a cura di Andrea de Prisco

Il controllo della concorrenza

di Anna Pugliese

terza puntata

Non so se avete tutti presente il turnomatic. Ma sì, quella macchinetta distributrice di numerini che sta al banco dei supermercati. Bene, quella macchinetta serve per implementare un meccanismo di controllo della concorrenza

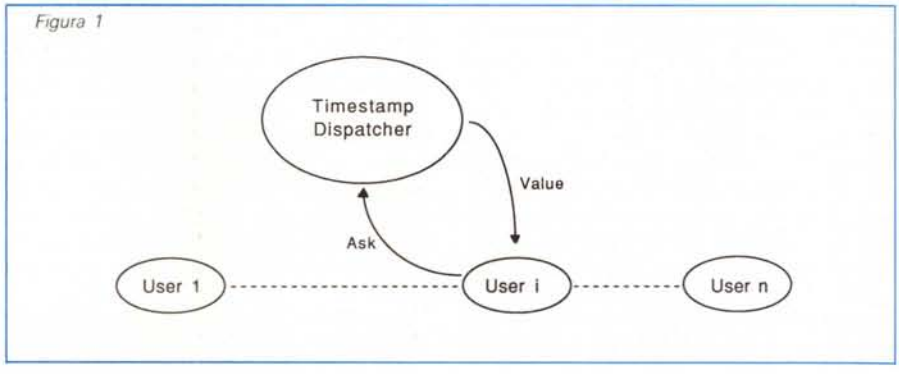
Mi spiego

Vorrei precisare anzitutto che la cosa non mi è venuta in mente durante l'attesa al banco, essendo impossibile pensare ad altro davanti a quel piccolo Eden di salumi, formaggi e affini bensì ripensandoci su a stomaco pie... pardon!, a mente fredda.

In cosa consiste il meccanismo è presto detto: il tutto comincia quando avvertite un'incredibile voglia di mascarpone fresco, vi precipitate al banco del più vicino supermercato alimentari e prelevate il vostro tagliandino dal turnomatic di cui sopra (mi raccomando: uno solo e non tre o quattro come fate di solito). A questo punto la vostra parte è finita, dovete solo aspettare che venga chiamato il vostro numero, la qual cosa verrà implementata dall'addetto al banco seguendo una politica molto particolare, in base alla quale se, per esempio, il cliente appena servito aveva il numero 67, il successivo numero da chiamare sarà il 67+1=68.

So perfettamente che la cosa in sé non fa ridere per niente; l'esempio è stato riportato al solo scopo di evidenziare come un meccanismo utilizzato nel vivere quotidiano possa essere applicato all'informatica, mantenendone la struttura ma complicandone in maniera incredibile l'esecuzione. È il caso dell'algoritmo di controllo della concorrenza che sarà presentato in questo numero della rivista, la cui utilizzazione sui sistemi di elaborazione è seconda solo al locking a due fasi visto il mese scorso.

Figura 1



Il Timestamp

L'idea che sta alla base del turnomatic dei supermercati è utilizzata in vari ambiti. Essa è diffusa già da tempo nel campo dell'informatica, dove ha preso il nome di Timestamp.

Un timestamp, che letteralmente significa marca temporale, è definito come un identificatore UNICO e GLOBALE generato da una sequenza monotóna crescente; esso è uno strumento utilizzato per implementare politiche di scheduling di attività concorrenti, ed in particolare, per quanto ci riguarda, esso permette di implementare meccanismi di controllo di consistenza sull'esecuzione di transazioni concorrenti. Vedremo nel seguito come, associando un timestamp ad ogni transazione, sarà possibile mantenere la consistenza del sistema, purché, si intende, l'uso delle risorse avvenga in accordo alla politica che vedremo e che prende il nome di Timestamping.

Concentriamoci per il momento sul problema della generazione dei timestamp.

La definizione parla di seguenza monotóna crescente e di unicità e globalità dei timestamp. È evidente che la cosa in pratica non è ammissibile, nel senso che prima o poi il valore di un timestamp sarà ripetuto (sui computer non esiste l'infinito). Questo tuttavia, non

crea alcun problema in quanto l'unicità dei timestamp è richiesta al solo scopo di garantire che non esistano, in un dato istante, due transazioni in esecuzione sul sistema che abbiano lo stesso timestamp. Il problema è analogo a quello dei turnomatic: due clienti con lo stesso numero creerebbero un po' di confusione per l'addetto al banco, ma il problema non esiste se i due clienti si presentano in giorni diversi; per cui è lecito distribuire tagliandi di uguale valore purché si assicuri che essi non siano presentati concorrentemente. Tornando ai timestamp possiamo dire che basterà realizzarli come interi, modulo il massimo intero rappresentabile sul sistema: due valori uguali saranno generati dopo un intervallo di tempo abbastanza lungo da garantire che le due transazioni non siano in esecuzione contemporaneamente.

La generazione dei timestamp può essere affidata ad un processo TD (Timestamp-Dispatcher) cui una transazione utente può chiedere un timestamp mediante l'invio di un messaggio «ASK» che provocherà la generazione e l'invio, da TD al processo che esegue la transazione, del valore richiesto. L'interazione tra i processi è schematizzata in figura 1.

