;
7. un sistema operativo di rete per il server: Windows NT, Windows 2000, Linux, ecc...;
8. un sistema operativo di rete per ogni client: Windows 95/98, NT Workstation, Windows 2000 Professional, Windows Me, Linux, ecc...

Il seguente è un tipico schema dei componenti presenti in una Rete Client/Server in ambiente Windows.



In questo caso e a topologia a stella cioe tutti i messaggi devono passare per un computer centrale che controlla il flusso dei dati. Per ovvi motivi, in una rete a stella è facile l'aggiunta e la rimozione di computer periferici.

D'altra parte è altrettanto ovvio che in una rete a stella se il computer centrale smette di funzionare l'intera rete (come tale) diviene inutilizzabile.

Come nasce la rete all'istituto Moretto (1999)

1.1.1. Introduzione

Questo documento raccoglie alcune proposte e idee riguardanti l'architettura di rete e l'architettura di applicazioni intranet all'istituto Moretto.

Le proposte nascono dall'analisi delle specifiche della rete definite nel documento "La rete di Istituto : riunione del 5.XII.98 (Preside, Modiano, Prandelli, Azzani)" e dalla successiva riunione del 24.3.99 presso l'istituto Moretto.

Questo documento tratta i seguenti aspetti:

- architettura di rete: soluzioni tecniche per garantire la sicurezza tra la area didattica e area amministrativa;
- requisiti dei principali nodi di rete: una proposta per i requisiti HW e SW per le principali macchine coinvolte nell'architettura di rete;
- architettura applicativa: proposta per l'architettura ad alto livello delle applicazioni Intranet

1.1.2. Architettura di rete

Per soddisfare i requisiti di sicurezza richiesti occorre sezionare elettricamente in due parti (sottoreti) la rete dell'istituto: la sotto-rete della didattica e la sotto-rete amministrativa.

La sotto-rete didattica collega tutte le macchine dei laboratori ed in generale le macchine accessibili fisicamente agli studenti (ovvero macchine poste in aule, locali pubblici etc.)

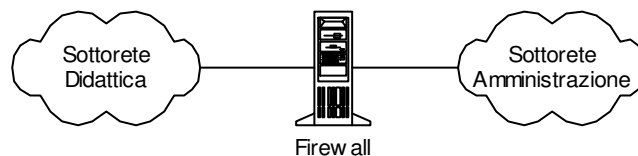
La sotto-rete amministrativa collega tutte le macchine poste negli uffici e non fisicamente accessibili dagli studenti.

Ogni sotto-rete realizza propri metodi sicurezza (es. accessi e password) ma, indipendentemente dalle regole di sicurezza impostate sulle singole reti valgono i seguenti principi:

- Nessuna macchina attestata sulla rete didattica ha accesso alla rete amministrativa.
- Le macchine attestate sulla rete amministrativa possono (se autorizzate) accedere alle macchine poste sulla rete didattica.

In sostanza si prevede il collegamento delle due sottoreti attraverso una speciale macchina chiamata "firewall" che, essendo collegata ad entrambe le reti tramite due schede, di governare l'interscambio di dati tra le due aree.

Per consentire l'accesso a Internet da entrambe le sottoreti si consiglia di impostare due accessi



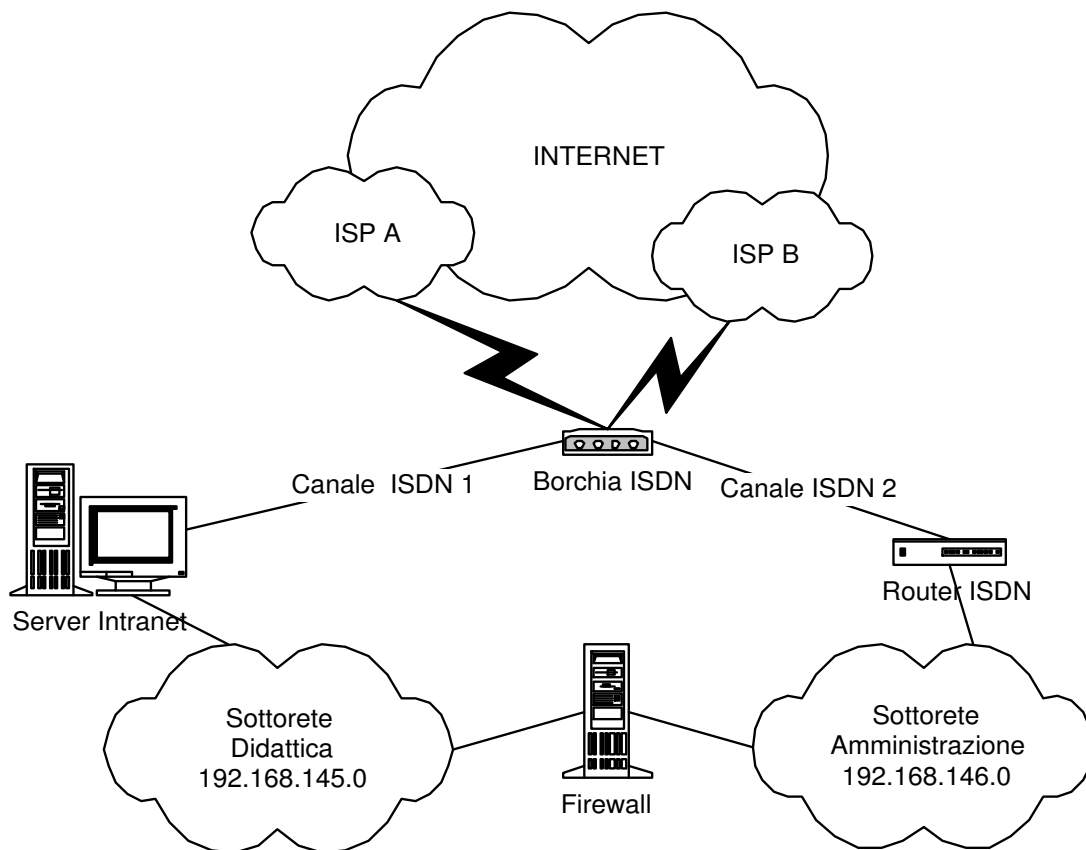
separati via ISDN. La soluzione proposta presenta i seguenti vantaggi:

- consente di impostare politiche di accesso differenti tra rete amministrativa e didattica (scelta ISP, velocità, etc.)
- consente agli studenti più capaci (sempre sotto adeguato controllo) di sperimentare le tecnologie di comunicazione senza porre problemi di sicurezza e affidabilità alla rete amministrativa.

- aumenta il throughput complessivo della rete, separando il traffico amministrativo da quello degli studenti.
- Gestisce in modo ottimale la capacità della linea ISDN che consente appunto due connessioni contemporanee a 64k.

Lo svantaggio di questa soluzione è dato dalla necessaria duplicazione degli apparati di collegamento (schede o router) e la duplicazione del contratto con ISP. Si sottolinea che una singola connessione ISDN (BRI) supporta due canali dati indipendenti a 64Kbps ed e' quindi sufficiente per supportare due accessi ISDN Internet distinti in completa sicurezza.

In sostanza l'architettura di rete è schematizzata dalla seguente figura:



I principali componenti della architettura sono:

Server Intranet	Server primario della rete didattica con funzioni di: - accesso alla rete Internet e funzioni di NAT (Network Address Translation) - server Web intranet - data-warehouse per le applicazioni intranet - mail server della rete didattica (studenti) - DNS primario di rete
Router amministrazione	Router per accesso Internet da parte della sotto-rete amministrativa con funzioni di NAT
Firewall	Punto di contatto tra la sotto-rete didattica e la sotto-rete amministrativa con le seguenti funzioni:

	- firewall - mail server rete amministrativa - default gateway - DNS secondario di rete
Borchia ISDN	Accesso BRI ISDN dedicato che consente l'attivazione di due connessioni a 64Bbps verso due ISP

1.1.3. Requisiti dei principali nodi di rete

Questo capitolo raccoglie i requisiti HW, SW e di servizio dei principali nodi di rete definiti nella architettura descritta nel capitolo precedente.

