

Manuale del Corso introduttivo IBMi (As400) 2023

Claudio Neroni

I capitoli ">NC" sono "Non Completi"

Sommario

F1.	Premessa.....	2
2.	Scarichi dal sito neroni.it.....	3
3.	Uso dei sorgenti.....	4
4.	Valori di difetto.....	5
5.	Prompter & Help.....	6
6.	Interfacce.....	7
7.	Ricerca documentazione IBMi.....	8
8.	Collegamento ad IBMi.....	10
9.	Creazione ambiente di sviluppo del corso.....	15
10.	Utilizzo del PDM (Programming Development Manager).....	18
11.	Impostazione della lista librerie.....	26
12.	Gestione delle autorità.....	28
13.	Cambio lingua.....	31
14.	Descrizione lavori.....	32
15.	Definizione di file fisici e logici con DDS per uno schema Testata-Righe.....	33
16.	Schema Anagrafica-Tabelle.....	40
17.	Definizione di tabelle e indici con DDL (SQL).....	44
18.	Scaricare dati sotto forma di sorgente.....	47
19.	Processo di stampa Testata-Righe con CLLE, RPGLE e DSPF.....	49
20.	Processo di interrogazione Testata-Righe RPGLE e DSPF.....	50
21.	Processo di gestione Anagrafica-Tabelle.....	51
22.	Subfile e sue tipologie.....	83
23.	Subfile di visualizzazione.....	84
24.	Subfile di visualizzazione con massimizzazione dell'uso di subroutine.....	95
25.	Subfile di scelta.....	102
26.	Subfile quadrato con scelta, decodifica e refresh.....	118
27.	Subfile quadrato con scelta in finestra.....	126
28.	Gestione Anagrafica-Tabelle chiama Finestra di scelta codici.....	134
29.	Printer file>NC.....	145
30.	Salvataggi e ripristini.....	146
31.	Trasporto di file di salvataggio tra IBMi e PC (e viceversa) via FTP.....	149
32.	Tool jSAVF.....	150
33.	Giornale.....	151
34.	Debug.....	154
35.	Sottosistemi e code.....	156
36.	Gestione dei lavori.....	159
37.	Integrated file system (IFS).....	167
38.	Sorgenti e IFS.....	172
39.	Ulteriori nozioni di database>NC.....	173
40.	Elenco istruzioni di calcolo in RPGLE>NC.....	174
41.	Tecniche di modularizzazione della programmazione su IBMi>NC.....	175
42.	SQL>NC.....	177
43.	SQL in RPGLE>NC.....	178
44.	OPNQRYF (Open Query File)>NC.....	179
45.	RPG Free.....	180
46.	Strutture dei programmi interattivi>NC.....	193
47.	Strutture dei programmi batch>NC.....	194
48.	Ftp (File Transfer Protocol)>NC.....	195
49.	Aree dati>NC.....	196
50.	Cenni ad altri tipi di oggetto>NC.....	197
51.	Reperimento di Tool dal web>NC.....	198
52.	Cobol>NC.....	199
53.	Link.....	200
54.	Esercitazione finale 1: Convogli ferroviari.....	201
55.	Esercitazione finale 2: Orario scolastico.....	202

1. Premessa

- 1.1. Il corrente manuale contiene le tracce di un corso di utilizzo del software di base IBMi e della sua programmazione adatto a chi inizia (anche da zero). È costituito da un campionamento di argomenti mirati a rendere autosufficiente il programmatore principiante nel successivo utilizzo e studio di IBMi.
- 1.2. Per ottenere il migliore avviamento alla programmazione, subito dopo il corso occorre comunque un periodo di tirocinio a supporto di un programmatore esperto. Auspicabile un anno di affiancamento prima di una sufficiente autonomia.
- 1.3. Gli argomenti non sono trattati nel manuale in modo sistematico né nell'ordine di esposizione del corso che presenterà per primi dei semplici esempi per poi esaurire tutta la presente lista nel modo più adatto alle necessità rilevate nella prima fase. Di seguito un esempio di avvio del corso.
 - 1.3.1. Innanzitutto, nozione di base su bit, byte, record, file, tracciati, database, tipi di oggetti, programmi, sorgenti, compilazioni, "processi".
 - 1.3.2. Uno spiegone di DDS e DDL elementari e qualche esercizio di costruzione database (tabelle, anagrafiche, testate, righe).
 - 1.3.3. Esempi di stampa file, visualizzazione con navigazione tra videate, gestione anagrafica semplificata.
 - 1.3.4. Utilizzo delle strutture logiche per i flussi dei programmi.
 - 1.3.5. Attività di debug per verificare e correggere i primi programmi.
 - 1.3.6. Programmi CL di gestione processi batch.
 - 1.3.7. Esempi di processi più articolati e basati su routine, programmi e moduli riutilizzabili.
 - 1.3.8. Solo a questo punto trattazione delle principali istruzioni DDS, DDL, RPGLE, CLLE.
 - 1.3.9. Altri esempi di soluzione modulare ai problemi tipici di programmazione per una gestione documento a tre livelli (testata, righe, sottorighe, altri tipi di righe).
- 1.4. Tempificazione del corso.
 - 1.4.1. Il corso può consistere di 200 ore a mezze giornate a tre per settimana per la parte Sistema-DDS-DDL-CLLE-RPGLE.
 - 1.4.2. Per velocizzare le sessioni potranno diventare giornaliere e, invece che di 4 ore, potranno essere di 5 o, al massimo, 6 ore.
 - 1.4.3. Nelle 200 ore è previsto un test di 3 mezze giornate per la fabbricazione di un processo interattivo, compreso un test esemplificativo iniziale.
- 1.5. Il manuale si fa carico di sintetizzare gli argomenti trattati per consolidarli e ricordarli nel tempo.
- 1.6. Nota sulla terminologia. Dalla notte dei tempi, per "procedura" si intendeva l'insieme dei programmi che realizzano una elaborazione complessa. Oggi, invece, nell'ambito delle strutture ILE, il termine "procedura" definisce una certa tipologia di sezione di programma e la correlata figura di chiamata composta da alcune specifiche istruzioni. Per non creare confusione, userò "processo" per il primo significato.

2. Scarichi dal sito neroni.it

- 2.1. L'ultima versione del presente manuale è scaricabile dal link
www.neroni.it/CORSO2023/Corso-As400-2023.docx
- 2.2. Tutti i sorgenti menzionati nel manuale sono reperibili via web all'indirizzo seguente, sostituendo la parola "xxxxxxxxxx" con il nome del sorgente.
www.neroni.it/CORSO2023/A-CORSO.CORSO/xxxxxxxxxx.txt
Ad esempio, se si cerca il sorgente del file fisico CNA1, si userà il link:
www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1.txt
- 2.3. Al link seguente è reperibile l'elenco di tutti i sorgenti scaricabili singolarmente come file di testo.
www.neroni.it/CORSO2023/A-CORSO.CORSO/AAAMBRLIST.txt
- 2.4. I sorgenti sono scaricabili anche come file di salvataggio al release V7R2 dal link:
www.neroni.it/CORSO2023/CORSO.NERONIS.bin
Visto che il mio sito non permette lo scarico di un ".savf", il file di salvataggio ha l'estensione ".bin". Scaricatolo, cambia l'estensione del file da ".bin" a ".savf". Per leggerne e addirittura copiare i sorgenti contenuti nel file di salvataggio, vedi il paragrafo [Tool jSAVF](#).

3. Uso dei sorgenti

- 3.1. Questo paragrafo sulle regole di ingaggio dei sorgenti è fuori sequenza: consideralo quando inizi a creare e modificare i sorgenti.
- 3.2. Attenzione: i programmi che permettono l'edit (modifica) dei sorgenti, e cioè il SEU del PDM e l'LPEX dell'RDI, allocano i membri sorgenti che usano.
- 3.3. Ma, mentre dal SEU si esce sempre a fine modifica di un membro perché il singolo lavoro interattivo apre un singolo membro alla volta...
- 3.4. ...dall'LPEX, che può aprire contemporaneamente molti membri in modifica e permette di mantenerli aperti anche durante la sconnessione di fine lavoro, si tende a dimenticare l'uscita.
- 3.5. Una singola sessione RDI sul PC ha inoltre bisogno di collaborare con una decina di lavori suoi figli a bordo dell'IBMi.
- 3.6. Oltre alla diversa logica di occupazione di LPEX, quindi, succede anche che i lavori figli sull'IBMi restino appesi dopo lo scollegamento o la caduta del lavoro padre sul PC.
- 3.7. Tutto questo impedisce ad altri di usare e salvare membri anche mentre chi li occupa è scollegato.
- 3.8. Ne segue che per continuare occorre abbattere i lavori altrui con perdita di tempo ma soprattutto con rischi aggiuntivi di perdita di modifiche.
- 3.9. Non tenere quindi allocati sorgenti in modifica al di fuori del necessario!

4. Valori di difetto

- 4.1. Il software di sistema prevede spesso la scelta di opzioni alternative tra loro.
- 4.2. Il sistema assume in proprio la più parte delle scelte che l'utente non gli fornisce.
- 4.3. Dove si richiede un valore non basta quindi avere la possibilità di fornirlo ma occorre che, dove non sia dato esplicitamente, il sistema sia messo in grado di usare un valore detto "di difetto".
- 4.4. Due gli argomenti che soprattutto lo richiedono, anche se non i soli.
 - 4.4.1. I valori di sistema che regolano i comportamenti del software di base verso tutti gli utenti.
 - 4.4.2. I moltissimi parametri presenti nei comandi del Control Language.
- 4.5. Valori di sistema.
 - 4.5.1. Visualizzabili con il comando `DSPSYSVAL`.
 - 4.5.2. L'elenco dei valori previsti si ottiene con la sequenza `DSPSYSVAL + F4 +F4`
 - 4.5.3. L'help dei valori previsti si ottiene con la sequenza `DSPSYSVAL + F1`
 - 4.5.4. Ad esempio si esamini il risultato di `DSPSYSVAL SYSVAL(QHSTLOGSIZ)`

```
Visualizzazione del valore di sistema
Val. di sistema . . . . : QHSTLOGSIZ
Descr. . . . . : Num. mass. record cronologico
Numero di record . . . . . : 5000      1-10000000, *DAILY
```

- 4.5.5. Osserva che il valore è già impostato ma che potrebbe essere cambiato con il comando `CHGSYSVAL SYSVAL(QHSTLOGSIZ) VALUE(10000)` che però è riservato al sistemista.
- 4.6. Valori di parametri dei comandi CL.
 - 4.6.1. Visualizzabili, ad esempio per il comando `WRKJOB`, con la sequenza `WRKJOB + F4`

```
Gestione lavoro (WRKJOB)
Immettere le scelte e premere Invio.
Nome lavoro . . . . . JOB      *
Utente . . . . .
Numero . . . . .
Emissione . . . . . OUTPUT    *
Opzione . . . . . OPTION     *SELECT
```

- 4.6.2. Osserva che tutti i parametri, pur non essendo stati digitati, vengono presentati con dei valori di difetto che, premendo Invio, verranno passati dal comando al programma esecutore del comando.

5. Prompter & Help

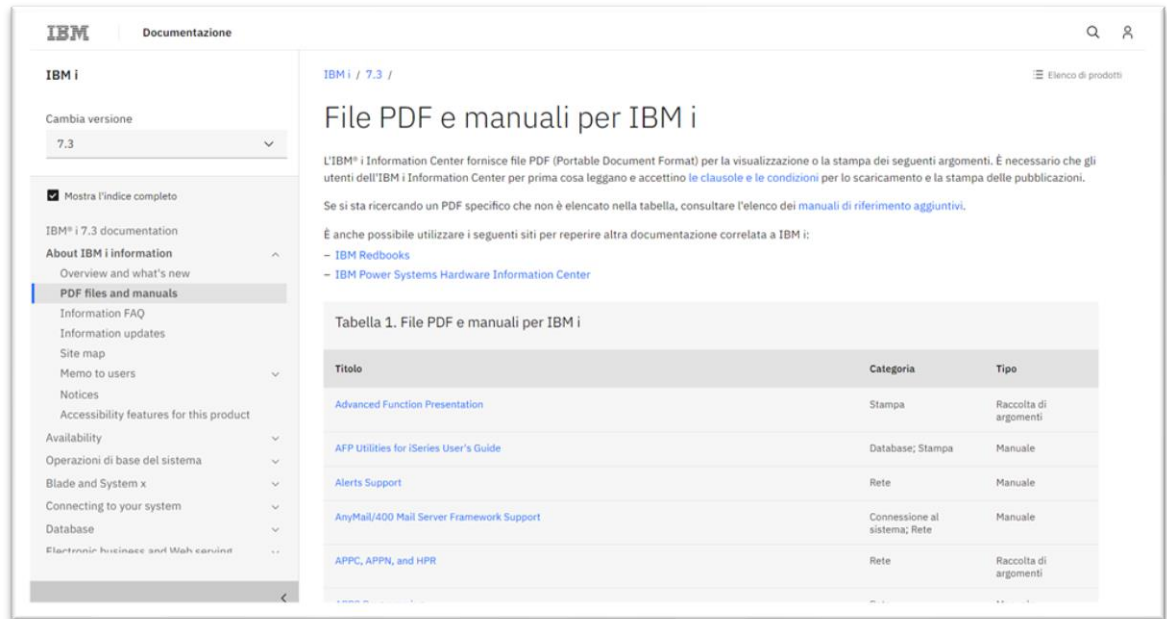
- 5.1. I programmi di utilità su IBMi, sia quelli in dotazione al sistema che quelli aggiunti da altri, sono chiamati attraverso call specializzate che suggeriscono, aggiustano e controllano i parametri da passare ai programmi stessi. Tali call specialistiche si chiamano "Comandi" e godono a loro volta di supporti specializzati. I programmi chiamati dai comandi si definiscono invece "Command Processing Program".
- 5.2. I comandi possono usare la forma posizionale come le call di ogni altro linguaggio, oppure la forma con parole chiavi, autoesplicativa ed ulteriormente assistita da prompter ed help.
- 5.3. I comandi del Control Language di IBMi sono assistiti sul sistema...
 - 5.3.1. Da un servizio, chiamato da F4, di prompter (suggeritore) che governa il passaggio parametri e funge da promemoria dei possibili valori.
 - 5.3.2. Da un servizio, chiamato da F1, di help on line (aiuto in linea) che fornisce testi esplicativi ed esempi sui parametri e sui possibili valori.
- 5.4. Prompter ed help sono le caratteristiche di maggiore originalità di questo linguaggio di controllo.
- 5.5. L'help ha fatto molti proseliti su altri linguaggi, ma raramente è, come qui, sensibile al contesto.
- 5.6. Il prompter, invece, è rimasto figlio unico di madre vedova.
 - 5.6.1. Non solo non ha fatto proseliti, ma anche non si è diffuso nemmeno sul resto del software IBMi.
 - 5.6.2. Le procedure e le built in function dell'RPG ILE sono nate senza niente del genere e per digitarle correttamente occorre ancora consultare paradigmi statici distraendo l'attenzione dalla minutazione in corso.
 - 5.6.3. Stessa fregatura per l'SQL a bordo dell'IBMi che ha implementato solo un prompter parziale, inadatto ai casi reali di uso produttivo.
- 5.7. Il prompter e l'help in linea, cioè, permettono di scrivere Control Language senza perdere tempo in consultazioni mentre l'RPG (anche il free) e l'SQL vanno recitati a memoria o niente.
- 5.8. Si noti che il supporto comandi, con l'aggiunta di qualche regola di implementazione dei comandi, si fa anche carico della evoluzione futura delle chiamate ai programmi esecutivi perché non richiede la modifica dei programmi CL compilati usando le vecchie edizioni dei comandi poi modificati.

6. Interfacce

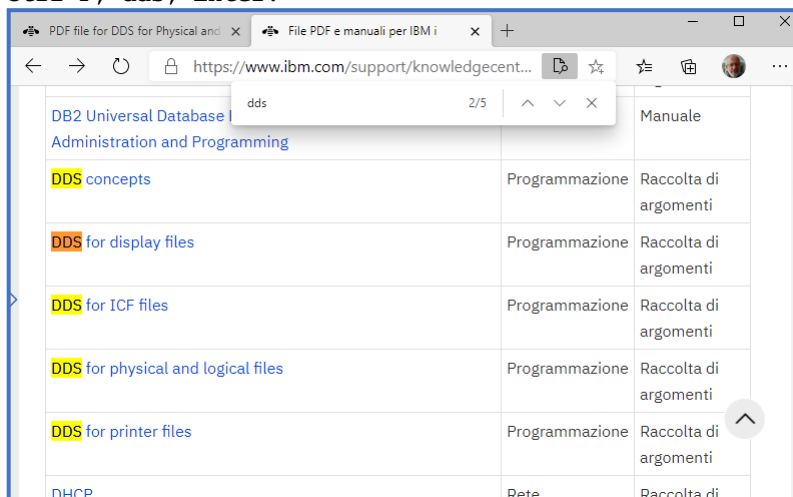
- 6.1. Il corso assume l'uso dei seguenti programmi di interfaccia forniti dal software di sistema.
- 6.2. **QCMD** (Immissione comandi) per i comandi.
 - 6.2.1. È il programma esecutore nativo e fondamentale di tutti i lavori sul sistema.
 - 6.2.2. Interfaccia sia interattiva che batch sempre presente in lista di chiamata dei lavori.
 - 6.2.3. Interfaccia interattiva principale per il programmatore e il sistemista.
 - 6.2.4. Riceve le richieste sotto forma di comandi eseguibili tramite la coda messaggi del lavoro *EXT (external) nella quale va a sfrugliare prima di presentare il proprio video comandi (in interattivo) o di andare a fine lavoro (in batch).
 - 6.2.5. Nell'interattivo, le richieste da elaborare sono accodate dai comandi chiamati dall'utente tramite il video comandi stesso o da un programma eseguito precedentemente nello stesso lavoro.
 - 6.2.6. Nell'interattivo, i messaggi di richiesta alla coda esterna sono accodati dal video comandi o dall'esecuzione di un SNDPGMMSG (Send Program Message) eseguito da un programma all'interno del lavoro.
 - 6.2.7. Nel batch, le richieste da elaborare sono accodate dal comando di sottomissione del lavoro o da un programma eseguito precedentemente nello stesso lavoro.
 - 6.2.8. Nel batch, i messaggi di richiesta alla coda esterna sono accodati:
 - 6.2.8.1. dai comandi SBMJOB (Submit Job) o SMBDBJOB (Submit Database Job) eseguiti da un altro lavoro (sottomettitore) prima che il lavoro batch nasca
 - 6.2.8.2. o dall'esecuzione di un SNDPGMMSG (Send Program Message) eseguito da un programma all'interno del lavoro stesso.
 - 6.2.9. Se richiesto, interattivamente, chiama a sua volta il prompter e l'help sui comandi.
 - 6.2.10. QCMD è da preferire interattivamente alle righe comandi presenti su ogni menù di sistema a causa della migliore evidenziazione del log dopo l'esecuzione di un comando.
 - 6.2.11. Logga le attività documentando i messaggi generati dai programmi chiamati nel lavoro.
 - 6.2.12. Stampa il log a fine lavoro.
 - 6.2.13. F10 permette due livelli commutabili di dettaglio nella visualizzazione.
 - 6.2.14. Strutturalmente in grado di gestire qualsiasi comando anche creato dal programmatore.
- 6.3. **SEU** (Source Entry Utility) per la gestione dei sorgenti.
 - 6.3.1. Fornito gratuitamente a corredo del software di base ma non più implementato sicuramente dopo il rilascio V6R1.
 - 6.3.2. Funziona ma il controllo di sintassi segnala errore sulle specifiche per le istruzioni RPG nate dopo la dismissione. Può essere ancora usato con qualche fastidio su queste ultime.
 - 6.3.3. IBM vende a caro prezzo (962€ per utente all'anno) il prodotto aggiuntivo RDI (IBM Rational Developer for i) che contiene LPEX sostitutivo del vecchio SEU e basato su programmi PC. Il sito Faq400, al link [Rational Rdi - Versioni, costi e Products Code \(IT\) - BlogFaq400](#), indica una licenza primo anno di 1000€ e un rinnovo annuale di 200€ ma il sito IBM [IBM Rational Developer for i - Prezzo - Italia](#) non parla degli anni successivi.
 - 6.3.4. L'uso di RDI diventerà in futuro inevitabile per restare al passo con le novità nei linguaggi di programmazione e per implementare la qualità degli strumenti di sviluppo sui sorgenti.
- 6.4. **PDM** (Programming Development Manager) per le operazioni di programmazione.
 - 6.4.1. Fornito gratuitamente a corredo del software di base ma non più implementato dopo, mi pare, il rilascio V6R1.
 - 6.4.2. Funziona senza problemi e resta il principale strumento di gestione degli oggetti su IBMi.
 - 6.4.3. Non ha sostituzioni valide dentro RDI che è intollerabilmente macchinoso e prolisso negli elenchi di oggetti.
 - 6.4.4. Presenta elenchi di librerie, oggetti o membri e permette opzioni di riga anche definite dall'utente.
- 6.5. **QSH** (UNIX-like interface).
 - 6.5.1. Da usare per alcuni servizi disponibili solo in versione UNIX, come SFTP o JAR.
 - 6.5.2. Interfaccia modesta e povera da usare con l'apposito manuale sott'occhio.

7. Ricerca documentazione IBMi

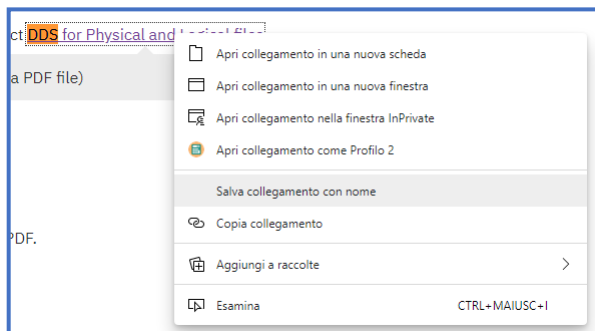
- 7.1. Nelle ricerche seguenti ho usato Windows10 ed EDGE ma non Bing di Microsoft (che Edge suggerisce e spesso forza) perché presenta elenchi diversi da Google, che preferisco perché più prevedibile e collaudato.
- 7.2. Dove e come reperire la manualistica in formato pdf.
 - 7.2.1. La manualistica pdf ha sostituito integralmente da molti anni quella cartacea che per questo sistema operativo non viene più distribuita. L'ultima pila di manuali di un certo release che ho buttato forse nel 1995 era alta almeno due metri.
 - 7.2.2. È opportuno cercare la documentazione dello stesso release dell'IBMi di cui si dispone. Supponiamo che si tratti della versione 7.3.
 - 7.2.3. Googlare "ibm_i pdf manual".
 - 7.2.4. Dal primo link che Google elenca si passa facilmente al seguente <https://www.ibm.com/docs/it/i/7.3?topic=documentation-pdf-files-manuals> scegliendo eventualmente la versione 7.3 su "Cambia versione".



- 7.2.5. Esce una lista di titoli di manuali sulla quale cercare, ad esempio, la parola "DDS" digitando: **Ctrl F, dds, Enter**.



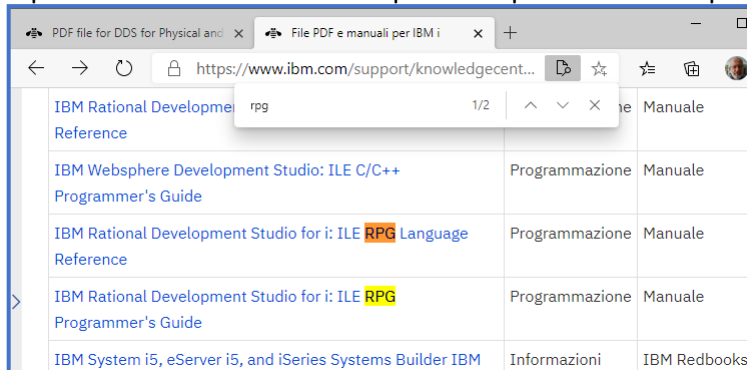
- 7.2.6. Cliccare destro DDS for physical and logical files
selezionando **Salva collegamento con nome...**
- 7.2.7. Cliccare "Salva collegamento con nome"
(non lasciarsi fuorviare dall'improprio termine **collegamento**!)



e continuare lo scarico scegliendo dove.

Conviene cambiare o meglio completare il nome *rzakbpdf.pdf* con il titolo interno in modo che diventi *DDS for Physical and Logical files rzakbpdf.pdf*.

7.2.8. Ripetere lo stesso meccanismo per altre parole. Ad esempio "RPG".



7.2.9. I firewall presenti sulle reti aziendali possono interferire con lo scarico dei pdf.

7.3. Dove e come reperire la documentazione in linea sui comandi del linguaggio di controllo.

7.3.1. Si tenga conto che da qualche anno IBM non produce più i numerosi manuali pdf contenenti tutti i circa 2000 comandi Control Language che fanno parte del sistema operativo e che su questo argomento esiste solo questo tipo di documentazione web.

7.3.2. Si cerchi ad esempio **WRKACTJOB** (Work with Active Jobs).

7.3.3. Googlare "ibm_i wrkactjob".

7.3.4. Come già visto sopra, dal primo link che Google elenca si passa facilmente al seguente <https://www.ibm.com/docs/it/i/7.3?topic=jobs-display-using-wrkactjob> scegliendo eventualmente la versione 7.3 su "Cambia versione o prodotto".

7.4. Dove e come reperire la documentazione in linea sulle istruzioni RPG.

7.4.1. Per argomenti diversi dal CL esistono sia i manuali pdf che la documentazione web sui singoli contenuti.

7.4.2. Si cerchi ad esempio il codice operativo CHAIN (Random Retrieval from a File).

7.4.3. Googlare "ibm_i chain".

7.4.4. Come già visto sopra, dal uno dei primi link che Google elenca si passa facilmente al seguente <https://www.ibm.com/docs/en/i/7.3?topic=codes-chain-random-retrieval-from-file> scegliendo eventualmente la versione 7.3 su "Cambia versione o prodotto".

7.5. Ancora documentazione in linea sulle istruzioni RPG.

7.5.1. Si cerchi ad esempio la BIF (Built-in Functions) %TRIM (Trim Characters at Edges).

7.5.2. Googlare "ibm_i %trim".

7.5.3. Come già visto sopra, dal uno dei primi link che Google elenca si passa facilmente al seguente <https://www.ibm.com/docs/en/i/7.3?topic=functions-trim-trim-characters-edges> n scegliendo eventualmente la versione 7.3 su "Cambia versione o prodotto".

8. Collegamento ad IBMi

8.1. Via Terminale 5250.

8.1.1. I vecchi computer, dotati di versioni precedenti del software di base IBMi, collegavano terminali nativi via cavo twinax tramite una apposita unità di controllo. Tali terminali non sono più disponibili da molti anni, né le macchine fisiche recenti dispongono oggi di tale unità di controllo.

8.2. Via PC e Telnet.

8.2.1. Se l'IBMi è in rete con un PC, è possibile accedere all'IBMi dal PC chiamando il comando DOS TELNET, presente di default sulle versioni vecchie di Windows prima della 7 e comunque installabile a richiesta sulle versioni più recenti.

8.2.2. Digitando sul video comandi DOS il seguente comando, ci si può collegare, ad esempio, alla macchina che ho noleggiato presso un imprenditore informatico tedesco, Holger Scherer:

```
telnet DEV720.RZKH.DE
```

Si presenta un video di signon che permette il collegamento se si dispone di utente e password.



8.2.3. L'interfaccia ottenuta simula il vecchio terminale ma inciampa in numerose occasioni impedendo di fatto l'uso completo delle funzionalità dell'IBMi.

8.2.4. È comunque possibile che il collegamento TELNET ad una macchina sia esplicitamente impedito a livello sistemistico.

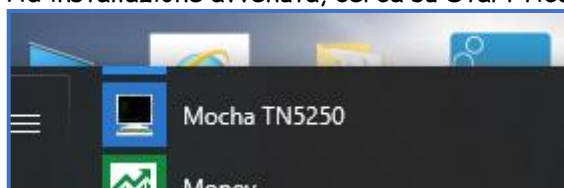
8.3. Via PC ed Emulazione.

8.3.1. Per accedere all'interfaccia carattere di IBMi usando un PC ed emulando i terminali nativi, la soluzione migliore è usare un programma di emulazione 5250 che usa il protocollo TELNET ma risolve i problemi dell'interfaccia TELNET elementare vista sopra.

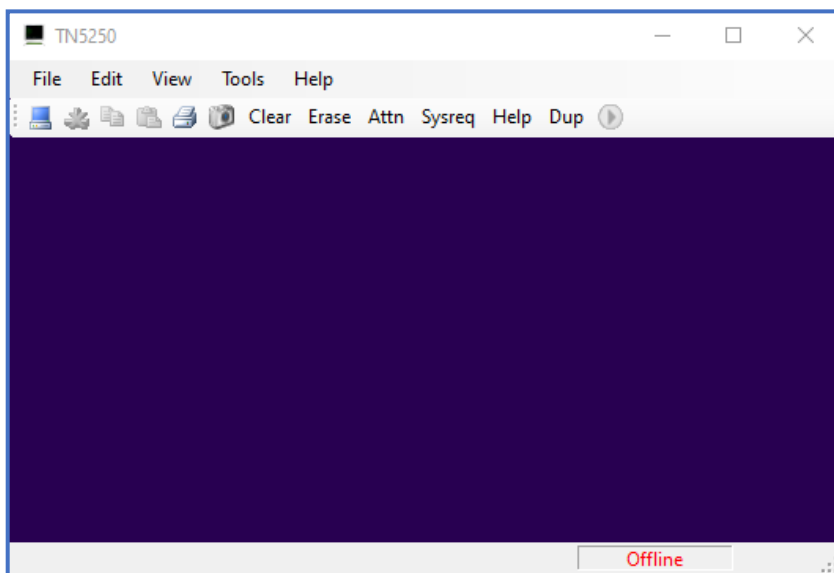
Tra i vari reperibili sul web, ho scelto da molti anni quello che oggi si può scaricare al link <https://mochasoft.dk/tn5250.htm>

La pagina contiene il link per il download e l'installazione del pacchetto.

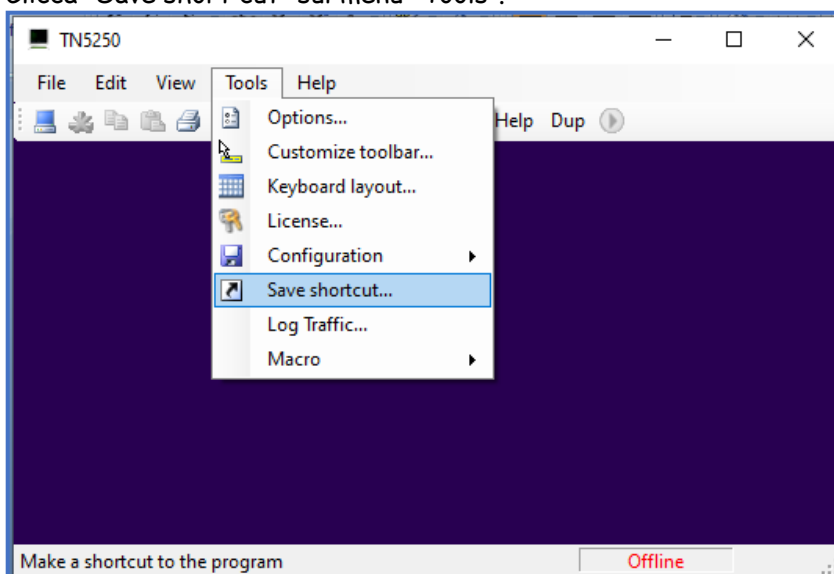
Ad installazione avvenuta, cerca su Start l'icona "Mocha TN5250" e cliccala.



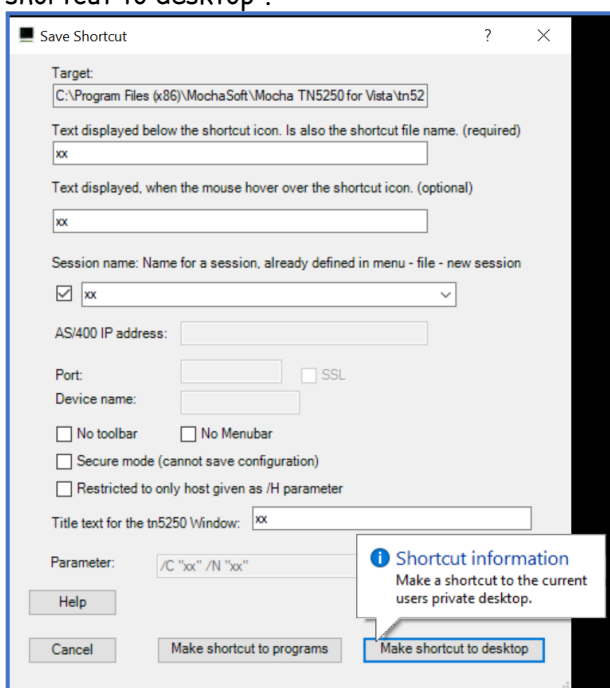
Compare l'interfaccia operativa dell'emulatore.



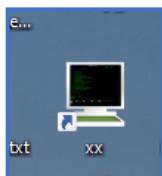
Clicca "Save short cut" sul menù "Tools".



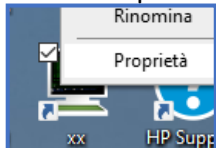
Compila la finestra "Save short cut". Scrivi "xx" (oppure un nome device che lo renda riconoscibile come tuo, ad esempio: N1NERO) nei campi di immissione come qui di seguito e clicca "Make shortcut to desktop".



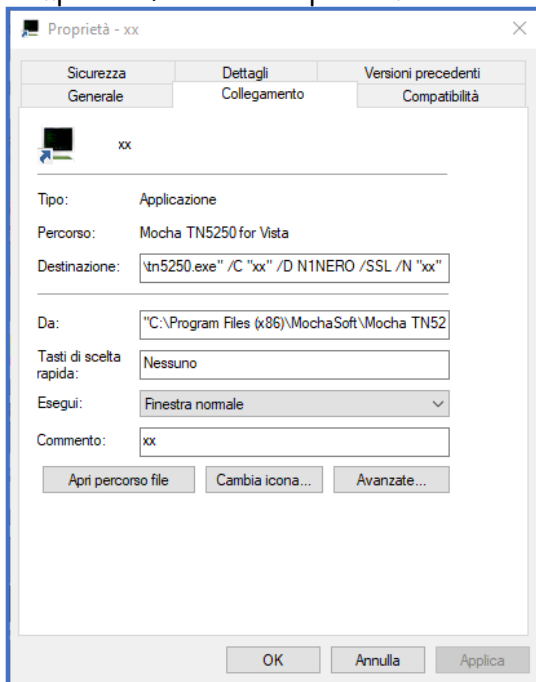
Vai sul desktop e clicca di destro sull'icona appena creata.



Clicca "Proprietà".



Compare la finestra "Proprietà".



Nel campo "Destinazione" sostituisci la stringa:

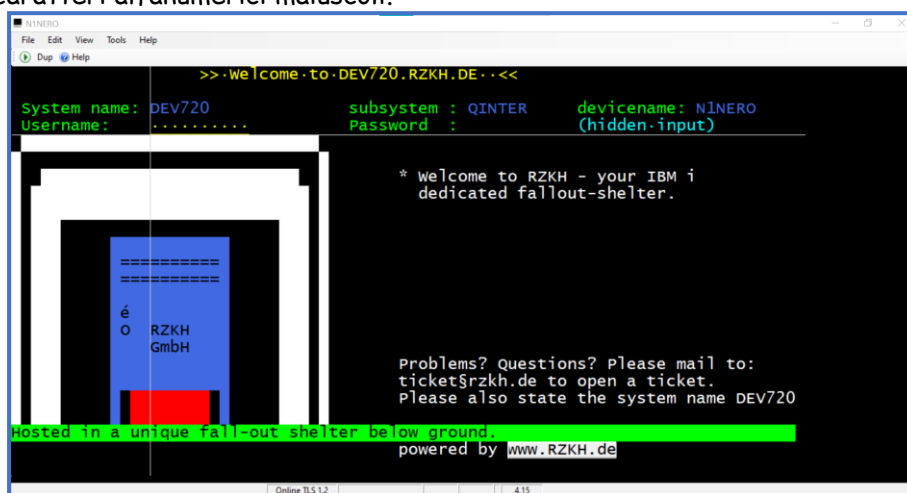
"C:\Program Files (x86)\MochaSoft\Mocha TN5250 for Vista\tn5250.exe" /C "xx" /D N1NERO /SSL /N "xx"

qualunque valore abbia, con la seguente:

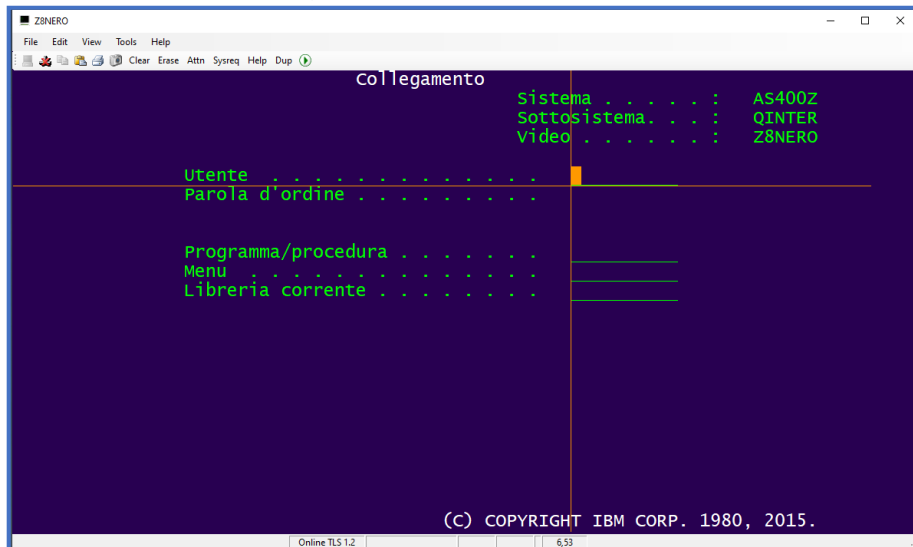
"C:\Program Files (x86)\MochaSoft\Mocha TN5250 for Vista\tn5250.exe" /H DEV720.RZKH.DE /D N1NERO /N N1NERO /SSL

E clicca "OK".

- 8.3.2. Vai di nuovo sul desktop e clicca sull'icona della quale hai appena modificato le proprietà. Compare il video di collegamento al sistema DEV720 sul quale digitare utente e password. Sul sistema vige un livello della parola d'ordine (DPSYSVAL SYSVAL (QPWDLVL)) uguale a "0" (zero) che usa solo 10 caratteri alfanumerici maiuscoli.

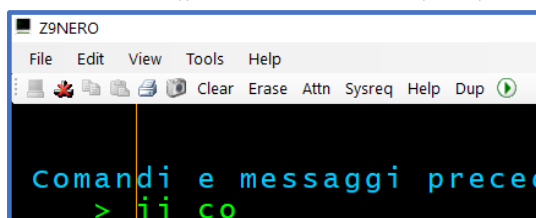


Visto che tale sistema ha un video di SIGNON piuttosto manipolato, riporto anche quello di un altro sistema che usa la forma standard.

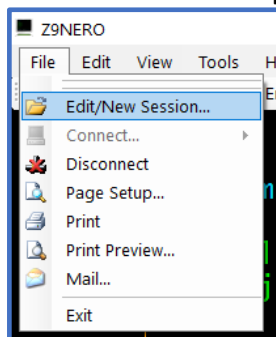


- 8.3.3. Da quando ho installato la fibra e, quindi, ho cambiato modem, per non far cadere le sessioni sull'emulazione senza VPN, ho dovuto eseguire un settaggio aggiuntivo che espongo per completezza anche se mi è necessario solo per raggiungere via web la partizione che ho noleggiato sull'IBMi di Holger Scherer.

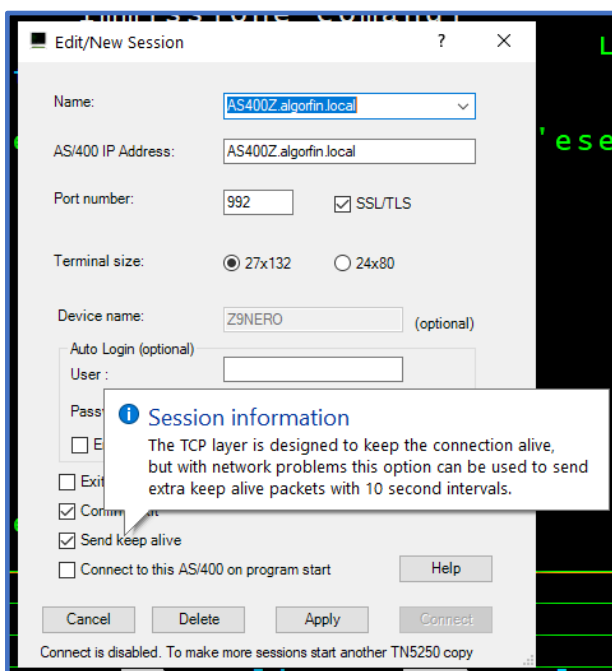
Sul video di emulazione clicco su "File".



Sulla tendina che compare clicco su "Edit/New Session".

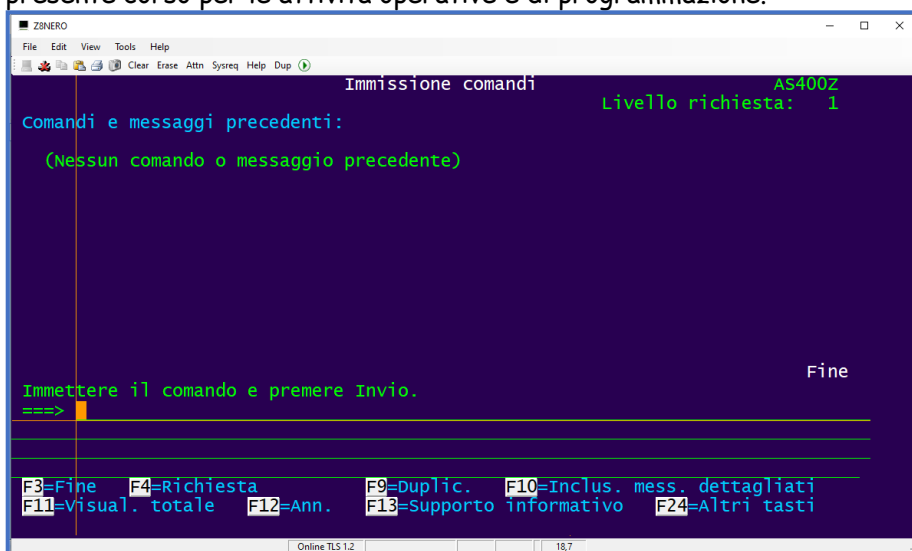


Compare la finestra di settaggio, per la maggior parte già riempito grazie ai parametri inseriti nella stringa di personalizzazione della chiamata. Spunto il flag "Send keep alive", clicco "Apply" e chiudo la finestra.



Purtroppo, il settaggio non viene salvato quando si chiude l'emulazione e quindi non si ripresenta alla prossima chiamata e va rieseguito ad ogni sessione che si apre.

- 8.3.4. Scopo del collegamento tramite emulazione è ottenere il video comandi, interfaccia preferita nel presente corso per le attività operative e di programmazione.



Ove non comparisse subito il video comandi, digita CALL QCMD su una riga comandi.

- 8.3.5. Il font dei caratteri della videata di emulazione migliore dal punto di vista della visibilità, in particolare quando lo schermo viene condiviso con spettatori remoti attraverso un software tipo TeamViewer, è il font Lucida Console, meglio se grassetto (bold). Infatti, la riproduzione remota di una videata di emulazione tende ad assottigliare i caratteri tanto da renderne difficoltosa la visione. Per impostare il font, la sequenza di click dal video di emulazione è: Tools, Options..., Font. Il font Lucida Console viene fornito dall'emulazione stessa e quindi è sempre disponibile. Clicca Lucida Console, Bold, Apply.

8.4. Via PC e RDI (ed Emulazione interna a RDI).

- 8.4.1. Infine, se si comincerà ad usare RDI, sarà naturalmente opportuno valutare l'emulatore 5250 presente come componente integrato dentro quel pacchetto.

9. Creazione ambiente di sviluppo del corso

- 9.1. Organizzazione degli oggetti su IBMi in librerie e similitudine con indirizzari PC.
 - 9.1.1. Gli oggetti su IBMi sono organizzati in librerie.
 - 9.1.2. Le librerie coincidono concettualmente con gli indirizzari sul PC ma una libreria non può contenere un'altra libreria. Solo la libreria di sistema QSYS contiene tutte le altre librerie.
 - 9.1.3. In realtà sul sistema esistono solo gli oggetti che sono posizionati in memoria dove il sistema operativo ritiene opportuno.
 - 9.1.4. Nella testata dell'oggetto esiste un campo "libreria" che permette di raggruppare gli oggetti in base al contenuto di tale campo.
 - 9.1.5. Non esiste quindi una zona di memoria che contenga fisicamente tutti gli oggetti targati con un certo nome di libreria. Tuttavia, pur essendo la libreria un luogo virtuale, ci si può comportare come se fosse un luogo reale.
- 9.2. Creazione libreria.
 - 9.2.1. Creo la libreria CORSO di tipo *TEST per contenere gli oggetti del corso.
> CRTLIB LIB(CORSO) TYPE(*TEST) TEXT('Corso 2023')
È stata creata la libreria CORSO.
- 9.3. Lista librerie di un lavoro.
 - 9.3.1. Ad ogni lavoro sul sistema è associata una lista librerie che permette di trovare più rapidamente gli oggetti che stanno in una libreria della lista.
 - 9.3.2. Il funzionamento della lista librerie corrisponde al PATH del DOS.
 - 9.3.3. Quando un oggetto viene cercato in lista librerie (indicata come *LIBL), la ricerca avviene innanzitutto nella prima libreria della lista, poi nella seconda e così via fino ad arrestare l'azione alla prima ricorrenza dell'oggetto in cui ci si imbatte. L'oggetto trovato viene usato ed ogni altro nelle librerie successive trascurato.
- 9.4. Modifica lista librerie del lavoro.
 - 9.4.1. Nel lavoro corrente impongo la lista librerie
"QTEMP CORSO QGPL"
tramite il comando
CHGLIBL LIBL(QTEMP CORSO QGPL)
 - 9.4.2. Da qui in poi, dentro il lavoro corrente, le ulteriori ricerche di oggetti (ma non le creazioni) potranno facoltativamente essere fatte e soddisfatte senza menzionare nei comandi la libreria dell'oggetto cercato ma lasciando nel nome libreria il default *LIBL (Library List).
- 9.5. File sorgenti e membri di file sorgenti.
 - 9.5.1. I sorgenti dei programmi possono risiedere o nei file sorgenti o in file di testo negli indirizzari IFS.
 - 9.5.2. La seconda possibilità è nata in tempi più recenti e non è frequentata come la prima. Nel corso useremo la prima.
 - 9.5.3. La prima presenta anche una maggiore trattabilità dei sorgenti come dati leggibili e manipolabili a programma.
 - 9.5.4. Il singolo membro sorgente viene trasformato in oggetto eseguibile o in modulo tramite uno dei comandi di compilazione.
- 9.6. Creazione di un file sorgente.
 - 9.6.1. Un file sorgente ha un tracciato standard e automatico:
Sequenza
DataAggiornamento
DatiSorgente
 - 9.6.2. Il campo dati di un file sorgente era abitualmente di 80 caratteri fino all'avvento della versione "ILE" dei compilatori che preferiscono invece un campo dati di 100 caratteri.
 - 9.6.3. Le lunghezze record corrispondenti vanno aumentate di 12 byte per contenere Sequenza e Data.
 - 9.6.4. Ogni record contiene nel campo dati una singola istruzione di un programma sorgente.
 - 9.6.5. Tutte le istruzioni di un singolo programma sorgente stanno in un membro del file sorgente.
 - 9.6.6. Un file sorgente contiene normalmente molti membri, cioè molti programmi sorgenti.
 - 9.6.7. Il nome di un programma oggetto o di un modulo è normalmente uguale al nome del membro sorgente dal quale viene generato.
 - 9.6.8. Il programma oggetto o il modulo incorpora il nome qualificato e il tempo di ultima modifica del sorgente dal quale è stato compilato.

- 9.6.9. Se richiesto in compilazione, l'oggetto generato può contenere anche il sorgente stesso, accessibile per il debug o anche per il recupero tramite apposite API.
- 9.6.10. Nei pacchetti applicativi più complessi sono presenti di solito più file sorgenti con nomi file tradizionali basati sul tipo membro (ovvero tipo sorgente).

Rpg file	RPGLE	QRPGLESRC
Cobol file	CBLLE	QCBLLSRC
Control Language file	CLLE	QCLLESRC
Command	CMD	QCMDSRC
Physical File	PF	QDDSSRC
Logical file	LF	QDDSSRC
Display File	DSPF	QDDSSRC
Panel Group	DSPF	QPNLSRC

- 9.6.11. Nei pacchetti con un numero ridotto di sorgenti, tipo i tool, si usa di solito un solo file sorgente, omonimo del tool, che contiene tutti i tipi di membri sorgenti necessari al tool stesso.

- 9.6.12. Nella libreria CORSO creo un solo file di tipo *SRC (sorgente).

```
> CRTSRCPF FILE(CORSO/CORSO) RCDLEN(112) TEXT('Corso 2023. Sorgenti')
File CORSO creato nella libreria CORSO.
```

- 9.7. Aggiunta e riempimento di un membro del file sorgente con un semplice programma sorgente in Control Language.

- 9.7.1. Per creare un membro nel file sorgente, uso il comando apposito che aggiunge un membro vuoto e poi chiamo il SEU (Source Entry Utility) per scriverci il contenuto sottostante.

```
> ADDPFM FILE(CORSO/NERO) MBR(CNINI) TEXT('Programma iniziale Corso') SRCTYPE(CLLE)
Membro CNINI aggiunto al file CORSO in CORSO.
> STRSEU SRCFILE(CORSO/NERO) SRCMBR(CNINI) OPTION(2)
Modificati 10 record del membro CNINI nel file CORSO/CORSO.
```

- 9.7.2. Oppure, chiamo solo il SEU (Source Entry Utility) con più parametri per fare entrambe le cose.

```
> STRSEU SRCFILE(CORSO/NERO) SRCMBR(CNINI) TYPE(CLLE) OPTION(2) TEXT('Programma
iniziale Neroni')
Membro CNINI aggiunto al file CORSO/CORSO.
Modificati 10 record del membro CNINI nel file CORSO/CORSO.
```

- 9.7.3. Comunque, usando il SEU (Source Entry Utility), nel sorgente digito il seguente contenuto.

Da qui, invece di digitare, puoi copiare e incollare con **alt + trascinamento tasto sinistro mouse** escludendo i numeri a sinistra.

```
0001.00 /* Claudio Neroni 02-11-2020 Creato per Corso */
0002.00 /* CNINI */
0003.00 /* Programma iniziale Neroni */
0004.00
0005.00 /* Inizio programma */
0006.00         PGM
0007.00 /* Imposta la lista librerie del corso */
0008.00         CHGLIBL      LIBL(QTEMP CORSO QGPL)
0009.00 /* Predispone il debug a 132 colonne invece che 80 */
0010.00         ADDENVVAR   ENVVAR(ILE_DEBUGGER_1) +
0011.00                     VALUE(ALLOW_WIDE_SCREEN) LEVEL(*JOB) +
0012.00                     REPLACE(*YES)
0013.00 /* Manda un messaggio al programma chiamante */
0014.00         SNDPGMMSG    MSG('Eseguito programma iniziale Corso')
0015.00 /* Fine programma */
0016.00         ENDPGM
```

- 9.8. Creazione del programma oggetto dal sorgente digitato.

- 9.8.1. Dal sorgente creo un oggetto eseguibile nella libreria QGPL (General Purpose Library, che si mette sempre in lista librerie) chiedendo inoltre l'incorporazione della lista di compilazione nell'oggetto eseguibile per poterlo debuggare.

```
> CRTBNDCLE PGM(QGPL/CNINI) SRCFILE(CORSO/CORSO) DBGVIEW(*LIST)
Programma CNINI creato nella libreria QGPL.
```

- 9.9. Chiamata in esecuzione del programma creato.

- 9.9.1. All'inizio delle prossime sessioni eseguirò il programma per facilitare le attività successive nella sessione stessa.

```
> CALL PGM(CNINI)
Eseguito programma iniziale Corso
```

- 9.10. Per eseguire alcune delle azioni sopra elencate posso valermi dell'interfaccia fornita dal tool di sistema **PDM** (Programming Development Manager) facendo uso dei menù con opzioni native o create dall'utente.

- 9.10.1. In particolare, posso fare uso dell'interfaccia chiamata dal comando

```
WRKLIBPDM (Work Library PDM) alias JJJJ
```


o di quella chiamata dal comando
WRKOBJPDM (Work Object PDM) alias JJJ

10. Utilizzo del PDM (Programming Development Manager)

10.1. Il PDM è il principale tool a bordo di IBMi per la gestione di sorgenti, oggetti e compilazioni. Tramite PDM il programmatore e il sistemista accedono al sistema per svolgere gran parte del loro lavoro.

10.2. Comandi di ingresso al tool PDM.

10.2.1. STRPDM (Start PDM)

Digitando il comando più invio si ottiene il menù preferito da molti, che io non uso mai.

```

Programming Development Manager (PDM)

select one of the following:

1. work with libraries
2. work with objects
3. work with members

9. work with user-defined options

Selection or command
===>

F3=Exit      F4=Prompt      F9=Retrieve      F10=Command entry
F12=Cancel   F18=Change defaults

(C) COPYRIGHT IBM CORP. 1981, 2007.

```

Le opzioni utili del menù sono la 1 e la 2 a cui però preferisco accedere tramite comando diretto.

10.2.2. WRKLIBPDM (Work with Libraries Using PDM)

Digitando il seguente comando si ottiene l'elenco delle librerie che iniziano con "CORSO".

```
> wrklibpdm corso*
```

```

Gestione delle librerie con il PDM                                DEV720
Tipo di elenco . . . . . *ALL                Inizio elenco da . . .
Immettere le opzioni e premere Invio.
 2=Modifica          3=Copia  4=Cancellazione  5=Visualizzazione
 7=Ridenominazione  8=Visual. descrizione  9=Salvataggio  10=Ripristino...
                                ASP
Opz  Libreria      Tipo      Unità      Testo
    CORSO         *TEST
    CORSO2018     *TEST
                                Corso Creditis 2021
                                Corso Almaviva 2018

Parametri o comando
==>
F3=Fine              F4=Richiesta          F5=Rivisualizzaz.      F6=Creazione
F9=Duplicazione      F10=Immissione comandi  F23=Altre opzioni     F24=Altri tasti
Questo è il sottogruppo di un elenco.

```

10.2.3. WRKOBJPDM (Work with Objects Using PDM)

Digitando il seguente comando si ottiene l'elenco degli oggetti contenuti nella libreria "CORSO".

```
> wrkobjpdm corso
```

```

32
Gestione degli oggetti con il PDM
DEV720

Libreria. . . . . CORSO                Inizio elenco da. . . . .
                                      Inizio elenco da tipo . . .

Immettere le opzioni e premere Invio.
  2=Modifica      3=Copia      4=Cancel laz.   5=visualiz.      7=Ridenominaz.
  8=Visual. descrizione  9=Salvatag. 10=ripristino 11=Trasferimento...

Opz  Oggetto      Tipo      Attributo      Testo
----
CNI1      *PGM        CLLE          Hello boy
VUOTO2    *PGM        CLP           Solo PGM - ENDPGM : compilabile debu
CNPFI1    *FILE       PF-DTA        Dati uno - fisico
CNPFI10   *FILE       LF            Dati uno - logico con chiave unica
CORSO     *FILE       PF-SRC        Corso Creditis 2021. Sorgenti

Parametri o comando
====>
F3=Fine          F4=Richiesta      F5=Rivisualizzaz.  F6=Creazione
F9=Duplicazione  F10=Immissione comandi F23=Altre opzioni  F24=Altri tasti
(C) COPYRIGHT IBM CORP. 1981. 2007.

```

10.2.4. Vista la frequenza con cui li chiamo, ne ho fatto degli alias di nome `jjjj` e `jjj` nella libreria `QGPL` mediante i seguenti comandi.

```
> CRTRPRXCMD CMD(QGPL/JJJJ) TGTCMD(WRKLIBPDM) TEXT('WRKLIBPDM alias')
Comando JJJJ creato nella libreria QGPL.
```

```
> CRTPRXCMD CMD(QGPL/JJJ) TGTCMD(WRKOBJPDM) TEXT('WRKOBJPDM alias')
Comando JJJ creato nella libreria QGPL.
```

Quindi, ad esempio, i due comandi sopra visti diventano rispettivamente:

```
> wrklibpdm corso*jjjj corso*
> wrkobjpdm corso jjj corso
```

10.3. Esempio dettagliato d'uso del comando WRKOBJPDM (Work with Objects Using PDM).

10.3.1. Per semplificare, do per caricato il contenuto dei membri sorgente CNPF1, utilizzato qui di seguito.

10.3.2. Sull'elenco di oggetti nella libreria già visto, paginando eventualmente in avanti, raggiungo la riga del mio file sorgente e vi esercito l'opzione 12 + Invio.

```

Gestione degli oggetti con il PDM                                DEV720

Libreria. . . . . CORSO                                     Inizio elenco da. . . . .
                                                                Inizio elenco da tipo . . .

Immettere le opzioni e premere Invio.
 2=Modifica   3=Copia   4=Cancellaz.  5=Visualiz.   7=Ridenominaz.
 8=Visual. descriz.  9=Salvatag. 10=Ripristino 11=Trasferimento...

Opz Oggetto      Tipo      Attributo  Testo
---
CNINI          *PGM      CLLE      Programma iniziale Allievo
CN1            *PGM      CLLE      Hello boy
VUOTO2         *PGM      CLP       Solo PGM - ENDPGM : compilabile debu
CNPF1          *FILE     PF-DTA    Dati uno - fisico
12 CORSO       *FILE     PF-SRC    Corso Creditis 2021. sorgenti

Parametri o comando
==>

F3=Fine      F4=Richiesta  F5=Rivisualizzaz.  F6=Creazione
F9=Duplicazione F10=Immissione comandi F23=Altre opzioni  F24=Altri tasti

```

10.3.3. Si apre l'elenco dei membri sorgenti sul quale scelgo di vedere il contenuto del file fisico **CNPF1** (Dati uno) tramite l'opzione 5 + Invio.

```

Gestione dei membri con il PDM                                DEV720

File . . . . . CORSO                                     Inizio elenco da . . .
Libreria. . . . . CORSO

Immettere le opzioni e premere Invio.
 2=Modifica  3=Copia  4=Cancel.  5=Visualiz.  6=Stampa  7=Ridenom.
 8=Visual. descriz.  9=Salvat. 13=Modif. testo 14=Costruz. 15=Creaz. modulo...

Opz Membro      Tipo      Testo
---
CNINI          CLLE      Programma iniziale Allievo
CNLL           CLLE      Lista librerie Allievo
5 CNPF1        PF        Dati uno - fisico
CNPF10         LF        Dati uno - logico con chiave unica
CN1            CLLE      Hello boy
NUDSP          RPGLE     Visualizza documento
NUDSPW         DSPF      Visualizza documento
NUDSPW_0       DSPF      Visualizza documento

Parametri o comando
==>

F3=Fine      F4=Richiesta  F5=Rivisualizzaz.  F6=Creazione
F9=Duplicazione F10=Immiss. comandi F23=Altre opzioni  F24=Altri tasti

```

10.3.4. Il SEU visualizza il membro sorgente come qui riportato.

```

0000.10      * Claudio Neroni 02-11-2020 Creato per Corso
0000.20      *      CNPF1
0000.30      *      Dati uno - fisico
0001.40      *
0001.00      A          R PF1
0002.00      A          TEXT('Dati uno')
0003.00      A          PF1ALFA1      10A
0004.00      A          COLHDG('Campo' +
0005.00      A          'alfa' +
0006.00      A          'uno' )
0007.00      A          TEXT('Campetto uno alfanumerico')
0008.00      A          ALIAS(CAMPO_ALFA_UNO)
0009.00      A          PF1ALFA2      3A
0010.00      A          COLHDG('Campo' +
0011.00      A          'alfa' +
0012.00      A          'due' )
0013.00      A          DFT('VAI')
0014.00      A          PF1NUME1      7S 0
0015.00      A          COLHDG('Numero' +
0016.00      A          'uno' )
0017.00      A          DFT(3221)
0018.00      A          PF1NUME2      7P 0
0019.00      A          COLHDG('Numero' +
0020.00      A          'due' )
0021.00      A          PF1DATA1      L
0022.00      A          COLHDG('Data' +
0023.00      A          'uno' )
0024.00      A          DFT('2001-01-02')

```

0025.00	A	PF1ALFA3	2		
0026.00	A			COLHDG ('Campo'	+
0027.00	A			'alfa'	+
0028.00	A			'tre'	)

- 10.3.5. Uscito dal SEU mediante F3, sulla stessa riga esercito l'opzione "14 + Invio" che compila il sorgente creando l'oggetto corrispondente e omonimo del membro sorgente. Si noti che l'opzione 14 è sensibile al tipo SEU e che questo deve essere congruente con il contenuto del sorgente.

