

# Target Cross

The screenshot displays the Target Cross ERP interface with several key components:

- Ordini Clienti:** A window showing order details for client 000203, dated 10.06.2002, with a total value of 367.
- Totale per categorie:** A 3D bar chart titled 'Vendite comparate' showing 'Importi in Euro' for 'Vendite prec.' and 'Vendite periodo' across 'Categorie prodotti'. The chart is attributed to 'Four solutions srl'.
- Diagramma di Gantt:** A Gantt chart showing work cycles for 'Assemblaggio' and 'Fresatura' from Sab 14.09.02 to Dom 15.09.02.
- Ciclo di lavoro:** A pop-up window for 'Ciclo di lavoro 2002-DRP-0000007/0001' showing machine and operator times.
- Carico macchine:** A chart showing machine load percentages for various work centers (BO.05, BO.06) from Mer 13.09 to Gio 21.0.
- Table: Totale per categorie**

categoria	Descrizione categoria	Vendite	Venduto precedente
01	Schede madri	417.555,00	417.555,00
	Bullonerie	481.060,63	784.340,63
	Semilavorati generali	61.975,00	61.975,00
	Articoli in stampa normale	61.975,00	61.975,00
<b>Codice categoria</b>		<b>Vendite</b>	<b>Venduto precedente</b>
		<b>1.027.965,63</b>	<b>1.331.245,63</b>

## Top technology

L'architettura di Target Cross rappresenta una pietra miliare dell'evoluzione degli ERP.

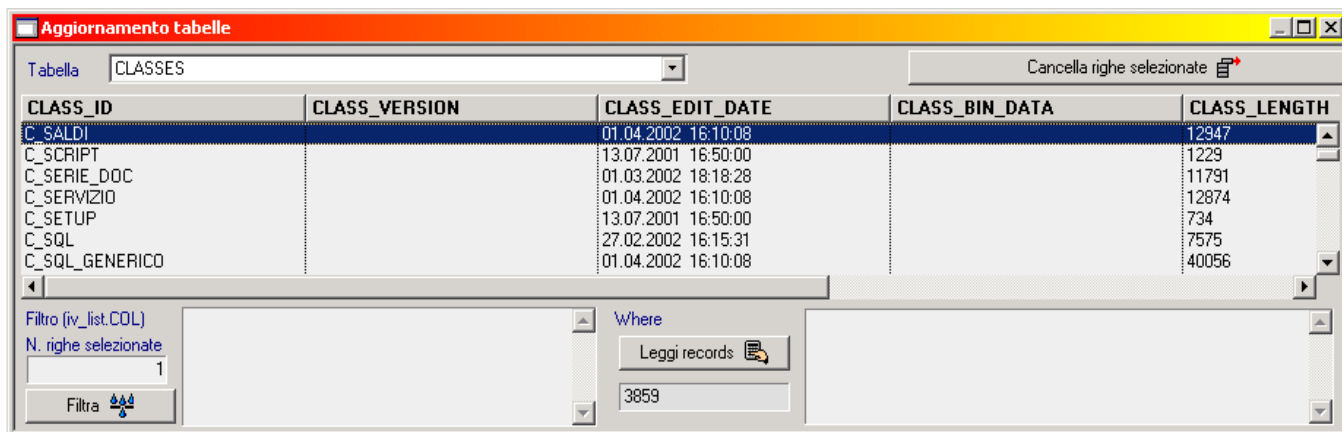
Per comprendere le avanzatissime caratteristiche del prodotto, è necessario gettare uno sguardo alla sua struttura interna ed a come è stato ingegnerizzato.

In questa presentazione saranno illustrati i principali punti che rendono Target Cross veramente unico e diverso nel panorama del software gestionale.

# L'architettura di sistema

Le straordinarie funzionalità e prestazioni del nostro software sono determinate e rese possibili da una serie di esclusività tecniche che nel loro insieme determinano una architettura di progetto estremamente avanzata.

## Il programma è memorizzato in un database



CLASS_ID	CLASS_VERSION	CLASS_EDIT_DATE	CLASS_BIN_DATA	CLASS_LENGTH
C_SALDI		01.04.2002 16:10:08		12947
C_SCRIPT		13.07.2001 16:50:00		1229
C_SERIE_DOC		01.03.2002 18:18:28		11791
C_SERVIZIO		01.04.2002 16:10:08		12874
C_SETUP		13.07.2001 16:50:00		734
C_SQL		27.02.2002 16:15:31		7575
C_SQL_GENERICO		01.04.2002 16:10:08		40056

Le esclusive caratteristiche di Target Cross derivano e sono rese possibili dal suo particolare tipo di struttura interna.

Come risultato della programmazione **object oriented**, la metodologia di sviluppo più avanzata attualmente disponibile, il programma è costituito da un certo numero di **oggetti**; ad esempio, una finestra è un oggetto, un menu un altro oggetto, ecc. Possiamo dire che oggi tutti i software più moderni presentano una struttura di questo tipo.

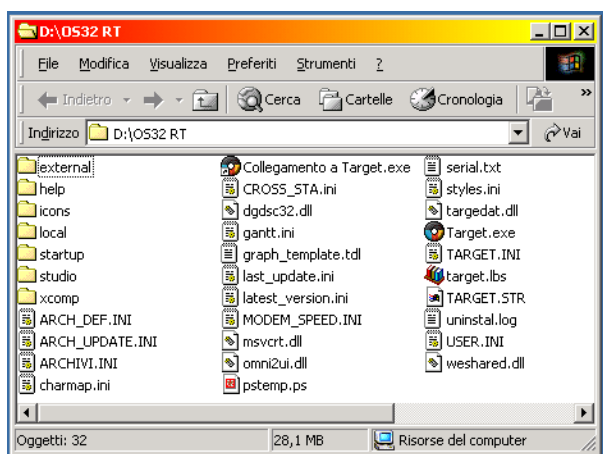
Le analogie però finiscono qui: a differenza degli altri programmi, **Target Cross non è né interpretato né compilato, ma memorizzato in un database di tipo particolare**. In pratica, come si memorizzano in un archivio Clienti, articoli e fatture, vengono memorizzati menu, finestre e tutti gli elementi del programma.

Questo costituisce la base del motore di personalizzazioni, in quanto il programma può modificare sé stesso mentre sta girando.

Oltre a questo, la peculiare struttura organizzativa permette una serie di avanzatissime caratteristiche che non sarebbero altrimenti possibili.

## Target Cross non deve essere installato

L'installazione del programma consiste essenzialmente nel copia e incolla di una cartella. Tutto qui.



In questa cartella **sono contenuti tutti i files** necessari al programma.

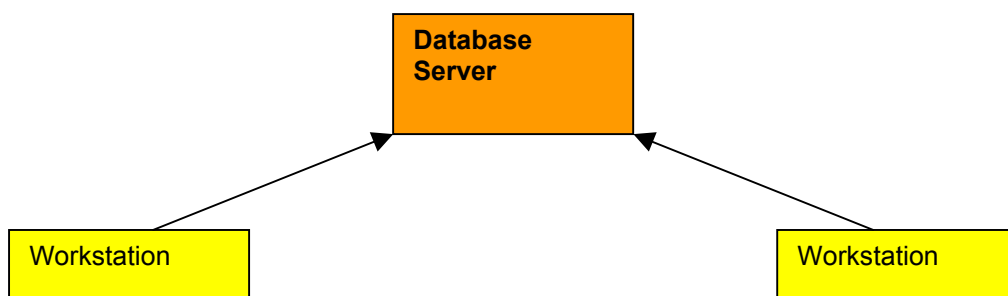
Nessuna procedura da seguire, **nessuna DLL che va in conflitto** con altri programmi o richiede una versione diversa. Il programma è interamente autocontenuto, non utilizza alcuna risorsa esterna e non modifica alcun file del sistema operativo (Windows o altri). Tra l'altro, come conseguenza di questa impostazione, il software presenta una **elevatissima stabilità** con una **probabilità di blocco estremamente bassa**.

## Programma centralizzato o distribuito?

Questo è il dilemma che si presenta in qualunque installazione non singolo utente.

I gestionali attualmente sono divisi tra questi due modelli organizzativi, in quanto ognuno di essi presenta vantaggi e svantaggi.

### Installazione distribuita



In questo modello, ogni workstation ha **una installazione del programma che lavora in modo indipendente**; l'unica cosa condivisa è il database server, a cui ogni Client **accede indipendentemente**.

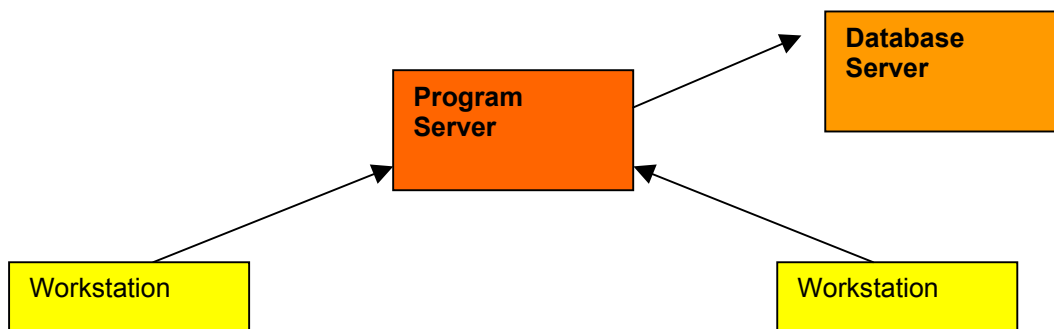
#### Vantaggi:

- Minimo traffico di rete (viaggiano solo i dati).
- Utilizzo delle risorse di calcolo di ogni singola macchina; in pratica è come avere un megacomputer, la cui potenza è data dalla somma delle singole macchine.
- Aggiungendo un posto di lavoro, il sistema degrada poco le prestazioni in quanto esso porta con sé nuove risorse di calcolo in aggiunta.