Oltre che di unicità, la definizione del timestamp, parla di globalità. Per comprendere la necessità di questo requisito, occorre ricordare che il problema del controllo della concorrenza coinvolge generiche transazioni che non necessariamente sono in esecuzione sulla stessa macchina o comunque sullo stesso processore, ma in generale su un sistema distribuito. Una transazione è cioè, una sequenza di operazioni invocate da un processo in esecuzione su un preciso «NODO» del sistema, su risorse condivise del sistema, residenti su nodi qualsiasi. Non si perde di generalità se si considera sempre solo il caso di un sistema distribuito. La figura 2 illustra un insieme di nodi del sistema distribuito, coinvolti da una transazione.

Le risorse coinvolte da una transazione, possono essere suddivise in risorse LOCALI (al nodo cui appartiene la transazione) e risorse REMOTE (sempre al nodo stesso).

Le risorse locali possono essere usate da una transazione in maniera diretta (scambiando messaggi con il processo gestore). Per ciò che riguarda le risorse remote, la loro invocazione è filtrata dai meccanismi di «naming globale» e dirottata verso i nodi su cui risiedono le risorse, dove il sistema operativo distribuito provvederà ad inoltrarle simulando una transazione locale. Il modo in cui ciò avviene esula dal contesto. Concentriamoci sull'istante in cui una certa risorsa riceve un'invocazione da una certa transazione.

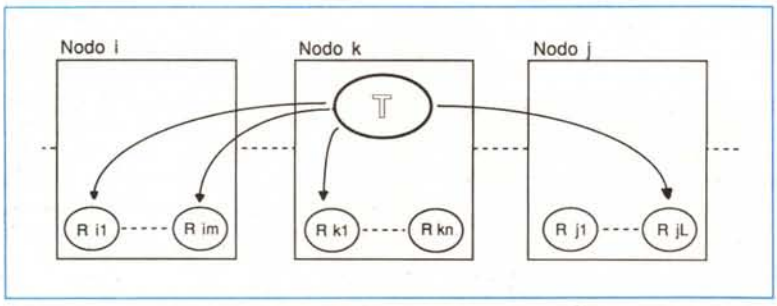


Figura 2 - Tre nodi di un Sistema Distribuito, il nodo i, il nodo k ed il nodo j, aventi rispettivamente: M, N ed L risorse condivise dalle applicazioni dell'intero sistema. La transazione I, in esecuzione sul nodo K, coinvolge, mediante le sue operazioni, sia risorse residenti sul suo stesso nodo (Rk1) che risorse di altri nodi (Ri1, Rim ed Rjl).

Locale o remota che sia, l'invocazione porta con sé il timestamp della transazione che l'ha generata. A questo punto dovrebbe essere chiaro cosa si intende per globalità dei timestamp: essi devono essere validi sull'intero sistema, per poter essere riconosciuti anche dalle risorse remote. Questo problema non è banale perché se ci fosse un unico Timestamp-Dispatcher sull'intero sistema, questo costituirebbe un collo di bottiglia per tutte le transazioni (che dovrebbero «fare la fila» per ottenere un timestamp e poter quindi partire); per cui di TD ne occorre uno per ogni nodo. A questo punto però nasce un nuovo problema, per comprendere il quale, è utile fare un'analogia con l'esempio del supermercato. È chiaro che l'analogia è del tutto forzata e non rispecchia una situazione reale.

Proviamo ad immaginare una catena di supermercati futuristica. In ognuno di questi supermercati c'è un unico banco. Supponiamo che il Sig. Rossi sia uno dei clienti del banco del supermercato X, e che egli, come tutti i clienti della catena, abbia la possibilità di acquistare formaggi non presenti al banco di X, in un qualsiasi altro banco della catena che ne sia provvisto, diciamo al banco del supermercato Z. Quello che vogliamo è che il suo tagliandino prelevato dal turnomatic di X sia valido anche al banco di Z, in modo da realizzare un turnomatic globale e decentralizzato che eviti di far rifare la fila al Rossi.

Se i vari turnomatic funzionano così come sono attualmente, può accadere che il tagliando del Rossi abbia il numero 50, e che al banco X sia arrivato il suo turno, mentre al banco Z, dove il Rossi (ottima forchetta) vorrebbe acquistare un prelibatissimo gorgonzola non presente al banco X, essendoci stato più movimento di clienti, siano già al numero 80. È evidentemente impensabile far attendere il Rossi fino a quando al banco Z si richiamerà il numero 50, cosa che avverrà dopo 69 clienti, essendo (80+69)mod(100)=49. Oltretutto, dopo un paio d'ore anche al banco Z qualcuno prenderà il tagliando 50 (di una nuova serie) e finirà con il litigare paurosamente con il Rossi già abbastanza innervosito. Quest'ultimo problema tuttavia, è di facile risoluzione; basterà che i tagliandi dei diversi supermercati abbiano, a parità di numero, una priorità predefinita in modo che, ad esempio, il tagliando 50 del supermercato Z, che indicheremo con (50,Z), venga dopo del (50,X) ma prima sia del (51,X) che del (51,Z). Ma il problema grosso è l'altro.

La risoluzione del problema consiste nel far riferimento al tempo: il valore dei tagliandini distribuiti dal turnomatic sarà l'orario, compreso di centesimi di secondo (non si sa mai) in cui il cliente lo preleva. Sarà dunque necessario associare al turnomatic un orologio ed una stampantina. Anche in questo caso va adottato l'espediente della priorità predefinita a parità di valore del tagliando per poter gestire, improbabili coincidenze. Il risultato ottenuto in questo caso, soddisfa la politica che volevamo adottare: in qualunque supermercato della catena, il Sig. Rossi ha un tagliando valido, unico e soddisfacente. Posto, ovviamente, che l'addetto al banco serva il cliente che ha il tagliando di valore minimo, chi deve aspettare non farà storie.

