

UNIVERSITA' DEGLI STUDI DI PADOVA

Facoltà di Scienze Statistiche

Corso di laurea triennale in
Statistica e tecnologie informatiche

TESI DI LAUREA

Un caso di studio per il reverse engineering: lo SCADA per la vetreria.

RELATORE: PROF. MAURO MIGLIARDI

LAUREANDA: ERIKA GIOACHIN

Anno accademico 2010/2011

*Il calcolatore è straordinariamente veloce,
accurato e stupido. L'uomo è
incredibilmente lento, impreciso e creativo.
L'insieme dei due costituisce una forza
incalcolabile.*
Albert Einstein

Indice generale

1 – Introduzione.....	4
2 - Ingegnerizzazione e funzionamento dell'impianto.....	6
2.1 - Perché SCADA.....	6
2.2 - Descrizione del funzionamento dell'impianto.....	7
3 - LE TECNOLOGIE USATE.....	12
3.1 – Lo SCADA.....	12
3.1.1 – Analisi del processo controllato.....	13
3.1.2 - Realtime.....	13
3.1.3 – Alta affidabilità.....	14
3.1.4 – Alta disponibilità.....	14
3.1.5 - HMI.....	14
3.1.6 – Funzioni.....	15
3.1.7 – Presentazione dei dati.....	16
3.1.8 – Gestione dei comandi operatore.....	17
3.1.9 – Controllo gestione comandi.....	18
3.1.10 – Eventi e Allarmi.....	18
3.1.11 – Archiviazione dati.....	20
3.2 – Il PLC.....	22
3.2.1 – Specifiche hardware.....	22
3.2.2 – Scansione ciclica.....	24
3.2.3 – Linguaggi di programmazione.....	25
3.2.4 – Linguaggi grafici.....	26
3.2.5 – Linguaggi testuali.....	27
4 - StorePRO.....	28
4.1 - Struttura.....	28
4.1.1 - I Sinottici.....	29
4.1.2 – Dichiarazione variabili.....	37
4.1.3 – Protocollo e comunicazione.....	39
4.1.4 - Database.....	40

5 – Miglioramenti.....	45
5.1 – Problemi del progetto.....	45
5.2 – Richieste dal cliente.....	46
5.3 – Richieste dal mercato.....	48
CONCLUSIONI.....	50
Bibliografia.....	51

1 – Introduzione

In questo elaborato verrà analizzata una parte del programma che fa parte del processo di automazione di una macchina industriale, nella fattispecie una macchina per il magazzinaggio del vetro piano.

Lo scopo principale della tesi è quello di analizzare tutte le parti che compongono il programma, darne una valutazione descrivendone pregi e difetti, concludendo poi con delle proposte di miglioramento per rendere il programma più flessibile a possibili evoluzioni.

Questo tipo di analisi, effettuata al fine di migliorare un artefatto esistente, si definisce *reverse engineering*.

"Reverse Engineering, in ogni campo dell'ingegneria, significa semplicemente analizzare una macchina che non hai costruito, per apprendere come è stata fatta. È una pratica legittima e accettata in tutti i campi dell'ingegneria".

- Richard Stallman. [1]

Il problema del reverse engineering di un'applicazione esistente, comune anche nel campo del software engineering, è una realtà molto importante e presente.

In genere le pratiche di reversing vengono utilizzate quando una software-house mantiene un codice proprietario per i suoi prodotti. A quel punto il reverser analizza i programmi e cerca di capirne il funzionamento senza avere a disposizione un sorgente.

Il reverse engineering viene comunemente utilizzato per evitare di violare i diritti d'autore sulla funzionalità desiderata, e può venire usato per cercare di eludere un brevetto, sebbene ciò sia un po' rischioso: i brevetti si applicano alle funzionalità, non a loro specifiche implementazioni.

Questo tipo di analisi è accettata solo ed esclusivamente a scopo di interoperabilità con altri sistemi ed è considerato un atto pienamente lecito ai sensi dell'art. 64 della legge 633 del 22 aprile 1941, come modificata dall'art. 5 del D. Lgs. 518/1992 e vale sia in

senso "leggero" qualora egli compia tali atti durante operazioni di caricamento, visualizzazione, esecuzione, trasmissione o memorizzazione del programma che egli ha il diritto di eseguire; che in senso di decompilazione vera e propria, ma solo al fine di permettere l'interoperabilità del software con altri programmi. L'accezione di software è estesa per analogia a concetti informatici quali il formato di un file o la struttura interna di un protocollo.

Il lavoro che verrà esaminato in questo elaborato è frutto del lavoro svolto all'interno della ditta Gioachin.

L'azienda è specializzata nella soluzione di problematiche nel campo dell'Automazione Industriale ed è in grado di fornire la progettazione dello schema elettrico, la realizzazione del quadro elettrico, i programmi, le tarature ed il collaudo finale della macchina. In poche parole, la completa automazione di macchine o impianti.

I settori in cui ha più esperienza vanno da quello alimentare a quello degli ascensori, dalla movimentazione delle macchine per il magazzinaggio del vetro piano, agli impianti teatrali. (*source www.gioachin.it*)

Nel caso di questa tesi di laurea andrò ad esaminare una parte del lavoro che è stato svolto su un impianto per la movimentazione del vetro piano automatizzato dalla ditta Gioachin.

Questa tipologia di impianto viene installata nei magazzini delle vetrerie e serve prima di tutto per classificare i tipi di lastre che un'azienda può lavorare, serve inoltre per prelevare le lastre dallo stesso magazzino per portarle poi al banco di taglio.

2 - Ingegnerizzazione e funzionamento dell'impianto.

2.1 - Perché SCADA

Prima di iniziare l'analisi del programma, facciamo un passo indietro per spiegare il processo di analisi ed ingegnerizzazione dei requisiti.

E' doveroso premettere che per impianti di questo tipo alcune tecnologie, come il plc, sono strumenti che si possono definire “scontati”, cioè che verranno sicuramente utilizzati per il progetto perchè sono stati creati proprio per l'automazione industriale. L'unica scelta che resta da prendere riguarda la capacità computazionale del plc, cioè quante informazioni è in grado di elaborare, quanta memoria può contenere. Questo aspetto varia da impianto ad impianto a seconda delle dimensioni della macchina.

Per prima cosa bisogna analizzare quanti dati in input e output devono essere scambiate tra la macchina e il plc e dal plc alla macchina. Nel primo caso (macchina a plc), sono stati installati diversi dispositivi per il monitoraggio della macchina per fare in modo che ogni situazione potenzialmente pericolosa (e si consideri che nel caso di una macchina industriale il pericolo puo' essere anche molto forte) generi un opportuno allarme in modo da rendere consapevole l'utente finale.

Nel caso invece della comunicazione da plc a macchina, questo deve essere in grado di fornire alla macchina tutte le indicazioni per il corretto prelievo dal o ritorno al magazzino.

Per raggiungere tale scopo sarebbe sufficiente un dispositivo HMI (Human-Machine Interface), cioè un pannello operatore che permetta di inserire tutti i dati necessari per movimentare la macchina. L'HMI però ha dei limiti, infatti non può memorizzare grandi quantità di dati come anche il plc che pur avendo grandi capacità, non è in grado di gestire grandi database. Per questo è stato necessario trovare una soluzione per unire l'HMI con un sistema che fosse in grado di gestire una grande mole di dati.

La scelta è caduta su un dispositivo di controllo, supervisione e acquisizione dati anche chiamato SCADA (*Supervisory Control And Data Acquisition*)[2]. Questo strumento si è rivelato la scelta vincente perchè unisce la possibilità di colloquiare con l'operatore

con la gestione di un database e la capacità di memorizzare grandi quantità di dati con un sistema standard (ODBC Vedi cap.4).

Il linguaggio di base per la programmazione di uno SCADA è il Visual Basic e può essere utilizzato all'interno dello stesso per creare funzioni, subroutine, accessi ai database così da personalizzare ancora di più il prodotto finale. Tutte queste caratteristiche sono un valore aggiunto per lo SCADA perchè grazie ad esse in futuro sarà molto più facile compiere modifiche.

Questa tipologia di strumento per il controllo, supervisione e acquisizione dati ha ovviamente un suo costo per il cliente finale perchè per il funzionamento del programma nella macchina è richiesta una licenza, che prende fisicamente la forma di chiave usb. Per questo motivo lo SCADA è adatto a programmi che non vengono venduti in serie, ma che richiedono specifici adattamenti per ciascun singolo impianto. Infatti se i programmi realizzati fossero installati sempre uguali su un numero molto grande di impianti, non sarebbe più conveniente usare lo SCADA e quindi pagare la chiave di attivazione, ma sarebbe più intelligente sviluppare un progetto per esempio in Visual basic realizzandolo da zero. In questo modo il grande investimento di tempo e denaro per realizzare il progetto, verrebbe ripagato dal grande numero di vendite del programma che a quel punto non richiederebbe più modifiche, ma resterebbe sempre uguale.

L'insieme delle scelte effettuate per adattare il sistema alle caratteristiche precedentemente descritte ha portato ad un prodotto finale molto complesso, in grado di gestire molti dati, in continua evoluzione grafica per rendere l'interfaccia utente sempre più semplice e intuitiva, aperto a numerosi cambiamenti sia richiesti dal cliente sia pensati in ditta per rendere il prodotto competitivo nel mercato.

2.2 - Descrizione del funzionamento dell'impianto

Il sistema è un sistema di movimentazione e magazzinaggio del vetro piano per vetriere. si compone di un magazzino che può essere manuale o automatico. Un magazzino manuale è composto da cavallette fisse dove vengono posizionati in

verticale con inclinazione di 3-4 gradi i pacchi di vetro in genere da mm 6000x3210 oppure mezze lastre da mm 3210x2500. Un magazzino automatico invece serve per ridurre gli spazi quindi fa in modo di compattare gli elementi che contengono il vetro in modo che lo spazio venga ottimizzato. L'apertura di un elemento viene effettuata solo nel momento in cui serve un prelievo da tale elemento o si deve depositare la lastra all'elemento di destinazione. Questi sistemi di magazzinaggio automatico vengono comunemente chiamati Classificatori e possono essere fatti in diversi modi con diverse dimensioni, quantità di elementi e portate per il diverso piano di carico.

I sistemi di movimentazione veri e propri possono essere composti da un carro ponte con le gambe a terra o su guide aeree e si tratta di un carro ponte completamente automatico con una serie di sensori e motorizzazioni adeguate per il movimento del vetro. Su questo carro ponte viene montato un sistema a rastrelliera sulla quale sono montate le ventose che servono per prelevare ad una ad una le lastre che si trovano nel classificatore o sulle cavallette fisse per portarle poi al banco di taglio. Il sistema del carro ponte può prevedere anche delle guide aeree con il sistema su binari che si muove sopra il magazzino e quindi la rastrelliera si infila da sopra all'interno del magazzino, che sia automatico o che siano cavallette, per andare a prelevare le lastre attraverso un sistema a ventose che possono essere attivate o disattivate in base alla dimensione della lastra.