#### Svantaggi:

- E' necessaria una installazione per ogni posto di lavoro
- In molti casi, specie negli ERP di alto livello, la dimensione .del programma (diversi GigaBytes) impedisce di fatto l'installazione sui clients.
- Ogni aggiornamento deve essere installato su tutti i posti di lavoro.

### Installazione centralizzata



In questo tipo di installazione, il programma non gira su ogni singolo posto di lavoro, ma sul server in maniera centralizzata; sui Client vediamo solo le vedute generate sul server.

Questo rappresenta un evidente passo indietro in quanto **si torna per certi aspetti al terminale dei sistemi a carattere**, anche se in versione riveduta e corretta.

**Vantaggi:**

- Il programma viene installato una volta sola.
- Se esso è molto vasto, può comunque essere installato sul server
- Ogni aggiornamento è immediatamente disponibile in ogni Client

**Svantaggi**

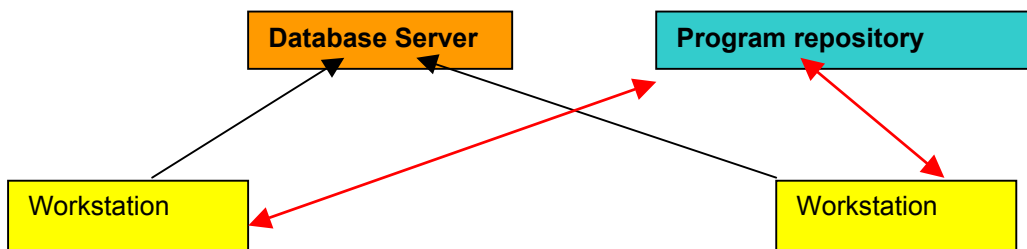
- Non si utilizzano le risorse di calcolo delle workstation
- E' necessario un server potente e costoso, con una quantità molto grande di Ram
- Ogni nuovo posto di lavoro grava interamente sul server e peggiora le prestazioni del sistema.
- Il traffico di rete è maggiore, in quanto tutte le operazioni si svolgono attraverso la rete.

## La soluzione di Target Cross

L'impostazione di Target Cross permette di ottenere i vantaggi di entrambi i sistemi, minimizzandone gli svantaggi.

Il programma può infatti lavorare in modo centralizzato, ma anche in modalità distribuita, senza presentarne gli svantaggi caratteristici.

Un Program Repository centralizzato infatti consente a tutte le workstation di lavorare in modo indipendente ma di essere allineate con la stessa versione del software in tempo reale.



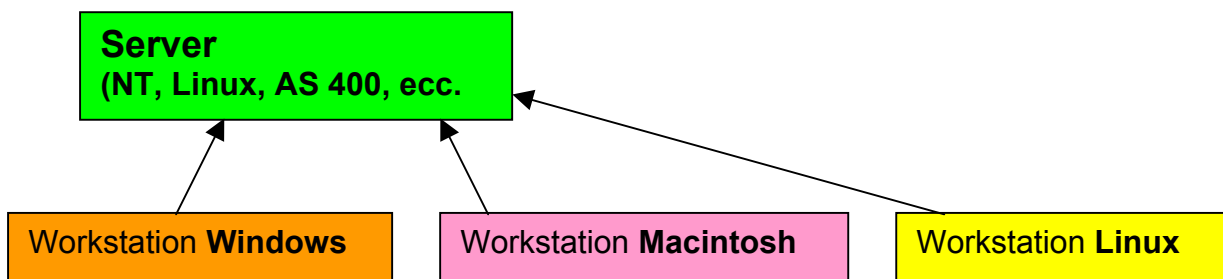
- **Installazione** per ogni posto di lavoro: il programma, come abbiamo visto precedentemente, non deve essere installato, quindi nessun onere aggiuntivo in questo senso.
- **Dimensioni del programma:** tutti i file di una installazione occupano circa 36 Mbytes.
- **Aggiornamenti:** essendo memorizzato un database, il programma può aggiornare sé stesso con una procedura automatica e rapidissima, in quanto riguarda solo gli oggetti che hanno subito una variazione dall'ultimo aggiornamento.

In questo modo vediamo che possiamo godere dei vantaggi dell'installazione distribuita, senza doverne sopportare gli svantaggi. Il programma potrà utilizzare fino in fondo le risorse di una rete di computer, sarà possibile avere un server molto meno potente, la rete sarà meno trafficata, senza per questo avere oneri aggiuntivi in caso di aggiornamenti e senza avere complicate e pesanti installazioni sui vari Client.

# L'architettura operativa

Le modalità funzionali di Target Cross sono state studiate per garantire agli Utenti la massima flessibilità ed efficacia operativa.

## Cross platform



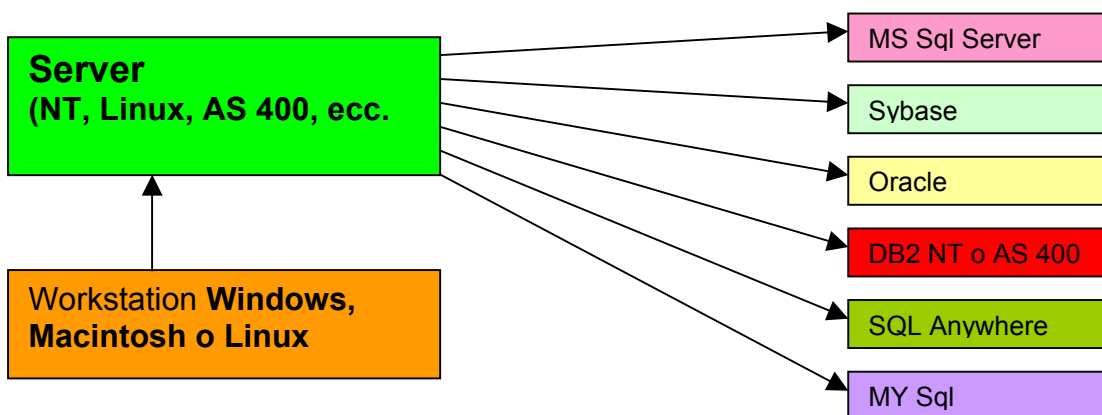
Target Cross può funzionare praticamente **su tutti i computer con interfaccia grafica**. Questo significa tutte le versioni di **Windows** (95, 98, NT, 2000, Xp) ma anche **Macintosh** (System X e sistemi operativi precedenti), **Linux** (tutte le distribuzioni) e **Sun Solaris**.

In ambiente Linux **non si limita ad utilizzare servers Linux**, come ormai tutti i programmi fanno, ma gira sui **client Linux** utilizzandone l'interfaccia grafica. Se si utilizza Linux come server, possiamo utilizzare **MySQL**, il database nativo e gratuito di Linux.

## Cross database

Target Cross può funzionare **praticamente con tutti i database in commercio**. Al momento sono certificati

- **SQL Server**
- **Sybase**
- **Oracle**
- **DB2 NT**
- **DB2 AS/400**
- **SQL Anywhere**
- **MySQL Linux**



E in generale tutti i database con un driver **ODBC**, oltre a potere utilizzare un database proprietario per piccole installazioni o dimostrativi.

# Massima completezza funzionale

## All in one

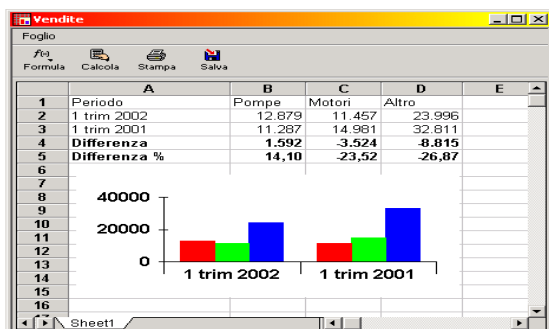
La filosofia di Target Cross è quella di implementare le funzioni internamente, quando molti prodotti, anche di gran nome, si appoggiano a prodotti esterni. La nostra opinione è che l'interazione tra prodotti diversi, che nella maggior parte dei casi si basa su import – export di dati, porta ad uno scadimento delle funzionalità e ad una limitazione per l'Utente.

Di conseguenza, una serie di funzioni che in prodotti concorrenti sono demandate a prodotti esterni in Target Cross sono incorporate nel programma. Ad esempio:

- **Foglio elettronico incorporato Excel compatibile**
- **Schedulazione grafica con diagrammi di Gantt**
- **Generatore di reports**
- **Data mining con drill down ad infiniti livelli**
- **Grafici bi e tridimensionali.**
- **Traduttore**
- **E-mail e corrispondenza**
- **Analisi del carico macchine**
- **Schedulatore della produzione**

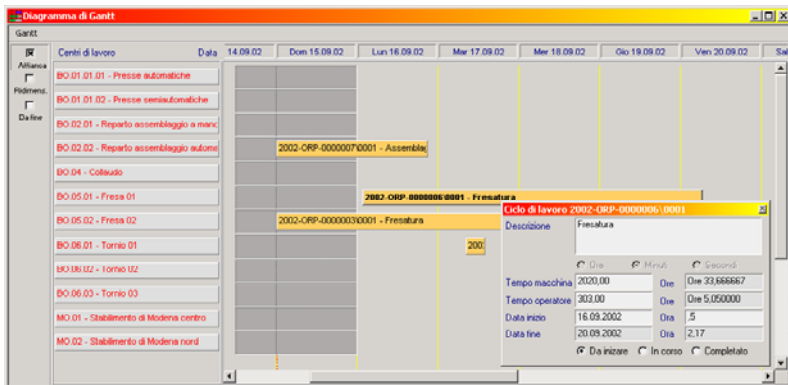
Questa impostazione porta molti vantaggi: ad esempio, l'e-mail incorporata ci permette di spedire direttamente i documenti (ordini, fatture, ecc.) generando un messaggio e-mail che include il documento, in formato PDF, come allegato, con completa interattività.