Opz	Membro	Tipo	Testo
14	CNPF1	PF	Dati uno
	CNPF10	LF	Dati uno
	CN1	CLLE	Hello boy

- 10.3.6. Se l'oggetto esiste già, il PDM chiede il consenso a ricoprirlo. **ATTENZIONE:** la ricompilazione perde i record eventualmente già scritti nel file.

```

Conferma compilazione membro

Per l'operazione di compilazione già esiste il seguente oggetto:

Oggetto esistente . . . . . : CNPF1
Libreria . . . . . : CORSO
Tipo oggetto . . . . . : *FILE

Membro da compilare. . . . . : CNPF1
File . . . . . : CORSO
Libreria . . . . . : CORSO

Immettere la scelta e premere Invio.
Premere F12=Annull. per terminare e non eseguire l'operazione di compilazione.

Cancellazione oggetto esistente. . . . N Y=Si, N=No

```

- 10.3.7. Scelta Y=Ricopertura + Invio, la creazione inciampa perché non può cancellare il file fisico puntato da un logico e si presenta il seguente errore.

```

Opz  Membro  Tipo  Testo
14  CNPF1    PF    Dati uno
    CNPF10   LF    Dati uno
    CN1      CLLE   Hello boy
    CN2      CLLE   bis
    JLIBLC   CLLE   Library list from jobd. Cpp

Parametri o comando
====>
F3=Fine      F4=Richiesta  F5=Rivisualizzaz.  F6=Crea
F9=Duplicazione F10=Immiss. comandi F23=Altre opzioni  F24=Al
Non è possibile cancellare il file CNPF1 in CORSO o un suo membro.

```

Supponendo invece che il logico non esista o che venga preliminarmente cancellato, l'esito sarebbe favorevole.

```

Opz  Membro  Tipo  Testo
    CNINIT1  CLLE   Programma iniziale
    CNLL      CLLE   Lib1 Neroni
    CNPF1     PF    Dati uno
    CNPF10    LF    Dati uno
    CN1       CLLE   Hello boy
    CN2       CLLE   bis
    JLIBLC    CLLE   Library list from jobd. cpp

Parametri o comando
====>
F3=Fine      F4=Richiesta  F5=Rivisualizzaz.  F6=Creazione
F9=Duplicazione F10=Immiss. comandi F23=Altre opzioni  F24=Altri tasti
E' stato cancellato l'oggetto CNPF1 in CORSO di tipo *FILE.

```

Si osservi che al piede di pagina il PDM presenta un messaggio con un "+" all'estrema destra: se si porta il cursore sulla riga e si pagina in avanti, si presenta un altro messaggio e poi un altro ancora.

```

F3=Duplicazione F10=Immiss. comandi F23=Altre opzioni  F24=Altri tasti
File CNPF1 creato nella libreria CORSO.

```

10.4. Parametri di personalizzazione del PDM

- 10.4.1. Da tutte le videate funzionali PDM si può raggiungere la videata di personalizzazione del PDM premendo F18. La seconda videata contiene un altro paio di scelte.

Modifica valori predefiniti			
Immettere le scelte e premere Invio.			
Libreria oggetti	*SRCLIB	Nome, *CURLIB, *SRCLIB	
Sostituzione oggetto	N	Y=Si, N=No	
Compilazione in batch	N	Y=Si, N=No	
Esecuzione in batch	N	Y=Si, N=No	
Salv. val. predef. sessione.	Y	Y=Si, N=No	
Opzioni Salv./Ripristino . .	1	1=Singolo, 2=Tutto	
Descrizione lavoro	CORSO	Nome, *USRPRF, F4 per elenco	
Libreria	CORSO	Nome, *CURLIB, *LIBL	
Modifica tipo e testo	Y	Y=Si, N=No	
File opzioni	DOPT	Nome	
Libreria	CORSO	Nome, *CURLIB, *LIBL	
Membro	CORSO	Nome	
Modalità schermo intero . . .	N	Y=Si, N=No	
Segue...			
F3=Fine F4=Richiesta F5=Rivisualizzazione F12=Annullamento			
Registraz. comandi opzioni . .	Y	Y=Si, N=No	
Uscita da elenchi con INVIO . .	.	Y=Si, N=No	

10.4.2. Nel nostro caso

- la **Libreria oggetti** corrisponde con la libreria sorgente,
- la **Sostituzione oggetto** avviene solo dopo una richiesta di consenso,
- alla **Compilazione in batch** si preferisce quella interattiva perché sulle macchine moderne le compilazioni sono fulminee,
- la **Descrizione lavoro** è specifica per il programmatore,
- il **File opzioni** è uno già dotato di numerose opzioni personalizzate che possiamo condividere e che possiamo modificare senza interferire con altri,
- la **Modalità schermo intero** (senza descrizione delle opzioni e dei comandi permessi per far posto ad un maggior numero di righe negli elenchi) non viene usata perché in fase di apprendimento conviene vedere sempre opzioni e comandi funzionali,
- la **Registrazione comandi opzioni** nel log del lavoro avviene per documentare, rivedere ed eventualmente duplicare i comandi già eseguiti nella sessione tramite le opzioni PDM.

10.4.3. Le scelte non menzionate sono di default e raramente modificate.

10.4.4. Le personalizzazioni si conservano come impostate fino ad ulteriore modifica.

10.5. Opzioni personalizzate

- 10.5.1. Sulle righe di tutti gli elenchi del PDM è possibile manipolare l'oggetto o il membro indicato sulla riga facendo uso delle opzioni precostituite e indicate sul video stesso nella forma "schermo NON intero" e meglio documentate nell'help raggiungibile con F1 da schermo intero o non intero.
- 10.5.2. La stessa opzione chiama in molti casi esecuzioni sensibili al tipo di oggetto o di membro della riga permettendo, ad esempio, con la stessa opzione 14 la compilazione di un membro con il comando specifico applicabile al tipo seu del membro stesso. Continuando l'esempio: tipo seu RPGLE chiama il comando CRTBNDRPG, tipo seu CLLE chiama il comando CRTBNDCL e così via.
- 10.5.3. Ma, se le opzioni precostituite (sempre numeriche) non bastano, è possibile definire delle opzioni personalizzate con un nome di uno o due caratteri.
- 10.5.4. Da tutte le videate funzionali PDM si può raggiungere la videata di **Gestione opzioni definite dall'utente** del PDM premendo F16.

```

Gestione opzioni definite dall'utente                                DEV720
File . . . . . : JOPT                      Membro . . . . . : CORSO
Libreria . . . . : CORSO                  Inizio elenco da . : 
Immettere le opzioni e premere Invio.
2=Modifica      3=Copia      4=Cancellazione  5=Visualizzazione

Opz  Opzione  Comando
---  -
C    ?CHGPFM FILE(&L/&F) MBR(&N) SRCTYPE(*SAME)
X    JXYZ FUNCTION(X) SRCTYP(&S) SRCLIB(&L) SRCFIL(&F) SRCMBR(&N)
Y    JXYZ FUNCTION(Y) SRCTYP(&S) SRCLIB(&L) SRCFIL(&F) SRCMBR(&N)
Z    JXYZ FUNCTION(Z)
$F    ? CLRPFM FILE(&L/&N) MBR(*FIRST)
$M    ? CLRPFM FILE(&L/&F) MBR(&N)
AK    CHGLIB LIB(&N) CRTAUT(NERONI)
AL    GRTOBJAUT OBJ(&L/&N) OBJTYPE(&T) AUTL(NERONI)
AM    ? ALCOBJ OBJ(&L/&F *FILE *EXCL &N)) WAIT(1)

Comando
==>
F3=Fine      F4=Richiesta      F5=Rivisualizzazione      F6=Creazione
F9=Duplicazione      F10=Immissione comandi      F24=Altri tasti

```

10.5.5. Se si vuole creare una opzione nuova, si procede con F6=Creazione.

```

creazione di una opzione definita dall'utente
Immettere l'opzione e il comando e premere Invio.

Opzione . . . . . $1  Opzione da creare
Comando . . . . . runqry

F3=Fine      F4=Richiesta      F12=Annullamento

```

10.5.6. Si osserva che su questo video, mettendo il cursore 5250 sul campo "Comando" e premendo F1 si può ottenere il testo d'aiuto sulle variabili che il PDM prevede.

10.5.7. Sulla videata si digita un nome opzione non ancora esistente e un comando da eseguire. In questo caso si ipotizza di voler creare l'opzione "S1" che chiamerà un query di default parametrizzato sulla base della riga su cui, a tempo di esecuzione, si scriverà l'opzione "S1" + Invio.

10.5.8. Comunque, digitati opzione e comando, si preme F4 per ottenere il prompter del comando digitato.

```

Esecuzione del query (RUNQRY)
Immettere le scelte e premere Invio.

Query . . . . . QRY
Libreria . . . . . QRYFILE
File Query:
File . . . . .
Libreria . . . . .
Membro . . . . . + per altri valori
Tipo emissione prospetto . . . . . OUTTYPE
Formato dell'emissione . . . . . OUTFORM
Selezione record . . . . . RCDSLT

*LIBL
*LIBL
*FIRST
*RUNOPT
*RUNOPT
*NO

F3=Fine      F4=Richiesta      F5=Rivisualizzazione      F12=Annullamento
F13=Come usare lo schermo      F24=Altri tasti

```

10.5.9. Sul prompter si digitano le variabili PDM e si completano i valori costanti.

```

Esecuzione del Query (RUNQRY)

Immettere le scelte e premere Invio.

Query . . . . . QRY
Libreria . . . . . *LIBL
File Query: . . . . . QRYFILE
File . . . . . &N
Libreria . . . . . &L
Membro . . . . . *FIRST
+ per altri valori
Tipo emissione prospetto . . . . . OUTTYPE
Formato dell'emissione . . . . . OUTFORM
Selezione record . . . . . RCDSLT
*RUNOPT
*RUNOPT
*YES

Fine

F3=Fine F4=Richiesta F5=Rivisualizzazione F12=Annullamento
F13=Come usare lo schermo F24=Altri tasti

```

10.5.10. Premendo Invio si riempie la stringa per l'opzione "S1".

```

Creazione di una opzione definita dall'utente

Immettere l'opzione e il comando e premere Invio.

Opzione . . . . . $1 Opzione da creare
Comando . . . . . RUNQRY QRYFILE((&L/&N)) RCDSLT(*YES)

F3=Fine F4=Richiesta F12=Annullamento

```

10.5.11. Premendo Invio si ottiene la creazione dell'opzione "S1".

```

Gestione opzioni definite dall'utente DEV720

File . . . . . JOPT Membro . . . . . CORSO
Libreria . . . . . CORSO Inizio elenco da . . . . .

Immettere le opzioni e premere Invio.
2=Modifica 3=Copia 4=Cancellazione 5=Visualizzazione

Opz Opzione Comando
C ?CHGPFM FILE(&L/&F) MBR(&N) SRCTYPE(*SAME)
X JXYZ FUNCTION(X) SRCTYP(&S) SRCLIB(&L) SRCFIL(&F) SRCMBR(&N)
Y JXYZ FUNCTION(Y) SRCTYP(&S) SRCLIB(&L) SRCFIL(&F) SRCMBR(&N)
Z JXYZ FUNCTION(Z)
$F ? CLRPFM FILE(&L/&N) MBR(*FIRST)
$M ? CLRPFM FILE(&L/&F) MBR(&N)
$1 RUNQRY QRYFILE((&L/&N)) RCDSLT(*YES)
AK CHGLIB LIB(&N) CRTAUT(NERONI)
AL GRTOBJAUT OBJ(&L/&N) OBJTYPE(&T) AUTL(NERONI)

Segue...

Comando
==>
F3=fine F4=Richiesta F5=Rivisualizzazione F6=Creazione
F9=Duplicazione F10=Immissione comandi F24=Altri tasti
E' stata creata l'opzione definita dall'utente $1.

```

10.5.12. Per usare l'opzione qui creata basterà digitare "S1" + Invio sulla riga corrispondente ad un file fisico per chiamare la visualizzazione del contenuto del file tramite il query di default.

```

Gestione degli oggetti con il PDM DEV720

Libreria. . . . . CORSO Inizio elenco da. . . . .
Inizio elenco da tipo . . . . .

Immettere le opzioni e premere Invio.
2=Modifica 3=Copia 4=Cancellaz. 5=Visualiz. 7=Ridenominaz.
8=Visual. descrizione 9=Salvatag. 10=Ripristino 11=Trasferimento...

Opz Oggetto Tipo Attributo Test
CN1 *PGM CLLE Hello boy
VUOTO2 *PGM CLP Solo PGM - ENDPGM : compilabile debu
$1 CNPF1 *FILE PF-DTA Dati uno - fisico
CNPF10 *FILE LF Dati uno - logico con chiave unica
CORSO *FILE PF-SRC Corso Creditis 2021. Sorgenti
JOPT *FILE PF-DTA Corso Creditis 2021. OpzioniPDM (QAUO
CORSO *JOBOD Corso Creditis 2021. DescrizioneLavor

Fine

Parametri o comando
==>
F3=Fine F4=Richiesta F5=Rivisualizzaz. F6=Creazione
F9=Duplicazione F10=Immissione comandi F23=Altre opzioni F24=Altri tasti

```


10.5.13. All'invio, comparirà dapprima il video di scelta del query.

Scelta dei record

Immettere il confronto e premere Invio. Inserire OR per ogni nuovo gruppo.
Test: EQ, NE, LE, GE, LT, GT, RANGE, LIST, LIKE, IS, ISNOT...

AND/OR	Campo	Test	Valore (Campo, Numero, 'Caratteri', o ...)

Fine

Campo	Testo	Lung	Dec
PF1ALFA1	Campetto uno alfanumerico	10	
PF1ALFA2	Campo alfa due	3	
PF1NUME1	Numero uno	7	0
PF1NUME2	Numero due	7	0
PF1DATA1	Data uno	10	L

Segue...

F3=Fine F9=Inserimento F11=Visualizz. solo nomi F12=Annullamento
F18=File F19=Gruppo succ. F20=Riorganizzazione F24=Altri tasti

(C) COPYRIGHT IBM CORP. 1988

10.5.14. E poi, ancora Invio, l'elenco del contenuto.

Visualizzazione prospetto

Larghezza prospetto . : 60
Spostamento a colonna . . .

Riga	1	2	3	4	5	6
Campo	Campo	Numero	Numero	Data	Campo	
alfa	alfa	uno	due	uno	alfa	
uno	due				tre	
000001	aaaaaaaaa	bbb	1,111,111	2,222,222	2021-02-16	cc
000002	ddddddddd	eee	3,333,333	4,444,444	2021-02-17	ff

***** Fine prospetto *****

Fine

F3=Fine F12=Annullamento F19=Sinistra F20=Destra F21=Divisione F22=Larghezza 80

10.5.15. Ravanando nel log interattivo generato dal PDM, si trova il comando generato ed eseguito dall'opzione S1

```
> RUNQRY QRYFILE((CORSO/CNPF1)) RCDSLT(*YES)
```

10.6. Manipolazione di sorgenti su elenco membri PDM tramite opzioni che chiamano SEU.

10.6.1. Opzione **2** = Modifica

Chiama il SEU (Source Entry Utility) in modifica sul sorgente contenuto nel membro.

Il tipo del membro determina il controllo di sintassi esercitato dal SEU.

10.6.2. Opzione **5** = Visualizzazione

Chiama il SEU (Source Entry Utility) in visualizzazione sul sorgente contenuto nel membro.

10.6.3. Opzione **25** = Ricerca stringa + F13 (ripete l'opzione su tutti i membri successivi)

Chiama il SEU (Source Entry Utility) in visualizzazione o modifica con ricerca di una stringa su tutti i sorgenti che la contengono.

La stringa e le modalità di ricerca (2=Modifica o 5=Visualizzazione) si forniscono sull'apposito video preliminare "Ricerca di stringa".

È possibile anche la sostituzione una ad una o di massa della stringa con un'altra da fornire alla prima corrispondenza trovata.

10.6.4. Opzione **54** = Confronto

Confronta il membro scelto con un altro da designare nel prompt successivo all'Invio tramite il comando CMPPFM (Compare Physical File Member) che genera uno spool delle differenze tra i due membri.

10.7. Compilazioni di sorgenti.

10.7.1. Opzione **14** = Compilazione

Chiama il comando di compilazione corrispondente al tipo membro su cui si esercita l'opzione.

La compilazione, su un tipo membro che può generare programma o modulo, genera un programma eseguibile monomodulo.

10.7.2. Opzione **15** = Creazione modulo

Chiama il comando di creazione modulo non eseguibile corrispondente al tipo membro su cui si esercita l'opzione.

Il modulo per diventare eseguibile dovrà essere utilizzato da un **CRTPGM** (Create Program) che crea un programma eseguibile assemblando uno o più moduli che si chiamano tra loro.

11. Impostazione della lista librerie

- 11.1. Verifica della lista librerie nel lavoro corrente.
 - 11.1.1. Il comando `DSPLIBL` (Display Library List) visualizza la lista librerie corrente.
- 11.2. La lista librerie si compone di quattro segmenti che possono essere impostati ciascuno a modo suo.
 - 11.2.1. System library list.
 - 11.2.1.1. Contiene le librerie del software di base ed è gestita una libreria alla volta dal comando `CHGSYSLIBL` (Change System Library List).
 - 11.2.1.2. Assume un valore iniziale uguale in tutti i lavori prelevato dal valore di sistema `QSYSLIBL` (System library list).
 - 11.2.2. Product library.
 - 11.2.2.1. Inserita automaticamente e temporaneamente in lista quando è in corso di esecuzione un comando che lo richiede.
 - 11.2.3. Current library.
 - 11.2.3.1. Gestibile con `CHGCURLIB` (Change Current Library) e `CHGLIBL`.
 - 11.2.3.2. Sostituita automaticamente e temporaneamente quando è in corso di esecuzione un comando che lo richiede.
 - 11.2.4. User library list.
 - 11.2.4.1. Nei lavori interattivi assume il valore iniziale presente sulla descrizione lavoro attribuita al profilo utente che può essere quella prelevato dal valore di sistema `QUSRLIBL` (User library list).
 - 11.2.4.2. Nei lavori batch assume il valore iniziale richiesto dal comando di sottomissione che fa nascere il lavoro batch. Il lanciatore può indicarla esplicitamente, o usare la propria al momento del lancio o quella di una job description scelta durante il lancio.
 - 11.2.4.3. La variazione della lista librerie utente durante il lavoro (o al suo inizio) viene discussa qui di seguito.
- 11.3. Natura e funzione di una lista librerie utente in un lavoro attivo.
 - 11.3.1. Gli oggetti sul sistema sono radunati a gruppi mediante l'uso delle librerie che svolgono la funzione di contenitori.
 - 11.3.2. I programmi possono essere radunati ad esempio per tipologia di elaborazione. Nelle aziende industriali, ad esempio: contabilità, ordini-bolle-fatture, produzione, paghe, cespiti, trasporti. Nelle aziende finanziarie: per tipologie di servizio finanziario. Nelle aziende di logistica: per via mare, via gomma, via aerea. E via cantando.
 - 11.3.3. I file sono radunati invece per azienda o per divisione di azienda.
 - 11.3.4. Un ambiente elaborativo sarà costituito dai programmi di una o più tipologie e normalmente dai file di una singola azienda.
 - 11.3.5. Ne segue che un ambiente elaborativo si definisce elencando le librerie che contengono i file di una azienda e le librerie dei programmi che li devono elaborare.
 - 11.3.6. Se la regola di ricerca dei file da elaborare e dei programmi che li elaborano prevede di non usare i nomi delle singole librerie cablati nei programmi ma solo la lista di librerie usata dal lavoro corrente, si può evitare anche il cablaggio del nome della lista librerie.
 - 11.3.7. Quindi per cambiare azienda da elaborare basta cambiare la lista librerie del lavoro.
 - 11.3.8. In sintesi: le liste librerie sono elenchi di librerie che vengono attribuiti ai lavori attivi sul sistema per reperire gli oggetti necessari alle elaborazioni.
- 11.4. Come impostare una lista librerie a inizio lavoro per eseguire le elaborazioni di un certo ambiente elaborativo.
 - 11.4.1. Tramite la descrizione lavori dell'utente.
 - 11.4.2. Tramite il programma iniziale dell'utente.
 - Cablando la libl (lista librerie) dentro il comando `CHGLIBL` in un programma `CLLE` e chiamandolo quando necessario. Abbiamo visto l'esempio con il programma `CNINI`.
- 11.5. Come variare la lista librerie durante il lavoro corrente per eseguire le elaborazioni di un certo ambiente elaborativo.
 - 11.5.1. Tramite `CHGLIBL` (Change Library List).
 - 11.5.1.1. Il comando permette di sostituire integralmente la lista librerie grazie ad un unico elenco inserito nei parametri.
 - 11.5.1.2. Il comando nel log può essere ripreso e ripetuto.
 - 11.5.2. Tramite `EDTLIBL` (Edit Library List).

- 11.5.2.1. Il comando visualizza l'elenco delle librerie attualmente in lista e ne permette la modifica interattiva.
- 11.5.2.2. Permette la correzione, l'aggiunta o la cancellazione dei nomi delle librerie dalla lista.
- 11.5.2.3. Permette lo spostamento delle librerie in altra posizione modificando il campo numerico di sequenza in lista.
- 11.5.2.4. Non traccia nel log un comando ripetibile.
- 11.5.2.5. Poco raccomandabile se non si vogliono introdurre elementi di incertezza nelle elaborazioni.
- 11.5.3. Tramite **ADDLIBLE** (Add Library List Entry) e **RMVLIBLE** (Remove Library List Entry).
- 11.5.3.1. I comandi aggiustano l'elenco delle librerie una libreria alla volta.
- 11.5.3.2. Utili solo per modifiche parziali o estemporanee.
- 11.5.4. Tramite tool appositi come **JLIBL** (Library list from jobd) che recuperano la lista librerie da una descrizione lavori e poi con quella eseguono il **CHGLIBL**.
- 11.5.5. Tramite programmi di inizializzazione di ambiente che contengono in particolare il settaggio della lista librerie eseguito non importa come. L'importante è solo una corrispondenza mnemonica tra il nome dell'ambiente e quello del programma che esegue il settaggio iniziale per le elaborazioni dell'ambiente.

12. Gestione delle autorità

- 12.1. Le autorizzazioni, pur se gestite nativamente dal sistema, entrano in gioco solo su installazioni con una elevata complessità e in presenza di molte utenze e molti accessi. Sono comunque marginali rispetto al servizio applicativo e toccano soprattutto il servizio sistemistico e la presenza sul web del sistema.
- 12.2. In assenza di forzature che vedremo dopo, quando il sistema crea un oggetto su richiesta di un utente:
- 12.2.1. assegna l'oggetto in proprietà all'utente creatore al quale fornisce anche tutte le autorizzazioni possibili;
- 12.2.2. definisce sempre quali autorizzazioni sull'oggetto fornire al pubblico, cioè a tutti gli utenti non esplicitamente autorizzati all'oggetto stesso.
- 12.3. Ad esempio, l'utente NERONI crei la libreria CORSOAUT e ne visualizzi le autorizzazioni.

```
> CRTLIB LIB(CORSOAUT) TYPE(*TEST) TEXT('Prova autorità')
È stata creata la libreria CORSOAUT.
> EDTOBJAUT OBJ(CORSOAUT) OBJTYPE(*LIB)
```

Modifica autorizzazione oggetto											
Oggetto	:	:	:	:	:	:	:	:	:	:	:
Libreria	:	:	:	:	:	:	:	:	:	:	:
Tipo oggetto	:	:	:	:	:	:	:	:	:	:	:
CORSOAUT			Propriet.			NERONI					
QSYS			Gruppo principale			*NONE					
*LIB			Unità ASP			*SYSBAS					
Immettere le modifiche alle autorizzazioni correnti e premere Invio.											
Oggetto protetto da elenco autoriz.										*NONE	
Utente	Gruppo	Autoriz. oggi.	-----Oggetto-----					-----Dati-----			
*PUBLIC		*EXCLUDE	O	M	E	A	R	R	A	U	D E
NERONI		*ALL	X	X	X	X	X	X	X	X	X X
Fine											
F3=Fine F5=Rivis. F6=Aggiunta nuovi utenti											
F10=Concessione con oggetto riferimento F12=Annullamento F24=Altri tasti											
(C) COPYRIGHT IBM CORP. 1980, 2013.											

- 12.4. Si osservi l'utente proprietario in alto a destra e le due righe che riportano il dettaglio delle autorizzazioni concesse. Ogni "X" documenta l'esistenza di una autorizzazione. Per interpretare una certa "X", si porti il cursore video sulla "X" presente o assente e si preme F1 per vedere il testo d'aiuto sensibile al contesto.
- 12.5. Lo stesso utente crei l'area dati NN nella libreria CORSOAUT e ne visualizzi le autorizzazioni.

```
> CRTDTAARA DTAARA(CORSOAUT/NN) TYPE(*CHAR) VALUE('Prova autorità') TEXT('Prova
autorità')
Area dati NN creata nella libreria CORSOAUT.
> EDTOBJAUT OBJ(CORSOAUT/NN) OBJTYPE(*DTAARA)
```

Modifica autorizzazione oggetto											
Oggetto	:	NN	Propriet.	:	NERONI						
Libreria	:	CORSOAUT	Gruppo principale	:	*NONE						
Tipo oggetto	:	*DTAARA	Unità ASP	:	*SYSBAS						
Immettere le modifiche alle autorizzazioni correnti e premere Invio.											
oggetto protetto da elenco autoriz.					:	*NONE					
Utente	Gruppo	Autoriz. oggi.	-----Oggetto-----					-----Dati-----			
*PUBLIC		*CHANGE	O	M	E	A	R	R	A	U	D E
NERONI		*ALL	X	X	X	X	X	X	X	X	X X
Fine											
F3=Fine F5=Rivis. F6=Aggiunta nuovi utenti											
F10=Concessione con oggetto riferimento F12=Annullamento F24=Altri tasti											
(C) COPYRIGHT IBM CORP. 1980, 2013.											

- 12.6. Si osservino gli stessi elementi del video precedente ma, in particolare, le autorizzazioni che il sistema ha impostato per l'utente pubblico. Queste, pur se desunte da lecite personalizzazioni di sistema, potrebbero non corrispondere ai desiderata attuali, né un programmatore di solito si impiccia o può impicciarsi delle fantasie dei sistemisti che si sfogano coi settaggi di sistema.
- 12.7. È naturalmente possibile intervenire sulle autorizzazioni tramite il video del comando di Edit Object Authority (quello dell'immagine precedente) o tramite i comandi qui di seguito illustrati.
- 12.8. Gestione delle autorizzazioni su un singolo oggetto.
- 12.8.1. GRTOBJAUT (Grant Object Authority). Concessione di una autorizzazione su un singolo oggetto.
- 12.8.2. RVKOBJAUT (Revoke Object Authority). Revoca di una autorizzazione su un singolo oggetto.
- 12.8.3. EDTOBJAUT (Edit Object Authority) Modifica interattiva di una autorizzazione su un singolo oggetto.

12.8.4. Modifica batch di una autorizzazione su un singolo oggetto eseguendo in sequenza prima RVKOBJAUT e poi GRTOBJAUT opportuni.

12.9. Tuttavia, la gestione e la fasatura delle autorizzazioni risulta onerosa per la numerosità degli interventi necessari, specialmente dopo la creazione di un gruppo nutrito di oggetti o nel caso di un cambio di strategia nel fornire le autorizzazioni a soggetti diversi dai precedenti. Da non trascurare che in una libreria rischiano di convivere strategie di autorizzazione diverse e contrastanti anche dove si volesse omogeneità e tempestività di attuazione di nuove strategie di autorizzazione.

12.10. Facendo però uso delle "Liste delle autorizzazioni", è possibile impostare le autorizzazioni in modo da settarle una volta sola in un oggetto specializzato e poi fare in modo che ogni oggetto creato in una libreria assuma tale settaggio automaticamente sin dalla nascita e a cura del sistema. Se in futuro si cambierà strategia di autorizzazione, basterà correggere la lista di autorizzazione della libreria per adeguare automaticamente tutti gli oggetti della libreria.

12.10.1. Esempio di impostazione della strategia di gestione delle autorizzazioni per libreria.

```
> CRTAUTL AUTL(CORSOAUT) TEXT('Prova autorita') AUT(*USE)
È stata creata la lista di autorizzazioni CORSOAUT.
> CRTLIB LIB(CORSOAUT) TYPE(*TEST) TEXT('Prova autorita') AUT(*LIBCRTAUT)
CRTAUT(CORSOAUT)
È stata creata la libreria CORSOAUT.
> CRTDTAARA DTAARA(CORSOAUT/NN) TYPE(*CHAR) VALUE('Prova autorita') TEXT('Prova
autorita')
Area dati NN creata nella libreria CORSOAUT.
> EDTOBJAUT OBJ(CORSOAUT/NN) OBJTYPE(*DTAARA)
```

Modifica autorizzazione oggetto											
Oggetto		Libreria	Tipo oggetto	Propriet.	Gruppo principale	Unità ASP					
NN		CORSOAUT	*DTAARA								
Immettere le modifiche alle autorizzazioni correnti e premere Invio.											
oggetto protetto da elenco autoriz.							CORSOAUT				
Utente	Gruppo	Autoriz. oggi.	-----Oggetto-----				-----Dati-----				
			O	M	E	A	R	R	A	U	D
*PUBLIC		*AUTL									
NERONI		*ALL	X	X	X	X	X	X	X	X	X
Fine											
F3=Fine F5=Rivis. F6=Aggiunta nuovi utenti											
F10=Concessione con oggetto riferimento F12=Annullamento F24=Altri tasti											
(C) COPYRIGHT IBM CORP. 1980, 2013.											

12.11. Si noti tuttavia che tale strategia dovrebbe essere applicata sin dall'inizio della vita di una libreria, già dal comando CRTLIB. Questo perché la modalità di assegnazione automatica delle autorizzazioni riguarda soltanto gli oggetti di nuova creazione dentro la libreria. Su una libreria preesistente si può comunque cambiare strategia tramite il comando CHGLIB, ma allora occorre adeguare le autorizzazioni di tutti gli oggetti preesistenti in libreria. Per farla breve, occorre un tool semplice in grado di agire automaticamente su tutti gli oggetti di una libreria.

12.12. Tool JAUTLIB.

12.12.1. Essendo faticoso indagare e normalizzare le autorizzazioni già assegnate ad una libreria e a tutti gli oggetti ivi contenuti, mi sono costruito un tool che impone con facilità e immediatezza la strategia di autorizzazione illustrata da ultimo.

12.12.2. Il tool JAUTLIB attribuisce infatti ad una libreria, e a tutti gli oggetti contenuti, una nuova lista autorizzazioni e, se richiesto, un nuovo proprietario.

12.12.3. Azioni svolte dal tool sulla libreria:

12.12.4. Se richiesto, cambia proprietario alla libreria.

12.12.4.1. Impone la nuova lista autorizzazioni alla creazione di oggetti in libreria.

12.12.4.2. La libreria richiesta viene assoggettata alla nuova lista autorizzazioni.

12.12.4.3. Viene tolta ogni altra autorizzazione esplicita alla libreria.

12.12.5. Solo se tutte le azioni sulla libreria hanno esito positivo, il tool procede alla modifica degli oggetti contenuti nella libreria.

12.12.6. Azioni svolte dal tool su ogni oggetto della libreria:

12.12.6.1. Se richiesto, cambia proprietario all'oggetto.

12.12.6.2. L'oggetto viene assoggettato alla nuova lista autorizzazioni.

12.12.6.3. Viene tolta ogni altra autorizzazione esplicita all'oggetto.

- 12.12.7. Il comando è destinato all'uso di un utente autorevole, cioè, autorevole per l'uso dei comandi necessari, del contenuto della libreria da manipolare e dei profili proprietari degli oggetti prima e dopo l'intervento. Si sconsiglia l'uso del tool con utenze di autorità limitata perché si corre il rischio di togliersi autorità su una libreria ed i suoi oggetti senza avere l'autorità di riassegnarli all'utenza destinataria.
- 12.12.8. L'help completo è a bordo del comando medesimo, raggiungibile premendo F1 sul prompter del comando.

13. Cambio lingua

13.1. Prerequisito.

13.1.1. Sul sistema deve essere installata la lingua di destinazione. Gli oggetti della lingua principale sono installati nella libreria QSYS.

Lingua	Libreria
Principale	QSYS
Italiano	QSYS2932
Inglese	QSYS2924
Francese	QSYS2928
Tedesco	QSYS2929

13.1.2. La lingua principale (qualunque sia) non ha una sua libreria sul sistema. Quindi, una installazione con lingua principale in italiano non possiederà una libreria QSYS2932.

13.2. Se si possiede l'autorità di cambiare la system library list (uso del comando CHGSYSLIBL).

13.2.1. Anteporre la libreria della lingua alla system library list.

13.2.2. La sessione corrente si presenta subito nella lingua richiesta.

13.3. Se NON si possiede l'autorità di cambiare la system library list (uso del comando CHGSYSLIBL).

13.3.1. Cambiare il proprio profilo utente con uno dei comandi seguenti, secondo la lingua.

CHGPRF LANGID(ITA) CNTRYID(IT)

CHGPRF LANGID(ENG) CNTRYID(EN)

CHGPRF LANGID(FRA) CNTRYID(FR)

CHGPRF LANGID(DEU) CNTRYID(DE)

13.3.2. Eseguire SIGNOFF e ricollegarsi con lo stesso profilo. La sessione si presenterà nella lingua richiesta.

14. Descrizione lavori

- 14.1. La descrizione lavori (Job Description) ha il tipo oggetto *JOB_D e contiene informazioni da assegnare ad un lavoro quando questo viene creato sul sistema.
- 14.2. Viene spesso gestita dai sistemisti ai quali, nel caso, va chiesta sia la creazione che la modifica.
- 14.3. L'elenco dei comandi che la riguardano può essere ottenuto con
> GO MENU (CMDJOB_D)
- 14.4. I lavori interattivi utilizzano la descrizione lavori presente sul profilo utente che spesso è quella di default QGPL/QDFTJOB_D perché le caratteristiche del lavoro interattivo possono essere facilmente manipolate dal comando CHGJOB (Change Job) che permette di cambiare la più parte delle caratteristiche inizialmente mutate dalla descrizione lavori.
- 14.5. Invece, i lavori batch, che costituiscono la maggior parte delle elaborazioni in ambienti bancari e finanziari, prendono le caratteristiche cablate nelle descrizioni lavori strettamente collegate ad ogni singolo ambiente elaborativo. Il lavoro batch nasce dal comando SBMJOB (Submit Job) che impone quale descrizione lavori usare.
- 14.6. In particolare, la job_d fornisce al job che la usa: la lista librerie, la coda di immissione, la coda di emissione, a volte un utente particolare, la modalità di log, le priorità di input e di output.

15. Definizione di file fisici e logici con DDS per uno schema Testata-Righe

15.1. Level Check

15.1.1. Identificativo univoco del tracciato record.

15.1.1.1. È un identificativo univoco che il sistema operativo assegna automaticamente ad ogni tracciato di record di file di qualunque tipo in qualunque modo generato. In particolare, ai record di file fisici e logici, ma anche ai display file.

15.1.2. Basato su quali caratteristiche.

15.1.2.1. È basato sul nome del tracciato e su nome, tipo, lunghezza, numero e sequenza dei campi che compongono il tracciato.

15.1.2.2. Cambia se cambia una delle caratteristiche di base.

15.1.2.3. È uguale per file uguali su sistemi diversi.

15.1.2.4. Contiene tredici caratteri esadecimali (scelti tra 0123456789ABCDEF).

15.1.3. Che uso ne fanno i programmi.

15.1.3.1. Viene annotato al momento della compilazione nei programmi che usano i tracciati come descritti esternamente.

15.1.3.2. Quando un programma viene eseguito, al momento dell'apertura di un file che contiene tracciati descritti esternamente, il sistema operativo controlla la corrispondenza tra il levelcheck al momento della compilazione e il levelcheck al momento dell'esecuzione. Se non corrispondono, genera un errore grave.

15.1.3.3. In sostanza, l'uso del levelcheck garantisce la necessaria congruenza tra programmi e file usati dai programmi e assicura che i campi dei file contengano byte congruenti con la definizione del tracciato incorporata alla nascita nel file stesso.

15.1.4. Conseguenze di LVLCHK(*NO).

15.1.4.1. Purtroppo, nel passato, molti hanno deciso di evitare la revisione e la ricompilazione di tutti i programmi che usano un file in occasione di un cambio tracciato. In questo caso deprecabile, i comandi di creazione dei file utilizzano l'opzione LVLCHK(*NO) che evita il controllo e permette ad un programma di usare per il file un tracciato vecchio senza segnalare l'errore in apertura.

15.1.4.2. La scelta LVLCHK(*NO) comporta anche che il programma legga e scriva su un tracciato nuovo rispettando le regole di un vecchio tracciato provocando potenzialmente gravi errori nei dati. Gli errori così introdotti verranno rilevati solo in occasione di una lettura con un tracciato più recente del file rispetto al tracciato usato per provocare il danno.

15.1.4.3. In un solo caso il LVLCHK(*NO) non provoca danni: se campi nuovi alfanumerici vengono aggiunti in coda ad un vecchio tracciato. Dopo l'aggiunta, un vecchio programma vi leggerà dei blank, non procederà ad alcun update su quei campi e vi scriverà dei blank in caso di aggiunta. Tuttavia, questa situazione, apparentemente sopportabile, non può essere facilmente controllata e garantita e lascia la porta aperta agli errori nei dati senza evidenze automatiche e tempestive.

15.1.4.4. Se il file prevede LVLCHK(*YES), è possibile aggirare gli obblighi che ne derivano imponendogli provvisoriamente LVLCHK(*NO) tramite opportuna override.

15.1.4.5. Se il file prevede LVLCHK(*NO), NON è possibile imporgli una override che lo costringa a comportarsi come se fosse creato con LVLCHK(*YES).

15.1.5. Uso nell'SQL.

15.1.5.1. L'SQL non prevede invece la possibilità di generare file con LVLCHK(*NO).

15.1.5.2. La scrittura in un file tramite SQL non permette di introdurre errori nei dati.

15.2. Per illustrare i concetti del database, usiamo uno schema **Testata-Righe**. Supponiamo di creare alcuni file ed oggetti esemplificativi nella libreria del corso usando l'interfaccia PDM o l'interfaccia QCMD con i comandi STRSEU e i vari comandi del gruppo "Create..."

15.2.1. Creare il sorgente del file fisico testate CNU1.

15.2.2. Compilare l'oggetto del file fisico testate con CRTPF (Create physical file).

15.2.3. Creare il sorgente del file logico testate CNU11 sul fisico CNU1 con chiave unica documento.

15.2.4. Compilare l'oggetto del file logico testate CNU11 con CRTLF (Create logical file).

15.2.5. Creare il sorgente del file fisico righe CNU2.

15.2.6. Compilare l'oggetto del file fisico righe con CRTPF (Create physical file).

15.2.7. Creare il sorgente del file logico righe CNU21 sul fisico NU2.

15.2.8. Compilare l'oggetto del file logico righe CNU21 con CRTLF (Create logical file).

15.2.9. Creare il sorgente del file file logico multifisico CNUX che riunisce testata e riga con chiave documento-riga.

15.2.10. Compilare l'oggetto del file logico multifisico CNUX con CRTLF (Create logical file).

15.3. Sorgente file fisico testata

15.3.1. Esempio di file fisico. Si osservino i valori evidenziati.

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
* Claudio Neroni 10-11-2020 Creato per Corso
*   CNU1
*   Testata documento. Fisico
```

Nome del record e Testo del record:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A           R N1
A                                     TEXT('Testata documento')
```

Nome Lunghezza Tipo Decimali Intestazioni di colonna dei campi:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A           N1DOC                5S 0
A                                     COLHDG('Numero'      +
A                                     'documento'      )
A           N1TIP                 3
A                                     COLHDG('Tipo'        +
A                                     'doc'          )
```

15.4. Creazione file fisico testata

15.4.1. Sull'elenco membri sorgenti del PDM si usi l'opzione 14 oppure, sul video comandi, il seguente comando:

```
> CRTPF FILE(CORSO/CNU1) SRCFILE(CORSO/CORSO2) LVLCHK(*YES)
File CNU1 creato nella libreria CORSO.
Membro CNU1 aggiunto al file CNU1 in CORSO.
```

15.5. Sorgente file logico testata con chiavi

15.5.1. Esempio di file logico. Si osservino i valori evidenziati.

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
* Claudio Neroni 10-11-2020 Creato per Corso
*   CNU11
*   Testata documento. Logico per numero documento
```

Chiave unica:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A                                     UNIQUE
```

Nome del record e Nome del file fisico su cui si basa il file logico:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A           R N1                    PFILE(CNU1)
```

Elenco dei campi chiave del file logico:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A           K N1DOC
```

15.6. Creazione file logico testata

15.6.1. Sull'elenco membri sorgenti del PDM si usi l'opzione 14 oppure, sul video comandi, il seguente comando:

```
> CRTLF FILE(CORSO/CNU11) SRCFILE(CORSO/CORSO) LVLCHK(*YES)
File CNU11 creato nella libreria CORSO.
Membro CNU11 aggiunto al file CNU11 in CORSO.
```

15.7. Sorgente file fisico riga

15.7.1. Come per la testata. Si riporta solo il sorgente.

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
* Claudio Neroni 10-11-2020 Creato per Corso
*   CNU2
*   Riga documento. Fisico
A           R N2
A                                     TEXT('Riga documento')
A           N2DOC                5S 0
A                                     COLHDG('Numero'      +
A                                     'documento'      )
A           N2RIG                 3S 0
A                                     COLHDG('Numero'      +
A                                     'riga'          )
A           N2TXT                 50
A                                     COLHDG('Testo'        +
A                                     'riga'          )
```

15.8. Creazione file fisico riga

15.8.1. Sull'elenco membri sorgenti del PDM si usi l'opzione 14 oppure, sul video comandi, il seguente comando:

```
> CRTPF FILE(CORSO/CNU2) SRCFILE(CORSO/CORSO) LVLCHK(*YES)
File CNU2 creato nella libreria CORSO.
Membro CNU2 aggiunto al file CNU2 in CORSO.
```

15.9. Sorgente file logico riga con chiavi

15.9.1. Come per la testata. Si riporta solo il sorgente.

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
* Claudio Neroni 10-11-2020 Creato per Corso
* CNU21
* Riga documento. Logico per numero documento e riga
A                                     UNIQUE
A      R N2                          PFILE(CNU2)
A      K N2DOC
A      K N2RIG
```

15.10. Creazione file logico riga

```
> CRTLF FILE(CORSO/CNU21) SRCFILE(CORSO/CORSO) LVLCHK(*YES)
File CNU21 creato nella libreria CORSO.
Membro CNU21 aggiunto al file CNU21 in CORSO.
```

15.11. Sorgente file logico multifisico testata e riga con chiavi

15.11.1. Questo tipo di file è utile solo:

15.11.1.1. Con dfu (Data File Utility) perché vede i record in ordine.

15.11.1.2. In stampe di massa perché utilizzano il ciclo automatico RPG di lettura del logico.

15.11.2. Se ne sconsiglia l'uso altrove.

15.11.3. Si osservino i valori evidenziati.

Record Fisici di base Elenco dei campi chiave dei record logici:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
* Claudio Neroni 10-11-2020 Creato per Corso
* CNUX
* Testata e riga documento. Logico multifisico
A      R N1                          PFILE(CNU1)
A      K N1DOC
A      K *NONE
A      R N2                          PFILE(CNU2)
A      K N2DOC
A      K N2RIG
```

15.11.4. Il numero di campi chiave deve essere lo stesso per i due tracciati. Non avendo un campo riga sulla testata, si utilizza al suo posto *NONE che si comporta come se ci fosse un campo riga con valore *LOVAL (= tutti esadecimali x'00').

15.12. Creazione file logico multifisico

15.12.1. Sull'elenco membri sorgenti del PDM si usi l'opzione 14 oppure, sul video comandi, il seguente comando:

```
> CRTLF FILE(CORSO/CNUX) SRCFILE(CORSO/CORSO) LVLCHK(*YES)
File CNUX creato nella libreria CORSO.
Membro CNUX aggiunto al file CNUX in CORSO.
```

15.13. Come modificare il tracciato di un fisico senza perdere i dati contenuti.

15.13.1. Processo standard.

15.13.1.1. Modifica al sorgente.

15.13.1.2. Cancellazione logici soprastanti.

15.13.1.3. Ridenominazione del vecchio fisico.

15.13.1.4. Ricompilazione del fisico e dei suoi logici.

15.13.1.5. Copia del contenuto del vecchio file fisico nel nuovo.

15.13.2. Processo se la modifica prevede solo aggiunta o allargamento di campi.

15.13.2.1. Copia del file fisico con altro nome per sicurezza.

15.13.2.2. Modifica del file fisico e di tutti i suoi logici con il solo comando CHGPF del fisico puntando al nuovo sorgente.

```
> CHGPF FILE(CORSO/CNU1) SRCFILE(CORSO/CORSO)
```

Il file CNU1 nella libreria CORSO è stato modificato.

15.13.2.3. Il fisico adegua il proprio tracciato e mantiene i dati.

15.14. Visualizzazione delle caratteristiche di un file.

15.14.1. I file possono essere indagati con il comando DSPFD (Display File Description), che restituisce informazioni sui file. Richiamabile anche come opzione PDM "5" dal video di un WRKOBJPDM (Work

Object PDM). Ad esempio, sul file fisico CNU1 si esegua il comando seguente. Il video risultante porta una serie ampia di informazioni.

> DSPFD FILE(CORSO/CNU1)

```

15/11/20                                Visualizzazione descrizione file
Immissione comando DSPFD
File . . . . . : FILE          CNU1
Libreria. . . . . :             CORSO
Tipo di informazioni . . . . . : TYPE          ALL
Attributi file . . . . . : FILEATR    *ALL
Sistema . . . . . : SYSTEM      *LCL

Intestazione descrizione file
File . . . . . : FILE          CNU1
Libreria. . . . . :             CORSO
Tipo di file . . . . . :         Fisico
Tipo di file . . . . . : FILETYPE  *DATA
ID ASP (auxiliary storage pool) . . . . . : 00001

Attributi del file di data base
File descritto esternamente . . . . . : Si
Identificativo livello file . . . . . : 1201115171533
Data di creazione . . . . . : 10/11/20
'Descrizione' testuale. . . . . : TEXT      Testata documento
File distribuito. . . . . : No
Tabella SQL partizionata. . . . . : No
A capacità DBCS . . . . . : No
Numero massimo di membri. . . . . : MAXMBRS    1
Numero di vincoli . . . . . : 0
Numero di trigger . . . . . : 0
Numero di membri. . . . . : 1
Dimensione del membro . . . . . : SIZE
Numero iniziale di record . . . . . : 10000000
Incremento numero di record . . . . . : 32000
Numero massimo di incrementi . . . . . : 3
Capacità record . . . . . : 10096000
Allocazione memoria . . . . . : ALLOCATE  *NO
Memoria contigua . . . . . : CONTIG      *NO
Unità di memoria preferita . . . . . : INDIR   *ANY
Record per cui forzare una scrittura. . . . : FRCRATIO *NONE
Tempo massimo di attesa file. . . . . : WAITFILE 30
Tempo massimo di attesa record . . . . . : WAITRCD 60
% massima consentita di record cancellati . : DLTPCT  *NONE
Riutilizzare record cancellati. . . . . : REUSEDLT *NO
CCSID (Coded character set identifier). . . : CCSID 280
Consentire operazione di lettura . . . . . : Si
Consentire operazione di scrittura. . . . . : Si
Consentire operazione di aggiornamento . . : ALWUPD  *YES
Consentire operazione di cancellazione . . : ALWDLT  *YES
Controllo a livello di formato record . . . : LVLCHK  *NO
Percorso accesso. . . . . : Arrivo
Dimensione percorso accesso . . . . . : ACCPTHsiz *MAX1TB
Dimensione pagina logica percorso accesso. . : PAGEsize *KEYLEN
Lunghezza massima record. . . . . : 8
Volatile . . . . . : No
Conservazione in memoria. . . . . : KEEPINMEM *NO
File correntemente registrato su giornale . : No

Descrizione del percorso di accesso
Percorso accesso. . . . . : Arrivo
Sequenza di ordinamento. . . . . : SRTSEQ    *HEX
Identificativo lingua. . . . . : LANGID     ITA

Descrizione del membro
Membro . . . . . : MBR          CNU1
Identificativo livello membro. . . . . : 1201110110349
Data di creazione membro. . . . . : 10/11/20
'Descrizione' testuale. . . . . : TEXT      Testata documento
Data di scadenza per membro. . . . . : EXPDATE *NONE
Dimensione del membro . . . . . : SIZE
Numero iniziale di record . . . . . : 10000000
Incremento numero di record . . . . . : 32000
Numero massimo di incrementi . . . . . : 3
Numero corrente di incrementi . . . . . : 0
Capacità record. . . . . : 10096000
Numero corrente di record. . . . . : 3
Numero di record cancellati . . . . . : 0
Allocazione memoria . . . . . : ALLOCATE  *NO
Memoria contigua. . . . . : CONTIG      *NO
Unità di memoria preferita. . . . . : INDIR   *ANY
Record di cui forzare una scrittura. . . . : FRCRATIO *NONE
Condivisione percorso dati aperto . . . . : SHARE    *NO
% massima consentita di record cancellati . : DLTPCT  *NONE
Statistiche attività spazio dati . . . . . :
Dimensione spazio dati in byte. . . . . : 8192
Accessi aperti file fisico . . . . . :
Accessi chiusi file fisico . . . . . :

```

```

Operazioni scrittura. . . . . :
Operazioni aggiornamento . . . . . :
Operazioni cancellazione. . . . . :
Letture logiche . . . . . : 12
Letture fisiche . . . . . :
Operazioni di ripulitura . . . . . :
Operazioni di copia spazio dati . . . . . :
Operazioni di riorganizzazione. . . . . :
Creazioni/ricreazioni percorsi di accesso : 4
Record rifiutati mediante sel chiave. . . :
Record rif. mediante sel senza chiave . . :
Record rif. mediante sel raggr. per . . . :
Statistiche attività percorso accesso . . :
Letture logiche percorso accesso. . . . :
Letture fisiche percorso accesso. . . . :
Condivisione implicita percorso accesso . : No
Data/ora ultima modifica. . . . . : 15/11/20 17:15:34
Data/ora ultimo salvataggio . . . . . : 14/11/20 18:21:33
Data/ora ultimo ripristino. . . . . :
Data ultimo utilizzo. . . . . :
Conteggio giorni di utilizzo. . . . . : 0
Data reimpostazione . . . . . :
Oggetto ripristinato con transazioni parziali,
richiesta applicazione modifiche giornale. . . : *NO
Esiste una transazione parziale,
rollback terminato. . . . . : *NO
Avvio ricevitore di giornale da applicare . . . :
Libreria. . . . . :
Unità ASP . . . . . :
Elenco dei formati record
Formato      Campi      Lungh.  Identificativo
record      livello formato
N1          2          8  2645070A4DF4A
Testo. . . . . : Testata documento
Numero totale di formati. . . . . : 1
Numero totale di campi. . . . . : 2
Lunghezza totale record. . . . . : 8
Elenco membri
Membro      Dim.      Tipo Data      Ultima modifica
orig. creaz. Data      Ora      Record      Record
CNU1          8192      orig. creaz. Data      Ora      Record      cancellati
CNU1          8192      10/11/20 15/11/20 17:15:34      3          0
Testo: Testata documento
Numero totale di membri. . . . . : 1
Numero totale di membri non disponibili . : 0
Record totali . . . . . : 3
Totale record cancellati. . . . . : 0
Totale delle dimensioni membro . . . . . : 8192

```

15.15. Visualizzazione del tracciato di un file.

15.15.1. I file possono essere indagati con il comando DSPFFD (Display File Field Description), che permette di visionare l'elenco dei campi del tracciato.

> DSPFFD FILE(CORSO/CNU1)

```

Visualizzazione descrizione campi del file
Parametri di immissione
File . . . . . : CNU1
Libreria . . . . . : *LIBL
Inform. sul file
File . . . . . : CNU1
Libreria . . . . . : CORSO
Ubicazione del file . . . . . : *LCL
Descrizione esternamente . . . . . : Si
Numero di formati record . . . . . : 1
Tipo di file . . . . . : Fisico
Data di creazione del file . . . . . : 10/11/20
Testo 'descrittivo' . . . . . : Testata documento
Informazioni sul formato record
Formato record . . . . . : N1
Identificativo livello formato . . . . . : 2645070A4DF4A
Numero di campi . . . . . : 2
Lunghezza record . . . . . : 8
Testo formato . . . . . : Testata documento
Informazioni a livello di campo
Campo      Tipo      Lungh  immiss Posizione      Uso      Intestazione
campo      tipo      campo  buffer  buffer      Campo      Colonna
N1DOC      ZONED      5 0      5      1      Entrambi Numero
documenti
Testo del campo . . . . . : Numero documento
N1TIP      CHAR      3      3      6      Entrambi Tipo
doc
Testo del campo . . . . . : Tipo doc
Identificativo serie caratteri codificati: 280

```


15.16. Visualizzazione delle relazioni tra fisici e logici.

15.16.1. I file fisici possono essere indagati con il comando **DSPDBR** (Display Data Base Relations), che permette di visionare l'elenco dei logici che dipendono dal fisico.

> **DSPFFD FILE(CORSO/NU1)**

15/11/20 Visualizzazioni relazioni database				
Immissione comando DSPDBR				
File	.	.	.	FILE NU1
Libreria.	.	.	.	CORSO
Membro	.	.	.	*NONE
Formato record	.	.	.	*NONE
Emissione	.	.	.	*
Specifiche				
Tipo di file	.	.	.	Fisico
File	.	.	.	NU1
Libreria.	.	.	.	CORSO
Membro	.	.	.	*NONE
Formato record	.	.	.	*NONE
Numero di file dipendenti	.	.	.	2
File dipendenti dal file specificato				
File dipendente	Libreria	Data	JREF	Restrizione
NU11	CORSO	dipendenza		
NUX	CORSO	dipendenza		

15.17. Membri fisici.

15.17.1. Un file fisico è sostanzialmente un contenitore di informazioni. Innanzitutto delle proprie caratteristiche globali, poi delle caratteristiche del formato e dei suoi campi.

15.17.2. In particolare, la definizione del tracciato viene messa a disposizione dei programmi al momento della loro compilazione in modo tale che in esecuzione possano leggere e scrivere i dati rispettando il tracciato stesso.

15.17.3. Un file fisico contiene da zero a molti membri fisici.

15.17.4. I membri fisici sono i contenitori di separati gruppi di record di dati.

15.17.5. Normalmente i file dati sono monomembro per evitare una maggiore complessità nell'uso. Un file monomembro è ovviamente più maneggevole anche se questo finisce per creare una minor distinzione tra dati da tenere e dati da sfortire e comporta membri più affollati del ragionevole.

15.17.6. La struttura "multimembro" gestisce nel file una sola definizione per molti membri, permettendo un risparmio di spazio ma, nel tempo, è stata poco usata se non per i file che contengono sorgenti o per l'accantonamento periodico di dati storici.

15.17.7. Il reindirizzamento permette di chiedere al programma di usare un file con un nome diverso da quello cablato nel programma o/e in una libreria fuori dalla lista librerie del lavoro.

15.17.8. Il reindirizzamento di un membro è possibile usando il nome membro nel comando **OVRDBF** (Override Data Base File), altrimenti viene usato il primo aggiunto al file, di solito quello creato in compilazione insieme al file e omonimo del file stesso.

15.17.9. Occorre poi dire che, mentre in nativo basta utilizzare il comando **OVRDBF** prima di aprire un membro in un programma, in SQL il reindirizzamento ad un membro è macchinoso perché si richiede per farlo la creazione di un oggetto intermediario permanente SQL chiamato **ALIAS**.

15.17.10. Il comando di reindirizzamento permette inoltre di variare temporaneamente molte delle caratteristiche che risiedono nella definizione del file.

15.18. Membri logici.

15.18.1. Solitamente l'unico membro logico di un file logico indirizza l'unico membro fisico di un file fisico.

15.18.2. Se presente nel file fisico un secondo membro fisico, si aggiungerà nel logico un secondo membro logico che indirizzerà il secondo membro fisico e così via.

15.18.3. Possibile ma deprecabile creare un membro logico che vede una scelta arbitraria di membri fisici.

15.19. Gestione elementare dei dati nei file fisici e logici.

15.19.1. Per la gestione elementare dei file fisici e logici, ad esempio su fisici tabellari di bassa mobilità, è ragionevole usare il tool generalizzato di gestione dati (no campi variabili).

15.19.2. L'opzione 18 su un file di database dell'elenco oggetti **WRKOBJPDM** o il comando **UPDDTA** sullo stesso file chiamano l'interfaccia di immissione e modifica dei record fornita dal tool di sistema **DFU** (Data File Utility).

> **UPDDTA FILE(CORSO/CNU11)**

15.19.3. I soli controlli esercitati dal Dfu sono quelli che costringono a rispettare le keyword scritte nelle DDS che hanno generato il file fisico che contiene i dati.