Oltre al carro ponte automatico esiste anche una caricatrice con guide a terra che si muove completamente a terra e attraverso dei sistemi a guide si infila all'interno del magazzino, quindi dal basso, e va a fare la stessa operazione prelevando le lastre dal magazzino automatico oppure da cavallette fisse. Questo tipo di caricatrice permette, una volta uscita dal magazzino, di posizionare il vetro in orizzontale e a portarsi davanti al banco da taglio selezionato.

Diversamente il carro ponte dispone di un sistema di ribaltamento della lastra. Il carro ponte quindi prenderà una lastra in verticale, la depositerà su un sistema di ribaltamento (ribaltatrice) il quale posizionerà la lastra in orizzontale e la porterà al tavolo da taglio.

Le necessità che ha un sistema per lavorare sono quelle di conoscere le posizioni, le

dimensioni delle lastre e di conoscere la presenza o meno delle lastre nel magazzino.

Il sistema di gestione avrà proprio lo scopo di tenere traccia di queste informazioni e fornirle al sistema di movimentazione.

Per evitare perdite di tempo è stata inserita la possibilità di creare delle liste di carico e scarico dal magazzino per fare in modo che il sistema possa muoversi in automatico e con delle liste precaricate possa gestire il tutto per un tempo abbastanza lungo senza che l'operatore debba intervenire per selezionare di volta in volta quale lastra vuole.

Quindi attraverso delle liste e attraverso un archivio nel quale è presente una lista di tutte le lastre del magazzino, il sistema è in grado di selezionare automaticamente dove prelevare la lastra piuttosto che su quale elemento riposizionarla.

Possiamo dire che i sistemi di movimentazione e di magazzino automatico hanno lo scopo di eseguire delle operazioni e quindi saranno degli slave o macchine che servono per arrivare allo scopo. Al contrario, il sistema di gestione di gestione del magazzino può essere considerato master in quanto, interfacciato con carroponte o caricatrice con sistema di movimentazione, ha lo scopo di fornire le informazioni necessarie per far muovere in automatico il sistema e quindi per dare le istruzioni di quali sono le operazioni una dopo l'altra. Sono gestori che permettono lo scambio di informazione da operatore a macchina.

Per parlare del sistema di movimentazione automatica o meglio del carroponte, dobbiamo ricordare che il sistema è controllato da un PLC (Programmable Logic Controller) il quale ha lo scopo di gestire in automatico tutto il funzionamento di tutta la macchina attraverso attuatori e sensori. Gli attuatori cioè motori, cilindri pneumatici e sistemi idraulici hanno lo scopo di muovere tutte le parti del carroponte quindi considerando gli assi X, Y e Z che saranno il movimento nello spazio del magazzino (assi X e Y) e l'elevazione dal terreno (asse Z), avverrà un sollevamento per fare in modo di sollevare le lastre. Un'altra motorizzazione servirà per la movimentazione della rastrelliera porta-ventose per poter ruotare le lastre e poterle depositare in condizioni particolari che si possono verificare, cioè se devo depositare le lastre in un senso piuttosto che in un altro. Un'altra componente servirà per inclinare più o meno le lastre in modo che il carroponte si possa adattare all'inclinazione del magazzino e

quindi anche ad inclinazioni diverse a seconda dell'elemento del magazzino.

Tutte queste motorizzazioni saranno controllate da sensori di posizione per fare in modo che la posizione fisica nello spazio non sia casuale, ma sia ben controllata dal plc, per fare in modo quindi che il carro ponte conosca esattamente tutte le posizioni e a seconda del ciclo che deve eseguire saprà esattamente in quali posizioni andranno tutti i vari motori e quindi gli attuatori. Oltre a questi sensori di posizione avrà altri sensori per trovare la posizione esatta del vetro quando l'elemento si scarica oppure verificare se esiste o non esiste il vetro, verificare se viene effettuato il vuoto correttamente sulle ventose o se sia insufficiente a causa di una rottura o addirittura ad una non perfetta adesione delle ventose; in più ci saranno tutta una serie di sensori per verificare extracorsa o altre condizioni particolari che possono creare condizioni di allarme. Quindi le condizioni d'allarme saranno collegate a tutti i sensori che sono presenti sulla macchina e alla buona riuscita o meno del ciclo.

Per parlare della connessione del sistema di gestione del magazzino e la macchina che esegue la movimentazione, bisogna considerare le informazioni che vengono inviate al sistema di movimentazione e quindi l'elemento di prelievo, quello di deposito, dimensione della lastra e tipo di ciclo sono informazioni che servono per eseguire in automatico, anche per una giornata intera lavorativa, il ciclo di movimentazione delle lastre.

L'operatore che si occupa di sovrintendere questo lavoro, ha la possibilità di creare queste liste attraverso un'interfaccia grafica che gli permetterà di inserire tutte le informazioni necessarie all'individuazione della lastra e alla sua successiva movimentazione. L'interfaccia che verrà utilizzata dall'utente nel nostro caso è elaborata attraverso uno SCADA collegato ad un database nel quale vengono immagazzinate tutte le informazioni sulle lastre presenti nel magazzino, tutte le liste di carico e scarico per la movimentazione del vetro, tutti gli allarmi che eventualmente verranno segnalati e le informazioni sui fornitori.

Questa parte del progetto è la più delicata perchè si incarica di passare informazioni, elaborarle e modificare il database ed è anche quella che subisce più modifiche ogni volta che un cliente richiede delle personalizzazioni. Per questi motivi l'interfaccia

utente e tutte le sue funzioni interne, sono la parte del progetto che richiede maggiormente un intervento di reverse engineering.

3 - LE TECNOLOGIE USATE

3.1 – Lo SCADA

Per spiegare cos'è un sistema scada si può partire dal significato del suo acronimo. SCADA sta per Supervisory Control And Data Acquisition (acquisizione dati, supervisione e controllo) e sintetizza le tre funzioni fondamentali svolte da questo genere di sistema. In uno SCADA l'acquisizione dati è funzionale allo svolgimento delle funzioni di supervisione, osservazione dell'evoluzione del processo controllato, e di controllo, attuazione di azioni volte alla gestione degli stati nei quali il processo controllato si trova e delle transizioni tra gli stati nei quali il processo può venire a trovarsi.

Quanto finora detto non si limita a qualificare i sistemi SCADA, ma caratterizza un insieme molto ampio di sistemi di controllo che differiscono tra loro per elementi quali la distribuzione geografica del sistema, la distribuzione (o la centralizzazione) dell'intelligenza del controllo, il grado di interazione tra operatore umano e sistema, i tempi di reazione a un evento prodotto dal processo controllato e molti altri fattori.

L'acquisizione dati entra nella definizione di sistema SCADA per il fatto che non è possibile espletare funzioni di supervisione senza acquisire informazioni sullo stato in cui si trova il processo così come non è possibile orientarne il comportamento, cioè controllarlo, senza avere la possibilità di influenzare lo stato cambiando il valore di parametri che lo caratterizzano.

La supervisione è la funzione per mezzo della quale un sistema SCADA rende possibile l'osservazione dello stato e dell'evoluzione degli stati di un processo controllato. A questa funzione appartengono tutte le funzionalità di visualizzazione delle informazioni relative allo stato attuale del processo, di gestione delle informazioni storiche, di gestione degli stati che costituiscono eccezioni rispetto alla normale evoluzione del processo controllato. La funzione di supervisione costituisce un fine per qualsiasi sistema SCADA. Questa funzione è determinante nella caratterizzazione di un sistema nel senso che un sistema che non permetta di accedere

alle informazioni di stato corrente e/o storiche del processo osservato e/o controllato non può essere definito come sistema SCADA.

La funzione di controllo rappresenta la capacità di un sistema di prendere decisioni relative all'evoluzione dello stato del processo controllato in funzione dell'evoluzione del processo stesso. Le funzionalità di controllo sono quindi concentrate nel sistema di elaborazione il quale, una volta eseguite opportune procedure di elaborazione, sfrutta il sistema di acquisizione dati in senso inverso per cambiare il valore di opportuni parametri di stato del processo controllato.

3.1.1 – Analisi del processo controllato

Prima di procedere è opportuno porre l'attenzione sull'importanza rivestita dall'analisi del processo controllato nella definizione delle caratteristiche architetture e funzionali del sistema di controllo. Questo tipo di analisi impone vincoli significativi nella realizzazione di sistemi di controllo rendendolo un processo che di volta in volta richiede tecniche di sviluppo adeguate difficilmente riconducibili a modelli convenzionali.

L'analisi del processo controllato produce informazioni che influenzano le scelte progettuali sia dal punto di vista tecnologico che organizzativo ed è buona norma l'approfondimento della conoscenza del processo prima della definizione delle caratteristiche del sistema destinato a controllarlo.

Di seguito verranno riportati alcuni elementi classici di indagine per la determinazione dei requisiti fondamentali del sistema di controllo.

3.1.2 - Realtime

Il termine Realtime si riferisce alla capacità del sistema di reagire alle sollecitazioni del processo con ritardi trascurabili rispetto alla dinamica evolutiva del processo medesimo.

L'evoluzione dei sistemi di calcolo rende realistico il desiderio di creare un sistema

realtime, ma anche l'uso delle migliori tecnologie non sarà mai in grado di rendere nulli i tempi del trasferimento dell'informazione.

Ai limiti imposti dalla tecnologia si aggiungono le problematiche di affidabilità e di disponibilità. La capacità di operare correttamente con continuità può essere assicurata per mezzo di processi di elaborazione che in molti casi si sovrappongono a quelli di acquisizione e controllo invalidandone le caratteristiche dinamiche.

3.1.3 – Alta affidabilità

Una qualità della quali i sistemi di controllo non possono fare a meno è l'altra affidabilità dei componenti del sistema stesso (reliability). Un sistema è un insieme organizzato di componenti, ognuno dei quali ha un grado di affidabilità, cioè una probabilità di malfunzionamento calcolato in percentuale di tempo di servizio del componente stesso. Per programmi realizzati ex-novo è necessario inserire un sistema di controllo degli errori, compiere quindi un'analisi con lo scopo di raggiungere un grado di affidabilità soddisfacente per l'intero sistema.

3.1.4 – Alta disponibilità

La disponibilità è la percentuale di tempo durante il quale deve essere assicurato lo stato di esercizio del sistema. Tale stato sarà complementare al tempo in cui il sistema è fermo a causa di un guasto, aggiornamenti o manutenzione. Per alcune macchine è chiesta un'alta disponibilità per esempio quando si devono monitorare temperature, ma per l'impianto che si sta analizzando questa disponibilità è diminuita notevolmente perchè i movimenti dell'impianto sono molto lenti rispetto alla velocità di sistema.

3.1.5 - HMI

La realizzazione delle funzioni di un sistema di supervisione e controllo comporta sempre la realizzazione di sottosistemi responsabili dell'interazione tra gli operatori e

il sistema medesimo denominati interfacce uomo-macchina (in inglese sono detti HMI che significa *human-machine interface*). La complessità dello sviluppo è funzione del tipo di interazione richiesta che può essere di sola supervisione come il monitoraggio delle temperature oppure funzionalità che richiedono l'intervento dell'operatore per realizzare procedure manuali. Nei casi che prevedono forme di controllo oltre che di supervisione, risulta fondamentale la realizzazione di interfacce di facile utilizzo e di funzionalità accessorie necessarie alla comprensione dello stato di esercizio del sistema, funzionalità quali la gestione della notifica degli allarmi e la visualizzazione di grafici relativi alle grandezze più rappresentative.