È questo il modo in cui il problema è risolto nel caso dei timestamp delle

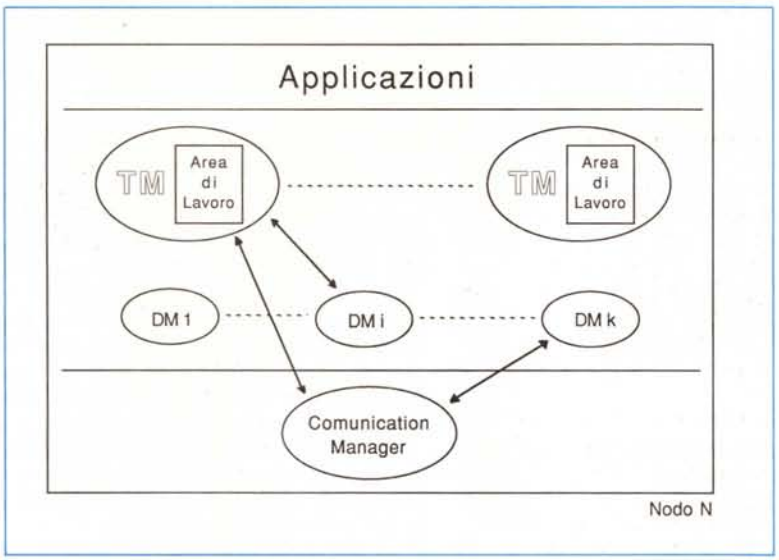


Figura 3 - Un generico nodo del sistema. Su di esso le risorse condivise sono inglobate in processi gestori (DM) e le applicazioni concorrenti o parallele, che vogliono utilizzare risorse del sistema, possono farlo solo affidando l'esecuzione di transazioni, di processi di sistema (TM) preposti a farlo.

transazioni. Però questo tipo di soluzione, fa nascere un nuovo problema che nel caso del turnomatic ha poca importanza ma nel caso dei timestamp assume una rilevanza notevole, come si capirà meglio quando vedremo il timestamping. È il problema del cosiddetto tempo assoluto. In parole semplici: il clock, così come i timestamp dispatcher, sono distribuiti sul sistema, cioè ce n'è uno per ogni nodo. Ne consegue che i vari clock, col passare del tempo, finiranno con l'essere sfalsati tra di loro. Per questo motivo sarà necessario implementare meccanismi di sincronizzazione dei clock, argomento che richiederebbe una trattazione a sé stante.

Ricapitolando: su ogni nodo del sistema distribuito, verrà implementato un processo timestamp-dispatcher che interagisce, oltre che con le transazioni, anche con un processo clock e con gli altri TD del sistema. Ogni transazione, prima di partire nell'esecuzione, genera una richiesta al TD del suo nodo, ricevendo una marca temporale assai simile a quella che avrebbe ottenuto se ci fosse stato un clock assoluto.

Dopo questa lunga chiacchierata sui problemi inerenti alla realizzazione dei timestamp, non ci resta che entrare nel vivo della politica di controllo della concorrenza.

Il Timestamping

Il meccanismo in esame, diversamente da quello del Locking a due fasi esaminato il mese scorso, non è statico (basato sulla struttura delle transazioni) ma dinamico (cioè basato sull'adozione a run-time di opportune specifiche di interazione e di gestione). Tutto ciò vuol dire che il livello del sistema che interpreta le transazioni sarà organizzato «ad hoc», e precisamente nel seguente modo.

Il modello

Ogni transazione, generata da un programma applicativo, è affidata ad un Transaction Manager TM, che si farà carico di eseguire la sequenza di operazioni, sfruttando particolari primitive di interazione con i gestori sono detti Data Manager DM. Durante l'esecuzione di una transazione T, il TM incaricato sarà un processo di sistema, residente sullo stesso nodo cui appartiene l'applicazione richiedente l'esecuzione T, che utilizzerà allo scopo un'area di lavoro dove saranno mantenuti i valori parziali della transazione (vedi figura 3).

Un TM può comunicare con i vari DM coinvolti dalla transazione, mediante scambio di messaggi. Tali messaggi sono del tipo «Call-Reply», consistono cioè in invocazioni del TM sul DM, seguite da una risposta del DM al TM; i messaggi tra un TM ed un DM residenti su nodi diversi, sono dirottati verso i Communication Manager che sono processi di sistema gestori delle comunicazioni in remoto.

La tabella in figura 4, mostra i possibili Reply del DM per ogni Call del TM; è necessario tenerla sott'occhio per comprendere la politica con cui sono gestite le varie invocazioni al fine di implementare il timestamping.

Messaggi di call

Un TM comunica con ogni DM necessario, al più tre volte durante l'esecuzione della transazione.

1) Se le operazioni della transazione T richiedono la conoscenza del valore dell'oggetto gestito dal DM, tale valore può essere ottenuto invocando una READ. Il DM può rispondere o con il valore dell'oggetto OBJVALUE, oppure rigettando la richiesta, la qual cosa costringerà il TM ad abortire T e ricominciarne daccapo l'esecuzione (richiedendo un nuovo timestamp per T). etto può essere letto al più una volta, il valore ottenuto verrà poi mantenuto nell'area di lavoro del TM, e soggetto ad eventuali modifiche. È chiaro che se T opera sulla risorsa in sola scrittura, la richiesta di READ non sarà generata.