1.1.4. Requisiti di collegamento delle macchine alla rete

Per garantire l'isolamento delle due sottoreti è necessario il rispetto delle seguenti regole:

- nessuno switch o hub attestato sulla rete Moretto può contenere sia macchine della sotto-rete amministrativa che macchine della sotto-rete didattica.
- le macchine della sotto-rete didattica e le macchine della sotto-rete amministrativa devono avere indirizzi tratti da due classi di indirizzi IP distinte. Si consiglia di utilizzare un indirizzamento statico (gestito tramite DNS) in cui le macchine della rete didattica abbiano indirizzi appartenenti alla rete 192.168.145.0 mentre le macchine della rete amministrativa abbiano un indirizzo della rete 192.168.146.0.

Operativamente:

- tutte le macchine della rete didattica devono avere assegnato un IP univoco scelto nel range 192.168.145.3 - 192.168.145.254 (l'indirizzo 1 viene riservato per il firewall, il 2 al server intranet)
- tutte le macchine della rete amministrativa devono avere assegnato un IP univoco scelto nel range 192.168.146.2 - 192.168.146.254 (l'indirizzo 1 viene riservato per il firewall)
- agli indirizzi va assegnato un nome mnemonico da registrare tramite DNS sul server intranet (di entrambe le reti)
- nessun hub o nessuno switch può avere collegato contemporaneamente macchine con indirizzi 192.168.145.x e macchine 192.168.146.x

Le macchine nella sotto-rete didattica avranno quindi la seguente configurazione di rete:

Host name:	<nome univoco>
IP address:	192.168.145. x (x= numero univoco tra 3 e 254)
Netmask:	255.255.255.0
Default gateway:	192.168.145.1 (il firewall)
DNS primario:	192.168.146.2 (server intranet)
Domain:	rd.moretto.edu

Le macchine nella sotto-rete amministrativa avranno la seguente configurazione di rete:

Host name:	<nome univoco>
IP address:	192.168.146. x (x= numero univoco tra 3 e 254)
Netmask:	255.255.255.0
Default gateway:	192.168.146.1 (il firewall)
DNS primario:	192.168.145.1 (firewall intranet)
Domain:	ra.moretto.edu

1.1.5. Router amministrazione

Il router dell'amministrazione ha la funzione di collegare ad Internet la sotto-rete dell'amministrazione. E' necessario che sia programmato per:

- gestire la chiamata al provider quando necessario (dial-on-demand)
- mascherare gli indirizzi interni con l'indirizzo assegnato dal provider (NAT)

Operativamente si può mantenere l'attuale router con lievi modifiche alla configurazione oggi in uso.

1.1.6. Server Intranet

Il server intranet è una macchina centrale nella architettura di rete proposta. Si trova nella sotto-rete didattica e svolge i seguenti importanti compiti:

- collega la rete didattica a Internet
- gestisce il DNS (Domain name system) principale per entrambe le reti
- gestisce la posta elettronica degli studenti
- fa funzionare le applicazioni intranet su interfaccia Web
- gestisce il database con le informazioni messe a disposizione dalla area amministrativa

Pur gestendo importanti servizi, un malfunzionamento sul server intranet non pregiudica in alcun modo l'operatività e la sicurezza della rete amministrativa.

Ciò detto è comunque indispensabile che il server intranet sia amministrato con attenzione e utilizzando tutte le tecniche di controllo e prevenzione accessi non autorizzati. Il server intranet dovrebbe essere posto in un locale non accessibile agli studenti, possibilmente non nello stesso locale del firewall.

Si consiglia la seguente configurazione HW:

- PC intel Pentium II (anche un compatibile va bene)
- Minimo 128Mbyte di RAM (256 consigliati)
- 8GB hd
- Scheda ISDN
- Scheda rete Ethernet

Si consiglia inoltre la seguente configurazione per il SW di base:

- NT Server (4.0 o Windows 2000 server quando disponibile)
- IIS 4.0
- MS Proxy server
- MS SQL Server 7.0
- MS Exchange

Il software può essere acquistato da Microsoft usufruendo delle particolari condizioni riservate agli istituti didattici. Probabilmente esistono agevolazioni specifiche a livello di provveditorato.

1.1.7. Firewall

Come abbiamo spiegato prima il firewall è un nodo di rete particolarmente delicato, il suo compito principale è di mettere in comunicazione le due sottoreti governandone il traffico in modo controllato tramite regole (policy).

Fisicamente la macchina ha due schede ethernet di cui una ha un indirizzo nella sotto-rete didattica (192.168.145.1) e l'altra ha un indirizzo della sotto-rete amministrativa (192.168.146.1).

La configurazione della macchina è delicata e soggetta a poche modifiche. La gestione di tale macchina è quindi complicata anche se relativamente poco onerosa (sono richiesti interventi solo a fronte di guasti).

Esistono in commercio molti tipi di firewall, alcuni esclusivamente HW altri basati su HW e SW. In generale si sconsiglia l'utilizzo di macchine NT come base per sistemi Firewall.

Il firewall basato su ONION Interprise è in grado di svolgere le seguenti funzioni:

- controllo del traffico tra le varie sottoreti e tra le singole macchine di reti diverse;
- routing tra le sottoreti;
- DNS secondario;
- Mail server per area amministrativa;

1.1.8. Vincoli sull'architettura delle applicazioni

La realizzazione della infrastruttura descritta nei capitoli precedenti non è sufficiente per garantire la realizzazione delle specifiche richieste.

E' infatti necessario organizzare le applicazioni di rete su un modello che si potrebbe definire di data-warehouse.

In questo modello, la macchina che realizza i servizi disponibili in Intranet, non accede direttamente alla sorgente dei dati ma interroga un proprio database che contiene una copia dei dati reali, riorganizzata sia per logica che per tecnologia di accesso.

I dati vengono periodicamente rinfrescati dalle applicazioni nell'area amministrativa

In breve, le applicazioni Intranet avranno le seguenti caratteristiche:

- funzioneranno tutte sul server Intranet (ad esempio potrebbero essere delle applicazioni scritte con la tecnologia ASP)
- le applicazioni intranet accederanno ai dati posti su un database (es. MS SQL server 7.0 o Access) posto sul server intranet stesso.
- Il database conterrà solo i dati strettamente necessari alla realizzazione delle applicazioni. Non conterrà le informazioni sensibili che risiedono esclusivamente nella rete amministrativa e quindi tali dati non potranno essere acceduti dalle applicazioni Intranet.
- L'accesso al database ed alle applicazioni sarà soggetto alle regole di sicurezza impostate sul server Intranet tramite le funzionalità di NT server.
- Gli applicativi nella sotto-rete amministrativa dovranno, ad intervalli di tempo schedulati (es. tutte le notti), inviare in un formato standard (ASCII o Access) porzioni dei propri archivi sulla macchina Intranet. In caso di archivi molto grossi, (oltre 10M) l'invio potrebbe riguardare solo i record modificati. Nel caso gli applicativi dell'area amministrativa non prevedano una funzione di export parziale dei dati occorrerà approntare dei piccoli programmi specifici.
- Periodicamente (es. ogni notte) il database sul server Intranet processerà i nuovi dati ricevuti inserendoli nella propria struttura. In pratica l'intero database sul server Intranet potrebbe essere completamente ricostruito ogni notte.

- Occorre sottolineare che, grazie all'impostazione del firewall, gli applicativi dell'area amministrativa potranno senza problemi accedere a tutti i dati presenti sul server Intranet. Un'altra possibilità di comunicazione tra applicativi Intranet e l'area amministrativa è data dall'e-mail.