Tutto quello che viene progettato per la realizzazione dell'interfaccia uomo-macchina tiene conto del fatto che l'utente finale è un uomo e che al centro dell'attenzione ci devono essere le sue esigenze. Non solamente gli aspetti che riguardano la visualizzazione di informazione, ma anche la facilitazione di tutte quelle operazioni che implicano l'interazione con il sistema.

Curare tutti questi aspetti, corrisponde a mettere a proprio agio l'operatore che vedrà nel sistema un aiuto durante il lavoro di ogni giorno. Questo fa sì che si crei una sorta di fiducia tra uomo e macchina che trova il suo vantaggio nel momento in cui si presentino situazioni di allarme. In questo caso infatti l'operatore, fidandosi della stabilità del sistema, sarà in grado di affrontare più lucidamente le situazioni critiche.

La riduzione e la semplificazione delle operazioni, l'automazione delle operazioni ripetitive, l'utilizzo di dispositivi ergonomici sono elementi che possono determinare il successo dell'adozione di un sistema SCADA.

3.1.6 – Funzioni

La funzione principale di un'interfaccia uomo-macchina, come abbiamo detto, è la gestione delle interazione tra SCADA e operatore. Le informazioni che possono essere scambiate sono divisibili in due categorie:

- presentazione dei dati
- gestione dei comandi operatore

Vediamo di seguito singolarmente le caratteristiche di tali tipi di dati.

3.1.7 – Presentazione dei dati

La funzione di presentazione dei dati permette all'operatore di essere informato sugli stati del processo di controllo visualizzando i dati rilevati o elaborati dal sistema. La visualizzazione di queste informazioni può essere realizzata per mezzo di tre diverse modalità: rappresentazioni schematiche, tabellari e temporali.

Le rappresentazioni schematiche servono per dare all'operatore che deve sovrintendere l'impianto una vista sullo stato del processo. Per questo motivo si tende ad avere come schermata principale uno schema “mimico” degli impianti contenente le informazioni principali riguardanti la macchine e il processo in corso. Grazie allo sviluppo delle tecnologie, in questo caso degli elaboratori, si è in grado sempre di più di rappresentare informazioni utilizzando questa modalita'.

Con l'uso del computer infatti è stato possibile disporre di più pagine video per la rappresentazione del processo, associare a ogni pagina informazioni diverse permettendo di suddividere il processo in più sezioni e, quindi, avere rappresentazioni sempre più dettagliate.

Esistono poi le rappresentazioni tabellari utilizzate per inserire in una tabella un insieme di valori tra loro correlati. E' utile far uso di tabelle nei casi in cui si debba:

- mettere in relazione due valori prelevati in punti diversi dell'impianto;
- confrontare un valore con i limiti di funzionamento della grandezza che il valore rappresenta;
- mostrare in un'unica pagina una serie di valori caratteristici del processo che ne diano una visione di immediata interpretazione.

Infine esiste la possibilità di rappresentare le informazioni in diagrammi di tempo. Questo strumento permette di visualizzare l'andamento di dati significativi in funzione del tempo tracciando curve che permettono di conoscere l'evoluzione del dato stesso. In questo modo l'operatore può avere un'informazione immediata e fare delle scelte

anche istantanee per riportare i valori allo stato ottimale.

Nel caso specifico della macchina che si sta analizzando, le pagine (anche chiamate sinottici) non sono rappresentazioni dei movimenti della macchina, questo perchè il progetto prevede la visualizzazione di dati e l'impostazione di diversi valori che poi faranno muovere l'impianto. In questo particolare caso quindi non è richiesto lo schema mimico o una suddivisione di esso, ma una visualizzazione chiara e intuitiva dei dati. Per quanto riguarda i diagrammi temporali questi non sono mai stati inseriti pur essendo previsti dallo SCADA in utilizzo, sia perchè il cliente non ha mai richiesto una funzionalità del genere, sia perchè non è ancora stato individuato un dato che cambiando nel tempo possa servire a prendere decisioni riguardo alla movimentazione del vetro.

3.1.8 – Gestione dei comandi operatore

Una parte importante dei sistemi SCADA è la possibilità di inviare comandi. Esistono due tipi di comandi, quelli manuali (che richiedono l'intervento dell'operatore) e quelli automatici (gestiti dal sistema). Ogni operazione che viene avviata deve essere gestita e controllata in modo da non provocare errori o, nel peggiore dei casi, il blocco dell'impianto. Per ottenere questo è necessario che il sistema attui delle procedure di controllo all'interno del sistema ogni volta che un'operazione viene avviata così da rendere l'operatore il più possibile tranquillo. Infatti se esso non è sottoposto a stress ed è sicuro che le azioni che sta compiendo non produrranno errori, sarà più rilassato e lavorerà meglio.

Ogni volta che l'operatore manifesta l'intenzione di avviare un'operazione il sistema deve chiedere una conferma così da evitare azioni intempestive. Un sistema che è in grado di prevenire un errore da parte dell'operatore infatti è da preferirsi.

Per tutte quelle azioni che possono essere svolte automaticamente dal sistema è preferibile lasciare all'operatore il solo onere di avviare la procedura. In questo modo, fornendo un valido sistema di supporto, si innalza notevolmente il livello di sicurezza e qualità di tutto l'impianto.

3.1.9 – Controllo gestione comandi

Ogni avvio di un processo è potenzialmente pericoloso per il sistema, esistono per questo due tecniche per fare in modo che alcuni comandi potenzialmente pericolosi non possano essere impartiti. Il primo è quello di gestire gli accessi degli operatori con delle password. In questo modo si definiscono subito i compiti che un operatore può o non può eseguire già in fase di configurazione.

Ogni volta che un comando viene richiesto il sistema controlla se l'operatore ha i diritti d'accesso per poterlo richiedere. Nel caso il risultato sia positivo è necessario comunque che venga richiesta una conferma. In questo modo all'operatore è richiesta una doppia conferma per una sola operazione. Questa si ritiene essere la tecnica migliore per evitare che siano avviate operazioni involontarie.

Per sistemi più complessi che necessitano di un più alto grado di controllo, sono previsti gestioni di password a diversi livelli, password a tempo etc.

Alla fine di ogni operazione deve seguire una risposta dal sistema SCADA che mostri all'operatore se il comando ha effettivamente eseguito il suo compito. Se l'operatore si trova nei pressi dell'impianto che deve movimentare questo ha una risposta immediata al suo comando nell'esecuzione del movimento, ma se il sistema di controllo è posizionato in un'altra stanza l'operatore deve sapere se il sistema ha preso in carico il suo comando e poi se è andato a buon fine oppure no. Questo tipo di comunicazione può essere di tipo testuale con messaggi come “Operazione eseguita” oppure con segnali luminosi o simboli di facile interpretazione. Un'operazione sintetica giunge più chiaramente all'operatore.

3.1.10 – Eventi e Allarmi

Sia gli allarmi che gli eventi sono stati del funzionamento del sistema che richiedono una elaborazione da parte del sistema SCADA. Allarmi ed eventi sembrano essere la stessa cosa, ma in realtà non lo sono. Un evento indica il verificarsi di una particolare condizione, un allarme indica lo stato di una particolare condizione.

L'allarme si riferisce a condizioni di funzionamento anomalo del sistema di controllo che richiede una reazione gestita dall'operatore o dal sistema stesso automaticamente. Gli eventi invece vengono usati per mettere a conoscenza l'operatore che una determinata azione è stata eseguita, il suo scopo quindi è puramente informativo e non richiede interventi. Entrambi però sono accomunati dal dato temporale che il sistema SCADA attribuisce ad entrambi per facilitarne la gestione.

Facendo un esempio pratico che riguarda l'impianto che si sta analizzando forse la differenza risulterà più chiara.

Ogni volta che viene eseguito uno spostamento di una lastra al banco di taglio o viceversa, si attiva un evento che permette l'aggiornamento del file di magazzino, un file formato testo che in ogni momento della giornata lavorativa riassume tutto quello che è presente nel magazzino con descrizioni, quantità e misure.

Un esempio di allarme invece si verifica quando si cerca di prelevare una lastra da un elemento (cavalletta) vuoto. In questo caso una spia luminosa si attiva e fa sapere all'operatore che si è verificato un errore.

I sistemi di controllo sono in grado di funzionare per diverso tempo senza che l'operatore sia presente per avviare continuamente i processi, in questo caso è necessario che nel momento in cui si verifica un'anomalia questa sia notificata attraverso sistemi di telecomunicazione per poter così raggiungere l'operatore che potrebbe spostarsi continuamente nel territorio. Per questo motivo è stata introdotta la possibilità di una notifica via e-mail oppure via sms dell'allarme.

Bisogna fare attenzione a questi tipi di scelte però, perchè in base alla gravità della notifica questa deve essere letta dall'operatore in tempo reale, ma i mezzi a disposizione non assicurano l'effettiva lettura da parte dell'operatore.

In definitiva la possibilità di non presidiare l'impianto comporta un grande vantaggio per chi lo gestisce ma al tempo stesso richiede uno sforzo organizzativo e progettuale notevole dovuto proprio al fatto che gli impianti non sono presidiati, cosa che comporta un notevole aumento dei rischi.

3.1.11 – Archiviazione dati

L'archiviazione dei dati permette di memorizzare i dati su file opportunamente formattati e rimandare al momento in cui si verifica la necessità di accedere alle informazioni l'attività di trasferimento dei dati dai file a una banca dati relazionale gestita da RDBMS.

La disponibilità dell'informazione nella forma di banche dati gestite da RDBMS ha un importante significato dal punto di vista dell'interoperabilità tra sistemi concorrenti. Questo in virtù del fatto che i sistemi di gestione delle banche dati espletano la doppia funzione di gestori, occupandosi di organizzazione, integrità e quant'altro, e di fornitori di meccanismi d'interrogazione, per mezzo di un linguaggio di interrogazione definito secondo uno standard consolidato, il noto Structured Query Language o SQL. La potenza del linguaggio SQL garantisce la massima accessibilità possibile alle basi di dati senza compromettere l'efficacia delle transazioni, anche quando le strutture dei dati sono molto complesse. Le caratteristiche di linguaggio definito come standard sia dall'ANSI (American National Standards Institute) che dal mercato del software lo qualificano come protocollo di comunicazione per l'interoperabilità dei sistemi diffuso e stabile. Le metodologie di accesso a basi di dati relazionali tramite SQL sono ormai consolidate e si possono dividere in due categorie principali: quelle che utilizzano interfacce standardizzate e aperte; quelle che utilizzano interfacce proprietarie e specifiche del singolo RDBMS.

Alla prima categoria appartengono i software comunemente classificati come drivers, cioè strati intermedi interposti tra RDBMS e applicativi. Questi traducono le query provenienti dagli applicativi in comandi comprensibili al RDBMS e i risultati delle query in forme comprensibili agli applicativi. Il più importante e diffuso driver, detto ODBC (Open DataBase Connectivity), fa riferimento a un metodo definito dallo SQL Access group nel 1992 ed è lo strumento più usato per la realizzazione di applicazioni che non sono soggette a vincoli prestazionali e di affidabilità particolarmente stringenti come nel caso del progetto per le vetrerie che si sta analizzando.