## Foglio elettronico



I fogli elettronici sono memorizzati direttamente nel database in formato binario. Una apposita gestione a formule permette di leggere i dati dall'archivio con estrema facilità, ed anche di aggiornare gli archivi direttamente dal foglio elettronico. Il foglio elettronico incorpora una gestione di grafici, che può essere utilizzata in alternativa a quella collegata al data mining.

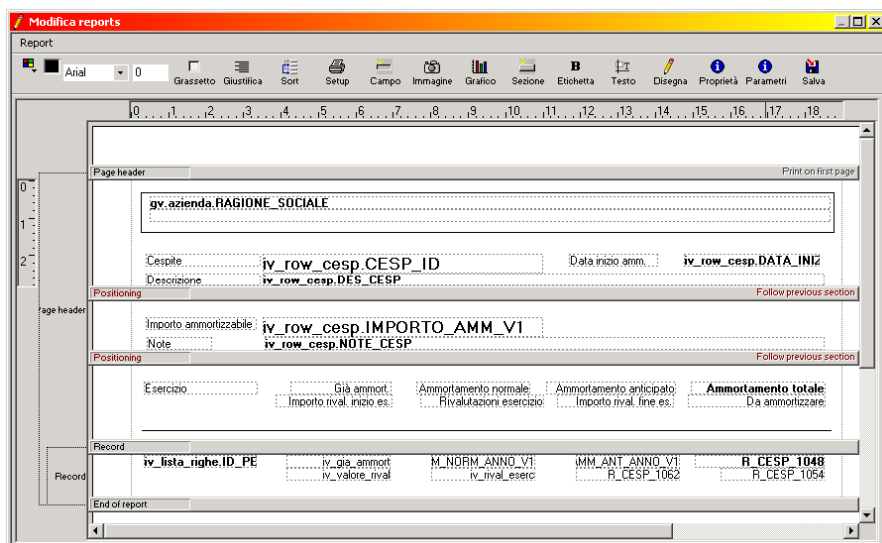
## Schedulazione grafica con diagrammi di Gantt



Il diagramma di Gantt incorporato permette di avere una visione grafica dei cicli di lavoro da pianificare: con un drag and drop possiamo spostare i cicli di lavoro su date ed ore diverse, o sui centri di lavoro ammessi in base ad una tabella

di compatibilità, può ricalcolare tempi e istantaneamente, e permette aggiornamenti istantanei dei cicli di lavoro modificati. Questo non sarebbe certo possibile interfacciandosi ad esempio a MS Project o altro.

## Generatore di reports



L'utente può generare i report desiderati, abbinandoli a filtri di ricerca che permettono di estrarre i dati desiderati. I filtri di ricerca possono richiedere un input di dati (es. date variabili); se l'input di dati fa riferimento ad una tabella (es. il codice di un agente) può essere programmata una lista di help.

Il generatore di reports può essere utilizzato anche per **modificare tutti i reports standard** del programma.

## Data mining

The screenshot shows a window titled 'Totale per categorie' with a table of sales data. The table has three columns: 'Codice categoria', 'Descrizione categoria', and 'Vendite'. The data rows are:

Codice categoria	Descrizione categoria	Vendite
Hw.02	Schede madri	417.555,00
MIN.01.01	Bulloneria	784.340,63
SE.02	Semilavorati generali	61.975,00
ST.02	Articoli in stampa normale	61.975,00
xxx	Altro	5.400,00
<b>Codice categoria</b>		<b>Vendite</b>
		<b>1.331.245,63</b>

At the bottom of the window, there are three radio buttons: 'Includi solo righe selezionate' (unchecked), 'Totali complessivi' (checked), and 'Totali delle righe selezionate' (unchecked).

Questo potente strumento permette di effettuare ogni tipo di **analisi sui dati aziendali**. Si possono definire delle query di interrogazione da abbinare a **filtri di ricerca** (vedi Reports). E' possibile definire dei limiti per la visualizzazione di righe separate; ad esempio categorie troppo piccole possono essere raggruppate sotto una unica voce. E' possibile fare comparazioni della stessa tabella con un secondo filtro di ricerca, duplicando una o più colonne sotto altro nome (ad esempio venduto anno comparato a venduto anno precedente)

La **gestione a formule** permette di parametrizzare ogni aspetto delle tabelle, dalle selezioni al titolo da visualizzare.

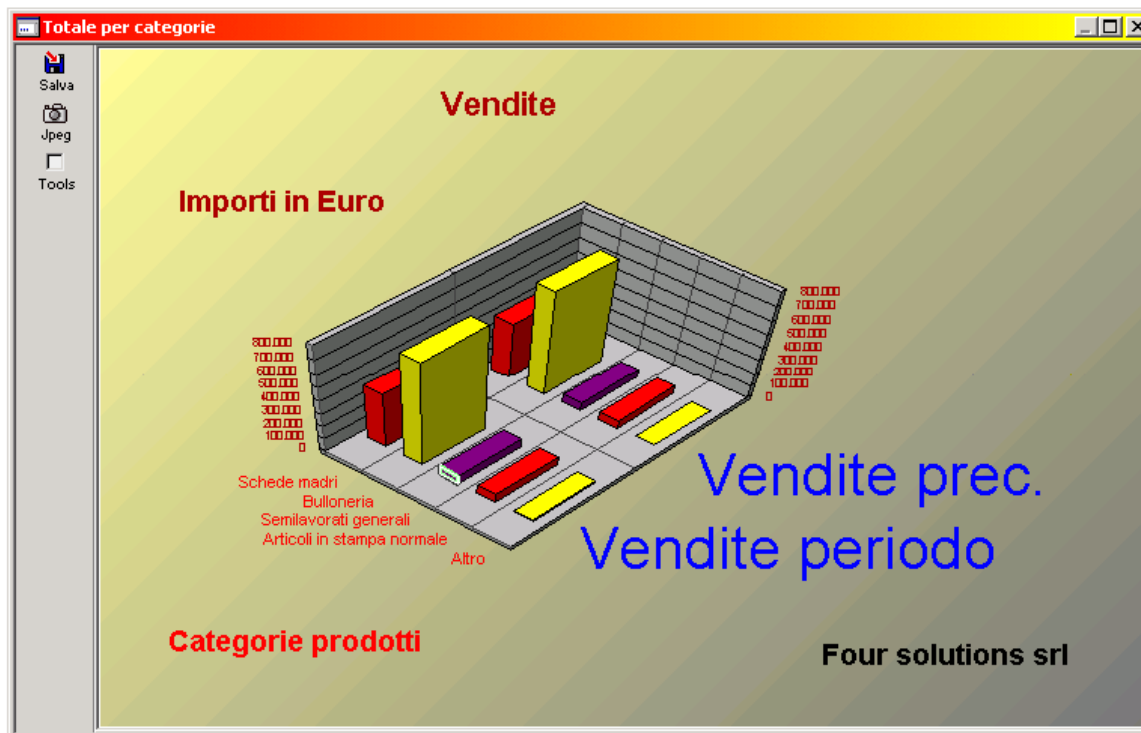
Sempre grazie alle formule, possiamo generare delle **tabelle formate da dati eterogenei**, e non dei semplici subtotali o accorpamenti. Ad esempio potremmo avere una tabella con in ogni riga la categorie di prodotti, il totale venduto del periodo ed il totale ordinato nello stesso periodo. Dato che questi vengono da due tabelle di database diverse, normalmente sarebbe impossibile averle. Utilizzando le formule, questo diventa possibile.

Ogni tabella può essere **abbinata ad un report**, che verrà utilizzato per le stampe.

Ogni tabella può **essere collegata ad altre tabelle in cascata** (drill down). Per esempio, possiamo avere una tabella di venduto per categorie di prodotto. Da una riga di categorie possiamo vedere l'esplosione per articoli, o per zone. Da ogni zona possiamo avere l'esplosione per clienti, ecc.

In questa esplosione possiamo definire un certo numero di **parametri da passare** alla seconda tabella; ad esempio nel caso sopra dal venduto per categorie se vogliamo esplodere una categoria passeremo 3 parametri : data iniziale, data finale e codice categoria. Questo automatizza completamente il drill down.

## Grafici



Il generatore di grafici lavora abbinato al data mining, permettendo la creazione automatica di grafici a partire dalle tabelle generate. E' possibile scegliere le colonne da inserire, che non debbono necessariamente essere contigue, le intestazioni, ecc. Interessante è il fatto che il numero di righe può essere variabile; se ad esempio in una analisi per categorie di prodotti si aggiunge una categoria, il grafico si modificherà aggiungendo un nuovo elemento.

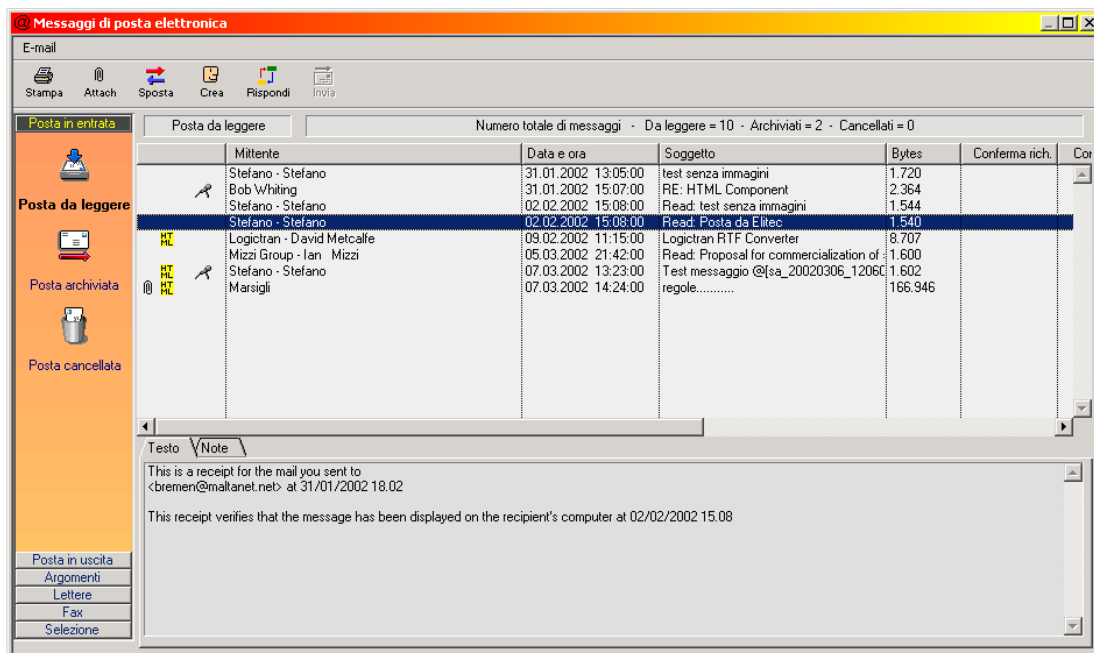
## Traduttore

Stringa	Traduzione
Bilancio esercizio	Balance sheet fiscal year
Cancella ordine	Delete order
Cancella riga	Delete line
Categorie	Categories
Codice contiene:	Code contains:
Codice,Descrizione	Code,Description
Descrizione contiene:	Description contains:
Descrizioni	Descriptions
Entra nella sezione	Go to section
Esegui invio ordine	Execute post order