Questa architettura presenta molti vantaggi:

- il server può accedere solo ad un insieme di dati limitato (quindi sono protetti i dati più sensibili)
- i dati sono presentati alle applicazioni con una logica ad esse ottimizzata. Ne consegue una maggiore semplicità di scrittura delle applicazioni.
- i dati possono essere organizzati in un formato coerente e efficiente indipendentemente dal formato dei dati sorgente.
- un problema di sicurezza o sistemistico sul server Intranet non pregiudica i livelli minimi di sicurezza e non pregiudica le applicazioni dell'area amministrativa

Esistono ovviamente alcuni svantaggi:

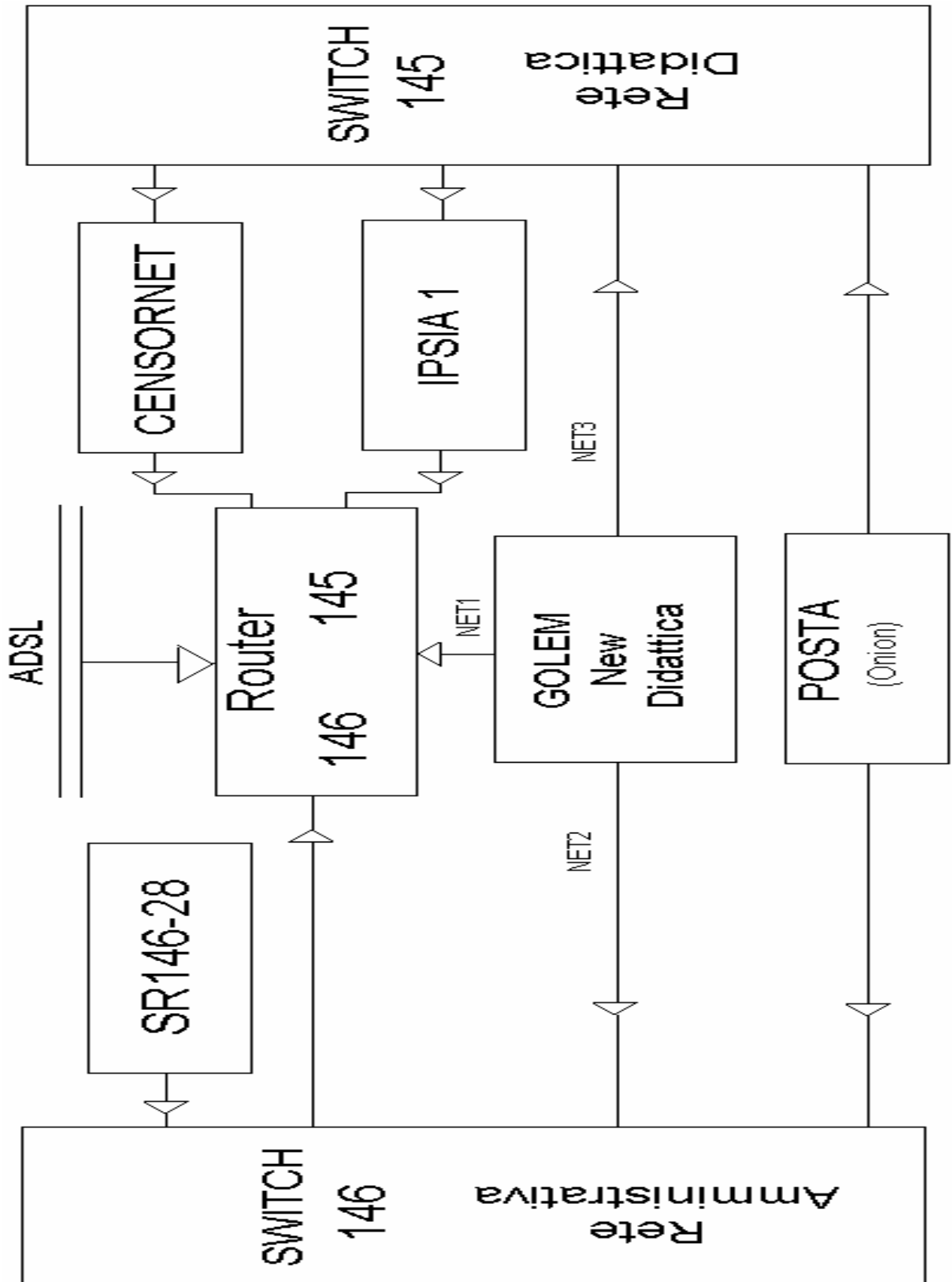
- i dati importati sono in read-only e non possono essere modificati dagli applicativi Intranet
- i dati non sono aggiornati in real-time ma il loro aggiornamento dipende dalla frequenza di rinfresco (tipicamente una volta al giorno)
- i dati più sensibili non sono trasmessi dalla amministrazione al server intranet e quindi non sarà possibile per le applicazioni intranet accedervi.

1.1.9. Sintesi delle attività

In sintesi ed ad alto livello ecco le attività che si consigliano di realizzare:

1. Separare le macchine della didattica dalle macchine dell'amministrazione utilizzando hub e switch differenti.
2. Attuare le diverse politiche di indirizzamento IP sulle varie macchine
3. Assicurarci l'utilizzo dedicato di una linea ISDN (borchia) scollegata dal centralino
4. Impostare il server intranet con particolare attenzione ai componenti di accesso a Internet (RAS e Proxy)
5. Attivare un nuovo accesso Internet per la didattica
6. Riconfigurare il router amministrazione
7. Installare e configurare il firewall
8. Richiedere dominio Internet (es. moretto.edu o moretto.it) e impostare il DNS
9. Impostare e configurare la posta elettronica per la parte amministrativa su firewall
10. Impostare e configurare la posta elettronica sul server Intranet (Exchange)
11. Realizzare su server intranet la struttura in HTML del sito intranet
12. Configurare tutte le macchine
13. Definire le politiche di sicurezza e di uso della rete
14. Eseguire i corsi per diffondere l'uso della posta elettronica e intranet e le policy di sicurezza
15. Pianificare e progettare le applicazioni Intranet
16. Installare e definire la struttura del database sul server Intranet
17. Pianificare e realizzare l'export dei dati sulle applicazioni amministrative
18. Pianificare e realizzare l'import dei dati su server intranet
19. Scrivere le applicazioni Intranet

Oggi la struttura della rete si presenta nel seguente modo:



Delphi

Delphi è un ambiente di programmazione visuale ad oggetti per lo sviluppo rapido di applicazioni (RAD/Rapid Application Development) a carattere generale e di applicazioni client/server per Windows 95 e 98 e Windows NT. Con Delphi è possibile creare applicazioni Windows altamente efficienti riducendo al minimo i tempi di programmazione.

Delphi comprende una libreria di componenti riutilizzabili VCL e un insieme di strumenti di progettazione RAD, tra cui i modelli di applicazioni standard e di schede expert di programmazione. Con questi strumenti e con il compilatore Delphi a 32 bit è possibile creare rapidamente e testare prototipi, trasformandoli in robuste applicazioni perfettamente in linea con le moderne esigenze.

Delphi può essere utilizzato per sviluppare qualsiasi tipo di applicazione, dalle utility di analisi e test dei PC, fino ai più sofisticati strumenti di accesso ai database.

Gli strumenti di gestione dei database e i componenti di gestione dei dati previsti in Delphi permettono di sviluppare strumenti di gestione dati e applicazioni client/server in tempi notevolmente ridotti.

Con i controlli di gestione dei dati di Delphi, i dati vengono visualizzati direttamente durante la creazione dell'applicazione, consentendo una immediata verifica del risultato delle interrogazioni al database e delle modifiche all'interfaccia dell'applicazione.