Una caratteristica importante del metodo ODBC sta nel fatto che i driver che lo

realizzano definiscono un'interfaccia standard indipendente dal RDBMS, lasciando agli applicativi che accedono alle informazioni il solo onere di conoscere la struttura dei dati.

Le API sono librerie software rese disponibili dai produttori di software RDBMS nei principali linguaggi di programmazione utilizzati per lo sviluppo di software applicativi (Visual Basic, C, C++, ...). Con queste `e possibile realizzare un controllo completo su installazione, uso e distruzione dei collegamenti tra applicativi e RDBMS, il tutto fatto con grande riguardo per le prestazioni e l'affidabilità delle transazioni.

3.2 – Il PLC

Il cuore di tutto il lavoro è indubbiamente il plc. Infatti esso colloquia con la macchina tramite ingressi e uscite e scambia informazioni con lo SCADA per la gestione del magazzino.

Il plc (programmable logic controller) è un sistema a microprocessore che esegue un programma ed elabora i segnali digitali ed analogici provenienti da sensori e diretti agli attuatori presenti in un impianto industriale. La caratteristica principale è la sua robustezza, infatti normalmente il PLC è posto in quadri elettrici in ambienti rumorosi, con molte interferenze elettriche, con temperature elevate o con grande umidità. In certi casi il PLC resta in funzione 24 ore su 24, 365 giorni all'anno senza mai fermarsi per necessità dell'impianto. Questi pre-requisiti richiesti dal mercato hanno fatto sì che venisse sviluppato questo tipo di apparecchiatura industriale capace di fornire questo livello di disponibilità.

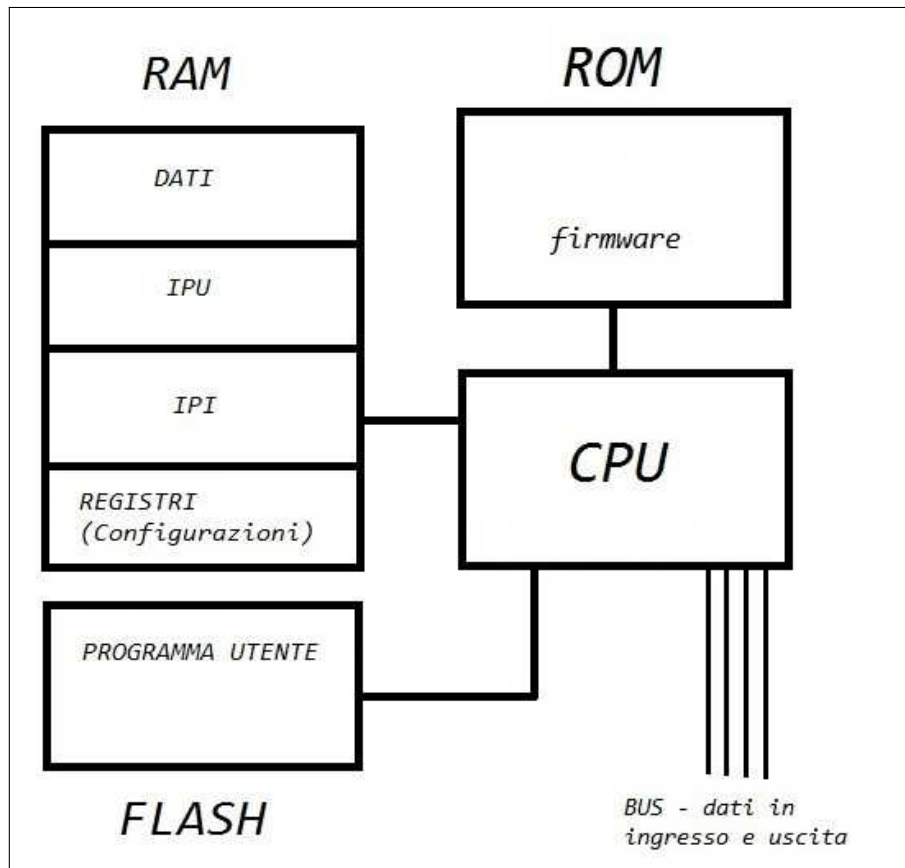
3.2.1 – Specifiche hardware

Nel plc sono presenti diverse parti che, collegate tra loro, permettono il funzionamento del plc. Le specifiche hardware del plc sono trattate dalle norme IEC 61131-1, per tale motivo tutti i plc devono rispondere a dei requisiti minimi e comuni. Le parti hardware sono: microprocessore, una memoria ROM che contiene il firmware dove il costruttore mette il proprio sotto programma che serve per far lavorare il microprocessore con il metodo della scansione ciclica. Il firmware contiene anche tutte le funzioni, dette anche API, che saranno le uniche utilizzabili nel programma utente. Questo significa che il programma utente non ha accesso al set di istruzioni del processore. Nei plc di ultima generazione il programma utente viene sempre scritto in una memoria flash (oppure EEPROM) per evitare la perdita del programma. Nei plc di vecchia generazione invece il programma veniva scritto nella memoria RAM e mantenuto da una batteria tampone in mancanza di alimentazione esterna. Con il tempo la batteria tampone perdeva la sua efficacia con il conseguente rischio di perdere il programma.

Un'altra parte del plc previsto dalla normativa è la memoria RAM. La funzione di questo componente è quella di memorizzare tutti i dati di processo e renderli disponibili per l'elaborazione del programma utente. Questa memoria viene suddivisa in varie parti secondo quanto predisposto dal firmware. In alcuni casi il programma utente o la configurazione del programma utente possono suddividere in altre parti la memoria messa a disposizione dal firmware.

Le parti principali della RAM sono:

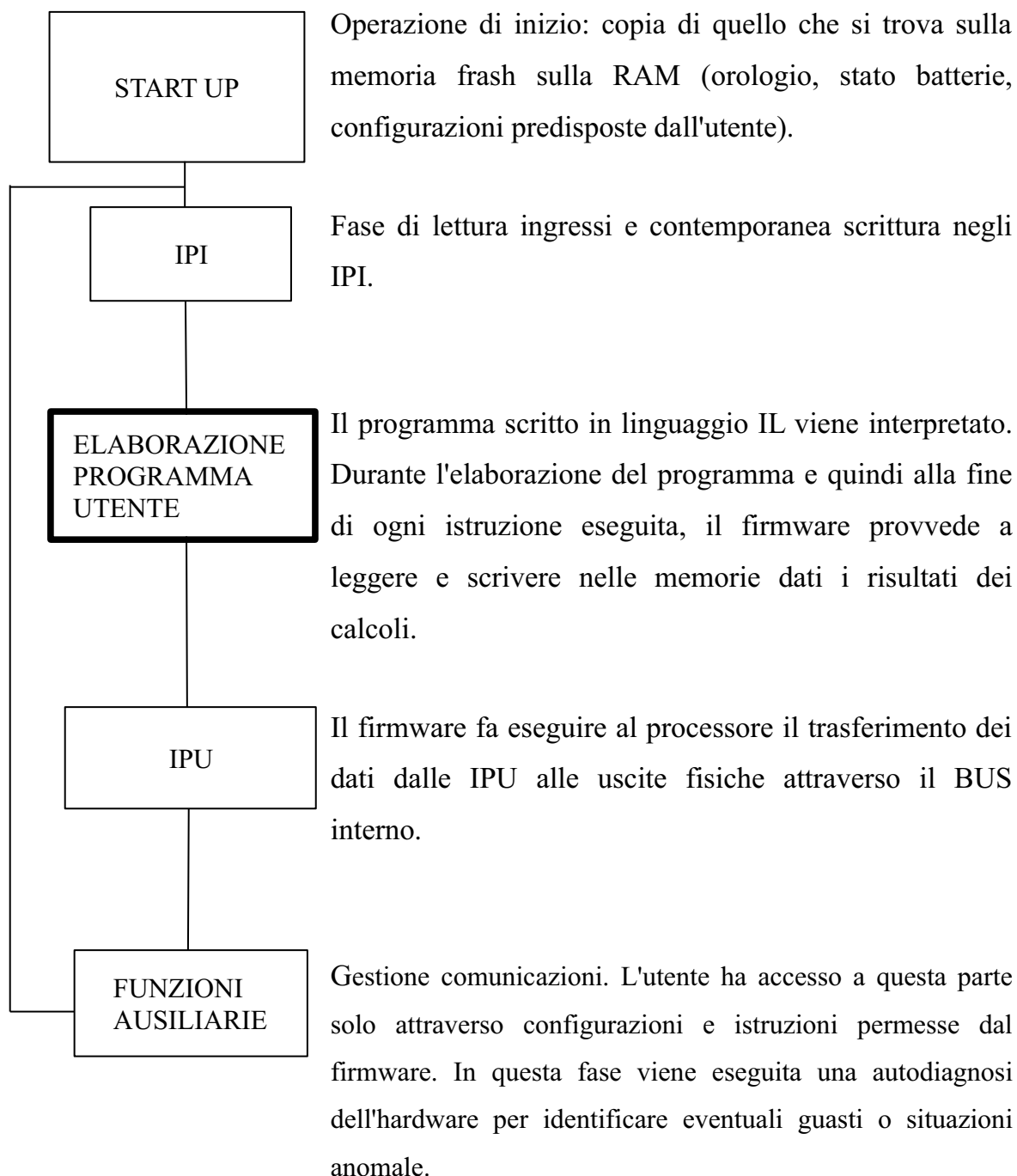
1. Memoria di sistema (orologio, calendario) o registri nella memoria di sistema.
2. Memoria Dati che a seconda di tipo di PLC può essere suddivisa in diversi pezzi per essere indirizzata scegliendo la tipologia di dati fra il bit, il byte, word e doppia word.
3. IPU – IPI (Immagine processo uscite/immagine processo ingressi). Si tratta di un'immagine di tutti i dati in uscita e in ingresso così da ottimizzare i tempi di trasmissione sul BUS. Infatti la CPU dopo l'elaborazione dei dati, mette i risultati nella memoria dedicata agli IPU, successivamente la ROM deciderà quando i dati nella IPU saranno mandati alle uscite fisiche attraverso il BUS.
4. Periferiche. Una delle periferiche più importanti è sicuramente il BUS interno di comunicazione che permette al plc di espandere il suo numero di ingressi e uscite. Le espansioni comunicando attraverso il BUS interno, possono scambiare velocemente le informazioni con le IPI e le IPU. In genere questo tipo di BUS è proprietario ed è utilizzato in termini di tempi di trasmissione. Altre periferiche importanti sono le comunicazioni seriali. In genere queste sono asincrone rispetto alla scansione plc, utilizzano memoria RAM propria come buffer di ricezione e trasmissione e permettono la comunicazione con apparecchiature esterne come lo SCADA senza compromettere il tempo di scansione del plc.



3.2.2 – Scansione ciclica

Il plc è una apparecchiatura industriale basata su un sistema a microprocessore. La caratteristica principale è la scansione ciclica. Questa operazione permette di elaborare tutti i network del programma utente ad ogni scansione. Questo significa che tutte le parti del programma utente verranno sicuramente elaborate entro un tempo molto breve che è il tempo della scansione. Il tempo della scansione può variare di molto in base alla dimensione del programma utente, al tipo di istruzioni utilizzate e alla velocità del microprocessore. La scansione ciclica prevede una sequenza di operazioni che vengono eseguite in media ogni 10-15 millisecondi, tutte operazioni calcolate attraverso l'algebra booleana e funzioni matematiche.