15.20. Visualizzazione elementare dei dati nei file fisici e logici.

15.20.1. Si può usare il tool di sistema Query con il comando **RUNQRY**.

15.20.2. Per visualizzare estemporaneamente il contenuto dei record presenti nei dati dei file fisici e logici, è possibile usare la visualizzazione estemporanea dei dati fornita dal tool di sistema Query e richiamabile con il comando **RUNQRY**. Unico difetto: i record sono nell'ordine d'arrivo del fisico anche quando il fisico o il logico interrogati hanno una chiave.

```
> RUNQRY QRYFILE((CORSO/CNU11))
```

```
Riga      ....+....1..
          Numero  Tip
          doc      doc
000001      1    FAT
000002      2    NOT
000003      3    VUO
000004      5    VEN
***** Fine prospetto *****
```

```
> RUNQRY QRYFILE((CORSO/CNU21))
```

```
Riga      ....+....1....+....2....+....3....+....4
          Numero  Num  Testo
          doc      rig  riga
000001      1      10  PRIMA riga
000002      1     120- Seconda RIGA
000003      2       1  Tanto
000004      2       2  va la
000005      2       3  gatta
000006      2       4  al lardo
000007      2       5  che ci lascia lo zampino.
000008      4       1  Riga senza testata
000009      5      10  Nel
000010      5      20  mezzo
000011      5      30  del
000012      5      40  cammin
000013      5      50  di
000014      5      60  nostra
000015      5      70  vita
000016      5      80  mi
000017      5      90  ritrovai
000018      5     100  per
000019      5     110  una
000020      5     120  selva
000021      5     130  oscura
000022      5     140  che
000023      5     150  la
000024      5     160  diritta
000025      5     170  via
000026      5     180  era
000027      5     190  smarrita
000028      5     200  Ahi
000029      5     210  quanto
000030      5     220  a
000031      5     230  dir ...
***** Fine prospetto *****
```

15.21. Gestione con altri tool dei dati nei file fisici e logici.

15.21.1. IBMi è privo di un tool nativo decente per la modifica del database.

15.21.2. A pagamento si può usare Work with Database File (**WRKDBF**). Collaudatissimo da quando era freeware, molto utile. Ora costa 500\$ su un solo serial number.

15.21.3. Oppure DBU di ProData, più costoso, molto potente. Fornisce il prezzo solo chiedendo il preventivo.

16. Schema Anagrafica-Tabelle

- 16.1. A integrazione dello schema **Testata-Righe**, si discute brevemente di uno schema più elementare **Anagrafica-Tabelle** che, oltre ad essere uno schema indispensabile negli sviluppi applicativi, risulta essere anche un punto di ingresso più intuitivo per comprendere la struttura di database.
- 16.2. Per illustrare lo schema **Anagrafica-Tabelle**, supponiamo di creare alcuni file ed oggetti esemplificativi nella libreria del corso usando l'interfaccia PDM o l'interfaccia QCMD con i comandi **STRSEU** e i vari comandi del gruppo "Create...".
- 16.3. Sorgente file fisico anagrafica persone
- 16.3.1. Si osservino i valori evidenziati.

```
* Claudio Neroni 19-02-2021 Creato per Corso
*   CNA1
*   Anagrafica persone
*
```

Nome e Testa del record:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A          R A1
A                                     TEXT('Anagrafica persone')
```

Nome Lunghezza Intestazioni di colonna dei campi:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A          A1CDPE          6          COLHDG('Codice' 'persona')
A          A1DEPE          50         COLHDG('Descrizione' 'persona')
A          A1PRNA          2          COLHDG('Provincia' 'nascita')
A          A1PRRE          2          COLHDG('Provincia' 'residenza')
```

- 16.4. Oggetto file fisico anagrafica persone

16.4.1. Sull'elenco membri sorgenti del PDM si usi l'opzione 14 oppure, sul video comandi, il seguente comando:

```
> CRTPF FILE(CORSO/CNA1) SRCFILE(CORSO/CORSO) LVLCHK(*YES)
File CNA1 creato nella libreria CORSO.
Membro CNA1 aggiunto al file CNA1 in CORSO.
```

- 16.5. Sorgente file fisico tabella provincie

16.5.1. Si osservino i valori evidenziati.

```
* Claudio Neroni 19-02-2021 Creato per Corso
*   CNT1
*   Tabella provincie
*
```

Nome e Testa del record:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A          R T1
A                                     TEXT('Tabella provincie')
```

Nome Lunghezza Intestazioni di colonna dei campi:

```
A.....T.Nome+++++.Lun++TPdB.....Funzioni+++++
A          T1CDPR          2          COLHDG('Codice' 'provincia')
A          T1DEPR          50         COLHDG('Descrizione' 'provincia')
```

- 16.6. Oggetto file fisico tabella provincie

16.6.1. Sull'elenco membri sorgenti del PDM si usi l'opzione 14 oppure, sul video comandi, il seguente comando:

```
> CRTPF FILE(CORSO/CNT1) SRCFILE(CORSO/CORSO) LVLCHK(*YES)
File CNT1 creato nella libreria CORSO.
Membro CNT1 aggiunto al file CNT1 in CORSO.
```

- 16.7. Si provveda come si crede a scrivere qualche record persona nel file CNA1 e qualche record provincia del file CNT1.

16.7.1. Ad esempio di esegua RUNSQLSTM sul sorgente CNA1S.

```
INSERT INTO CORSO/CNA1
(A1CDPE, A1DEPE, A1PRNA, A1PRRE)
VALUES
('BLU', 'Barba Blu', 'CO', 'GE');
INSERT INTO CORSO/CNA1 VALUES
('JACK', 'Jack lo squartatore', 'GE', 'CO');
INSERT INTO CORSO/CNA1 VALUES
('AMEN', 'Alleluia', 'MI', 'RM');
```

16.7.2. Stessa cosa per il sorgente CNT15

```
INSERT INTO CORSO/CNT1
(T1CDPR, T1DEPR)
VALUES
('CO', 'Como');
INSERT INTO CORSO/CNT1 VALUES
('GE', 'Genova');
INSERT INTO CORSO/CNT1 VALUES
('MI', 'Milano');
```

16.8. La definizione dei soli file fisici e il loro riempimento permette di usare subito dei comandi SQL.

16.8.1. Si copiaincolla in STRSQL il comando contenuto nel sorgente CNA1SQL e lo si esegue.

```
-- Claudio Neroni 21-02-2021 Creato per Corso
-- CNA1SQL
-- Anagrafica persone
-- Elenco anagrafica con decodifica
-- Per eseguire, copiaincollare le righe a partire da SELECT
-- sul video di uno STRSQL con la lib CORSO in lista.
SELECT
A1CDPE,
SUBSTR(A1DEPE, 1, 20)
AS "Descrizione      persona",
A1PRNA,
SUBSTR((SELECT T1DEPR from CNT1
WHERE A1.A1PRNA = T1CDPR), 1, 20)
AS "Descrizione      provincia nascita",
A1PRRE,
SUBSTR((SELECT T1DEPR from CNT1
WHERE A1.A1PRRE = T1CDPR), 1, 20)
AS "Descrizione      provincia residenza"
FROM CNA1 A1
ORDER BY A1CDPE
```

16.8.2. Si ottiene la seguente visualizzazione.

Codice	"Descrizione	Provincia	"Descrizione	Provincia	"Descrizione
persona	persona"	nascita	provincia nascita"	residenza	provincia
BLU	Barba Blu	CO	Como	GE	Genova
JACK	Jack lo squartatore	GE	Genova	CO	Como
AMEN	Alleluia	MI	Milano	RM	-

Dove si osserva che le due provincie presenti sui record persone sono correttamente decodificate senza fare ricorso ad altre definizioni di database.

16.9. Per fare uso invece dei dati dei fisici nei programmi RPG o Cobol, è necessario definire chiavi di accesso che, pur essendo possibile, non è opportuno prevedere direttamente nei fisici.

16.9.1. Si procede quindi alla definizione di un file logico con chiave codice persona per il file fisico persone.

```
* Claudio Neroni 19-02-2021 Creato per Corso
* CNA11
* Anagrafica persone
*
A                               UNIQUE
A      R A1                     PFILE(CNA1)
A      K A1CDPE
```

16.9.2. Allo stesso modo, un file logico con chiave codice provincia per il file fisico provincie.

```
* Claudio Neroni 21-02-2021 Creato per Corso
* CNT11
* Tabella provincie
*
A                               UNIQUE
A      R T1                     PFILE(CNT1)
A      K T1CDPR
```

16.9.3. Creati gli oggetti al solito modo.

```
> CRTLF FILE(CORSO/CNA11) SRCFILE(CORSO/CORSO) SRCMBR(CNA11)
File CNA11 creato nella libreria CORSO.
Membro CNA11 aggiunto al file CNA11 in CORSO.
> CRTLF FILE(CORSO/CNT11) SRCFILE(CORSO/CORSO) SRCMBR(CNT11)
File CNT11 creato nella libreria CORSO.
Membro CNT11 aggiunto al file CNT11 in CORSO.
```

16.9.4. Si osservi il sorgente di un RPG di stampa che fa uso dei logici appena creati.

```
* Claudio Neroni 19-02-2021 Creato per Corso
* CNA1PRT1
* Anagrafica persone stampa
*-----
* Anagrafica persone logico codice
FCNA11  if  e          k disk
* Provincie.
FCNT11  if  e          k disk
* Stampa.
FCNPRT  o   f  75      printer
```

```

*-----
D                                     ds
D A1DEPE
D A1DEPE$                               1      20
D                                     ds
D A1PRNAD
D A1PRNAD$                             1      20
D                                     ds
D A1PRRED
D A1PRRED$                             1      20
*-----
* Intestazione.
C                                     except    INT1
C                                     except    INT2
C                                     except    INT3
* Cicla lettura.
C                                     do          *hival
*   Legge persona.
C                                     read      A1
*   Se non trova, abbandona.
C                                     if        %eof
C                                     leave
C                                     endif
*   Cerca provincia di nascita su provincie.
C A1PRNA chain T1
*   Se trova, accantona decodifica.
C                                     if        %found
C *like define T1DEPR A1PRNAD
C eval A1PRNAD = T1DEPR
*   Se non trova, accantona interrogativi.
C                                     else
C eval A1PRNAD = *all'?'
C                                     endif
*   Cerca provincia di residenza su provincie.
C A1PRRE chain T1
*   Se trova, accantona decodifica.
C                                     if        %found
C *like define T1DEPR A1PRRED
C eval A1PRRED = T1DEPR
*   Se non trova, accantona interrogativi.
C                                     else
C eval A1PRRED = *all'?'
C                                     endif
*   Stampa anagrafica persona.
C                                     except    A1P
* Cicla lettura.
C                                     enddo
* Chiude
C                                     seton                                1r
*-----
OCNPRT e INT1 1
O +0 'Codice'
O +1 'Descrizione'
O +1 'Pr'
O +1 'Descrizione'
O +1 'Pr'
O +1 'Descrizione'
OCNPRT e INT2 1
O +0 'person'
O +1 'persona'
O +1 'na'
O +1 'provincia nascita'
O +1 're'
O +1 'provincia residenza'
OCNPRT e INT3 1
O +0 '-----'
O +1 '-----'
O +1 '--'
O +1 '-----'
O +1 '--'
O +1 '-----'
OCNPRT e SPA 1
OCNPRT e A1P 1
O A1CDPE +0
O A1DEPE$ +1
O A1PRNA +1
O A1PRNAD$ +1
O A1PRRE +1
O A1PRRED$ +1
*-----

```

16.9.5. Compilato

```
> CRTBNDRPG PGM(CORSO/CNA1PRT1) SRCFILE(CORSO/CORSO) DBGVIEW(*LIST) TGTRLS(*CURRENT)
```

16.9.6. E chiamato

> CALL PGM(CNA1PRT1)

16.9.7. Genera la seguente stampa.

Codice	Descrizione	Pr	Descrizione	Pr	Descrizione
person	persona	na	provincia nascita	re	provincia residenza
-----	-----	--	-----	--	-----
AMEN	Alleluia	MI	Milano	RM	????????????????
BLU	Barba Blu	CO	Como	GE	Genova
JACK	Jack lo squartatore	GE	Genova	CO	Como

17. Definizione di tabelle e indici con DDL (SQL)

- 17.1. Il primo approccio di un quattrecentista alle DDL (Data Description Language) è di solito per sostituzione della forma DDS del database. Seguo qui questo schema adatto a chi maneggia un database fatto in DDS e non lo vuol riscrivere tutto ma vuole aggiungere file nuovi alla nuova maniera.
- 17.2. Occorre essere consapevoli che nella forma DDL:
- 17.2.1. Non sono state implementate alcune keyword che controllavano valori ed escursioni di valori.
 - 17.2.2. Non sono previsti file logici multifisici.
 - 17.2.3. È molto macchinoso l'uso dei membri.
- 17.3. Esempifico la creazione di specifiche DDL CNW* equivalenti alle DDS CNU* di simile nome che ho descritto in precedenza.
- 17.3.1. Tabella testata CNW1 che corrisponde al file fisico testata CNU1;
 - 17.3.2. Indice testata CNW1 che corrisponde al file logico testata CNU1;
 - 17.3.3. Tabella righe CNW2 che corrisponde al file fisico righe CNU2;
 - 17.3.4. Indice righe CNW2 che corrisponde al file logico righe CNU2.
 - 17.3.5. NOTA: il file logico multifisico CNUX non è riproducibile con le DDL.
 - 17.3.6. La scrittura dei sorgenti DDL può essere eseguita ex novo oppure ci si può aiutare con il comando JRTVDDL (Retrieve DDL) che può restituire il sorgente DDL del prototipo creato con DDS.
 - 17.3.7. L'esecuzione delle DDL scritte nei nuovi sorgenti avviene usando il comando RUNSQLSTM (Run SQL Statements).
- 17.4. Sorgente tabella testata
- 17.4.1. Sorgente ricavato con il comando seguente e poi aggiustato.

```
> JRTVDDL OBJNAME(CNU1) OBJLIB(CORSO) SRCFILE(CORSO) SRCLIB(CORSO) SRCMBR(CNW1)
SEVLVL(10)
Generate DDL
```

```
DROP TABLE CNU1 ;

CREATE OR REPLACE TABLE CNU1 (
-- SQL150B 10 REUSEDLT(*NO) nella tabella CNU1 in CORSO ignorata.
  N1DOC NUMERIC(5, 0) NOT NULL DEFAULT 0 ,
  N1TIP CHAR(3) CCSID 280 NOT NULL DEFAULT '' )

RCDFMT N1 ;

LABEL ON TABLE CNU1
  IS 'Testata documento. Fisico' ;

LABEL ON COLUMN CNU1
( N1DOC IS 'Numero          doc' ,
  N1TIP IS 'Tip            doc' ) ;

LABEL ON COLUMN CNU1
( N1DOC TEXT IS 'Numero documento' ,
  N1TIP TEXT IS 'Tipo documento' ) ;

GRANT ALTER , DELETE , INDEX , INSERT , REFERENCES , SELECT , UPDATE
ON CNU1 TO NERONI WITH GRANT OPTION ;

GRANT DELETE , INSERT , SELECT , UPDATE
ON CNU1 TO PUBLIC ;
```

- 17.4.2. Sorgente scritto ex novo o ricavato da quello recuperato dal file prototipo cambiando la parola giallata CNU1 in CNW1 e cancellando le righe violettate.

```
DROP TABLE CNW1 ;

CREATE OR REPLACE TABLE CNW1 (
  N1DOC NUMERIC(5, 0) NOT NULL DEFAULT 0 ,
  N1TIP CHAR(3) CCSID 280 NOT NULL DEFAULT '' )

RCDFMT N1 ;

LABEL ON TABLE CNW1
  IS 'Testata documento. Fisico' ;

LABEL ON COLUMN CNW1
( N1DOC IS 'Numero          doc' ,
  N1TIP IS 'Tip            doc' ) ;

LABEL ON COLUMN CNW1
( N1DOC TEXT IS 'Numero documento' ,
  N1TIP TEXT IS 'Tipo documento' ) ;
```

17.5. Creazione tabella testata

17.5.1. Sull'elenco membri sorgenti del PDM si usi l'opzione SQ

```
Option Command
SQ ? RUNSQLSTM SRCFILE(&L/&F) SRCMBR(&N) COMMIT(*NONE) ERRVL(20)
DFTRDBCOL(&L)
```

oppure, sul video comandi, il seguente comando:

```
> RUNSQLSTM SRCFILE(CORSO/CORSO) SRCMBR(CNW1) COMMIT(*NONE) ERRVL(20)
DFTRDBCOL(CORSO)
```

Si osservi il parametro **giallato** DFTRDBCOL (Default collection) che decide in che libreria verrà creato l'oggetto.

17.5.2. È possibile consultare l'esito del comando SQL precedente che, in modo disomogeneo rispetto ai comandi in stile As400, sputa una stampa ma zero messaggi in log.

```
5770SS1 V7R2M0 140418      Esecuzione istruzioni SQL      CNW1      22/02/21 19:13:44      PAGE      1
File origine.....CORSO/CORSO
Membro.....CNW1
Commit.....*NONE
Denominazione.....*SYS
Livello generazione.....20
Formato data.....*JOB
Separatore data.....*JOB
Formato ora.....*HMS
Separatore ora.....*JOB
Margine destro.....80
Raccolta predefinita.....CORSO
Indicatore IBM SQL.....*NOFLAG
Indicatore ANS.....*NONE
Virgola decimale.....*JOB
Sequenza di ordinamento...*JOB
ID lingua.....*JOB
File di stampa.....*LIBL/QSYSPRT
CCSID file origine.....280
CCSID lavoro.....280
Elaborazione istruzioni...*RUN
Permessi la copia dati....*OPTIMIZE
Consentito il blocco.....*ALLREAD
Regole SQL.....*DB2
Opzioni risultati decimali:
  Precisione massima.....31
  Scala massima.....31
  Scala divisione minima....0
Accesso contemporaneo
  risoluzione.....*DFT
Membro origine modificato in 22/02/21 19:13:25
5770SS1 V7R2M0 140418      Esecuzione istruzioni SQL      CNW1      22/02/21 19:13:44      PAGE      2
Record *...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8      SEQNBR      Ultima modifica
1 DROP TABLE CNW1 ;
2
3 CREATE OR REPLACE TABLE CNW1 (
4   N1DOC NUMERIC(5, 0) NOT NULL DEFAULT 0 ,
5   N1TIP CHAR(3) CCSID 280 NOT NULL DEFAULT ' ' )
6
7   RCDFMT N1      ;
8
9 LABEL ON TABLE CNW1
10  IS 'Testata documento. Fisico' ;
11
12 LABEL ON COLUMN CNW1
13  ( N1DOC IS 'Numero      doc' ,
14   N1TIP IS 'Tip      doc' ) ;
15
16 LABEL ON COLUMN CNW1
17  ( N1DOC TEXT IS 'Numero documento' ,
18   N1TIP TEXT IS 'Tipo documento' ) ;
***** FINE ORIGINI *****
5770SS1 V7R2M0 140418      Esecuzione istruzioni SQL      CNW1      22/02/21 19:13:44      PAGE      3
Record *...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8      SEQNBR      Ultima modifica
ID MES  SEV  RECORD  TESTO
SQL0204  20      1  Posizione 1 CNW1 in CORSO di tipo *FILE non trovato.
SQL7905  20      3  Posizione 1 La tabella CNW1 in CORSO è stata creata ma
non registrata su giornale.
SQL7965   0      9  Posizione 1 LABEL ON per CNW1 in CORSO completato.
SQL7965   0     12  Posizione 1 LABEL ON per CNW1 in CORSO completato.
SQL7965   0     16  Posizione 1 LABEL ON per CNW1 in CORSO completato.
Riepilogo messaggi
Totale Informativo Avvertenza Errore Grave Terminale
5          3          0          2          0          0
Trovati nell'origine 20 errori gravi
```

17.6. Sorgente indice testata

17.6.1. Stessa commedia per il logico. L'API va indirizzata al tipo di recupero da eseguire tramite il parametro **OBJTYPE (INDEX)**.

```
> JRTVDDL OBJNAME(CNU11) OBJLIB(CORSO) OBJTYPE (INDEX) SRCFILE(CORSO) SRCLIB(CORSO)
SRCMBR(CNW11) SEVLVL(10)
Generate DDL
```

Per ricavare e aggiustare il sorgente dell'indice sulla testata.

```

DROP INDEX CNW11 ;

CREATE UNIQUE INDEX CNW11
  ON CNW1 ( N1DOC ASC )
  ;

LABEL ON INDEX CNW11
  IS 'Testata documento. Logico per numero documento' ;

```

17.7. Creazione indice testata

17.7.1. Sull'elenco membri sorgenti del PDM si usi l'opzione SQ oppure, sul video comandi, il seguente comando:

```

> RUNSQLSTM SRCFILE(CORSO/CORSO) SRCMBR(CNW11) COMMIT(*NONE) ERRlvl(20)
DFTRDBCOL(CORSO)

```

17.7.2. Anche qui è possibile consultare l'esito del comando SQL precedente tramite la stampa generata che non si riporta.

17.8. Sorgente tabella riga e creazione tabella riga

17.8.1. Come per la testata, ma si riporta solo il sorgente.

```

DROP TABLE CNW2 ;

CREATE OR REPLACE TABLE CNW2 (
  N2DOC NUMERIC(5, 0) NOT NULL DEFAULT 0 ,
  N2RIG NUMERIC(3, 0) NOT NULL DEFAULT 0 ,
  N2TXT CHAR(50) CCSID 280 NOT NULL DEFAULT '' )

  RCDFMT N2          ;

LABEL ON TABLE CNW2
  IS 'Riga documento' ;

LABEL ON COLUMN CNW2
( N2DOC IS 'Numero          documento' ,
  N2RIG IS 'Numero          riga' ,
  N2TXT IS 'Testo           riga' ) ;

LABEL ON COLUMN CNW2
( N2DOC TEXT IS 'Numero documento' ,
  N2RIG TEXT IS 'Numero riga' ,
  N2TXT TEXT IS 'Testo riga' ) ;

```

17.9. Sorgente indice riga e creazione indice riga

17.9.1. Come per la testata, ma si riporta solo il sorgente.

```

DROP INDEX CNW21 ;

CREATE UNIQUE INDEX CNW21
  ON CNW2 ( N2DOC ASC , N2RIG ASC )
  ;

LABEL ON INDEX CNW21
  IS 'Riga documento. Logico per numero documento e riga' ;

```

18. Scaricare dati sotto forma di sorgente

18.1. Tool JTOSQL (Unload database to SQL).

- 18.1.1. Se occorre scaricare dati da un file in modo che si possano trasportare dentro un sorgente, si può usare il comando JTOSQL (Unload database to SQL) che si esemplifica su testata e righe.

```
> JTOSQL FROMFILE(CORSO/CNU1) TOFILE(CORSO/CNU1) TOSRCMBR(CNU1SQL)
TOSRCFILE(CORSO/CORSO) SRCLEN(112)
File CORSO/CNU1 attr PF tipo *DATA SQL in CORSO/CORSO
> /*SqlStream*/STRSEU SRCFILE(CORSO/CORSO) SRCMBR(CNU1SQL) OPTION(5)
> JTOSQL FROMFILE(CORSO/CNU2) TOFILE(CORSO/CNU2) TOSRCMBR(CNU2SQL)
TOSRCFILE(CORSO/CORSO) SRCLEN(112)
File CORSO/CNU2 attr PF tipo *DATA SQL in CORSO/CORSO
> /*SqlStream*/STRSEU SRCFILE(CORSO/CORSO) SRCMBR(CNU2SQL) OPTION(5)
```

- 18.1.2. Si ottengono i due sorgenti contenenti istruzioni sql INSERT TO che fotografano i valori dei record presenti nei file. Si noti che lo scarico avviene nell'ordine d'arrivo in cui i record figurano nel file fisico.

```
INSERT INTO CORSO/CNU1
(N1DOC, N1TIP)
VALUES
(1, 'FAT');
INSERT INTO CORSO/CNU1 VALUES
(2, 'NOT');
INSERT INTO CORSO/CNU1 VALUES
(3, 'VUO');
INSERT INTO CORSO/CNU1 VALUES
(5, 'VEN');
```

```
INSERT INTO CORSO/CNU2
(N2DOC, N2RIG, N2TXT)
VALUES
(1, 10, 'PRIMA riga');
INSERT INTO CORSO/CNU2 VALUES
(1, 20, 'seconda RIGA');
INSERT INTO CORSO/CNU2 VALUES
(2, 1, 'Tanto');
INSERT INTO CORSO/CNU2 VALUES
(2, 2, 'va la');
INSERT INTO CORSO/CNU2 VALUES
(2, 3, 'gatta');
INSERT INTO CORSO/CNU2 VALUES
(2, 4, 'al lardo');
INSERT INTO CORSO/CNU2 VALUES
(4, 1, 'Riga senza testata');
INSERT INTO CORSO/CNU2 VALUES
(5, 10, 'Nel');
INSERT INTO CORSO/CNU2 VALUES
(5, 20, 'mezzo');
INSERT INTO CORSO/CNU2 VALUES
(5, 30, 'del');
INSERT INTO CORSO/CNU2 VALUES
(5, 40, 'cammin');
INSERT INTO CORSO/CNU2 VALUES
(5, 50, 'di');
INSERT INTO CORSO/CNU2 VALUES
(5, 60, 'nostra');
INSERT INTO CORSO/CNU2 VALUES
(5, 70, 'vita');
INSERT INTO CORSO/CNU2 VALUES
(5, 80, 'mi');
INSERT INTO CORSO/CNU2 VALUES
(5, 90, 'ritrovai');
INSERT INTO CORSO/CNU2 VALUES
(5, 100, 'per');
INSERT INTO CORSO/CNU2 VALUES
(5, 110, 'una');
INSERT INTO CORSO/CNU2 VALUES
(5, 120, 'selva');
INSERT INTO CORSO/CNU2 VALUES
(5, 130, 'oscura');
INSERT INTO CORSO/CNU2 VALUES
(5, 140, 'che');
INSERT INTO CORSO/CNU2 VALUES
(5, 150, 'la');
INSERT INTO CORSO/CNU2 VALUES
(5, 160, 'diritta');
INSERT INTO CORSO/CNU2 VALUES
(5, 170, 'via');
INSERT INTO CORSO/CNU2 VALUES
(5, 180, 'era');
```



```

INSERT INTO CORSO/CNU2 VALUES
(5, 190, 'smarrita');
INSERT INTO CORSO/CNU2 VALUES
(5, 200, 'Ahi');
INSERT INTO CORSO/CNU2 VALUES
(5, 210, 'quanto');
INSERT INTO CORSO/CNU2 VALUES
(5, 220, 'a');
INSERT INTO CORSO/CNU2 VALUES
(5, 230, 'dir ...');
INSERT INTO CORSO/CNU2 VALUES
(2, 5, 'che ci lascia lo zampino.');
```

- 18.1.3. I due sorgenti eseguiti con RUNSQLSTM, permettono di ricaricare i dati nel file originali ma vuoti.
- ```

> RUNSQLSTM SRCFILE(CORSO/CORSO) SRCMBR(CNU1SQL) COMMIT(*NONE) ERRlvl(20)
DFTRDBCOL(CORSO)
> RUNSQLSTM SRCFILE(CORSO/CORSO) SRCMBR(CNU2SQL) COMMIT(*NONE) ERRlvl(20)
DFTRDBCOL(CORSO)
```
- 18.1.4. Inoltre, cambiando, ad esempio, nel primo sorgente, la costante CORSO/CNU1 in NUOVALIBRERIA/NUOVOFILE si può riempire un altro file che abbia lo stesso tracciato.
- 18.1.5. Si osserva che il comando RUNSQLSTM dotato del parametro DFTRDBCOL(CORSO) permette eventualmente di non menzionare la libreria CORSO nel sorgente.

```

INSERT INTO CNU1
(NIDOC, N1TIP)
VALUES
(1, 'FAT');
INSERT INTO CNU1 VALUES
(2, 'NOT');
INSERT INTO CNU1 VALUES
(3, 'VUO');
INSERT INTO CNU1 VALUES
(5, 'VEN');
```

## 19. Processo di stampa Testata-Righe con CLLE, RPGLE e DSPF

- 19.1. Usando i file testata e riga visti, costruire un processo di stampa usando CLLE, RPGLE posizionale e DDS per file video.
- 19.2. Scrivere il programma RPGLE CNUPRT1 che stampa in un unico file di spool tutti i documenti contenuti nei due file dati usando il logico multifisico.
  - 19.2.1. Sorgenti commentati al massimo livello ragionevole per favorirne la comprensione.
  - 19.2.2. Utilizzo delle specifiche F, I, C, O.
  - 19.2.3. Creazione del programma eseguibile con CRTBNDRPG.
  - 19.2.4. Esecuzione del programma ottenuto.
  - 19.2.5. Vedi il sorgente:  
[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT1.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT1.txt)
- 19.3. Scrivere il programma RPGLE CNUPRT2 variante della versione 1
  - 19.3.1. Usare input primario e secondario in match e rotture di livello.
  - 19.3.2. Usare un paio di BIF (Built In Function): %trim che contorna un campo dai blank e %editc che edita un campo numerico.
  - 19.3.3. Compilare e provare.
  - 19.3.4. Vedi il sorgente:  
[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT2.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT2.txt)
- 19.4. Scrivere il programma RPGLE CNUPRT3 variante della versione 2
  - 19.4.1. Abbandonare il ciclo RPG ed usare i file in modo "full procedural" e usare parametri in entrata e uscita per stampare un solo documento. Usare istruzioni PARM, KLIST e KFLD, CHAIN, SETLL, READE, DOW, LEAVE, ENDDO, EXSR, BEGSR, ENDSR.
  - 19.4.2. Vedi il sorgente:  
[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT3.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT3.txt)
- 19.5. Scrivere il programma CLLE CNUPRT3C per richiamare CNUPRT3 più volte.
  - 19.5.1. Usare istruzioni DOWHILE, ENDDO, LEAVE, RCLRSC, OVRPRTF.
  - 19.5.2. Vedi il sorgente:  
[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT3C.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT3C.txt)
  - 19.5.3. Creare il File video CNUPRT4W per immettere interattivamente il parametro da passare al programma CNUPRT3.
  - 19.5.4. Scrivere le DDS ma gestirle oltre che con il SEU anche con il tool SDA (Screen Design Aids).
  - 19.5.5. Per creare il video eseguibile usare il comando CRTDSPF.
  - 19.5.6. Vedi il sorgente:  
[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT4W.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT4W.txt)
- 19.6. Creare il programma CLLE CNUPRT4 utilizzando il file video CNUPRT4W e richiamando il programma di stampa CNUPRT3 a richiesta dell'utente che opera sul video.
  - 19.6.1. Per creare il programma eseguibile usare il comando CRTBNDCL.
  - 19.6.2. Vedi il sorgente:  
[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT4.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUPRT4.txt)

## 20. Processo di interrogazione Testata-Righe RPGLE e DSPF

- 20.1. Usando i file testata e riga visti, costruire un processo di interrogazione interattiva con RPGLE posizionale e DDS per file video.
- 20.2. Scrivere sorgente video CNUDSPW per visualizzare i documenti contenuti nei due file dati.
  - 20.2.1. Creare sorgente di tipo DSPF vuoto tramite il SEU o tramite ADDPFM.
  - 20.2.2. Gestire sorgente tramite l'opzione 17 del PDM che chiama il comando STRSDA.
  - 20.2.3. Comporre il video interattivamente tramite SDA.
  - 20.2.4. Esaminare con SEU il sorgente generato dall'SDA.
  - 20.2.5. Il sorgente conterrà la definizione dei seguenti formati:
    - W0 Guida per ricerca documento
    - S1 Righe documento (Subfile o Elenco)
    - C1 Testata documento (Controllo elenco)
    - B1 Annuncio di documento senza righe (Bianco, alternativo a S1)
    - P1 Piede documento
  - 20.2.6. Compilare il sorgente con CRTDSPF.
  - 20.2.7. Vedi il sorgente:  
[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUDSPW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUDSPW.txt)
- 20.3. Scrivere il sorgente RPGLE CNUDSP che userà il file video CNUDSPW per visualizzare i documenti contenuti nei due file dati.
  - 20.3.1. Il programma utilizza gli stessi file logici CNU11 e CNU21 usati nella stampa del singolo documento.
  - 20.3.2. Dichiarare il file video CNUDSPW che contiene un subfile controllato tramite il relative record number "RRN".
  - 20.3.3. Usare WRITE, READ ed EXFMT sui formati video e gestire il subfile.
  - 20.3.4. Compilare il programma con CRTBNDRPG.
  - 20.3.5. Vedi il sorgente:  
[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUDSP.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNUDSP.txt)
- 20.4. Eseguire la chiamata del programma e valutarne il funzionamento.

## 21. Processo di gestione Anagrafica-Tabelle

- 21.1. Costruzione di programmi vari su Anagrafica-Tabelle.
- 21.2. Contiene una serie di edizioni progressive di uno stesso programma in modo da valutare
  - 21.2.1. l'opportunità di adottare una struttura in grado di crescere senza sconvolgimenti
  - 21.2.2. ma soprattutto la tecnica di sviluppo per gradi di uno schema risolutivo.
- 21.3. Essendo forniti i sorgenti di tutte le versioni, è possibile comparare due edizioni contigue nello sviluppo di video o di programmi tramite il comando **CMPPFM** (Compare Physical File Member).
- 21.4. In questo paragrafo il colore di fondo dei sorgenti riportati ha i seguenti significati. <

### 21.4.1. Display file

```
Riga sorgente.
Riga sorgente modificata o aggiunta rispetto alla versione precedente.
```

### 21.4.2. Rpg Ile o Cl Ile

```
Riga sorgente.
Riga sorgente modificata o aggiunta rispetto alla versione precedente.
```

## 21.5. Visualizzazione anagrafica con video dati ma senza video guida (oggetti CNA1A\*).

- 21.5.1. Per costruirlo con il tool di sistema SDA, si veda il capitolo che ne spiega il maneggio elementare.
- 21.5.2. SDA e SEU possono essere usati indifferentemente per comporre il display file.
- 21.5.3. L'SDA è molto utile per creare la prima versione dei record a video.
- 21.5.4. Il SEU risulta più veloce nelle piccole modifiche.
- 21.5.5. Sorgente commentato del video CNA1AW Dspf. Scritto ex novo.

```
* Claudio Neroni 25-02-2021 Creato per Corso.
* CNA1AW
* Anagrafica persone dsp no guida
```

### 21.5.5.1. Innanzitutto scelgo una misura del video tra le due permesse 24x80 o 27x132.

```
A DSPSI2(24 80 *DS3)
```

### 21.5.5.2. Assegno un nome al record video e chiedo il lampeggiamento del cursore.

```
A R W1
A TEXT('Video Dati')
A BLINK
```

### 21.5.5.3. A livello di record permetto i comandi citati nel successivo promemoria tramite le apposite parole chiave.

```
A CA03(03 'Fine')
A PAGEUP(25 'Precedente')
A PAGEDOWN(26 'Successivo')
```

### 21.5.5.4. Definisco una costante sottolineata e in alta luminosità come titolo e la posiziono in alto sul video.

```
A 1 32'Anagrafica persona'
A DSPATR(UL)
A DSPATR(HI)
```

### 21.5.5.5. Scelgo i quattro campi dell'anagrafica e li incolonna nelle prime righe del video.

### 21.5.5.6. A sinistra di ogni campo aggiusto una costante ricavata dalla sua intestazione di colonna.

```
A 4 10'Codice persona:'
A W1CDPE R O 4 27REFFLD(A1CDPE CNA1)
A DSPATR(HI)
A 5 5'Descrizione persona:'
A W1DEPE R O 5 27REFFLD(A1DEPE CNA1)
A DSPATR(HI)
A 6 7'Provincia nascita:'
A W1PRNA R O 6 27REFFLD(A1PRNA CNA1)
A DSPATR(HI)
A 7 5'Provincia residenza:'
A W1PRRE R O 7 27REFFLD(A1PRRE CNA1)
A DSPATR(HI)
```

### 21.5.5.7. Sulla penultima riga scrivo un promemoria dei comandi che voglio usare quando uso il video.

```
A 23 2'F3=Fine'
A +2'PagUp=Prec'
A +2'PagDw=Succ'
```

### 21.5.5.8. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1AW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1AW.txt)

### 21.5.6. Sorgente commentato del programma CNA1AD Rpgle.

```
* Claudio Neroni 25-02-2021 Creato per Corso.
* CNA1AD
* Anagrafica persone dsp no guida
*-----
```

21.5.6.1. Anagrafico persone per codice con le caratteristiche: input, full procedural, external described, keyed, disk

```
* Anagrafica persone logico codice
FCNA11 if e k disk
```

21.5.6.2. Video del programma con le caratteristiche: combined, full procedural, external described, workstation.

```
* Anagrafica persone no guida
FCNA1AW cf e workstn
*-----
```

21.5.6.3. Settaggi iniziali commentati.

```
* Predispone chiusura.
C seton lr
* Si posiziona ad inizio file.
C *loval setll A1
* Predispone lettura in avanti simulando PageDown.
C eval *in26=*on
```

21.5.6.4. Loop intorno al video. Inizio.

```
* Cicla intorno a lettura e video.
C do *hival
```

21.5.6.5. Richiesta del record successivo.

```
* Se richiesto PageDown.
C if *in26
* Legge in avanti.
C read A1
* Se non trova record.
C if %eof
* Governa file vuoto.
C exsr EMPTY
* Si riposiziona ad inizio file.
C *loval setll A1
* Ricicla.
C iter
* Se non trova record.
C endif
* Azzerà contatore letture fallite.
C clear fallite 3 0
* Se richiesto PageDown.
C endif
```

21.5.6.6. Richiesta del record precedente

```
* Se richiesto PageUp.
C if *in25
* Legge all'indietro.
C readp A1
* Se non trova record.
C if %eof
* Governa file vuoto.
C exsr EMPTY
* Si riposiziona a fine file.
C *hival setgt A1
* Ricicla.
C iter
* Se non trova record.
C endif
* Azzerà contatore letture fallite.
C clear fallite 3 0
* Se richiesto PageUp.
C endif
```

21.5.6.7. Trascrive record su video e lo emette.

```
* Trascrive dati dal record di database al video.
C exsr DBW1
* Emette video dati.
C exfmt W1
```

21.5.6.8. Comandi richiesti dal video.

```
* Se richiesta fine, abbandona.
C 03 leave
* Se richiesto PageDown, ricicla.
C 26 iter
* Se richiesto PageUp, ricicla.
C 25 iter
```

21.5.6.9. Loop intorno al video. Fine.

```
* Cicla intorno a lettura e video.
C enddo
*-----
```

21.5.6.10. Subroutine di trascrizione campi a video.

```
* Trascrive dati dal record di database al video.
```

```
C DBW1 begsr
C clear W1
C eval W1CDPE=A1CDPE
C eval W1DEPE=A1DEPE
C eval W1PRNA=A1PRNA
C eval W1PRRE=A1PRRE
C endsr
```

```
*-----
```

21.5.6.11. Subroutine che conta le letture fallite per uscire in caso di file vuoto.

```
* Governa file vuoto.
```

```
C EMPTY begsr
* Conta letture fallite.
C eval fallite+=1
* Se il contatore letture fallite supera 1.
C if fallite > 1
* Pulisce video.
C clear W1
* Usa un campo descrittivo per segnalare.
C eval W1DEPE=x'2B'+ 'File vuoto'+x'20'
* Emette video dati per segnalare.
C exfmt W1
* Ritorna.
C return
* Se il contatore letture fallite supera 1.
C endif
C endsr
```

```
*-----
```

21.5.6.12. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1AD.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1AD.txt)

21.5.7. Funzionamento del programma CNA1AD.

21.5.7.1. Entra visualizzando il primo record del file.

21.5.7.2. Su ogni record visualizzato permette di abbandonare oppure di passare a visualizzare il successivo o il precedente.

21.5.7.3. Quando incontra inizio o fine file si comporta come se il file fosse circolare (dopo l'ultimo record si trova ancora il primo e prima del primo si trova l'ultimo).

21.5.7.4. In qualunque momento, se il file è vuoto, presenta un record video con i dati in bianco salvo il campo descrittivo in cui inserisce un avviso di vuoto prima di andare a fine.

21.5.8. Tool JDUBDB di duplicazione del database (fisico-logici).

21.5.8.1. Il tool JDUPDB prende in considerazione un file fisico prototipo sul quale si basano alcuni logici monofisici con i nomi costruiti a partire dallo stesso suffisso del fisico. Genera quindi un fisico con un eventuale nuovo prefisso e duplica tutti i logici monofisici del prototipo basando i nuovi file sul fisico duplicato e nominandoli con l'eventuale nuovo prefisso del fisico.

```
> JDUPDB BASPF(CNA1) TOLIB(CORSO) FROMLIB(CORSO) PFXBFR(CNA1) PFXAFT(CNB1)
Nome libreria temporanea JTMP082886
```

```
JDUPDB BASPF(CNA1) TOLIB(CORSO) FROMLIB(CORSO) PFXBFR(CNA1) PFXAFT(CNB1)
DETAIL(*MAX) REFPGM(JDUPDBC)
```

```
Duplicato logico CORSO/CNA11 in JTMP082886/CNB11
```

```
Fisico CORSO/CNA1 duplicato in JTMP082886/CNB1 . Duplicati 1 logici su 1.
```

```
Nome invariato per 0 file
```

```
Mossi 2 oggetti da libreria JTMP082886 a libreria CORSO
```

```
Duplicato fisico CORSO/CNA1 in CORSO/CNB1 con 1 logici su 1
```

21.5.8.2. La creazione dei duplicati fisico-logici costituisce un nuovo database sul quale possono essere reindirizzati gli stessi programmi che lavorano sul database prototipo.

21.5.9. Programma CNA1AC Cile di cappello con reindirizzamento ad altro database.

21.5.9.1. A scopo esemplificativo si scrive un Cile "di cappello" che si occupa di reindirizzare il programma di visualizzazione a dei duplicati del database conosciuto dalla compilazione.

```
/* Claudio Neroni 25-02-2021 Creato per Corso. */
/* CNA1AC */
/* Anagrafica persone dsp no guida */
/*
PGM PARM(&DB)
/* Riceve Database. */
DCL VAR(&DB) TYPE(*CHAR) LEN(1)
/* Sceglie database. */
SELECT
/* Se manca parametro database, */
/* indirizza il primo file. */
WHEN COND(%ADDR(&DB) *EQ *NULL) THEN(DO)
OVRDBF FILE(CNA11) TOFILE(CNA11)
ENDDO
```

```

/* Se database "A", */
/* indirizza il primo file. */
 WHEN COND(&DB *EQ A) THEN(DO)
 OVRDBF FILE(CNA11) TOFILE(CNA11)
 ENDDO
/* Se database "B", */
/* indirizza il secondo file. */
 WHEN COND(&DB *EQ B) THEN(DO)
 OVRDBF FILE(CNA11) TOFILE(CNB11)
 ENDDO
/* Se database "C", */
/* indirizza il terzo file. */
 WHEN COND(&DB *EQ C) THEN(DO)
 OVRDBF FILE(CNA11) TOFILE(CNC11)
 ENDDO
/* Altrimenti, */
/* segnala e ritorna. */
 OTHERWISE CMD(DO)
 SNDPGMMSG MSG('Scegli A, B o C!')
 RETURN
 ENDDO
/* Sceglie database. */
 ENDSELECT
/* Chiama la visualizzazione. */
 CALL PGM(CNA1AD)
 MONMSG MSGID(CEE0000) EXEC(DO)
 SNDPGMMSG MSG('Database "' *TCAT &DB *TCAT '"' non +
 esiste!')
 ENDDO
/* Elimina il reindirizzamento e trascura eventuale errore. */
 DLTOVR FILE(CNA11)
 MONMSG MSGID(CPF0000)
 ENDPGM

```

21.5.9.2. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1AC.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1AC.txt)

## 21.6. Visualizzazione con video guida di posizionamento e video dati (oggetti CNA1B\*).

21.6.1. Sorgente commentato del video CNA1BW Dspf. Ottenuto modificando la versione CNA1AW.

```

* Claudio Neroni 26-02-2021 Creato per Corso.
* CNA1BD
* Anagrafica persone dsp guida

```

21.6.1.1. Per poter controllare meglio il video e non dover ritoccare caratteristiche ad ogni versione, a partire da questa, utilizzo per i campi tutte le caratteristiche che mi serviranno in tutte le successive versioni del programma. Trasformo quindi i campi dati in tipo "B" (Input/Output) anche se mi servirà solo in modifica, più avanti. Per ora questo mi permette la clear del video dati.

```

A DSPSIZ(24 80 *DS3)

```

21.6.1.2. Aggiungo CHGINPDFT per eliminare il default di sottolineato e alta luminosità sui campi di input.

```

A CHGINPDFT

```

21.6.1.3. Aggiungo INDARA che toglie dai buffer del video gli indicatori.

```

A INDARA

```

21.6.1.4. Aggiungo INDTXT che applica un testo esplicativo ad un indicatore.

```

A INDTXT(44 'Protezione campi')

```

21.6.1.5. Aggiungo il video guida con il solo campo di ricerca codice persona.

```

A R W0
A TEXT('Video Guida')
A BLINK
A CA03(03 'Fine')
A PAGEUP(25 'Precedente')
A PAGEDOWN(26 'Successivo')
A 1 32'Anagrafica persona'
A DSPATR(UL)
A DSPATR(HI)
A 2 38'Guida'
A DSPATR(HI)
A 4 10'Codice persona:'
A W0CDPE R B 4 27REFFLD(AlCDPE CNA1)
A DSPATR(UL)
A DSPATR(HI)
A 23 2'F3=Fine'
A 23 11'PagDw=Succ'
A 23 23'PagUp=Prec'
A R W1
A TEXT('Video Dati')
A BLINK
A CA03(03 'Fine')

```

21.6.1.6. Aggiungo il comando F12=Guida al video Dati.

|   |                           |
|---|---------------------------|
| A | CA12(12 'Guida')          |
| A | PAGEUP(25 'Precedente')   |
| A | PAGEDOWN(26 'Successivo') |
| A | 1 32'Anagrafica persona'  |
| A | DSPATR(UL)                |
| A | DSPATR(HI)                |
| A | 2 39'Dati'                |
| A | DSPATR(HI)                |

21.6.1.7. Trasformo i campi di Input in Both (Input-Output) per usare nei prossimi sviluppi un solo video in visualizzazione o aggiunta-modifica.

21.6.1.8. Aggiungo a tutti i campi l'indicatore 44 di commutazione modificabile=sottolineato / non\_modificabile=protetto.

|       |                                    |
|-------|------------------------------------|
| A     | 4 10'Codice persona:'              |
| A     | W1CDPE R B 4 27REFFLD(A1CDPE CNA1) |
| A 44  | DSPATR(PR)                         |
| A N44 | DSPATR(UL)                         |
| A     | DSPATR(HI)                         |
| A     | 5 5'Descrizione persona:'          |
| A     | W1DEPE R B 5 27REFFLD(A1DEPE CNA1) |
| A 44  | DSPATR(PR)                         |
| A N44 | DSPATR(UL)                         |
| A     | DSPATR(HI)                         |
| A     | 6 7'Provincia nascita:'            |
| A     | W1PRNA R B 6 27REFFLD(A1PRNA CNA1) |
| A 44  | DSPATR(PR)                         |
| A N44 | DSPATR(UL)                         |
| A     | DSPATR(HI)                         |
| A     | 7 5'Provincia residenza:'          |
| A     | W1PRRE R B 7 27REFFLD(A1PRRE CNA1) |
| A 44  | DSPATR(PR)                         |
| A N44 | DSPATR(UL)                         |
| A     | DSPATR(HI)                         |
| A     | 23 2'F3=Fine'                      |

21.6.1.9. Aggiungo la costante descrittiva del comando F12=Guida .

|   |                |
|---|----------------|
| A | +2'F12=Guida'  |
| A | +2'PagDw=Succ' |
| A | +2'PagUp=Prec' |

21.6.1.10. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1BW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1BW.txt)

21.6.2. Sorgente commentato del programma CNA1BD Rpgle. Riscritto integralmente rispetto alla versione CNA1AD.

```
* Claudio Neroni 26-02-2021 Creato per Corso.
* CNA1BD
* Anagrafica persone dsp guida
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
* Anagrafica persone dsp guida
FCNA1BW cf e workstn
*-----
```

21.6.2.1. Usando le subroutine sia per il video guida che per il video dati, la mainroutine si riduce a poche righe.

```
* Predispone chiusura.
C seton lr
* Predispone protezione campi.
C seton 44
* Gestisce video guida.
C exsr SRW0
*-----
```

21.6.2.2. Tutte le routine che gestiscono un video hanno un loop (infinito ma con uscite a richiesta) intorno a quel video

```
* Gestisce video guida.
C SRW0 begsr
* Pulisce video guida.
C clear W0
```

21.6.2.3. Loop intorno al video guida. Inizio.

```
* Cicla intorno al video guida.
C do *hival
* Emette video guida.
C exfmt W0
```

21.6.2.4. Subito dopo la scrittura e riletture del video, si testa innanzitutto la richiesta di uscita.

```
* Se richiesta fine, abbandona.
```



|   |    |       |
|---|----|-------|
| C | 03 | leave |
|---|----|-------|

21.6.2.5. Gestisce la richiesta di record precedente al codice, digitato o no.

|                                                     |         |          |
|-----------------------------------------------------|---------|----------|
| * Se richiesto PagUp.                               |         |          |
| C                                                   | if      | *in25    |
| * Si posiziona prima della chiave di ricerca.       |         |          |
| C                                                   | WOCDPPE | setll A1 |
| * Legge all'indietro.                               |         |          |
| C                                                   | readp   | A1       |
| * Se non trova record, ritenta a partire dal fondo. |         |          |
| C                                                   | if      | %eof     |
| C                                                   | *hival  | setgt A1 |
| C                                                   | readp   | A1       |
| C                                                   | endif   |          |

21.6.2.6. Gestisce la richiesta di record successivo al codice, digitato o no. Si applica anche all'Invio, essendo quest'ultimo non impedibile.

|                                               |         |          |
|-----------------------------------------------|---------|----------|
| * Se richiesto PagUp o Enter,                 |         |          |
| C                                             | else    |          |
| * Si posiziona prima della chiave di ricerca. |         |          |
| C                                             | WOCDPPE | setll A1 |
| * Legge in avanti.                            |         |          |
| C                                             | read    | A1       |
| * Se richiesto...                             |         |          |
| C                                             | endif   |          |

21.6.2.7. Nel caso di lettura non possibile per mancanza di record, ripresenta il video guida.

|                                 |       |      |
|---------------------------------|-------|------|
| * Se non trova record, ricicla. |       |      |
| C                               | if    | %eof |
| C                               | iter  |      |
| C                               | endif |      |

21.6.2.8. Avendo acquisito un record, chiama la subroutine di gestione del video dati

|                        |      |      |
|------------------------|------|------|
| * Gestisce video dati. |      |      |
| C                      | exsr | SRW1 |

21.6.2.9. Al ritorno dal video dati, testa la richiesta di uscita proveniente da quel video.

|                                 |    |       |
|---------------------------------|----|-------|
| * Se richiesta fine, abbandona. |    |       |
| C                               | 03 | leave |

21.6.2.10. Loop intorno al video guida. Fine.

|                                 |       |  |
|---------------------------------|-------|--|
| * Cicla intorno al video guida. |       |  |
| C                               | enddo |  |
| C                               | endsr |  |
| *-----                          |       |  |

21.6.2.11. La gestione del video dati presume che un record dati sia già stato letto e pronto all'elaborazione

|                        |      |       |
|------------------------|------|-------|
| * Gestisce video dati. |      |       |
| C                      | SRW1 | begsr |

21.6.2.12. Loop intorno al video dati. Inizio.

|                                |    |        |
|--------------------------------|----|--------|
| * Cicla intorno al video dati. |    |        |
| C                              | do | *hival |

21.6.2.13. Anche la trascrizione dal record di database al record video viene chiamata come subroutine per togliere l'elenco dei campi dalla routine di gestione video.

|                                                   |       |      |
|---------------------------------------------------|-------|------|
| * Trascrive dati dal record di database al video. |       |      |
| C                                                 | exsr  | DBW1 |
| * Emette video dati.                              |       |      |
| C                                                 | exfmt | W1   |

21.6.2.14. Dopo l'emissione e la rilettura del video si testano gli abbandoni.

|                                  |    |       |
|----------------------------------|----|-------|
| * Se richiesta fine, abbandona.  |    |       |
| C                                | 03 | leave |
| * Se richiesta Guida, abbandona. |    |       |
| C                                | 12 | leave |

21.6.2.15. Testa la richiesta di vedere il record precedente preoccupandosi di posizionamento e lettura. Per il posizionamento usa l'ultimo codice letto dal file.

|                                                 |         |          |
|-------------------------------------------------|---------|----------|
| * Se richiesto PagUp.                           |         |          |
| C                                               | if      | *in25    |
| * Si posiziona prima della chiave ultima vista. |         |          |
| C                                               | WOCDPPE | setll A1 |
| * Legge all'indietro.                           |         |          |
| C                                               | readp   | A1       |

21.6.2.16. Testa la richiesta di vedere il record successivo preoccupandosi di posizionamento e lettura. Per il posizionamento usa l'ultimo codice letto dal file.

|                                              |         |          |
|----------------------------------------------|---------|----------|
| * Se richiesto PagDw o Enter,                |         |          |
| C                                            | else    |          |
| * Si posiziona oltre la chiave ultima vista. |         |          |
| C                                            | WOCDPPE | setgt A1 |

```

* Legge in avanti.
C read A1
* Se richiesto...
C endif

```

21.6.2.17. Se non ha trovato record, pulisce l'ultima chiave letta e abbandona per tornare al video guida..

```

* Se non trova record, abbandona.
C if %eof
C clear W0CDPE
C leave
C endif

```

21.6.2.18. Loop intorno al video dati. Fine.

```

* Cicla intorno al video dati.
C enddo
C endsr
*-----

```

21.6.2.19. Oltre a trascrivere i campi sul video dati, annota anche l'ultima chiave letta nello stesso campo del file guida. I due usi del campo (chiave di ricerca e ultima chiave letta) sono in sinergia e non in contrasto per il modo in cui vengono usati.

```

* Trascrive dati dal record di database al video.
C DBW1 begsr
* Accantona la chiave nel video guida.
C eval W0CDPE=A1CDPE
* Trascrive i campi dal record di database al video dati.
C clear W1
C eval W1CDPE=A1CDPE
C eval W1DEPE=A1DEPE
C eval W1PRNA=A1PRNA
C eval W1PRRE=A1PRRE
C endsr
*-----

```

21.6.2.20. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1BD.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1BD.txt)

21.6.3. Funzionamento del programma CNA1BD.

21.6.3.1. Entra sul video guida.

21.6.3.2. Sul video guida si può abbandonare il programma premendo Fine.

21.6.3.3. Sul video guida si può, col campo di ricerca vuoto, richiedere la visualizzazione del primo record o dell'ultimo record premendo PageDown o PageUp.

21.6.3.4. Sul video guida si può, col campo di ricerca valorizzato, richiedere la visualizzazione del record con la chiave uguale o successiva premendo PageDown.

21.6.3.5. Sul video guida si può, col campo di ricerca valorizzato, richiedere la visualizzazione del record con la chiave precedente premendo PageUp.

21.6.3.6. Dal video dati si può abbandonare il programma premendo Fine.

21.6.3.7. Sul video dati si può passare a visualizzare il record successivo o il precedente premendo PageDown o PageUp.

21.6.3.8. Quando paginando dal video dati incontra inizio o fine file, ripresenta il video guida.

21.6.3.9. Se dal video dati si richiede il video guida, questo si ripresenta riportando la chiave del record dati anche sul video guida.

## 21.7. Visualizzazione anagrafica con video guida e video dati con decodifica (oggetti CNA1C\*).

21.7.1. Sorgente commentato del video CNA1CW Dspf. Ottenuto modificando la versione CNA1BW.

```

* Claudio Neroni 27-02-2021 Creato per Corso.
* CNA1CW
* Anagrafica persone dsp decodifica
A DSPSIZ(24 80 *DS3)
A CHGINPDFI
A INDARA
A INDXT(44 'Protezione campi')
A R W0
A TEXT('Video Guida')
A BLINK
A CA03(03 'Fine')
A PAGEUP(25 'Precedente')
A PAGEDOWN(26 'Successivo')
A 1 32'Anagrafica persona'
A DSPATR(UL)
A DSPATR(HI)
A 2 38'Guida'
A DSPATR(HI)
A 4 10'Codice persona:'
A W0CDPE R B 4 27REFFLD(A1CDPE CNA1)
A DSPATR(UL)

```

|       |        |   |    |                           |
|-------|--------|---|----|---------------------------|
| A     |        |   |    | DSPATR(HI)                |
| A     |        |   | 23 | 2'F3=Fine'                |
| A     |        |   | 23 | 11'PagDw=Succ'            |
| A     |        |   | 23 | 23'PagUp=Prec'            |
| A     | R W1   |   |    |                           |
| A     |        |   |    | TEXT('Video Dati')        |
| A     |        |   |    | BLINK                     |
| A     |        |   |    | CA03(03 'Fine')           |
| A     |        |   |    | CA12(12 'Guida')          |
| A     |        |   |    | PAGEUP(25 'Precedente')   |
| A     |        |   |    | PAGEDOWN(26 'Successivo') |
| A     |        |   | 1  | 32'Anagrafica persona'    |
| A     |        |   |    | DSPATR(UL)                |
| A     |        |   |    | DSPATR(HI)                |
| A     |        |   | 2  | 39'Dati'                  |
| A     |        |   |    | DSPATR(HI)                |
| A     |        |   | 4  | 10'Codice persona:'       |
| A     | W1CDPE | R | B  | 4 27REFFLD(A1CDPE CNA1)   |
| A 44  |        |   |    | DSPATR(PR)                |
| A N44 |        |   |    | DSPATR(UL)                |
| A     |        |   |    | DSPATR(HI)                |
| A     |        |   | 5  | 5'Descrizione persona:'   |
| A     | W1DEPE | R | B  | 5 27REFFLD(A1DEPE CNA1)   |
| A 44  |        |   |    | DSPATR(PR)                |
| A N44 |        |   |    | DSPATR(UL)                |
| A     |        |   |    | DSPATR(HI)                |
| A     |        |   | 6  | 7'Provincia nascita:'     |
| A     | W1PRNA | R | B  | 6 27REFFLD(A1PRNA CNA1)   |
| A 44  |        |   |    | DSPATR(PR)                |
| A N44 |        |   |    | DSPATR(UL)                |
| A     |        |   |    | DSPATR(HI)                |

#### 21.7.1.1. Aggiungo decodifica della provincia di nascita.

|       |         |   |   |                           |
|-------|---------|---|---|---------------------------|
| A     | W1PRNAD | R | O | 6 30REFFLD(T1DEPR CNT1)   |
| A     |         |   |   | 7 5'Provincia residenza:' |
| A     | W1PRRE  | R | B | 7 27REFFLD(A1PRRE CNA1)   |
| A 44  |         |   |   | DSPATR(PR)                |
| A N44 |         |   |   | DSPATR(UL)                |
| A     |         |   |   | DSPATR(HI)                |

#### 21.7.1.2. Aggiungo decodifica della provincia di residenza.

|   |         |   |    |                         |
|---|---------|---|----|-------------------------|
| A | W1PRRED | R | O  | 7 30REFFLD(T1DEPR CNT1) |
| A |         |   | 23 | 2'F3=Fine'              |
| A |         |   | 23 | 11'F12=Guida'           |
| A |         |   | 23 | 22'PagDw=Succ'          |
| A |         |   | 23 | 34'PagUp=Prec'          |

#### 21.7.1.3. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1CW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1CW.txt)

#### 21.7.2. Sorgente commentato del programma CNA1CD Rpgle. Ottenuto modificando la versione CNA1BD.