Ogni **stampa del programma può essere tradotta nelle lingue desiderata**. Possono venire tradotti non solo le scritte fisse presenti nel report, ma anche le tabelle di riferimento. Se ad esempio vogliamo tradurre un bilancio, possiamo tradurre il piano dei conti; se vogliamo tradurre un inventario con subtotali per categorie, potremo inserire le descrizioni in lingua degli articoli e tradurre le categorie merceologiche.

Dato che **lo stesso termine può corrispondere in un'altra lingua a diversi termini** (es. l'italiano "fondo" può essere tradotto in inglese come "fund" o "bottom" a seconda del contesto), è possibile definire diverse tabelle di traduzione.

## E-mail

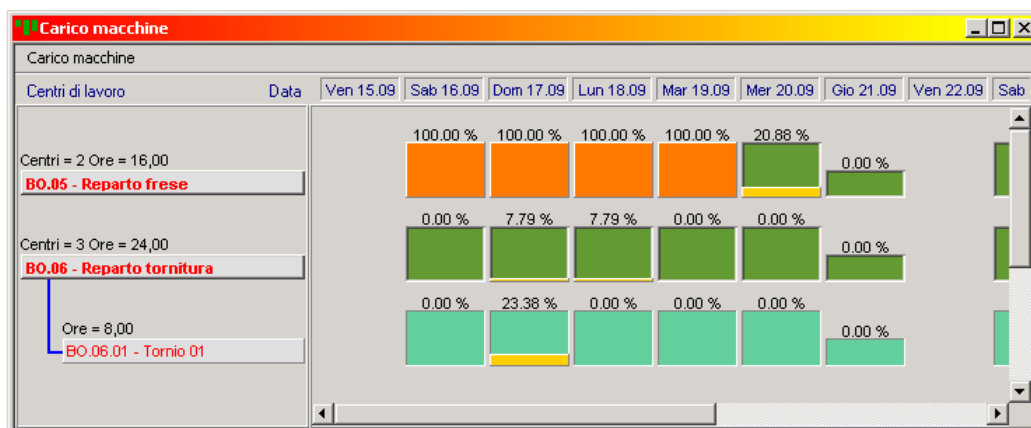


Target Cross incorpora un **completo programma di posta elettronica**.

I vantaggi di avere un programma di questo tipo incorporato nel gestionale, dato che esistono tanti prodotti per questo scopo, possono essere così sintetizzati:

- I messaggi di posta elettronica sono **memorizzati nell'archivio aziendale**: maggior sicurezza, backup automatico, visibili da tutti gli autorizzati.
- **L'archivio dei nominativi** coincide con quello aziendale: nessuna duplicazione o disallineamento.
- Messaggi **visibili senza scaricarli dal server di posta**: quelli indesiderati o dubbi possono venire cancellati senza scaricarli.
- **Messaggistica interna** in aggiunta all'e-mail, per altri utenti
- Possibilità di **ridirigere** un messaggio ad altro utente interno, senza e-mail
- Integrazione tra e-mail, fax e lettere
- Creazione automatica di **"discussioni"** che raggruppano più messaggi sullo stesso argomento
- **Invio di documenti** (ordini, fatture, ecc.) direttamente come allegato di e-mail in formato PDF.

## Analisi del carico macchine



Questa funzione permette di analizzare per ogni centro di lavoro il livello di carico previsto. I centri possono essere visti in maniera riassuntiva (reparto) o dettagliata. Con un doppio click su un giorno si vedono tutti i cicli previsti, che con un drag and drop possono venire spostati su altre date o centri di lavoro.

## Schedulatore della produzione

Parametri schedulazione

Cambio attrezzatura (ore): 36

Puntualità (giorni ritardo): 69

Ottimizzazione utilizzo (ore vuote): 35

Cambio stampo: 36

Cambio colore: 64

Cambio dimensioni: 50

Visualizza risultati durante il calcolo

Formula di rettifica

- Cambio colore
- Cambio dimensioni
- Ottimizzazione
- Cambio stampo

Cambio attrezzatura

Puntualità

OK Annulla

Con questo strumento i cicli di lavoro da eseguire possono essere **disposti secondo una strategia ottimizzata**, tenendo conto di vari parametri. Di questi tre sono standard:

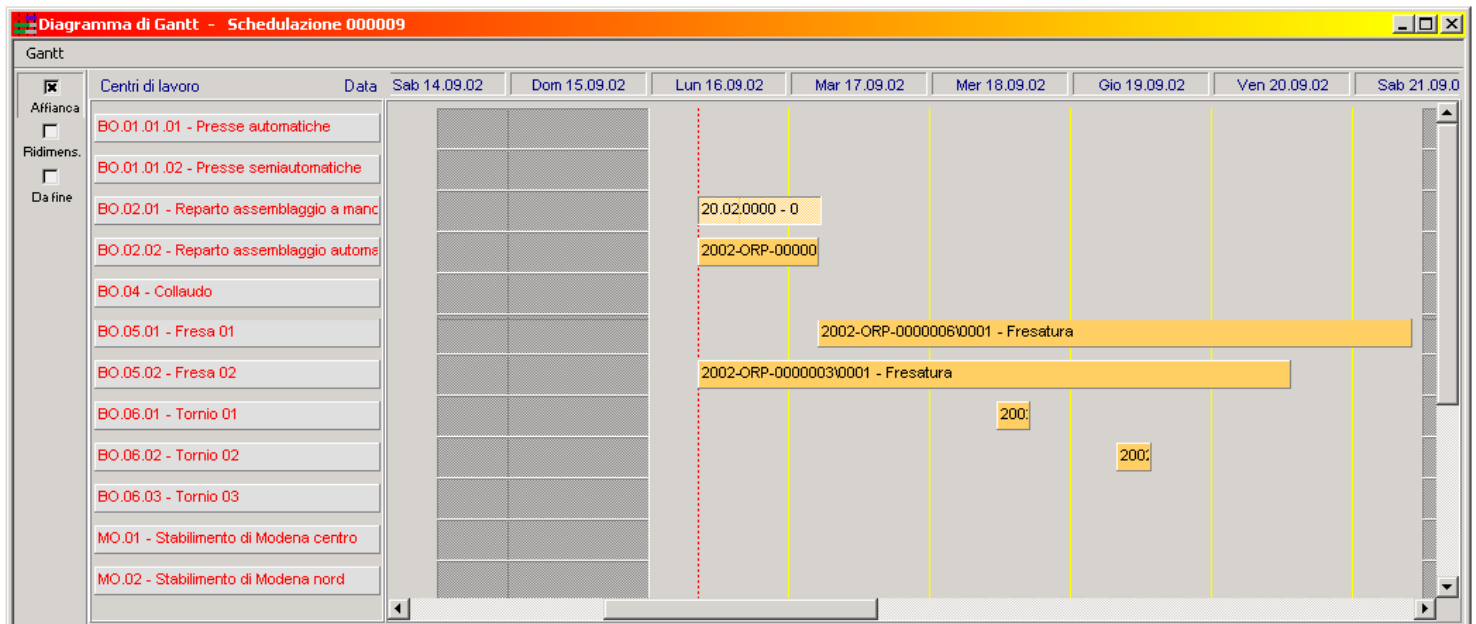
- **Cambio attrezzatura**
- **Puntualità**
- **Ottimizzazione utilizzo impianti**

A altri tre **sono definibili dall'Utente**.

Per ognuno di questi parametri può essere attribuito un peso variabile, da 0 a 100; il programma ottimizzerà la distribuzione dei cicli tenendo conto di questo peso.

L'importanza dei parametri può essere non solo lineare, ma variabile in base a formule; ad esempio, possiamo fare in modo che per un certo Cliente l'importanza della puntualità sia maggiore, o che per un centro di lavoro particolarmente costoso l'importanza dell'utilizzo massimale sia più grande.

Le **schedulazioni sono memorizzate**, e possiamo quindi avere varie ipotesi con parametri diversi.



Da ogni schedulazione possiamo **generare un diagramma di Gantt** che ce la mostra in forma grafica, per potere apportare a mano gli aggiustamenti eventualmente necessari.

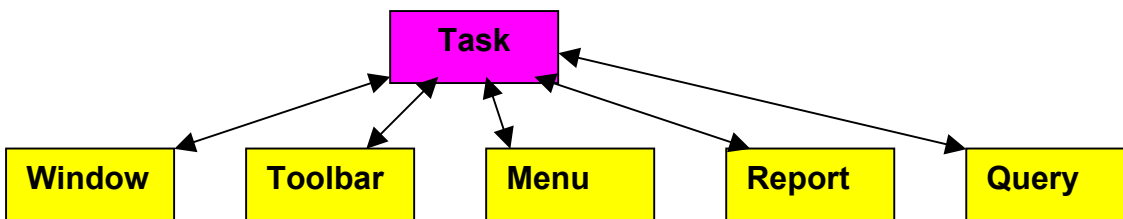
# Interfaccia grafica

## Gestione a task

Data la crescente quantità e complessità delle informazioni gestite, assume una crescente importanza il modo e l'efficienza con cui esse sono mostrate a video.