Vediamo di seguito da quali fasi è composta la scansione ciclica.



3.2.3 – Linguaggi di programmazione

Per ottemperare al suo compito il PLC deve essere programmato. La programmazione del PLC è effettuata normalmente con un PC sul quale un software specializzato permette di creare programmi da scaricare nella memoria della CPU del PLC.

Questi software di programmazione possono leggere il programma direttamente dalla

memoria della CPU e visualizzare il programma sul PC. Normalmente il programma viene scritto su PC, quindi scaricato sul PLC, e salvato sul PC stesso, per ulteriori modifiche o per sicurezza.

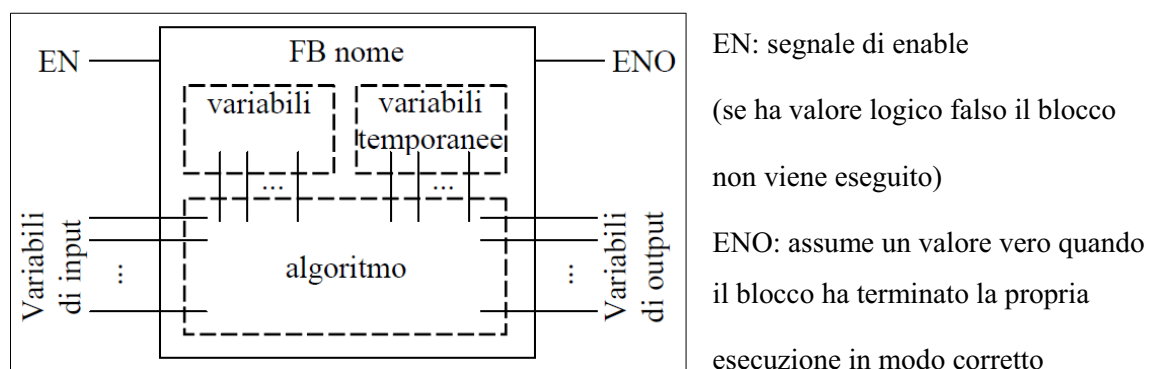
La normativa IEC 61131-3 del 1993 ha standardizzato 5 linguaggi di programmazione, di cui 3 grafici e 2 testuali (la Normativa è stata successivamente aggiornata con l'uscita della "CEI EN 61131-3" detta anche "CEI 65-40" Prima Edizione: 1° giugno 1996). Con l'introduzione di uno standard sono stati introdotti diversi benefici, tra questi i più immediati sono l'aver fornito modelli, concetti e terminologia comuni ai tecnici del settore, aver costruito un riferimento per la realizzazione di strumenti di sviluppo, verifica e simulazione, aver facilitato l'interazione tra progettisti e il riuso di elementi del progetto ed infine aver consentito la sopravvivenza a diverse generazioni di tecnologie.

3.2.4 – Linguaggi grafici

I linguaggi grafici sono quei tipi di linguaggi che usano simboli grafici per descrivere processi sequenziali. Per il plc secondo le normative, i linguaggi previsti sono:

- Ladder diagram (*LD o KOP*) detto Linguaggio a contatti - È il linguaggio più usato fino a pochi anni fa, in quanto era la trasposizione informatica dei circuiti elettrici usati dagli elettrotecnici. Il programmatore semplicemente utilizza simboli logici corrispondenti a segnali di ingresso e di uscita per implementare la logica non più cablando i relè, ma disegnando gli schemi elettrici nel software di programmazione.
- Sequential function chart (SFC) detto Diagramma funzionale sequenziale - Viene usato anche come strumento di specifica. Tale linguaggio permette di implementare facilmente una macchina (o automa) a stati finiti.
- Function Block Diagram (FBD o FUP) detto Diagramma a blocchi funzionali - Analogo ai diagrammi circuitali. Il *Diagramma a Blocchi Funzionali* (*Function Block Diagram, FBD*) può essere utilizzato per esprimere il

comportamento di funzioni, blocchi funzionali e programmi. E' analogo ai diagrammi di circuiti elettrici in cui le connessioni elettriche rappresentano i percorsi dei segnali tra i componenti. Un blocco funzionale ha due caratteristiche principali, la definizione dei *dati* (ingressi e uscite) e un *algoritmo*, che viene lanciato ogni volta che viene eseguito il blocco. Esso processa i valori correnti dei parametri di ingresso e i valori delle variabili interne (locali o globali) e produce i nuovi valori dei parametri di uscita.



3.2.5 – Linguaggi testuali

I linguaggi testuali sono quei tipi di linguaggi scritti che hanno un lessico, una sintassi e una semantica ben definita. Per la normativa i linguaggi sono:

- Instruction List (IL o AWL) detto Lista di istruzioni - La *Lista Istruzioni* è un linguaggio di basso livello simile all'ASSEMBLER. E' adatto per compiti molto specifici quali l'interfacciamento di hardware particolare. Anch'esso come il LD è disponibile per tutti i PLC. Un'istruzione IL è composta da un'eventuale label (etichetta) seguita da un operatore, un'eventuale modificatore ed un operando. Questo tipo di linguaggio si può facilmente ricavare dal linguaggio LD.
- Structured Text (ST) detto Testo strutturato - Linguaggio di alto livello simile al Pascal.

4 - StorePRO

StorePro è l'interfaccia uomo-macchina creata dalla ditta Gioachin tramite l'utilizzo di Panaway, lo SCADA adottato per la realizzazione del progetto.

Panaway è creato dalla ditta Progea con il nome di Movicon e distribuito dalla Panasonic Electric Works Italia con il nome di Panaway. Installando il programma in runtime su un qualsiasi computer con alla base un sistema operativo windows (xp o vista), è possibile visualizzare e usare StorePro.

StorePro è stato realizzato per i magazzini delle vetrerie e in particolare per permettere agli operatori che devono movimentare le lastre di vetro, di passare le informazioni alla macchina. Partendo da questo presupposto è stato sviluppato un programma che non solo permette gli spostamenti, ma permette anche di tenere sotto controllo l'intero magazzino visto che è in grado di mostrare l'esatto contenuto dello stesso in ogni momento.

4.1 - Struttura

Le “pagine” che vengono visualizzate dall'operatore sono chiamate sinottici e servono per colloquiare con la macchina. I sinottici possono contenere campi per la visualizzazione dei dati, per la scelta di un dato con menù a tendina, pulsanti per far partire un processo, spie luminose per segnalare gli stati del processo come gli allarmi. Ogni componente di questi sinottici è associato ad una variabile globale e può attivare un processo interno di elaborazione di dati, oppure può attivare lo scambio di informazioni con il plc che poi andrà a far muovere la macchina.

4.1.1 - I Sinottici

I) MAGAZZINO

I sinottici che fanno parte del progetto sono al massimo 12, per spostarsi attraverso di essi esiste un menù sempre presente nella parte bassa della pagina.

Il sinottico principale è quello di visualizzazione delle lastre: qui sono presenti tutte le informazioni delle lastre una per una. Per fare questo è stato necessario associare ad ogni campo una variabile e poi creare uno script in modo che scorrendo gli elementi e le lastre le informazioni puntino direttamente alla lastra che interessa visualizzare.

In questo modo scorrendo le lastre, i dati cambiano in tempo reale e forniscono all'operatore tutte le informazioni necessarie. Da qui è possibile anche aggiungere, inserire, salvare o eliminare lastre all'interno del magazzino. Per aggiungere basta indicare l'elemento, il codice, il numeri di lastre e le misure, per inserire basta posizionarsi nell'elemento e nella posizione in cui si desidera inserire la lastra, salvare serve nel momento in cui si vanno a modificare dei dati di una lastra che devono essere conservati in memoria. Esistono infine due tipi di eliminazione lastra: eliminazione di una singola lastra e quella di tutte le lastre di un elemento. Per cancellare una sola lastra basta visualizzarla e dare il comando “cancella”, per cancellare tutte le lastre di

un elemento basta sia selezionato l'elemento e con il pulsante apposito cancellare le lastre.

Tutti i pulsanti di questo sinottico non danno avviso di successo o insuccesso dell'operazione e non richiedono conferme per eseguire o meno il procedimento, trannè il tasto che elimina tutte le lastre di un elemento che chiede conferma per poter cancellare le lastre dall'elemento selezionato.

II) CREA LASTRA FORNITORI

The screenshot shows the 'STORE Pro v.1.15' application window with the title 'Crea lastra e fornitori'. It features two main data entry forms. The first form, titled 'Lastra', contains a 'Codice' field with the value '4+4acid' and a 'Descrizione' field with the value 'laminato 4+4 acidato'. The second form, titled 'Fornitore', contains a 'Fornitore' field with the value '210' and a 'Descrizione' field with the value '210'. Each form includes icons for adding, editing, and deleting records, and a red 'X' icon for deletion. The application title bar shows flags for Italian, English, and French.

Dalla pagina principale si può accedere alla pagina di creazione o cancellazione di un tipo di lastra o di un fornitore. Da qui è possibile creare un nuovo tipo di lastra che andrà ad essere disponibile nella lista delle lastre da inserire nel magazzino, oppure un nuovo fornitore che sarà anch'esso disponibile nella lista dei fornitori.

Per quanto riguarda la cancellazione essa compie una verifica prima di procedere all'eliminazione del dato, infatti potrebbe essere che esista ancora nel magazzino una quantità di lastre del tipo che si sta cercando di cancellare. Lo script di cancellazione quindi compie un'analisi e se trova tra le lastre una in utilizzo, non cancella la lastra.

In questo modo si evita di avere informazioni errate in memoria.

III) FILE



Dal sinottico “File” è possibile gestire le liste di ottimizzazione per i banchi da taglio. Le liste vengono create da un programma apposito per ottimizzare il taglio delle lastre in modo da avere il meno spreco possibile di vetro.

Da questo sinottico è possibile importare le liste create dal programma di ottimizzazione, importarle direttamente al tavolo di taglio, eliminarle dal tavolo di taglio e anche dalla memoria del programma.

Nel momento in cui si importa il file dal programma di ottimizzazione in StorePro, il file passa ad una macro che lo trasforma in un file di testo. Il programma poi userà tale file di testo per tutte le sue operazioni. Questo processo è stato pensato in questo modo perchè se la ditta che produce il programma di ottimizzazione dovesse cambiassse, per mantenere la funzionalita' in StorePro sarebbe sufficiente modificare la macro per creare il file testo ed il resto sarebbe gestito normalmente.

Quando invece si carica una lista nel banco, il programma prima verifica la presenza o meno delle lastre richieste nel magazzino. Nel caso queste non siano tutte presenti appare un message box con la possibilità di caricare lo stesso la lista o annullare l'operazione.

E' anche possibile che dal tavolo di taglio venga creata una lista dall'utente e che

esso voglia salvarla per attivarla in un altro momento, da questo sinottico è possibile salvare un file sempre in formato testo con la lista creata dall'utente. Per fare questo basta inserire un nome nel campo apposito e lanciare il comando che svuota la lista dal banco.