```
* Claudio Neroni 27-02-2021 Creato per Corso.
* CNA1CD
* Anagrafica persone dsp decodifica
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
```

#### 21.7.2.1. Tabella provincie per codice con le caratteristiche: input, full procedural, external described, keyed, disk

```
* Tabella provincie
FCNT11 if e k disk
* Anagrafica persone dsp decodifica
FCNA1CW cf e workstn
*-----
* Predispone chiusura.
C seton lr
* Predispone protezione campi.
C seton 44
* Gestisce video guida.
C exsr SRW0
*-----
* Gestisce video guida.
C SRW0 begsr
* Pulisce video guida.
C clear W0
* Cicla intorno al video guida.
C do *hival
* Emette video guida.
C exfmt W0
* Se richiesta fine, abbandona.
C 03 leave
* Se richiesto PagUp.
C if *in25
```

```

* Si posiziona prima della chiave di ricerca.
C W0CDPE setll A1
* Legge all'indietro.
C readp A1
* Se non trova record, ritenta a partire dal fondo.
C if %eof
C *hival setgt A1
C readp A1
C endif
* Se richiesto PagUp o Enter,
C else
* Si posiziona prima della chiave di ricerca.
C W0CDPE setll A1
* Legge in avanti.
C read A1
* Se richiesto...
C endif
* Se non trova record, ricicla.
C if %eof
C iter
C endif
* Gestisce video dati.
C exsr SRW1
* Se richiesta fine, abbandona.
C 03 leave
* Cicla intorno al video guida.
C enddo
C endsr
*-----
* Gestisce video dati.
C SRW1 begsr
* Cicla intorno al video dati.
C do *hival
* Trascrive dati dal record di database al video.
C exsr DBW1

```

#### 21.7.2.2. Chiamo la decodifica dei codici presenti nel video.

```

* Decodifica video dati.
C exsr DEW1
* Emette video dati.
C exfmt W1
* Se richiesta fine, abbandona.
C 03 leave
* Se richiesta Guida, abbandona.
C 12 leave
* Se richiesto PagUp.
C if *in25
* Si posiziona prima della chiave ultima vista.
C W0CDPE setll A1
* Legge all'indietro.
C readp A1
* Se richiesto PagDw o Enter,
C else
* Si posiziona oltre la chiave ultima vista.
C W0CDPE setgt A1
* Legge in avanti.
C read A1
* Se richiesto...
C endif
* Se non trova record, abbandona.
C if %eof
C clear
C W0CDPE
C leave
C endif
* Cicla intorno al video dati.
C enddo
C endsr
*-----
* Trascrive dati dal record di database al video.
C DBW1 begsr
* Accantona la chiave nel video guida.
C eval W0CDPE=A1CDPE
* Trascrive i campi dal record di database al video dati.
C clear
C W1
C eval W1CDPE=A1CDPE
C eval W1DEPE=A1DEPE
C eval W1PRNA=A1PRNA
C eval W1PRRE=A1PRRE
C endsr
*-----

```

#### 21.7.2.3. Aggiungo subroutine di decodifica dei codici presenti nel video.

```

* Decodifica video dati.
C DEW1 begsr

```

#### 21.7.2.4. Decodifica il primo codice.

Se il codice è in bianco, non decodifica.

Se il codice è valorizzato. Assume descrizione di interrogativi.

Se trova il codice nell'apposita tabella, ne utilizza la descrizione.

```
* Decodifica provincia di nascita.
C clear W1PRNAD
C if W1PRNA<>*blank
C eval W1PRNAD=*all'?'
C W1PRNA chain T1
C if %found
C eval W1PRNAD=T1DEPR
C endif
C endif
```

#### 21.7.2.5. Decodifica il secondo codice.

Utilizza la stessa logica per tutte le decodifiche necessarie.

```
* Decodifica provincia di residenza.
C clear W1PRRED
C if W1PRRE<>*blank
C eval W1PRRED=*all'?'
C W1PRRE chain T1
C if %found
C eval W1PRRED=T1DEPR
C endif
C endif
C endsr
*-----
```

#### 21.7.2.6. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1CD.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1CD.txt)

#### 21.7.3. Funzionamento del programma CNA1CD.

##### 21.7.3.1. Identico alla versione precedente.

##### 21.7.3.2. Aggiunte soltanto le decodifiche della provincia di nascita e di residenza sul video dati.

#### 21.8. Creazione di una vista logica per descrizione sull'anagrafica persone.

##### 21.8.1. Per poter accedere all'anagrafica in ordine di descrizione persona, creo il seguente logico. Fornisco solo il sorgente che va compilato al solito modo.

```
* Claudio Neroni 28-02-2021 Creato per Corso.
* CNA12
* Anagrafica persone logico descrizione
*
* Chiave unica virtuale perche' codice e' chiave unica nel logico 1.
A*** UNIQUE
* Nelle chiavi rialza le minuscole tramite una tabella di sistema.
A ALTSEQ(QSYSTRNTBL)
A R A1 PFILE(CNA1)
A K A1DEPE
* Non rialza il campo codice.
A K A1CDPE NOALTSEQ
```

##### 21.8.1.1. La chiave è virtualmente unica perché include un campo chiave già usato come chiave unica nell'altro logico del file fisico. Non occorre la parola chiave UNIQUE!

##### 21.8.1.2. La tabella (tipo oggetto \*TBL) menzionata nella keyword ALTSEQ(QSYSTRNTBL) (System translation table che rialza i caratteri minuscoli a caratteri maiuscoli) fa trattare le chiavi del file come se fossero in maiuscolo.

##### 21.8.1.3. NOALTSEQ sul campo codice impedisce il rialzamento ad uso delle chiavi che così agisce solo sul campo descrizione.

##### 21.8.1.4. La descrizione potrebbe essere uguale per persone diverse e usata da sola non garantirebbe l'univocità nell'ordinamento dei record. L'aggiunta del campo codice in coda alla chiave risolve il problema rendendo virtualmente unica la chiave con due campi.

##### 21.8.1.5. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA12.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA12.txt)

#### 21.9. Visualizzazione anagrafica con doppio video guida e video dati con decodifica (oggetti CNA1D\*).

##### 21.9.1. Sorgente commentato del video CNA1DW Dspf. Ottenuto modificando la versione CNA1CW.

```
* Claudio Neroni 28-02-2021 Creato.
* CNA1DW
* Anagrafica persone dsp guida doppia
A DSPSIZ(24 80 *DS3)
A CHGINPDFT
A INDARA
A INDTXT(44 'Protezione campi')
```

##### 21.9.1.1. Il vecchio video guida W0 cambia nome e diventa W01.

|   |       |                                |
|---|-------|--------------------------------|
| A | R W01 |                                |
| A |       | TEXT('Video Guida per codice') |
| A |       | BLINK                          |
| A |       | CA03(03 'Fine')                |

21.9.1.2. Permetto la commutazione tra i due video guida con CF11.

|   |        |   |      |                           |
|---|--------|---|------|---------------------------|
| A |        |   |      | CF11(11 'PerDescrizione') |
| A |        |   |      | PAGEUP(25 'Precedente')   |
| A |        |   |      | PAGEDOWN(26 'Successivo') |
| A |        |   | 1 32 | 'Anagrafica persona'      |
| A |        |   |      | DSPATR(UL)                |
| A |        |   |      | DSPATR(HI)                |
| A |        |   | 2 38 | 'Guida'                   |
| A |        |   |      | DSPATR(HI)                |
| A |        |   | 4 10 | 'Codice persona:'         |
| A | W0CDPE | R | B 4  | 27REFFLD(A1CDPE CNA1)     |
| A |        |   |      | DSPATR(UL)                |
| A |        |   |      | DSPATR(HI)                |
| A |        |   | 23 2 | 'F3=Fine'                 |

21.9.1.3. Aggiungo la costante descrittiva di CF11=xDes.

|   |  |  |  |                |
|---|--|--|--|----------------|
| A |  |  |  | +2'F11=xDes'   |
| A |  |  |  | +2'PagDw=Succ' |
| A |  |  |  | +2'PagUp=Prec' |

21.9.1.4. Si duplica il video guida per codice W01 nel nuovo video guida W02 per descrizione e codice. I due video guida saranno impaginati in modo che campi e costanti presenti in entrambi occupino, il più possibile, le stesse posizioni.

|   |       |  |
|---|-------|--|
| A | R W02 |  |
|---|-------|--|

21.9.1.5. Correggo il testo del video guida.

|   |  |  |  |                                     |
|---|--|--|--|-------------------------------------|
| A |  |  |  | TEXT('Video Guida per descrizione') |
| A |  |  |  | BLINK                               |
| A |  |  |  | CA03(03 'Fine')                     |

21.9.1.6. Correggo 'PerDescrizione' in 'PerCodice'.

|   |  |  |      |                           |
|---|--|--|------|---------------------------|
| A |  |  |      | CF11(11 'PerCodice')      |
| A |  |  |      | PAGEUP(25 'Precedente')   |
| A |  |  |      | PAGEDOWN(26 'Successivo') |
| A |  |  | 1 32 | 'Anagrafica persona'      |
| A |  |  |      | DSPATR(UL)                |
| A |  |  |      | DSPATR(HI)                |
| A |  |  | 2 38 | 'Guida'                   |
| A |  |  |      | DSPATR(HI)                |

21.9.1.7. Aggiungo il campo descrizione persona. I campi sono disposti sul video come nella chiave del file logico.

|   |        |   |      |                           |
|---|--------|---|------|---------------------------|
| A |        |   |      | 3 5'Descrizione persona:' |
| A | W0DEPE | R | B 3  | 27REFFLD(A1DEPE CNA1)     |
| A |        |   |      | DSPATR(UL)                |
| A |        |   |      | DSPATR(HI)                |
| A |        |   | 4 10 | 'Codice persona:'         |
| A | W0CDPE | R | B 4  | 27REFFLD(A1CDPE CNA1)     |
| A |        |   |      | DSPATR(UL)                |
| A |        |   |      | DSPATR(HI)                |
| A |        |   | 23 2 | 'F3=Fine'                 |

21.9.1.8. Correggo la costante descrittiva di CF11 da 'xDes' a 'xCod'.

|   |      |  |      |                           |
|---|------|--|------|---------------------------|
| A |      |  |      | +2'F11=xCod'              |
| A |      |  |      | +2'PagDw=Succ'            |
| A |      |  |      | +2'PagUp=Prec'            |
| A | R W1 |  |      |                           |
| A |      |  |      | TEXT('Video Dati')        |
| A |      |  |      | BLINK                     |
| A |      |  |      | CA03(03 'Fine')           |
| A |      |  |      | CA12(12 'Guida')          |
| A |      |  |      | PAGEUP(25 'Precedente')   |
| A |      |  |      | PAGEDOWN(26 'Successivo') |
| A |      |  | 1 32 | 'Anagrafica persona'      |
| A |      |  |      | DSPATR(UL)                |
| A |      |  |      | DSPATR(HI)                |

21.9.1.9. Aggiungo informazioni sullo stato attivo per l'ordinamento della visualizzazione.

|       |        |   |      |                            |
|-------|--------|---|------|----------------------------|
| A 41  |        |   |      | 2 33'Dati per codice'      |
| A     |        |   |      | DSPATR(HI)                 |
| A 42  |        |   |      | 2 31'Dati per descrizione' |
| A     |        |   |      | DSPATR(HI)                 |
| A     |        |   | 4 10 | 'Codice persona:'          |
| A     | W1CDPE | R | B 4  | 27REFFLD(A1CDPE CNA1)      |
| A 44  |        |   |      | DSPATR(PR)                 |
| A N44 |        |   |      | DSPATR(UL)                 |
| A     |        |   |      | DSPATR(HI)                 |
| A     |        |   | 5 5  | 'Descrizione persona:'     |

|   |         |   |    |   |                        |
|---|---------|---|----|---|------------------------|
| A | W1DEPE  | R | B  | 5 | 27REFFLD(A1DEPE CNA1)  |
| A | 44      |   |    |   | DSPATR(PR)             |
| A | N44     |   |    |   | DSPATR(UL)             |
| A |         |   |    |   | DSPATR(HI)             |
| A |         |   | 6  | 7 | 'Provincia nascita:'   |
| A | W1PRNA  | R | B  | 6 | 27REFFLD(A1PRNA CNA1)  |
| A | 44      |   |    |   | DSPATR(PR)             |
| A | N44     |   |    |   | DSPATR(UL)             |
| A |         |   |    |   | DSPATR(HI)             |
| A | W1PRNAD | R | O  | 6 | 30REFFLD(T1DEPR CNT1)  |
| A |         |   | 7  | 5 | 'Provincia residenza:' |
| A | W1PRRE  | R | B  | 7 | 27REFFLD(A1PRRE CNA1)  |
| A | 44      |   |    |   | DSPATR(PR)             |
| A | N44     |   |    |   | DSPATR(UL)             |
| A |         |   |    |   | DSPATR(HI)             |
| A | W1PRRED | R | O  | 7 | 30REFFLD(T1DEPR CNT1)  |
| A |         |   | 23 | 2 | 'F3=Fine'              |
| A |         |   |    |   | +2'F12=Guida'          |
| A |         |   |    |   | +2'PagDw=Succ'         |
| A |         |   |    |   | +2'PagUp=Prec'         |

21.9.1.10. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1DW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1DW.txt)

21.9.2. Sorgente commentato del programma CNA1DD Rpgle. Ottenuto modificando la versione CNA1CD.

```

* Claudio Neroni 28-02-2021 Creato per Corso.
* CNA1DD
* Anagrafica persone dsp guida doppia
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk rename (A1:A11)

```

21.9.2.1. Aggiungo la vista logica per descrizione-codice dello stesso fisico visto dal file precedente.

```

* Anagrafica persone logico descrizione
FCNA12 if e k disk rename (A1:A12)
* Tabella provincie
FCNT11 if e k disk
* Anagrafica persone dsp guida doppia
FCNA1DW cf e workstn
*-----
* Predisporre chiusura.
C seton LR

```

21.9.2.2. Assegno gli indicatori 41 e 42 al condizionamento dei due video guida

```

* Assume di usare video guida per codice.
C seton 41
C setoff 42
* Predisporre protezione campi.
C seton 44
* Gestisce video guida.
C exsrr SRW0
*-----

```

21.9.2.3. Un'unica subroutine gestisce entrambi i video guida.

```

* Gestisce video guida.
C SRW0 begsrr

```

21.9.2.4. Quando la chiave di un file è composta da un solo campo, quel campo può fungere anche da chiave di ricerca rpg nelle operazioni di posizionamento e lettura.

21.9.2.5. Quando invece la chiave di un file è composta da almeno due campi, è necessario definire una chiave di ricerca esplicita. È il caso del logico per descrizione-codice.

21.9.2.6. Per omogeneità conviene definire una chiave esplicita anche per il logico per codice.

```

* Definisce chiave per codice.
C K1 klist
C kfld W0CDPE
* Definisce chiave per descrizione.
C K2 klist
C kfld W0DEPE
C kfld W0CDPE

```

21.9.2.7. Un discreto numero di istruzioni della subroutine alla versione precedente vanno ora raddoppiate condizionandole con gli indicatori di guida in uso 41 e 42.

```

* Pulisce video guida.
C clear W01
C clear W02
* Cicla intorno al video guida.
C do *hival
* Emette video guida.
C 41 exfmt W01
C 42 exfmt W02
* Se richiesta fine, abbandona.
C 03 leave

```

|   |                                                                    |    |
|---|--------------------------------------------------------------------|----|
|   | * Se richiesto swith tra guida per codice e guida per descrizione, |    |
|   | * commuta la scelta del video guida e ricicla.                     |    |
| C | if *in11                                                           |    |
| C | if *in41                                                           |    |
| C | setoff                                                             | 41 |
| C | seton                                                              | 42 |
| C | else                                                               |    |
| C | seton                                                              | 41 |
| C | setoff                                                             | 42 |
| C | endif                                                              |    |
| C | iter                                                               |    |
| C | endif                                                              |    |
|   | * Se richiesto PagUp.                                              |    |
| C | if *in25                                                           |    |
|   | * Si posiziona prima della chiave di ricerca.                      |    |

21.9.2.8. Si osservi qui per la prima volta sul primo logico l'utilizzo della chiave dove nella versione precedente c'era solo il campo codice.

|   |                                                     |       |        |
|---|-----------------------------------------------------|-------|--------|
| C | 41K1                                                | setll | A11    |
| C | 42K2                                                | setll | A12    |
|   | * Legge all'indietro.                               |       |        |
| C | 41                                                  | readp | A11    |
| C | 42                                                  | readp | A12    |
|   | * Se non trova record, ritenta a partire dal fondo. |       |        |
| C |                                                     | if    | %eof   |
| C | 41*hival                                            | setgt | A11    |
| C | 42*hival                                            | setgt | A12    |
| C | 41                                                  | readp | A11    |
| C | 42                                                  | readp | A12    |
| C |                                                     | endif |        |
|   | * Se richiesto PagUp o Enter,                       |       |        |
| C |                                                     | else  |        |
|   | * Si posiziona prima della chiave di ricerca.       |       |        |
| C | 41K1                                                | setll | A11    |
| C | 42K2                                                | setll | A12    |
|   | * Legge in avanti.                                  |       |        |
| C | 41                                                  | read  | A11    |
| C | 42                                                  | read  | A12    |
|   | * Se richiesto ...                                  |       |        |
| C |                                                     | endif |        |
|   | * Se non trova record, ricicla.                     |       |        |
| C |                                                     | if    | %eof   |
| C |                                                     | iter  |        |
| C |                                                     | endif |        |
|   | * Gestisce video dati.                              |       |        |
| C |                                                     | exsr  | SRW1   |
|   | * Se richiesta fine, abbandona.                     |       |        |
| C | 03                                                  | leave |        |
|   | * Cicla intorno al video guida.                     |       |        |
| C |                                                     | enddo |        |
| C |                                                     | endsr |        |
|   | *-----                                              |       |        |
|   | * Gestisce video dati.                              |       |        |
|   | * Va chiamata con un record gia' in canna.          |       |        |
| C | SRW1                                                | begsr |        |
|   | * Cicla intorno a video dati.                       |       |        |
| C |                                                     | do    | *hival |
|   | * Trascrive dati dal record di database al video.   |       |        |
| C |                                                     | exsr  | DBW1   |
|   | * Decodifica video dati.                            |       |        |
| C |                                                     | exsr  | DEW1   |
|   | * Emette video dati.                                |       |        |
| C |                                                     | exfmt | W1     |
|   | * Se richiesta fine, abbandona.                     |       |        |
| C | 03                                                  | leave |        |
|   | * Se richiesta Guida, abbandona.                    |       |        |
| C | 12                                                  | leave |        |
|   | * Se richiesto PagUp.                               |       |        |
| C |                                                     | if    | *in25  |
|   | * Si posiziona prima della chiave ultima vista.     |       |        |
| C | 41K1                                                | setll | A11    |
| C | 42K2                                                | setll | A12    |
|   | * Legge all'indietro.                               |       |        |
| C | 41                                                  | readp | A11    |
| C | 42                                                  | readp | A12    |
|   | * Se richiesto PagDw o Enter,                       |       |        |
| C |                                                     | else  |        |
|   | * Si posiziona oltre la chiave ultima vista.        |       |        |
| C | 41K1                                                | setgt | A11    |
| C | 42K2                                                | setgt | A12    |
|   | * Legge in avanti.                                  |       |        |
| C | 41                                                  | read  | A11    |
| C | 42                                                  | read  | A12    |



```

* Se richiesto ...
C endif
* Se non trova record, pulisce chiave accantonata e abbandona.
C if %eof
C clear W0CDPE
C clear W0DEPE
C leave
C endif
* Cicla intorno a video dati.
C enddo
C endsr
*-----

```

21.9.2.9. L'ultima chiave letta è ora composta da due campi.

```

* Trascrive dati dal record di database al video.
C DBW1 begsr
* Accantona la chiave nel video guida.
C eval W0CDPE=A1CDPE
C eval W0DEPE=A1DEPE
* Trascrive i campi dal record di database al video dati.
C clear W1
C eval W1CDPE=A1CDPE
C eval W1DEPE=A1DEPE
C eval W1PRNA=A1PRNA
C eval W1PRRE=A1PRRE
C endsr
*-----
* Decodifica video dati.
C DEW1 begsr
* Decodifica provincia di nascita.
C clear W1PRNAD
C if W1PRNA<>*blank
C eval W1PRNAD=*all'?'
C W1PRNA chain T1
C if %found
C eval W1PRNAD=T1DEPR
C endif
C endif
* Decodifica provincia di residenza.
C clear W1PRRED
C if W1PRRE<>*blank
C eval W1PRRED=*all'?'
C W1PRRE chain T1
C if %found
C eval W1PRRED=T1DEPR
C endif
C endif
C endsr
*-----

```

21.9.2.10. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1DD.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1DD.txt)

21.9.3. Funzionamento del programma CNA1DD.

21.9.3.1. Entra sul video guida per codice.

21.9.3.2. Da ognuno dei due video guida si può scegliere di commutare sull'altro.

21.9.3.3. Il video guida da cui si entra nel video dati determina anche l'ordine di esplorazione dei record dati stessi in accordo con una delle due viste logiche usate dal programma.

21.9.3.4. Identico per il resto alla versione precedente.

**21.10. Visualizzazione e modifica anagrafica con doppio video guida e video dati con decodifica e controllo errori (oggetti CNA1E\*).**

21.10.1. Sorgente commentato del video CNA1EW Dspf. Ottenuto modificando la versione CNA1DW.

21.10.1.1. Le aggiunte e le modifiche riguardano l'introduzione della funzione di modifica e delle segnalazioni di errore.

```

* Claudio Neroni 04-03-2021 Creato per Corso.
* CNA1EW
* Anagrafica persone dsp-chg
A DSPSIZ(24 80 *DS3)
A CHGINPDFT
A INDARA
A INDTXT(44 'Protezione campi')

```

21.10.1.2. ERRSFL evita che gli errori ERRMSG mandino in blocco la tastiera.

```

A ERRSFL
A R W01
A TEXT('Video Guida per codice')
A BLINK
A CA03(03 'Fine')

```

21.10.1.3. Abilita il comando di commutazione tra visualizzazione e modifica.

|   |  |  |   |    |                           |
|---|--|--|---|----|---------------------------|
| A |  |  |   |    | CF07(07 'Dsp/Chg')        |
| A |  |  |   |    | CF11(11 'PerDescrizione') |
| A |  |  |   |    | PAGEUP(25 'Precedente')   |
| A |  |  |   |    | PAGEDOWN(26 'Successivo') |
| A |  |  | 1 | 32 | 'Anagrafica persona'      |
| A |  |  |   |    | DSPATR(UL)                |
| A |  |  |   |    | DSPATR(HI)                |

21.10.1.4. Campo descrittivo della modalità corrente di funzionamento del programma.

|   |        |   |   |   |    |                       |
|---|--------|---|---|---|----|-----------------------|
| A | WWMOD  |   | 3 | O | 1  | 76TEXT('Modalita''')  |
| A |        |   |   |   | 2  | 38'Guida'             |
| A |        |   |   |   |    | DSPATR(HI)            |
| A |        |   |   |   | 4  | 10'Codice persona:'   |
| A | W0CDPE | R |   | B | 4  | 27REFFLD(A1CDPE CNA1) |
| A |        |   |   |   |    | DSPATR(UL)            |
| A |        |   |   |   |    | DSPATR(HI)            |
| A |        |   |   |   | 23 | 2'F3=Fine'            |

21.10.1.5. Costanti descrittive del comando di commutazione tra visualizzazione e modifica.

|   |       |  |  |  |    |                                     |
|---|-------|--|--|--|----|-------------------------------------|
| A | 44    |  |  |  | 23 | 11'F7=Chg'                          |
| A | N44   |  |  |  | 23 | 11'F7=Dsp'                          |
| A |       |  |  |  |    | +2'F11=xDes'                        |
| A |       |  |  |  |    | +2'PagDw=Succ'                      |
| A |       |  |  |  |    | +2'PagUp=Prec'                      |
| A | R W02 |  |  |  |    |                                     |
| A |       |  |  |  |    | TEXT('Video Guida per descrizione') |
| A |       |  |  |  |    | BLINK                               |
| A |       |  |  |  |    | CA03(03 'Fine')                     |

21.10.1.6. Abilita il comando di commutazione tra visualizzazione e modifica.

|   |  |  |   |    |                           |
|---|--|--|---|----|---------------------------|
| A |  |  |   |    | CF07(07 'Dsp/Chg')        |
| A |  |  |   |    | CF11(11 'PerCodice')      |
| A |  |  |   |    | PAGEUP(25 'Precedente')   |
| A |  |  |   |    | PAGEDOWN(26 'Successivo') |
| A |  |  | 1 | 32 | 'Anagrafica persona'      |
| A |  |  |   |    | DSPATR(UL)                |
| A |  |  |   |    | DSPATR(HI)                |

21.10.1.7. Campo descrittivo della modalità corrente di funzionamento del programma.

|   |        |   |  |   |    |                         |
|---|--------|---|--|---|----|-------------------------|
| A | WWMOD  | R |  | O | 1  | 76REFFLD(WWMOD *SRC)    |
| A |        |   |  |   | 2  | 38'Guida'               |
| A |        |   |  |   |    | DSPATR(HI)              |
| A |        |   |  |   | 3  | 5'Descrizione persona:' |
| A | W0DEPE | R |  | B | 3  | 27REFFLD(A1DEPE CNA1)   |
| A |        |   |  |   |    | DSPATR(UL)              |
| A |        |   |  |   |    | DSPATR(HI)              |
| A |        |   |  |   | 4  | 10'Codice persona:'     |
| A | W0CDPE | R |  | B | 4  | 27REFFLD(A1CDPE CNA1)   |
| A |        |   |  |   |    | DSPATR(UL)              |
| A |        |   |  |   |    | DSPATR(HI)              |
| A |        |   |  |   | 23 | 2'F3=Fine'              |

21.10.1.8. Costanti descrittive del comando di commutazione tra visualizzazione e modifica.

|   |      |  |  |  |    |                    |
|---|------|--|--|--|----|--------------------|
| A | 44   |  |  |  | 23 | 11'F7=Chg'         |
| A | N44  |  |  |  | 23 | 11'F7=Dsp'         |
| A |      |  |  |  |    | +2'F11=xCod'       |
| A |      |  |  |  |    | +2'PagDw=Succ'     |
| A |      |  |  |  |    | +2'PagUp=Prec'     |
| A | R W1 |  |  |  |    |                    |
| A |      |  |  |  |    | TEXT('Video Dati') |
| A |      |  |  |  |    | BLINK              |
| A |      |  |  |  |    | CA03(03 'Fine')    |

21.10.1.9. Abilita il comando di commutazione tra visualizzazione e modifica.

|   |  |  |  |  |                           |
|---|--|--|--|--|---------------------------|
| A |  |  |  |  | CF07(07 'Dsp/Chg')        |
| A |  |  |  |  | CA12(12 'Guida')          |
| A |  |  |  |  | PAGEUP(25 'Precedente')   |
| A |  |  |  |  | PAGEDOWN(26 'Successivo') |

21.10.1.10. CHANGE(33)="digitati dati". Serve solo per il programma farlocco CNA1EX.

|   |  |  |   |    |                      |
|---|--|--|---|----|----------------------|
| A |  |  |   |    | CHANGE(33)           |
| A |  |  | 1 | 32 | 'Anagrafica persona' |
| A |  |  |   |    | DSPATR(UL)           |
| A |  |  |   |    | DSPATR(HI)           |

21.10.1.11. Campo descrittivo della modalità corrente di funzionamento del programma.

|   |        |   |  |   |   |                          |
|---|--------|---|--|---|---|--------------------------|
| A | WWMOD  | R |  | O | 1 | 76REFFLD(WWMOD *SRC)     |
| A | 41     |   |  |   | 2 | 33'Dati per codice'      |
| A |        |   |  |   |   | DSPATR(HI)               |
| A | 42     |   |  |   | 2 | 31'Dati per descrizione' |
| A |        |   |  |   |   | DSPATR(HI)               |
| A |        |   |  |   | 4 | 10'Codice persona:'      |
| A | W1CDPE | R |  | B | 4 | 27REFFLD(A1CDPE CNA1)    |

#### 21.10.1.12. Segnalazioni di errore dopo il tentativo di ricalco o di aggiunta record.

|         |                                         |
|---------|-----------------------------------------|
| A 60 61 | ERRMSG('Record occupato da altri')      |
| A 60 62 | ERRMSG('Record cancellato da altri')    |
| A 60 63 | ERRMSG('Record modificato da altri')    |
| A 60 64 | ERRMSG('Chiave gia' presente nel file') |

#### 21.10.1.13. Segnalazioni di errore sul contenuto del campo.

|         |                                    |
|---------|------------------------------------|
| A 60 71 | ERRMSG('Dato obbligatorio')        |
| A 44    | DSPATR(PR)                         |
| A N44   | DSPATR(UL)                         |
| A       | DSPATR(HI)                         |
| A       | 5 5'Descrizione persona:'          |
| A       | W1DEPE R B 5 27REFFLD(A1DEPE CNA1) |

#### 21.10.1.14. Segnalazioni di errore sul contenuto del campo.

|         |                                                                |
|---------|----------------------------------------------------------------|
| A 60 72 | ERRMSG('Dato obbligatorio')                                    |
| A 60 73 | ERRMSG('Descrizione deve essere piu-<br>' ' lunga del codice') |
| A 44    | DSPATR(PR)                                                     |
| A N44   | DSPATR(UL)                                                     |
| A       | DSPATR(HI)                                                     |
| A       | CHECK(LC)                                                      |
| A       | 6 7'Provincia nascita:'                                        |
| A       | W1PRNA R B 6 27REFFLD(A1PRNA CNA1)                             |

#### 21.10.1.15. Segnalazioni di errore sul contenuto del campo.

|         |                                     |
|---------|-------------------------------------|
| A 60 74 | ERRMSG('Dato obbligatorio')         |
| A 60 75 | ERRMSG('Manca in tabella')          |
| A 44    | DSPATR(PR)                          |
| A N44   | DSPATR(UL)                          |
| A       | DSPATR(HI)                          |
| A       | W1PRNAD R O 6 30REFFLD(T1DEPR CNT1) |
| A       | 7 5'Provincia residenza:'           |
| A       | W1PRE R B 7 27REFFLD(A1PRE CNA1)    |

#### 21.10.1.16. Segnalazioni di errore sul contenuto del campo.

|         |                                     |
|---------|-------------------------------------|
| A 60 76 | ERRMSG('Manca in tabella')          |
| A 44    | DSPATR(PR)                          |
| A N44   | DSPATR(UL)                          |
| A       | DSPATR(HI)                          |
| A       | W1PRRED R O 7 30REFFLD(T1DEPR CNT1) |
| A       | 23 2'F3=Fine'                       |

#### 21.10.1.17. Abilita il comando di commutazione tra visualizzazione e modifica.

|       |                |
|-------|----------------|
| A 44  | 23 11'F7=Chg'  |
| A N44 | 23 11'F7=Dsp'  |
| A     | +2'F12=Guida'  |
| A     | +2'PagDw=Succ' |
| A     | +2'PagUp=Prec' |

#### 21.10.1.18. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1EW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1EW.txt)

#### 21.10.2. Sorgente commentato del programma CNA1ED Rpgle. Ottenuto modificando la versione CNA1DD.

```
* Claudio Neroni 04-03-2021 Creato per Corso.
* CNA1ED
* Anagrafica persone dsp-chg
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk rename(A1:A11)
```

#### 21.10.2.1. Accede alla File information data structure del logico.

|                                         |                            |
|-----------------------------------------|----------------------------|
| F                                       | infd(FIDS1)                |
| * Anagrafica persone logico descrizione |                            |
| FCNA12                                  | if e k disk rename(A1:A12) |

#### 21.10.2.2. Accede alla File information data structure del logico.

|                              |              |
|------------------------------|--------------|
| F                            | infd(FIDS2)  |
| * Tabella provincie          |              |
| FCNT11                       | if e k disk  |
| * Anagrafica persone dsp-chg |              |
| FCNA1EW                      | cf e workstn |

#### 21.10.2.3. Aggiungo il fisico per eseguire il ricalco dei record. Il file verrà aperto soltanto al momento del primo uso.

|                             |                                 |
|-----------------------------|---------------------------------|
| * Anagrafica persone fisico |                                 |
| FCNA1                       | uf e disk rename(A1:UPD) usropn |
| *-----                      |                                 |

#### 21.10.2.4. Struttura dati dei campi letti dal file dati e suo salvataggio.

|                                                   |                       |
|---------------------------------------------------|-----------------------|
| * Struttura dati del file dati.                   |                       |
| D A1D                                             | e ds extname(CNA1:A1) |
| * Salvataggio della Struttura dati del file dati. |                       |

```

D A1DSV e ds extname(CNA1:A1) prefix(SV)
*-----

```

#### 21.10.2.5. Struttura dati dei campi letti dal video dati e suo salvataggio.

```

* Struttura dati del video dati.
D W1D e ds extname(CNA1EW:W1)
* Salvataggio della Struttura dati del video dati.
D W1DSV e ds extname(CNA1EW:W1) prefix(SV)
*-----

```

#### 21.10.2.6. File information data structure che permettono di conoscere i relative record number dei logici in lettura per ritrovare lo stesso record da aggiornare sul fisico.

```

* File information data structure
D FIDS1 ds
* Relative record number in data memmber
D FIDS1RR 397 400b 0
* File information data structure
D FIDS2 ds
* Relative record number in data memmber
D FIDS2RR 397 400b 0
*-----
* Predisporre chiusura.
C seton 1r
* Assume di usare video guida per codice.
C seton 41
C setoff 42
* Predisporre protezione campi.
C seton 44
* Gestisce video guida.
C exsr SRW0
*-----
* Gestisce video guida.
C SRW0 begsr
* Definisce chiave per codice.
C K1 klist
C kfld W0CDPE
* Definisce chiave per descrizione.
C K2 klist
C kfld W0DEPE
C kfld W0CDPE
* Pulisce video guida.
C clear W01
C clear W02
* Cicla intorno al video guida.
C do *hival

```

#### 21.10.2.7. Espone la modalità di esecuzione corrente del programma,

```

* Prepara modo.
C 44 eval WWMOD='DSP'
C N44 eval WWMOD='CHG'
* Emette video guida.
C 41 exfmt W01
C 42 exfmt W02
* Se richiesta fine, abbandona.
C 03 leave

```

#### 21.10.2.8. Commuta tra visualizzazione e modifica.

```

* Se richiesto switch tra visualizzazione e modifica,
* la esegue e ricicla.
C if *in07
C *IN44 comp *off 44
C iter
C endif
* Se richiesto switch tra guida per codice e guida per descrizione,
* commuta la scelta del video guida e ricicla.
C if *in11
C if *in41
C setoff 41
C seton 42
C else
C seton 41
C setoff 42
C endif
C iter
C endif
* Se richiesto PagUp.
C if *in25
* Si posiziona prima della chiave di ricerca.
C 41K1 setll A11
C 42K2 setll A12
* Legge all'indietro.
C 41 readp A11
C 42 readp A12
* Se non trova record, ritenta a partire dal fondo.

```

```

C if %eof
C 41*hival setgt A11
C 42*hival setgt A12
C 41 readp A11
C 42 readp A12
C endif
* Se richiesto PagUp o Enter,
C else
* Si posiziona prima della chiave di ricerca.
C 41K1 setll A11
C 42K2 setll A12
* Legge in avanti.
C 41 read A11
C 42 read A12
* Se richiesto ...
C endif
* Se non trova record, ricicla.
C if %eof
C iter
C endif
* Gestisce video dati.
C exsr SRW1
* Se richiesta fine, abbandona.
C 03 leave
* Cicla intorno al video guida.
C enddo
C endsr
*-----
* Gestisce video dati.
* Va chiamata con un record gia' in canna.
C SRW1 begsr
* Cicla intorno al video dati.
C do *hival

```

21.10.2.9. Se errore, diversifica la preparazione per l'emissione del video dati.

```

* Se non corre errore.
C if not *in60
* Trascrive dati dal record di database al video.
C exsr DBW1
* Decodifica video dati.
C exsr DEW1
* Se non corre errore.
C endif

```

21.10.2.10. La write senza errore (indicatore 60 provvisoriamente spento) prima dell'exfmt serve a far vedere le decodifiche che altrimenti non verrebbero rinfrescate a causa delle caratteristiche della emissione di errore a ricalco di un formato già presente sul video (che emette solo messaggio di errore e posizionamento cursore ma non i nuovi valori dei campi).

```

* Se corre errore, scrive base per errore.
C if *in60
C setoff 60
C write W1
C seton 60
C endif

```

21.10.2.11. Espone la modalità di esecuzione corrente del programma,

```

* Prepara modo.
C 44 eval WWMOD='DSP'
C N44 eval WWMOD='CHG'
* Emette video dati.
C exfmt W1

```

21.10.2.12. Spegne gli indicatori di errore dopo l'uso sul video.

```

* Spegne tutti gli errori.
C exsr OFF
* Se richiesta fine, abbandona.
C 03 leave
* Se richiesta Guida, abbandona.
C 12 leave

```

21.10.2.13. Commuta tra visualizzazione e modifica.

```

* Se richiesto switch tra visualizzazione e modifica,
* la esegue e ricicla.
C if *in07
C comp *off 44
C iter
C endif

```

21.10.2.14. Gestisce la modifica.

```

* Se corre modifica.
C if not *in44
* Decodifica video dati.
C exsr DEW1

```

```

* Controlla video dati.
C exsr CTW1
* Se errore, ricicla.
C 60 iter
* Se il video e' stato modificato.
C if W1D<>W1DSV
* Esegue update.
C exsr UPDB
* Se errore, ricicla.
C 60 iter
* Se il video e' stato modificato.
C endif
* Se corre modifica.
C endif

* Se richiesto PagUp.
C if *in25
* Si posiziona prima della chiave ultima vista.
C 41K1 setll A11
C 42K2 setll A12
* Legge all'indietro.
C 41 readp A11
C 42 readp A12
* Se richiesto PagDw o Enter,
C else
* Si posiziona oltre la chiave ultima vista.
C 41K1 setgt A11
C 42K2 setgt A12
* Legge in avanti.
C 41 read A11
C 42 read A12
* Se richiesto ...
C endif
* Se non trova record, pulisce chiave accantonata e abbandona.
C if %eof
C clear
C clear W0CDPE
C clear W0DEPE
C leave
C endif
* Cicla intorno al video dati.
C enddo
C endsr

*-----
* Trascrive dati dal record di database al video.
C DEW1 begsr
* Accantona la chiave nei campi dei video guida.
C eval W0CDPE=A1CDPE
C eval W0DEPE=A1DEPE
* Pulisce il video dati.
C clear
C clear W1
* Trascrive i campi dal database al video.
C eval W1CDPE=A1CDPE
C eval W1DEPE=A1DEPE
C eval W1PRNA=A1PRNA
C eval W1PRRE=A1PRRE

```

21.10.2.15. Salva dati sia del record che del video per farne uso più avanti per sapere che è avvenuta una modifica.

```

* Salva i dati del record letto.
C eval A1DSV=A1D
* Salva i dati del video.
C eval W1DSV=W1D
C endsr

*-----
* Decodifica video dati.
C DEW1 begsr
* Decodifica provincia di nascita.
C clear
C clear W1PRNAD
C if W1PRNA<>*blank
C eval W1PRNAD=*all'?'
C W1PRNA chain T1
C if %found
C eval W1PRNAD=T1DEPR
C endif
C endif
* Decodifica provincia di residenza.
C clear
C clear W1PRRED
C if W1PRRE<>*blank
C eval W1PRRED=*all'?'
C W1PRRE chain T1
C if %found
C eval W1PRRED=T1DEPR
C endif
C endif

```

```

C endsr
*-----

```

#### 21.10.2.16. Spegne gli indicatori di errore.

```

* Spegne tutti gli errori.
C OFF begsr
C move *off OFF31 31
C movea OFF31 *in(60)
C endsr
*-----

```

#### 21.10.2.17. Controlla gli errori.

```

* Controlla video dati.
C CTW1 begsr
* Spegne tutti gli errori.
C exsr OFF
* Esegue.
C do
* Codice obbligatorio.
C if W1CDPE=*blank
C seton
C leave
C endif
* Descrizione obbligatoria.
C if W1DEPE=*blank
C seton
C leave
C endif
* Descrizione piu' corta del codice.
C if %len(%trim(W1DEPE)) <= %len(%trim(W1CDPE))
C seton
C leave
C endif
* Provincia di nascita obbligatoria.
C if W1PRNA=*blank
C seton
C leave
C endif
* Provincia di nascita manca in tabella.
C if W1PRNAD=*all'?'
C seton
C leave
C endif
* Provincia di residenza manca in tabella.
C if W1PRRED=*all'?'
C seton
C leave
C endif
* Esegue.
C enddo
C endsr
*-----

```

#### 21.10.2.18. Trascrive da video a database.

```

* Trascrive dati dal video al record di database
C W1DB begsr
C eval A1CDPE=W1CDPE
C eval A1DEPE=W1DEPE
C eval A1PRNA=W1PRNA
C eval A1PRRE=W1PRRE
C endsr
*-----

```

#### 21.10.2.19. Ricalca il record controllando gli errori.

```

* Esegue update.
C UPDB begsr
* Se il file fisico per update non e' aperto, lo apre.
C if not %open(CNA1)
C open CNA1
C endif
* Esegue.
C do
* Recupera il record fisico per ricalcarlo.
C 41FIDS1RR chain(e) UPD
C 42FIDS2RR chain(e) UPD
* Se il record e' occupato, accende errore e abbandona.
C if %error
C seton
C leave
C endif
* Se il record non esiste, accende errore e abbandona.
C if not %found
C seton
C leave

```

```

C endif
* Se il dati del record sono cambiati, accende errore e abbandona.
C if A1D<>A1DSV
C seton 6063
C unlock CNA1
C leave
C endif
* Trascrive dati dal video al record di database
C exsr W1DB
* Ricalca il record.
C update(e) UPD
* Se la chiave del record esiste gia', accende errore e abbandona.
C if %error
C seton 6064
C leave
C endif
* Esegue.
C enddo
C endsr
*-----

```

21.10.2.20. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1ED.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1ED.txt)

21.10.3. Funzionamento del programma CNA1ED.

21.10.3.1. Entra in visualizzazione (modo DSP). In DSP decodifica ma non controlla i valori dei campi dati.

21.10.3.2. Sia dal video guida che dal video dati in modo DSP, si può commutare con F7 in modifica (modo CHG).

21.10.3.3. In modo CHG, controlla gli errori formali terminali presenti sul video dati al momento della paginazione o dell'invio e li segnala ripresentando gli stessi dati con la nuova decodifica, con reverse image e position cursor sul campo e con un messaggio esplicativo dell'errore in ultima riga video.

21.10.3.4. Se il video porta dati validi, procede all'aggiornamento del record di database da cui li ha caricati.

21.10.3.5. Non alloca il record dati mentre lo presenta a video.

21.10.3.6. L'aggiornamento viene rifiutato e segnalato...

Se il file fisico che deve ricevere la modifica del record non può essere aperto per motivi di allocazione da parte di altri lavori, ad esempio se il file o la libreria sono in corso di salvataggio.

Se il record è allocato per update da altri.

Se il record è sparito tra la visualizzazione e la richiesta di update.

Se il record è modificato da altri tra la visualizzazione e la richiesta di update.

Se l'update del record provocherebbe una chiave doppia sulla chiave unica del file.

21.10.3.7. Dopo la modifica il programma presenta il record successivo, se richiesto PageDown o Entero, o il record precedente, se richiesto PageUp.

21.10.3.8. Identico per il resto alla versione precedente.

21.10.4. Sorgente del programma CNA1EX Rpgle.

21.10.4.1. Programma di servizio che permette di modificare i record del file fisico mantenendoli allocati mentre sono a video. Serve a controllare il programma CNA1ED contrapponendovisi come antagonista meno evoluto e creatore di allocazioni indesiderate.

```

* Claudio Neroni 06-03-2021 Creato.
* CNA1EX
* Anagrafica persone modifica alloca record a video
* Gestione elementare dimostrativa di un programma di modifica
* che mette a video dati un record
* e lo tiene erroneamente allocato sul database.
* Serve a mostrare che la versione ottimale CNA1ED riesce a competere
* correttamente anche con i programmi fatti male.
*-----
* Anagrafica persone logico codice
FCNA1 uf e disk
* Anagrafica persone modifica
FCNA1EW cf e workstn
* Tabella provincie
FCNT11 if e k disk
*-----
* Predisporre chiusura.
C seton 1r
* Legge il primo record del file.
C read A1
* Cicla intorno al video dati.
C do *hival

```



```

* Trascrive da database a video dati.
C exsr DBW1
* Decodifica video dati.
C exsr DEW1
* Emette video.
C exfmt W1
* Se richiesto, esce.
C 03 leave
* Se il video dati ha subito una digitazione di dati,
* ricalca il record di database con i dati del video.
C 33 exsr UPDB
* Se richiesto record precedente.
C if *in25
* Legge il record precedente.
C readp A1
* Se non trova.
C if %eof
* Si posiziona in fondo al file.
C *hival setgt A1
* Legge il record precedente (l'ultimo).
C readp A1
* Se non trova.
C endif
* Se richiesto record precedente.
C endif
* Se richiesto record successivo.
C if *in26
* Legge il record successivo.
C read A1
* Se non trova.
C if %eof
* Si posiziona all'inizio del file.
C 1 settl A1
* Legge il record successivo (il primo).
C read A1
* Se non trova.
C endif
* Se richiesto record successivo.
C endif
* Cicla intorno al video dati.
C enddo
*-----
* Trascrive dati dal record di database al video.
C DBW1 begsr
* Pulisce il video dati.
C clear W1
* Trascrive i campi dal database al video.
C eval W1CDPE=A1CDPE
C eval W1DEPE=A1DEPE
C eval W1PRNA=A1PRNA
C eval W1PRRE=A1PRRE
C endsr
*-----
* Trascrive dati dal video al record di database.
C W1DB begsr
* Pulisce il record dati.
C clear A1
* Trascrive i campi dal database al video.
C eval A1CDPE=W1CDPE
C eval A1DEPE=W1DEPE
C eval A1PRNA=W1PRNA
C eval A1PRRE=W1PRRE
C endsr
*-----
* Ricalca il record di database con i dati del video.
C UPDB begsr
* Trascrive dati dal video al record di database.
C exsr W1DB
* Ricalca.
C update A1
C endsr
*-----
* Decodifica video dati.
C DEW1 begsr
* Decodifica provincia di nascita.
C clear W1PRNAD
C if W1PRNA<>*blank
C eval W1PRNAD=*all'?'
C W1PRNA chain T1
C if %found
C eval W1PRNAD=T1DEPR
C endif
C endif
* Decodifica provincia di residenza.

```

```

C clear W1PRRED
C if W1PRRE<>*blank
C eval W1PRRED=*all'?'
C W1PRRE chain T1
C if %found
C eval W1PRRED=T1DEPR
C endif
C endif
C endsr
C *-----

```

21.10.4.2. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1EX.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1EX.txt)

21.11. Gestione anagrafica completa (visualizzazione, modifica, cancellazione e aggiunta) con doppio video guida, decodifica anche a richiesta e controllo errori (oggetti CNA1F\*).

21.11.1. Sorgente commentato del video CNA1FW Dspf. Ottenuto modificando la versione CNA1EW.

21.11.1.1. Le aggiunte e le modifiche riguardano l'introduzione delle funzioni di aggiunta, cancellazione e sola decodifica.

```

* Claudio Neroni 17-03-2021 Creato per Corso.
* CNA1FD
* Anagrafica persone dsp-chg-dlt-add
A DSPSI2(24 80 *DS3)
A CHGINPDFT
A INDARA
A INDTXT(44 'Protezione campi')
A INDTXT(45 'Aggiunta record')
A ERRSFL
A R W01
A TEXT('Video Guida per codice')
A BLINK
A CA03(03 'Fine')
A CF07(07 'Visualizzazione/Modifica')
A CF09(09 'Aggiunta')
A CF11(11 'PerDescrizione')
A PAGEUP(25 'Precedente')
A PAGEDOWN(26 'Successivo')
A 1 32'Anagrafica persona'
A DSPATR(UL)
A DSPATR(HI)
A WWMOD 3 O 1 76TEXT('Modalita''')
A 2 38'Guida'
A DSPATR(HI)
A 4 10'Codice persona:'
A W0CDPE R B 4 27REFFLD(A1CDPE CNA1)
A DSPATR(CS)
A DSPATR(HI)
A 23 2'F3=Fine'
A 44 23 11'F7=Chg'
A N44 23 11'F7=Dsp'
A +2'F9=Add'
A +2'F11=xDes'
A +2'PagDw=Succ'
A +2'PagUp=Prec'
A R W02
A TEXT('Video Guida per descrizione')
A BLINK
A CA03(03 'Fine')
A CF07(07 'Visualizzazione/Modifica')
A CF09(09 'Aggiunta')
A CF11(11 'PerCodice')
A PAGEUP(25 'Precedente')
A PAGEDOWN(26 'Successivo')
A 1 32'Anagrafica persona'
A DSPATR(UL)
A DSPATR(HI)
A WWMOD R O 1 76REFFLD(WWMOD *SRC)
A 2 38'Guida'
A DSPATR(HI)
A 3 5'Descrizione persona:'
A W0DEPE R B 3 27REFFLD(A1DEPE CNA1)
A DSPATR(CS)
A DSPATR(HI)
A 4 10'Codice persona:'
A W0CDPE R B 4 27REFFLD(A1CDPE CNA1)
A DSPATR(CS)
A DSPATR(HI)
A 23 2'F3=Fine'
A 44 23 11'F7=Chg'
A N44 23 11'F7=Dsp'
A +2'F9=Add'
A +2'F11=xCod'

```

|   |        |         |   |        |                                      |
|---|--------|---------|---|--------|--------------------------------------|
| A |        |         |   |        | +2'PagDw=Succ'                       |
| A |        |         |   |        | +2'PagUp=Prec'                       |
| A |        | R W1    |   |        |                                      |
| A |        |         |   |        | TEXT('Video Dati')                   |
| A |        |         |   |        | BLINK                                |
| A |        |         |   |        | CA03(03 'Fine')                      |
| A |        |         |   |        | CF04(04 'Decodifica')                |
| A | N45    |         |   |        | CF07(07 'Visualizzazione/Modifica')  |
| A |        |         |   |        | CA12(12 'Guida')                     |
| A |        |         |   |        | CA23(23 'Cancellazione')             |
| A | N45    |         |   |        | PAGEUP(25 'Precedente')              |
| A | N45    |         |   |        | PAGEDOWN(26 'Successivo')            |
| A |        |         |   |        | CHANGE(33)                           |
| A |        |         |   | 1 32   | 'Anagrafica persona'                 |
| A |        |         |   |        | DSPATR(UL)                           |
| A |        |         |   |        | DSPATR(HI)                           |
| A |        | WWMOD   | R | O 1 76 | REFFLD(WWMOD *SRC)                   |
| A | 41N45  |         |   | 2 33   | 'Dati per codice'                    |
| A |        |         |   |        | DSPATR(HI)                           |
| A | 42N45  |         |   | 2 31   | 'Dati per descrizione'               |
| A |        |         |   |        | DSPATR(HI)                           |
| A |        |         |   | 4 10   | 'Codice persona:'                    |
| A |        | W1CDPE  | R | B 4 27 | REFFLD(A1CDPE CNA1)                  |
| A | 60 61  |         |   |        | ERRMSG('Record occupato da altri')   |
| A | 60 62  |         |   |        | ERRMSG('Record cancellato da altri') |
| A | 60 63  |         |   |        | ERRMSG('Record modificato da altri') |
| A | 60 64  |         |   |        | ERRMSG('Chiave gia' presente nel f-  |
| A |        |         |   |        | ile')                                |
| A | 60 65  |         |   |        | ERRMSG('File non disponibile +       |
| A |        |         |   |        | per modifica e aggiunta')            |
| A | 60 66  |         |   |        | ERRMSG('Record non cancellabile')    |
| A | 60 71  |         |   |        | ERRMSG('Dato obbligatorio')          |
| A | 44     |         |   |        | DSPATR(PR)                           |
| A | N44    |         |   |        | DSPATR(CS)                           |
| A |        |         |   |        | DSPATR(HI)                           |
| A |        |         |   | 5 5    | 'Descrizione persona:'               |
| A |        | W1DEPE  | R | B 5 27 | REFFLD(A1DEPE CNA1)                  |
| A | 60 72  |         |   |        | ERRMSG('Dato obbligatorio')          |
| A | 60 73  |         |   |        | ERRMSG('Descrizione deve essere piu- |
| A |        |         |   |        | ' lunga del codice')                 |
| A | 44     |         |   |        | DSPATR(PR)                           |
| A | N44    |         |   |        | DSPATR(CS)                           |
| A |        |         |   |        | DSPATR(HI)                           |
| A |        |         |   |        | CHECK(LC)                            |
| A |        |         |   | 6 7    | 'Provincia nascita:'                 |
| A |        | W1PRNA  | R | B 6 27 | REFFLD(A1PRNA CNA1)                  |
| A | 60 74  |         |   |        | ERRMSG('Dato obbligatorio')          |
| A | 60 75  |         |   |        | ERRMSG('Manca in tabella')           |
| A | 44     |         |   |        | DSPATR(PR)                           |
| A | N44    |         |   |        | DSPATR(CS)                           |
| A |        |         |   |        | DSPATR(HI)                           |
| A |        | W1PRNAD | R | O 6 30 | REFFLD(T1DEPR CNT1)                  |
| A |        |         |   | 7 5    | 'Provincia residenza:'               |
| A |        | W1PRRE  | R | B 7 27 | REFFLD(A1PRRE CNA1)                  |
| A | 60 76  |         |   |        | ERRMSG('Manca in tabella')           |
| A | 44     |         |   |        | DSPATR(PR)                           |
| A | N44    |         |   |        | DSPATR(CS)                           |
| A |        |         |   |        | DSPATR(HI)                           |
| A |        | W1PRRED | R | O 7 30 | REFFLD(T1DEPR CNT1)                  |
| A |        |         |   | 23 2   | 'F3=Fine'                            |
| A |        |         |   |        | +2'F4=Decod'                         |
| A | 44N45  |         |   | 23 21  | 'F7=Chg'                             |
| A | N44N45 |         |   | 23 21  | 'F7=Dsp'                             |
| A |        |         |   |        | +2'F12=Guida'                        |
| A | 44N45  |         |   |        | +2'F23=Dlt'                          |
| A | N45    |         |   |        | +2'PagDw=Succ'                       |
| A | N45    |         |   |        | +2'PagUp=Prec'                       |

21.11.1.2. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1FW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1FW.txt)

21.11.2. Sorgente commentato del programma CNA1FD Rpgle. Ottenuto modificando la versione CNA1ED.

```

* Claudio Neroni 17-03-2021 Creato per Corso.
* CNA1FD
* Anagrafica persone dsp-chg-dlt-add
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk rename(A1:A11)
F infds(FIDS1)
* Anagrafica persone logico descrizione
FCNA12 if e k disk rename(A1:A12)
F infds(FIDS2)
* Tabella provincie
FCNT11 if e k disk

```

```
* Anagrafica persone dsp-chg-dlt-add
FCNA1FW cf e workstn
```

#### 21.11.2.1. Permette anche l'aggiunta record al fisico per update.

```
* Anagrafica persone fisico
FCNA1 uf a e disk rename(A1:UPD) usropn
*-----
* Struttura dati del file dati.
D A1D e ds extname(CNA1:A1)
* Salvataggio della Struttura dati del file dati.
D A1DSV e ds extname(CNA1:A1) prefix(SV)
*-----
* Struttura dati del video dati.
D W1D e ds extname(CNA1EW:W1)
* Salvataggio della Struttura dati del video dati.
D W1DSV e ds extname(CNA1EW:W1) prefix(SV)
*-----
* File information data structure
D FIDS1 ds
* Relative record number in data memmber
D FIDS1RR 397 400b 0
* File information data structure
D FIDS2 ds
* Relative record number in data memmber
D FIDS2RR 397 400b 0
*-----
* Predisporre chiusura.
C seton 1r
* Assume di usare video guida per codice.
C seton 41
C setoff 42
* Predisporre protezione campi.
C seton 44
* Gestisce video guida.
C exsr SRW0
*-----
* Gestisce video guida.
C SRW0 begsr
* Definisce chiave per codice.
C K1 klist
C kfld W0CDPE
* Definisce chiave per descrizione.
C K2 klist
C kfld W0DEPE
C kfld W0CDPE
* Pulisce video guida.
C clear W01
C clear W02
* Cicla intorno al video guida.
C do *hival
```

#### 21.11.2.2. Per utilizzare il video dati, oltre che in DSP e CHG anche in ADD, utilizza l'indicatore 45 e quindi qui lo imposta per DSP-CHG.

```
* Annota non aggiunta.
C setoff 45
* Prepara modo.
C 44 eval WWMOD='DSP'
C N44 eval WWMOD='CHG'
* Emette video guida.
C 41 exfmt W01
C 42 exfmt W02
* Se richiesta fine, abbandona.
C 03 leave
* Se richiesto switch tra visualizzazione e modifica,
* la esegue e ricicla.
C if *in07
C *IN44 comp *off 44
C iter
C endif
```

#### 21.11.2.3. Se dal video guida è richiesta aggiunta, esegue la nuova subroutine di aggiunta facendo ancora uso del video dati opportunamente commmutato.

```
* Se richiesta aggiunta.
C if *in09
* Esegue aggiunta.
C exsr SRW1A
* Se richiesta fine, abbandona.
C 03 leave
* Ricicla.
C iter
* Se richiesta aggiunta.
C endif
* Se richiesto switch tra guida per codice e guida per descrizione,
```

```

* commuta la scelta del video guida e ricicla.
C if *in11
C if *in41
C setoff
C seton
C else
C seton
C setoff
C endif
C iter
C endif
* Se richiesto PagUp.
C if *in25
* Si posiziona prima della chiave di ricerca.
C 41K1 setll A11
C 42K2 setll A12
* Legge all'indietro.
C 41 readp A11
C 42 readp A12
* Se non trova record, ritenta a partire dal fondo.
C if %eof
C 41*hival setgt A11
C 42*hival setgt A12
C 41 readp A11
C 42 readp A12
C endif
* Se richiesto PagUp o Enter,
C else
* Si posiziona prima della chiave di ricerca.
C 41K1 setll A11
C 42K2 setll A12
* Legge in avanti.
C 41 read A11
C 42 read A12
* Se richiesto...
C endif
* Se non trova record, ricicla.
C if %eof
C iter
C endif
* Gestisce video dati.
C exsr SRW1
* Se richiesta fine, abbandona.
C 03 leave
* Cicla intorno al video guida.
C enddo
C endsr
*-----
* Gestisce video dati.
* Va chiamata con un record gia' in canna.
C SRW1 begsr
* Cicla intorno al video dati.
C do *hival
* Se non corre errore.
C if not *in60
* Trascrive dati dal record di database al video.
C exsr DBW1
* Decodifica video dati.
C exsr DEW1
* Se non corre errore.
C endif
* Se corre errore, scrive base per errore.
C if *in60
C setoff
C write W1
C seton
C endif
* Prepara modo.
C 44 eval WWMOD='DSP'
C N44 eval WWMOD='CHG'
* Emette video dati.
C exfmt W1
* Spegne tutti gli errori.
C exsr OFF
* Se richiesta fine, abbandona.
C 03 leave
* Se richiesta Guida, abbandona.
C 12 leave
* Se richiesto switch tra visualizzazione e modifica,
* la esegue e ricicla.
C if *in07
C *IN44 comp *off
C iter
C endif

```

21.11.2.4. Gestisce la richiesta di decodifica estemporanea senza passare alla successiva esecuzione che si otterrebbe con l'invio o la paginazione.

```

* Se corre decodifica.
C if *in04
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C exsr CTW1
* Finge errore per non ricaricare da database.
C seton
* Ricicla.
C iter
* Se corre decodifica.
C endif

```

60

21.11.2.5. Durante la visualizzazione (modo DSP), gestisce la cancellazione del record corrente.