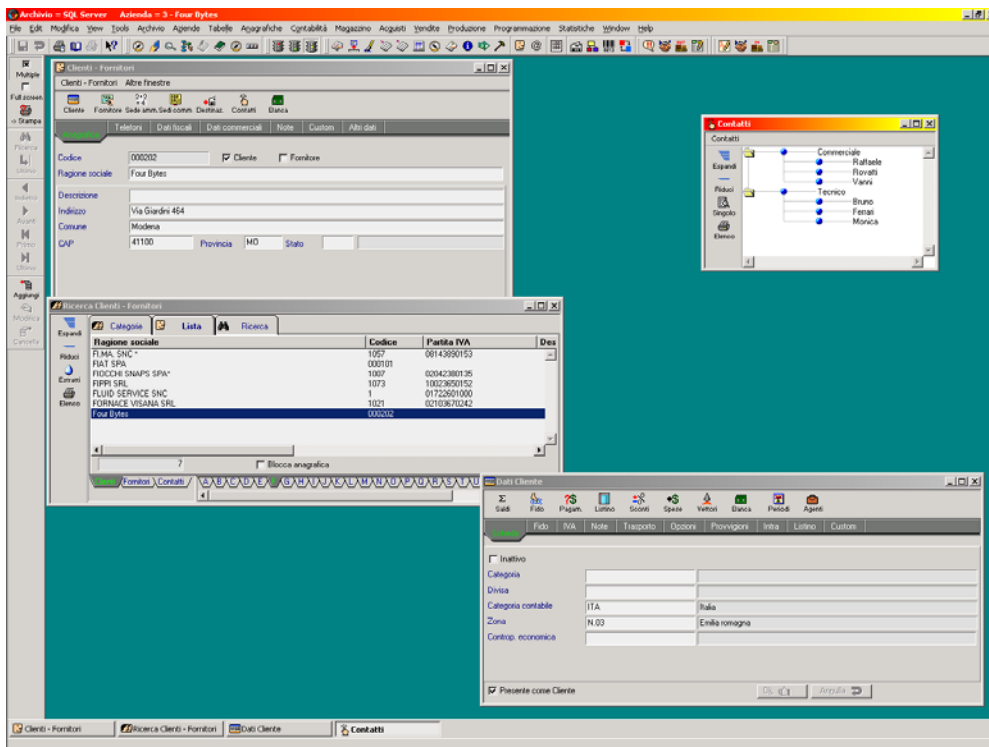
**Nella quasi totalità dei prodotti Windows, ogni ambiente operativo (ad esempio l'immissione di ordini) consiste di una sola finestra.** Questo avviene per l'impostazione di questi programmi, nei quali ogni finestra o form è un microcosmo chiuso, e peggiora notevolmente l'operatività, in quanto una unica complicatissima finestra deve contenere tutti i dati.

In Target Cross l'unità operativa è il task, che raggruppa una serie di oggetti (finestre, menu o altro) che hanno uno scopo comune.



Questo rende estremamente più facile personalizzare il programma, in **quanto i nuovi oggetti aggiunti in un task ne condividono le variabili.**

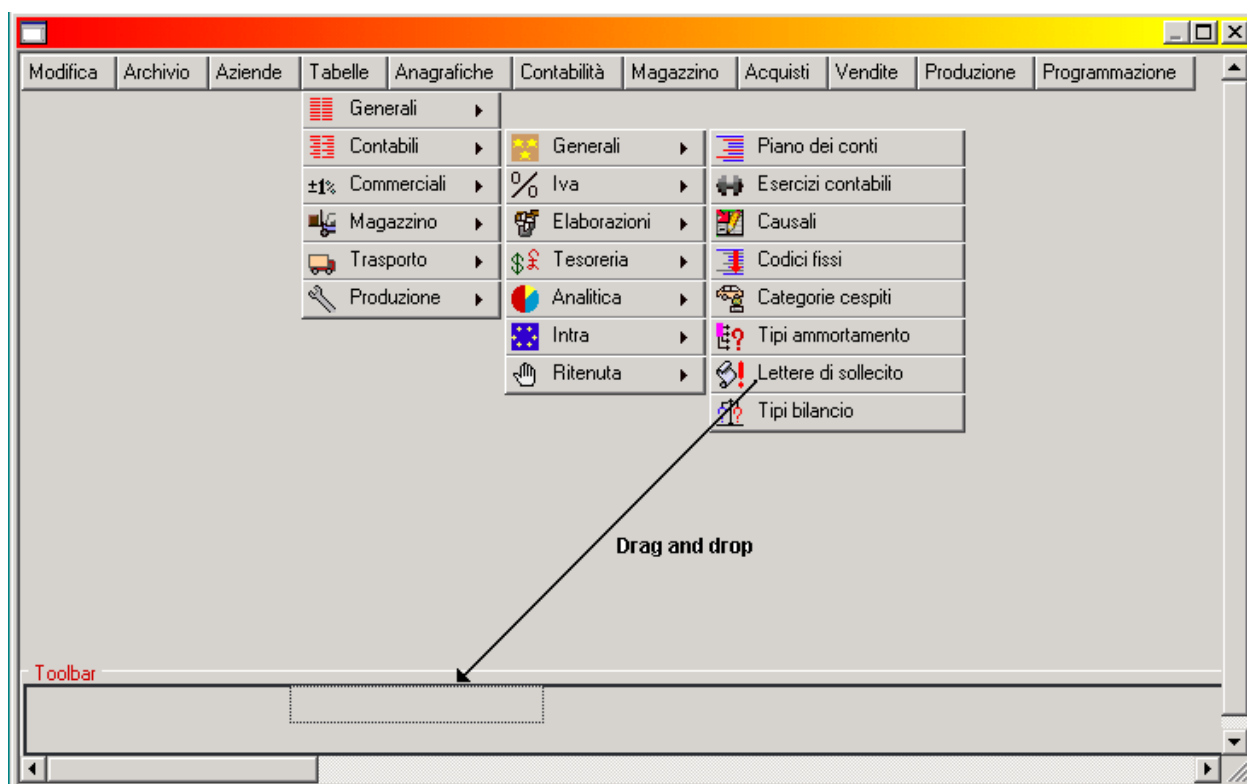
## Interfaccia multiwindow



Questo approccio ha permesso di sviluppare una intercaccia grafica avanzatissima, in cui ogni ambiente operativo o Task è composto da un numero variabile di finestre; queste possono essere aperte e chiuse quando necessario senza perdere le informazioni (che sono memorizzate a livello di Task), ridimensionate (memorizzando le dimensioni) e personalizzate a livello di aspetto (toolbar, menu, campi visibili, abilitati, ecc.)

## Toolbar personalizzabili

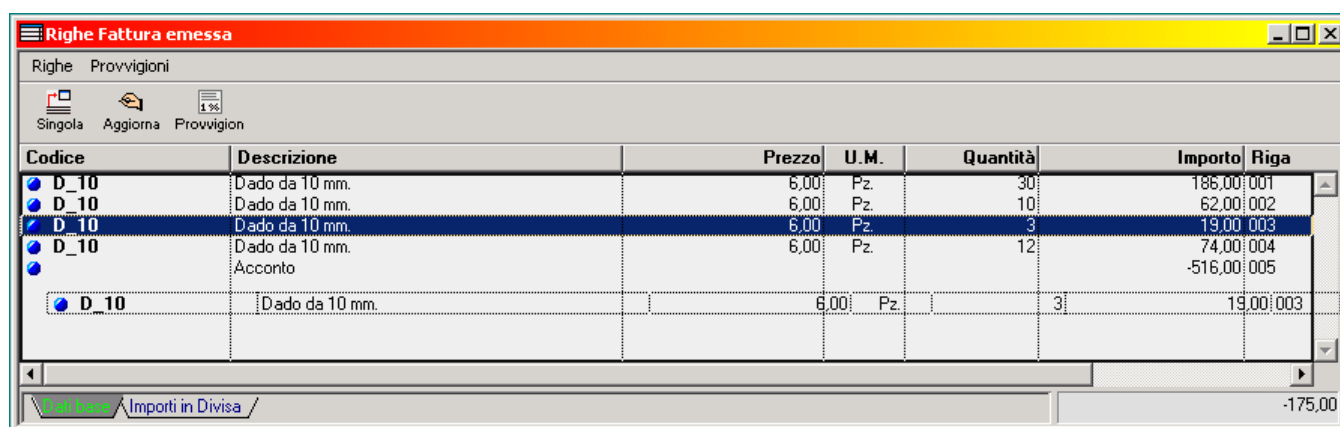
Target Cross utilizza sia i menu che le toolbars. Le toolbars sono viste come un sottoinsieme dei comandi di menu, con quelli più spesso utilizzati che sono duplicati come toolbar per comodità dell'utente.



Un semplice sistema di drag and drop delle voci di menu permette di costruirsi le proprie **toolbars personalizzate**, memorizzate separatamente su ogni posto di lavoro.

## Drag and drop

Il programma fa un uso estensivo del drag and drop per gestire informazioni in modo semplice e veloce.