L'ambiente di sviluppo integrato di Delphi IDE (Integrated, Development, Enviroment) mantiene sviluppo, verifica e gestione delle applicazioni in un unico ambiente. E' possibile creare o modificare una applicazione compreso schede di inserimento dati, report, menu, finestre di dialogo, database e definizioni di file, moduli dati, componenti, senza uscire da Delphi.

Per i nostri progetti aiutati sicuramente da professore Azzani abbiamo creato grazie a delphi i programmi utili per il controllo continuo dei PC in rete.

[Il modello TCP/IP](#)

Il modello TCP/IP, il quale prende il nome da i due protocolli principali di Internet progettati nel 1974 da Robert E. Kahn e Vinton G. Cerf a Berkeley, descrive il processo di trasmissione dati riassumendo i 7 livelli del modello ISO/OSI.

- **Application** (*Applicazione*): rappresenta i layer **Application**, **Presentation** e **Session** del modello OSI. Questo livello si occupa di fornire alle applicazioni (FTP, Telnet, Mail) i **servizi di rete**, compresi la **rappresentazione** dei dati, ed il mantenimento di **sessioni**;
- **Transport** (*Trasporto*): rappresenta l'omonimo layer del modello OSI. Questo livello si occupa del trasporto attraverso i protocolli **TCP** (Transmission Control Protocol) di tipo **connection-oriented** e **UDP** (User Datagram Protocol) di tipo **connectionless** permettendo quindi di **stabilire comunicazioni** tra due host;

- **Internet** (*Internet*): rappresenta il layer **Network** del modello OSI. Questo livello si occupa di **indirizzare, suddividere e instradare** i pacchetti sulla rete. Il protocollo **IP** lavora in questo layer offrendo un metodo di **indirizzamento** ed un tipo di trasmissione **connectionless**;

• **Network** (*Rete*): rappresenta i layer **Data Link** e **Physical** del modello OSI, lavora a stretto contatto con l'hardware. Questo livello si occupa di **prelevare ed immettere** i frame dati, diversi a seconda della tipologia di rete, sul cavo di rete e di controllarne la correttezza tramite un algoritmo CRC (*cyclic redundancy check*). Rispetto al modello ISO/OSI non ci sono distinzioni tra schede di rete e driver con il vantaggio di poter implementare TCP/IP su ogni tipo di rete;

Un modello organizzato in questo modo presenta diversi vantaggi, infatti un nuovo protocollo in uno qualsiasi dei layer **TCP/IP** deve interagire **solamente** con quello **adiacente** a livello superiore od inferiore, semplificando e conseguentemente riducendo la possibilità di errori nello sviluppo e nella gestione.

[Il protocollo IP](#)

Il protocollo IP si occupa di fornire un metodo di indirizzamento logico e di gestione frammentazione/riasseblaggio per la trasmissione dati tra gli host di una rete.

Il protocollo **IP**, descritto sopra, nasce negli anni '70 grazie a una serie di ricerche fatte dalle università americane su richiesta del ministro della difesa, allo scopo di realizzare una rete in grado di trasportare diversi tipi di informazioni. L'IP protocol definisce una tecnica di trasmissione dati **non orientata alla connessione** (*connectionless*) e **senza riscontro** (non c'è garanzia che i pacchetti giungano a destinazione e nella sequenza corretta). Esso prevede che le informazioni vengano strutturate in unità chiamate **datagrammi IP** (IP datagram), di lunghezza massima 65535 byte, suddivise in due aree: il **campo dati** (*data*) che contiene il messaggio da inviare e l'**intestazione** (*header*) che contiene le informazioni necessarie per instradare il pacchetto.

HEADER IP

L'intestazione di un pacchetto IP è suddivisa nei seguenti campi: **Version**: campo di 4 bit, descrive la versione del protocollo;

IHL: campo di 4 bit, definisce la lunghezza dell'intestazione. Il campo **IHL** (*Internet Header Length*) viene sottratto da **Packet Length** per determinare l'inizio dei dati;

Service Type: campo di 8 bit, che definisce le caratteristiche del servizio in **Precedence** (importanza del pacchetto),

Delay, Throughput, Reliability (informazioni per il trasporto, servono a determinare il percorso che il pacchetto farà);

Packet Length: campo di 16 bit, definisce la lunghezza totale del pacchetto (Intestazione+Dati);

Packet Fragmentation: è formato da tre campi, **identification** di 16 bit, **flags** di 3 bit e **fragment offset** di 13 bit i quali permettono di suddividere il pacchetto in unità più piccole quando esso deve passare attraverso una rete che prevede frame di dimensioni minori, e di essere ricostruito all'uscita dalla rete;

Time-to-live (TTL): campo di 8 bit, definisce il tempo massimo di permanenza del pacchetto nella rete, ad ogni hop (*router attraversato*) il suo valore diminuisce di uno, una volta raggiunto il valore zero il pacchetto viene scartato;

Protocol: campo di 8 bit, definisce il protocollo ad alto livello utilizzato per creare il messaggio contenuto nel campo dati (TCP, UDP, ICMP ecc); **Header Checksum:** campo di 16 bit, definisce un checksum per il controllo della correttezza dei dati contenuti nell'intestazione del pacchetto; **Ip Address Source:** campo di 32 bit, contiene l'indirizzo IP del mittente;

Ip Address Destination: campo di 32 bit, contiene l'indirizzo IP del destinatario;

Options: campo di dimensioni variabili, è opzionale, contiene informazioni sulle operazioni che devono essere effettuate durante il percorso;

Padding: campo di dimensioni variabili, è utilizzato per far raggiungere all'area d'intestazione una dimensione di 32 bit o un suo multiplo;

Data: i dati trasportati dal protocollo.

[Indirizzi IP, classi e Subnetting](#)

L'indirizzamento IP permette di identificare ogni host all'interno di una rete TCP/IP. Grazie all'utilizzo delle classi di indirizzi ed al subnetting è possibile organizzare e gestire in modo più efficiente il proprio network.

Un indirizzo IP, chiamato anche indirizzo **logico**, rappresenta un identificativo software per le interfacce di rete, esso viene utilizzato in combinazione con l'indirizzo fisico (*MAC*), il quale consente di determinare in modo **univoco** ogni interfaccia di un dispositivo di rete. Un IP Address è un numero di **32 bit** suddiviso in quattro gruppi da **8 bit** ciascuno, la forma con la quale viene solitamente rappresentato è detta **decimale puntata** (*Dotted Decimal*).

Essendo ogni numero rappresentato da **8 bit**, può assumere un range di valori da **0** a **255**. Utilizzando **32 bit** per indirizzo è possibile avere **4.294.967.296** combinazioni di indirizzi differenti. In realtà esistono alcuni indirizzi particolari, di conseguenza non tutti i valori sono disponibili al fine di identificare un host nella rete.

Un esempio di Ip Address:

Rete.	Rete.	Rete.	Host	
192.	168.	5.	2	: Rappresentazione decimale
11000000.	10101000.	00000101.	000000010	: Rappresentazione binaria

Questo è un esempio di indirizzo (*Classe C*) in cui 192.168.5 identifica la rete di appartenenza dell'host 2.