Call (da TM a DM)	Reply (da DM a TM)
Read	Objvalue | Reject
Prewrite	Accept | Reject
Write	———
Abort	———

Figura 4 - Tabella dei messaggi scambiati tra TM e DM, nel Timestamping.

2) Quando tutte le operazioni di T sono state eseguite (nell'area di lavoro), il TM genera una PREWRITE su ogni oggetto da modificare fisicamente. L'invocazione di prewrite può essere considerata come la richiesta del permesso a committare la transazione. Ogni DM risponderà o accettando o rigettando la prewrite.

3) Quando tutti i Reply delle prewrite generate, son giunte al TM, se essi sono tutti messaggi di accettazione allora verrà invocata su ogni DM, la scrittura del nuovo valore dell'oggetto, altrimenti, se almeno un DM ha rigettato la prewrite del TM, allora quest'ultimo comunicherà un ABORT a tutti i DM cui aveva precedentemente inviato la prewrite. Sia write (objvalue) che abort, sono Call con Reply indefinite.

Algoritmo di Scheduling dei DM

Vediamo ora come si comporta un generico DM in risposta alle diverse invocazioni dei TM.

Dal punto di vista del DM, ciò che identifica una transazione è il suo timestamp; ogni invocazione del TM dovrà quindi portare con sé il valore del timestamp della transazione che sta eseguendo; indicheremo tale valore con

ts.

Il DM si avvale di alcune strutture dati proprie del meccanismo di scheduling, in particolare ci saranno tre variabili:

— WTS contiene il ts più grande fra quelli delle transazioni che hanno invocato write sull'oggetto.

— RTS contiene il ts più grande fra quelli delle transazioni che hanno invocato read sull'oggetto.

— PTS contiene il ts più piccolo fra quelli delle transazioni che hanno invocato prewrite sull'oggetto, ma non hanno ancora inviato write o abort.

Saranno inoltre necessarie alcune code d'attesa.

La politica di scheduling è la seguente:

(READ,ts)

è accettata solo se ts>WTS, ma l'esecuzione è ritardata fino a quando tutte le prewrite accettate, di transazioni con timestamp minori di ts, non sono state seguite dalle corrispodenti write o abort.

(PREWRITE,ts)

è accettata solo se non ci sono state letture o scritture di transazioni con timestamp maggiori di ts, altrimenti è rigettata.

(WRITE,ts)

è sempre accettata, poiché è stata accettata la precedente prewrite, ma viene accodata ed eseguita in ordine di timestamp crescenti rispetto alle transazioni che hanno read accodate o prewrite non ancora seguite da write o abort.

La figura 5 riporta l'algoritmo in notazione Pascal-like.

Tra parentesi quadre, sono racchiuse alcune «chiamate di procedura» che servono per operare sulle strutture dati WTS, RTS, PTS e sulle code d'attesa. La procedura AGGIORNA, oltre che aggiornare tali strutture, si preoccupa di rimuovere dalle code ed eseguire, le richieste pendenti invocate sull'oggetto. Tali procedure non son riportate, per evitare di appesantire la trattazione con dettagli che non coinvolgono la politica del Timestamping.

```
(Read, ts):
Begin
  if ts > wts then
              if ts < pts then Reply(Objvalue)
                          else [accoda la richiesta]
              else Reply ("Reject"); Aggiorna
end


(Prewrite, ts):
Begin
  if ts >= rts and ts > wts
  then  Replay("Accept")
  else  Replay("Reject"); Aggiorna
end


(Write, ts):
Begin
  if  (ts < [ogni read accodata]) and ts <= pts then
  begin
     Replay("Accept");
     [Esegui Write];
  end
  else [accoda la richiesta];
  Aggiorna
end


(Abort, ts):
Begin
   Aggiorna
end
```

Figura 5

Conclusioni

È difficile tirare le somme di un argomento così astratto ed articolato come il meccanismo del timestamping. Conscia di tale difficoltà, prima ancora di accingermi a farlo, vorrei incoraggiare i lettori più interessati ad effettuare studi e simulazioni sul meccanismo stesso; allo scopo è riportata, in fondo, una breve bibliografia sull'argomento.

Il meccanismo permette un grado di parallelismo molto vicino a quello del Two Phases Locking, nonostante abbia una complessità maggiore di quest'ultimo. La linea-guida del timestamping può essere espressa con un motto che suona all'incirca così: «largo ai giovani», nel senso che in presenza di scorrette serializzazioni, vengono privilegiate le transazioni più giovani (con timestamp maggiori). Ci si può rendere conto facilmente di questo privilegio, considerando (vedi figura 5) che l'avvenuta lettura del valore dell'oggetto da parte di transazioni giovani, preclude l'accettazione di successive prewrite generate da transazioni più vecchie.

La vera potenza del timestamping, risiede nella capacità di accodare READ e WRITE, ed eseguirle solo quando è giunto il momento. In tal modo molte transazioni evitano di essere abortite.

Sul meccanismo sono state effettuate numerose, lunghe e noiose prove di correttezza. È sicuramente meno noioso, anche se meno corretto, provare il meccanismo con qualche esempio.