```

* Se corre cancellazione.
C if *in23
* Esegue cancellazione.
C exsr DLDB
* Se errore, ricicla.
C 60 iter
* Se corre cancellazione.
C endif
* Se corre modifica.
C if not *in44
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C exsr CTW1
* Se errore, ricicla.
C 60 iter
* Se il video e' stato modificato.
C if W1D<>W1DSV
* Esegue update.
C exsr UPDB
* Se errore, ricicla.
C 60 iter
* Se il video e' stato modificato.
C endif
* Se corre modifica.
C endif
* Se richiesto PagUp.
C if *in25
* Si posiziona prima della chiave ultima vista.
C 41K1 setll A11
C 42K2 setll A12
* Legge all'indietro.
C 41 readp A11
C 42 readp A12
* Se richiesto PagDw o Enter,
C else
* Si posiziona oltre la chiave ultima vista.
C 41K1 setgt A11
C 42K2 setgt A12
* Legge in avanti.
C 41 read A11
C 42 read A12
* Se richiesto...
C endif
* Se non trova record, pulisce chiave accantonata e abbandona.
C if %eof
C clear
C clear
C leave
C endif
* Cicla intorno al video dati.
C enddo
C endsr
*-----
* Trascrive dati dal record di database al video.
C DEW1 begsr

```

21.11.2.6. Dedico una subroutine separata all'accantonamento delle chiavi per poterla riutilizzare anche in aggiunta record.

```

* Accantona la chiave nel video guida.
C exsr DBW1KK
* Trascrive i campi dal record di database al video dati.
C clear
C eval W1CDPE=A1CDPE
C eval W1DEPE=A1DEPE
C eval W1PRNA=A1PRNA

```

```

C eval W1PRRE=A1PRRE
C * Salva i dati del record letto.
C eval A1DSV=A1D
C * Salva i dati del video.
C eval W1DSV=W1D
C endsr
C -----
C * Decodifica video dati.
C DEW1 begsr
C * Decodifica provincia di nascita.
C clear W1PRNAD
C if W1PRNA<>*blank
C eval W1PRNAD=*all'?'
C W1PRNA chain T1
C if %found
C eval W1PRNAD=T1DEPR
C endif
C endif
C * Decodifica provincia di residenza.
C clear W1PRRED
C if W1PRRE<>*blank
C eval W1PRRED=*all'?'
C W1PRRE chain T1
C if %found
C eval W1PRRED=T1DEPR
C endif
C endif
C endsr
C -----
C * Spegne tutti gli errori.
C OFF begsr
C move *off OFF31 31
C movea OFF31 *in(60)
C endsr
C -----
C * Controlla video dati.
C CTW1 begsr
C * Spegne tutti gli errori.
C exsr OFF
C * Esegue.
C do
C * Codice obbligatorio.
C if W1CDPE=*blank
C seton 6071
C leave
C endif
C * Descrizione obbligatoria.
C if W1DEPE=*blank
C seton 6072
C leave
C endif
C * Descrizione piu' corta del codice.
C if %len(%trim(W1DEPE))<=%len(%trim(W1CDPE))
C seton 6073
C leave
C endif
C * Provincia di nascita obbligatoria.
C if W1PRNA=*blank
C seton 6074
C leave
C endif
C * Provincia di nascita manca in tabella.
C if W1PRNAD=*all'?'
C seton 6075
C leave
C endif
C * Provincia di residenza manca in tabella.
C if W1PRRED=*all'?'
C seton 6076
C leave
C endif
C * Esegue.
C enddo
C endsr
C -----
C * Trascrive dati dal video al record di database
C W1DB begsr
C eval A1CDPE=W1CDPE
C eval A1DEPE=W1DEPE
C eval A1PRNA=W1PRNA
C eval A1PRRE=W1PRRE
C endsr
C -----
C * Ricalca database.

```

```

C UPDB begsr
* Esegue.
C do

```

21.11.2.7. Dedico una subroutine separata all'apertura del database per poterla riutilizzare anche in aggiunta e cancellazione record.

```

* Apre database.
C exsr OPDB
* Se errore in apertura, abbandona.
C 60 leave
* Recupera il record fisico per ricalcarlo.
C 41FIDS1RR chain(e) UPD
C 42FIDS2RR chain(e) UPD
* Se il record e' occupato, accende errore e abbandona.
C if %error
C seton
C leave 6061
C endif
* Se il record non esiste, accende errore e abbandona.
C if not %found
C seton
C leave 6062
C endif
* Se il dati del record sono cambiati, accende errore e abbandona.
C if A1D<>A1DSV
C seton
C unlock CNA1
C leave
C endif
* Trascrive dati dal video al record di database
C exsr W1DB
* Ricalca il record.
C update(e) UPD
* Se la chiave del record esiste gia', accende errore e abbandona.
C if %error
C seton
C leave 6064
C endif
* Esegue.
C enddo
C endsr
*-----

```

21.11.2.8. Aggiungo la subroutine di gestione dell'aggiunta record.

```

* Gestisce video dati in aggiunta record.
C SRW1A begsr
* Assume campi agibili.
C setoff
C 44
* Assume aggiunta.
C seton
C 45
* Cicla intorno a video dati.
C do *hival
* Se non corre errore.
C if not *in60
* Pulisce video dati.
C clear
C W1
* Decodifica video dati.
C exsr DEW1
* Se non corre errore.
C endif
* Se corre errore, scrive base per errore.
C if *in60
C setoff
C write W1
C seton
C 60
C endif
* Prepara modo.
C eval WWMOD='ADD'
* Emette video dati.
C exfmt W1
* Spegne tutti gli errori.
C exsr OFF
* Se richiesta fine, abbandona.
C 03 leave
* Se richiesta Guida, abbandona.
C 12 leave
* Se corre decodifica.
C if *in04
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C exsr CTW1
* Finge errore per non pulire video.
C seton
C 60

```



```

* Ricicla.
C iter
* Se corre decodifica.
C endif
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C exsr CTW1
* Se errore, ricicla.
C 60 iter
* Scrive database.
C exsr WRDB
* Se errore, ricicla.
C 60 iter
* Cicla intorno a video dati.
C enddo
C endsr
*-----

```

#### 21.11.2.9. Aggiungo la subroutine di scrittura record.

```

* Scrive database.
C WRDB begsr
* Esegue.
C do
* Apre database.
C exsr OPDB
* Se errore in apertura, abbandona.
C 60 leave
* Trascrive dati dal video al record di database
C exsr W1DB
* Accantona la chiave database nei campi dei video guida.
C exsr DBW1KK
* Aggiunge il record.
C write(e) UPD
* Se la chiave del record esiste gia', accende errore e abbandona.
C if %error
C seton
C leave 6064
C endif
* Esegue.
C enddo
C endsr
*-----

```

#### 21.11.2.10. Aggiungo la subroutine di apertura del file fisico di update e add.

```

* Apre database.
C OPDB begsr
* Se il file fisico per update non e' aperto.
C if not %open(CNA1)
* Tenta l'apertura del file.
C open(e) CNA1
* Se errore in apertura, accende errore.
C if %error
C seton
C leave 6065
C endif
* Se il file fisico per update non e' aperto.
C endif
C endsr
*-----

```

#### 21.11.2.11. Aggiungo la subroutine accantonamento delle chiavi nel video guida.

```

* Accantona la chiave database nei campi dei video guida.
C DBW1KK begsr
C eval W0CDPE=A1CDPE
C eval W0DEPE=A1DEPE
C endsr
*-----

```

#### 21.11.2.12. Aggiungo la subroutine di cancellazione record.

```

* Cancella database.
C DLDB begsr
* Esegue.
C do
* Apre database.
C exsr OPDB
* Se errore in apertura, abbandona.
C 60 leave
* Recupera il record fisico per ricalcarlo.
C 41FIDS1RR chain(e) UPD
C 42FIDS2RR chain(e) UPD
* Se il record e' occupato, accende errore e abbandona.
C if %error
C seton
C leave 6061
C endif
C endsr

```

```

C endif
* Se il record non esiste, accende errore e abbandona.
C if not %found
C seton 6062
C leave
C endif
* Se il dati del record sono cambiati, accende errore e abbandona.
C if A1D<>A1DSV
C seton 6063
C unlock CNA1
C leave
C endif
* Cancella il record.
C delete(e) UPD
* Se la cancellazione fallisce, accende errore e abbandona.
C if %error
C seton 6066
C leave
C endif
* Esegue.
C enddo
C endsr
*-----

```

21.11.2.13. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1FD.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1FD.txt)

21.11.3. Funzionamento del programma CNA1FD.

21.11.3.1. Entra in visualizzazione (modo DSP). In DSP decodifica ma non controlla i valori dei campi dati.

21.11.3.2. Solo dal video guida si può chiedere l'aggiunta di un nuovo record nel database tramite F9 che presenta vuoto il video dati da riempire (modo ADD).

21.11.3.3. In aggiunta controlla gli errori formali terminali presenti sul video dati al momento dell'invio e li segnala ripresentando gli stessi dati con la nuova decodifica, con reverse image e position cursor sul campo e con un messaggio esplicativo dell'errore in ultima riga video, esattamente come in modifica.

21.11.3.4. Se il video porta dati validi, procede alla scrittura di un nuovo record di database.

21.11.3.5. L'aggiornamento viene rifiutato e segnalato...

Se il file fisico che deve ricevere il nuovo record non può essere aperto per motivi di allocazione da parte di altri lavori, ad esempio se il file o la libreria sono in corso di salvataggio.

Se l'update del record provocherebbe una chiave doppia sulla chiave unica del file.

21.11.3.6. Dopo l'aggiunta il programma presenta di nuovo il video dati vuoto, pronto all'aggiunta di un altro record.

21.11.3.7. Dal video dati, sia in modo CHG che in modo ADD, si può chiedere decodifica e controllo formale dei dati a video tramite F4. Avviene l'immediata riemissione della videata decodificata con eventuali messaggi di errore generati dal controllo ma senza in nessun modo modificare o aggiungere record nel database.

21.11.3.8. Dal video dati in modo DSP, si può chiedere la cancellazione del record visualizzato tramite F23. In tal modo, avviene l'immediata cancellazione del record dal database.

21.11.3.9. La cancellazione viene rifiutata e segnalata...

Se il file fisico che deve ricevere la cancellazione del record non può essere aperto per motivi di allocazione da parte di altri lavori, ad esempio se il file o la libreria sono in corso di salvataggio.

Se il record è allocato per update da altri.

Se il record è sparito tra la visualizzazione e la richiesta di cancellazione.

Se il record è modificato da altri tra la visualizzazione e la richiesta di cancellazione.

21.11.3.10. Identico per il resto alla versione precedente.

## 21.12. Altre implementazioni della Gestione anagrafica completa.

21.12.1. Secondo il tempo a disposizione, si può trattare l'inserimento nello schema fin qui maturato di ulteriori prestazioni aggiuntive, in gran parte relative alla navigazione sensibile al contesto tra tutti gli oggetti elaborativi presenti nel database, in parte rivolte alla "industrializzazione" degli schemi di programmazione.

21.12.1.1. Fornire elenchi a pieno video dei soggetti richiamabili dai video guida.

21.12.1.2. Introdurre ricerche di parole nelle descrizioni dei soggetti anagrafici e tabellari.

- 21.12.1.3. Presentazione a richiesta con F2/F14 di un elenco in finestra dei valori digitabili nel campo ove giace il cursore 5250.
- 21.12.1.4. Presentazione a richiesta con F1 di un testo di aiuto in finestra per il campo, il record video o il file video ove giace il cursore 5250.
- 21.12.1.5. Chiamabilità in cascata delle manutenzioni delle tabelle, delle anagrafiche o, addirittura, dei documenti, sempre a partire dal campo ove giace il cursore 5250.
- 21.12.1.6. Estrapolazione nei /COPY delle routine che si ripresentano inalterate in tutti i programmi. Naturalmente dopo una revisione degli schemi dei programmi standard orientata a questo scopo.
- 21.12.1.7. Estrapolazione delle funzioni riutilizzabili, sia tecniche che applicative, in programmi richiamabili tramite call, call procedure, moduli, programmi di servizio, bind directory, funzioni sql,...
- 21.12.2. Per alcune di queste e per altre ancora la convenienza si ha quando possono essere fatte come preparazione ad uno sviluppo che costi almeno qualche anno uomo, poiché buoni schemi velocizzano il lavoro ripetitivo quando vanno ripetuti su centinaia di anagrafiche e sulle corrispondenti tabelle.
- 21.12.3. Tuttavia ogni implementazione tecnica e funzionale che si innesca sugli schemi in uso porta sempre ad un beneficio che può essere propagato anche a richiesta e non solo a tappeto, quindi non solo in sviluppo ma anche in manutenzione.
- 21.12.4. Occorre cioè costruirsi un repertorio di schemi e di soluzioni tecniche in grado di soddisfare le richieste in modo progressivo, compatibile anche col materiale già in uso o ad esso sostituibile in modo migliorativo e tale da non rinunciare alle prestazioni già consolidate nel vecchio.

## 22. Subfile e sue tipologie

- 22.1. Questo capitolo, come altri, presenta un discreto numero di esempi di coppie di display file e programmi di gestione dei vari tipi di subfile. Consiglio di percorrerli dal primo in avanti senza saltare passi perché le note sul secondo spiegano cosa è stato aggiunto al primo per diventare il secondo e così via. Cioè ogni versione porta solo le spiegazioni che la differenziano da quella immediatamente precedente.
- 22.2. L'interfaccia che il sistema fornisce per i Display File contiene un supporto specifico per la gestione di elenchi di record in
  - 22.2.1. visualizzazione,
  - 22.2.2. visualizzazione per scelta singola o multipla,
  - 22.2.3. aggiunta o modifica di record completi sull'elenco.
- 22.3. Non prenderò in considerazione la terza tipologia perché chiede un comportamento faticoso all'utente finale: caricare molti soggetti e attendere fino all'invio di fine caricamento per avere la diagnosi degli eventuali errori anche sui soggetti caricati per primi. Questo chiede, in caso di errore, di riportare l'attenzione continuamente indietro a documenti già visti e già inputati, frazionando e moltiplicando l'attenzione da porre ad ogni soggetto.
- 22.4. Tratterò invece la prima tipologia (visualizzazione) e la seconda (visualizzazione per scelta) in tre versioni.
  - 22.4.1. Subfile a riempimento totale.
    - 22.4.1.1. L'elenco viene riempito subito con tutto il contenuto candidato alla visione.
    - 22.4.1.2. È disponibile il posizionamento a partire da una chiave a scelta.
    - 22.4.1.3. Le pagine successive già in memoria sono navigabili paginando senza ricaricamento.
    - 22.4.1.4. Dopo un posizionamento, paginando non sono più caricabili i record precedenti la chiave.
  - 22.4.2. Subfile a riempimento graduale.
    - 22.4.2.1. L'elenco viene presentato solo con la prima pagina di quanto è candidato alla visione.
    - 22.4.2.2. È disponibile il posizionamento a partire da una chiave a scelta.
    - 22.4.2.3. Le pagine successive si aggiungono una alla volta paginando in avanti.
    - 22.4.2.4. Le pagine già caricate restano navigabili paginando in entrambi i sensi.
    - 22.4.2.5. Dopo un posizionamento, paginando non sono più caricabili i record precedenti la chiave.
  - 22.4.3. Subfile "quadrato" o a pagina singola.
    - 22.4.3.1. L'elenco viene presentato una pagina alla volta e la paginazione ricarica sempre la pagina nuova in entrambi i sensi.
    - 22.4.3.2. È disponibile il posizionamento a partire da una chiave a scelta.
    - 22.4.3.3. Dopo un posizionamento, paginando sono raggiungibili anche i record precedenti la chiave.

## 23. Subfile di visualizzazione

### 23.1. Elenco anagrafico su subfile totale (oggetti CNA1S1\*).

23.1.1. Sorgente commentato del video CNA1S1W Dspf. Scritto ex novo.

```
* Claudio Neroni 22-03-2021 Creato per Corso.
* CNA1S1W
* Sfl totale
*-----
A DSPSIZ(24 80 *DS3)
A CHGINPDEF
A INDARA
A ERRSFL
A INDTXT(52 'Subfile Clear')
A INDTXT(53 'Subfile Display Control')
A INDTXT(54 'Subfile Display')
A INDTXT(56 'Subfile End')
*-----
```

23.1.1.1. T1 (Testata) definisce le intestazioni dell'elenco.

```
A R T1
A TEXT('Testata')
```

23.1.1.2. La scrittura di un qualunque record elimina preliminarmente dal video tutti i record che lo occupavano precedentemente. OVERLAY, invece, cancella preliminarmente dal video solo i record che occupano almeno una riga video interferente col nuovo.

23.1.1.3. Tutti i record del display file, tranne S1 che va a video solo tramite C1, sono corredati da OVERLAY.

```
A OVERLAY
A 1 27'Elenco persone'
A DSPATR(HI)
A DSPATR(UL)
A 2 2'Codice Descrizione Pr Pr' -
A DSPATR(HI)
A 3 2'person persona Na Re' -
A DSPATR(HI)
A DSPATR(UL)
```

23.1.1.4. S1 (Riga) definisce la riga dell'elenco.

23.1.1.5. S1 e C1 sono interdipendenti. Pagine di S1 vanno a video solo comandate da C1 tramite la keyword SFLDSP di C1. Si osservi che S1 si usa solo nelle istruzioni RPGLE WRITE, CHAIN e UPDATE con uso del Relative Record Number.

```
A R S1 SFL
A TEXT('Riga')
A SICDPE R O 4 2REFFLD(A1CDPE CNA1)
A S1DEPE R O +1REFFLD(A1DEPE CNA1)
A S1PRNA R O +1REFFLD(A1PRNA CNA1)
A S1PRRE R O +1REFFLD(A1PRRE CNA1)
*-----
```

23.1.1.6. C1 (Controllo) definisce le caratteristiche dell'elenco, i comandi, i campi di piede.

23.1.1.7. S1 e C1, come detto, sono interdipendenti.

```
A R C1 SFLCTL(S1)
A TEXT('Controllo')
A OVERLAY
```

23.1.1.8. SFLSIZ dispone quante righe dati stanno nel subfile all'apertura del display file. In realtà, un valore qualsiasi maggiore di SFLPAG permette al sistema di allargare gradualmente lo spazio assegnato al subfile man mano che viene riempito.

```
A SFLSIZ(0010)
```

23.1.1.9. SFLPAG dispone quante righe dati stanno in una pagina di subfile a video.

```
A SFLPAG(0005)
```

23.1.1.10. Se l'indicatore è acceso, pulisce in memoria tutte le righe del subfile.

```
A 52 SFLCLR
```

23.1.1.11. Se l'indicatore è acceso, emette il controllo sul video.

```
A 53 SFLDSPCTL
```

23.1.1.12. Se l'indicatore è acceso, emette le righe sul video.

```
A 54 SFLDSP
```

23.1.1.13. Il condizionamento negativo con un qualunque indicatore sempre spento provoca la gestione automatica del "+" in basso a destra delle righe del subfile. Se una pagina mostra il "+", i dati continuano in una pagina successiva raggiungibile con PageDown.

|       |        |
|-------|--------|
| A N56 | SFLEND |
|-------|--------|

23.1.1.14. Autorizza la digitazione dei comandi elencati

|   |                 |
|---|-----------------|
| A | CA03(03 'Fine') |
| A | CF06(06 'Da')   |

23.1.1.15. Permette la digitazione di una chiave di partenza, utilizzata poi dal command F6-

|        |        |   |       |                                  |
|--------|--------|---|-------|----------------------------------|
| A      | CICDPE | R | B 22  | 2REFFLD(A1CDPE CNA1)             |
| A      |        |   |       | TEXT('Chiave di posizionamento') |
| A      |        |   |       | DSPATR(HI)                       |
| A      |        |   |       | DSPATR(CS)                       |
| A      |        |   |       | DSPATR(PC)                       |
| A      |        |   | 22    | 9'<--dal codice'                 |
| A      |        |   | 23    | 2'F3=Fine'                       |
| A      |        |   | 23 11 | 'F6=Da'                          |
| *----- |        |   |       |                                  |

23.1.1.16. B1 (Bianco) dichiara "Nessun record" quando l'elenco è vuoto e fa sparire dal video il cadavere della vecchia emissione di righe del subfile.

|        |      |                                 |
|--------|------|---------------------------------|
| A      | R B1 |                                 |
| A      |      | TEXT('Bianco')                  |
| A      |      | OVERLAY                         |
| A      | 5    | 4' (Nessun record da mostrare)' |
| *----- |      |                                 |

23.1.1.17. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S1W.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S1W.txt)

23.1.2. Sorgente commentato del programma CNA1S1 Rpgle. Scritto ex novo.

23.1.2.1. \*FREE nelle prime cinque colonne della prima riga permette di mescolare la minutazione a tracciato libero e quella a tracciato posizionale.

|        |                                               |
|--------|-----------------------------------------------|
| *FREE  | * Claudio Neroni 22-03-2021 Creato per Corso. |
| *      | CNA1S1                                        |
| *      | Sfl totale                                    |
| *----- |                                               |

23.1.2.2. Vista logica per codice sul file dati.

|                                    |    |   |        |
|------------------------------------|----|---|--------|
| * Anagrafica persone logico codice |    |   |        |
| FCNA11                             | if | e | k disk |

23.1.2.3. Display file del subfile a riempimento totale.

|              |    |   |         |
|--------------|----|---|---------|
| * Sfl totale |    |   |         |
| FCNA1S1W     | cf | e | workstn |

23.1.2.4. Definisce il Relative Record Number che permette di trattare le righe di subfile come un file diretto in memoria.

|                                                           |                 |
|-----------------------------------------------------------|-----------------|
| * Associa il relative record number al record di subfile. |                 |
| F                                                         | sfile(S1:S1RRN) |

23.1.2.5. Evidenzia le informazioni che il programma conosce sul display file. Inutile ai fini delle elaborazioni strettamente necessarie. Inserito per eventuali verifiche e futuri usi.

|                                         |               |
|-----------------------------------------|---------------|
| * Display Specific Feedback Information |               |
| F                                       | infds(DSPFBK) |
| *-----                                  |               |

23.1.2.6. Info conosciute dal programma sul display file in formato free. Inutile ai fini delle elaborazioni strettamente necessarie. Inserito per eventuali verifiche e futuri usi.

23.1.2.7. Con il /COPY si apre la possibilità di spezzare un programma in più parti, ciascuna in un sorgente separato e riutilizzabile, riunite al momento della compilazione.

|                                         |  |
|-----------------------------------------|--|
| * Display Specific Feedback Information |  |
| /COPY CORSO,DSPFBK                      |  |
| *-----                                  |  |

23.1.2.8. Definisce il Relative Record Number di cui sopra.

|                                       |   |   |   |
|---------------------------------------|---|---|---|
| * Relative record number del subfile. |   |   |   |
| D S1RRN                               | S | 5 | 0 |
| *-----                                |   |   |   |

23.1.2.9. Prenotazione di chiusura del programma a fine calcolo.

|                         |       |  |    |
|-------------------------|-------|--|----|
| * Predisporre chiusura. |       |  |    |
| C                       | seton |  | lr |

23.1.2.10. Loop intorno al video di controllo. Inizio

23.1.2.11. Riempie il subfile con tutti i record disponibili.

|                           |    |  |        |
|---------------------------|----|--|--------|
| * Cicla intorno al video. |    |  |        |
| C                         | do |  | *hival |

#### 23.1.2.12. Posizionamento della lettura della vista logica sulla chiave di partenza.

|                                              |        |       |    |
|----------------------------------------------|--------|-------|----|
| * Si posiziona prima della chiave richiesta. |        |       |    |
| C                                            | C1CDPE | setll | A1 |

#### 23.1.2.13. Azzeramento del contatore che fa da relative record number.

|                                                    |       |       |  |
|----------------------------------------------------|-------|-------|--|
| * Azzer il numero relativo di record di scrittura. |       |       |  |
| C                                                  | clear | S1RRN |  |

#### 23.1.2.14. Elimina preliminarmente il precedente contenuto del subfile.

|                                                    |        |      |                  |
|----------------------------------------------------|--------|------|------------------|
| * Pulisce il subfile.                              |        |      |                  |
| C                                                  | if     | 1=1  |                  |
| * Prenota la pulizia del subfile.                  |        |      |                  |
| C                                                  | seton  | 52   | sflclr           |
| * Inibisce visualizzazione di controllo e subfile. |        |      |                  |
| C                                                  | setoff | 5354 | sfldspctl sfldsp |
| * Emette il controllo per la sola pulizia.         |        |      |                  |
| C                                                  | write  | C1   |                  |
| * Sprenota la pulizia del subfile.                 |        |      |                  |
| C                                                  | setoff | 52   | sflclr           |
| * Pulisce il subfile.                              |        |      |                  |
| C                                                  | endif  |      |                  |

#### 23.1.2.15. Legge tutti i record del file e li carica nelle righe di subfile.

|                                                         |       |               |  |
|---------------------------------------------------------|-------|---------------|--|
| * Cicla sulla lettura del file per riempire il subfile. |       |               |  |
| C                                                       | do    | *hival        |  |
| * Legge il prossimo record.                             |       |               |  |
| C                                                       | read  | A1            |  |
| * Se non ce ne sono altri, abbandona lettura.           |       |               |  |
| C                                                       | if    | %eof          |  |
| C                                                       | leave |               |  |
| C                                                       | endif |               |  |
| * Incrementa il numero relativo di record di scrittura. |       |               |  |
| C                                                       | eval  | S1RRN +=1     |  |
| * Trascrive il record letto nella riga di subfile.      |       |               |  |
| C                                                       | eval  | S1CDPE=A1CDPE |  |
| C                                                       | eval  | S1DEPE=A1DEPE |  |
| C                                                       | eval  | S1PRNA=A1PRNA |  |
| C                                                       | eval  | S1PRRE=A1PRRE |  |
| * Scrive la riga di subfile.                            |       |               |  |
| C                                                       | write | S1            |  |
| * Cicla sulla lettura del file per riempire il subfile. |       |               |  |
| C                                                       | enddo |               |  |

#### 23.1.2.16. Visualizza il subfile con il contorno di intestazioni e piede.

#### 23.1.2.17. Se il subfile è vuoto, lo sostituisce con la dicitura di vuoto.

|                                                                  |            |       |           |
|------------------------------------------------------------------|------------|-------|-----------|
| * Emette il subfile.                                             |            |       |           |
| C                                                                | if         | 1=1   |           |
| * Prenota emissione del controllo.                               |            |       |           |
| C                                                                | seton      | 53    | sfldspctl |
| * Se subfile pieno, ne prenota emissione.                        |            |       |           |
| C                                                                | S1RRN comp | *zero | 54 sfldsp |
| * Scrive testata.                                                |            |       |           |
| C                                                                | write      | T1    |           |
| * Se subfile vuoto, sbianca area di visualizzazione del subfile. |            |       |           |
| C                                                                | N54 write  | B1    | sblblank  |
| * Emette controllo e, se pieno, subfile.                         |            |       |           |
| C                                                                | exfmt      | C1    |           |
| * Emette il subfile.                                             |            |       |           |
| C                                                                | endif      |       |           |

#### 23.1.2.18. Testa le condizioni di uscita, partendo da F3.

|                            |    |       |  |
|----------------------------|----|-------|--|
| * Se richiesto, abbandona. |    |       |  |
| C                          | 03 | leave |  |

#### 23.1.2.19. Se richiesto, ricicla per ricaricare il subfile cambiando eventualmente la chiave di partenza.

|                                         |    |      |  |
|-----------------------------------------|----|------|--|
| * Se richiesto posizionamento, ricicla. |    |      |  |
| C                                       | 06 | iter |  |

#### 23.1.2.20. A fine visualizzazione, abbandona.

|              |       |  |  |
|--------------|-------|--|--|
| * Abbandona. |       |  |  |
| C            | leave |  |  |

#### 23.1.2.21. Loop intorno al video di controllo. Fine

|                           |       |  |  |
|---------------------------|-------|--|--|
| * Cicla intorno al video. |       |  |  |
| C                         | enddo |  |  |
| *-----                    |       |  |  |

#### 23.1.2.22. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S1.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S1.txt)

### 23.1.3. Funzionamento del programma CNA1S1.

#### 23.1.3.1. Carica e visualizza l'elenco di tutti i record della vista logica.

23.1.3.2. La visualizzazione parte dall'inizio del subfile e la paginazione avanti e indietro è affidata al sistema.

23.1.3.3. Dopo il caricamento e durante la visualizzazione, il subfile totale non viene aggiornato nel contenuto e paginando vengono mostrate le righe con i valori ottenuti al momento del caricamento.

23.1.3.4. È possibile riposizionare e aggiornare la visualizzazione premendo F6. Si ricarica allora il subfile partendo dalla chiave indicata al piede del subfile.

## 23.2. Elenco anagrafico su subfile a riempimento graduale (oggetti CNA1S2\*).

23.2.1. Sorgente commentato del video CNA1S2W Dspf. Ottenuto modificando la versione CNA1S1W.

```
* Claudio Neroni 22-03-2021 Creato per Corso.
* CNA1S2W
* Sfl graduale
*-----
A DSPSIZ(24 80 *DS3)
A CHGINPDFT
A INDARA
A ERRSFL
A INDTXT(52 'Subfile Clear')
A INDTXT(53 'Subfile Display Control')
A INDTXT(54 'Subfile Display')
A INDTXT(56 'Subfile End')
*-----
A R T1
A TEXT('Testata')
A OVERLAY
A 1 27'Elenco persone'
A DSPATR(HI)
A DSPATR(UL)
A 2 2'Codice Descrizione Pr Pr' -
A DSPATR(HI)
A 3 2'person persona Na Re' -
A DSPATR(HI)
A DSPATR(UL)
*-----
A R S1
A SFL
A TEXT('Riga')
A SICDPE R O 4 2REFFLD(A1CDPE CNA1)
A S1DEPE R O +1REFFLD(A1DEPE CNA1)
A S1PRNA R O +1REFFLD(A1PRNA CNA1)
A S1PRRE R O +1REFFLD(A1PRRE CNA1)
*-----
A R C1
A SFLCTL(S1)
A TEXT('Controllo')
A OVERLAY
A SFLSIZ(0010)
A SFLPAG(0005)
A 52 SFLCLR
A 53 SFLDSPCTL
A 54 SFLDSP
A N56 SFLEND
A CA03(03 'Fine')
A CF06(06 'Da')
```

23.2.1.1. Intercetta PageDown per poter gestire il caricamento della pagina successiva quando necessaria..

```
A PAGEDOWN(26 'Pagina successiva')
```

23.2.1.2. Definisce il campo relative record number che permette di emettere a video la pagina di subfile caricata dal programma in risposta alla richiesta di PageDown.

```
A S1RRNTW 4 0H SFLRCDNBR(CURS0R)
A TEXT('RRN to view')
A C1CDPE R B 22 2REFFLD(A1CDPE CNA1)
A TEXT('Start code')
A DSPATR(HI)
A DSPATR(CS)
A DSPATR(PC)
A +1'<--dal codice'
A 23 2'F3=Fine'
A +2'F6=Da'
*-----
A R B1
A TEXT('Bianco')
A OVERLAY
A 5 4'(Nessun record da mostrare)'
*-----
```

23.2.1.3. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S2W.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S2W.txt)



### 23.2.2. Sorgente commentato del programma CNA1S2 Rpgle. Ottenuto modificando la versione CNA1S1.

```
*FREE * Claudio Neroni 22-03-2021 Creato per Corso.
* CNA1S2
* Sfl graduale
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
* Sfl graduale
FCNA1S2W cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds(DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
```

23.2.2.1. Non essendo possibile recuperarlo automaticamente, occorre annotare in una costante un valore uguale al subfile page scelto nel display file.

```
* Annota subfile page.
D SFLPAG c const(0005)
* Relative record number del subfile.
D S1RRN S 5 0
*-----
* Predispone chiusura.
C seton 1r
* Cicla intorno al video.
C do *hival
```

23.2.2.2. Condiziona lo svuotamento del subfile che non avviene quando richiesta pagina successiva.

```
* Se non corre richiesta di prossima pagina.
C if not *in26
* Si posiziona prima della chiave richiesta.
C C1CDPE setll A1
* Azzerà il numero relativo di record di scrittura.
C clear S1RRN
* Pulisce il subfile.
C if 1=1
* Prenota la pulizia del subfile.
```

23.2.2.3. Accende 56 per sflend.

```
C seton 52 56 sflclr sflend
* Inibisce visualizzazione di controllo e subfile.
C setoff 5354 sfldspctl sfldsp
* Emette il controllo per la sola pulizia.
C write C1
* Sprenota la pulizia del subfile.
C setoff 52 sflclr
* Pulisce il subfile.
C endif
```

23.2.2.4. Condiziona lo svuotamento del subfile che non avviene quando richiesta pagina successiva.

```
* Se non corre richiesta di prossima pagina.
C endif
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il prossimo record.
C read A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
```

23.2.2.5. Spegne 56 per sflend.

```
C setoff 56 sflend
C leave
C endif
* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
* Trascrive il record letto nella riga di subfile.
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRE=A1PRE
```

23.2.2.6. Prende nota dell'ultima chiave letta e caricata nel subfile per poter ripartire da dopo quella quando caricherà la pagina successiva.

```
* Salva ultima chiave letta.
C eval A1CDPESV=A1CDPE
C *like define A1CDPE A1CDPESV
* Scrive la riga di subfile.
C write S1
```

23.2.2.1. Prende nota dell'ultimo relative record number caricato nel subfile per poter ripartire da dopo quello quando caricherà la pagina successiva.

|                                                                   |                    |
|-------------------------------------------------------------------|--------------------|
| * Trascrive il relative record number corrente nell'ultimo usato. |                    |
| C                                                                 | eval S1RRNTW=S1RRN |

23.2.2.2. Se corre l'ultimo record della pagina di subfile in corso di caricamento, sospende il caricamento di record nel subfile.

|                                                         |                             |
|---------------------------------------------------------|-----------------------------|
| * Se fine pagina, abbandona.                            |                             |
| C                                                       | if %rem(S1RRN:SFLPAG)=*zero |
| C                                                       | leave                       |
| C                                                       | endif                       |
| * Cicla sulla lettura del file per riempire il subfile. |                             |
| C                                                       | enddo                       |

23.2.2.3. Caricata una pagina, tenta una lettura extra senza caricarla nel subfile al solo scopo di spegnere 56 per sflend. Poi rimette a posto il cursore di lettura del database.

|                                                                  |                      |
|------------------------------------------------------------------|----------------------|
| * Tenta una lettura oltre l'ultimo record messo su subfile.      |                      |
| C                                                                | read A1              |
| * Se non ce ne sono altri, spegne sflend.                        |                      |
| C                                                                | if %eof              |
| C                                                                | setoff 56 sflend     |
| C                                                                | endif                |
| * Si riposiziona.                                                |                      |
| C                                                                | AlCDPESV setgt A1    |
| * Emette il subfile.                                             |                      |
| C                                                                | if 1=1               |
| * Prenota emissione del controllo.                               |                      |
| C                                                                | seton 53 sfldspctl   |
| * Se subfile pieno, ne prenota emissione.                        |                      |
| C                                                                | S1RRN comp 54 sfldsp |
| * Emette intestazione.                                           |                      |
| C                                                                | write T1             |
| * Se subfile vuoto, ricopre area di visualizzazione del subfile. |                      |
| C N54                                                            | write B1 sflblank    |
| * Emette controllo e, se pieno, subfile.                         |                      |
| C                                                                | exfmt C1             |
| * Emette il subfile.                                             |                      |
| C                                                                | endif                |
| * Se richiesto, abbandona.                                       |                      |
| C                                                                | 03 leave             |

23.2.2.4. Se richiesta pagina successive, ricicla per caricare una pagina aggiuntiva.

|                                            |         |
|--------------------------------------------|---------|
| * Se richiesta pagina successiva, ricicla. |         |
| C                                          | 26 iter |
| * Se richiesto posizionamento, ricicla.    |         |
| C                                          | 06 iter |
| * Ricicla.                                 |         |
| C                                          | iter    |
| * Cicla intorno al video.                  |         |
| C                                          | enddo   |
| *-----                                     |         |

23.2.2.5. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S2.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S2.txt)

23.2.3. Funzionamento del programma CNA1S2.

23.2.3.1. Inizialmente, carica e visualizza dalla vista logica l'elenco di una sola pagina di record.

23.2.3.2. La visualizzazione parte dalla prima pagina del subfile. Quando la pagina successiva non è ancora caricata, la paginazione avanti è affidata al programma. Quando la pagina successiva è già caricata, la paginazione avanti è affidata al sistema che ripresenta le righe senza rileggere il database. La paginazione indietro è invece sempre affidata al sistema.

23.2.3.3. Le righe caricate nel subfile graduale contengono i valori presenti nelle corrispondenti righe di database al momento del loro caricamento. Ogni pagina contiene cioè i valori presenti nel database al momento della prima visualizzazione della pagina stessa. Ne segue che le singole pagine riflettono valori prelevati dal database in tempi diversi e successivi tra loro. Paginando all'indietro, per ogni pagina si rivedono i valori ottenuti al momento del caricamento di ciascuna pagina.

23.2.3.4. È possibile riposizionare e aggiornare la visualizzazione premendo F6. Si ricarica allora il subfile partendo dalla chiave indicata al piede del subfile.

23.2.3.5. Subfile end (+ in basso a destra), che dice "puoi paginare in avanti", viene gestito correttamente grazie ad una lettura in più dopo aver caricato la pagina provvisoriamente ultima del subfile.

### 23.3. Elenco anagrafico su subfile quadrato (oggetti CNA1S3\*).

23.3.1. Sorgente commentato del video CNA1S3W Dspf. Ottenuto modificando la versione CNA1S2W.

```

* Claudio Neroni 24-03-2021 Creato per Corso.
* CNA1S3W
* Sfl quadrato
*-----
A DSPSIZ(24 80 *DS3)
A CHGINPDFT
A INDARA
A ERRSFL
A INDTXT(52 'Subfile Clear')
A INDTXT(53 'Subfile Display Control')
A INDTXT(54 'Subfile Display')

```

23.3.1.1. Si aggiunge l'indicatore 55 per condizionare PageUp e si promuove l'indicatore 56 da solo subfile end a condizionamento di PageDown.

```

A INDTXT(55 'PageUp permesso')
A INDTXT(56 'PageDown permesso')
*-----
A R T1 TEXT('Testata')
A OVERLAY
A 1 27'Elenco persone'
A DSPATR(HI)
A DSPATR(UL)
A 2 2'Codice Descrizione' -
A Pr Pr'
A DSPATR(HI)
A 3 2'person persona' -
A Na Re'
A DSPATR(HI)
A DSPATR(UL)
*-----
A R S1 SFL
A TEXT('Riga')
A S1CDPE R O 4 2REFFLD(A1CDPE CNA1)
A S1DEPE R O +1REFFLD(A1DEPE CNA1)
A S1PRNA R O +1REFFLD(A1PRNA CNA1)
A S1PRRE R O +1REFFLD(A1PRRE CNA1)
*-----
A R C1 SFLCTL(S1)
A TEXT('Controllo')
A OVERLAY

```

23.3.1.2. Subfile size e subfile page hanno lo stesso valore e il subfile non avrà altre pagine oltre a quella in vista.

```

A SFLSIZ(0005)
A SFLPAG(0005)
A 52 SFLCLR
A 53 SFLDSPCTL
A 54 SFLDSP
A N56 SFLEND
A CA03(03 'Fine')
A CF06(06 'Da')

```

23.3.1.3. Le paginazioni sono ora entrambe presenti e sono condizionate dalla presenza di record che appartengano alla pagina da visualizzare.

```

A 55 PAGEUP(25 'Pagina precedente')
A 56 PAGEDOWN(26 'Pagina successiva')

```

23.3.1.4. Eliminato il campo relative record number che permetteva sul subfile graduale di emettere a video l'ultima pagina di subfile caricata dal programma.

```

A*** S1RRNTW 4 0H SFLRCDNBR(CURS)
A*** TEXT('RRN to view')
A C1CDPE R B 22 2REFFLD(A1CDPE CNA1)
A TEXT('Start code')
A DSPATR(HI)
A DSPATR(CS)
A DSPATR(PC)
A +1'<--dal codice'
A 23 2'F3=Fine'
A +2'F6=Da'

```

23.3.1.5. Aggiunte le costanti descrittive delle paginazioni permesse.

```

A 55 +2'PGUP=Prec'
A 56 +2'PGDW=Succ'
*-----
A R B1 TEXT('Bianco')
A OVERLAY
A 5 4'(Nessun record da mostrare)'
*-----

```

23.3.1.6. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S3W.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S3W.txt)

23.3.2. Sorgente commentato del programma CNA1S3 Rpgle. Contiene poche specifiche provenienti dalla versione CNA1S2 e viene quindi commentato come schema originale.

```
*FREE * Claudio Neroni 24-03-2021 Creato per Corso.
* CNA1S3
* Sfl quadrato
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
* Sfl quadrato
FCNA1S3W cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds (DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
```

23.3.2.1. L'annotazione del numero di righe per pagina è anche qui la trascrizione del contenuto del valore usato come SFLPAG nel display file.

```
* Annota subfile page.
* Uguale al contenuto della keyword SFLPAG(nnnn) del display file.
D SFLPAG c const(0005)
* Relative record number del subfile.
D S1RRN S 5 0
*-----
```

23.3.2.2. L'idea di fondo del programma è scrivere una subroutine di paginazione avanti e una di paginazione indietro che annotano le chiavi uniche prima e ultima presentate nella pagina che caricano. Le chiavi "First" e "Last" sono in doppia versione "Ante" e "Post" esecuzione. Le routine ricevono prima e ultima chiave della pagina precedente in Ante e restituiscono la prima e ultima della pagina che caricano in Post.

```
* Definisce prima chiave presentata sulla pagina corrente.
D KFA s like (A1CDPE) KeyFirstAnte
D KFP s like (A1CDPE) KeyFirstPost
* Definisce ultima chiave presentata sulla pagina corrente.
D KLA s like (A1CDPE) KeyLastAnte
D KLP s like (A1CDPE) KeyLastPost
*-----
* Predisporre chiusura.
C seton 1r
* Riempie la pagina in avanti assumendo posizionamento.
C seton 06
C exsr PGDW
* Cicla intorno al video.
C do *hival
* Emette il subfile.
C if 1=1
* Prenota emissione del controllo.
C seton 53 sfldspctl
* Se subfile pieno (rrn compreso tra 1 e SFLPAG)
* ne prenota emissione.
C setoff 54 sfldsp
C if S1RRN>=1 and S1RRN<=SFLPAG
C seton 54 sfldsp
C endif
* Emette intestazione.
C write T1
* Se subfile vuoto, ricopre area di visualizzazione del subfile.
C N54 write B1 sflblank
* Emette controllo e, se pieno, subfile.
C exfmt C1
* Emette il subfile.
C endif
* Se richiesto, abbandona.
C 03 leave
* Se richiesta pagina precedente.
C if *in25
* Riempie la pagina all'indietro.
C exsr PGUP
* Ricicla.
C iter
* Se richiesta pagina precedente.
C endif
* Se richiesta pagina successiva.
C if *in26
* Riempie la pagina in avanti.
C exsr PGDW
* Ricicla.
```

```

C iter
* Se richiesta pagina successiva.
C endif
* Se richiesto posizionamento.
C if *in06
* Forza le chiavi after ultima pagina vista.
C clear KFA
C eval KLA=C1CDPE
* Riempie la pagina in avanti.
C exsr PGDW
* Ricicla.
C iter
* Se richiesto posizionamento.
C endif
* Abbandona.
C leave
* Cicla intorno al video.
C enddo
*-----

```

23.3.2.3. La subroutine di pagina avanti viene sfruttata anche per il riposizionamento a partire dal codice quando trova acceso l'indicatore 06 che testimonia la richiesta, In tal caso, oltre all'indicatore, le viene fornita la chiave di ricerca come se fosse l'ultima chiave della pagina precedente e il caricamento comprende anche la chiave fornita, se esistente.

```

* Riempie la pagina in avanti.
* Riceve KFA chiave del primo record sulla pagina vista prima.
* Riceve KLA chiave dell'ultimo record sulla pagina vista prima.
* Carica la pagina di subfile leggendo in avanti da dopo KLA.
* Ritorna KFP chiave del primo record sulla pagina caricata.
* Ritorna KLP chiave dell'ultimo record sulla pagina caricata.
C PGDW begsr
* Pulisce le chiavi di pagina post.
C clear KFP
C clear KLP
* Azzera il numero relativo di record di scrittura.
C clear S1RRN
* Pulisce il subfile.
C exsr CLS1

```

23.3.2.4. L'indicatore 06 acceso posiziona il cursore di lettura del database prima della chiave ricevuta che così occuperà la prima riga della pagina in fase di caricamento.

23.3.2.5. L'indicatore 06 spento posiziona il cursore di lettura del database dopo la chiave ricevuta che così resterà esclusa dalla pagina in fase di caricamento.

```

* Se corre posizionamento.
C if *in06
* Si posiziona prima della chiave di posizionamento.
C KLA setll A1

```

23.3.2.1. Prima del caricamento della pagina, tenta una lettura prima della pagina per autorizzare la richiesta di pagina indietro.

```

* Cerca il record precedente la chiave.
C readp A1
* Se non c'e', disabilita PageUp.
C if %eof
C setoff 55 pageup
C endif
* Si riposiziona prima della chiave di posizionamento.
C KLA setll A1
* Se corre paginazione.
C else
* Si posiziona oltre l'ultimo record della pagina vista prima.
C KLA setgt A1
* Se corre...
C endif
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il prossimo record.
C read A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C setoff 56 pagedown
C leave
C endif
* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
* Trascrive il record letto nella riga di subfile.
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRRE=A1PRRE

```

```

* Se non ancora fatto,
* accantona la chiave corrente come prima della pagina.
C if KFP=*blank
C eval KFP=A1CDPE
C endif
* Accantona sempre la chiave corrente come ultima della pagina.
C eval KLP=A1CDPE
* Scrive la riga di subfile.
C write S1
* Se fine pagina, abbandona.
C if S1RRN>=SFLPAG
C leave
C endif
* Cicla sulla lettura del file per riempire il subfile.
C enddo

```

23.3.2.2. Dopo il caricamento della pagina, se non è ancora arrivato a fine file, tenta una lettura oltre la pagina per autorizzare la richiesta di pagina avanti.

```

* Se PageDown e' permesso.
C if *in56
* Cerca il primo record della pagina successiva.
C read A1
* Se non c'e', disabilita PageDown.
C if %eof
C setoff 56 pagedown
C endif
* Se PageDown e' permesso.
C endif
* Ribalta le chiavi post nelle chiavi ante.
C eval KFA=KFP
C eval KLA=KLP
C endsr
*-----

```

23.3.2.3. Il riempimento all'indietro carica per primo l'ultima riga della pagina, poi la penultima e così via. Se i record si esauriscono durante il caricamento, la pagina resta senza le righe in alto,

```

* Riempie la pagina all'indietro.
* Riceve KFA chiave del primo record sulla pagina vista prima.
* Riceve KLA chiave dell'ultimo record sulla pagina vista prima.
* Carica la pagina di subfile leggendo all'indietro da prima di KFA.
* Ritorna KFP chiave del primo record sulla pagina caricata.
* Ritorna KLP chiave dell'ultimo record sulla pagina caricata.
C PGUP begsr
* Pulisce le chiavi di pagina post.
C clear KFP
C clear KLP
* Imposta a subfile page + 1 il numero relativo di record di scrittura.
C eval S1RRN=SFLPAG+1
* Pulisce il subfile.
C exsr CLS1
* Si posiziona prima del primo record della pagina vista prima.
C KFA setll A1
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il record precedente.
C readp A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C setoff 55 pageup
C leave
C endif
* Decrementa il numero relativo di record di scrittura.
C eval S1RRN -=1
* Trascrive il record letto nella riga di subfile.
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRRE=A1PRRE
* Se non ancora fatto,
* accantona la chiave corrente come ultima della pagina.
C if KLP=*blank
C eval KLP=A1CDPE
C endif
* Accantona sempre la chiave corrente come prima della pagina.
C eval KFP=A1CDPE
* Scrive la riga di subfile.
C write S1
* Se fine pagina, abbandona.
C if S1RRN=1
C leave
C endif
* Cicla sulla lettura del file per riempire il subfile.
C enddo

```

```

* Se PageUp e' permesso.
C if *in55
* Cerca l'ultimo record della pagina precedente.
C readp A1
* Se non c'e', disabilita PageUp.
C if %eof
C setoff 55 pageup
C endif
* Se PageUp e' permesso.
C endif
* Ribalta le chiavi post nelle chiavi ante.
C eval KFA=KFP
C eval KLA=KLP
C endsr
*-----

```

23.3.2.4. Visto l'utilizzo da parte delle due subroutine di caricamento delle pagine, si isola la pulizia del subfile in una subroutine autonoma.

```

* Pulisce il subfile.
C CLS1 begsr
* Prenota la pulizia del subfile.
C seton 52 sflclr
* Inibisce visualizzazione di controllo e subfile.
C setoff 5354 sfldspct1 sfldsp
* Emette il controllo per la sola pulizia.
C write C1
* Sprenota la pulizia del subfile.
C setoff 52 sflclr
* Permette pageup e pagedown.
C seton 55 pageup
C seton 56 pagedown
C endsr
*-----

```

23.3.2.5. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S3.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S3.txt)

23.3.3. Funzionamento del programma CNA1S3.

- 23.3.3.1. Inizialmente, carica e visualizza dalla vista logica la prima pagina di record.
- 23.3.3.2. Pagina successiva e pagina precedente sono autorizzati solo se esiste una pagina da mostrare.
- 23.3.3.3. La costante di invito e il comando di pagina successiva sono condizionati dalla presenza di almeno un record dopo la pagina presentata.
- 23.3.3.4. La costante di invito e il comando di pagina precedente sono condizionati dalla presenza di almeno un record prima della pagina presentata.
- 23.3.3.5. Il programma intercetta sempre sia la richiesta di pagina successiva che di pagina precedente e ricarica la nuova pagina eliminando la memoria della precedente.
- 23.3.3.6. Ogni pagina del subfile quadrato viene quindi sempre vista subito dopo il suo caricamento dal database e i dati sono sempre ricaricati anche quando la navigazione porta a rivedere una pagina già vista nella stessa chiamata del programma.
- 23.3.3.7. È possibile riposizionare la visualizzazione premendo F6. Si ricarica allora una pagina partendo dalla chiave indicata al piede del subfile.
- 23.3.3.8. Dopo il posizionamento, a differenza fondamentale del totale e del graduale, è disponibile anche la pagina precedente, se contiene record.
- 23.3.3.9. Subfile end (+ in basso a destra) viene gestito correttamente grazie allo stesso condizionamento che regola la presentazione della costante di pagina successiva.

## 24. Subfile di visualizzazione con massimizzazione dell'uso di subroutine

24.1. Propongo, dei tre programmi del capitolo precedente, le versioni nelle quali ho esasperato l'uso delle subroutine per avere una base migliore per le successive implementazioni degli schemi. I tre programmi di gestione del subfile totale, graduale e quadrato sono stati modificati con l'uso sistematico delle subroutine. La modifica è puramente tecnica e quindi li riporto senza commenti e spiegazioni extra.

### 24.1.1. Sorgente del programma CNA1S1E Rpgle.

```
*FREE * Claudio Neroni 29-03-2021 Creato per Corso.
* CNA1S1E
* Sfl totale con exsr
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
* Sfl totale
FCNA1S1W cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds(DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
* Relative record number del subfile.
D S1RRN S 5 0
*-----
* Predispone chiusura.
C seton lr
* Esegue il video di controllo.
C exsr SRC1
*-----
* Esegue il video di controllo.
C SRC1 begsr
* Cicla intorno al video di controllo.
C do *hival
* Si posiziona prima della chiave richiesta.
C C1CDPE setll A1
* Azzerà il numero relativo di record di scrittura.
C clear S1RRN
* Pulisce il subfile.
C exsr CLS1
* Riempie il subfile.
C exsr SRS1
* Emette il subfile.
C exsr EXC1
* Se richiesto, abbandona.
C 03 leave
* Se richiesta posizionamento, ricicla.
C 06 iter
* Abbandona.
C leave
* Cicla intorno al video di controllo.
C enddo
C endsr
*-----
* Pulisce il subfile.
C CLS1 begsr
* Prenota la pulizia del subfile.
C seton 52 sflclr
* Inibisce visualizzazione di controllo e subfile.
C setoff 5354 sfldspctl sfldsp
* Emette il controllo per la sola pulizia.
C write C1
* Sprenota la pulizia del subfile.
C setoff 52 sflclr
* Pagedown automatico.
C setoff 56 pagedown
C endsr
*-----
* Trascrive il record di database nella riga di subfile.
C DBS1 begsr
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRRE=A1PRRE
C endsr
*-----
* Riempie il subfile.
C SRS1 begsr
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
```



```

* Legge il prossimo record.
C read A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C leave
C endif
* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
* Trascrive il record di database nella riga di subfile.
C exsr DBS1
* Scrive la riga di subfile.
C write S1
* Cicla sulla lettura del file per riempire il subfile.
C enddo
C endsr
*-----
* Emette il subfile.
C EXC1 begsr
* Prenota emissione del controllo.
C seton 53 sfldspctl
* Se subfile pieno, ne prenota emissione.
C S1RRN comp *zero 54 sfldsp
* Scrive testata.
C write T1
* Se subfile vuoto, sbianca area di visualizzazione del subfile.
C N54 write B1 sb1blank
* Emette controllo e, se pieno, subfile.
C exfmt C1
C endsr
*-----

```

24.1.1.1. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S1E.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S1E.txt)

#### 24.1.2. Sorgente del programma CNA1S2E Rpgle.

```

*FREE * Claudio Neroni 29-03-2021 Creato per Corso.
* CNA1S2E
* Sfl graduale +exsr
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
* Sfl graduale
FCNA1S2W cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds(DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
* Annota subfile page.
D SFLPAG c const(0005)
* Relative record number del subfile.
D S1RRN S 5 0
*-----
* Predispone chiusura.
C seton 1r
* Esegue il video di controllo.
C exsr SRC1
*-----
* Esegue il video di controllo.
C SRC1 begsr
* Cicla intorno al video di controllo.
C do *hival
* Se non corre richiesta di prossima pagina.
C if not *in26
* Si posiziona prima della chiave richiesta.
C C1CDPE setll A1
* Azzerà il numero relativo di record di scrittura.
C clear S1RRN
* Pulisce il subfile.
C exsr CLS1
* Se non corre richiesta di prossima pagina.
C endif
* Riempie il subfile.
C exsr SRS1
* Emette il subfile.
C exsr EXC1
* Se richiesto, abbandona.
C 03 leave
* Se richiesta pagina successiva, ricicla.
C 26 iter
* Se richiesta posizionamento, ricicla.

```

```

C 06 iter
* Ricicla.
C iter
* Cicla intorno al video di controllo.
C enddo
C endsr
*-----
* Pulisce il subfile.
C CLS1 begsr
* Prenota la pulizia del subfile.
C seton 52 sflclr
* Inibisce visualizzazione di controllo e subfile.
C setoff 5354 sfldspctl sfldsp
* Emette il controllo per la sola pulizia.
C write C1
* Sprenota la pulizia del subfile.
C setoff 52 sflclr
* Pagedown automatico.
C seton 56 pagedown
C endsr
*-----
* Trascrive il record di database nella riga di subfile.
C DBS1 begsr
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRRE=A1PRRE
C endsr
*-----
* Riempie il subfile.
C SRS1 begsr
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il prossimo record.
C read A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C setoff 56 pagedown
C leave
C endif
* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
* Trascrive il record di database nella riga di subfile.
C exsr DBS1
* Salva ultima chiave letta.
C eval A1CDPESV=A1CDPE
C *like define A1CDPE A1CDPESV
* Scrive la riga di subfile.
C write S1
* Trascrive il relative record number corrente in quello da vedere.
C eval S1RRNTW=S1RRN
* Se fine pagina, abbandona.
C if %rem(S1RRN:SFLPAG)=*zero
C leave
C endif
* Cicla sulla lettura del file per riempire il subfile.
C enddo
* Tenta una lettura oltre l'ultimo record messo su subfile.
C read A1
* Se non ce ne sono altri, spegne sflend.
C if %eof
C setoff 56 pagedown
C endif
* Si riposiziona.
C A1CDPESV setgt A1
C endsr
*-----
* Emette il subfile.
C EXC1 begsr
* Prenota emissione del controllo.
C seton 53 sfldspctl
* Se subfile pieno, ne prenota emissione.
C S1RRN comp *zero 54 sfldsp
* Scrive testata.
C write T1
* Se subfile vuoto, sbianca area di visualizzazione del subfile.
C N54 write B1 sblblank
* Emette controllo e, se pieno, subfile.
C exfmt C1
C endsr
*-----

```

24.1.2.1. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S2E.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S2E.txt)

### 24.1.3. Sorgente del programma CNA1S3E Rpgle.