INDIRIZZI SPECIALI: NETWORK, BROADCAST E LOOPBACK

Esistono alcuni particolari indirizzi di rete che non possono essere assegnati per l'identificazione di un host, tra questi abbiamo: **network** e **broadcast** e **loopback**:

- **Network**: quando i bit dell'ottetto che rappresenta l'host hanno tutti valore **0**, l'indirizzo è detto di **rete** o **Network Address**: 192.168.5.0 oppure in binario 11000000.10101000.00000101.00000000;
- **Broadcast**: quando i bit del numero che rappresenta l'host hanno tutti valore **1**, l'indirizzo è detto di **broadcast** o **broadcast address**, e rappresenta tutti gli host di quella rete. Inviare un pacchetto all'indirizzo 192.168.5.255 o in forma binaria 11000000.10101000.00000101.11111111 equivale a mandare un pacchetto a tutti gli host della rete 192.168.5.0;
- **Broadcast di rete**: abbiamo questo tipo di indirizzo quando tutti i bit, sia della parte relativa all'host sia della parte relativa alla rete hanno valore 1. Inviare un pacchetto a **255.255.255.255** o in binario **11111111.11111111.11111111.11111111** significa inoltrarlo verso tutti gli host della rete corrente;
- **Loopback**: è utilizzato per funzioni di test del protocollo TCP/IP, non genera traffico di rete e corrisponde all'indirizzo **127.0.0.1**;

CLASSI DI INDIRIZZI

- Per permettere una migliore organizzazione della rete, gli indirizzi disponibili sono stati suddivisi in classi in base alle **dimensioni** del network da gestire. In questo modo verranno utilizzate le classi più adatte ad alla dimensioni della rete, con conseguente minore spreco di IP address. Sono disponibili **cinque** classi di indirizzi IP, di cui solo le prime tre possono essere utilizzate per assegnare indirizzi agli host.

Indirizzi di classe A

Il valore del primo ottetto è compreso tra **1** e **126** (I primi otto bit di questo indirizzo saranno: **0*******). E' rappresentata da indirizzi di tipo: **Rete.Host.Host.Host** ovvero 8 bit per la identificare la rete (di cui il primo fisso) e 24 per identificare gli host. Permette di ottenere **126** reti formate da **16.774.214** host ciascuna.

Indirizzi di classe B

Il valore del primo ottetto è compreso tra **128** e **191** (I primi otto bit di questo indirizzo saranno: **10*******). E' rappresentata da indirizzi di tipo: **Rete.Rete.Host.Host** ovvero 16 bit per la identificare la rete(di cui i primi due fissi) e 16 per identificare gli host. E' possibile ottenere **16.384** reti formate da **65.534** host ciascuna.

Indirizzi di classe C

Il valore del primo ottetto è compreso tra **192** e **223** (I primi otto bit di questo indirizzo saranno: **110******). E' rappresentata da indirizzi di tipo: **Rete.Rete.Rete.Host** ovvero 24 bit per la identificare la rete (di cui i primi tre fissi) e 8 per identificare gli host. E' possibile ottenere **2.097.152** reti con **254** host ciascuna. La classe di tipo C è quella dunque usata nella rete Moretto.

Indirizzi di classe D

Il valore del primo ottetto è compreso tra **224** e **239** (I primi otto bit di questo indirizzo saranno: **1110******). Sono indirizzi di rete riservati ai gruppi multicast e non assegnabili ai singoli host.

Indirizzi di classe E

Il valore del primo ottetto è compreso tra **240** e **255** (I primi otto bit di questo indirizzo saranno: **1111******). Sono indirizzi riservati per usi futuri.

LE SUBNET MASK

Per il corretto funzionamento di una rete, ogni host deve poter distinguere quale parte dell'indirizzo identifica l'host e quale la rete. Questo può avvenire grazie all'ausilio delle subnet mask (*Maschere di sottorete*). Per quanto riguarda le classi A B C standard, cioè non ulteriormente suddivise, esistono delle subnet di default:

- Classe A: **Rete**.Host.Host.Host ha come subnet **255.0.0.0**;
- Classe B: **Rete.Rete**.Host.Host ha come subnet **255.255.0.0**
- Classe C: **Rete.Rete.Rete**.Host ha come subnet **255.255.255.0**

Il processo di messa in AND Per determinare se il destinatario dei propri pacchetti si trova sulla propria sottorete ogni host utilizza la propria maschera di sottorete durante un processo chiamato **di messa in AND** (*ANDing process*). Questo processo consiste nel confrontare il risultato dell'operazione di **AND** (*matematica booleana*) bit a bit tra il proprio indirizzo e la propria maschera subnet mask con quello tra l'indirizzo del destinatario e la propria subnet mask.

Avendo un Host A con IP **192.168.0.5** con subnet **255.255.255.0** che vuole inviare dei pacchetti ad un Host B **192.168.0.25** con subnet **255.255.255.0**, esso deve determinare se B è sulla stessa sua sottorete:

Host A: 192.168.0.5

Ip address Host A	1000000.10101000.00000000.000000101 :	192.168.0.5
Subnet mask Host A	11111111.11111111.11111111.00000000 :	Subnet
Risultato AND bit a bit	11000000.10101000.00000101.00000000 :	

Host B: 192.168.0.25

Ip address Host B 11000000.10101000. 00000000.000011001 : **192.168.0.25**

Subnet mask Host B 11111111.11111111.11111111.00000000 :

Risultato AND bit a bit 11000000.10101000.00000101.00000000 :

Il risultato è identico, quindi, i due host possono inviarsi **direttamente** i pacchetti in quanto sulla stessa sottorete. Qualora il processo di AND avesse evidenziato valori diversi, i due host **non** avrebbero potuto comunicare direttamente, ma sarebbe stato necessario un router tra di essi.

NOTAZIONI

Esistono **due principali notazioni** attraverso le quali è possibile indicare un indirizzo IP:

- **Indicando espressamente la subnet mask:**

49.22.5.3	255.0.0.0	Classe	A;
172.16.20.5	255.255.0.0	Classe	B;
192.168.15.4	255.255.255.0	Classe	C;

- **Indicando il numero di bit a "1" che compongono la subnet mask:**

49.22.5.3/8	Classe	A;
172.16.20.5/16	Classe	B;
192.168.15.4/24	Classe	C;

SUBNETTING

L'utilizzo della classe di rete corrispondente alle dimensioni che più si avvicinano a quella che si vuole gestire a volte non è sufficiente. Può essere necessario, dover suddividere la rete in ulteriori sottoreti. Per fare questo è possibile utilizzare la tecnica del **subnetting**. Il subnetting di una rete comporta diversi vantaggi:

Minor spreco di indirizzi: in quanto è possibile scegliere il numero di host che faranno parte della sottorete;

Riduzione del traffico di rete: in quanto si riduce il dominio di broadcast (*broadcast domain*);

Miglioramento delle performance della rete: in conseguenza della riduzione del traffico;

Il subnetting consiste nell'utilizzare alcuni bit "presi in prestito" (*borrowed*) dalla parte host dell'indirizzo di rete. E' possibile procedere alla suddivisione della rete in sottoreti più piccole tramite lo scheda seguente:

Determinare il numero di sottoreti necessarie.

E' necessario tenere presente che il numero di subnet che si possono creare è dato da $2^{(x-2)}$ dove x è rappresentato dai bit presi in prestito dalla parte host dell'indirizzo ai quali naturalmente bisogna levare l'indirizzo di broadcast quello di rete non assegnabili.

Esempio: Prendendo in prestito 4 bit, sarà possibile creare 14 sottoreti;

Determinare il numero di host per ogni sottorete

Questo valore è dato da $2^{(y-2)}$ dove y è il numero di bit rimasti per la rappresentazione degli host;

Esempio: se i bit rimanenti sono 6 si potranno avere sottoreti formate da 62 host l'una;

Determinare le subnet valide

Questo valore è dato da **256-z**, dove 256 dove z rappresenta il valore della subnetmask. Esempio: con una subnetmask di valore 224 avremmo avuto $256-224=32$. Questo valore è il valore della prima subnet valida ed è anche la base per le successive, la cui progressione sarà: 32, 64, 96, 128, 160, 192;

Determinare gli host validi.

Sono rappresentati da tutti i valori compresi tra le subnet create togliendo gli indirizzi di broadcast e network;

Determinare degli indirizzi di broadcast e network delle subnet.