Anche in questo caso per ogni azione che viene eseguita non viene dato nessun tipo di messaggio all'operatore sullo stato di avanzamento dell'operazione.

IV) BANCHI

STORE Pro v.1.15
Lista di carico banco 1

Elemento	Commessa	Codice	Fornitore	Lotto	Larghezza	Altezza	Spessore
22		FLOAT3MM	*	*	3210	2550	3

Indice: 7
Ritorno: 0
Elemento: 1
Codice: 4+4
Larghezza: 6000
Altezza: 3210
Commessa: *
Quantità: 1
Spessore: 0

Elimina cartone
Continua

Indice	Commessa	Ritorno	Elemento	Codice	Larghezza	Altezza	Spessore
1	1	*	1	4+4	6000	3210	0
2	2	*	0	3+3M	6000	3210	0
3	3	*	0	5+5acid	6000	3210	0
4	4	*	1	4+4	6000	3210	0
5	5	*	0	4+4	6000	3210	0
6	6	*	0	4+4	6000	3210	0
7	7	*	0	4+4	6000	3210	0

Da	A	Codice	Fornitore	Lotto	Peso	Larghezza	Altezza	Spessore	Cartone	Rotta
0	0				0	0	0	0	0	0

I sinottici dei banchi, che possono essere al massimo 4, permettono la gestione delle liste di movimentazione che possono essere liste di ottimizzazione oppure liste create dall'operatore. Attualmente esistono due possibilità di prelievo: quello automatico e quello manuale. Per il manuale basta indicare l'elemento esatto dal quale prelevare la lastra, per l'automatico si indica il tipo di lastra e la misura richiesta e il programma cerca da solo la lastra indicata. Per quanto riguarda il ritorno, la lastra può essere riposizionata nell'elemento di prelievo, oppure se non viene indicato nessun elemento, sarà riposizionata nel Restore.

Il Restore è semplicemente una parte del magazzino che serve per conservare le lastre che hanno già compiuto almeno un ciclo nel banco da taglio. Le lastre che

vengono posizionate in questi elementi non sono più una sopra all'altra, ma sono una a fianco dell'altra con la possibilità di avere al massimo 3 lastre per elemento.

Il particolare tecnico del Restore riguarda lo scambio dati tra StorePro e il plc. Di solito la lastra da prelevare è sempre l'ultima del pacco, cioè quella più esterna, per il Restore invece tutte e tre le lastre sono disponibili. Quindi la tabella con il riepilogo delle lastre dovrà comprendere tutte e tre le lastre del Restore più tutte le ultime del resto del magazzino. A questo punto però nasce un problema perchè l'indice delle lastre per StorePro è inverso rispetto a quello del plc. Infatti il criterio di archiviazione per StorePro è LIFO (last in-first out) mentre quello del plc è di tipo FIFO (first in-first out). Per questo è stato creato uno script apposito per passare al plc l'indice giusto. Questo serve perchè al momento del prelievo la macchina deve posizionarsi e prelevare la lastra in maniera corretta.

La formula inserita per passare al plc l'indice giusto per il prelievo è:

$$((n + 1) - sap_numerolastra)$$

dove n rappresenta il numero delle lastre presenti nell'elemento del restore e sap_numerolastra è il numero della lastra da prelevare secondo il criterio di StorePro.

Facciamo un esempio. Se nell'elemento del restore sono presenti 3 lastre e quella da prelevare per StorePro è la terza, allora al plc dovrà essere passato l'indice 1. Se nell'elemento sono presenti 2 lastre e quella da prelevare è la prima, al plc dovrà essere passato come indice il numero 2. Ecco che con questa formula si realizza proprio questo e al plc viene passata l'informazione esatta.

V) SPOSTAMENTI

STORE Pro v.1.15
Spostamenti

Automatico

Indice: 4

Quantità: 0

Da elemento: 0

A elemento: 0

Elimina cartone

Continua

	Indice	Da elemento	A elemento	Quantità
1	1	2	1	2
2	2	21	48	3
3	3	48	30	1

Da	A	Codice	Fornitore	Lotto	Peso	Larghezza	Altezza	Spessore	Cartone	Rotta
0	0				0	0	0	0	0	0

Da questo sinottico è possibile creare una lista per spostare le lastre di vetro da un elemento all'altro, non dall'elemento al banco di taglio come nel precedente sinottico. Quindi da questa pagina l'utente può spostare le lastre solo all'interno del magazzino per la movimentazione. Il Restore è escluso da questi spostamenti.

VI) MANUALE

STORE Pro v.1.15

Movimenti manuali

SPOSTAMENTO BANCO N° -> ELEMENTO

Elemento	Commessa	Codice	Fornitore	Lotto	Larghezza	Altezza	Spessore
48	proooova	4+4acid	1	*	6000	3210	0

Scegli banco: 1 2 3 4

Elemento: 0 Larghezza: 0 Altezza: 0 Ritorno

SPOSTAMENTO ELEMENTO -> ELEMENTO

Quantità: 0

Da elemento: 0

A elemento: 0 Sposta

SPOSTAMENTO ELEMENTO -> BANCO N°

Elemento: 1 Larghezza: 6000

Numero lastra: 1 Altezza: 3210

Codice: 4+4acid

Commessa: _____

Invia a banco n° → 1 2 3 4

Questo sinottico è stato creato sia sotto richiesta di un cliente, ma soprattutto perchè facendo il collaudo della macchina è stato notato che a volte c'è la necessità di movimentare le lastre manualmente per integrare la lista di ottimizzazione del sinottico BANCO. Per questo in questa pagina sono presenti i comandi per tutti i tipi di movimentazioni possibili con in più la possibilità di cambiare l'informazione della lastra presente nel banco di taglio dal sinottico collegato CAMBIA LASTRA.

STORE Pro v.1.15 

Movimenti manuali

Banco n° Elemento Numero lastra

Codice Lastra

Codice Descrizione

Fornitore

Fornitore Descrizione Lotto

Quantità Larghezza Altezza

Cartone B.E.

Lastra rotta

Data

Spessore Peso

VII) REPORT

STORE Pro v.1.15

Report - Disponibile

Magazzino

	Elemento	Codice	Larghezza	Altezza	Quantità
1	1	4+4	6000	3210	1

Codice

Elemento

Liste disponibili

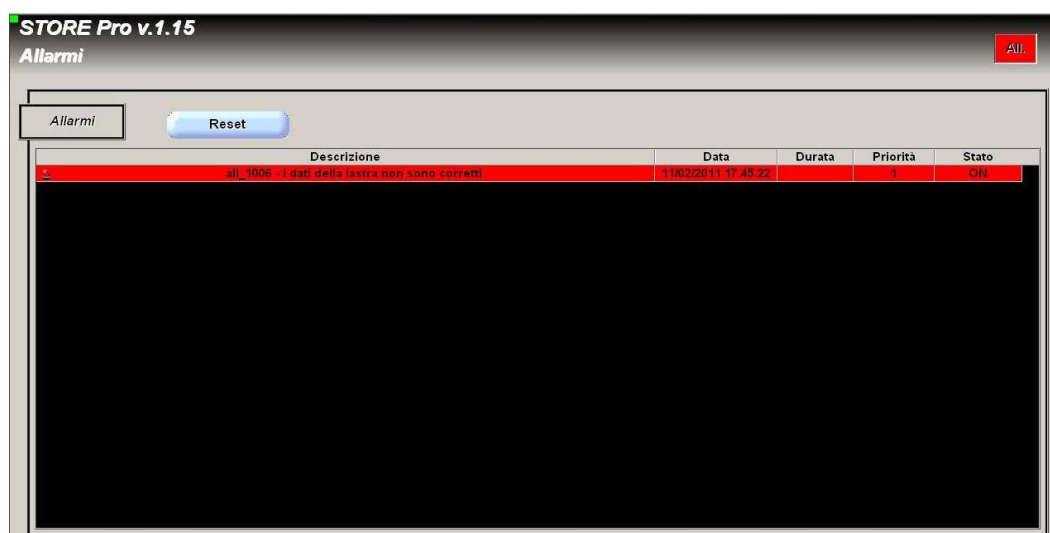
Il sinottico dei Report è molto importante per la gestione del magazzino perchè è in grado di mostrare all'operatore le liste di lastre secondo un criterio specifico, per esempio tutte le lastre presenti nel magazzino con un determinato codice.

I Report possibili sono:

- lista completa di tutto il magazzino
- lista di tutti gli elementi
- lista delle lastre disponibili
- liste dei banchi
- liste dei codici delle lastre
- lista dei fornitori
- lista in base ad un codice
- liste in base ad un elemento
- liste di ottimizzazione

Tutte queste informazioni si rivelano indispensabili per l'operatore che deve gestire il carico-scarico magazzino, la movimentazione delle lastre per il taglio, in pratica tutto il funzionamento di una vetreria.

VIII) ALLARMI



In questo sinottico vengono riportati tutti gli allarmi (attenzione, non gli eventi) che possono scatenarsi durante un procedimento. In questo modo l'operatore può prendere visione del tipo di errore verificatosi e prendere una decisione per riportare in funzionamento l'impianto. Dopo che è stata presa visione della notifica si può procedere all'acknowledgement, cioè la conferma al sistema che l'operatore si è reso conto dell'allarme attivo.

4.1.2 – Dichiarazione variabili

Esistono diversi tipi di variabile all'interno dello SCADA:

- Globali
- Interne allo script
- Globali di scambio
- Collegate ad eventi

Le variabili globali sono quelle variabili comuni a tutto il programma e possono essere collegate ad un pulsante, ad un campo, ad una spia all'interno del sinottico e servono per la gestione interna dello SCADA.

Le variabili interne allo script nascono e muoiono all'interno dello stesso, vengono definite all'inizio, utilizzate nel codice e una volta che lo script termina, finisce anche la vita di questa variabile che può essere riutilizzata in un altro procedimento.

Per variabili globali di scambio si intendono quelle variabili che sono definite come variabili globali, ma in più hanno nella loro definizione il collegamento ad una cella di memoria del plc. Queste variabili sono quelle che permettono l'effettivo scambio di informazioni tra SCADA e plc.

Infine le variabili collegate ad eventi sono tutte quelle variabili globali e globali di scambio che nel momento in cui cambiano stato, scatenano l'evento al quale sono collegate.

Queste ultime sono estremamente importanti al fine della comunicazione tra SCADA e PLC.

Esistono delle variabili globali di scambio all'interno del progetto, dedicate

esclusivamente all'attivazione di eventi e sono definite come SAP (SCADA a plc) e PAS (plc a SCADA). Ogni variabile di questo tipo è definita come word a 16 bit ed è in grado di evitare errori di comunicazione tramite una procedura ben precisa. Ogni volta che si attiva una comunicazione tra i dispositivi, quindi la necessità di avviare una procedura, il plc passa allo SCADA una variabile PAS con un solo bit alzato. Ogni PAS inviato attiva una procedura diversa all'interno dello scada, che può essere per esempio quella di cancellare una lastra dal magazzino oppure di prelievo eseguito, quindi di eliminare la prima riga della lista di ottimizzazione. Nel momento in cui lo SCADA riceve la variabile PAS, uno script ha il compito di verificare che non ci siano errori, quindi “ribalta” le variabili, cioè le rimanda al plc come SAP e il plc si incarica di controllare che le PAS inviate e le SAP ricevute siano uguali. Se il risultato è positivo allora il plc manderà conferma allo SCADA per poter iniziare il procedimento, se invece il risultato è negativo si scatena un allarme e la procedura non viene eseguita.