Se consideriamo la seguente sequenza di invocazioni sull'oggetto gestito da un generico DMi: (READ, 1); (PREWRITE, 1); (WRITE, 1); (READ, 2); (PREWRITE, 2); (WRITE, 2), si avrà che DMi accetterà tutte le richieste e le due transazioni saranno committate entrambe, posto che sugli altri DM si presentino nello stesso ordine e non vengano disturbate da altre transazioni.

La seguente sequenza:
(READ, 1); (PREWRITE, 2); (PREWRITE, 1); (WRITE, 2); (WRITE, 1), sarà anch'essa accettata da DMi, grazie al fatto che la (WRITE, 2) verrà bufferizzata ed eseguita dopo la (WRITE, 1).

Infine invece, la sequenza:
(READ, 1); (PREWRITE, 2); (WRITE, 2); (PREWRITE, 1); (WRITE, 1), nonostante sia molto simile a quella precedente, provocherà l'accettazione della transazione con timestamp 2, ma l'aborto di quella con timestamp 1. In realtà la (WRITE, 1) non sarà affatto generata in quanto il TM riceverà REJECT in risposta alla richiesta (PREWRITE, 1), essendo in quell'istante WTS=2 e quindi maggiore del suo ts. Si noti, che se una transazione viene abortita, è sempre a causa di transazioni più giovani.

Two Phases Locking e Timestamping, sono solo le due più diffuse politiche di controllo della concorrenza. La ricerca in questo campo, è aperta alla realizzazione di meccanismi che permettano un più alto grado si concorrenza fra le transazioni, anche in considerazione della sempre crescente diffusione di sistemi a parallelismo massiccio. MC

Bibliografia

J. B. Moss
«Nested Transaction»
MIT PRESS Series in Information System.

B. Liskov, R. Scheifler
«Guardians and Action: Linguistic Support for Robust, Distributed Programs»
ACM Transaction on Programming Languages and Systems 5,3 July 1983.

P. A. Bernstein, D. W. Shipman, W. S. Wong
«Formal Aspect of Serializability in Database Concurrency Control»
IEEE Transaction on Soft. Eng. SE-5, 3 May 1979.

P. A. Bernstein, N. Goodman
«Timestamp-based Algorithms for Concurrency Control in Distributed Database Systems».

Il Computer e' servito!

postalbit

VENDITA PER CORRISPONDENZA
Terminal Centro e Sud d'Italia
Tel. 06/5424303

PERSONAL COMPUTER MANAGER

COD.	DESCRIZIONE ARTICOLO	PREZZO I.E
101	XT TURBO Frequenza di clock 4,7/8 Mhz 256KRAM On board espandibili a 640K Scheda video monocromatica o colore Scheda controller Porta stampante parallela Tastiera tipo esteso	603.000
102	XT TURBO Caratteristiche tecniche come 101 1 Floppy disk drive 360K 5'' 1/4	720.000
103	XT TURBO Caratteristiche tecniche come 101 2 Floppy disk drive 360 K	837.000
104	XT TURBO Caratteristiche tecniche come 101 1 Floppy disk drive 360K 5'' 1/4 1 Hard disk 20 Mb 1 Controller Hard disk	1.204.000
105	XT TURBO Caratteristiche tecniche come 101 1 Floppy disk drive 720K 3'' 1/2	740.000
106	XT TURBO Caratteristiche tecniche come 101 2 Floppy disk drive 720K 3'' 1/2	875.000
107	XT TURBO Caratteristiche tecniche come 101 1 Floppy disk drive 720K 3'' 1/2 1 Hard disk 20 Mb 1 Controller Hard disk Cavi di collegamento	1.223.000
108	XT TURBO Caratteristiche tecniche come 101 1 Floppy disk drive 720K 3'' 1/2 1 Floppy disk drive 360K 5'' 1/4	856.000
301	XT TRAVELLER Frequenza di clock 4,7/8 MHZ 256KRAM On board Exp 640K Scheda video colore Monitor LCD orientabile 2 Floppy Disk Drive 720K o 360K Scheda parallela Tastiera	1.775.000
302	XT TRAVELLER Caratteristiche tecniche come 301 Hard disk 20Mb	2.258.000
303	XT TRAVELLER Caratteristiche tecniche come 301 Scheda video EGA Monitor LCD EGA	1.894.000
304	XT TRAVELLER Caratteristiche tecniche come 302 Scheda video EGA Monitor LCD EGA	2.378.000
305	AT TRAVELLER Frequenza di clock 12 Mhz 512k Ram exp 1 Mb on board Scheda video Colore Monitor LCD CGA orientabile Porta stampante parallela Tastiera 1 Floppy Disk Drive 1,2 Mb Hard Disk 20 Mb	2.693.000
306	AT TRAVELLER Caratteristiche tecniche come 305 Hard Disk 40Mb	3.009.000
307	AT TRAVELLER Caratteristiche tecniche come 305 Scheda video EGA Monitor LCD EGA	2.813.000
308	AT TRAVELLER Caratteristiche tecniche come 306 Scheda video EGA Monitor LCD EGA	3.129.000
309	OPZIONE VELOX Supplemento per MB 16Mhz	135.000
310	BORSA TRAVEL	67.000
401	AT Frequenza di clock 12 Mhz 512K Ram exp 1 Mb on board Scheda video Colore Grafica o Hercules Porta stampante parallela Tastiera Avanzata 1 Floppy Disk Drive 1,2 Mb	1.456.000
402	AT Caratteristiche tecniche come 401 Hard Disk 20 Mb	1.846.000
403	AT Caratteristiche tecniche come 401 Hard Disk 40	2.162.000
404	AT Caratteristiche tecniche come 401 Hard Disk 70 Mb	2.989.000
405	OPZIONE VELOX AT Supplemento per MB 16Mhz	135.000
406	OPZIONE CASE TOWER 386 LIKE	250.000
501	386 TOWER Microprocessore Intel 80386 32 Bit Doppio Sigma 2 Mb Ram On Board Scheda video Grafica Colore o Hercules Porta stampante parallela Real Time clock Tastiera avanzata 1 Floppy Disk Drive 1,2 Mb MS Dos e manuale d'uso	4.707.000
502	386 TOWER Caratteristiche tecniche come 501 Hard Disk Drive 20 Mb	5.098.000
503	386 TOWER Caratteristiche tecniche come 501 Hard Disk 40 Mb	5.413.000
504	386 TOWER Caratteristiche tecniche come 501 Hard Disk 70 Mb	6.240.000