Sono gli indirizzi in cui rispettivamente i bit della parte host sono settati a 1 (broadcast) e a 0 (network);

Delphi (in inglese)

1.1.10. Da Wikipedia, l'enciclopedia libera.

Delphi is a programming language that a development atmosphere. It has been created from the Borland (than for a sure period it has changed its name in Inprise). The Delphi language, previously known like Object Pascal (a version of oriented Pascal to the objects), has been developed initially for Microsoft Windows, of recent they have been published versions for GNU/Linux (Kylix) and framework the .NET of Microsoft (like will be possible to see in the prosiegua of this article).

It is a lot used for the development of applications desktop and applications enterprise that they use database, but being an instrument of development of generic character can be used for many types of plans. It has been one of first to being known like instrument RAD, Rapid Application Development, when it was realized in 1995: it was the first atmosphere of completely visual development, and its compiler extremely fast if was compared to those existing to the age. Delphi 2, realized the successive year, allowed the development of applications for Windows atmospheres to 32-bit, ability that was introduced little years after also in the atmosphere of development for C++, C++Builder (always realized from the Borland). In 2001 version for Linux known like Kylix has been released one (Greek vase). One new version of this atmosphere has been released every year until reaching the version Delphi 7 Study in 2002, the used language is hour known like Delphi instead of Object Pascal and has been added to the support for Linux (through Kylix) and .NET (with one preliminary version of the compiler). For platform .NET Borland release C# Builder that allowed I use it of the C# language and framework the .NET. The full support for .NET came later on introduced with Delphi 8, one version of Delphi oriented exclusively to the development for platform .NET, omitting therefore that one for Win32. Subsequently to Delphi 8 they have been released Delphi 2005 (that it has finally re-united in an only atmosphere of development the support to Delphi for Win32, Delphi for .NET and C#) and Borland Developer Study 2006, that has added moreover the support to C/C++.

The main members of Delphi and Kylix are the language, Delphi, the VCL/CLX (Visual Component Library), and an easy and sturdy connettività to the database, arranges with a powerful IDE (Integrated Development Environment) and other instruments to you of support.

The worthy characteristics of famous of the Delphi language include:

- The transparent manipulation of the objects through references/gunlayers;
- The property like parts of the language, that they allow to the reading and the settaggio in transparent way through the variable access to the member.

Maninder Bansal Vital Ivo 5BZ

- Property of type “index” and the property of default that they allow an access to the collections of objects in simple and transparent way.
- Delegates, that is gunlayers to methods with control of the type, that they come used in order to connect the events notified from the members.
- Delegation of the implementation of interface to one variable member or one property of the class.
- Easy implementation of managers for the messages of Windows through tie between a method of the class with the number or name of the Windows message to manage.

Many of the functionalities previously listed have been before introduced from Delphi and later on adopted also in other languages. The head architect of the Delphi plan, and its predecessore Turbo Pascal, has been Anders Hejlsberg until the moment of its passage to Microsoft in 1996.

The Delphi product comes distributed in several configurations, each of which offers greater functionalities regarding the smaller versions:

- Personal
- Professional
- Enterprise
- Architect

Good reasons in order to use Delphi:

- One community of developers much informative one and great aid, with excellent relationship noise/information on the newsgroup news://forums.borland.com or http://info.borland.com/newsgroups/ng_delphi.html
- It can compile in a single executable file, becoming simpler the distribution and reducing the problems of various versions of the DLL.
- the VCL and the members of other producers are usually available with the source code
- Powerful, fast Compiler and optimizing native coded Compilation for various platforms from the same code source
- Support for the most modern technologies and standard

Program of example

[Hello, world!](#)

The following example prints the text “Hello world”.

```
program HelloWorld;
{$APPTYPE CONSOLE}
```

```
begin
  writeln('Hello World');
```

Maninder Bansal Vital Ivo 5BZ

end.

this code prints the word in one type window consul. In order to visualize the word Hello World inside of a text box instead it is necessary to use the following code. Taken one text box Edit1:

```
program HelloWorld;  
{ $APPTYPE CONSOLE }  
  
begin  
  Edit1.text:='Hello World';  
end.
```

Programmi realizzati

Settaggio dell' orario:

Il seguente programma serve per Impostare l'orara in cui ogni computer dovrà restare spento o accesso. Il programma va a compilare il Database nominato "NETMORETTO" in cui sono inseriti e descritti tutti computer appartenente alla sottorete dell' 'Istituto Moretto, nel database oltre ai dati personali del computer sono inseriti in fondo alla descrizione i giorni della settimana che vanno compilati con il seguente programma. Ogni giorno o stringa e costituito da 24 caratteri (corrispondenti alle 24 ore) dove il nostro programma andrà a compilare con 1 che sta a significare accesso e con 0 che sta a significare spento. Prima di compilare il database il programma andrà a controllare se il computer che abbiamo scelto non sia un Server , se qualora fosse un server allora il programma andrà automaticamente a settare l'orario mettendo tutti a 1 perché essendo un server e sempre accesso.

Con il seguente programma realizziamo le seguenti funzioni:

- Decidiamo quale computer deve essere settato mettendo valori tra 1-254 in una Edit (un campo dove si possono inserire valori);
 - Decidiamo quale sottorete si vuole usare per settare il computer giusto attraverso ItemIndex che ci permette di scegliere tra la rete Didattica cioè la 145 e la amministrativa la 146 ;
 - Impostiamo il giorno in cui deve essere regolato l'orario giusto sempre attraverso ItemIndex ;
 - Settiamo l'orario attraverso appositi Box che ci permettono di selezionare o deselectare l'orario di ogni computer in cui essi devono funzionare infatti l'orario e composta dal mattino pomeriggio e serale selezionando così con i rispettivi orari giornalieri;
- Nel inserimento dei valori attraverso un apposita Edit, si deve ricordare che i valori devono essere compresi tra 1 e 254 perché lo 0 e il 255 sono ip Riservati che devono essere pingati.

Orari di Attività dei PC in Rete - IPSIA MORETTO BRESCIA

145 - DIDATTICA
 146 - AMMINISTRATIVA

1

ID	IP	MAC	HOST	LUOGO	INVE
1	192.168.145.0		RWS145-0		
2	192.168.145.1		RWS145-1		
3	192.168.145.2		RWS145-2		
4	192.168.145.3		RWS145-3		
5	192.168.145.4		RWS145-4		

LUNEDI'
 MARTEDI'
 MERCOLEDI'
 GIOVEDI'
 VENARDI'
 SABATO
 DOMENICA

START

	MATTINO	POMERIGGIO	SERALE		
<input checked="" type="checkbox"/>	08-09	<input type="checkbox"/>	14-15	<input checked="" type="checkbox"/>	18-19
<input checked="" type="checkbox"/>	09-10	<input type="checkbox"/>	15-16	<input checked="" type="checkbox"/>	19-20
<input checked="" type="checkbox"/>	10-11	<input type="checkbox"/>	16-17	<input checked="" type="checkbox"/>	20-21
<input checked="" type="checkbox"/>	11-12	<input type="checkbox"/>	17-18	<input checked="" type="checkbox"/>	21-22
<input checked="" type="checkbox"/>	12-13	<input type="button" value="ON"/>		<input checked="" type="checkbox"/>	22-23
<input checked="" type="checkbox"/>	13-14	<input type="button" value="OFF"/>		<input type="button" value="ON"/>	
	<input type="button" value="ON"/>			<input type="button" value="OFF"/>	
	<input type="button" value="OFF"/>				

La parte pratica:

```

Procedure TForm1.Button1Click(Sender: TObject);
VAR ST, ST1, st2, ST3 : STRING;
    NIP : integer;
Begin
  NIP := StrToInt(Edit2.Text);
  edit2.Text := IntToStr(NIP);
  If (NIP < 1) OR (NIP > 254) // decidiamo che il numero deve essere tra 1 e 254
  Then
    begin
      Showmessage('Inserisci valori compresi fra 1 e 254'); // viene fuori il messaggio tra le parentesi
      Exit;
    end;
  st1 := '192.168.';
  IF RG2.ItemIndex = 0 // vede che tipo di sottorete e sata scelta
  THEN ST1 := ST1+'145.'+ EDIT2.Text
  ELSE ST1 := ST1+'146.'+ EDIT2.Text;

  st2 := 'SELECT * FROM NETMORETTO '+
    'WHERE IP="'+ST1+'";
  
```



```
QueryRete.Post; // scrive su disco  
end;
```

```
end. // chiude tutta la procedure che abbiamo compilato
```

Test ping e ed invio messaggio al amministratore per eventuali errori

Il seguente programma si appoggia come il primo all database "NETMORETTO" in cui abbiamo già prestabilito l'ora in cui i computer devono funzionare grazie al programma precedente.