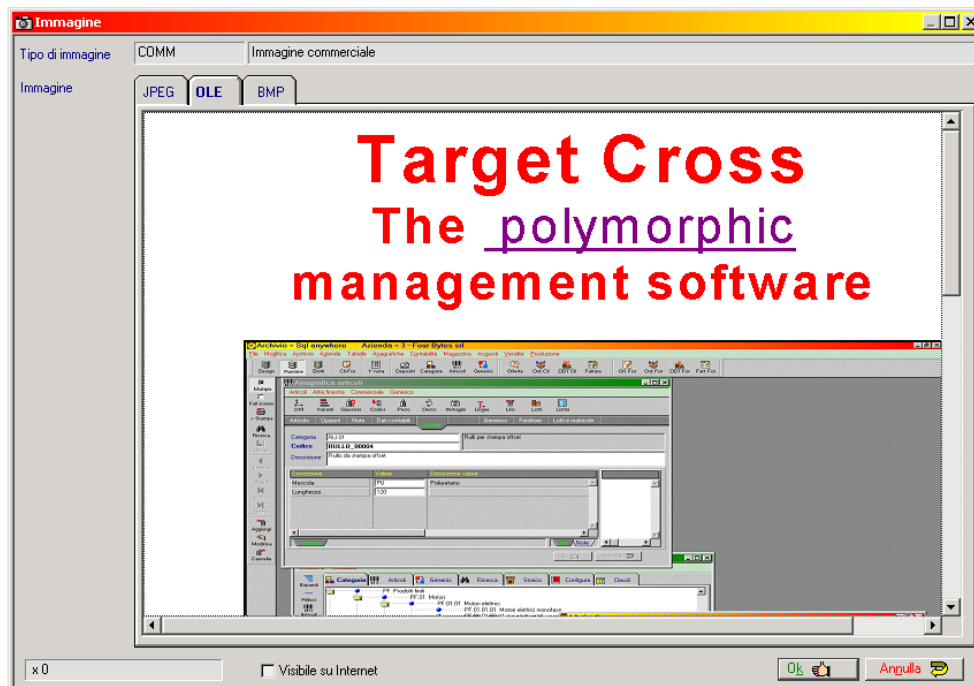


Con questa tecnica possiamo ad esempio "prendere" il campo che contiene il valore che ci interessa da una finestra e portarlo sulla finestra di una formula; possiamo spostare in su ed in giù le righe di un documento già compilato, per metterle nella posizione desiderata, ecc.

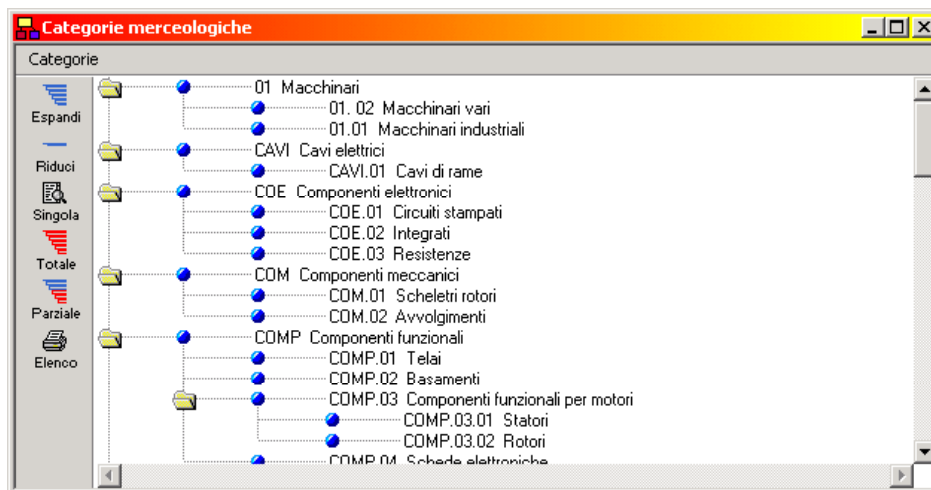
## Altre caratteristiche funzionali

### OLE automation.

L'utilizzo delle formule, unito al linguaggio di scripting interno, permette di creare interfacce per OLE automation, consentendo sofisticate interazioni con Office o altri OLE servers.



### Tabelle gerarchiche



In Target Cross, tutte le tabelle di una certa complessità sono organizzate gerarchicamente, con il tipo di struttura di solito riservata al piano dei conti. Le tabelle possono avere fino a 6 livelli, con rami di lunghezza diversa. Questo consente una maggiore flessibilità per l'utilizzo di queste tabelle e per le statistiche. Ad esempio utilizzando depositi gerarchici, possiamo avere inventari o MRP a qualunque livello di aggregazione.

# Un innovativo approccio ad Internet

Il settore che in questi anni ha visto il maggior numero di soluzioni affacciarsi sul mercato è stato senz'altro quello di Internet.

Il nostro prodotto si propone come qualcosa di diverso, stabilendo nuovi parametri per quanto riguarda la velocità operativa, fattore sempre critico anche con i nuovi collegamenti, l'interfaccia grafica e la completezza di funzioni.

## Internet Web Client

The screenshot shows a Microsoft Internet Explorer window titled "Remote Form Test - RF\_LOGON - Microsoft Internet Explorer fornito da Infostrada LIBERO". The address bar shows the URL "D:\O532\html\RF\_LOGON.htm". The main content area is divided into two sections. The top section, titled "Ricerca", "Prezzi", and "Disponibilità", shows a tree view of product categories on the left and a table of products on the right. The bottom section, titled "Lista ordinata", shows a table of the current order with columns for "Codice", "Descrizione", "Prezzo lordo", "Sconti", "Prezzo netto", "U.M.", "Quantità", and "Importo".

Codice	Descrizione	Prezzo lordo	Sconti	Prezzo netto	U.M.	Quantità	Importo
1 MOT	Motore monocilindrico	234,76	10+7+4+3	182,97		1	182,97

At the bottom of the order list, there are buttons for "Aggiungi ad ordine" and "Cancella riga ordine", a "Quantità" input field with the value "1", and a "Totale ordine" field with the value "182,97".

Grazie ad un plug-in di 400 Kbytes installabile automaticamente, il programma può funzionare attraverso Internet **scaricando i form ed i componenti necessari sul computer Client**, la prima volta che li eseguiamo.

Da questo momento in poi, **non si scarica più nulla dalla rete** eccetto naturalmente le informazioni visualizzate; se ad esempio carichiamo una lista di articoli, solo la lista stessa in formato carattere viaggia su internet; la visualizzazione grafica avviene localmente.

Il risultato è una **velocità operativa semplicemente incredibile**, che deve essere constatata di persona e paragonata ai prodotti concorrenti per essere pienamente apprezzata.

Il programma inoltre presenta una interfaccia decisamente completa e sofisticata, che incrementa ulteriormente la produttività dell'utilizzatore.

# La tecnologia spreadsheet

La maggior parte di coloro che utilizzano un computer usa anche un foglio elettronico. Questo tipo di software infatti, si adatta ad una vasta gamma di necessità, dato che può essere in un certo senso “programmato” inserendo le formule che ne determinano la logica di funzionamento

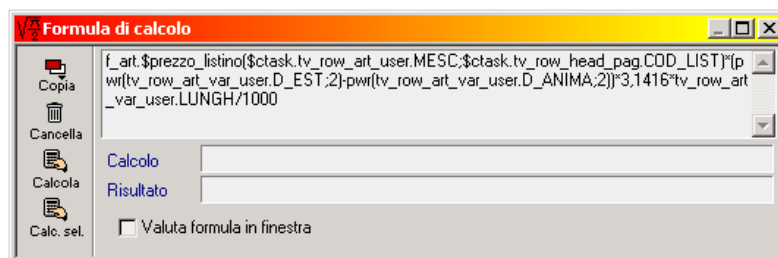
## Massima flessibilità del programma

**Target Cross introduce nel gestionale lo stesso grado di flessibilità che siamo abituati a trovare nel foglio elettronico.**

Un insieme di motori di parametrizzazione e di formule permettono di astrarre le modifiche apportate dal programma di base, così come il foglio elettronico creato è del tutto separato dal programma su cui gira.

Il concetto di formula, mutuato dal foglio elettronico, è stato poi allargato agli array di formule, che ne semplificano la creazione e la manutenzione, ad al linguaggio di scripting.

**Il risultato è un gestionale che si adatta alla realtà in cui deve operare meglio di qualunque altro.**



La tecnologia spreadsheet è composta di quattro funzioni fondamentali:

- Formula manager
- Function manager
- Array di formule
- Repository di formule

## Formula manager

Le formule consentono di sostituire, come in un foglio elettronico, un valore numerico, alfabetico o di data con una formula che sarà risolta ogni volta che abbiamo bisogno di questo valore. Ad esempio in una scheda articolo possiamo avere la formula del calcolo del consumo di lamiera in uno stampaggio, che può fare riferimento ad una **qualunque delle variabili utilizzate**.

(Incidentalmente, questo caso illustra i limiti del sistema di “forms” a tenuta stagna: le variabili di quale form saranno utilizzate?)

Ad esempio, la formula potrebbe essere del tipo **consumo = superficie unitaria \* (quantità + sfrido)**. In questo caso, ogni volta che emettiamo un ordine di produzione le quantità vengono ricalcolate in base alla formula.

In tutte le tabelle dove vengono effettuati dei calcoli o definiti dei valori, oltre ai valori numerici abbiamo delle formule che ci permettono di esprimere i valori stessi in forma dinamica.

Alcuni esempi:

- **Formula del calcolo delle provvigioni** (e non semplicemente percentuale di provvigioni)
- **Formula di selezione del listino** (possiamo scegliere listini diversi a seconda del contesto)
- **Formula del prezzo di vendita**, in base al contesto. (Es. prezzo di vendita dato dal costo aggiornato della materia prima + un fisso per la lavorazione).
- **Formula della scorta minima e del punto di riordino**, in base ad esempio al consumo medio.

- **Formula di calcolo del valore statistico INTRA** (per esempio in base alle zone di trasporto ed al tipo di trasporto).

Le formule di Target Cross sono del tutto simili a quelle presenti in un foglio elettronico, anche se lavorando su una base dati sono ovviamente ottimizzate per la loro funzione.