ESPANSIONI MEMORIA MANAGER

601	Espansione Memoria 256/512	Telefonare
602	Espansione Memoria 512/640	Telefonare
603	Espansione Memoria 640/1MB	Telefonare
604	Espansione Memoria 1MB/2/MB	Telefonare

SCHEDE OPZIONALI MANAGER

601	Scheda Video CGA W/Printer	70.000
602	Scheda Video CGA Professional	70.000
603	Scheda Printer	31.000
604	Scheda Video HERCULES W/Printer	70.000
605	Scheda Video HERCULES W/PRT 16MHZ	126.000
606	Scheda Video EGA	211.000
607	Scheda Video EGA 480	260.000
608	Scheda Video VGA	370.000
609	Scheda RS232	30.000
610	Chips II Porta RS232	25.000
611	Scheda Multifunzione	75.000
612	Scheda GAME	30.000
613	Scheda CLOCK	40.000
614	Scheda MODEM	150.000
615	Box esterno per MODEM	136.000
616	RAMCARD 2,5Mb OK	130.000
617	Seriale 4 Porte per XENIX	250.000
618	Scheda Software Protection	440.000
619	EVA 600x800 256K	586.000
620	EVA 1024x1024 512K	836.000
621	Scheda FAX W/Software	863.000

MEMORIE DI MASSA MANAGER

701	Floppy Disk Drive 360K	118.000
702	Floppy Disk Drive 720K W/Adapter	153.000
703	Floppy Disk Drive 1,2 MB	185.000
704	Hard Disk 20 Mb	382.000
705	Hard Disk 40 Mb	691.000
706	Hard Disk 70 Mb	1.515.000
707	Hard Disk Card 30 Mb	620.000
708	Back Up 20Mb MS-DOS	650.000
709	Natro per 708	53.000
710	Back Up 40/60Mb MS-DOS	1.019.000
711	Back Up 40/60Mb XENIX	1.150.000

ACCESSORI HARDWARE MANAGER

801	Deviatore 1/2 PC/Stampanti	50.000
802	Cavo per Collegamento a D1	20.000
803	Mouse Meccanico	60.000
804	Mouse LOGITEK Comp.le	80.000
805	Joystuck PC/AT/386	25.000

COPROCESSORI MANAGER

901	Coprocessore Math XT	300.000
902	Coprocessore Math AT	545.000
903	Coprocessore Math 387/16	1.100.000

RICAMBI E PARTI MANAGER

1001	Cabinet XT W/Alimentatore	110.000
1002	Cabinet AT W/Alimentatore	160.000
1003	Cabinet TOWER W/Alimentatore	310.000
1004	Cabinet TRAVEL W/Alim.Kb/Mon CGA	1.160.000
1005	Cabinet TRAVEL W/Alim.Kb Mon EGA	1.280.000
1006	Tastiera Avanzata	68.000
1007	Tastiera Avanzata Microswitch	80.000
1008	Mother Board XT 4,7/8 MHZ O RAM	115.000
1009	Mother Board AT 10 MHZ O RAM	414.000
1010	Mother Board AT 16 MHZ O RAM	500.000
1011	Mother Board 386 32 BIT 2MB RAM	4.500.000
1012	Floppy Disk Controller XT W/CAVI	25.000
1013	Hard Disk Controller XT W/Cavi	95.000
1014	Controller AT W/Cavi	195.000

CAVI E CONNETTORI MANAGER

1015	Cavo Centronics	13.000

MONITOR MANAGER

2001	TTL Fosfori Verdi 12''	125.000
2002	TTL Fosfori Verdi 12'' Basculante	130.000
2003	Bifrequenza Fosfori Bianchi 14'' Basculante	195.000
2004	Colori Media Risoluzione	385.000
2006	Colori E.G.A. Bascualnte	645.000
2006	Colori multisync Basculante	850.000

PHILIPS Computer

PHI	PHILPS 3105/1 CPU 8088-2-512K RAM 1 Floppy Disk Drive 3'' 720K Uscita seriale (9 PIN) e parallela Scheda Video monocromatica e colore Clock di sistema Tastiera italiana 84 tasti	1.120.000
PH2	PHILIPS 3105/2 CPU 8088-2 786K RAM 2 Floppy disk drive 3'' 1/2 720K Uscita seriale (9 PIN) e parallela Scheda video monocromatica e colore Clock di sistema Tastiera italiana 84 tasti	1.225.000
PH3	PHILIPS 3105/3 Come sopra ma con 1 Floppy disk drive 3'' 1/2 720K 1 Hard disk 3'' 1/2 20Mb	1.810.000
PH4	PHILIPS 3204/1 CPU 80286 640K RAM 1 Floppy disk drive 5'' 1/4 1,2Mb 1 Hard disk 3'' 1/2 20Mb Uscita seriale (9 PIN) e parallela Clock di sistema Scheda video monocromatica e colore	2.910.000