Per dialogare con il plc è indispensabile che lo SCADA sia in grado di comunicare con esso, che quindi esista un protocollo di comunicazione che traduca i comandi da SCADA a PLC e viceversa.

4.1.3 – Protocollo e comunicazione

La comunicazione tra SCADA e plc viene realizzata attraverso delle funzioni e il protocollo di comunicazione. Nel momento del salvataggio lo SCADA compila un file in formato XML con una lista organizzata dei dati da scambiare con il plc, lo scambio poi avverrà in sequenza in base alle caratteristiche del protocollo. Tale lista verrà ciclata continuamente dallo SCADA per fare in modo di monitorare continuamente lo stato di quelle variabili che, nel momento in cui cambiano di stato, scatenano un evento o un allarme (dando precedenza agli eventi). Il modo di comunicare appena descritto può essere suddiviso in varie funzioni ognuna delle quali ha sistemi per la correzione di eventuali errori. Le singole parti sono: il protocollo di comunicazione che, prevedendo all'interno della stringa un codice di controllo, è in grado di

identificare tutti gli errori di trasmissione impedendo di considerare validi quei dati che sono stati alterati nella linea di comunicazione per esempio da disturbi elettromagnetici.

Il protocollo di comunicazione è uno standard per lo scambio dati tra due dispositivi, in questo caso tra uno SCADA e un plc. Esistono diversi protocolli, uno per ogni casa costruttrice di plc e ne esistono di standardizzati. Considerato che entrambi i prodotti utilizzati dalla ditta Gioachin sono costruiti e forniti da Panasonic electric works, il protocollo in uso è quello ideato dalla stessa azienda: il Mewtocol.

Il Mewtocol è un protocollo ASCII che permette il multidroop, cioè la comunicazione da un master a più slave.

Un fatto rilevante è che il dispositivo SCADA al suo interno possiede già tutti i protocolli standard per la comunicazione, il più importante è il Modbus RTU che essendo libero, è stato implementato in tutti i dispositivi che prevedono una comunicazione. Anche lo SCADA della Panasonic è dotato di questo protocollo al suo interno, questo fa sì che sia possibile collegarlo ad ogni apparecchio rendendo il sistema di supervisione e controllo uno strumento molto potente che si adatta ai cambiamenti.

Un'altra funzione del metodo di comunicazione è il sistema pooling configurabile all'interno dello SCADA. Questo fa sì che per ogni serie di dati predisposti dalla lista precedentemente compilata dei dati da scambiare, possano essere eseguite più trasmissioni se i dati ricevuti risultano contenere degli errori. In questo modo un errore di trasmissione anche se già identificato dal protocollo, ripete più volte la stessa richiesta fino a quando il dato non arriva corretto. In questo modo tutti i dati vengono memorizzati nello SCADA in maniera corretta anche in presenza di disturbi che alterano la trasmissione. Il pooling fornisce informazione allo SCADA nei problemi di comunicazione qualora si superino il numero di tentativi impostati nella configurazione dello SCADA. Tale informazione può essere usata per prendere dei provvedimenti all'interno degli script gestendo per esempio gli allarmi, onde evitare di proseguire nelle elaborazioni se queste hanno bisogno di dati sempre aggiornati.

Attraverso tutte queste funzionalità sarà sempre possibile tenere sotto controllo il

traffico dei dati e la loro correttezza rendendo estremamente affidabile il sistema in quanto in grado di identificare qualsiasi problema nell'insieme della comunicazione.

4.1.4 - Database

Il database è stato creato in formato mdb (Microsoft DataBase) e gestito con Microsoft Access. Le tabelle sono 30 più tre tabelle create da Panaway per la gestione dei messaggi di sistema, le comunicazioni tra dispositivi e per gli allarmi.

Le tabelle sono:

- *banco 1, 2, 3, 4* e contengono le liste di ottimizzazione di ogni banco
- *bancotemp* è una tabella di appoggio per la gestione interna delle liste dei banchi
- *C0, 1, 2, 3, 4* servono per memorizzare le operazioni di ricerca della lastra nel procedimento di prelievo automatico.
- *confronta* serve come tabella di appoggio per le operazioni di ricerca automatica della lastra da prelevare, per confrontare quale sia la lastra che deve essere prelevata nel caso in cui ne esista più di una
- *copiabanco* è una tabella di appoggio per la gestione dei banchi
- *copiadilastra*
- *copiadispostamenti* che serve per la gestione interna degli spostamenti lastra dal sinottico SPOSTAMENTI
- *elemento* è la tabella con il riepilogo di tutti gli elementi del magazzino
- *fornitore* è la tabella dei fornitori
- *lastra* è l'elenco di tutti i tipi di lastre che è possibile caricare nel magazzino
- *lastra_b* è l'elenco di tutti i tipi di lastre con dettagli in più rispetto a *lastra*
- *memoriabanco* è la tabella con al massimo 4 righe che tiene in memoria la lastra presente nel banco di taglio
- *ordinabancotemp* è la tabella di appoggio per la gestione delle liste di ottimizzazione ordinata per codice, larghezza, altezza

- *ordinalastra* è la tabella che raggruppa tutte le lastre da riordinare (per esempio quando viene cancellata una lastra all'interno di un pacco)
- *ordinapacchi* serve per la gestione dei pacchi di lastre
- *ordinaultimalastra* è la tabella che raccoglie tutte le ultime lastre degli elementi, quindi quelle disponibili, più tutte le lastre del Restore
- *pacchi* è la lista di tutti i pacchi di lastre
- *spostamenti* raccoglie la lista degli spostamenti del sinottico SPOSTAMENTI
- *testicodici* raccoglie tutti i codici con la relativa descrizione delle lastre che si possono caricare in magazzino
- *ultimalastra* è una tabella di appoggio che raccoglie tutte le ultimelastre degli elementi più il Restore
- *ultimalastra_b* è un'altra tabella di appoggio che raccoglie i dati di tutte le ultimelastre degli elementi, ma che a differenza di *ultimalastra*, al suo interno ha più colonne quindi più informazioni. Quest tabella è il risultato dell'intersezione di delle tabelle *lastra* e *elemento*.

Per poter scrivere dati sul database o estrarre dati da esso, è necessario stabilire una connessione. La connessione al database è possibile grazie agli script, quindi attraverso i comandi di Visul Basic.

Per attivare una connessione quindi è necessario richiamare negli script, nel momento in cui si vuole compiere un'azione di scrittura o lettura, l'oggetto Recordset. Questo oggetto contiene tutti i dati che vengono letti da un database o che si intende inviare ad esso e può includere diverse righe e colonne di dati. Ogni riga è un record e ogni colonna è un campo interno al record; è possibile accedere solo a una riga alla volta (la cosiddetta riga corrente o record corrente) ed è possibile spostarsi in un Recordset cambiando il record corrente.

Tramite gli oggetti Recordset è possibile creare un oggetto di tipi Recordset senza essere connessi a un database oppure recuperare un Recordset da un database, chiudere la connessione, modificare i dati presenti nel Recordset e infine ristabilire la connessione per trasmettere gli aggiornamenti.

```

Dim Conn1 As New ADODB.Connection
Dim Rs1 As New ADODB.Recordset
Dim sQuery1 As String
Conn1.ConnectionString=sAccessConnect(*)
Conn1.Open
sQuery1= "SELECT * FROM elemento ORDER BY
elemento ASC;"
Rs1.CursorType=adOpenKeyset
Rs1.LockType=adLockOptimistic
Rs1.Open sQuery1, Conn1, , , adCmdText
[...]
Rs1.Close
Set Rs1=Nothing
Conn1.Close
Set Conn1=Nothing

```

(*) sAccessConnect è definito nello Script di inizializzazione come:
sAccessConnect="Provider=Microsoft.Jet.OLEDB.4.0;Data" &
"Source=C:\STORE Pro\magazzino.mdb; User ID=Admin;Password=";

La proprietà più significativa dell'oggetto Recordset è probabilmente la proprietà Source, una stringa contenente il nome della tabella, il nome della stored procedurale o il testo della query SQL utilizzata.

Importante per la connessione è anche il cursore, cioè un set di valori che rappresenta il risultato di una query. I cursori possono contenere i dati effettivi o solo puntatori ai record del database, il meccanismo che recupera i dati è comunque trasparente. E' possibile specificare dove deve essere creato il cursore, il tipo di cursore e il tipo di lock.

La proprietà CursorType indica quale tipo di cursore deve essere creato e può rappresentare una delle seguenti costanti: 0-adOpenForwardOnly, 1-adOpenKeyset, 2-adOpenDynamic o 3-adOpenStatic.

I cursori keyset, usati negli script di StorePro, consistono di un gruppo di puntatori ai dati . Ogni qualvolta vengono richiesti i dati, un record utilizza il puntatore per leggere il valore effettivo. I cursori keyset vengono aggiornati in modo automatico quando l'utente aggiunge o annulla un record oppure apporta modifiche ad un qualsiasi record che si trovi già nel Recordset.

Il dato adCmdText indica in che modo debba essere interpretata la stringa in CommandText, in questo caso indica il testo di una query SQL.

Ogni script che ha necessità di scambiare dati con il database quindi userà questa procedura di apertura e chiusura della connessione.

Ci sono oggetti però come le griglie visualizzate nei sinottici BANCO1, 2, 3, 4, SPOSTAMENTI e REPORT che necessitano di un collegamento ODBC per l'utilizzo dei Drivers per la comunicazione con il database oltre ai comandi di Visual basic. Anche in questo caso in ogni script è presente l'apertura della comunicazione e la sua chiusura.

```
Dim screen As SynopticCmdTarget
Dim griglia As GridWndCmdTarget
Set screen=GetSynopticInterface("report")
Set
griglia=scree.GetSubObject("vista4").GetObjectInterface
griglia.Query="SELECT * FROM banco1 ORDER BY indice
ASC;"
griglia.Refresh()
Set screen=Nothing
Set griglia=Nothing
```

L'oggetto SynopticCmdTarget permette di accedere alle proprietà ed ai metodi specifici del sinottico contenitore del simbolo, in questo caso la griglia.

La funzione GridWNDCmdTarget restituisce il valore booleano True se il contenuto della cella referenziata con numero di riga e il numero di colonna è editabile.

L'oggetto GetSynopticInterface permette di accedere ai metodi e alle proprietà descritte nella SynopticCmdTarget da risorsa script.

Alla fine dello script viene chiusa la connessione e la griglia richiamata resta visualizzata nel sinottico. Alla successiva richiesta di visualizzazione di una griglia la connessione verrà riaperta e la griglia verrà aggiornata con la nuova query. Il fatto di aprire e riaprire connessioni di questo tipo però comporta una spesa notevole per StorePro in termini di complessità computazionale.

5 – Miglioramenti

Sicuramente i miglioramenti che si possono apportare al progetto sono molti, partendo dalla grafica fino al database.