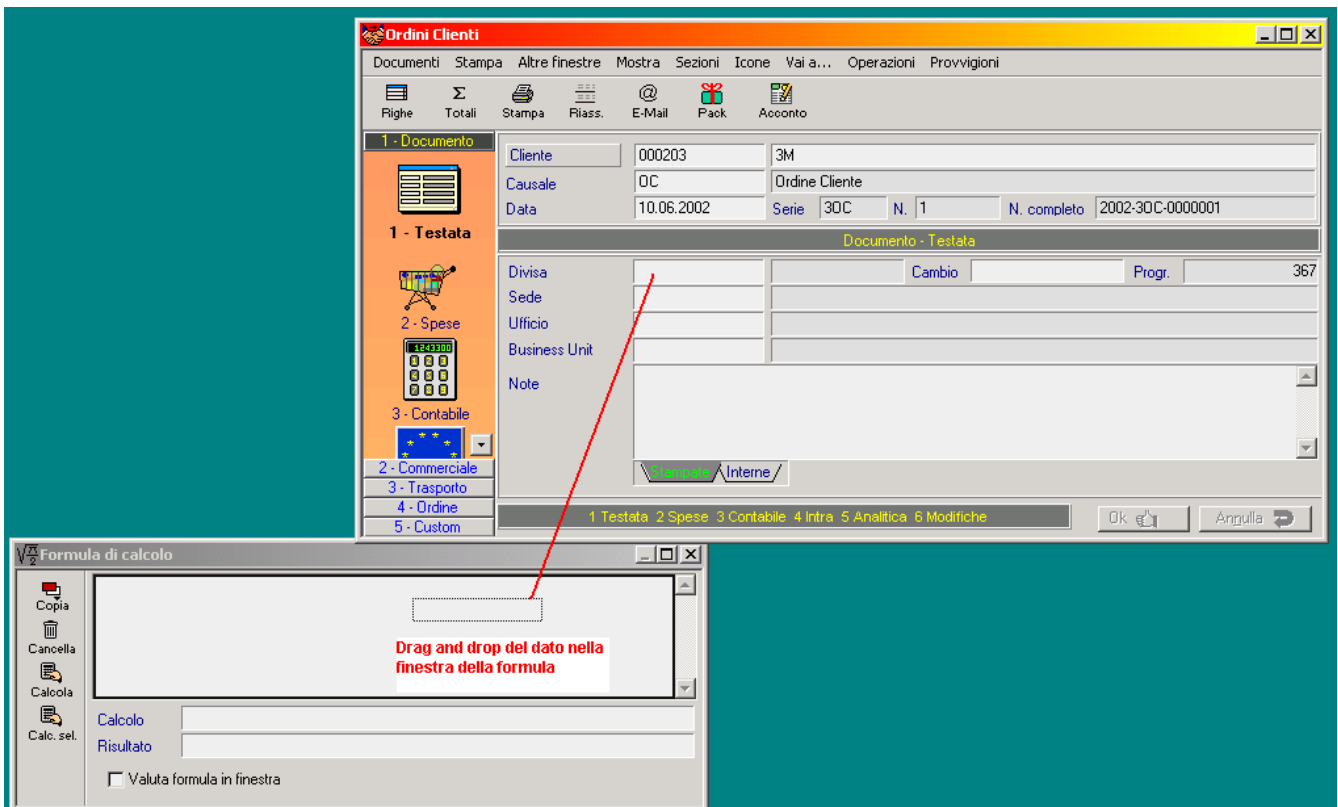
Una caratteristica molto interessante è la possibilità di inserire funzioni SQL direttamente nelle formule. Ad esempio una formula potrebbe calcolare il costo di un oggetto d'oro moltiplicando il peso dell'articolo, letto nella sua scheda, per il prezzo dell'oro del giorno, letto da un apposito archivio, e sommando ad esso il costo della lavorazione espresso in ore di personale specializzato.

**Il meccanismo delle formule è presente, in forma embrionale, anche su alcuni prodotti concorrenti. Le differenze tra queste implementazioni e Target Cross sono molte e profonde; ad esempio:**

- Le formule sono **presenti nel programma ovunque ci sia un valore da calcolare**, e non sono limitate ad alcune sezioni ben precise (es. calcolo quantità in distinta)
- Le formule possono essere **non solo di tipo algebrico, ma anche logiche**, quindi effettuare delle scelte (es. più pagamenti sul Cliente con formula di selezione)
- Le formule possono **fare riferimento non solo ad alcuni valori previsti, ma a tutti i valori attuali** delle varie tabelle e variabili.
- Le formule **possono fare riferimento ai valori dei record presenti, ma anche a quelli di altri records**. Per esempio, possiamo calcolare il prezzo di vendita di un articolo in oro in base al costo di un articolo ORO, anche se quest'ultimo non è l'articolo su cui stiamo lavorando, tramite la possibilità di inserire interrogazioni SQL direttamente nelle formule.

## Editor di formule

La creazione delle formule avviene per mezzo di un editor di formule. Nelle formula possiamo fare riferimento a tutti i dati presenti in memoria, oppure a valori memorizzati nel database. Per accedere ai primi, possiamo semplicemente prendere il dato che ci interessa con drag and drop dalla finestra che lo rappresenta.

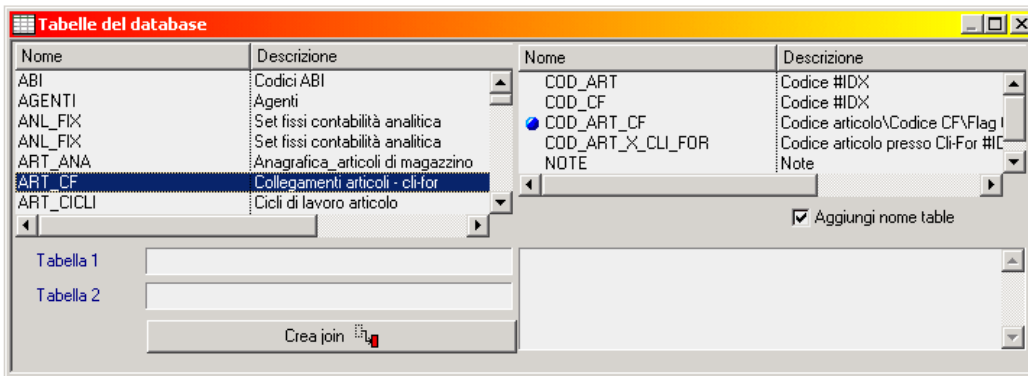


E' anche possibile accedere ai dati in altro modo, nei casi in cui questo dato non sia rappresentato a video.

Apposite funzioni presenti nel programma ci permettono di inserire nelle formule delle vere e proprie istruzioni di lettura dei dati tramite interrogazioni SQL. Possiamo quindi inserire in una formula una funzione che tradotta in chiaro suona in questo modo: "peso dell'articolo di codice ABC" oppure "cambio alla data del documento della divisa del documento", oppure ancora "totale venduto della categoria POMPE nella provincia di MILANO", ecc.

## Struttura del database

Per accedere alle tabelle del database, abbiamo a disposizione la struttura con tabelle e campi.



L'utilizzo delle formule è un metodo efficace per personalizzare il comportamento del programma senza variazioni; infatti **la funzione è memorizzata in un oggetto a parte, mentre la formula sta nei dati**, quindi il programma non viene toccato.

**Il formula manager già da solo ci trasforma il gestionale in un foglio elettronico, ma noi non ci siamo accontentati di questo.**

**In Target Cross sono presenti altre funzioni correlate che nemmeno i più moderni fogli elettronici possiedono e che ne estendono notevolmente le potenzialità**

## Function manager

In Target Cross, oltre ad utilizzare le funzioni esistenti all'interno delle formule, possiamo creare le nostre proprie funzioni, che saranno memorizzate in un apposito repository e potranno essere utilizzate come funzioni standard. Per esempio, l'esempio illustrato sopra potrebbe venire convertito in una funzione **\$prezzo\_articolo\_oro** (le funzioni debbono iniziare con il carattere "\$") che accetta come parametri il codice articolo e la data a cui calcolare il prezzo, e svolge tutto il lavoro in background.

L'esempio di cui sopra potrebbe quindi diventare da così

```
((f_sql.$select_num("SELECT PESO_ART FROM ART_PESO WHERE  
COD_ART=@[tv_row_art.COD_ART]"))*(f_sql.$select_num("SELECT VALORE FROM VALORI_VAR WHERE  
COD_VAR='ORO' AND DATA_INIZ<=@[#D] AND DATA_FINE>=@[#D] ")))+(f_sql.$select_num("SELECT  
COSTO_STANDARD FROM CLASSI_PERS WHERE COD_CP='PS'")*3.5)
```

a così

```
f_sql.$prezzo_articolo_oro(tv_row_art.COD_ART]
```

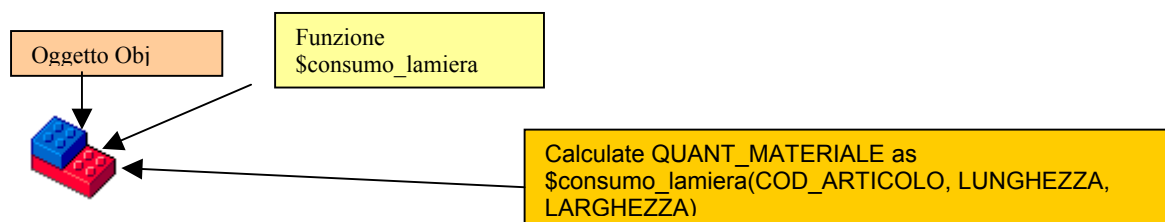
Questa impostazione presenta un triplice vantaggio:

- **Il meccanismo della formula è centralizzato**; se vogliamo modificarlo, basta modificare una sola volta la funzione corrispondente.
- **La funzione può essere implementata una sola volta** dal rivenditore o comunque da personale specializzato, ed il suo uso, molto più semplice, può essere affidato a persone molto meno preparate.
- **E' possibile creare funzioni per ogni possibile scopo**. In realtà la funzione è una procedura che viene eseguita risolvendo la formula, e che può essere di lunghezza e complicazione illimitata. Questa possibilità rende il motore di personalizzazione di Target Cross ben più completo e funzionale, in quanto non vi sono limiti a quello che può essere programmato con una funzione.