```

*FREE * Claudio Neroni 30-03-2021 Creato per Corso.
* CNA1S3E
* Sfl quadrato con exsr
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
* Sfl quadrato
FCNA1S3W cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds(DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
* Annota subfile page.
* Uguale al contenuto della keyword SFLPAG(nnnn) del display file.
D SFLPAG c const(0005)
* Relative record number del subfile.
D S1RRN S 5 0
*-----
* Definisce prima chiave presentata sulla pagina corrente.
D KFA s like(A1CDPE) KeyFirstAnte
D KFP s like(A1CDPE) KeyFirstPost
* Definisce ultima chiave presentata sulla pagina corrente.
D KLA s like(A1CDPE) KeyLastAnte
D KLP s like(A1CDPE) KeyLastPost
*-----
* Predispone chiusura.
C seton 1r
* Esegue il video di controllo.
C exsr SRC1
*-----
* Esegue il video di controllo.
C SRC1 begsr
* Riempie la pagina in avanti assumendo posizionamento.
C seton 06
C exsr PGDW
* Cicla intorno al video.
C do *hival
* Emette il subfile.
C exsr EXC1
*---
* Se richiesto, abbandona.
C 03 leave
*---
* Se richiesto posizionamento.
C if *in06
* Forza le chiavi after ultima pagina vista.
C clear KFA
C eval KLA=C1CDPE
* Riempie la pagina in avanti.
C exsr PGDW
* Ricicla.
C iter
* Se richiesto posizionamento.
C endif
*---
* Se richiesta pagina precedente.
C if *in25
* Riempie la pagina all'indietro.
C exsr PGUP
* Ricicla.
C iter
* Se richiesta pagina precedente.
C endif
*---
* Se richiesta pagina successiva.
C if *in26
* Riempie la pagina in avanti.
C exsr PGDW
* Ricicla.
C iter
* Se richiesta pagina successiva.
C endif
*---
* Ricicla.
C iter
* Cicla intorno al video.
C enddo
C endsr

```

```

*-----
* Riempie la pagina in avanti.
* Riceve KFA chiave del primo record sulla pagina vista prima.
* Riceve KLA chiave dell'ultimo record sulla pagina vista prima.
* Carica la pagina di subfile leggendo in avanti da dopo KLA.
* Ritorna KFP chiave del primo record sulla pagina caricata.
* Ritorna KLP chiave dell'ultimo record sulla pagina caricata.
C PGDW begsr
* Pulisce le chiavi di pagina post.
C clear KFP
C clear KLP
* Azzera il numero relativo di record di scrittura.
C clear S1RRN
* Pulisce il subfile.
C exsr CLS1
* Se corre posizionamento.
C if *in06
* Si posiziona prima della chiave di posizionamento.
C KLA setll A1
* Cerca il record precedente la chiave.
C readp A1
* Se non c'e', disabilita PageUp.
C if %eof
C setoff 55 pageup
C endif
* Si riposiziona prima della chiave di posizionamento.
C KLA setll A1
* Se corre paginazione.
C else
* Si posiziona oltre l'ultimo record della pagina vista prima.
C KLA setgt A1
* Se corre paginazione.
C endif
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il prossimo record.
C read A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C setoff 56 pagedown
C leave
C endif
* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
* Trascrive il record di database nella riga di subfile.
C exsr DBS1
* Se non ancora fatto,
* accantona la chiave corrente come prima della pagina.
C if KFP=*blank
C eval KFP=A1CDPE
C endif
* Accantona sempre la chiave corrente come ultima della pagina.
C eval KLP=A1CDPE
* Scrive la riga di subfile.
C write S1
* Se fine pagina, abbandona.
C if S1RRN>=SFLPAG
C leave
C endif
* Cicla sulla lettura del file per riempire il subfile.
C enddo
* Se PageDown e' permesso.
C if *in56
* Cerca il primo record della pagina successiva.
C read A1
* Se non c'e', disabilita PageDown.
C if %eof
C setoff 56 pagedown
C endif
* Se PageDown e' permesso.
C endif
* Ribalta le chiavi post nelle chiavi ante.
C eval KFA=KFP
C eval KLA=KLP
C endsr
*-----
* Riempie la pagina all'indietro.
* Riceve KFA chiave del primo record sulla pagina vista prima.
* Riceve KLA chiave dell'ultimo record sulla pagina vista prima.
* Carica la pagina di subfile leggendo all'indietro da prima di KFA.
* Ritorna KFP chiave del primo record sulla pagina caricata.
* Ritorna KLP chiave dell'ultimo record sulla pagina caricata.
C PGUP begsr
* Pulisce le chiavi di pagina post.

```

```

C clear KFP
C clear KLP
* Imposta a subfile page + 1 il numero relativo di record di scrittura.
C eval S1RRN=SFLPAG+1
* Pulisce il subfile.
C exsr CLS1
* Si posiziona prima del primo record della pagina vista prima.
C KFA setll A1
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il record precedente.
C readp A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C setoff 55 pageup
C leave
C endif
* Decrementa il numero relativo di record di scrittura.
C eval S1RRN -=1
* Trascrive il record di database nella riga di subfile.
C exsr DBS1
* Se non ancora fatto,
* accantona la chiave corrente come ultima della pagina.
C if KLP=*blank
C eval KLP=A1CDPE
C endif
* Accantona sempre la chiave corrente come prima della pagina.
C eval KFP=A1CDPE
* Scrive la riga di subfile.
C write S1
* Se inizio pagina, abbandona.
C if S1RRN=1
C leave
C endif
* Cicla sulla lettura del file per riempire il subfile.
C enddo
* Se PageUp e' permesso.
C if *in55
* Cerca l'ultimo record della pagina precedente.
C readp A1
* Se non c'e', disabilita PageUp.
C if %eof
C setoff 55 pageup
C endif
* Se PageUp e' permesso.
C endif
* Ribalta le chiavi post nelle chiavi ante.
C eval KFA=KFP
C eval KLA=KLP
C endsr
* -----
* Pulisce il subfile.
C CLS1 begsr
* Prenota la pulizia del subfile.
C seton 52 sflclr
* Inibisce visualizzazione di controllo e subfile.
C setoff 5354 sfldspctl sfldsp
* Emette il controllo per la sola pulizia.
C write C1
* Sprenota la pulizia del subfile.
C setoff 52 sflclr
* Permette pageup e pagedown.
C seton 55 pageup
C seton 56 pagedown
C endsr
* -----
* Emette il subfile.
C EXC1 begsr
* Prenota emissione del controllo.
C seton 53 sfldspctl
* Se subfile pieno (rrn compreso tra 1 e SFLPAG)
* ne prenota emissione.
C setoff 54 sfldsp
C if S1RRN>=1 and S1RRN<=SFLPAG
C seton 54 sfldsp
C endif
* Emette intestazione.
C write T1
* Se subfile vuoto, ricopre area di visualizzazione del subfile.
C N54 write B1 sflblank
* Emette controllo e, se pieno, subfile.
C exfmt C1
C endsr
* -----

```

```
* Trascrive il record di database nella riga di subfile.
C DBS1 begsr
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRRE=A1PRRE
C endsr
*-----
```

24.1.3.1. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S3E.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1S3E.txt)

## 25. Subfile di scelta

25.1. Le versioni seguenti aggiungono ai tre schemi la scelta sulle righe di subfile.

25.2. Elenco anagrafico su subfile a riempimento totale con scelta e subroutine (oggetti CNA1Z1\*).

25.2.1. Sorgente commentato del video CNA1Z1W Dspf. Ottenuto modificando la versione CNA1S1W.

```
* Claudio Neroni 22-03-2021 Creato per Corso.
* CNA1Z1W
* Sfl totale con scelta
*-----
A DSPSIZ(24 80 *DS3)
A CHGINPDFT
A INDARA
A ERRSFL
A INDTXT(52 'Subfile Clear')
A INDTXT(53 'Subfile Display Control')
A INDTXT(54 'Subfile Display')
A INDTXT(56 'Subfile End')
```

25.2.1.1. Fornisce un testo agli indicatori che mettono in position cursor e reverse image la scelta sulla riga di subfile.

```
A INDTXT(58 'Subfile Row +
A Position Cursor')
A INDTXT(59 'Subfile Row +
A Reverse Image')
*-----
A R T1
A TEXT('Testata')
A OVERLAY
A 1 27'Elenco persone'
A DSPATR(HI)
A DSPATR(UL)
```

25.2.1.2. Modifica le intestazioni di colonna della riga di subfile a causa dell'introduzione della scelta.

```
A 2 2'S Codice Descrizione -
A Pr Pr'
A DSPATR(HI)
A 3 2'c person persona -
A Na Re'
A DSPATR(HI)
A DSPATR(UL)
*-----
A R S1
A SFL
A TEXT('Riga')
```

25.2.1.3. Introduce il campo scelta sulla riga di subfile.

```
A SISCE 1 B 4 2TEXT('Scelta')
A DSPATR(UL)
A 58 DSPATR(PC)
A 59 DSPATR(RI)
A S1CDPE R O +1REFFLD(A1CDPE CNA1)
A S1DEPE R O +1REFFLD(A1DEPE CNA1)
A S1PRNA R O +1REFFLD(A1PRNA CNA1)
A S1PRRE R O +1REFFLD(A1PRRE CNA1)
*-----
A R C1
A SFLCTL(S1)
A TEXT('Controllo')
A OVERLAY
A SFLSIZ(0010)
A SFLPAG(0005)
A 52 SFLCLR
A 53 SFLDSPCTL
A 54 SFLDSP
A N56 SFLEND
A CA03(03 'Fine')
A CF06(06 'Da')
```

25.2.1.4. Aggiunge gli errori riguardanti la scelta sul subfile.

```
A 60 61 SFLMSG('Scelta diversa da X e Y')
A 60 62 SFLMSG('>="=Scelta eseguita')
A 60 64 SFLMSG('Nessuna scelta')
```

25.2.1.5. Aggiunge il relative record number della prima riga in errore perché si presenti la pagina che la contiene.

```
A S1RRNTW 4 0H SFLRCDNBR(CURS)
A TEXT('RRN to view')
A C1CDPE R B 22 2REFFLD(A1CDPE CNA1)
A TEXT('Start code')
A DSPATR(HI)
A DSPATR(CS)
```

25.2.1.6. Il cursore si posiziona sulla chiave di riposizionamento solo se non ci sono errori in corso di segnalazione.

|        |      |                                |
|--------|------|--------------------------------|
| A      | N60  | DSPATR(PC)                     |
| A      |      | +1'<--dal codice'              |
| A      | 23   | 2'F3=Fine'                     |
| A      |      | +2'F6=Da'                      |
| *----- |      |                                |
| A      | R B1 |                                |
| A      |      | TEXT('Bianco')                 |
| A      |      | OVERLAY                        |
| A      | 5    | 4'(Nessun record da mostrare)' |
| *----- |      |                                |

25.2.1.7. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z1W.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z1W.txt)

25.2.2. Sorgente commentato del programma CNA1Z1E Rpgle. Ottenuto modificando la versione CNA1S1E.

|                                                           |      |                 |
|-----------------------------------------------------------|------|-----------------|
| *FREE * Claudio Neroni 30-03-2021 Creato per Corso.       |      |                 |
| * CNA1Z1E                                                 |      |                 |
| * Sfl totale con scelta +exsr                             |      |                 |
| *-----                                                    |      |                 |
| * Anagrafica persone logico codice                        |      |                 |
| FCNA11                                                    | if e | k disk          |
| * Sfl totale con scelta                                   |      |                 |
| FCNA1Z1W                                                  | cf e | workstn         |
| * Associa il relative record number al record di subfile. |      |                 |
| F                                                         |      | sfile(S1:S1RRN) |
| * Display Specific Feedback Information                   |      |                 |
| F                                                         |      | infds(DSPFBK)   |
| *-----                                                    |      |                 |
| * Display Specific Feedback Information                   |      |                 |
| /COPY CORSO,DSPFBK                                        |      |                 |
| *-----                                                    |      |                 |
| * Relative record number del subfile.                     |      |                 |
| D S1RRN                                                   | S    | 5 0             |

25.2.2.1. Numero di record totali nel subfile.

|                                        |   |             |
|----------------------------------------|---|-------------|
| * Numero di record nel subfile finale. |   |             |
| D S1RRNEND                             | S | like(S1RRN) |

25.2.2.2. Numero del record di subfile in esame.

|                                                  |   |             |
|--------------------------------------------------|---|-------------|
| * Indice di loop durante la lettura del subfile. |   |             |
| D XX                                             | S | like(S1RRN) |
| *-----                                           |   |             |

25.2.2.3. Campo di servizio per l'operazione display sostitutiva dell'esecuzione di una scelta.

|                                        |       |        |
|----------------------------------------|-------|--------|
| * Comodo per risposta dsply.           |       |        |
| D X                                    | s     | 1      |
| *-----                                 |       |        |
| * Predispone chiusura.                 |       |        |
| C                                      | seton | lr     |
| * Esegue il video di controllo.        |       |        |
| C                                      | exsr  | SRC1   |
| *-----                                 |       |        |
| * Esegue il video di controllo.        |       |        |
| C                                      | SRC1  | begsr  |
| * Cicla intorno al video di controllo. |       |        |
| C                                      | do    | *hival |

25.2.2.4. Se corre errore, non ricarica il subfile.

|                                              |        |           |
|----------------------------------------------|--------|-----------|
| * Se non corre errore.                       |        |           |
| C                                            | if     | not *in60 |
| * Si posiziona prima della chiave richiesta. |        |           |
| C                                            | C1CDPE | setll A1  |
| * Pulisce il subfile.                        |        |           |
| C                                            | exsr   | CLS1      |
| * Riempie il subfile.                        |        |           |
| C                                            | exsr   | SRS1      |
| * Se non corre errore.                       |        |           |
| C                                            | endif  |           |
| * Emette il subfile.                         |        |           |
| C                                            | exsr   | EXC1      |
| * Se richiesto, abbandona.                   |        |           |
| C                                            | 03     | leave     |
| * Se richiesta posizionamento, ricicla.      |        |           |
| C                                            | 06     | iter      |

25.2.2.5. Chiama il controllo e l'esecuzione delle scelte.

|                             |      |      |
|-----------------------------|------|------|
| * Esamina ed esegue scelte. |      |      |
| C                           | exsr | SCS1 |

25.2.2.6. Se errore durante le scelte, ricicla.

|            |    |      |
|------------|----|------|
| C          | 60 | iter |
| * Ricicla. |    |      |



```

C iter
* Cicla intorno al video di controllo.
C enddo
C endsr
*-----
* Pulisce il subfile.
C CLS1 begsr

```

#### 25.2.2.7. Assume e annota numeri relativi di record.

```

* Azzera il numero relativo di record di scrittura.
C clear S1RRN
* Azzera il numero relativo di record ultimo usato.
C clear S1RRNEND
* Predispone la visualizzazione della prima pagina.
C eval S1RRNTW=1
* Prenota la pulizia del subfile.
C seton 52 sflclr
* Inibisce visualizzazione di controllo e subfile.
C setoff 5354 sfldspctl sfldsp
* Emette il controllo per la sola pulizia.
C write C1
* Sprenota la pulizia del subfile.
C setoff 52 sflclr
* Pagedown automatico.
C setoff 56 pagedown
C endsr
*-----
* Trascrive il record di database nella riga di subfile.
C DBS1 begsr

```

#### 25.2.2.8. Sbianca la scelta sulla riga caricata nel subfile.

```

* Scelta in bianco.
C eval S1SCE =*blank
* Dati.
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRRE=A1PRRE
C endsr
*-----
* Riempie il subfile.
C SRS1 begsr
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il prossimo record.
C read A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C leave
C endif
* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
* Trascrive il record di database nella riga di subfile.
C exsr DBS1
* Scrive la riga di subfile.
C write S1

```

#### 25.2.2.9. Tiene nota del numero di record caricati nel subfile.

```

* Trascrive il relative record number corrente nell'ultimo usato.
C eval S1RRNEND=S1RRN
* Cicla sulla lettura del file per riempire il subfile.
C enddo
C endsr
*-----
* Emette il subfile.
C EXC1 begsr
* Prenota emissione del controllo.
C seton 53 sfldspctl
* Se subfile pieno, ne prenota emissione.
C S1RRN comp *zero 54 sfldsp
* Scrive testata.
C write T1
* Se subfile vuoto, sbianca area di visualizzazione del subfile.
C N54 write B1 sblblank

```

#### 25.2.2.10. Perfeziona quale pagina visualizzerà.

```

* Se la pagina da vedere non e' stata scelta, sceglie la prima.
C if S1RRNTW=*zero
C eval S1RRNTW=1
C endif
* Emette controllo e, se pieno, subfile.
C exfmt C1

```

#### 25.2.2.11. Resetta tutti gli indicatori di errore, position cursor e reverse image della riga di subfile.

|                  |        |        |
|------------------|--------|--------|
| * Spegne errori. |        |        |
| C                | setoff | 5859   |
| C                | setoff | 606162 |
| C                | setoff | 6364   |
| C                | endsr  |        |
| *-----           |        |        |

#### 25.2.2.12. Aggiunge l'esame e l'esecuzione delle scelte sulle righe di subfile.

|                                                                    |                                                |        |
|--------------------------------------------------------------------|------------------------------------------------|--------|
| * Esamina ed esegue scelte.                                        |                                                |        |
| * Legge piu' volte tutti i record del subfile con un loop di chain |                                                |        |
| * per trattare le scelte.                                          |                                                |        |
| C                                                                  | SCS1 begsr                                     |        |
| * Esegue.                                                          |                                                |        |
| C                                                                  | do                                             |        |
| *----                                                              |                                                |        |
| * ELIMINA REVERSE IMAGE E POSITION CURSOR DALLE RIGHE              |                                                |        |
| * Spegne reverse image e position cursor sulla riga.               |                                                |        |
| C                                                                  | setoff                                         | 5859   |
| * Cicla sulle righe di subfile.                                    |                                                |        |
| C                                                                  | 1 do S1RRNEND XX                               |        |
| * Legge una riga.                                                  |                                                |        |
| C                                                                  | XX chain S1                                    |        |
| * Elimina dalla riga reverse image e position cursor.              |                                                |        |
| C                                                                  | update S1                                      |        |
| * Cicla sulle righe di subfile.                                    |                                                |        |
| C                                                                  | enddo                                          |        |
| *----                                                              |                                                |        |
| * Prenota l'uscita della prima pagina di subfile.                  |                                                |        |
| C                                                                  | eval S1RRNTW=*zero                             |        |
| * Spegne errori.                                                   |                                                |        |
| C                                                                  | setoff                                         | 5859   |
| C                                                                  | setoff                                         | 606162 |
| C                                                                  | setoff                                         | 6364   |
| *----                                                              |                                                |        |
| * PREPARA LA SEGNALAZIONE DI SCELTE NON PREVISTE SULLE RIGHE       |                                                |        |
| * Assume non ancora trovata scelta qualsiasi.                      |                                                |        |
| C                                                                  | setoff                                         | 99     |
| * Cicla sulle righe di subfile.                                    |                                                |        |
| C                                                                  | 1 do S1RRNEND XX                               |        |
| * Legge una riga.                                                  |                                                |        |
| C                                                                  | XX chain S1                                    |        |
| * Se la riga porta una scelta qualsiasi, lo annota.                |                                                |        |
| C                                                                  | if S1SCE<>*blank                               |        |
| C                                                                  | seton                                          | 99     |
| C                                                                  | endif                                          |        |
| * Se la riga porta una scelta non prevista,                        |                                                |        |
| * prenota PC, RI ed errore.                                        |                                                |        |
| C                                                                  | if S1SCE<>*blank and S1SCE<>'X' and S1SCE<>'Y' |        |
| C                                                                  | seton                                          | 5859   |
| C                                                                  | seton                                          | 6061   |
| C                                                                  | endif                                          |        |
| * Se corre l'ultima riga e non ci sono scelte,                     |                                                |        |
| * prenota PC, RI ed errore.                                        |                                                |        |
| C                                                                  | if XX=S1RRNEND and not *in99                   |        |
| C                                                                  | seton                                          | 58     |
| C                                                                  | seton                                          | 6064   |
| C                                                                  | endif                                          |        |
| * Se la riga e' in errore.                                         |                                                |        |
| C                                                                  | if *in58 or *in59                              |        |
| * Prenota la pagina contenente il primo errore.                    |                                                |        |
| C                                                                  | if S1RRNTW=*zero                               |        |
| C                                                                  | eval S1RRNTW=XX                                |        |
| C                                                                  | endif                                          |        |
| * Ricalca il record in errore per PC e RI.                         |                                                |        |
| C                                                                  | update S1                                      |        |
| * Spegne PC e RI.                                                  |                                                |        |
| C                                                                  | setoff                                         | 5859   |
| * Se la riga e' in errore.                                         |                                                |        |
| C                                                                  | endif                                          |        |
| * Cicla sulle righe di subfile.                                    |                                                |        |
| C                                                                  | enddo                                          |        |
| *----                                                              |                                                |        |
| * Se errore, abbandona.                                            |                                                |        |
| C                                                                  | 60 leave                                       |        |
| *----                                                              |                                                |        |
| * ESEGUE LE SCELTE PREVISTE SULLE RIGHE                            |                                                |        |
| * Cicla sulle righe di subfile.                                    |                                                |        |
| C                                                                  | 1 do S1RRNEND XX                               |        |
| * Legge una riga.                                                  |                                                |        |
| C                                                                  | XX chain S1                                    |        |
| * Se corre una scelta valida.                                      |                                                |        |
| C                                                                  | if S1SCE<>*blank                               |        |
| * Esegue la scelta.                                                |                                                |        |
|                                                                    | dsply (S1SCE+' '+S1CDPE) ' ' x;                |        |

```

* Trasforma scelta in "eseguita",
C eval S1SCE='>'
* Prenota PC.
C seton
* Finge errore solo per non ricaricare il sfl e segnalare.
C seton
* Prenota la pagina contenente la prima scelta eseguita.
C if S1RRNTW=*zero
C eval S1RRNTW=XX
C endif
* Ricalca il record per PC.
C update S1
* Spegne errori.
C setoff
* Se corre una scelta valida.
C endif
* Cicla sulle righe di subfile.
C enddo
*----
* Esegue.
C enddo
C endsr
*-----

```

25.2.2.13. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z1E.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z1E.txt)

25.2.3. Funzionamento del programma CNA1Z1E.

25.2.3.1. Sulle righe di subfile è presente una scelta che può assumere il valore X o Y.

25.2.3.2. Le scelte diverse sono segnalate con reverse image. La rivisualizzazione del subfile presenta la pagina con il primo errore in position cursor.

25.2.3.3. Se le scelte superano il controllo, vengono eseguite tramite un'esecuzione simbolica (operazione dsply).

## 25.3. Elenco anagrafico su subfile a riempimento graduale con scelta e subroutine (oggetti CNA1Z2\*).

25.3.1. Sorgente commentato del video CNA1Z2W Dspf. Ottenuto modificando la versione CNA1S2W.

```

* Claudio Neroni 30-03-2021 Creato per Corso.
* CNA1Z2W
* Sfl graduale con scelta
*-----
A DSPSIZ(24 80 *DS3)
A CHGINPDFT
A INDARA
A ERRSFL
A INDTXT(52 'Subfile Clear')
A INDTXT(53 'Subfile Display Control')
A INDTXT(54 'Subfile Display')
A INDTXT(56 'Subfile End')

```

25.3.1.1. Fornisce un testo agli indicatori che mettono in position cursor e reverse image la scelta sulla riga di subfile.

```

A INDTXT(58 'Subfile Row +
A Position Cursor')
A INDTXT(59 'Subfile Row +
A Reverse Image')
*-----
A R T1
A TEXT('Testata')
A OVERLAY
A 1 27'Elenco persone'
A DSPATR(HI)
A DSPATR(UL)

```

25.3.1.2. Modifica le intestazioni di colonna della riga di subfile a causa dell'introduzione della scelta.

```

A 2 2'S Codice Descrizione
A Pr Pr'
A DSPATR(HI)
A 3 2'c person persona
A Na Re'
A DSPATR(HI)
A DSPATR(UL)
*-----
A R S1
A SFL
A TEXT('Riga')

```

25.3.1.3. Introduce il campo scelta sulla riga di subfile.

```

A S1SCE 1 B 4 2TEXT('Scelta')
A DSPATR(UL)
A 58 DSPATR(PC)
A 59 DSPATR(RI)
A S1CDPE R O +1REFFLD(A1CDPE CNA1)

```

```

A S1DEPE R O +1REFFLD(A1DEPE CNA1)
A S1PRNA R O +1REFFLD(A1PRNA CNA1)
A S1PRRE R O +1REFFLD(A1PRRE CNA1)
*-----
A R C1 SFLCTL(S1)
A TEXT('Controllo')
A OVERLAY
A SFLSIZ(0010)
A SFLPAG(0005)
A 52 SFLCLR
A 53 SFLDSPCTL
A 54 SFLDSP
A N56 SFLEND
A CA03(03 'Fine')
A CF06(06 'Da')
A PAGEDOWN(26 'Pagina successiva')

```

25.3.1.4. Aggiunge gli errori riguardanti la scelta sul subfile.

```

A 60 61 SFLMSG('Scelta diversa da X e Y')
A 60 62 SFLMSG('>"=Scelta eseguita')
A 60 64 SFLMSG('Nessuna scelta')
*-----
A S1RRNTW 4 0H SFLRCDNBR(CURSOR)
A TEXT('RRN to view')
A C1CDPE R B 22 2REFFLD(A1CDPE CNA1)
A TEXT('Start code')
A DSPATR(HI)
A DSPATR(CS)
A DSPATR(PC)
A +1'<--dal codice'
A 23 2'F3=Fine'
A +2'F6=Da'
*-----
A R B1
A TEXT('Bianco')
A OVERLAY
A 5 4'(Nessun record da mostrare)'
*-----

```

25.3.1.5. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z2W.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z2W.txt)

25.3.2. Sorgente commentato del programma CNA1Z2E Rpgle. Ottenuto modificando la versione CNA1S2E.

```

*FREE * Claudio Neroni 30-03-2021 Creato per Corso.
* CNA1Z2E
* Sfl graduale con scelta +exsr
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
* Sfl graduale con scelta
FCNA1Z2W cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds(DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
* Annota subfile page.
D SFLPAG c const(0005)
*-----
* Relative record number del subfile.
D S1RRN S 5 0

```

25.3.2.1. Numero di record totali nel subfile.

```

* Numero di record nel subfile finale.
D S1RRNEND S like(S1RRN)

```

25.3.2.2. Numero del record di subfile in esame.

```

* Indice di loop durante la lettura del subfile.
D XX S like(S1RRN)
*-----

```

25.3.2.3. Campo di servizio per l'operazione display sostitutiva dell'esecuzione di una scelta.

```

* Comodo per risposta dsply.
D X s 1
*-----
* Predispone chiusura.
C seton 1r
* Esegue il video di controllo.
C exsr SRC1
*-----
* Esegue il video di controllo.
C SRC1 begsr
* Cicla intorno al video di controllo.

```

|   |    |        |
|---|----|--------|
| C | do | *hival |
|---|----|--------|

#### 25.3.2.4. Se corre errore, non ricarica il subfile.

|                                              |        |           |
|----------------------------------------------|--------|-----------|
| * Se non corre errore.                       |        |           |
| C                                            | if     | not *in60 |
| * Se non corre richiesta di prossima pagina. |        |           |
| C                                            | if     | not *in26 |
| * Si posiziona prima della chiave richiesta. |        |           |
| C                                            | C1CDPE | setll A1  |
| * Pulisce il subfile.                        |        |           |
| C                                            | exsr   | CLS1      |
| * Se non corre richiesta di prossima pagina. |        |           |
| C                                            | endif  |           |
| * Riempie il subfile.                        |        |           |
| C                                            | exsr   | SRS1      |
| * Se non corre errore.                       |        |           |
| C                                            | endif  |           |
| * Emette il subfile.                         |        |           |
| C                                            | exsr   | EXC1      |
| * Se richiesto, abbandona.                   |        |           |
| C                                            | 03     | leave     |
| * Se richiesta pagina successiva, ricicla.   |        |           |
| C                                            | 26     | iter      |
| * Se richiesta posizionamento, ricicla.      |        |           |
| C                                            | 06     | iter      |

#### 25.3.2.5. Chiama il controllo e l'esecuzione delle scelte.

|                             |      |      |
|-----------------------------|------|------|
| * Esamina ed esegue scelte. |      |      |
| C                           | exsr | SCS1 |

#### 25.3.2.6. Se errore durante le scelte, ricicla.

|                                        |      |       |
|----------------------------------------|------|-------|
| C                                      | 60   | iter  |
| * Ricicla.                             |      |       |
| C                                      |      | iter  |
| * Cicla intorno al video di controllo. |      |       |
| C                                      |      | enddo |
| C                                      |      | endsr |
| *-----                                 |      |       |
| * Pulisce il subfile.                  |      |       |
| C                                      | CLS1 | begsr |

#### 25.3.2.7. Assume e annota numeri relativi di record.

|                                                          |        |       |                  |
|----------------------------------------------------------|--------|-------|------------------|
| * Azzerare il numero relativo di record di scrittura.    |        |       |                  |
| C                                                        | clear  |       | S1RRN            |
| * Azzerare il numero relativo di record ultimo usato.    |        |       |                  |
| C                                                        | clear  |       | S1RRNEND         |
| * Predispone la visualizzazione della prima pagina.      |        |       |                  |
| C                                                        | eval   |       | S1RRNTW=1        |
| * Prenota la pulizia del subfile.                        |        |       |                  |
| C                                                        | seton  | 52    | sflclr           |
| * Inibisce visualizzazione di controllo e subfile.       |        |       |                  |
| C                                                        | setoff | 5354  | sfldspctl sfldsp |
| * Emette il controllo per la sola pulizia.               |        |       |                  |
| C                                                        | write  | C1    |                  |
| * Sprenota la pulizia del subfile.                       |        |       |                  |
| C                                                        | setoff | 52    | sflclr           |
| * Pagedown automatico.                                   |        |       |                  |
| C                                                        | seton  | 56    | pagedown         |
| C                                                        | endsr  |       |                  |
| *-----                                                   |        |       |                  |
| * Trascrive il record di database nella riga di subfile. |        |       |                  |
| C                                                        | DBS1   | begsr |                  |

#### 25.3.2.8. Sbianca la scelta sulla riga caricata nel subfile.

|                                                         |        |        |               |
|---------------------------------------------------------|--------|--------|---------------|
| * Scelta in bianco.                                     |        |        |               |
| C                                                       | eval   |        | S1SCE =*blank |
| * Dati.                                                 |        |        |               |
| C                                                       | eval   |        | S1CDPE=A1CDPE |
| C                                                       | eval   |        | S1DEPE=A1DEPE |
| C                                                       | eval   |        | S1PRNA=A1PRNA |
| C                                                       | eval   |        | S1PRRE=A1PRRE |
| C                                                       | endsr  |        |               |
| *-----                                                  |        |        |               |
| * Riempie il subfile.                                   |        |        |               |
| C                                                       | SRS1   | begsr  |               |
| * Cicla sulla lettura del file per riempire il subfile. |        |        |               |
| C                                                       | do     | *hival |               |
| * Legge il prossimo record.                             |        |        |               |
| C                                                       | read   | A1     |               |
| * Se non ce ne sono altri, abbandona lettura.           |        |        |               |
| C                                                       | if     | %eof   |               |
| C                                                       | setoff | 56     | pagedown      |
| C                                                       | leave  |        |               |
| C                                                       | endif  |        |               |

```

* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
* Trascrive il record di database nella riga di subfile.
C exsr DBS1
* Salva ultima chiave letta.
C eval A1CDPESV=A1CDPE
C *like define A1CDPE A1CDPESV
* Scrive la riga di subfile.
C write S1

```

#### 25.3.2.9. Tiene nota del numero di record caricati nel subfile.

```

* Trascrive il relative record number corrente nell'ultimo usato.
C eval S1RRNEND=S1RRN
* Trascrive il relative record number corrente in quello da vedere.
C eval S1RRNTW =S1RRN
* Se fine pagina, abbandona.
C if %rem(S1RRN:SFLPAG)=*zero
C leave
C endif
* Cicla sulla lettura del file per riempire il subfile.
C enddo
* Tenta una lettura oltre l'ultimo record messo su subfile.
C read A1
* Se non ce ne sono altri, spegne sflend.
C if %eof
C setoff
C endif
C 56 pagedown
* Si riposiziona.
C A1CDPESV setgt A1
C endsr
* -----
* Emette il subfile.
C EXC1 begsr
* Prenota emissione del controllo.
C seton
C 53 sfldspctl
* Se subfile pieno, ne prenota emissione.
C S1RRN comp *zero
C 54 sfldsp
* Scrive testata.
C write T1
* Se subfile vuoto, sbianca area di visualizzazione del subfile.
C N54 write B1
C sblblank

```

#### 25.3.2.10. Perfeziona quale pagina visualizzerà.

```

* Se la pagina da vedere non e' stata scelta, sceglie la prima.
C if S1RRNTW=*zero
C eval S1RRNTW=1
C endif
* Emette controllo e, se pieno, subfile.
C exfmt C1

```

#### 25.3.2.11. Resetta tutti gli indicatori di errore, position cursor e reverse image della riga di subfile.

```

* Spegne errori.
C setoff
C 606162
C setoff
C 6364
C endsr
* -----

```

#### 25.3.2.12. Aggiunge l'esame e l'esecuzione delle scelte sulle righe di subfile.

```

* Esamina ed esegue scelte.
* Legge piu' volte tutti i record del subfile con un loop di chain
* per trattare le scelte.
C SCS1 begsr
* Esegue.
C do
* ---
* ELIMINA REVERSE IMAGE E POSITION CURSOR DALLE RIGHE
* Spegne reverse image e position cursor sulla riga.
C setoff
C 5859
* Cicla sulle righe di subfile.
C 1 do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Elimina dalla riga reverse image e position cursor.
C update S1
* Cicla sulle righe di subfile.
C enddo
* ---
* Prenota l'uscita della prima pagina di subfile.
C eval S1RRNTW=*zero
* Spegne errori.
C setoff
C 5859
C setoff
C 606162
C setoff
C 6364
* ---

```

```

* PREPARA LA SEGNALAZIONE DI SCELTE NON PREVISTE SULLE RIGHE
* Assume non ancora trovata scelta qualsiasi.
C setoff 99
* Cicla sulle righe di subfile.
C 1 do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Se la riga porta una scelta qualsiasi, lo annota.
C if S1SCE<>*blank
C seton 99
C endif
* Se la riga porta una scelta non prevista,
* prenota PC, RI ed errore.
C if S1SCE<>*blank and S1SCE<>'X' and S1SCE<>'Y'
C seton 5859
C seton 6061
C endif
* Se corre l'ultima riga e non ci sono scelte,
* prenota PC, RI ed errore.
C if XX=S1RRNEND and not *in99
C seton 58
C seton 6064
C endif
* Se la riga e' in errore.
C if *in58 or *in59
* Prenota la pagina contenente il primo errore.
C if S1RRNTW=*zero
C eval S1RRNTW=XX
C endif
* Ricalca il record in errore per PC e RI.
C update S1
* Spegne PC e RI.
C setoff 5859
* Se la riga e' in errore.
C endif
* Cicla sulle righe di subfile.
C enddo
*----
* Se errore, abbandona.
C 60 leave
*----
* ESEGUE LE SCELTE PREVISTE SULLE RIGHE
* Cicla sulle righe di subfile.
C 1 do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Se corre una scelta valida.
C if S1SCE<>*blank
* Esegue la scelta.
C dsply (S1SCE+' '+S1CDPE) ' ' x;
* Trasforma scelta in "eseguita",
C eval S1SCE='>'
* Prenota PC.
C seton 58
* Finge errore solo per non ricaricare il sfl e segnalare.
C seton 6062
* Prenota la pagina contenente la prima scelta eseguita.
C if S1RRNTW=*zero
C eval S1RRNTW=XX
C endif
* Ricalca il record per PC.
C update S1
* Spegne errori.
C setoff 5859
* Se corre una scelta valida.
C endif
* Cicla sulle righe di subfile.
C enddo
*----
* Esegue.
C enddo
C endsr
*-----

```

25.3.2.13. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z2E.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z2E.txt)

25.3.3. Funzionamento del programma CNA1Z2E.

25.3.3.1. Per le scelte sul subfile graduale valgono le stesse regole che valgono sul subfile totale.

## 25.4. Elenco anagrafico su subfile quadrato con scelta e subroutine (oggetti CNA1Z3\*).

25.4.1. Sorgente commentato del video CNA1Z3W Dspf. Ottenuto modificando la versione CNA1S3W.

```

* Claudio Neroni 26-03-2021 Creato per Corso.
* CNA1Z3W

```

|                           |                                      |
|---------------------------|--------------------------------------|
| * Sfl quadrato con scelta |                                      |
| *-----                    |                                      |
| A                         | DSPSIZ(24 80 *DS3)                   |
| A                         | CHGINPDFT                            |
| A                         | INDARA                               |
| A                         | ERRSFL                               |
| A                         | INDTXT(52 'Subfile Clear')           |
| A                         | INDTXT(53 'Subfile Display Control') |
| A                         | INDTXT(54 'Subfile Display')         |
| A                         | INDTXT(55 'PageUp permesso')         |
| A                         | INDTXT(56 'PageDown permesso')       |

25.4.1.1. Fornisce un testo agli indicatori che mettono in position cursor e reverse image la scelta sulla riga di subfile.

|        |                          |
|--------|--------------------------|
| A      | INDTXT(58 'Subfile Row + |
| A      | Position Cursor')        |
| A      | INDTXT(59 'Subfile Row + |
| A      | Reverse Image')          |
| *----- |                          |
| A      | R T1                     |
| A      | TEXT('Testata')          |
| A      | OVERLAY                  |
| A      | 1 27 'Elenco persone'    |
| A      | DSPATR(HI)               |
| A      | DSPATR(UL)               |

25.4.1.2. Modifica le intestazioni di colonna della riga di subfile a causa dell'introduzione della scelta.

|        |                          |              |
|--------|--------------------------|--------------|
| A      | 2 2'S Codice Descrizione | -            |
| A      |                          | Pr Pr'       |
| A      | DSPATR(HI)               |              |
| A      | 3 2'c person persona     | -            |
| A      |                          | Na Re'       |
| A      | DSPATR(HI)               |              |
| A      | DSPATR(UL)               |              |
| *----- |                          |              |
| A      | R S1                     | SFL          |
| A      |                          | TEXT('Riga') |

25.4.1.3. Introduce il campo scelta sulla riga di subfile.

|        |        |   |   |   |                                  |
|--------|--------|---|---|---|----------------------------------|
| A      | SISCE  | 1 | B | 4 | 2TEXT('Scelta')                  |
| A      |        |   |   |   | DSPATR(UL)                       |
| A      | 58     |   |   |   | DSPATR(PC)                       |
| A      | 59     |   |   |   | DSPATR(RI)                       |
| A      | S1CDPE | R | O |   | +1REFFLD(A1CDPE CNA1)            |
| A      | S1DEPE | R | O |   | +1REFFLD(A1DEPE CNA1)            |
| A      | S1PRNA | R | O |   | +1REFFLD(A1PRNA CNA1)            |
| A      | S1PRRE | R | O |   | +1REFFLD(A1PRRE CNA1)            |
| *----- |        |   |   |   |                                  |
| A      | R C1   |   |   |   | SFLCTL(S1)                       |
| A      |        |   |   |   | SFLSIZ(0005)                     |
| A      |        |   |   |   | SFLPAG(0005)                     |
| A      |        |   |   |   | TEXT('Controllo')                |
| A      |        |   |   |   | OVERLAY                          |
| A      | 52     |   |   |   | SFLCLR                           |
| A      | 53     |   |   |   | SFLDSPCTL                        |
| A      | 54     |   |   |   | SFLDSP                           |
| A      | N56    |   |   |   | SFLEND                           |
| A      |        |   |   |   | CA03(03 'Fine')                  |
| A      |        |   |   |   | CF06(06 'Da')                    |
| A      | 55     |   |   |   | PAGEUP(25 'Pagina precedente')   |
| A      | 56     |   |   |   | PAGEDOWN(26 'Pagina successiva') |

25.4.1.4. Aggiunge gli errori riguardanti la scelta sul subfile.

|   |        |   |   |    |                                   |
|---|--------|---|---|----|-----------------------------------|
| A | 60 61  |   |   |    | SFLMSG('Scelta diversa da X e Y') |
| A | 60 62  |   |   |    | SFLMSG('>">=Scelta eseguita')     |
| A | 60 63  |   |   |    | SFLMSG('Scelte e paginazioni +    |
| A |        |   |   |    | incompatibili')                   |
| A | 60 64  |   |   |    | SFLMSG('Nessuna scelta')          |
| A | C1CDPE | R | B | 22 | 4REFFLD(A1CDPE CNA1)              |
| A |        |   |   |    | TEXT('Start code')                |
| A |        |   |   |    | DSPATR(HI)                        |
| A |        |   |   |    | DSPATR(CS)                        |

25.4.1.5. Il cursore si posiziona sulla chiave di riposizionamento solo se non ci sono errori in corso di segnalazione.

|        |      |  |    |   |                   |
|--------|------|--|----|---|-------------------|
| A      | N60  |  |    |   | DSPATR(PC)        |
| A      |      |  |    |   | +1'<--dal codice' |
| A      |      |  | 23 | 2 | 'F3=Fine'         |
| A      |      |  |    |   | +2'F6=Da'         |
| A      | 55   |  |    |   | +2'PGUP=Prec'     |
| A      | 56   |  |    |   | +2'PGDW=Succ'     |
| *----- |      |  |    |   |                   |
| A      | R B1 |  |    |   |                   |



```

A TEXT('Bianco')
A OVERLAY
A 5 4'(Nessun record da mostrare)'
*-----

```

25.4.1.6. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z3W.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z3W.txt)

25.4.2. Sorgente commentato del programma CNA1Z3E Rpgle. Ottenuto modificando la versione CNA1S3E.

```

*FREE * Claudio Neroni 27-03-2021 Creato per Corso.
* CNA1Z3E
* Sfl quadrato con scelta +exsr
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
* Sfl quadrato con scelta
FCNA1Z3W cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds(DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
* Annota subfile page.
* Uguale al contenuto della keyword SFLPAG(nnnn) del display file.
D SFLPAG c const(0005)
* Relative record number del subfile.
D S1RRN S 5 0

```

25.4.2.1. Relative record number delle righe iniziale e finale nella pagina di subfile.

```

* Numero di record nel subfile iniziale e finale.
D S1RRNBEG S like(S1RRN)
D S1RRNEND S like(S1RRN)

```

25.4.2.2. Numero del record di subfile in esame.

```

* Indice di loop durante la lettura del subfile.
D XX S like(S1RRN)
*-----
* Definisce prima chiave presentata sulla pagina corrente.
D KFA s like(A1CDPE) KeyFirstAnte
D KFP s like(A1CDPE) KeyFirstPost
* Definisce ultima chiave presentata sulla pagina corrente.
D KLA s like(A1CDPE) KeyLastAnte
D KLP s like(A1CDPE) KeyLastPost
*-----

```

25.4.2.3. Campo di servizio per l'operazione display sostitutiva dell'esecuzione di una scelta.

```

* Comodo per risposta dsply.
D X s 1
*-----
* Predisporre chiusura.
C seton 1r
* Esegue il video di controllo.
C exsr SRC1
*-----
* Esegue il video di controllo.
C SRC1 begsr
* Riempie la pagina in avanti assumendo posizionamento.
C seton 06
C exsr PGDW
* Cicla intorno al video.
C do *hival
* Emette il subfile.
C exsr EXC1
*-----
* Se richiesto, abbandona.
C 03 leave
*-----
* Se richiesto posizionamento.
C if *in06
* Forza le chiavi after ultima pagina vista.
C clear KFA
C eval KLA=C1CDPE
* Riempie la pagina in avanti.
C exsr PGDW
* Ricicla.
C iter
* Se richiesto posizionamento.
C endif
*-----

```

25.4.2.4. Chiama il controllo e l'esecuzione delle scelte.

```

* Esamina ed esegue scelte.

```

|   |      |      |
|---|------|------|
| C | exsr | SCS1 |
|---|------|------|

#### 25.4.2.5. Se errore durante le scelte, ricicla.

|                                                                  |       |       |
|------------------------------------------------------------------|-------|-------|
| C                                                                | 60    | iter  |
| *---                                                             |       |       |
| * Se richiesta pagina precedente.                                |       |       |
| C                                                                | if    | *in25 |
| * Riempie la pagina all'indietro.                                |       |       |
| C                                                                | exsr  | PGUP  |
| * Ricicla.                                                       |       |       |
| C                                                                | iter  |       |
| * Se richiesta pagina precedente.                                |       |       |
| C                                                                | endif |       |
| *---                                                             |       |       |
| * Se richiesta pagina successiva.                                |       |       |
| C                                                                | if    | *in26 |
| * Riempie la pagina in avanti.                                   |       |       |
| C                                                                | exsr  | PGDW  |
| * Ricicla.                                                       |       |       |
| C                                                                | iter  |       |
| * Se richiesta pagina successiva.                                |       |       |
| C                                                                | endif |       |
| *---                                                             |       |       |
| * Ricicla.                                                       |       |       |
| C                                                                | iter  |       |
| *---                                                             |       |       |
| * Cicla intorno al video.                                        |       |       |
| C                                                                | enddo |       |
| C                                                                | endsr |       |
| *-----                                                           |       |       |
| * Riempie la pagina in avanti.                                   |       |       |
| * Riceve KFA chiave del primo record sulla pagina vista prima.   |       |       |
| * Riceve KLA chiave dell'ultimo record sulla pagina vista prima. |       |       |
| * Carica la pagina di subfile leggendo in avanti da dopo KLA.    |       |       |
| * Ritorna KFP chiave del primo record sulla pagina caricata.     |       |       |
| * Ritorna KLP chiave dell'ultimo record sulla pagina caricata.   |       |       |
| C                                                                | PGDW  | begsr |
| * Pulisce le chiavi di pagina post.                              |       |       |
| C                                                                | clear | KFP   |
| C                                                                | clear | KLP   |

#### 25.4.2.6. Azzerà RRN di prima e ultima riga.

|                                                                |        |             |
|----------------------------------------------------------------|--------|-------------|
| * Azzerà primo e ultimo RRN della pagina.                      |        |             |
| C                                                              | clear  | S1RRNBEG    |
| C                                                              | clear  | S1RRNEND    |
| * Azzerà il numero relativo di record di scrittura.            |        |             |
| C                                                              | clear  | S1RRN       |
| * Pulisce il subfile.                                          |        |             |
| C                                                              | exsr   | CLS1        |
| * Se corre posizionamento.                                     |        |             |
| C                                                              | if     | *in06       |
| * Si posiziona prima della chiave di posizionamento.           |        |             |
| C                                                              | KLA    | setll A1    |
| * Cerca il record precedente la chiave.                        |        |             |
| C                                                              | readp  | A1          |
| * Se non c'e', disabilita PageUp.                              |        |             |
| C                                                              | if     | %eof        |
| C                                                              | setoff | 55 pageup   |
| C                                                              | endif  |             |
| * Si riposiziona prima della chiave di posizionamento.         |        |             |
| C                                                              | KLA    | setll A1    |
| * Se corre paginazione.                                        |        |             |
| C                                                              | else   |             |
| * Si posiziona oltre l'ultimo record della pagina vista prima. |        |             |
| C                                                              | KLA    | setgt A1    |
| * Se corre paginazione.                                        |        |             |
| C                                                              | endif  |             |
| * Cicla sulla lettura del file per riempire il subfile.        |        |             |
| C                                                              | do     | *hival      |
| * Legge il prossimo record.                                    |        |             |
| C                                                              | read   | A1          |
| * Se non ce ne sono altri, abbandona lettura.                  |        |             |
| C                                                              | if     | %eof        |
| C                                                              | setoff | 56 pagedown |
| C                                                              | leave  |             |
| C                                                              | endif  |             |
| * Incrementa il numero relativo di record di scrittura.        |        |             |
| C                                                              | eval   | S1RRN +=1   |
| * Trascrive il record di database nella riga di subfile.       |        |             |
| C                                                              | exsr   | DBS1        |
| * Se non ancora fatto,                                         |        |             |
| * accantona la chiave corrente come prima della pagina.        |        |             |
| C                                                              | if     | KFP=*blank  |
| C                                                              | eval   | KFP=A1CDPE  |

#### 25.4.2.7. Annota il RRN della prima riga.

|   |                                                               |                |
|---|---------------------------------------------------------------|----------------|
| C | eval                                                          | S1RRNBEG=S1RRN |
| C | endif                                                         |                |
| * | Accantona sempre la chiave corrente come ultima della pagina. |                |
| C | eval                                                          | KLP=A1CDPE     |

#### 25.4.2.8. Annota il RRN dell'ultima riga.

|       |                                                                       |                                                         |
|-------|-----------------------------------------------------------------------|---------------------------------------------------------|
| C     | eval                                                                  | S1RRNEND=S1RRN                                          |
| *     | Scrive la riga di subfile.                                            |                                                         |
| C     | write                                                                 | S1                                                      |
| *     | Se fine pagina, abbandona.                                            |                                                         |
| C     | if                                                                    | S1RRN>=SFLPAG                                           |
| C     | leave                                                                 |                                                         |
| C     | endif                                                                 |                                                         |
| *     | Cicla sulla lettura del file per riempire il subfile.                 |                                                         |
| C     | enddo                                                                 |                                                         |
| *     | Se PageDown e' permesso.                                              |                                                         |
| C     | if                                                                    | *in56                                                   |
| *     | Cerca il primo record della pagina successiva.                        |                                                         |
| C     | read                                                                  | A1                                                      |
| *     | Se non c'e', disabilita PageDown.                                     |                                                         |
| C     | if                                                                    | %eof                                                    |
| C     | setoff                                                                | 56 pagedown                                             |
| C     | endif                                                                 |                                                         |
| *     | Se PageDown e' permesso.                                              |                                                         |
| C     | endif                                                                 |                                                         |
| *     | Ribalta le chiavi post nelle chiavi ante.                             |                                                         |
| C     | eval                                                                  | KFA=KFP                                                 |
| C     | eval                                                                  | KLA=KLP                                                 |
| C     | endsr                                                                 |                                                         |
| ----- |                                                                       |                                                         |
| *     | Riempie la pagina all'indietro.                                       |                                                         |
| *     | Riceve                                                                | KFA chiave del primo record sulla pagina vista prima.   |
| *     | Riceve                                                                | KLA chiave dell'ultimo record sulla pagina vista prima. |
| *     | Carica la pagina di subfile leggendo all'indietro da prima di KFA.    |                                                         |
| *     | Ritorna                                                               | KFP chiave del primo record sulla pagina caricata.      |
| *     | Ritorna                                                               | KLP chiave dell'ultimo record sulla pagina caricata.    |
| C     | PGUP                                                                  | begsr                                                   |
| *     | Pulisce le chiavi di pagina post.                                     |                                                         |
| C     | clear                                                                 | KFP                                                     |
| C     | clear                                                                 | KLP                                                     |
| *     | Imposta a subfile page + 1 il numero relativo di record di scrittura. |                                                         |
| C     | eval                                                                  | S1RRN=SFLPAG+1                                          |
| *     | Pulisce il subfile.                                                   |                                                         |
| C     | exsr                                                                  | CLS1                                                    |
| *     | Si posiziona prima del primo record della pagina vista prima.         |                                                         |
| C     | KFA                                                                   | setll A1                                                |
| *     | Cicla sulla lettura del file per riempire il subfile.                 |                                                         |
| C     | do                                                                    | *hival                                                  |
| *     | Legge il record precedente.                                           |                                                         |
| C     | readp                                                                 | A1                                                      |
| *     | Se non ce ne sono altri, abbandona lettura.                           |                                                         |
| C     | if                                                                    | %eof                                                    |
| C     | setoff                                                                | 55 pageup                                               |
| C     | leave                                                                 |                                                         |
| C     | endif                                                                 |                                                         |
| *     | Decrementa il numero relativo di record di scrittura.                 |                                                         |
| C     | eval                                                                  | S1RRN -=1                                               |
| *     | Trascrive il record di database nella riga di subfile.                |                                                         |
| C     | exsr                                                                  | DBS1                                                    |
| *     | Se non ancora fatto,                                                  |                                                         |
| *     | accantona la chiave corrente come ultima della pagina.                |                                                         |
| C     | if                                                                    | KLP=*blank                                              |
| C     | eval                                                                  | KLP=A1CDPE                                              |

#### 25.4.2.9. Annota il RRN dell'ultima riga.

|   |                                                              |                |
|---|--------------------------------------------------------------|----------------|
| C | eval                                                         | S1RRNEND=S1RRN |
| C | endif                                                        |                |
| * | Accantona sempre la chiave corrente come prima della pagina. |                |
| C | eval                                                         | KFP=A1CDPE     |

#### 25.4.2.10. Annota il RRN della prima riga.

|   |                                                       |                |
|---|-------------------------------------------------------|----------------|
| C | eval                                                  | S1RRNBEG=S1RRN |
| * | Scrive la riga di subfile.                            |                |
| C | write                                                 | S1             |
| * | Se inizio pagina, abbandona.                          |                |
| C | if                                                    | S1RRN=1        |
| C | leave                                                 |                |
| C | endif                                                 |                |
| * | Cicla sulla lettura del file per riempire il subfile. |                |
| C | enddo                                                 |                |
| * | Se PageUp e' permesso.                                |                |
| C | if                                                    | *in55          |

```

* Cerca l'ultimo record della pagina precedente.
C readp A1
* Se non c'e', disabilita PageUp.
C if %eof
C setoff 55 pageup
C endif
* Se PageUp e' permesso.
C endif
* Ribalta le chiavi post nelle chiavi ante.
C eval KFA=KFP
C eval KLA=KLP
C endsr
*-----
* Pulisce il subfile.
C CLS1 begsr

```

#### 25.4.2.11. Spegne errori di riga.

```

* Spegne.
C exsr OFF
* Prenota la pulizia del subfile.
C seton 52 sflclr
* Inibisce visualizzazione di controllo e subfile.
C setoff 5354 sfldspctl sfldsp
* Emette il controllo per la sola pulizia.
C write C1
* Sprenota la pulizia del subfile.
C setoff 52 sflclr
* Permette pageup e pagedown.
C seton 55 pageup
C seton 56 pagedown
C endsr
*-----
* Emette il subfile.
C EXC1 begsr
* Prenota emissione del controllo.
C seton 53 sfldspctl
* Se subfile pieno (rrn compreso tra 1 e SFLPAG),
* ne prenota emissione.
C setoff 54 sfldsp
C if S1RRN>=1 and S1RRN<=SFLPAG
C seton 54 sfldsp
C endif
* Emette intestazione.
C write T1
* Se subfile vuoto, ricopre area di visualizzazione del subfile.
C N54 write B1 sflblank
* Emette controllo e, se pieno, subfile.
C exfmt C1
C endsr
*-----

```

#### 25.4.2.12. Aggiunge l'esame e l'esecuzione delle scelte sulle righe di subfile.

```

* Esamina ed esegue scelte.
* Legge piu' volte tutti i record del subfile con un loop di chain
* per trattare le scelte.
C SCS1 begsr
* Esegue.
C do
*-----
* ELIMINA REVERSE IMAGE E POSITION CURSOR DALLE RIGHE
* Spegne.
C exsr OFF
* Cicla sulle righe di subfile.
C S1RRNBEG do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Elimina dalla riga reverse image e position cursor.
C update S1
* Cicla sulle righe di subfile.
C enddo
*-----
* PREPARA LA SEGNALEZIONE DI SCELTE NON PREVISTE SULLE RIGHE
* Assume non ancora trovata scelta qualsiasi.
C setoff 99
* Cicla sulle righe di subfile.
C S1RRNBEG do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Se la riga porta una scelta qualsiasi, lo annota.
C if S1SCE<>*blank
C seton 99
C endif
* Se la riga porta una scelta non prevista,
* prenota PC, RI ed errore.

```

```

C if S1SCE<>*'blank and S1SCE<>'X' and S1SCE<>'Y'
C seton
C 5859
C seton
C 6061
C endif
* Se corre l'ultima riga, non ci sono scelte e non sta paginando,
* prenota PC ed errore.
C if XX=S1RRNEND and not *in99
C and not (*in25 or *in26)
C seton
C 58
C seton
C 6064
C endif
* Se la riga porta una scelta mentre e' richiesta paginazione,
* prenota PC, RI ed errore.
C if S1SCE<>*'blank and (*in25 or *in26)
C seton
C 5859
C seton
C 6063
C endif
* Se corre richiesta di PC o RI,
* ricalca la riga in errore per PC e RI e li spegne.
C if *in58 or *in59
C update S1
C setoff
C 5859
C endif
* Cicla sulle righe di subfile.
C enddo
*----
* Se errore, abbandona.
C 60 leave
*----
* ESEGUE LE SCELTE PREVISTE SULLE RIGHE
* Cicla sulle righe di subfile.
C S1RRNBEG do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Se corre una scelta valida.
C if S1SCE<>*'blank
* Esegue la scelta.
C dsply (S1SCE+' '+S1CDPE) ' ' x;
* Trasforma scelta in "eseguita",
C eval S1SCE='>'
* Prenota PC.
C seton
C 58
* Finge errore solo per non ricaricare il sfl e segnalare.
C seton
C 6062
* Ricalca il record per PC.
C update S1
* Spegne PC e RI.
C setoff
C 5859
* Se corre una scelta valida.
C endif
* Cicla sulle righe di subfile.
C enddo
*----
* Esegue.
C enddo
C endsr
*-----

```

25.4.2.13. Resetta tutti gli indicatori di errore, position cursor e reverse image della riga di subfile.

```

* Spegne.
C OFF begsr
* Spegne position cursor e reverse image sulla riga.
C setoff
C 5859
* Spegne errori e segnalazioni suk controllo.
C setoff
C 606162
C setoff
C 6364
C endsr
*-----
* Trascrive il record di database nella riga di subfile.
C DBS1 begsr

```

25.4.2.14. Sbianca la scelta sulla riga caricata nel subfile.

```

* Scelta in bianco.
C eval S1SCE =*'blank
* Dati.
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRRE=A1PRRE
C endsr
*-----

```

25.4.2.15. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z3E.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z3E.txt)

#### 25.4.3. Funzionamento del programma CNA1Z3E.

25.4.3.1. Per le scelte sul subfile quadrato valgono le stesse regole del subfile totale e graduale ma limitandosi ad una sola pagina. Cioè, possono essere eseguite scelte solo sulle righe della pagina in corso di visualizzazione.

25.4.3.2. Se presente almeno una scelta, non è contemporaneamente permessa la paginazione.

## 26. Subfile quadrato con scelta, decodifica e refresh

26.1. La prossima versione implementa moderatamente lo schema del subfile quadrato tanto per mostrare la capacità di crescita degli schemi nati e strutturati per farlo.

### 26.2. Elenco anagrafico su subfile quadrato doppia riga con scelta e subroutine (oggetti CNA1Z4\*).

26.2.1. Ogni record di questo subfile occupa due righe di video. Quindi anche la testata diventa di due righe video. La riga di subfile contiene anche le decodifiche dei codici presenti nel record. La pagina di subfile contiene abbastanza record da riempire le righe a disposizione. Rispetto allo schema precedente aggiunge un comando di rinfrescamento dei dati a video con il contenuto del database.

26.2.2. Sorgente commentato del video CNA1Z4W Dspf. Ottenuto modificando la versione CNA1Z3W.

```
* Claudio Neroni 01-04-2021 Creato per Corso.
* CNA1Z4W
* Sfl quadrato con scelta +exsr 2righe
*-----
A DSPSIZ(24 80 *DS3)
A CHGINPDFT
A INDARA
A ERRSFL
A INDTXT(52 'Subfile Clear')
A INDTXT(53 'Subfile Display Control')
A INDTXT(54 'Subfile Display')
A INDTXT(55 'PageUp permesso')
A INDTXT(56 'PageDown permesso')
A INDTXT(58 'Subfile Row +
A Position Cursor')
A INDTXT(59 'Subfile Row +
A Reverse Image')
*-----
```

26.2.2.1. Cambia nome del record Testata.

```
A R S1T
A TEXT('Testata elenco')
A OVERLAY
A 1 27'Elenco persone'
A DSPATR(HI)
A DSPATR(UL)
```

26.2.2.2. Modella le costanti della testata per facilitare la lettura delle righe del subfile.

```
A 2 2'S'
A DSPATR(HI)
A DSPATR(UL)
A 2 4'Person'
A DSPATR(HI)
A DSPATR(UL)
A 2 11'Descrizione persona
A '
A DSPATR(HI)
A DSPATR(UL)
A 3 11'Provincia nasc '
A DSPATR(UL)
A 3 27'Provincia resid'
A DSPATR(UL)
*-----
A R S1
A SFL
A TEXT('Riga elenco')
```

26.2.2.3. Prima riga video del record di subfile.

```
A S1SCE 1 B 4 2TEXT('Scelta')
A DSPATR(UL)
A 58 DSPATR(PC)
A 59 DSPATR(RI)
A S1CDPE R O +1REFFLD(A1CDPE CNA1)
A DSPATR(HI)
A S1DEPE R O +1REFFLD(A1DEPE CNA1)
A DSPATR(HI)
```

26.2.2.4. Seconda riga video del record di subfile con le decodifiche prelevate dalle tabelle.

```
A S1PRNA R O 5 11REFFLD(A1PRNA CNA1)
A S1PRNAD R O +1REFFLD(T1DCPR CNT1)
A S1PRRE R O +1REFFLD(A1PRRE CNA1)
A S1PRRED R O +1REFFLD(T1DCPR CNT1)
*-----
```

26.2.2.5. Cambia nome del record Controllo.

```
A R S1C SFLCTL(S1)
```

26.2.2.6. Allunga la pagina.

```
A SFLSIZ(0009)
A SFLPAG(0009)
A TEXT('Controllo elenco')
```

|       |                 |
|-------|-----------------|
| A     | OVERLAY         |
| A 52  | SFLCLR          |
| A 53  | SFLDSPCTL       |
| A 54  | SFLDSP          |
| A N56 | SFLEND          |
| A     | CA03(03 'Fine') |

#### 26.2.2.7. Aggiunge il comando di refresh.

|         |                                                |
|---------|------------------------------------------------|
| A       | CA05(05 'Aggiornamento')                       |
| A       | CF06(06 'Da')                                  |
| A 55    | PAGEUP(25 'Pagina precedente')                 |
| A 56    | PAGEDOWN(26 'Pagina successiva')               |
| A 60 61 | SFLMSG('Scelta diversa da X e Y')              |
| A 60 62 | SFLMSG('>="=Scelta eseguita')                  |
| A 60 63 | SFLMSG('Scelte e paginazioni + incompatibili') |
| A 60 64 | SFLMSG('Nessuna scelta')                       |
| A       | SICDPES R B 22 4REFFLD(A1CDPE CNA1)            |
| A       | TEXT('Start code')                             |
| A       | DSPATR(HI)                                     |
| A       | DSPATR(CS)                                     |
| A N60   | DSPATR(PC)                                     |
| A       | 22 11'<--dal codice'                           |
| A       | 23 2'F3=Fine'                                  |

#### 26.2.2.1. Aggiunge la costante descrittiva del comando di refresh.

|        |               |
|--------|---------------|
| A      | +2'F5=Aggior' |
| A      | +2'F6=Da'     |
| A 55   | +2'PGUP=Prec' |
| A 56   | +2'PGDW=Succ' |
| *----- |               |

#### 26.2.2.2. Cambia nome e testo del record Bianco.

|        |                                  |
|--------|----------------------------------|
| A      | R S1B                            |
| A      | TEXT('Bianco elenco')            |
| A      | OVERLAY                          |
| A      | 5 4'(Nessun record da mostrare)' |
| *----- |                                  |

#### 26.2.2.3. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z4W.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z4W.txt)

#### 26.2.3. Sorgente commentato del programma CNA1Z4E Rpgle. Ottenuto modificando la versione CNA1Z3E.