STAMPANTI

SO1	OKI ML 182PI80C 120 CPS NLQ Grafica	500.000
SO2	Come sopra SERIALE	580.000
SO3	OKI ML 192PI80C CPS NLQ Grafica Con inseritore Frontale di fogli singoli	750.000
SO4	Come sopra SERIALE	830.000
SO5	OKI ML 193P/1 136C 200 CPS NLQ Grafica Con inseritore Frontale di fogli singoli	850.000
SO6	Come sopra SERIALE STAMPANTI 18 AGHI	1.000.000
SO7	OKI ML292P/I/S 80C 240 CPS LQ Grafica stampa NERO e a COLORI SW per Hardcopy	900.000
SO8	OKI ML293 come sopra ma a 136C	1.150.000
SO9	OKI ML294P/I/S 136C 400 CPS LQ Grafica Stampa in NERO e a COLORI SW per Hardcopy STAMPANTI 24 AGHI	1.550.000
SO10	OKI ML 393P/I/S 136C 350 CPS LQ Grafica	2.700.000
SO11	Come sopra ma a Colori	3.000.000
SO12	OKI ML 390P/I 80C 270 CPS LQ Grafica	1.200.000
SO13	Come sopra SERIALE	1.450.000
SO14	OKI ML 391P/I 136C 270 CPS LQ Grafica	1.800.000
SO15	Come sopra SERIALE	2.050.000

STAMPANTI LASER

SO16	OKI LASERLINE 6 PPM 512K 300X300 DPI HPLJII, EPSON,IBM, DIABLO, QUME, NEC	4.000.000
SO17	Come sopra con RAM 1,5 Mb	4.500.000
SO18	Come sopra con RAM 2,5 Mb	5.200.000
SO19	OKI TPL 12 PPM LASER 512K 300X300 DPI HPLJ+, EPSON, DIABLO	6.200.000
SO20	Come sopra con RAM 2 Mb	7.300.000

ACCESSORI COMPUTER

AX1	SOTTOSTAMPANTE PLEXIGLASS 80C	67.000
AX2	SOTTOSTAMPANTE PLEXIGLASS 132C	112.500
AX3	SOTTOSTAMPANTE TUBOLARE 80C	45.000
AX4	SOTTOSTAMPANTE TUBOLARE 132C	55.500
AX5	DISK BOX 10 P. 3'' 1/2	3.370
AX6	DISK BOX 40 P. 3'' 1/2	21.000
AX7	DISK BOX 80 P. 3'' 1/2	36.500
AX8	DISK BOX 10 P. 5'' 1/4	3.370
AX9	DISK BOX 50 P. 5'' 1/4	21.000
AX10	DISK BOX 90 P. 5'' 1/4	23.500
AX11	TAPPETINO ANTISTATICO PER MOUSE	48.000
AX12	TAPPETINO ANTISTATICO PER TASTIERA	61.000
AX13	TAPPETINO ANTISTATICO PER CPU	57.000
AX14	PIASTRA GIREVOLE PER MONITOR	61.500
AX15	FILTRO ANTIRADIAZIONI MONITOR 12''	130.500
AX16	FILTRO ANTIRADIAZIONI MONITOR 14''	157.500
AX17	KIT PULIZIA PER COMPUTER	17.500
AX18	KIT PULIZIA PER FDD 5'' 1/4	22.000
AX19	KIT PULIZIA PER FDD 3'' 1/2	37.000
AX20	GAS DI PULIZIA SPRAY	10.000
AX21	COVER TASTIERA XT	26.000
AX22	COVER TASTIERA AT	26.000
AX23	COVER TASTIERA AVANZATA	26.000
AX24	COVER TASTIERA M19/M28	26.000
AX25	COVER TASTIERA M24	26.000
AX26	COPRICOMPUTER/MONITOR XT	22.000
AX27	COPRICOMPUTER/MONITOR AT	22.000
AX28	COPRICOMPUTER/MONITOR M19	22.300
AX29	COPRICOMPUTER/MONITOR M24	22.000
AX30	COPRICOMPUTER/MONITOR M28	22.000
CA1	1000 FOGLI CARTA BIANCA 80C	27.000
CA2	1000 FOGLI CARTA L.F. 80C	27.000
CA3	1000 FOGLI CARTA L.F. 136C	45.000
CA4	1000 ETICHETTE ADESIVE M.C.	18.000
CA5	500 FOGLI USO BOLLO M.C.	19.000
CA6	500 BUSTE IN MODULO CONTINUO	44.500
CA7	500 RICEVUTE BANCARIE M.C.	24.000

I prezzi non includono l'I.V.A. e il trasporto.

La merce viaggia a rischio del committente.

I prodotti sono garantiti 12 mesi dalla data dell'acquisto.