Le modifiche vengono fatte principalmente per due motivi: perchè è sorto un problema nel progetto oppure perchè il cliente o il mercato hanno avanzato delle richieste.

Esistono infiniti miglioramenti da fare nel programma e sarebbe impossibile individuarli tutti anche perchè molti derivano direttamente dalle richieste del cliente. Ho individuato modifiche da apportare a StorePro che derivano da problemi interni al progetto, richieste dal cliente e dal mercato.

5.1 – Problemi del progetto

Un problema che non è stato precedentemente gestito riguarda il database, sia la struttura che i problemi che si possono creare durante il suo utilizzo. La prima modifica riguarda la riduzione delle tabelle. Esistono diverse tabelle che contengono gli stessi dati che è possibile trovare in altre, quindi bisognerebbe individuare tali tabelle ed eliminarle. Questo procedimento però non è facile perchè richiede uno sconvolgimento dell'intero progetto. Gli script infatti usano tali tabelle per archiviare dati durante i procedimenti per la gestione interna ed esterna del programma, una modifica del genere comporterebbe la riscrittura di quasi tutti gli script, di conseguenza si andrebbe a cambiare il modo di gestione del database ideato all'inizio. Se si intende procedere con questa modifica quindi bisogna pensare di rifare il programma da capo e facendo così una progettazione ex novo di StorePro.

Un'altra modifica del database riguarda le sue dimensioni. Quando una vetreria si trova a gestire una notevole quantità di lastre, quindi inserisce e cancella continuamente dati dal database, questo si gonfia sempre di più diventando sempre più pesante e rallentando in questo modo tutto il programma. Per ovviare a questo problema bisogna procedere con la compattazione del database. Questo procedimento

però, per come è organizzato il programma, spetta interamente all'operatore che deve recuperare il file magazzino.mdb dal computer e deve avere le conoscenze per compiere questa operazione. Esiste uno script in visual basic che, mandato in esecuzione, è in grado di compattare da solo il database. Se tale procedimento venisse avviato ad ogni accensione della macchina o al suo spegnimento (una volta al giorno) si potrebbe far risparmiare tempo all'operatore che non avrebbe più la necessità di conoscere il procedimento esterno per compattare il database, ma gli basterebbe accendere o spegnere il programma. Questo tipo di modifica segue esattamente le linee guida descritte nel capitolo 3 riguardante la “fiducia” che l'operatore deve avere nel programma per lavorare meglio e quindi non commettere errori che possono portare ad incidenti.

L'ultima modifica per il progetto riguarda la connessione ODBC per la visualizzazione delle griglie. Il fatto che per ogni griglia venga attivata una connessione non comporta di per sé un grande problema di velocità di visualizzazione visto che il programma gira su un computer, però si potrebbe pensare di avere un solo script che apre la connessione una volta per tutte e si chiude alla chiusura del programma, così l'unica azione da compiere resta l'aggiornamento dei dati da visualizzare. In più nel sinottico REPORT per visualizzare 10 tabelle, sono stati sovrapposti 10 oggetti tabella che si attivano o disattivano a seconda del report richiesto. Dopo aver chiesto informazioni direttamente alla casa costruttrice di Panaway, cioè la Progea, abbiamo appreso che esiste la possibilità di scrivere un solo script che gestisce una sola tabella in modo da visualizzare diversi report senza doverne sovrapporre 10 come è stato fatto. In questa maniera si riduce la complessità computazionale, snellendo così l'intero programma e contribuendo a renderlo più veloce.

5.2 – Richieste dal cliente

Dai clienti arrivano continuamente richieste per adattare il programma al proprio modo di lavorare, quindi secondo la filosofia di chi gestisce la parte logistica delle

vetrerie. Una di queste riguarda le liste degli spostamenti. È possibile che l'operatore decida di spostare tutte le lastre da un elemento all'altro, quando questo procedimento inizia la macchina sposta una lastra alla volta e questo significa che potrebbe impiegare diverso tempo prima di finire il procedimento. Per questo il programma non blocca le altre operazioni mentre avvengono gli spostamenti. Esiste la possibilità però che finché la macchina è impegnata a compiere uno spostamento, un'altro operatore decida di avviare la lista di ottimizzazione. Se in tale lista è presente una lastra che la macchina si è incaricata di spostare, è possibile che la vada effettivamente a prelevare prima che l'operazione di spostamento sia finita. Se non vengono gestite queste azioni, la macchina si troverà in una situazione di allarme perché il procedimento di spostamento riconosce un errore nel fatto di avere una lastra in meno da spostare. La soluzione è quella di creare uno script che sia in grado di cancellare una lastra dalla quantità delle lastre da spostare, nel momento in cui un operatore ne prelevi una dall'elemento di partenza dello spostamento.

Un'altra modifica molto interessante richiesta dal cliente riguarda la creazione delle liste di ottimizzazione. Ricordiamo che le liste di ottimizzazione vengono create da un software esterno che le passa a StorePro che a sua volta si incarica di trasformare tale lista in formato testo. Accade che le lastre disponibili nel magazzino, cioè l'ultima lastra di ogni elemento e tutte le lastre del Restore, non soddisfino la lista di ottimizzazione quindi l'operatore prima di mandare in esecuzione il procedimento deve individuare le lastre che servono alla lista e andare a spostare manualmente le lastre le lastre che non servono nel magazzino in modo da avere disponibili quelle necessarie al compimento del lavoro. Per semplificare questo procedimento dovrebbe essere possibile per il programma nel momento in cui si carica la lista in un banco, fare il controllo di tutte le lastre del magazzino, non solo delle disponibili. In questo modo il programma potrebbe proporre due alternative all'operatore che potrebbe decidere di continuare a gestire il magazzino come prima, cioè spostando le lastre manualmente prima di mandare in esecuzione la lista, oppure potrebbe accettare di compilare automaticamente una lista di spostamenti che renderebbero disponibili tutte le lastre necessarie. In questo modo si risparmierebbe del lavoro all'operatore e tutto il tempo

che avrebbe dovuto dedicare al controllo delle liste e alla movimentazione manuale delle lastre viene notevolmente ridotto.

Nelle grandi vetrerie che arrivano ad avere 4 banchi da taglio, capita che ci sia un addetto per ognuno di essi. Questo comporta che tutti gli operatori si trovino ad usare StorePro per movimentare le lastre. Si crea così la necessità di riservare le lastre per operatore, cioè nel momento in cui un operatore importa una lista nel proprio banco, dovrebbe essere impossibile per l'operatore seguente prelevare le stesse lastre. Questo significa che il programma, quando è presente una lista caricata nel banco, deve rendere quelle lastre non più disponibili per gli altri operatori, anche se non sono ancora state effettivamente prelevate.

5.3 – Richieste dal mercato

Le richieste dal mercato si traducono nelle capacità del progettista del programma, questo perchè non esistono tanti concorrenti per un programma come StorePro perchè sono molto poche le aziende che hanno sviluppato un programma del genere dedicato alle vetrerie. Si tratta comunque di trovare delle modifiche che possano essere utili a tutti i clienti, indipendentemente dalla loro organizzazione.

La prima modifica riguarda gli avvisi sullo stato di avanzamento dei procedimenti oppure sul procedimento terminato. Come descritto nel capitolo 3 questi tipi di messaggi sono molto importanti per gli operatori che magari non sono in grado di vedere fisicamente l'operazione, ma riguarda anche procedimenti interni a StorePro che l'operatore non può vedere. Inizialmente avevo pensato ai message box, cioè una finestra che apparre sopra tutte le altre e che avvisa sullo stato. Questo tipo di scelta però comporta un problema perchè i message box necessitano di una conferma per poter continuare al procedimento e spesso l'operatore dopo aver mandato in esecuzione la lista, si allontana dallo schermo per continuare il suo lavoro. Quindi usando i message box si rischierebbe di lasciare la macchina bloccata soltanto per non aver dato conferma di presa visione del messaggio. Per ovviare a questo problema esiste un modo che è quello di usare i messaggi di pop-up resi disponibili da Panaway. I

messaggi di pop-up sono avvisi che compaiono sullo schermo, ma che scompaiono in automatico dopo un tempo preimpostato. Ecco che in questo modo viene completamente eliminato ogni problema legato ai message box.

Una delle modifiche più importanti che fanno parte delle richieste dal mercato è sicuramente la gestione delle scorte. Non è previsto per ora un indice che permetta di impostare una quantità minima di scorta per le lastre, in questo modo sta all'operatore o al responsabile di magazzino la gestione delle scorte. Se invece venisse inserite questo dato, sarebbe possibile avvertire l'operatore nel momento in cui venisse raggiunto il numero minimo di lastre di un determinato tipo. Per fare questo però StorePro deve compiere al suo interno un confronto tra tutte le lastre del magazzino e tutte le lastre caricate nelle liste di ottimizzazione anche se queste non sono ancora state eseguite. In questo modo si anticipa l'informazione all'operatore che può procedere con il riordino tempestivamente.

CONCLUSIONI

Le modifiche descritte nel capitolo precedente sono solo alcune di quelle che si possono fare perchè il progetto ha infinite possibilità di essere migliorato. L'obiettivo iniziale era quello di proporre delle modifiche per rendere StorePro migliore oltre che più adatto ai cambiamenti. Con tutte le modifiche proposte questo obiettivo si può dire raggiunto perchè i miglioramenti riguardano tutte le parti del progetto e se tutte le parti sono strutturate in modo migliore, allora sarà anche più facile compiere dei cambiamenti in futuro.

Fermo restando che il progetto ha innumerevoli potenzialità, una risorsa che bisogna sempre tener conto per migliorare il programma è sicuramente l'ascolto delle esigenze dell'operatore che usa effettivamente StorePro. Esso infatti è una fonte preziosissima di idee per lo sviluppo del programma, anche se poi resta al programmatore la gestione interna del programma. L'operatore infatti è in centro da cui partire per progettare un sistema SCADA perchè è lui la persona di riferimento da aiutare nel suo lavoro, rendendo ogni operazione semplice e il più possibile controllata.

Al programmatore resta quindi la responsabilità di mettere in atto tutte le proprie idee e le richieste che arrivano dai clienti e questa è la parte più difficile, ma anche quella che da più soddisfazione: non c'è niente come vedere una macchina funzionante e sapere che è il proprio lavoro che ha portato quei risultati.

Per poter svolgere tutte queste azioni diventa quindi necessario, se non indispensabile, conoscere il programma in tutte le sue parti, sinottico per sinottico, script per script.

Solo così, pur commettendo errori, sarà possibile correggerli.

Bibliografia

- [1] Wikipedia the free encyclopedia
- [2] Stefano Bimbo e Enrico Colaiacovo. *Sistemi SCADA – supervisory control and data acquisition*. APOGEO 2006
- [3] Dispense corso Panaway presso Panasonic Electric Works
- [4] Standard IEC 61131-3
- [5] Manuale Panaway
- [6] Francesco Balena. *Programmare Microsoft Visual Basic 6*. MONDADORI 1999
- [7] Paolo Atzeni, Stefano Ceri, Stefano Paraboschi e Riccardo Torlone. *Basi di dati – Modelli e linguaggi di interrogazione*. McGraw-Hill 2002