Con questo programma noi controlliamo i computer, che sono in rete, se sono spenti o sono accessi. Il primo controllo del programma viene effettuato sull' "TPING" del nostro database che significa Test ping. Nella stringa se e presente un 1 allora il test ping deve essere effettuato su quel IP se invece e presente uno 0 allora il test non deve essere effettuato, nel nostro caso sono solo quattro computer che non hanno bisogno di un controllo cioè quelli con IP finale 0 e 255.

Con il seguente programma abbiamo la possibilità di effettuare il test scegliendo i tipi di computer che abbiamo intenzione di interrogare, anche in questo caso il programma va ad controllare il database in cui e descritto che tipo di computer sono.

Nel nostro caso ci sono solo quattro tipi di computer:

```
S =server ;  
R = Riservato;  
F = Free (libero);  
A = Assegnato;
```

Nel programma viene aggiunto oltre a queste quattro scelte una quinta che permette di andare ad interrogare tutti gli IP.

Dopo aver effettuato il Ping il programma va a scrivere il risultato nel Database cioè nella stringa in cui è inserito il nome Rping .

Oltre a Pingare il programma va a controllare i vari errori che sono,.

- Va controllare che la stringa di 24 caratteri dei giorni sia compilata e che ci siano effettivamente 24 caratteri giusti;
- Se la stringa di 24 caratteri e compilata allora va ad controllare l'orario, cioè come già detto in precedenza la seguente stringa e stat compilata con dei 0 e con dei 1 dove 1 vuol dire accesso e 0 spento, se il computer IP in questione risulta accesso ed invece deve essere acceso allora risulterebbe un errore;

Tutti i seguenti errori vanno ad compilare un LOG file(cioè un blocco note) in cui sarà inserita la data in cui e iniziato il ingaggio e la data in cui si e concluso. Nel seguente LOG file verranno inseriti tutti gli errori che il programma ha rivelato per ogni IP effettivamente se per un IP non rivela nessun errore allora il programma non scriverà niente per quel IP.

Il seguente LOG file che il programma crea verrà ogni volta creato e il file con gli errori precedenti verrà cancellato automaticamente. Il LOG file che il programma crea siccome deve essere spedito via E-Mail viene prima creato con il nome LOG.text e poi viene cambiato il nome con Errori.txt, poiché il programma di posta fa attivare ogni ora il seguente programma va controllare se esiste il seguente file e solo quando esso c'è il programma spedirà il file.

Il nostro si basa fondamentalmente di controllare che i PC non vengano lasciati accessi nel momento in cui devono essere spento perché molte volte succede che alla sera il computer venga lasciato accesso per errore. Allora il nostro programma entra in azione avvisando l'amministratore con il nostro file in cui saranno inseriti gli errori.

```
procedure TForm1.Button1Click(Sender: TObject);  
  VAR Cont1 : integer;
```

```
begin
```

```
Tab1.First;
```

```
While Tab1.FieldName('IP').AsString <> '192.168.146.0'  
do Tab1.Next;
```

```
Timer1.Enabled := True;  
end;
```

```
Function TForm1.Giorno(gg : integer): string;  
Begin  
Case gg of  
1 : result := 'DOM';  
2 : result := 'LUN';  
3 : result := 'MAR';  
4 : result := 'MER';  
5 : result := 'GIO';  
6 : result := 'VEN';  
7 : result := 'SAB';  
end;  
End;
```

```
procedure TForm1.FormCreate(Sender: TObject);  
var Hour, Min, Sec, MSec: Word;  
ADate: TDateTime;  
begin  
DeleteFile('errori.txt'); // elimino il file di log  
STATO := 0;  
DecodeTime(Now, Hour, Min, Sec, MSec);  
Ora_Attuale := Hour;  
Giorno_sett := DayOfWeek(Now);  
logfile1.LogToFile(TimeToStr(Time)+' APERTURA LOGFILE');  
end;
```

```
procedure TForm1.Timer1Timer(Sender: TObject);  
VAR TSPing : Integer;  
TIPO : STRING;  
begin  
Case Stato of  
0 : Begin  
// Preparazione  
Inc(Stato);  
End;  
1 : Begin  
Ping1.Address := TAB1.FIELDBYNAME('IP').ASString;  
TSPING := TAB1.FIELDBYNAME('TPING').ASInteger;  
TIPO := TAB1.FIELDBYNAME('TYPE').ASString;  
IF TSPING = 1  
then
```

```

Begin
  If Copy(RG.Items.Strings[RG.ItemIndex],1,1) <> 'T'
  Then
    Begin
      If TRIM(TIPO) = Copy(RG.Items.Strings[RG.ItemIndex],1,1)
      Then Ping1.Ping
      Else
        Begin
          Tab1.Next;
          If Tab1.Eof Then Inc(Stato)
          Else Stato := 1;
        End;
      End
    End
  Else
    Begin
      Ping1.Ping;
    End;
  End
Else
  Begin
    Tab1.Next;
    If Tab1.Eof Then Inc(Stato)
    Else Stato := 1;
  End;
End;
2 : Begin
  Close;
End;
4 : Begin
  End;
End;
end;

```

```

procedure TForm1.Ping1EchoReply(Sender, Icmp: TObject; Status: Integer);
  VAR N : INTEGER;
      ST : STRING;
begin
  Tab1.Edit;
  If Status = 0 Then N := 0
  Else N := 1;
  Tab1.FieldName('RPING').AsInteger := N;

  ST := Tab1.FieldName(GIORNO(Giorno_sett)).AsString;
  If length(st) <> 24 Then
    Begin
      // Timer1.Enabled := False;
      logfile1.LogToFile(TimeToStr(Time)+' ERRORE 24 ch '+
TAB1.FIELDBYNAME('IP').ASString+'-'+
Maninder Bansal    Vital Ivo    5BZ

```

```
TAB1.FIELDBYNAME('LUOGO').ASString);
```

```
End
```

```
Else
```

```
// Se il PC deve risultare spento mentre in realtà dal Test viene riscontrato Accesso
```

```
// si procede a registrare la condizione di errore nel logfile.
```

```
If (st[Ora_Attuale] = '0') AND (N = 1)
```

```
Then logfile1.LogToFile(TimeToStr(Time)+' ERRORE: '+  
TAB1.FIELDBYNAME('IP').ASString+'-'+
```

```
TAB1.FIELDBYNAME('LUOGO').ASString);
```

```
Tab1.Post;
```

```
Tab1.Next;
```

```
Stato := 1;
```

```
end;
```

```
procedure TForm1.Ping1EchoRequest(Sender, Icmp: TObject);
```

```
begin
```

```
(* Memo1.Lines.add('Sending ' + IntToStr(Ping1.Size) + ' bytes to ' +  
Ping1.HostName + ' (' + Ping1.HostIP + ') '+ 'All');
```

```
*)
```

```
end;
```

```
procedure TForm1.Timer2Timer(Sender: TObject);
```

```
begin
```

```
Timer2.Enabled := False;
```

```
button1.Click;
```

```
end;
```

```
procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
```

```
begin
```

```
logfile1.LogToFile(TimeToStr(Time)+' CHIUSURA LOGFILE');
```

```
RenameFile('log.txt','errori.txt');
```

```
end;
```

Posta elettronica o E-Mail

Il nostro seguente programma è stato realizzato grazie ad un'altro programma che è stato modificato per adattarlo meglio alle nostre esigenze. Il programma modificato va ad eseguire il precedente programma ed andrà a controllare se il programma ha creato il file errori siccome il file errori esiste già il precedente programma va ad cancellarlo immediatamente andrà poi ad compilare un'altro file in cui saranno inseriti gli errori che poi al file verrà modificato il nome del file(errori.txt). Il nostro programma di posta rivelerà il seguente file e perciò andrà avanti spedendo il file all'amministratore.