```
*FREE * Claudio Neroni 27-03-2021 Creato per Corso.
* CNA1Z3E
* Sfl quadrato con scelta +exsr 2righe
*-----
* Anagrafica persone logico codice
FCNA11 if e k disk
```

#### 26.2.3.1. Aggiunge file per decodifica.

```
* Tabella provincie
FCNT11 if e k disk
* Sfl quadrato con scelta +exsr 2righe
FCNA1Z4W cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds(DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
* Annota subfile page.
* Uguaile al contenuto della keyword SFLPAG(nnnn) del display file.
```

#### 26.2.3.2. Imposta il massimo valore possibile di SFLPAG.

|                                                             |   |              |              |
|-------------------------------------------------------------|---|--------------|--------------|
| D SFLPAG                                                    | c | const(0009)  |              |
| * Relative record number del subfile.                       |   |              |              |
| D S1RRN                                                     | S | 5 0          |              |
| * Numero di record nel subfile iniziale e finale.           |   |              |              |
| D S1RRNBEG                                                  | S | like(S1RRN)  |              |
| D S1RRNEND                                                  | S | like(S1RRN)  |              |
| * Indice di loop durante la lettura del subfile.            |   |              |              |
| D XX                                                        | S | like(S1RRN)  |              |
| *-----                                                      |   |              |              |
| * Definisce prima chiave presentata sulla pagina corrente.  |   |              |              |
| D KFA                                                       | s | like(A1CDPE) | KeyFirstAnte |
| D KFP                                                       | s | like(A1CDPE) | KeyFirstPost |
| * Definisce ultima chiave presentata sulla pagina corrente. |   |              |              |
| D KLA                                                       | s | like(A1CDPE) | KeyLastAnte  |
| D KLP                                                       | s | like(A1CDPE) | KeyLastPost  |
| *-----                                                      |   |              |              |
| * Comodo per risposta dsply.                                |   |              |              |
| D X                                                         | s | 1            |              |



\*-----

### 26.2.3.3. Per condizionare i comportamenti delle subroutine di riempimento della pagina.

|                                                                  |       |       |
|------------------------------------------------------------------|-------|-------|
| * Flag di servizio per il caricamento delle pagine.              |       |       |
| D PRIMA                                                          | s     | n     |
| D DOPO                                                           | s     | n     |
| *-----                                                           |       |       |
| * Predispone chiusura.                                           |       |       |
| C                                                                | seton | lr    |
| * Esegue il video di controllo.                                  |       |       |
| C                                                                | exsr  | SRC1  |
| *-----                                                           |       |       |
| * Esegue il video di controllo.                                  |       |       |
| C                                                                | SRC1  | begsr |
| * Riempie la pagina in avanti posizionandosi prima della chiave. |       |       |

### 26.2.3.4. Pagina successiva ora riceve il flag "PRIMA".

|                            |       |           |
|----------------------------|-------|-----------|
| C                          | eval  | PRIMA=*on |
| C                          | exsr  | PGDW      |
| * Cicla intorno al video.  |       |           |
| C                          | do    | *hival    |
| * Emette il subfile.       |       |           |
| C                          | exsr  | EXC1      |
| *----                      |       |           |
| * Se richiesto, abbandona. |       |           |
| C 03                       | leave |           |
| *----                      |       |           |

### 26.2.3.5. Gestisce il comando di rinfrescamento della pagina corrente. Per farlo, chiama la subroutine pagina precedente passandole il flag "DOPO" o la subroutine pagina successiva passandole il flag "PRIMA".

|                                                                  |       |                 |
|------------------------------------------------------------------|-------|-----------------|
| * Se richiesto aggiornamento (refresh).                          |       |                 |
| C                                                                | if    | *in05           |
| * Se l'ultima riga e' vuota.                                     |       |                 |
| C                                                                | if    | S1RRNEND<SFLPAG |
| * Forza le chiavi after ultima pagina vista.                     |       |                 |
| C                                                                | eval  | KFA=KLP         |
| C                                                                | eval  | KLA=*blank      |
| * Riempie la pagina indietro posizionandosi dopo la chiave.      |       |                 |
| C                                                                | eval  | DOPO=*on        |
| C                                                                | exsr  | PGUP            |
| * Se l'ultima riga non e' vuota.                                 |       |                 |
| C                                                                | else  |                 |
| * Forza le chiavi after ultima pagina vista.                     |       |                 |
| C                                                                | eval  | KFA=*blank      |
| C                                                                | eval  | KLA=KFP         |
| * Riempie la pagina in avanti posizionandosi prima della chiave. |       |                 |
| C                                                                | eval  | PRIMA=*on       |
| C                                                                | exsr  | PGDW            |
| * Se l'ultima riga non e' vuota.                                 |       |                 |
| C                                                                | endif |                 |
| * Ricicla.                                                       |       |                 |
| C                                                                | iter  |                 |
| * Se richiesto aggiornamento (refresh).                          |       |                 |
| C                                                                | endif |                 |
| *----                                                            |       |                 |
| * Se richiesto posizionamento.                                   |       |                 |
| C                                                                | if    | *in06           |
| * Forza le chiavi after ultima pagina vista.                     |       |                 |
| C                                                                | clear | KFA             |

### 26.2.3.6. Cambiato nome alla chiave di riposizionamento.

|   |      |             |
|---|------|-------------|
| C | eval | KLA=S1CDPES |
|---|------|-------------|

### 26.2.3.7. Pagina successiva ora riceve il flag "PRIMA".

|                                                                  |       |           |
|------------------------------------------------------------------|-------|-----------|
| * Riempie la pagina in avanti posizionandosi prima della chiave. |       |           |
| C                                                                | eval  | PRIMA=*on |
| C                                                                | exsr  | PGDW      |
| * Ricicla.                                                       |       |           |
| C                                                                | iter  |           |
| * Se richiesto posizionamento.                                   |       |           |
| C                                                                | endif |           |
| *----                                                            |       |           |
| * Esamina ed esegue scelte.                                      |       |           |

### 26.2.3.8. Evita la subroutine delle scelte se la pagina è vuota.

|                                   |       |                |
|-----------------------------------|-------|----------------|
| C                                 | if    | S1RRNEND>*zero |
| C                                 | exsr  | SCS1           |
| C 60                              | iter  |                |
| C                                 | endif |                |
| *----                             |       |                |
| * Se richiesta pagina precedente. |       |                |
| C                                 | if    | *in25          |

```
* Riempie la pagina all'indietro.
```

#### 26.2.3.9. Pagina precedente ora riceve il flag "DOPO".

```
C eval DOPO=*off
C exsr PGUP
* Ricicla.
C iter
* Se richiesta pagina precedente.
C endif
*----
* Se richiesta pagina successiva.
C if *in26
```

#### 26.2.3.10. Pagina successiva ora riceve il flag "PRIMA".

```
* Riempie la pagina in avanti posizionandosi dopo la chiave.
C eval PRIMA=*off
C exsr PGDW
* Ricicla.
C iter
* Se richiesta pagina successiva.
C endif
*----
* Ricicla.
C iter
*----
* Cicla intorno al video.
C enddo
C endsr
*-----
* Riempie la pagina in avanti.
* Riceve KFA chiave del primo record sulla pagina vista prima.
* Riceve KLA chiave dell'ultimo record sulla pagina vista prima.
```

#### 26.2.3.11. Commenta il parametro che regola ora il posizionamento prima o dopo la chiave di riferimento.

```
* Riceve PRIMA=*on per posizionarsi
* prima dell'ultimo record della pagina vista prima.
* Riceve PRIMA=*off per posizionarsi
* dopo l'ultimo record della pagina vista prima.
* Carica la pagina di subfile leggendo in avanti da prima o dopo KLA.
* Ritorna KFP chiave del primo record sulla pagina caricata.
* Ritorna KLP chiave dell'ultimo record sulla pagina caricata.
C PGDW begsr
* Pulisce le chiavi di pagina post.
C clear KFP
C clear KLP
* Azzerà primo e ultimo RRN della pagina.
C clear S1RRNBEG
C clear S1RRNEND
* Azzerà il numero relativo di record di scrittura.
C clear S1RRN
* Pulisce il subfile.
C exsr CLS1
```

#### 26.2.3.12. Il posizionamento è regolato ora dal flag dedicato "PRIMA" e non più dall'indicatore 06 di ricaricamento.

```
* Se corre posizionamento prima della chiave.
C if PRIMA
* Si posiziona prima dell'ultimo record della pagina vista prima.
C KLA settl A1
* Cerca il record precedente la chiave.
C readp A1
* Se non c'e', disabilita PageUp.
C if %eof
C setoff 55 pageup
C endif
* Si riposiziona prima dell'ultimo record della pagina vista prima.
C KLA settl A1
* Se corre posizionamento dopo la chiave.
C else
* Si posiziona dopo l'ultimo record della pagina vista prima.
C KLA setgt A1
* Se corre posizionamento dopo la chiave.
C endif
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il prossimo record.
C read A1
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C setoff 56 pagedown
C leave
C endif
* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
```

|   |                                                        |      |
|---|--------------------------------------------------------|------|
| * | Trascrive il record di database nella riga di subfile. |      |
| C | exsr                                                   | DBS1 |

#### 26.2.3.13. Aggiunta la decodifica dei codici tabellari sulla riga.

|       |                                                               |                                                         |
|-------|---------------------------------------------------------------|---------------------------------------------------------|
| *     | Decodifica la riga di subfile.                                |                                                         |
| C     | exsr                                                          | DES1                                                    |
| *     | Se non ancora fatto,                                          |                                                         |
| *     | accantona la chiave corrente come prima della pagina.         |                                                         |
| C     | if                                                            | KFP=*blank                                              |
| C     | eval                                                          | KFP=AICDPE                                              |
| C     | eval                                                          | S1RRNBEG=S1RRN                                          |
| C     | endif                                                         |                                                         |
| *     | Accantona sempre la chiave corrente come ultima della pagina. |                                                         |
| C     | eval                                                          | KLP=AICDPE                                              |
| C     | eval                                                          | S1RRNEND=S1RRN                                          |
| *     | Scrive la riga di subfile.                                    |                                                         |
| C     | write                                                         | S1                                                      |
| *     | Se fine pagina, abbandona.                                    |                                                         |
| C     | if                                                            | S1RRN>=SFLPAG                                           |
| C     | leave                                                         |                                                         |
| C     | endif                                                         |                                                         |
| *     | Cicla sulla lettura del file per riempire il subfile.         |                                                         |
| C     | enddo                                                         |                                                         |
| *     | Se PageDown e' permesso.                                      |                                                         |
| C     | if                                                            | *in56                                                   |
| *     | Cerca il primo record della pagina successiva.                |                                                         |
| C     | read                                                          | A1                                                      |
| *     | Se non c'e', disabilita PageDown.                             |                                                         |
| C     | if                                                            | %eof                                                    |
| C     | setoff                                                        | 56 pagedown                                             |
| C     | endif                                                         |                                                         |
| *     | Se PageDown e' permesso.                                      |                                                         |
| C     | endif                                                         |                                                         |
| *     | Ribalta le chiavi post nelle chiavi ante.                     |                                                         |
| C     | eval                                                          | KFA=KFP                                                 |
| C     | eval                                                          | KLA=KLP                                                 |
| C     | endsr                                                         |                                                         |
| ----- |                                                               |                                                         |
| *     | Riempie la pagina all'indietro.                               |                                                         |
| *     | Riceve                                                        | KFA chiave del primo record sulla pagina vista prima.   |
| *     | Riceve                                                        | KLA chiave dell'ultimo record sulla pagina vista prima. |

#### 26.2.3.14. Commenta il parametro che regola ora il posizionamento prima o dopo la chiave di riferimento.

|   |                                                                           |                                                      |
|---|---------------------------------------------------------------------------|------------------------------------------------------|
| * | Riceve DOPO =*on per posizionarsi                                         |                                                      |
| * | dopo il primo record della pagina vista prima.                            |                                                      |
| * | Riceve DOPO =*off per posizionarsi                                        |                                                      |
| * | prima del primo record della pagina vista prima.                          |                                                      |
| * | Carica la pagina di subfile leggendo all'indietro da dopo o prima di KFA. |                                                      |
| * | Ritorna                                                                   | KFP chiave del primo record sulla pagina caricata.   |
| * | Ritorna                                                                   | KLP chiave dell'ultimo record sulla pagina caricata. |
| C | PGUP                                                                      | begsr                                                |
| * | Pulisce le chiavi di pagina post.                                         |                                                      |
| C | clear                                                                     | KFP                                                  |
| C | clear                                                                     | KLP                                                  |

#### 26.2.3.15. Se manca la chiave di posizionamento, assume la fine del file.

|   |                                                                       |                           |
|---|-----------------------------------------------------------------------|---------------------------|
| * | Se chiavi after vuote, *hival.                                        |                           |
| C | if                                                                    | KFA=*blank and KLA=*blank |
| C | eval                                                                  | KFA=*hival                |
| C | endif                                                                 |                           |
| * | Imposta a subfile page + 1 il numero relativo di record di scrittura. |                           |
| C | eval                                                                  | S1RRN=SFLPAG+1            |
| * | Pulisce il subfile.                                                   |                           |
| C | exsr                                                                  | CLS1                      |

#### 26.2.3.16. Aggiunto posizionamento dopo chiave per soddisfare la casistica dovuta alla richiesta di rinfrescamento.

|   |                                                               |             |
|---|---------------------------------------------------------------|-------------|
| * | Se corre posizionamento dopo la chiave.                       |             |
| C | if                                                            | DOPO        |
| * | Si posiziona dopo il primo record della pagina vista prima.   |             |
| C | KFA                                                           | setgt A1    |
| * | Cerca il record successivo alla chiave.                       |             |
| C | read                                                          | A1          |
| * | Se non c'e', disabilita PageDown.                             |             |
| C | if                                                            | %eof        |
| C | setoff                                                        | 56 pagedown |
| C | endif                                                         |             |
| * | Si riposiziona dopo il primo record della pagina vista prima. |             |
| C | KFA                                                           | setgt A1    |
| * | Se corre posizionamento prima della chiave.                   |             |
| C | else                                                          |             |
| * | Si posiziona prima del primo record della pagina vista prima. |             |
| C | KFA                                                           | setll A1    |
| * | Se corre posizionamento prima della chiave.                   |             |

|                                                          |        |           |           |
|----------------------------------------------------------|--------|-----------|-----------|
| C                                                        | endif  |           |           |
| * Cicla sulla lettura del file per riempire il subfile.  |        |           |           |
| C                                                        | do     | *hival    |           |
| * Legge il record precedente.                            |        |           |           |
| C                                                        | readp  | A1        |           |
| * Se non ce ne sono altri, abbandona lettura.            |        |           |           |
| C                                                        | if     | %eof      |           |
| C                                                        | setoff |           | 55 pageup |
| C                                                        | leave  |           |           |
| C                                                        | endif  |           |           |
| * Decrementa il numero relativo di record di scrittura.  |        |           |           |
| C                                                        | eval   | S1RRN -=1 |           |
| * Trascrive il record di database nella riga di subfile. |        |           |           |
| C                                                        | exsr   | DBS1      |           |

#### 26.2.3.17. Aggiunta la decodifica dei codici tabellari sulla riga.

|                                                                |        |                |                       |
|----------------------------------------------------------------|--------|----------------|-----------------------|
| * Decodifica la riga di subfile.                               |        |                |                       |
| C                                                              | exsr   | DES1           |                       |
| * Se non ancora fatto,                                         |        |                |                       |
| * accantona la chiave corrente come ultima della pagina.       |        |                |                       |
| C                                                              | if     | KLP=*blank     |                       |
| C                                                              | eval   | KLP=A1CDPE     |                       |
| C                                                              | eval   | S1RRNEND=S1RRN |                       |
| C                                                              | endif  |                |                       |
| * Accantona sempre la chiave corrente come prima della pagina. |        |                |                       |
| C                                                              | eval   | KFP=A1CDPE     |                       |
| C                                                              | eval   | S1RRNBEG=S1RRN |                       |
| * Scrive la riga di subfile.                                   |        |                |                       |
| C                                                              | write  | S1             |                       |
| * Se inizio pagina, abbandona.                                 |        |                |                       |
| C                                                              | if     | S1RRN=1        |                       |
| C                                                              | leave  |                |                       |
| C                                                              | endif  |                |                       |
| * Cicla sulla lettura del file per riempire il subfile.        |        |                |                       |
| C                                                              | enddo  |                |                       |
| * Se PageUp e' permesso.                                       |        |                |                       |
| C                                                              | if     | *in55          |                       |
| * Cerca l'ultimo record della pagina precedente.               |        |                |                       |
| C                                                              | readp  | A1             |                       |
| * Se non c'e', disabilita PageUp.                              |        |                |                       |
| C                                                              | if     | %eof           |                       |
| C                                                              | setoff |                | 55 pageup             |
| C                                                              | endif  |                |                       |
| * Se PageUp e' permesso.                                       |        |                |                       |
| C                                                              | endif  |                |                       |
| * Ribalta le chiavi post nelle chiavi ante.                    |        |                |                       |
| C                                                              | eval   | KFA=KFP        |                       |
| C                                                              | eval   | KLA=KLP        |                       |
| C                                                              | endsr  |                |                       |
| -----                                                          |        |                |                       |
| * Pulisce il subfile.                                          |        |                |                       |
| C                                                              | CLS1   | begsr          |                       |
| * Spegne.                                                      |        |                |                       |
| C                                                              | exsr   | OFF            |                       |
| * Prenota la pulizia del subfile.                              |        |                |                       |
| C                                                              | seton  |                | 52 sflclr             |
| * Inibisce visualizzazione di controllo e subfile.             |        |                |                       |
| C                                                              | setoff |                | 5354 sfldspctl sfldsp |
| * Emette il controllo per la sola pulizia.                     |        |                |                       |

#### 26.2.3.18. Cambiato nome al record video.

|                                                   |        |                            |              |
|---------------------------------------------------|--------|----------------------------|--------------|
| C                                                 | write  | S1C                        |              |
| * Sprenota la pulizia del subfile.                |        |                            |              |
| C                                                 | setoff |                            | 52 sflclr    |
| * Permette pageup e pagedown.                     |        |                            |              |
| C                                                 | seton  |                            | 55 pageup    |
| C                                                 | seton  |                            | 56 pagedown  |
| C                                                 | endsr  |                            |              |
| -----                                             |        |                            |              |
| * Emette il subfile.                              |        |                            |              |
| C                                                 | EXC1   | begsr                      |              |
| * Prenota emissione del controllo.                |        |                            |              |
| C                                                 | seton  |                            | 53 sfldspctl |
| * Se subfile pieno (rrn compreso tra 1 e SFLPAG), |        |                            |              |
| * ne prenota emissione.                           |        |                            |              |
| C                                                 | setoff |                            | 54 sfldsp    |
| C                                                 | if     | S1RRN>=1 and S1RRN<=SFLPAG |              |
| C                                                 | seton  |                            | 54 sfldsp    |
| C                                                 | endif  |                            |              |
| * Emette intestazione.                            |        |                            |              |

#### 26.2.3.1. Cambiato nome ai record video.

|                                                                  |       |     |  |
|------------------------------------------------------------------|-------|-----|--|
| C                                                                | write | S1T |  |
| * Se subfile vuoto, ricopre area di visualizzazione del subfile. |       |     |  |

```

C N54 write S1B sflblank
* Emette controllo e, se pieno, subfile.
C exfmt S1C
C endsr

* Esamina ed esegue scelte.
* Legge piu' volte tutti i record del subfile con un loop di chain
* per trattare le scelte.
C SCS1 begsr
* Esegue.
C do
*----
* ELIMINA REVERSE IMAGE E POSITION CURSOR DALLE RIGHE
* Spegne.
C exsr OFF
* Cicla sulle righe di subfile.
C S1RRNBEG do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Elimina dalla riga reverse image e position cursor.
C update S1
* Cicla sulle righe di subfile.
C enddo
*----
* PREPARA LA SEGNALEZIONE DI SCELTE NON PREVISTE SULLE RIGHE
* Assume non ancora trovata scelta qualsiasi.
C setoff 99
* Cicla sulle righe di subfile.
C S1RRNBEG do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Se la riga porta una scelta qualsiasi, lo annota.
C if S1SCE<>*blank
C seton 99
C endif
* Se la riga porta una scelta non prevista,
* prenota PC, RI ed errore.
C if S1SCE<>*blank and S1SCE<>'X' and S1SCE<>'Y'
C seton 5859
C seton 6061
C endif
* Se corre l'ultima riga, non ci sono scelte e non sta paginando,
* prenota PC ed errore.
C if XX=S1RRNEND and not *in99
C and not (*in25 or *in26)
C seton 58
C seton 6064
C endif
* Se la riga porta una scelta mentre e' richiesta paginazione,
* prenota PC, RI ed errore.
C if S1SCE<>*blank and (*in25 or *in26)
C seton 5859
C seton 6063
C endif
* Se corre richiesta di PC o RI,
* ricalca la riga in errore per PC e RI e li spegne.
C if *in58 or *in59
C update S1
C setoff 5859
C endif
* Cicla sulle righe di subfile.
C enddo
*----
* Se errore, abbandona.
C 60 leave
*----
* ESEGUE LE SCELTE PREVISTE SULLE RIGHE
* Cicla sulle righe di subfile.
C S1RRNBEG do S1RRNEND XX
* Legge una riga.
C XX chain S1
* Se corre una scelta valida.
C if S1SCE<>*blank
* Esegue la scelta.
C dsply (S1SCE+' '+S1CDPE) ' ' x;
* Trasforma scelta in "eseguita",
C eval S1SCE='>'
* Prenota PC.
C seton 58
* Finge errore solo per non ricaricare il sfl e segnalare.
C seton 6062
* Ricalca il record per PC.
C update S1
* Spegne PC e RI.

```

```

C setoff 5859
* Se corre una scelta valida.
C endif
* Cicla sulle righe di subfile.
C enddo
*---
* Esegue.
C enddo
C endsr
*-----
* Spegne.
C OFF begsr
* Spegne position cursor e reverse image sulla riga.
C setoff 5859
* Spegne errori e segnalazioni su controllo.
C setoff 606162
C setoff 6364
C endsr
*-----
* Trascrive il record di database nella riga di subfile.
C DBS1 begsr
* Scelta in bianco.
C eval S1SCE =*blank
* Dati.
C eval S1CDPE=A1CDPE
C eval S1DEPE=A1DEPE
C eval S1PRNA=A1PRNA
C eval S1PRRE=A1PRRE
C endsr
*-----

```

#### 26.2.3.2. Subroutine di decodifica.

```

* Decodifica la riga di subfile.
C DES1 begsr
* Decodifica provincia di nascita.
C clear S1PRNAD
C if S1PRNA<>*blank
C eval S1PRNAD='?'
C S1PRNA chain T1
C if %found
C eval S1PRNAD=T1DCPR
C endif
C endif
* Decodifica provincia di residenza.
C clear S1PRRED
C if S1PRRE<>*blank
C eval S1PRRED='?'
C S1PRRE chain T1
C if %found
C eval S1PRRED=T1DCPR
C endif
C endif
C endsr
*-----

```

#### 26.2.3.3. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z4E.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1Z4E.txt)

#### 26.2.4. Funzionamento del programma CNA1Z4E.

- 26.2.4.1. Usa la logica di subfile quadrato della versione CNA1Z3E.
- 26.2.4.2. Ogni riga di subfile occupa due righe di video.
- 26.2.4.3. La testata occupa due righe video di intestazione per i campi della riga di subfile.
- 26.2.4.4. La riga di subfile contiene anche le decodifiche dei codici visualizzati.
- 26.2.4.5. La pagina di subfile contiene abbastanza record da riempire le righe video a disposizione.
- 26.2.4.6. Aggiunto il comando di rinfrescamento F5.

Se la pagina rinfrescata è piena, ricarica in avanti dal primo record già visto in prima riga.

Se la pagina rinfrescata manca di almeno una riga in fondo, ricarica all'indietro dall'ultimo record del file.

Se la pagina rinfrescata manca di almeno una riga in testa, ricarica in avanti dal primo record del file.

## 27. Subfile quadrato con scelta in finestra

27.1. La prossima versione implementa lo schema del subfile quadrato per usarlo come finestra di scelta di un codice tra quelli elencati, in luogo della digitazione diretta su un video di immissione o modifica che lo richiede. Alla chiamata con un comando dedicato (F2) mentre il cursore 5250 si trova su un certo campo, il programma chiamante invoca la finestra adeguata al campo e permette la scelta di un contenuto per il campo stesso.

### 27.2. Elenco anagrafico su subfile quadrato con scelta su finestra (oggetti CNT1WIN\*).

27.2.1. Sorgente commentato del video CNT1WINW Dspf. Ottenuto modificando la versione CNA1Z3W.

27.2.2. Il subfile del video corrente si presenta in una finestra che permette la scelta di un singolo codice che viene trascritto nel campo.

|                                               |                                      |
|-----------------------------------------------|--------------------------------------|
| * Claudio Neroni 31-03-2021 Creato per Corso. |                                      |
| * CNT1WINW                                    |                                      |
| * Window provincie                            |                                      |
| *-----                                        |                                      |
| A                                             | DSPSIZ(24 80 *DS3)                   |
| A                                             | CHGINPDFT                            |
| A                                             | INDARA                               |
| A                                             | ERRSFL                               |
| A                                             | INDTXT(52 'Subfile Clear')           |
| A                                             | INDTXT(53 'Subfile Display Control') |
| A                                             | INDTXT(54 'Subfile Display')         |
| A                                             | INDTXT(55 'PageUp permesso')         |
| A                                             | INDTXT(56 'PageDown permesso')       |
| A                                             | INDTXT(58 'Subfile Row +             |
| A                                             | Position Cursor')                    |
| A                                             | INDTXT(59 'Subfile Row +             |
| A                                             | Reverse Image')                      |
| *-----                                        |                                      |
| A                                             | R T1                                 |
| A                                             | TEXT('Testata')                      |
| A                                             | OVERLAY                              |

27.2.2.1. Il record Testata comparirà in finestra.

|   |                                   |
|---|-----------------------------------|
| A | WINDOW(*DFT 10 55 *MSGLIN)        |
| A | WDWBORDER((*DSPATR RI) (*CHAR ' - |
| A | '))                               |
| A | WDWTITLE((*TEXT &INTESTA))        |

27.2.2.2. Nella cornice della finestra comparirà il seguente campo di intestazione.

|   |         |    |   |                               |
|---|---------|----|---|-------------------------------|
| A | INTESTA | 50 | P | TEXT('Intestazione finestra') |
|---|---------|----|---|-------------------------------|

27.2.2.3. Cambiano le intestazioni di colonna.

|        |       |                    |                 |
|--------|-------|--------------------|-----------------|
| A      | 1     | 1'S Cd Descrizione | -               |
| A      |       |                    |                 |
| A      |       | DSPATR(HI)         |                 |
| A      | 2     | 1'c pr provincia   | -               |
| A      |       |                    |                 |
| A      |       | DSPATR(HI)         |                 |
| A      |       | DSPATR(UL)         |                 |
| *----- |       |                    |                 |
| A      | R S1  | SFL                |                 |
| A      |       | TEXT('Riga')       |                 |
| A      | SISCE | 1 B 3              | 1TEXT('Scelta') |
| A      |       | DSPATR(UL)         |                 |
| A      | 58    | DSPATR(PC)         |                 |
| A      | 59    | DSPATR(RI)         |                 |

27.2.2.4. Cambiano i campi della riga.

|        |      |   |                       |
|--------|------|---|-----------------------|
| A      | SICD | R | +1REFFLD(T1CDPR CNT1) |
| A      | SIDE | R | +1REFFLD(T1DEPR CNT1) |
| *----- |      |   |                       |
| A      | R C1 |   | SFLCTL(S1)            |

27.2.2.5. Il record Controllo comparirà nella stessa finestra del record Testata.

|   |       |                                   |
|---|-------|-----------------------------------|
| A |       | WINDOW(T1)                        |
| A |       | SFLSIZ(0005)                      |
| A |       | SFLPAG(0005)                      |
| A |       | TEXT('Controllo')                 |
| A |       | OVERLAY                           |
| A | 52    | SFLCLR                            |
| A | 53    | SFLDSPCTL                         |
| A | 54    | SFLDSP                            |
| A | N56   | SFLEND                            |
| A |       | CA03(03 'Fine')                   |
| A |       | CF06(06 'Da')                     |
| A | 55    | PAGEUP(25 'Pagina precedente')    |
| A | 56    | PAGEDOWN(26 'Pagina successiva')  |
| A | 60 61 | SFLMSG('Scelta diversa da X e Y') |

|   |    |    |                                |
|---|----|----|--------------------------------|
| A | 60 | 62 | SFLMSG('">"=Scelta eseguita')  |
| A | 60 | 63 | SFLMSG('Scelte e paginazioni + |
| A |    |    | incompatibili')                |
| A | 60 | 64 | SFLMSG('Nessuna scelta')       |

#### 27.2.2.6. Aggiungo un errore di scelta "X" non unica.

|        |     |      |                                     |
|--------|-----|------|-------------------------------------|
| A      | 60  | 65   | SFLMSG('La scelta X=RitornaCodice + |
| A      |     |      | deve essere unica')                 |
| A      |     | C1CD | R B 8 3REFFLD(T1CDPR CNT1)          |
| A      |     |      | TEXT('Start code')                  |
| A      |     |      | DSPATR(HI)                          |
| A      |     |      | DSPATR(CS)                          |
| A      | N60 |      | DSPATR(PC)                          |
| A      |     |      | +1'<--dal codice'                   |
| A      |     |      | 9 1'F3=Fine'                        |
| A      |     |      | +2'F6=Da'                           |
| A      | 55  |      | +2'PGUP=Prec'                       |
| A      | 56  |      | +2'PGDW=Succ'                       |
| *----- |     |      |                                     |
| A      |     | R B1 |                                     |
| A      |     |      | TEXT('Bianco')                      |

#### 27.2.2.1. Il record Bianco comparirà nella stessa finestra del record Testata.

|        |  |  |                                   |
|--------|--|--|-----------------------------------|
| A      |  |  | WINDOW(T1)                        |
| A      |  |  | OVERLAY                           |
| A      |  |  | 4 2' (Nessun record da mostrare)' |
| *----- |  |  |                                   |

#### 27.2.2.2. Il record Assume, definito ma non usato esplicitamente dal programma e con la sua sola presenza nel file, permette alla finestra di sovrapporsi all'immagine presente sul video al momento dell'apertura del display file.

|        |  |          |                 |
|--------|--|----------|-----------------|
| A      |  | R ASSUME |                 |
| A      |  |          | TEXT('Assume.') |
| A      |  |          | ASSUME          |
| A      |  | ASSUME1  | 1 B 1 2         |
| *----- |  |          |                 |

#### 27.2.2.3. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNT1WINW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNT1WINW.txt)

#### 27.2.3. Sorgente commentato del programma CNT1WIN Rpgle. Ottenuto modificando la versione CNA1Z3E.

```
*FREE * Claudio Neroni 30-03-2021 Creato per Corso.
* CNT1WIN
* Window provincie
*-----
```

#### 27.2.3.1. Il programma lavora sulla tabella provincie. Per duplicare in futuro su ulteriori tabelle, rinomina il record con un nome generico.

|                                |    |   |                       |
|--------------------------------|----|---|-----------------------|
| * Tabella provincie per codice |    |   |                       |
| FCNT11                         | if | e | k disk rename(T1:DBR) |

#### 27.2.3.2. Finestra provincie.

```
* Window provincie
FCNT1WINW cf e workstn
* Associa il relative record number al record di subfile.
F sfile(S1:S1RRN)
* Display Specific Feedback Information
F infds(DSPFBK)
*-----
* Display Specific Feedback Information
/COPY CORSO,DSPFBK
*-----
* Annota subfile page.
* Uguale al contenuto della keyword SFLPAG(nnnn) del display file.
D SFLPAG c const(0005)
* Relative record number del subfile.
D S1RRN S 5 0
* Numero di record nel subfile iniziale e finale.
D S1RRNBEG S like(S1RRN)
D S1RRNEND S like(S1RRN)
* Indice di loop durante la lettura del subfile.
D XX S like(S1RRN)
*-----
* Definisce prima chiave presentata sulla pagina corrente.
```

#### 27.2.3.3. Punta le definizioni like sul codice rinominato col nome generico.

|                                                             |   |              |              |
|-------------------------------------------------------------|---|--------------|--------------|
| D KFA                                                       | s | like(DBRCD ) | KeyFirstAnte |
| D KFP                                                       | s | like(DBRCD ) | KeyFirstPost |
| * Definisce ultima chiave presentata sulla pagina corrente. |   |              |              |
| D KLA                                                       | s | like(DBRCD ) | KeyLastAnte  |
| D KLP                                                       | s | like(DBRCD ) | KeyLastPost  |
| *-----                                                      |   |              |              |

#### 27.2.3.4. Definisce il parametro sul codice rinominato.



27.2.3.5. Rinomina codice e descrizione elemento tabella provincie per usare lo stesso nome nei futuri duplicati.

27.2.3.6. Utilizza il codice ricevuto per posizionare l'elenco e lo pulisce per prepararne la restituzione in caso di scelta.

27.2.3.7. Usa nome generico del campo di posizionamento.

|     |                                   |                     |
|-----|-----------------------------------|---------------------|
| 202 | Corso-As400-2023-Manuale-V03.docx | 07/02/2023 18:51:00 |
|-----|-----------------------------------|---------------------|

```

* Ricicla.
C iter
* Se richiesta pagina successiva.
C endif
*----
* Ricicla.
C iter
*----
* Cicla intorno al video.
C enddo
C endsr
*-----
* Riempie la pagina in avanti.
* Riceve KFA chiave del primo record sulla pagina vista prima.
* Riceve KLA chiave dell'ultimo record sulla pagina vista prima.
* Carica la pagina di subfile leggendo in avanti da dopo KLA.
* Ritorna KFP chiave del primo record sulla pagina caricata.
* Ritorna KLP chiave dell'ultimo record sulla pagina caricata.
C PGDW begsr
* Pulisce le chiavi di pagina post.
C clear KFP
C clear KLP
* Azzerà primo e ultimo RRN della pagina.
C clear S1RRNBEG
C clear S1RRNEND
* Azzerà il numero relativo di record di scrittura.
C clear S1RRN
* Pulisce il subfile.
C exsr CLS1
* Se corre posizionamento.
C if *in06
* Si posiziona prima della chiave di posizionamento.

```

#### 27.2.3.8. Usa nome generico di record dati.

```

C KLA setll DBR
* Cerca il record precedente la chiave.
C readp DBR
* Se non c'è, disabilita PageUp.
C if %eof
C setoff 55 pageup
C endif
* Si riposiziona prima della chiave di posizionamento.
C KLA setll DBR
* Se corre paginazione.
C else
* Si posiziona oltre l'ultimo record della pagina vista prima.
C KLA setgt DBR
* Se corre paginazione.
C endif
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il prossimo record.
C read DBR
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C setoff 56 pagedown
C leave
C endif
* Incrementa il numero relativo di record di scrittura.
C eval S1RRN +=1
* Trascrive il record di database nella riga di subfile.
C exsr DBS1
* Se non ancora fatto,
* accantona la chiave corrente come prima della pagina.
C if KFP=*blank

```

#### 27.2.3.9. Usa nome generico del campo di codice.

```

C eval KFP=DBRCD
C eval S1RRNBEG=S1RRN
C endif
* Accantona sempre la chiave corrente come ultima della pagina.
C eval KLP=DBRCD
C eval S1RRNEND=S1RRN
* Scrive la riga di subfile.
C write S1
* Se fine pagina, abbandona.
C if S1RRN>=SFLPAG
C leave
C endif
* Cicla sulla lettura del file per riempire il subfile.
C enddo
* Se PageDown è permesso.
C if *in56
* Cerca il primo record della pagina successiva.

```

```

C read DBR
* Se non c'e', disabilita PageDown.
C if %eof
C setoff 56 pagedown
C endif
* Se PageDown e' permesso.
C endif
* Ribalta le chiavi post nelle chiavi ante.
C eval KFA=KFP
C eval KLA=KLP
C endsr
*-----
* Riempie la pagina all'indietro.
* Riceve KFA chiave del primo record sulla pagina vista prima.
* Riceve KLA chiave dell'ultimo record sulla pagina vista prima.
* Carica la pagina di subfile leggendo all'indietro da prima di KFA.
* Ritorna KFP chiave del primo record sulla pagina caricata.
* Ritorna KLP chiave dell'ultimo record sulla pagina caricata.
C PGUP begsr
* Pulisce le chiavi di pagina post.
C clear KFP
C clear KLP
* Imposta a subfile page + 1 il numero relativo di record di scrittura.
C eval S1RRN=SFLPAG+1
* Pulisce il subfile.
C exsr CLS1
* Si posiziona prima del primo record della pagina vista prima.
C KFA setll DBR
* Cicla sulla lettura del file per riempire il subfile.
C do *hival
* Legge il record precedente.
C readp DBR
* Se non ce ne sono altri, abbandona lettura.
C if %eof
C setoff 55 pageup
C leave
C endif
* Decrementa il numero relativo di record di scrittura.
C eval S1RRN --=1
* Trascrive il record di database nella riga di subfile.
C exsr DBS1
* Se non ancora fatto,
* accantona la chiave corrente come ultima della pagina.
C if KLP=*blank
C eval KLP=DBRCD
C eval S1RRNEND=S1RRN
C endif
* Accantona sempre la chiave corrente come prima della pagina.
C eval KFP=DBRCD
C eval S1RRNBEG=S1RRN
* Scrive la riga di subfile.
C write S1
* Se inizio pagina, abbandona.
C if S1RRN=1
C leave
C endif
* Cicla sulla lettura del file per riempire il subfile.
C enddo
* Se PageUp e' permesso.
C if *in55
* Cerca l'ultimo record della pagina precedente.
C readp DBR
* Se non c'e', disabilita PageUp.
C if %eof
C setoff 55 pageup
C endif
* Se PageUp e' permesso.
C endif
* Ribalta le chiavi post nelle chiavi ante.
C eval KFA=KFP
C eval KLA=KLP
C endsr
*-----
* Pulisce il subfile.
C CLS1 begsr
* Spegne.
C exsr OFF
* Prenota la pulizia del subfile.
C seton 52 sflclr
* Inibisce visualizzazione di controllo e subfile.
C setoff 5354 sfldspctl sfldsp
* Emette il controllo per la sola pulizia.
C write C1
* Sprenota la pulizia del subfile.

```

|                                                                    |                               |    |           |
|--------------------------------------------------------------------|-------------------------------|----|-----------|
| C                                                                  | setoff                        | 52 | sflclr    |
| * Permette pageup e pagedown.                                      |                               |    |           |
| C                                                                  | seton                         | 55 | pageup    |
| C                                                                  | seton                         | 56 | pagedown  |
| C                                                                  | endsr                         |    |           |
| -----                                                              |                               |    |           |
| * Emette il subfile.                                               |                               |    |           |
| C                                                                  | EXC1 begsr                    |    |           |
| * Prenota emissione del controllo.                                 |                               |    |           |
| C                                                                  | seton                         | 53 | sfldspctl |
| * Se subfile pieno (rrn compreso tra 1 e SFLPAG),                  |                               |    |           |
| * ne prenota emissione.                                            |                               |    |           |
| C                                                                  | setoff                        | 54 | sfldsp    |
| C                                                                  | if S1RRN>=1 and S1RRN<=SFLPAG |    |           |
| C                                                                  | seton                         | 54 | sfldsp    |
| C                                                                  | endif                         |    |           |
| * Emette intestazione.                                             |                               |    |           |
| C                                                                  | write T1                      |    |           |
| * Se subfile vuoto, ricopre area di visualizzazione del subfile.   |                               |    |           |
| C                                                                  | N54 write B1                  |    | sflblank  |
| * Emette controllo e, se pieno, subfile.                           |                               |    |           |
| C                                                                  | exfmt C1                      |    |           |
| C                                                                  | endsr                         |    |           |
| -----                                                              |                               |    |           |
| * Esamina ed esegue scelte.                                        |                               |    |           |
| * Legge piu' volte tutti i record del subfile con un loop di chain |                               |    |           |
| * per trattare le scelte.                                          |                               |    |           |
| C                                                                  | SCS1 begsr                    |    |           |
| * Esegue.                                                          |                               |    |           |
| C                                                                  | do                            |    |           |
| *----                                                              |                               |    |           |
| * ELIMINA REVERSE IMAGE E POSITION CURSOR DALLE RIGHE              |                               |    |           |
| * Spegne.                                                          |                               |    |           |
| C                                                                  | exsr OFF                      |    |           |
| * Cicla sulle righe di subfile.                                    |                               |    |           |
| C                                                                  | S1RRNBEG do S1RRNEND XX       |    |           |
| * Legge una riga.                                                  |                               |    |           |
| C                                                                  | XX chain S1                   |    |           |
| * Elimina dalla riga reverse image e position cursor.              |                               |    |           |
| C                                                                  | update S1                     |    |           |
| * Cicla sulle righe di subfile.                                    |                               |    |           |
| C                                                                  | enddo                         |    |           |
| *----                                                              |                               |    |           |
| * PREPARA LA SEGNALAZIONE DI SCELTE NON PREVISTE SULLE RIGHE       |                               |    |           |
| * Assume non ancora trovata scelta qualsiasi.                      |                               |    |           |

#### 27.2.3.10. Spegne 98=memoria di scelta X e 99=memoria di scelta qualsiasi.

|                                 |                         |      |  |
|---------------------------------|-------------------------|------|--|
| C                               | setoff                  | 9899 |  |
| * Cicla sulle righe di subfile. |                         |      |  |
| C                               | S1RRNBEG do S1RRNEND XX |      |  |
| * Legge una riga.               |                         |      |  |
| C                               | XX chain S1             |      |  |

#### 27.2.3.11. Segnala scelta non unica.

|                                            |                                             |      |  |
|--------------------------------------------|---------------------------------------------|------|--|
| * Se la riga porta una scelta X non unica, |                                             |      |  |
| * prenota PC, RI ed errore.                |                                             |      |  |
| C                                          | if (S1SCE='X' and (*in98 or *in99))         |      |  |
| C                                          | or (S1SCE<>'X' and S1SCE<>*blank and *in98) |      |  |
| C                                          | seton                                       | 5859 |  |
| C                                          | seton                                       | 6065 |  |
| C                                          | endif                                       |      |  |

#### 27.2.3.12. Annota 98=presente una scelta X.

|                                                                   |                                                |      |  |
|-------------------------------------------------------------------|------------------------------------------------|------|--|
| * Se la riga porta una scelta X, lo annota.                       |                                                |      |  |
| C                                                                 | if S1SCE='X'                                   |      |  |
| C                                                                 | seton                                          | 98   |  |
| C                                                                 | endif                                          |      |  |
| * Se la riga porta una scelta qualsiasi, lo annota.               |                                                |      |  |
| C                                                                 | if S1SCE<>*blank                               |      |  |
| C                                                                 | seton                                          | 99   |  |
| C                                                                 | endif                                          |      |  |
| * Se la riga porta una scelta non prevista,                       |                                                |      |  |
| * prenota PC, RI ed errore.                                       |                                                |      |  |
| C                                                                 | if S1SCE<>*blank and S1SCE<>'X' and S1SCE<>'Y' |      |  |
| C                                                                 | seton                                          | 5859 |  |
| C                                                                 | seton                                          | 6061 |  |
| C                                                                 | endif                                          |      |  |
| * Se corre l'ultima riga, non ci sono scelte e non sta paginando, |                                                |      |  |
| * prenota PC ed errore.                                           |                                                |      |  |
| C                                                                 | if XX=S1RRNEND and not *in99                   |      |  |
| C                                                                 | and not (*in25 or *in26)                       |      |  |
| C                                                                 | seton                                          | 58   |  |
| C                                                                 | seton                                          | 6064 |  |
| C                                                                 | endif                                          |      |  |

```

* Se la riga porta una scelta mentre e' richiesta paginazione,
* prenota PC, RI ed errore.
C if S1SCE<>*blank and (*in25 or *in26)
C seton
C seton
C endif
* Se corre richiesta di PC o RI,
* ricalca la riga in errore per PC e RI e li spegne.
C if *in58 or *in59
C update S1
C setoff
C endif
* Cicla sulle righe di subfile.
C enddo
*----
* Se errore, abbandona.
C 60 leave
*----
* ESEGUE LE SCELTE PREVISTE SULLE RIGHE
* Cicla sulle righe di subfile.
C S1RRNBEG do S1RRNEND XX
* Legge una riga.
C XX chain S1

```

#### 27.2.3.13. Restituisce la il codice scelto.

```

* Se corre la scelta X, restituisce il codice e ritorna.
C if S1SCE='X'
C eval PPCD=S1CD
C return
C endif
* Se corre una scelta valida.
C if S1SCE<>*blank
* Esegue la scelta.

```

#### 27.2.3.14. Usa nome generico del campo codice sulla riga di subfile.

```

 dsply (S1SCE+' '+S1CD) ' ' x;
* Trasforma scelta in "eseguita",
C eval S1SCE='>'
* Prenota PC.
C seton
C seton
* Finge errore solo per non ricaricare il sfl e segnalare.
C seton
* Ricalca il record per PC.
C update S1
* Spegne PC e RI.
C setoff
* Se corre una scelta valida.
C endif
* Cicla sulle righe di subfile.
C enddo
*----
* Esegue.
C enddo
C endsr
*-----
* Spegne.
C OFF begsr
* Spegne position cursor e reverse image sulla riga.
C setoff
* Spegne errori e segnalazioni sul controllo.
C setoff

```

#### 27.2.3.15. Spegne anche 65=doppia scelta,

```

C setoff
C endsr
*-----
* Trascrive il record di database nella riga di subfile.
C DBS1 begsr
* Scelta in bianco.
C eval S1SCE =*blank
* Dati.

```

#### 27.2.3.16. Usa nomi generici sulla riga di subfile.

```

C eval S1CD=DBRCD
C eval S1DE=DBRDE
C endsr
*-----

```

#### 27.2.3.17. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNT1WIN.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNT1WIN.txt)

#### 27.2.4. Funzionamento del programma CNT1WIN.

##### 27.2.4.1. Usa la logica di subfile quadrato della versione CNA1Z3E.

- 27.2.4.2. La finestra del programma si sovrappone all'immagine presente a video al momento della chiamata.
- 27.2.4.3. Permette una sola scelta X.
- 27.2.4.4. La scelta X restituisce il codice presente sulla riga di subfile scelta.

## 28. Gestione Anagrafica-Tabelle chiama Finestra di scelta codici

28.1. Gestione anagrafica completa (visualizzazione, modifica, cancellazione e aggiunta) con doppio video guida, decodifica anche a richiesta, controllo errori e finestre con scelta codici (oggetti CNA1G\*).

28.1.1. Sorgente commentato del video CNA1GW Dspfl. Ottenuto modificando la versione CNA1FW.

28.1.1.1. Le aggiunte e le modifiche riguardano l'introduzione delle finestre per la scelta dei codici.

28.1.1.2. Vista la numerosità crescente delle costanti esplicative dei comandi, non segnalo le righe delle costanti variate solo per accorciarle o riposizionarle.

```
* Claudio Neroni 03-04-2021 Creato per Corso.
* CNA1GW
* Anagrafica persone dsp-chg-dlt-add-sfl-rfr
A DSPSIZ(24 80 *DS3)
A CHGINPDFT
A INDARA
A INDTXT(44 'Protezione campi')
A INDTXT(45 'Aggiunta record')
A ERRSFL
A R W01
A TEXT('Video Guida per codice')
A BLINK
A CA03(03 'Fine')
A CF07(07 'Visualizzazione/Modifica')
A CF09(09 'Aggiunta')
A CF11(11 'PerDescrizione')
A PAGEUP(25 'Precedente')
A PAGEDOWN(26 'Successivo')
A 1 32'Anagrafica persona'
A DSPATR(UL)
A DSPATR(HI)
A WWMOD 3 O 1 76TEXT('Modalita'')
A 2 38'Guida'
A DSPATR(HI)
A 4 10'Codice persona:'
A W0CDPE R B 4 27REFFLD(A1CDPE CNA1)
A DSPATR(CS)
A DSPATR(HI)
A 23 2'F3=End'
A 23 9'F7=Chg'
A N44 23 9'F7=Dsp'
A +1'F9=Add'
A +1'F11=xDes'
A +1'Dw=Nxt'
A +1'Up=Prv'
A R W02
A TEXT('Video Guida per descrizione')
A BLINK
A CA03(03 'Fine')
A CF07(07 'Visualizzazione/Modifica')
A CF09(09 'Aggiunta')
A CF11(11 'PerCodice')
A PAGEUP(25 'Precedente')
A PAGEDOWN(26 'Successivo')
A 1 32'Anagrafica persona'
A DSPATR(UL)
A DSPATR(HI)
A WWMOD R O 1 76REFFLD(WWMOD *SRC)
A 2 38'Guida'
A DSPATR(HI)
A 3 5'Descrizione persona:'
A W0DEPE R B 3 27REFFLD(A1DEPE CNA1)
A DSPATR(CS)
A DSPATR(HI)
A 4 10'Codice persona:'
A W0CDPE R B 4 27REFFLD(A1CDPE CNA1)
A DSPATR(CS)
A DSPATR(HI)
A 23 2'F3=End'
A 23 9'F7=Chg'
A N44 23 9'F7=Dsp'
A +1'F9=Add'
A +1'F11=xCod'
A +1'Dw=Nxt'
A +1'Up=Prv'
A R W1
A TEXT('Video Dati')
A BLINK
```

28.1.1.3. Aggiunge il comando F2 per elencare i codici da scegliere.

|   |     |                   |
|---|-----|-------------------|
| A | N44 | CF02(02 'Elenca') |
| A |     | CA03(03 'Fine')   |

|                                                                                                         |                            |                                |             |                                                           |
|---------------------------------------------------------------------------------------------------------|----------------------------|--------------------------------|-------------|-----------------------------------------------------------|
| A                                                                                                       | CF04(04 'Decodifica')      |                                |             |                                                           |
| 28.1.1.4. Aggiunge il comando F5 per rinfrescare il record rileggendolo per chiave.                     |                            |                                |             |                                                           |
| A                                                                                                       | CA05(05 'Aggiorna chiave') |                                |             |                                                           |
| 28.1.1.5. Aggiunge il comando F17 per rinfrescare il record rileggendolo per numero relativo di record. |                            |                                |             |                                                           |
| A                                                                                                       | CA17(17 'Aggiorna rrn')    |                                |             |                                                           |
| A                                                                                                       | N45                        | CF07(07 'Visualizza/Modifica') |             |                                                           |
| A                                                                                                       |                            | CA12(12 'Guida')               |             |                                                           |
| A                                                                                                       |                            | CA23(23 'Cancella')            |             |                                                           |
| A                                                                                                       | N45                        | PAGEUP(25 'Precedente')        |             |                                                           |
| A                                                                                                       | N45                        | PAGEDOWN(26 'Successivo')      |             |                                                           |
| 28.1.1.6. Il record video restituisce nome record e nome campo su cui giace il cursore.                 |                            |                                |             |                                                           |
| A                                                                                                       | RTNCSRLOC(&RCD &FLD)       |                                |             |                                                           |
| 28.1.1.7. Se valorizzati, i campi seguenti posizionano il cursore..                                     |                            |                                |             |                                                           |
| A                                                                                                       | CSRLOC(LIN POS)            |                                |             |                                                           |
| 28.1.1.8. Definisce i campi che, se valorizzati, impongono la posizione al cursore.                     |                            |                                |             |                                                           |
| A                                                                                                       | LIN                        | 3S                             | 0H          |                                                           |
| A                                                                                                       | POS                        | 3S                             | 0H          |                                                           |
| 28.1.1.9. Definisce i campi che restituiscono informazioni sul cursore.                                 |                            |                                |             |                                                           |
| A                                                                                                       | RCD                        | 10A                            | H           |                                                           |
| A                                                                                                       | FLD                        | 10A                            | H           |                                                           |
| A                                                                                                       |                            |                                |             | 1 32'Anagrafica persona'                                  |
| A                                                                                                       |                            |                                |             | DSPATR(UL)                                                |
| A                                                                                                       |                            |                                |             | DSPATR(HI)                                                |
| A                                                                                                       | WWMOD                      | R                              | O           | 1 76REFFLD(WWMOD *SRC)                                    |
| A                                                                                                       | 41N45                      |                                |             | 2 33'Dati per codice'                                     |
| A                                                                                                       |                            |                                |             | DSPATR(HI)                                                |
| A                                                                                                       | 42N45                      |                                |             | 2 31'Dati per descrizione'                                |
| A                                                                                                       |                            |                                |             | DSPATR(HI)                                                |
| A                                                                                                       |                            |                                |             | 4 10'Codice persona:                                      |
| A                                                                                                       | W1CDPE                     | R                              | B           | 4 27REFFLD(A1CDPE CNA1)                                   |
| A                                                                                                       | 60 61                      |                                |             | ERRMSG('Record occupato da altri')                        |
| A                                                                                                       | 60 62                      |                                |             | ERRMSG('Record cancellato da altri')                      |
| A                                                                                                       | 60 63                      |                                |             | ERRMSG('Record modificato da altri')                      |
| A                                                                                                       | 60 64                      |                                |             | ERRMSG('Chiave gia'' presente nel + file')                |
| A                                                                                                       |                            |                                |             |                                                           |
| A                                                                                                       | 60 65                      |                                |             | ERRMSG('File non disponibile + per modifica e aggiunta')  |
| A                                                                                                       | 60 66                      |                                |             | ERRMSG('Record non cancellabile')                         |
| 28.1.1.10. Aggiunge un paio di errori riscontrabili durante il rinfrescamento.                          |                            |                                |             |                                                           |
| A                                                                                                       | 60 67                      |                                |             | ERRMSG('Chiave scomparsa')                                |
| A                                                                                                       | 60 68                      |                                |             | ERRMSG('Numero relativo di record + scomparso')           |
| A                                                                                                       |                            |                                |             |                                                           |
| A                                                                                                       | 60 71                      |                                |             | ERRMSG('Dato obbligatorio')                               |
| A                                                                                                       | 44                         |                                |             | DSPATR(PR)                                                |
| A                                                                                                       | N44                        |                                |             | DSPATR(CS)                                                |
| A                                                                                                       |                            |                                |             | DSPATR(HI)                                                |
| A                                                                                                       |                            |                                |             | 5 5'Descrizione persona:'                                 |
| A                                                                                                       | W1DEPE                     | R                              | B           | 5 27REFFLD(A1DEPE CNA1)                                   |
| A                                                                                                       | 60 72                      |                                |             | ERRMSG('Dato obbligatorio')                               |
| A                                                                                                       | 60 73                      |                                |             | ERRMSG('Descrizione deve essere piu- ' lunga del codice') |
| A                                                                                                       | 44                         |                                |             | DSPATR(PR)                                                |
| A                                                                                                       | N44                        |                                |             | DSPATR(CS)                                                |
| A                                                                                                       |                            |                                |             | DSPATR(HI)                                                |
| A                                                                                                       |                            |                                |             | CHECK(LC)                                                 |
| 28.1.1.11. Per i campi che sopportano elenco, sostituisce ":" con "?".                                  |                            |                                |             |                                                           |
| A                                                                                                       |                            |                                |             | 6 7'Provincia nascita?'                                   |
| A                                                                                                       | W1PRNA                     | R                              | B           | 6 27REFFLD(A1PRNA CNA1)                                   |
| A                                                                                                       | 60 74                      |                                |             | ERRMSG('Dato obbligatorio')                               |
| A                                                                                                       | 60 75                      |                                |             | ERRMSG('Manca in tabella')                                |
| A                                                                                                       | 44                         |                                |             | DSPATR(PR)                                                |
| A                                                                                                       | N44                        |                                |             | DSPATR(CS)                                                |
| A                                                                                                       |                            |                                |             | DSPATR(HI)                                                |
| A                                                                                                       | W1PRNAD                    | R                              | O           | 6 30REFFLD(T1DEPR CNT1)                                   |
| 28.1.1.1. Per i campi che sopportano elenco, sostituisce ":" con "?".                                   |                            |                                |             |                                                           |
| A                                                                                                       |                            |                                |             | 7 5'Provincia residenza?'                                 |
| A                                                                                                       | W1PRRE                     | R                              | B           | 7 27REFFLD(A1PRRE CNA1)                                   |
| A                                                                                                       | 60 76                      |                                |             | ERRMSG('Manca in tabella')                                |
| A                                                                                                       | 44                         |                                |             | DSPATR(PR)                                                |
| A                                                                                                       | N44                        |                                |             | DSPATR(CS)                                                |
| A                                                                                                       |                            |                                |             | DSPATR(HI)                                                |
| A                                                                                                       | W1PRRED                    | R                              | O           | 7 30REFFLD(T1DEPR CNT1)                                   |
| 28.1.1.2. Se i campi sono agibili, dichiara elenchi permessi.                                           |                            |                                |             |                                                           |
| A                                                                                                       | N44                        | 23                             | 2 'F2=Sf1'  |                                                           |
| A                                                                                                       |                            |                                | +1 'F3=End' |                                                           |



|   |             |
|---|-------------|
| A | +1 'F4=Dec' |
|---|-------------|

28.1.1.3. Dichiaro permessi i comandi di rinfrescamento.

|          |                |
|----------|----------------|
| A        | +1 'F5=RfK'    |
| A        | +1 'F17=RfN'   |
| A 44N45  | 23 38 'F7=Chg' |
| A N44N45 | 23 38 'F7=Dsp' |
| A        | +1 'F12=Gui'   |
| A 44N45  | +1 'F23=Dlt'   |
| A N45    | +1 'Dw=Nxt'    |
| A N45    | +1 'Up=Prv'    |

28.1.1.4. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1GW.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1GW.txt)

28.1.2. Sorgente commentato del programma CNA1GD Rpgle. Ottenuto modificando la versione CNA1FD.

|                                                     |                            |
|-----------------------------------------------------|----------------------------|
| *FREE * Claudio Neroni 03-04-2021 Creato per Corso. |                            |
| * CNA1GD                                            |                            |
| * Anagrafica persone dsp-chg-dlt-add-sfl-rfr        |                            |
| *-----                                              |                            |
| * Anagrafica persone logico codice                  |                            |
| FCNA11                                              | if e k disk rename(A1:A11) |
| F                                                   | infd (FIDS1)               |
| * Anagrafica persone logico descrizione             |                            |
| FCNA12                                              | if e k disk rename(A1:A12) |
| F                                                   | infd (FIDS2)               |
| * Tabella provincie                                 |                            |
| FCNT11                                              | if e k disk                |
| * Anagrafica persone dsp-chg-dlt-add-sfl-rfr        |                            |
| FCNA1GW                                             | cf e workstn               |

28.1.2.1. Legge la posizione del cursore video.

|                                                   |                                   |
|---------------------------------------------------|-----------------------------------|
| * Display Specific Feedback Information           |                                   |
| F                                                 | infd (DSPFBK)                     |
| * Anagrafica persone fisico                       |                                   |
| FCNA1                                             | uf a e disk rename(A1:UPD) usropr |
| *-----                                            |                                   |
| * Struttura dati del file dati.                   |                                   |
| D A1D                                             | e ds extname(CNA1:A1)             |
| * Salvataggio della Struttura dati del file dati. |                                   |
| D A1DSV                                           | e ds extname(CNA1:A1) prefix(SV)  |
| *-----                                            |                                   |
| * Struttura dati del video dati.                  |                                   |

28.1.2.2. Cambia nome del file video.

|                                                    |                                    |
|----------------------------------------------------|------------------------------------|
| D W1D                                              | e ds extname(CNA1GW:W1)            |
| * Salvataggio della Struttura dati del video dati. |                                    |
| D W1DSV                                            | e ds extname(CNA1GW:W1) prefix(SV) |
| *-----                                             |                                    |
| * File information data structure                  |                                    |
| D FIDS1                                            | ds                                 |
| * Relative record number in data member            |                                    |
| D FIDS1RR                                          | 397 400b 0                         |
| * File information data structure                  |                                    |
| D FIDS2                                            | ds                                 |
| * Relative record number in data member            |                                    |
| D FIDS2RR                                          | 397 400b 0                         |
| *-----                                             |                                    |

28.1.2.3. Titoli da mettere nelle cornici degli elenchi.

|             |   |                          |
|-------------|---|--------------------------|
| * Costanti  |   |                          |
| D NASCITA   | c | 'Provincia di Nascita'   |
| D RESIDENZA | c | 'Provincia di Residenza' |
| *-----      |   |                          |

28.1.2.4. File information data structure del file video per la lettura del cursore.

|                                           |        |       |
|-------------------------------------------|--------|-------|
| * Display Specific Feedback Information   |        |       |
| /COPY CORSO,DSPFBKCL                      |        |       |
| *-----                                    |        |       |
| * Predispone chiusura.                    |        |       |
| C                                         | seton  | 1r    |
| * Assume di usare video guida per codice. |        |       |
| C                                         | seton  | 41    |
| C                                         | setoff | 42    |
| * Predispone protezione campi.            |        |       |
| C                                         | seton  | 44    |
| * Gestisce video guida.                   |        |       |
| C                                         | exsr   | SRW0  |
| *-----                                    |        |       |
| * Gestisce video guida.                   |        |       |
| C                                         | SRW0   | begsr |
| * Definisce chiave per codice.            |        |       |
| C                                         | K1     | klist |
| C                                         | kfld   | WOCDE |
| * Definisce chiave per descrizione.       |        |       |

```

C K2 klist
C kfld WODEPE
C kfld WOCDEPE
* Pulisce video guida.
C clear W01
C clear W02
* Cicla intorno al video guida.
C do *hival
* Annota non aggiunta.
C setoff 45
* Prepara modo.
C 44 eval WWMOD='DSP'
C N44 eval WWMOD='CHG'
* Emette video guida.
C 41 exfmt W01
C 42 exfmt W02
* Se richiesta fine, abbandona.
C 03 leave
* Se richiesto switch tra visualizzazione e modifica,
* la esegue e ricicla.
C if *in07
C *IN44 comp *off 44
C iter
C endif
* Se richiesta aggiunta.
C if *in09
* Esegue aggiunta.
C exsr SRW1A
* Se richiesta fine, abbandona.
C 03 leave
* Ricicla.
C iter
* Se richiesta aggiunta.
C endif
* Se richiesto switch tra guida per codice e guida per descrizione,
* commuta la scelta del video guida e ricicla.
C if *in11
C if *in41
C setoff 41
C seton 42
C else
C seton 41
C setoff 42
C endif
C iter
C endif
* Se richiesto PagUp.
C if *in25
* Si posiziona prima della chiave di ricerca.
C 41K1 setll A11
C 42K2 setll A12
* Legge all'indietro.
C 41 readp A11
C 42 readp A12
* Se non trova record, ritenta a partire dal fondo.
C if %eof
C 41*hival setgt A11
C 42*hival setgt A12
C 41 readp A11
C 42 readp A12
C endif
* Se richiesto PagUp o Enter,
C else
* Si posiziona prima della chiave di ricerca.
C 41K1 setll A11
C 42K2 setll A12
* Legge in avanti.
C 41 read A11
C 42 read A12
* Se richiesto...
C endif
* Se non trova record, ricicla.
C if %eof
C iter
C endif
* Gestisce video dati.
C exsr SRW1
* Se richiesta fine, abbandona.
C 03 leave
* Cicla intorno al video guida.
C enddo
C endsr
*-----
* Gestisce video dati.

```

```

* Va chiamata con un record gia' in canna.
C SRW1 begsr
* Cicla intorno al video dati.
C do *hival
* Se non corre errore.
C if not *in60
* Trascrive dati dal record di database al video.
C exsr DBW1
* Decodifica video dati.
C exsr DEW1
* Se non corre errore.
C endif
* Se corre errore, scrive base per errore.
C if *in60
C setoff
C write W1
C seton
C endif
C 60
* Prepara modo.
C 44 eval WWMOD='DSP'
C N44 eval WWMOD='CHG'
* Emette video dati.
C exfmt W1

```

#### 28.1.2.5. Azzeri i campi che, se valorizzati, posizionano il cursore.