TUTTI I PREZZI SONO IVA ESCLUSA

Conservate questo listino - Resterà valido fino alla prossima uscita

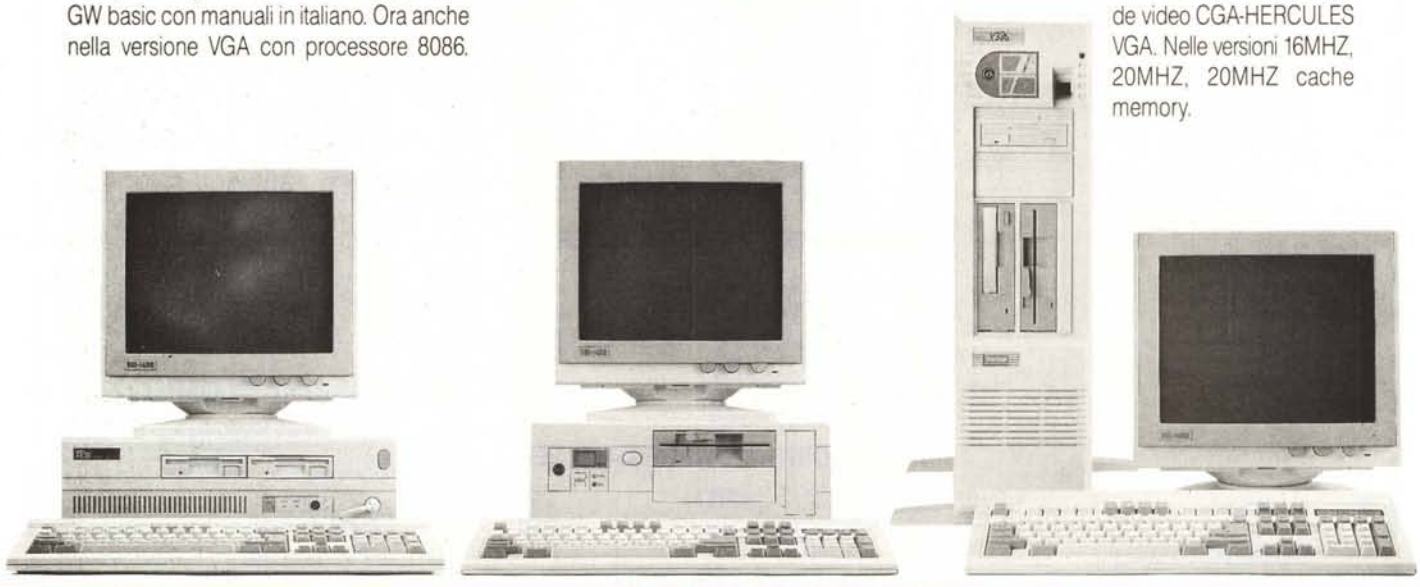
A GENOVA...
HARDWARE & DISTRIBUZIONE

PERSONAL COMPUTER INTERCOMP

XPC30 Personal computer MS/DOS compatibile con clock a 4.77/10MHZ, 640K a bordo, scheda video compatibile CGA-Hercules seriale, parallela, orologio, mouse adapter, spazio fino a 3 unità interne da 3.5" SLIM LINE, (drive 720/1.44MB, hard disk 20/40MB) dimensioni contenute, DOS e GW basic con manuali in italiano. Ora anche nella versione VGA con processore 8086.

XAT Personal computer 80286 a 10/12MHZ 1-0 WS 512K RAM espandibili a 1024 a bordo nuovo design a dimensioni contenute che lascia spazio a n. 2 alloggiamenti da 5.25" e 2 alloggiamenti da 3.5" SLIM. Tastiera estesa, DOS e GW BASIC con manuali in italiano. Adattatore video a scelta CGA-HERCULES-VGA.

X386 Processore 80386 1MB espandibili a 2 MB a bordo, 3 slot di espansione a 8 BIT, 2 a 16 BIT, 1 a 32 BIT, 8/16 porte seriali (opzionali) 2 alloggiamenti da 5.25" e due da 3.5" hard disk da 20 a 380MB, e fino a 800MB con disco ottico, tastiera estesa, DOS e GW BASIC con manuali in italiano, schede video CGA-HERCULES VGA. Nelle versioni 16MHZ, 20MHZ, 20MHZ cache memory.



ANCHE NELLE VERSIONI TRASPORTABILI

Elaboratori Intercomp
- Affidabilità
- Professionalità
- Design
- Budget

Soluzioni hardware e software per l'azienda e l'ufficio. Schede Add-On di espansione e interfaccia, accessori, mouse, reti, hard disk.

Monitor multisync NEC
- Multisync II 14"
- Multisync Plus 15" 960×720
- Multisync XL 20" 1024×768
- NEW Scheda video NEC MVA 1024×768 256 colori, drivers software in dotazione: Autocad, Ventura, Windows, Gem, 123, Symphony, e molti altri

Stampanti 24 aghi & laser NEC
- P 2200
- P 6 PLUS
- P 7 PLUS

ed accessori in pronta consegna. Tutti i supporti soft ed hard per la grafica ed il testo a 24 aghi

NEWS Modem per PS/2 con sotware completamente italiano in dotazione. Possibilità di collegarsi con Videotel, Itapac, Prestel, Minitel ecc. configurazione ed installazione automatiche software di telemanutenzione «freeway master» hard disk 40 ed 80 MB 19 millisecondi 3.5".

SILVER SYSTEM

PRODOTTI ED ACCESSORI PER L'INFORMATICA E L'OFFICE AUTOMATION
16156 PEGLI GENOVA - VIA TEVERE 2A-8-10
TEL. (010) 680.685-689.324 - FAX (010) 686609