Dopo aver finito di pingare il programma che era stato aperto viene automaticamente chiuso creando il file che ci serviva.

Come si può notare i due programmi hanno bisogno uno dell'altro per funzionare anche perché sono tutti e due creati in un ambiente Delphi perciò facilmente modificabili.

```
var
```

```
Form1: TForm1;
```

```
status : integer;
```

```

const
  host_smtp = 'mail.tin.it';
  host_port = 25;
  host_name = 'TEST-RETE';
  email_addr = 'manutenzione@ipsiamoretto.it';
implementation

{$R *.DFM}

procedure TForm1.Button1Click(Sender: TObject);
begin
  NMSMTP1.Host := host_smtp;
  NMSMTP1.Port := host_port;
  // NMSMTP1.UserID := Edit4.Text;
  NMSMTP1.Connect;
end;

(*
procedure TForm1.Button3Click(Sender: TObject);
// var TmpStr: String;
begin
  TmpStr := Edit3.Text; // Do this so the user can't change the edit box.
  If NMSMTP1.Verify(TmpStr) then
    ShowMessage(TmpStr+' verified')
  else
    ShowMessage(TmpStr+' not verified');

end;
*)

procedure TForm1.Button2Click(Sender: TObject);
begin
  NMSMTP1.Disconnect;
end;

procedure TForm1.NMSMTP1Connect(Sender: TObject);
begin
  StatusBar1.SimpleText := 'Connected';
  Inc(Status);
end;

procedure TForm1.NMSMTP1Disconnect(Sender: TObject);
begin
  If StatusBar1 <> nil then
    StatusBar1.SimpleText := 'Disconnected';
  Inc(Status);
end;

```

```

procedure TForm1.NMSMTP1Status(Sender: TComponent; Status: String);
begin
  If StatusBar1 <> nil then
    StatusBar1.SimpleText := status;
end;

```

```

procedure TForm1.Button6Click(Sender: TObject);
begin
  // dati del mittente
  NMSMTP1.PostMessage.FromAddress := email_addr;
  NMSMTP1.PostMessage.FromName := host_name;

  NMSMTP1.PostMessage.Subject := Edit10.Text;
  NMSMTP1.PostMessage.ToAddress.Add(Edit7.Text);
  // NMSMTP1.PostMessage.ToBlindCarbonCopy.Add(Edit9.Text);
  // NMSMTP1.PostMessage.ToCarbonCopy.Add(Edit8.Text);
  // NMSMTP1.PostMessage.Attachments.AddStrings(Listbox1.Items);
  NMSMTP1.PostMessage.Body.Assign(Memo1.Lines);
  NMSMTP1.SendMail;
end;

```

```

procedure TForm1.NMSMTP1EncodeStart(Filename: String);
begin
  StatusBar1.SimpleText := 'Encoding '+Filename;
end;

```

```

procedure TForm1.NMSMTP1EncodeEnd(Filename: String);
begin
  StatusBar1.SimpleText := 'Finished encoding '+Filename;
end;

```

```

procedure TForm1.NMSMTP1MailListReturn(MailAddress: String);
begin
  // Memo2.Lines.Add(MailAddress);
end;

```

```

procedure TForm1.NMSMTP1ConnectionFailed(Sender: TObject);
begin
  ShowMessage('Connection Failed');
end;

```

```

procedure TForm1.NMSMTP1ConnectionRequired(var handled: Boolean);
begin
  If MessageDlg('Connection Required. Connect?', mtConfirmation, mbOkCancel, 0) = mrOk then
    Begin
      Handled := TRUE;
      NMSMTP1.Connect;
    End;
end;

```

end;

```
procedure TForm1.NMSMTP1Failure(Sender: TObject);
begin
  StatusBar1.SimpleText := 'Failure';
end;
```

```
procedure TForm1.NMSMTP1HostResolved(Sender: TComponent);
begin
  StatusBar1.SimpleText := 'Host Resolved';
end;
```

```
procedure TForm1.NMSMTP1InvalidHost(var handled: Boolean);
var
  TmpStr: String;
begin
  If InputQuery('Invalid Host!', 'Specify a new host:', TmpStr) then
  Begin
    NMSMTP1.Host := TmpStr;
    Handled := TRUE;
  End;
end;
```

```
procedure TForm1.NMSMTP1PacketSent(Sender: TObject);
begin
  StatusBar1.SimpleText := IntToStr(NMSMTP1.BytesSent)+' bytes of
'+IntToStr(NMSMTP1.BytesTotal)+' sent';
end;
```

```
procedure TForm1.NMSMTP1RecipientNotFound(Recipient: String);
begin
  ShowMessage('Recipient "'+Recipient+'" not found');
end;
```

```
procedure TForm1.NMSMTP1SendStart(Sender: TObject);
begin
  StatusBar1.SimpleText := 'Sending message';
end;
```

```
procedure TForm1.NMSMTP1Success(Sender: TObject);
begin
  StatusBar1.SimpleText := 'Success';
  Inc(Status);
end;
```

```
procedure TForm1.NMSMTP1HeaderIncomplete(var handled: Boolean;
  hiType: Integer);
begin
```

```
ShowMessage('Header Incomplete.');
```

```
end;
```

```
procedure TForm1.FormCloseQuery(Sender: TObject; var CanClose: Boolean);  
begin  
  NMSMTP1.Abort;  
end;
```

```
procedure TForm1.Button3Click(Sender: TObject);  
  var email, body : string;  
      k : integer;  
begin  
  Tab_utenti.Open;  
  edit10.Text := 'SEGNALAZIONE ERRORI';  
  email := Tab_utenti.FieldName('EMAIL').AsString;  
  edit7.Text := email;  
  Memo1.Lines.LoadFromFile('errori.txt');  
  body := Memo1.Lines.Text;  
  Memo1.Lines.Text := body;  
  Button6.Click;
```

```
end;
```

```
procedure TForm1.FormCreate(Sender: TObject);  
begin  
  status := 0;  
end;
```

```
procedure TForm1.Timer1Timer(Sender: TObject);  
begin  
  case status of  
    0 : Begin  
  //  DeleteFile('log.txt'); // elimino il file di log  
    ShellExecute(HANDLE,'open',PChar('pTest_Rete.exe'),nil,nil,SW_SHOWNORMAL);  
    Inc(Status);  
    End;  
    1 : Begin  
    If FileExists('errori.txt') Then Inc(Status);  
    End;  
    2 : Begin  
    Button1.Click; // attiva la connessione  
    Inc(Status);  
    End;  
    4: Begin
```

```
        Button3.Click;
        Inc(Status);
    End;
6: Begin
    Button2.Click;
    Inc(Status);
    End;
8: Close;
end;
end;

procedure TForm1.FormActivate(Sender: TObject);
begin
    Timer1.Enabled := True;
end;

end.
```