Possiamo inoltre avere la necessità di effettuare dei calcoli **talmente complessi da non essere esprimibili con una semplice formula.**

Supponiamo di avere una procedura più complessa, che tenga conto di quanti pezzi si possono affiancare in un rotolo di lamiera, e magari ci scelga la larghezza più adatta.

In questo caso potremo creare ad esempio una funzione personalizzata **\$consumo\_lamiera**, che definiremo come una procedura che accetta in input le variabili necessarie e restituisce in output la quantità corretta.

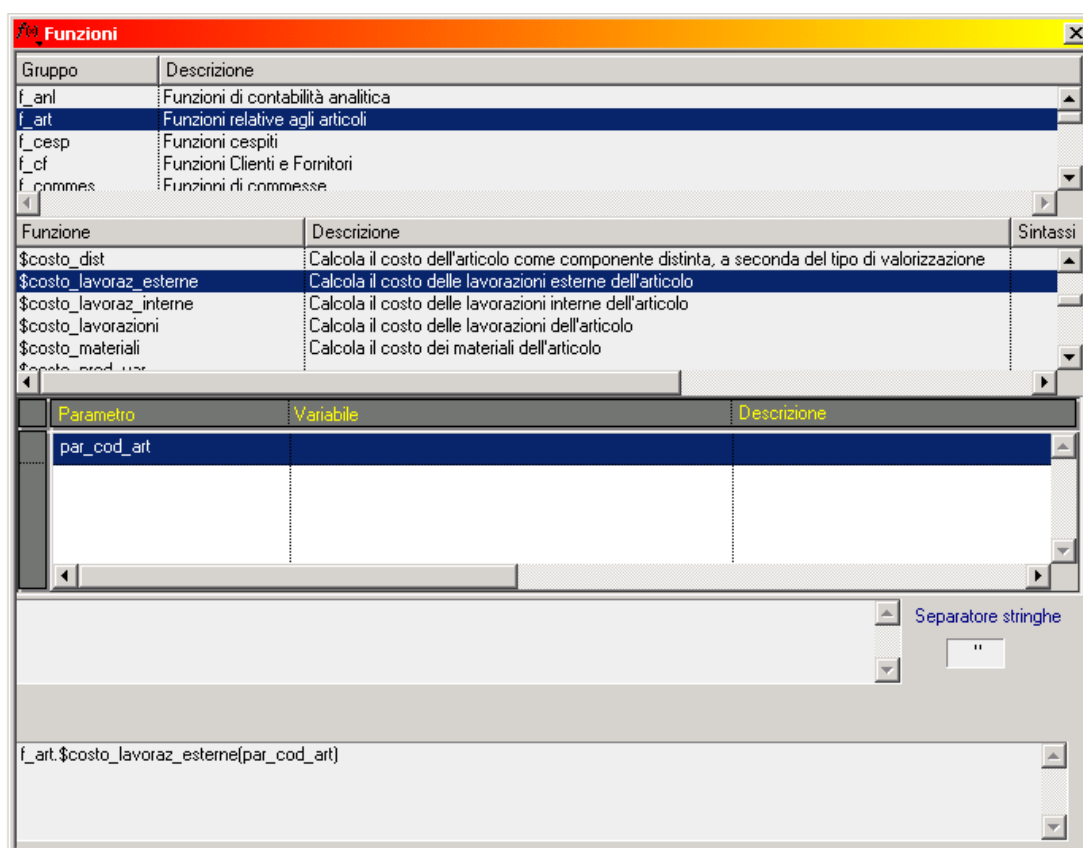


Questo è reso possibile utilizzando le classi di tipo **Object**: in queste classi possiamo creare dei metodi da inserire in formule, che entreranno a fare parte delle formule standard del linguaggio di programmazione.

In questo caso nella nostra formula inseriremo

**QUANT\_MATERIALE = \$consumo\_lamiera (COD\_ARTICOLO, LUNGHEZZA, LARGHEZZA, ecc.)**

e valutando questa formula il programma eseguirà il metodo **\$consumo\_lamiera**. Questo metodo può essere complesso a piacere, e può anche essere interattivo, ovvero chiedere degli input supplementari o proporre delle scelte.



**Le funzioni vengono memorizzate in un repository**, che consente un facile utilizzo della stesse. Infatti, sono organizzate in gruppi, ognuno dei quali contiene un certo numero di formule.

**Cliccando si una di esse, vengono mostrati i parametri necessari e la relativa descrizione.** In ogni riga di parametro potremo inserire il valore corrispondente o la variabile che lo contiene. Per inserire le variabili o i campi delle tabelle vale quanto esposto per le formule. La funzione completa viene quindi creata nella riga in basso e copiata negli appunti, per potere essere utilizzata dove desiderato.

## Array di formule

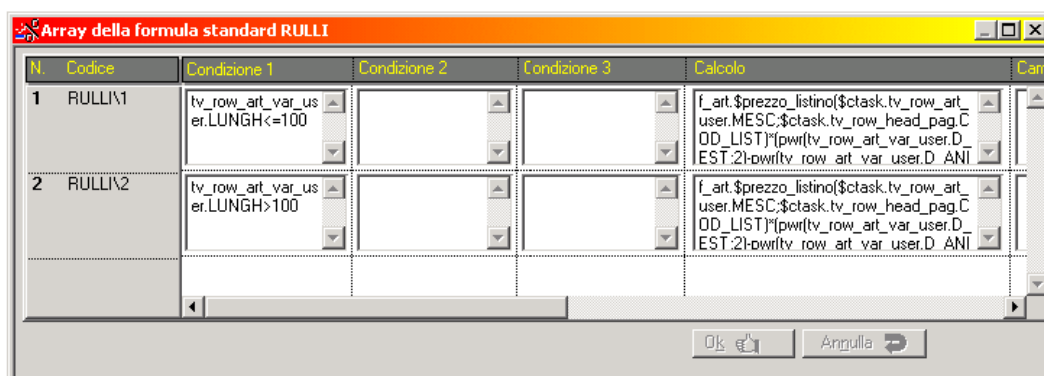
Chi ha utilizzato in un foglio elettronico **le formule con scelte multiple (il SE() di Excel)** avrà verificato di persona come facilmente si giunga a formule molto complesse e di difficile lettura.

**Gli array di formule di Target Cross risolvono questo problema.**

Un array di formule è costituito da un insieme di righe, ciascuna composta di due campi, condizione e formula. La prima riga la cui condizione è vera viene eseguita. Una riga senza condizione è sempre vera. La condizione è poi divisa su tre colonne, rappresentanti condizioni di primo, secondo e terzo livello, per una maggiore leggibilità.

In questo modo possiamo realizzare delle **strutture di controllo** del tipo

**If [condizione] calcola [formula]**  
**Else if [condizione] calcola [formula]**  
**Else [condizione] calcola [formula]**



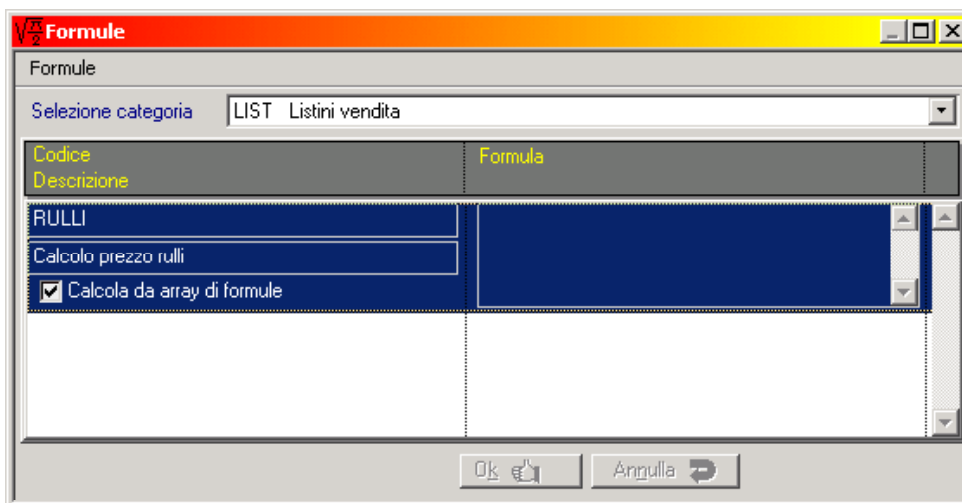
N.	Codice	Condizione 1	Condizione 2	Condizione 3	Calcolo	Can
1	RULLI\1	tv_row_art_var_user.LUNGH<=100			f_art.\$prezzo_listino(\$ctask.tv_row_art_user.MESC;\$ctask.tv_row_head_pag.COD_LIST)*(pwr(tv_row_art_var_user.DEST:2)-pwr(tv_row_art_var_user.DANI	
2	RULLI\2	tv_row_art_var_user.LUNGH>100			f_art.\$prezzo_listino(\$ctask.tv_row_art_user.MESC;\$ctask.tv_row_head_pag.COD_LIST)*(pwr(tv_row_art_var_user.DEST:2)-pwr(tv_row_art_var_user.DANI	

che semplificano notevolmente la creazione ed il mantenimento delle formule. Le condizioni sono tre diverse per ogni riga, che consentono quindi di creare un cubo di condizioni a tre dimensioni.

## Repository di formule

**Le formule o gli array di formula** possono essere memorizzati in un **repository o archivio centralizzato delle formule**.

Questo permette di scrivere la formula (o l'array di formule) una sola volta, e poterlo poi utilizzare in tutto il programma, semplicemente con il nome attribuitogli.



Codice	Formula
RULLI	
Calcolo prezzo rulli	
<input checked="" type="checkbox"/> Calcola da array di formule	

La logica è simile a quella delle funzioni, ma mentre per scrivere una funzione occorre un lavoro di programmazione, per creare una formula o un'array nel repository non è necessario entrare all'interno del programma.

Il vantaggio è che modificando la formula una sola volta, essa sarà modificata in tutte le parti del programma che ne fanno uso.