```

* Elimina posizionamento cursore.
C exsr CURZ
* Spegne tutti gli errori.
C exsr OFF
*----
* Se richiesta fine, abbandona.
C 03 leave
*----
* Se richiesta Guida, abbandona.
C 12 leave
*----
* Se richiesto switch tra visualizzazione e modifica,
* la esegue e ricicla.
C if *in07
C *IN44 comp *off
C iter
C endif
C 44
*----

```

#### 28.1.2.6. Gestisce il rinfrescamento del record rileggendolo per chiave.

```

* Se richiesto aggiornamento per chiave (key refresh).
C if *in05
* Rilegge l'ultimo record messo a video.
C 41K1 chain A11
C 42K2 chain A12
* Se il record non esiste, accende errore.
C if not %found
C seton
C endif
C 6067
* Ricicla.
C iter
* Se richiesto aggiornamento per chiave (key refresh).
C endif
*----

```

#### 28.1.2.1. Gestisce il rinfrescamento del record rileggendolo per numero relativo di record..

```

* Se richiesto aggiornamento per nr relativo di record (rrn refresh).
C if *in17
* Apre database.
C exsr OPDB
* Se errore in apertura, ricicla.
C 60 iter
* Rilegge per sola lettura l'ultimo record messo a video.
C 41FIDS1RR chain(n) UPD
C 42FIDS2RR chain(n) UPD
* Se il record non esiste, accende errore.
C if not %found
C seton
C endif
C 6068
* Ricicla.
C iter
* Se richiesto aggiornamento per nr relativo di record (rrn refresh).
C endif
*----

```

#### 28.1.2.2. Gestisce l'elenco con scelta dei valori permessi nel campo su cui si trova il cursore.

```

* Se richiesto elenco.
C if *in02
* Trascrive cursore per la riemissione.

```

```

C exsr CURT
* Chiama elenco su provincia nascita.
C if FLD='W1PRNA'
C call 'CNT1WIN'
C parm W1PRNA C DPR
C parm NASCITA INTESTA 50
C *like define W1PRNA C DPR
C if CDPR<>*blank
C eval W1PRNA=CDPR
C endif
C endif
* Chiama elenco su provincia residenza.
C if FLD='W1PRRE'
C call 'CNT1WIN'
C parm W1PRRE C DPR
C parm RESIDENZA INTESTA 50
C *like define W1PRRE C DPR
C if CDPR<>*blank
C eval W1PRRE=CDPR
C endif
C endif
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C**** exsr CTW1
* Finge errore per non ricaricare da database.
C seton
* Ricicla.
C iter
* Se richiesto elenco.
C endif
*---
* Se corre decodifica.
C if *in04
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C exsr CTW1
* Finge errore per non ricaricare da database.
C seton
* Ricicla.
C iter
* Se corre decodifica.
C endif
*---
* Se corre cancellazione.
C if *in23
* Esegue cancellazione.
C exsr DLDB
* Se errore, ricicla.
C 60 iter
* Se corre cancellazione.
C endif
*---
* Se corre modifica.
C if not *in44
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C exsr CTW1
* Se errore, ricicla.
C 60 iter
* Se il video e' stato modificato.
C if W1D<>W1DSV
* Esegue update.
C exsr UPDB
* Se errore, ricicla.
C 60 iter
* Se il video e' stato modificato.
C endif
* Se corre modifica.
C endif
*---
* Se richiesto PagUp.
C if *in25
* Si posiziona prima della chiave ultima vista.
C 41K1 setll A11
C 42K2 setll A12
* Legge all'indietro.
C 41 readp A11
C 42 readp A12
*---
* Se richiesto PagDw o Enter,
C else

```

```

* Si posiziona oltre la chiave ultima vista.
C 41K1 setgt A11
C 42K2 setgt A12
* Legge in avanti.
C 41 read A11
C 42 read A12
* Se richiesto...
C endif
* Se non trova record, pulisce chiave accantonata e abbandona.
C if %eof
C clear
C clear W0CDPE
C leave
C endif
*----
* Cicla intorno al video dati.
C enddo
C endsr
*-----
* Trascrive dati dal record di database al video.
C DBW1 begsr
* Accantona la chiave nel video guida.
C exsr DBW1KK
* Trascrive i campi dal record di database al video dati.
C clear W1
C eval W1CDPE=A1CDPE
C eval W1DEPE=A1DEPE
C eval W1PRNA=A1PRNA
C eval W1PRRE=A1PRRE
* Salva i dati del record letto.
C eval A1DSV=A1D
* Salva i dati del video.
C eval W1DSV=W1D
C endsr
*-----
* Decodifica video dati.
C DEW1 begsr
* Decodifica provincia di nascita.
C clear W1PRNAD
C if W1PRNA<>*blank
C eval W1PRNAD=*all'?'
C W1PRNA chain T1
C if %found
C eval W1PRNAD=T1DEPR
C endif
C endif
* Decodifica provincia di residenza.
C clear W1PRRED
C if W1PRRE<>*blank
C eval W1PRRED=*all'?'
C W1PRRE chain T1
C if %found
C eval W1PRRED=T1DEPR
C endif
C endif
C endsr
*-----
* Spegne tutti gli errori.
C OFF begsr
C move *off OFF31 31
C movea OFF31 *in(60)
C endsr
*-----
* Controlla video dati.
C CTW1 begsr
* Spegne tutti gli errori.
C exsr OFF
* Esegue.
C do
* Codice obbligatorio.
C if W1CDPE=*blank
C seton
C leave
C endif
* Descrizione obbligatoria.
C if W1DEPE=*blank
C seton
C leave
C endif
* Descrizione piu' corta del codice.
C if %len(%trim(W1DEPE))<=%len(%trim(W1CDPE))
C seton
C leave
C endif

```

```

* Provincia di nascita obbligatoria.
C if W1PRNA=blank
C seton
C leave
C endif
* Provincia di nascita manca in tabella.
C if W1PRNAD=*all'?'
C seton
C leave
C endif
* Provincia di residenza manca in tabella.
C if W1PRRED=*all'?'
C seton
C leave
C endif
* Esegue.
C enddo
C endsr

*-----
* Trascrive dati dal video al record di database
C W1DB begsr
C eval A1CDPE=W1CDPE
C eval A1DEPE=W1DEPE
C eval A1PRNA=W1PRNA
C eval A1PRRE=W1PRRE
C endsr
*-----
* Ricalca database.
C UPDB begsr
* Esegue.
C do
* Apre database.
C exsr OPDB
* Se errore in apertura, abbandona.
C 60 leave
* Recupera il record fisico per ricalcarlo.
C 41FIDS1RR chain(e) UPD
C 42FIDS2RR chain(e) UPD
* Se il record e' occupato, accende errore e abbandona.
C if %error
C seton
C leave
C endif
* Se il record non esiste, accende errore e abbandona.
C if not %found
C seton
C leave
C endif
* Se il dati del record sono cambiati, accende errore e abbandona.
C if A1D<>A1DSV
C seton
C unlock CNA1
C leave
C endif
* Trascrive dati dal video al record di database
C exsr W1DB
* Ricalca il record.
C update(e) UPD
* Se la chiave del record esiste gia', accende errore e abbandona.
C if %error
C seton
C leave
C endif
* Esegue.
C enddo
C endsr
*-----
* Gestisce video dati in aggiunta record.
C SRW1A begsr
* Assume campi agibili.
C setoff
* Assume aggiunta.
C seton
* Cicla intorno a video dati.
C do *hival
* Se non corre errore.
C if not *in60
* Pulisce video dati.
C clear
* Decodifica video dati.
C exsr DEW1
* Se non corre errore.
C endif
* Se corre errore, scrive base per errore.

```

```

C if *in60
C setoff
C write W1
C seton
C endif
C * Prepara modo.
C eval WWMOD='ADD'
C * Emette video dati.
C exfmt W1

```

28.1.2.3. Azzera i campi che, se valorizzati, posizionano il cursore.

```

* Elimina posizionamento cursore.
C exsr CURZ
* Spegne tutti gli errori.
C exsr OFF
*----
* Se richiesta fine, abbandona.
C 03 leave
*----
* Se richiesta Guida, abbandona.
C 12 leave
*----

```

28.1.2.4. Se richiesto rinfrescamento, ricicla buttando il contenuto del video di aggiunta.

```

* Se richiesto aggiornamento (refresh), ricicla.
C 05 iter
*----

```

28.1.2.5. Gestisce l'elenco con scelta dei valori permessi nel campo su cui si trova il cursore.

```

* Se corre elenco.
C if *in02
* Trascrive cursore per la riemissione.
C exsr CURT
* Chiama elenco su provincia nascita.
C if FLD='W1PRNA'
C call 'CNT1WIN'
C parm W1PRNA CDPR
C parm NASCITA INTESTA 50
C *like define W1PRNA CDPR
C if CDPR<>*blank
C eval W1PRNA=CDPR
C endif
C endif
* Chiama elenco su provincia residenza.
C if FLD='W1PRRE'
C call 'CNT1WIN'
C parm W1PRRE CDPR
C parm RESIDENZA INTESTA 50
C *like define W1PRRE CDPR
C if CDPR<>*blank
C eval W1PRRE=CDPR
C endif
C endif
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C*** exsr CTW1
* Finge errore per non ricaricare da database.
C seton
C * Ricicla.
C iter
* Se corre elenco.
C endif
*----
* Se corre decodifica.
C if *in04
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C exsr CTW1
* Finge errore per non pulire video.
C seton
C * Ricicla.
C iter
* Se corre decodifica.
C endif
*----
* Decodifica video dati.
C exsr DEW1
* Controlla video dati.
C exsr CTW1
* Se errore, ricicla.
C 60 iter
* Scrive database.

```

```

C exsr WRDB
* Se errore, ricicla.
C 60 iter
* Cicla intorno a video dati.
C enddo
C endsr
*-----
* Scrive database.
C WRDB begsr
* Esegue.
C do
* Apre database.
C exsr OPDB
* Se errore in apertura, abbandona.
C 60 leave
* Trascrive dati dal video al record di database
C exsr W1DB
* Accantona la chiave database nei campi dei video guida.
C exsr DBW1KK
* Aggiunge il record.
C write(e) UPD
* Se la chiave del record esiste gia', accende errore e abbandona.
C if %error
C seton
C leave
C endif
* Esegue.
C enddo
C endsr
*-----
* Apre database.
C OPDB begsr
* Se il file fisico per update non e' aperto.
C if not %open(CNA1)
* Tenta l'apertura del file.
C open(e) CNA1
* Se errore in apertura, accende errore.
C if %error
C seton
C endif
C endif
* Se il file fisico per update non e' aperto.
C endif
C endsr
*-----
* Accantona la chiave database nei campi dei video guida.
C DBW1KK begsr
C eval W0CDPE=A1CDPE
C eval W0DEPE=A1DEPE
C endsr
*-----
* Cancella database.
C DLDB begsr
* Esegue.
C do
* Apre database.
C exsr OPDB
* Se errore in apertura, abbandona.
C 60 leave
* Recupera il record fisico per ricalcarlo.
C 41FIDS1RR chain(e) UPD
C 42FIDS2RR chain(e) UPD
* Se il record e' occupato, accende errore e abbandona.
C if %error
C seton
C leave
C endif
* Se il record non esiste, accende errore e abbandona.
C if not %found
C seton
C leave
C endif
* Se il dati del record sono cambiati, accende errore e abbandona.
C if A1D<>A1DSV
C seton
C unlock CNA1
C leave
C endif
* Cancella il record.
C delete(e) UPD
* Se la cancellazione fallisce, accende errore e abbandona.
C if %error
C seton
C leave
C endif

```

6064

6065

6061

6062

6063

6066



```

* Esegue.
C enddo
C endsr
*-----

```

28.1.2.6. Azzera i campi che, se valorizzati, posizionano il cursore.

```

* Elimina posizionamento cursore.
C CURZ begsr
C clear LIN
C clear POS
C endsr
*-----

```

28.1.2.7. Trascrive la posizione del cursore dall'input all'output della stessa videata.

```

* Trascrive cursore per la riemissione.
C CURT begsr
C eval LIN=ROW
C eval POS=COL
C endsr
*-----

```

28.1.2.1. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1GD.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNA1GD.txt)

28.1.3. Funzionamento del programma CNA1GD.

- 28.1.3.1. Aggiunto elenco sui campi che hanno una tabella di decodifica.
- 28.1.3.2. Aggiunto rinfrescamento per chiave o per numero relativo di record.
- 28.1.3.3. Identico per il resto alla versione precedente.

## 29. Printer file>NC

- 29.1. Scrivere ed usare un printer file internamente descritto in RPG.
- 29.2. Scrivere ed usare un printer file esternamente descritto con DDS.
  - 29.2.1. Scrivere DDS con SEU
  - 29.2.2. Manipolare DDS con RLU (Report Layout Utility).

## 30. Salvataggi e ripristini

- 30.1. Il sistema IBMi ha comandi di salvataggio fuori linea specializzati, elencabili tramite il comando **GO CMDSAV**, che possono scaricare sia su nastro sia su un particolare tipo di file, detto di salvataggio. Ne esaminiamo due che agiscono sul sistema di indirizzamento oggetti tradizionale, detto "QSYS":

```
SAVLIB (Save Library)
SAVOBJ (Save Object)
```

- 30.2. Altri comandi simmetrici, elencati a loro volta da **GO CMDRST**, permettono di usare i salvataggi precedentemente generati e accantonati per ricaricare gli oggetti salvati ricreando o ricoprendo gli originali o generando gli stessi oggetti in un'altra libreria precedentemente vuota o comunque priva degli oggetti da ricaricare. I seguenti "Restore" corrispondono ai precedenti "Save":

```
RSTLIB (Restore Library)
RSTPBJ (Restore Object)
```

- 30.3. I quattro comandi elencati permettono non solo le principali attività sistemistiche di salvataggio (se declinati con qualche approfondimento ulteriore), ma costituiscono anche una dotazione essenziale del programmatore e lo rendono autonomo nell'esercizio delle cautele di programmazione verso il materiale sorgente, sua principale responsabilità.

- 30.4. In particolare, vista la scaricabilità dei file di salvataggio sul PC, è possibile salvare librerie sorgenti ad una certa data e conservarsele per garanzia futura contro il rischio di cancellazioni incaute, senza occupare spazio sui dischi di sistema, sempre troppo pieni e in balia di sistemisti inclini a cancellazioni liberatorie per necessità "superiori".

- 30.5. Salvataggi e ripristini possono essere limitati tramite assegnazione di autorità agli utenti opportuni.

- 30.6. Salvare una libreria intera.

30.6.1. Per non interferire con gli oggetti da salvare, meglio che il file di salvataggio (\*SAVF) risieda in un'altra libreria rispetto agli oggetti salvati, meglio ancora se dedicata.

```
> CRTLIB LIB(CORSOSAV) TYPE(*TEST) TEXT('Corso Salvataggi')
È stata creata la libreria CORSOSAV.
```

30.6.2. Si crea il file di salvataggio vuoto.

```
> CRTSAVF FILE(CORSOSAV/CNSAVLIB) TEXT('Salvataggio Libreria')
File CNSAVLIB creato nella libreria CORSOSAV.
```

30.6.3. Si salva la libreria nel file di salvataggio.

```
> SAVLIB LIB(CORSO) DEV(*SAVF) SAVF(CORSOSAV/CNSAVLIB) TGTRLS(*CURRENT)
50 oggetti salvati da CORSO. Quelli 2 non sono stati salvati.
```

Si osservi che il traduttore ha colpito ancora e che l'errore è lì da tempo immemorabile.

30.6.4. Se si incappa in oggetti allocati, si chiudono i lavori che allocano gli oggetti e si ripete il comando. Nel corso del secondo tentativo si dovrà rispondere "G" al messaggio di interrogazione CPA4067 che richiede di ricoprire il precedente contenuto del file di salvataggio.

```
> SAVLIB LIB(CORSO) DEV(*SAVF) SAVF(CORSOSAV/CNSAVLIB) TGTRLS(*CURRENT)
```

```
Il file di salvataggio CNSAVLIB in CORSOSAV contiene già dei dati. (C G)
Immettere la risposta e premere Invio.
Risposta .
```

52 oggetti salvati dalla libreria CORSO.

- 30.7. Questo ed altri comandi di salvataggio forniscono un log di dettaglio in caso di inciampo che permette di individuare la ragione di un salvataggio impossibile o parziale causato da oggetti allocati da altri lavori.

30.7.1. Ad esempio se un altro lavoro esegue il comando seguente che alloca esclusivamente un membro in un file della libreria::

```
> ALCOBJ OBJ((CORSO/CORSO *FILE *EXCL AAA)) WAIT(1)
```

Il salvataggio seguente della libreria termina in errore se richiesto PRECHK(\*YES);

```
> SAVLIB LIB(CORSO) DEV(*SAVF) SAVF(CORSOSAV/CNSAVLIB) TGTRLS(*CURRENT) PRECHK(*YES)
Non sono stati salvati oggetti della libreria CORSO.
```

Premendo F10 si accede al log dettagliato:

```
> SAVLIB LIB(CORSO) DEV(*SAVF) SAVF(CORSOSAV/CNSAVLIB) TGTRLS(*CURRENT) PRECHK(*YES)
Non è possibile allocare l'oggetto per il file CORSO in CORSO.
L'oggetto CORSO di tipo FILE nella libreria CORSO non è stato salvato.
La richiesta 'salvataggio durante l'utilizzo' per la libreria CORSO è
terminata in modo anomalo.
```

Non sono stati salvati oggetti della libreria CORSO.

Se invece non è richiesto PRECHK(\*YES), il salvataggio è parziale ed esclude l'intero file a cui appartiene il membro allocato:

```
> SAVLIB LIB(CORSO) DEV(*SAVF) SAVF(CORSOSAV/CNSAVLIB) TGTRLS(*CURRENT)
51 oggetti salvati da CORSO. Quelli 1 non sono stati salvati.
```

Log che al massimo dettaglio recita:

```
> SAVLIB LIB(CORSO) DEV(*SAVF) SAVF(CORSOSAV/CNSAVLIB) TGTRLS(*CURRENT)
Non è possibile allocare l'oggetto per il file CORSO in CORSO.
L'oggetto CORSO di tipo FILE nella libreria CORSO non è stato salvato.
Il file di salvataggio CNSAVLIB in CORSOSAV contiene già dei dati. (C G)
? G
L'elaborazione del checkpoint 'salvataggio durante l'utilizzo' della
libreria CORSO è stata completata.
51 oggetti salvati da CORSO. Quelli 1 non sono stati salvati.
```

### 30.8. Documentazione degli esiti dei comandi di salvataggio su file di emissione.

30.8.1. Nel caso sia utile conservare informazioni dettagliate su salvataggi non presidiati, si potrà fare uso del **file di emissione** previsto dai comandi di salvataggio, avendo magari cura di andare in aggiunta invece che in sostituzione e scegliendo il **tipo di informazioni** più adatto.

```
> SAVLIB LIB(CORSO) DEV(*SAVF) SAVF(CORSOSAV/CNSAVLIB) TGTRLS(*CURRENT) PRECHK(*YES)
OUTPUT(*OUTFILE) OUTFILE(CORSOSAV/SAVLIBOUT) OUTMBR(*FIRST *REPLACE) INFTYPE(*MBR)
```

30.8.2. Si esaminino tracciato e dati scaricati tramite il comando:

```
> RUNQRY QRYFILE((CORSOSAV/SAVLIBOUT)) RCDSLT(*YES)
```

### 30.9. Visualizzazione del contenuto di un file di salvataggio.

30.9.1. L'elenco degli oggetti presenti nel file di salvataggio può essere esplorato con l'uso del comando:

```
DSPSAVF (Display Save File)
```

30.9.2. Ad esempio, per vedere il contenuto del file di salvataggio della libreria CORSO, si userà:

```
> DSPSAVF FILE(CORSOSAV/CNSAVLIB)
```

ottenendo la seguente visualizzazione:

| Display Saved Objects                                    |          |      |           |        |          |         |
|----------------------------------------------------------|----------|------|-----------|--------|----------|---------|
| Library saved . . . . . : CORSO                          |          |      |           |        |          |         |
| Type Options, press Enter.                               |          |      |           |        |          |         |
| 5=Display                                                |          |      |           |        |          |         |
| Opt                                                      | Object   | Type | Attribute | Owner  | Size (K) | Data    |
|                                                          | CORSO    | *LIB | TEST      | NERONI | 144      | YES     |
|                                                          | ACLE     | *PGM | CLLE      | NERONI | 88       | YES     |
|                                                          | CNA1AC   | *PGM | CLLE      | NERONI | 112      | YES     |
|                                                          | CNA1AD   | *PGM | RPGLE     | NERONI | 160      | YES     |
|                                                          | CNA1BC   | *PGM | CLLE      | NERONI | 112      | YES     |
|                                                          | CNA1BD   | *PGM | RPGLE     | NERONI | 172      | YES     |
|                                                          | CNA1CD   | *PGM | RPGLE     | NERONI | 192      | YES     |
|                                                          | CNA1DD   | *PGM | RPGLE     | NERONI | 240      | YES     |
|                                                          | CNA1ED   | *PGM | RPGLE     | NERONI | 304      | YES     |
|                                                          | CNA1EX   | *PGM | RPGLE     | NERONI | 184      | YES     |
|                                                          | CNA1PRT1 | *PGM | RPGLE     | NERONI | 168      | YES     |
|                                                          | CNINI    | *PGM | CLLE      | NERONI | 96       | YES     |
|                                                          | CNUPRT1  | *PGM | RPGLE     | NERONI | 140      | YES     |
|                                                          |          |      |           |        |          | More... |
| F3=Exit F11=Alternate view F12=Cancel F16=Display header |          |      |           |        |          |         |

### 30.10. Ripristinare una libreria intera.

30.10.1. Il ripristino segue regole complesse circa il ricalco degli oggetti preesistenti nella libreria ricevente, soprattutto per quanto riguarda i membri dei file fisici, permettendo di scegliere quali membri ricaricare e quali no in base alla classificazione nuovi-vecchi e presenti-assenti nelle due edizioni dei file (corrente e salvata). Purtroppo queste scelte risultano molto arbitrarie senza l'esame puntuale delle differenze. È meglio perciò ripristinare in una libreria nuova e confrontare file correnti e file salvati a vista e non ciecamente a priori.

30.10.2. Si consiglia quindi fortemente di creare od usare come destinazione una libreria vuota e di ripescare da quella gli oggetti che interessano.

30.10.3. Ad esempio, dal salvataggio del caso precedente, la libreria salvata CORSO viene ripristinata integralmente nella non preesistente CORSORST, creata dal ripristino.

```
> RSTLIB SAVLIB(CORSO) DEV(*SAVF) SAVF(CORSOSAV/CNSAVLIB) RSTLIB(CORSORST)
52 oggetti ripristinati da CORSO a CORSORST.
```

### 30.11. Ripristinare singoli oggetti da un salvataggio di libreria intera.

30.11.1. Nel caso in cui serva ripristinare un solo oggetto, conviene comunque fare uso di una libreria ricevente vuota per evitare i già detti problemi di ricalco. Tuttavia, a differenza di quanto accade nel ripristino integrale della libreria, la libreria deve esistere prima del ripristino.

```
> CRTLIB CORSORST1
```

È stata creata la libreria CORSORST1.

```
> RSTOBJ OBJ(CORSO) SAVLIB(CORSO) DEV(*SAVF) OBJTYPE(*FILE) SAVF(CORSOSAV/CNSAVLIB)
RSTLIB(CORSORST1)
```

1 oggetti ripristinati da CORSO a CORSORST1.

30.11.2. Si osservi qui di seguito che il ripristino di un singolo oggetto può attingere sia al salvataggio di una intera libreria, come visto, sia al salvataggio di uno o più oggetti.

30.12. Salvare uno o più oggetti.

30.12.1. Anche in questo caso, meglio creare un file di salvataggio.

```
> CRTSAVF FILE(CORSOSAV/CNSAVOBJ) TEXT('Salvataggio Oggetti')
File CNSAVOBJ creato nella libreria CORSOSAV.
```

30.12.2. Si salvano gli oggetti nel file di salvataggio.

```
> SAVOBJ OBJ(CORSO) LIB(CORSO) DEV(*SAVF) OBJTYPE(*FILE) SAVF(CORSOSAV/CNSAVOBJ)
TGTRLS(*CURRENT)
1 oggetti salvati dalla libreria CORSO.
```

30.13. Ripristinare singoli oggetti da salvataggio di uno o più oggetti.

30.13.1. Il ripristino dal salvataggio di oggetti, mutatis mutandis, segue la stessa logica del ripristino dal salvataggio di libreria intera.

```
> CRTLIB CORSORST2
È stata creata la libreria CORSORST2.
> RSTOBJ OBJ(CORSO) SAVLIB(CORSO) DEV(*SAVF) OBJTYPE(*FILE) SAVF(CORSOSAV/CNSAVOBJ)
RSTLIB(CORSORST2)
1 oggetti ripristinati da CORSO a CORSORST2.
```

## 31. Trasporto di file di salvataggio tra IBMi e PC (e viceversa) via FTP

### 31.1. Trasporto di un file di salvataggio da IBMi a PC via FTP.

31.1.1. Se si dispone di un PC in rete con l'IBMi sul quale risiede il file di salvataggio, quest'ultimo può essere facilmente portato in un file PC tramite FTP.

31.1.2. Usando il file di salvataggio generato nel primo esempio presentato "Salvare una libreria intera", sull'immissione comandi del DOS, digitando le seguenti **istruzioni** si ottiene il seguente log e, soprattutto il file **C:\XX\cnsavlib.savf** sul PC che è l'immagine del file di salvataggio originale.

```
C:\>ftp DEV720.RZKH.DE
Connesso a DEV720.RZKH.DE.
220-QTCP at dev720.rzkh.de.
220 Connection will close if idle more than 5 minutes.
501 OPTS unsuccessful; specified subcommand not recognized.
Utente (DEV720.RZKH.DE:(none)): neroni
331 Enter password.
Password: xxxxxxxxxx
230 NERONI logged on.
ftp> bin
200 Representation type is binary IMAGE.
ftp> get "/qsys.lib/corsosav.lib/cnsavlib.savf" C:\XX\cnsavlib.savf
200 PORT subcommand request successful.
150 Retrieving member CNSAVLIB in file CNSAVLIB in library CORSOSAV.
226 File transfer completed successfully.
ftp: 4067712 bytes received in 7.20secondi 564.65Kbyte/sec)
ftp>
```

31.1.2.1. **DEV720.RZKH.DE** è il nome o l'IP dell'IBMi.

31.1.2.2. **neroni** è il nome utente e **xxxxxxxxxx** la sua password su IBMi.

31.1.2.3. **bin** richiede che non venga eseguita alcuna manipolazione sui bit contenuti nel file, che così passano indenni dall'ambiente EBCDIC di IBMi all'ambiente ASCII del PC.

### 31.2. Il file PC generato nei passi precedenti può:

31.2.1. restare sul PC di origine per conservazione e sicurezza dati;

31.2.2. essere inviato via email o in altro modo ad un PC di destinazione;

31.2.3. essere ricaricato sullo stesso o su un altro IBMi in rete col PC di origine o di destinazione, sempre con FTP ma usando l'istruzione **put** invece che **get**.

### 31.3. Trasporto di un file di salvataggio da PC a IBMi via FTP.

31.3.1. Nell'esempio seguente ricarico su IBMi con la **put** il file PC ottenuto in precedenza con la **get**, cambiandogli nome, nella stessa libreria di origine.

```
C:\>ftp DEV720.RZKH.DE
Connesso a DEV720.RZKH.DE.
220-QTCP at dev720.rzkh.de.
220 Connection will close if idle more than 5 minutes.
501 OPTS unsuccessful; specified subcommand not recognized.
Utente (DEV720.RZKH.DE:(none)): neroni
331 Enter password.
Password: xxxxxxxxxx
230 NERONI logged on.
ftp> bin
200 Representation type is binary IMAGE.
ftp> put C:\XX\cnsavlib.savf "/qsys.lib/corsosav.lib/cnsavlibrs.savf"
200 PORT subcommand request successful.
150 Sending file to member CNSAVLIBRS in file CNSAVLIBRS in library CORSOSAV.
226 File transfer completed successfully.
ftp: 4067712 bytes sent in 5.32secondi 764.32Kbyte/sec)
ftp>
```

31.3.1.1. **bin**, come prima, richiede che non venga eseguita alcuna manipolazione sui bit contenuti nel file, che così ripassano indenni dall'ambiente ASCII del PC all'ambiente EBCDIC di IBMi.

## 32. Tool jSAVF

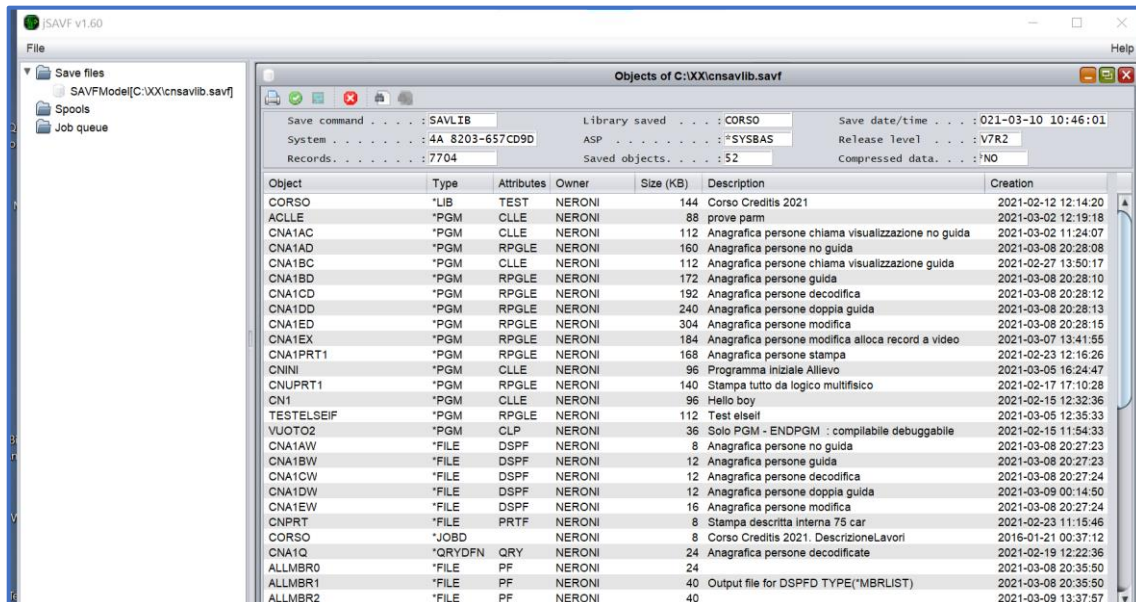
32.1. Visualizzazione del file sul PC tramite il tool jSAVF.

32.1.1. Il tool è reperibile al link:

[www.anerty.net/software/file/jSAVF/](http://www.anerty.net/software/file/jSAVF/)

e permette di visionare la tabella dei contenuti di un ".savf" trascritto sul PC col metodo descritto al paragrafo [Salvataggi e ripristini](#), anche quando il salvataggio fu generato in illo tempore da un release precedente.

32.1.2. Si ottiene la visualizzazione:



The screenshot shows the jSAVF v1.60 application window. The left pane displays the file structure with 'Save files' expanded, showing 'SAVFModel[C:\XX\cnsavlib.savf]'. The right pane, titled 'Objects of C:\XX\cnsavlib.savf', displays a table of objects. The table has columns: Object, Type, Attributes, Owner, Size (KB), Description, and Creation. The objects listed include CORSO, AQLLE, CNA1AC, CNA1AD, CNA1BC, CNA1BD, CNA1CD, CNA1DD, CNA1ED, CNA1EX, CNA1PRT1, CNINI, CNUPRT1, CN1, TESTELSEIF, VUOTO2, CNA1AW, CNA1BW, CNA1CW, CNA1DW, CNA1EW, CNPRT, CORSO, CNA1Q, ALLMBR0, ALLMBR1, and ALLMBR2.

| Object     | Type    | Attributes | Owner  | Size (KB) | Description                                        | Creation            |
|------------|---------|------------|--------|-----------|----------------------------------------------------|---------------------|
| CORSO      | *LIB    | TEST       | NERONI | 144       | Corso Credits 2021                                 | 2021-02-12 12:14:20 |
| AQLLE      | *PGM    | CLLE       | NERONI | 88        | prove perm                                         | 2021-03-02 12:19:18 |
| CNA1AC     | *PGM    | CLLE       | NERONI | 112       | Anagrafica persone chiama visualizzazione no guida | 2021-03-02 11:24:07 |
| CNA1AD     | *PGM    | RPGL       | NERONI | 160       | Anagrafica persone no guida                        | 2021-03-08 20:28:08 |
| CNA1BC     | *PGM    | CLLE       | NERONI | 112       | Anagrafica persone chiama visualizzazione guida    | 2021-02-27 13:50:17 |
| CNA1BD     | *PGM    | RPGL       | NERONI | 172       | Anagrafica persone guida                           | 2021-03-08 20:28:10 |
| CNA1CD     | *PGM    | RPGL       | NERONI | 192       | Anagrafica persone decodifica                      | 2021-03-08 20:28:12 |
| CNA1DD     | *PGM    | RPGL       | NERONI | 240       | Anagrafica persone doppia guida                    | 2021-03-08 20:28:13 |
| CNA1ED     | *PGM    | RPGL       | NERONI | 304       | Anagrafica persone modifica                        | 2021-03-08 20:28:15 |
| CNA1EX     | *PGM    | RPGL       | NERONI | 184       | Anagrafica persone modifica alloca record a video  | 2021-03-07 13:41:55 |
| CNA1PRT1   | *PGM    | RPGL       | NERONI | 168       | Anagrafica persone stampa                          | 2021-02-23 12:16:26 |
| CNINI      | *PGM    | CLLE       | NERONI | 96        | Programma iniziale Allievo                         | 2021-03-05 16:24:47 |
| CNUPRT1    | *PGM    | RPGL       | NERONI | 140       | Stampa tutto da logico multifisico                 | 2021-02-17 17:10:28 |
| CN1        | *PGM    | CLLE       | NERONI | 96        | Hello boy                                          | 2021-02-15 12:32:36 |
| TESTELSEIF | *PGM    | RPGL       | NERONI | 112       | Test elseif                                        | 2021-03-05 12:35:33 |
| VUOTO2     | *PGM    | CLP        | NERONI | 36        | Solo PGM - ENDPGM : compilabile debuggabile        | 2021-02-15 11:54:33 |
| CNA1AW     | *FILE   | DSPF       | NERONI | 8         | Anagrafica persone no guida                        | 2021-03-08 20:27:23 |
| CNA1BW     | *FILE   | DSPF       | NERONI | 12        | Anagrafica persone guida                           | 2021-03-08 20:27:23 |
| CNA1CW     | *FILE   | DSPF       | NERONI | 12        | Anagrafica persone decodifica                      | 2021-03-08 20:27:24 |
| CNA1DW     | *FILE   | DSPF       | NERONI | 12        | Anagrafica persone doppia guida                    | 2021-03-09 00:14:50 |
| CNA1EW     | *FILE   | DSPF       | NERONI | 16        | Anagrafica persone modifica                        | 2021-03-08 20:27:24 |
| CNPRT      | *FILE   | PRTF       | NERONI | 8         | Stampa descritta interna 75 car                    | 2021-02-23 11:15:46 |
| CORSO      | *JOB    |            | NERONI | 8         | Corso Credits 2021. DescrizioneLavori              | 2016-01-21 00:37:12 |
| CNA1Q      | *QRYDFN | QRY        | NERONI | 24        | Anagrafica persone decodificate                    | 2021-02-19 12:22:36 |
| ALLMBR0    | *FILE   | PF         | NERONI | 24        |                                                    | 2021-03-08 20:35:50 |
| ALLMBR1    | *FILE   | PF         | NERONI | 40        | Output file for DSPFD TYPE('MBRLIST)               | 2021-03-08 20:35:50 |
| ALLMBR2    | *FILE   | PF         | NERONI | 40        |                                                    | 2021-03-09 13:37:57 |

32.1.3. La visualizzazione del file di salvataggio da parte del tool a bordo PC, a differenza della visualizzazione a bordo IBMi, permette anche di vedere o scaricare in forma di testo il contenuto dei membri sorgenti, utile per indagini esplorative e recuperi mirati.



### 33. Giornale

- 33.1. Il sistema IBMi è dotato di una gestione fortemente automatizzata di giornalizzazione degli oggetti, in particolare dei file fisici.
- 33.1.1. Il giornale è un automatismo che tiene la traccia cronologica degli eventi sui dati per tutti i record di tutti i file di una libreria conservando i dati prima e dopo ogni evento di modifica, aggiunta e cancellazione.
- 33.1.2. I dati vengono conservati senza tracciato perché tutti i file dati utilizzano un solo giornale.
- 33.1.3. Il giornale vero e proprio contiene solo la definizione di quali file sono sottoposti a giornalizzazione e poche regole sulle conservazione delle immagini dei dati prima e dopo gli eventi.
- 33.1.4. I dati copiati dai file risiedono invece nei ricevitori di giornale che si riempiono e vengono staccati periodicamente o a riempimento della loro misura massima e sostituiti da ricevitori vuoti.
- 33.1.5. L'insieme dei ricevitori attaccati e poi staccati da un giornale forma una catena di contenitori ordinatamente nomenclati in modo solitamente automatico con prefisso descrittivo e suffisso numerico progressivo.
- 33.1.6. I ricevitori staccati vanno conservati in linea per un tempo predeterminato e poi accantonati su supporti fuori linea.
- 33.1.7. Una regola di comportamento stabilirà che, oltre una certa data o oltre certi eventi, i ricevitori fuori linea vanno gettati.
- 33.1.8. I ricevitori formano quindi una catena che può essere troncata ma non interrotta perché per un uso corretto sono sempre necessari tutti i ricevitori a partire da quello attaccato al giornale fino al più vecchio da considerare senza nessuna interruzione della catena.
- 33.2. Si esemplifica la messa sotto giornale dei file presenti nella libreria del corso.
- 33.2.1. Crea una libreria di servizio per il giornale.  
> CRTLIB LIB(CORSOJRN) TYPE(\*TEST) TEXT('Corso Giornale')  
È stata creata la libreria CORSOJRN.
- 33.2.2. Crea un ricevitore dati.  
> CRTJRNRCV JRNRCV(CORSOJRN/JRN0000001) TEXT('Corso Ricevitore')  
Ricevitore di giornale JRN0000001 creato nella libreria CORSOJRN.
- 33.2.3. Crea il giornale vero e proprio.  
> CRTJRN JRN(CORSOJRN/JRN) JRNRCV(CORSOJRN/JRN0000001) TEXT('Corso Giornale')  
Giornale JRN creato nella libreria CORSOJRN.
- 33.2.4. Avvia un primo file sul giornale.  
> STRJRNPF FILE(CORSO/CNU1) JRN(CORSOJRN/JRN) IMAGES(\*BOTH)  
1 di 1 file hanno avviato la registrazione su giornale.
- 33.2.5. Avvia un secondo file sul giornale.  
> STRJRNPF FILE(CORSO/CNU2) JRN(CORSOJRN/JRN) IMAGES(\*BOTH)  
1 di 1 file hanno avviato la registrazione su giornale.
- 33.3. Caricamento di dati nel ricevitore del giornale.
- 33.3.1. Il caricamento avviene automaticamente a cura del sistema senza altre richieste oltre allo Start Journal Physical File per i file richiesti.
- 33.3.2. Quando un programma opera su un record di un file sotto giornale, ogni modifica, aggiunta o cancellazione di record provoca registrazioni automatiche nel giornale a cura del sistema in virtù delle sole definizioni contenute nel giornale stesso.
- 33.4. Per esplorare le registrazioni nel giornale, si usa il DSPJRN (Display Journal).
- 33.4.1. Dato il comando.  
> DSPJRN JRN(CORSOJRN/JRN)  
37 voci convertite dal giornale JRN in CORSOJRN.
- 33.4.2. Si ottiene l'elenco che segue. Ci si aiuta a capirne il contenuto portando il cursore sul campo da decifrare e premendo F1.



```

Visualizzazione delle voci di giornale

Giornale : JRN Libreria : CORSOJRN
Numero di sequenza più grande su questa schermata . : 0000000000000000012
Immettere le opzioni e premere Invio.
 5=Visualizzazione intera voce

Opz Sequenza Cod. Tipo Oggetto Libreria Lavoro Ora
 █ 1 J PR NU1 CORSO Z9NERO 20:26:35
 — 2 D JF NU1 CORSO Z9NERO 20:30:08
 — 3 F JM NU1 CORSO Z9NERO 20:30:08
 — 4 D JF NU2 CORSO Z9NERO 20:30:21
 — 5 F JM NU2 CORSO Z9NERO 20:30:21
 — 6 F CR NU1 CORSO Z9NERO 21:20:33
 — 7 F CR NU2 CORSO Z9NERO 21:20:39
 — 8 F OP NU1 CORSO Z9NERO 21:22:11
 — 9 R PT NU1 CORSO Z9NERO 21:22:11
 — 10 F CL NU1 CORSO Z9NERO 21:22:11
 — 11 F OP NU1 CORSO Z9NERO 21:22:11
 — 12 R PT NU1 CORSO Z9NERO 21:22:11

F3=Fine F12=Annullamento
 Segue...
```

33.4.3. Ad esempio, portando il cursore sulla colonna "Cod." e premendo F1, si ottiene l'elenco dei valori previsti in quella colonna.

```

Aiuto Codice - Aiuto

Il codice di giornale. Questo codice è definito dal sistema per
indicare il tipo di voce di giornale che è stata inserita. Seguono
i codici giornale e le descrizioni brevi relative ad essi:
o A - Voce account di sistema
o B - Operazione IFS
o C - Operazione controllo di commit
o D - Operazione file di database
o E - Operazione area dati
o F - Operazione membro file di database
o I - Operazione interna
o J - Operazione giornale o ricevitore di giornale
o L - Gestione delle licenze
o M - Dati gestione rete
o P - Voce ottimizzazione prestazioni
o Q - Operazione coda dati
o R - Operazione a livello record
o S - Servizio di posta distribuita per servizi di distribuzione
SNA (SNADS), avvisi di rete o struttura server di posta

Segue...

F2=Aiuto esteso F3=Fine F10=Inizio dati F12=Ann.
F13=Supporto informativo F14=Stampa aiuto

```

33.4.4. Altra colonna, "Tipo", altro testo esplicativo che aiuta ad interpretare il significato delle singole righe della prima videata.

```

Aiuto Tipo - Aiuto

Il tipo di voce di giornale. Questo codice è definito dal sistema
per indicare il tipo specifico della voce di giornale. Per ogni
codice giornale, ci sono diversi tipi di voce.

Per il codice giornale A sono possibili i seguenti tipi di voce
di giornale:

 o DP - Informazioni dirette relative alla stampa
 o JB - Segmento account lavoro
 o SP - Informazioni relative alla stampa in spool

Per il codice B di giornale i tipi possibili di voce di giornale
sono:

 o AA - Modifica attributo controllo
 o AJ - Avvio dell'applicazione
 o AT - Termine dell'applicazione
 o BD - Oggetto IFS cancellato

F2=Aiuto esteso F3=Fine F10=Inizio dati F12=Ann.
F13=Supporto informativo F14=Stampa aiuto

```

33.4.5. Il giornale contiene una riga per ogni variazione in aggiunta ai record dati con alcuni campi tecnici e l'immagine bruta dei dati aggiunti

33.4.6. Due righe per ogni modifica con l'immagine prima e dopo la modifica.

33.4.7. Una riga per ogni record cancellato dai dati con l'immagine prima della cancellazione.

33.4.8. Non ultime, una moltitudine di righe informative su aperture e chiusure dei file giornalizzati.

33.5. Il giornale permette di usare sui dati la tecnica di programmazione del "controllo di sincronia".

33.5.1. Il controllo di sincronia permette di riunire in un unico aggiornamento tutte le modifiche ai dati comprese tra due "commit".

33.5.2. In caso di errore di qualsiasi specie, la transazione viene conclusa non con un commit ma con un rollback.

33.5.3. La tecnica permette di non azzoppare mai una transazione e garantisce che una modifica multipla venga eseguita integralmente o per nulla.

33.5.4. Applicando il controllo di sincronia, il commit e il rollback nella programmazione applicativa, le transazioni complesse non diventeranno mai transazioni azzoppate.

33.6. Il giornale permette anche di debuggare a posteriori chi e quando ha fatto una certa modifica.

33.7. Ultima ma faticosa prestazione, è possibile disapplicare o applicare ai file le modifiche giornalizzate.

- 33.7.1. Questo, salvando i giornali tra un salvataggio totale e l'altro, permette comunque di ricostruire la fotografia dei dati ad un momento intermedio qualsiasi.
- 33.8. Si annota infine che l'uso del giornale deve prevedere anche lo scarico periodico fuori linea e la cancellazione dei ricevitori che costituiscono un ingombro non gestibile in linea.

## 34. Debug

### 34.1. Cosa è il debug.

34.1.1. Il debug è una modalità operativa che, durante l'esecuzione di un programma, permette di esaminare la sequenza di istruzioni eseguita vedendo il sorgente del programma in esecuzione con l'istruzione corrente in bella evidenza e con la possibilità di vedere e modificare estemporaneamente il valore delle variabili del programma.

### 34.2. Campo di applicazione del debug.

34.2.1. Quando, tramite un comando *CALL*, il sistema operativo viene richiesto di eseguire un programma, il sistema esegue le istruzioni sorgente una per una ma nella loro forma oggetto, ottenuta in precedenza tramite un compilatore. Il compilatore trasforma non le singole istruzioni ma l'insieme di tutte le istruzioni di un programma sorgente in un unico programma oggetto.

34.2.2. L'esecuzione è sequenziale a partire dalla prima riga di calcolo ma si sposta eventualmente a una riga diversa dalla successiva, se l'istruzione stessa lo richiede.

34.2.3. Durante il test di un programma, se nei risultati attesi si rilevano errori non immediatamente comprensibili semplicemente rileggendo e riveditando il codice sorgente, può essere molto utile utilizzare la funzione di debug.

### 34.3. Modalità operativa normale.

34.3.1. La modalità di esecuzione normale (non di debug) esegue le istruzioni senza evidenziare quali istruzioni vengano percorse e quali valori assumano le variabili interne al programma. Il test si limita perciò all'osservazione che le variabili, emesse dal programma su un qualsiasi supporto, siano quelle attese da chi esegue il test.

### 34.4. Modalità operativa debug.

34.4.1. La modalità di esecuzione debug, che si avvia con il comando *STRDBG* (Start debug), permette invece di eseguire il programma stando prima di eseguire certe istruzioni. Durante ogni sosta viene visualizzato il sorgente del programma evidenziando la riga che sarà eseguita quando si richiederà la prosecuzione dell'esecuzione previo esame, se necessario, del contenuto attuale di qualsiasi variabile del programma.

34.4.2. All'avvio del debug (*STRDBG*) su un certo programma da testare, occorre perciò scegliere (F6) una riga del sorgente del programma da chiamare definendo così almeno un breakpoint (punto di interruzione). Quando l'esecuzione starà per eseguire una riga di breakpoint, l'esecuzione stessa si fermerà e visualizzerà il video di debug.

34.4.3. Sul video di debug si potranno aggiungere o rimuovere breakpoint, visualizzare variabili nonché richiedere l'esecuzione:

- di un solo passo assumendo un provvisorio breakpoint sulla prossima istruzione da eseguire (F10);
- o di tutti i passi per arrivare al prossimo breakpoint prestabilito (F12).

Dopodiché si ripresenterà un nuovo video di debug.

34.4.4. La modalità di debug si chiude eseguendo *ENDDBG* (End debug). Opportuno farlo per non appesantire inutilmente il sistema. Necessario farlo **prima** di ricompilare il programma testato, dopo aver ovviamente trovato e rimediato un errore, perché succede altrimenti che la vecchia versione, dentro lo stesso lavoro, torni in esecuzione al posto della nuova.

### 34.5. Come creare programmi che in esecuzione possano essere esaminati tramite debug.

34.5.1. Per poter esaminare il sorgente di un programma durante il debug, occorre utilizzare, durante la creazione (o compilazione) di un programma, il parametro *DBGVIEW(\*LIST)* per le compilazioni ILE oppure *OPTION(\*LSTDBG)* per le compilazioni OPM. Tali parametri provocano l'incorporazione della lista di compilazione nel programma o nel modulo compilato. La lista incorporata è quella mostrata dalle operazioni di debug e non necessita dei sorgenti originali in linea al momento del debug.

34.5.2. Usando invece i parametri *DBGVIEW(\*SOURCE)* per ILE o *OPTION(\*SRCDBG)* per OPM, il programma creato non incorporerà nessuna forma di sorgente e perciò il debug mostrerà il sorgente originale che dovrà essere in linea nello stesso file e libreria dove era al momento della compilazione.

34.5.3. **ILE** (Integrated Language Environment) è l'ultima versione dei linguaggi, quella strutturata con moduli eccetera. **OPM** (Original Program Model) è il nome che stato affibbiato alla vecchia edizione dei linguaggi dopo la nascita dell'ILE (1996).

### 34.6. Debug interattivo per autocorreggere gli errori.

34.6.1. Anche se la chiamata di un programma, quando si lavora in produzione, avviene spesso in modo batch, il programmatore che vuole debuggarlo durante lo sviluppo provvederà preferibilmente ad eseguirlo in un lavoro interattivo, dove il debug risulta più semplice e più efficace con le modalità sopra descritte.

34.7. Debug batch utile in assistenza operativa.

34.8. Inclusione sorgenti nel compilato e riservatezza nella visualizzazione.

## 35. Sottosistemi e code

35.1. Definizione di Sottosistema, Coda di immissione, Coda di emissione, Coda messaggi.

35.1.1. Il sistema gestisce i lavori con l'ausilio di diversi tipi di oggetto:

|       |                       |                          |
|-------|-----------------------|--------------------------|
| *SBSD | Subsystem Description | Descrizione sottosistema |
| *JOBQ | Job Queue             | Coda di immissione       |
| *OUTQ | Output Queue          | Coda di emissione        |
| *MSGQ | Message Queue         | Coda messaggi            |

35.2. Sottosistema.

35.2.1. Il sottosistema è l'oggetto che si incarica dell'esecuzione dei lavori.

35.2.2. Viene costituito eseguendo un comando CL che lo crea e una ulteriore serie di comandi che lo completano.

35.2.3. Non avendo un proprio sorgente, è opportuno conservarne la serie dei comandi di creazione e modifica in un sorgente CL che, eseguito tramite SBMJOB, lo ricrea integralmente.

35.3. Sottosistema batch.

35.3.1. Ha specifiche che gli permettono di eseguire lavori batch.

35.3.2. Ad esempio, con la seguente jobstream, si crea un sottosistema batch e se ne collauda il funzionamento:

```
//BCHJOB JOB(SBSD) JOBD(CORSO/CORSO) ENDSEV(60)
/*
/* Claudio Neroni 25-04-2021 Creato.
/* CNSBSD
/* Crea un sottosistema e lo collauda con la sottomissione di un lavoro
/* che manda un messaggio in una coda di servizio
/*
/* Chiude il sottosistema eventualmente gia' aperto.
ENDSBS SBS(CNSBSD)
/* Da' il tempo al sottosistema di chiudersi.
DLYJOB DLY(4)

/* Cancella la coda lavori eventualmente gia' presente.
DLTJOBQ JOBQ(CORSO/CNJOBQ)
/* Cancella il sottosistema eventualmente gia' presente.
DLTSBSD SBS(CORSO/CNSBSD)
/* Cancella la coda messaggi di comodo.
DLTMSGQ MSGQ(CORSO/CNMSGQ)

/* Crea la coda messaggi di comodo.
CRTMSGQ MSGQ(CORSO/CNMSGQ) TEXT('Prova per Corso')

/* Crea la coda lavori.
CRTJOBQ JOBQ(CORSO/CNJOBQ) TEXT('Prova per Corso')

/* Crea il sottosistema.
CRTSBSD SBS(CORSO/CNSBSD) POOLS((1 *BASE)) +
TEXT('Prova per Corso')

/* Aggiunge la coda lavori al sottosistema.
ADDJOBQE SBS(CORSO/CNSBSD) JOBQ(CORSO/CNJOBQ)

/* Aggiunge la specifica di instradamento al sottosistema.
ADDRTE SBS(CNSBSD) SEQNBR(1) CMPVAL(*ANY) PGM(QSYS/QCMD) +
CLS(QSYS/QBATCH)

/* Avvia il sottosistema.
STRSBS SBS(CORSO/CNSBSD)

/* Sottomette un lavoro tramite il sottosistema.
SBMJOB CMD(SNDMSG MSG('Lavoro sottomesso nella coda CNJOBQ ed +
eseguito dal sottosistema CNSBSD!')) TOMSGQ(CNMSGQ)) +
JOB(CNSBSDTEST) JOBD(CORSO/CORSO)

//ENDBCHJOB
```

35.3.3. Vedi il sorgente:

[www.neroni.it/CORSO2023/A-CORSO.CORSO/CNSBSD.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNSBSD.txt)

35.4. Sottosistema interattivo.

35.4.1. Ha specifiche che gli permettono di eseguire lavori interattivi.

35.4.2. In particolare, specifiche che permettono a dei terminali 5250 (o emulazione 5250) di collegarsi tramite il video di signon fornito dal sottosistema stesso.

35.4.3. Un sottosistema interattivo non necessita di una coda lavori se si limita a far partire lavori interattivi tramite il video di signon. Ne fa uso se riceve il "Transfer job" di un interattivo già avviato.

#### 35.4.4. Si esemplifica qui la jobstream di creazione di un sottosistema interattivo.

```
//BCHJOB JOB(CNINTER) JOBD(CORSO/CORSO) ENDSEV(60)
/*
/* Claudio Neroni 04-06-2021 Creato.
/* CNINTER
/* Crea sottosistema interattivo
/*
/* Chiude il sottosistema eventualmente gia' aperto.
ENDSBS SBS(CNINTER)
/* Da' il tempo al sottosistema di chiudersi.
DLYJOB DLY(4)

/* Cancella la coda lavori eventualmente gia' presente.
DLTJOBQ JOBQ(CORSO/CNINTER)
/* Cancella il sottosistema eventualmente gia' presente.
DLTSBSD SBSD(CORSO/CNINTER)

/* Crea la coda lavori.
CRTJOBQ JOBQ(CORSO/CNINTER) TEXT('Prova per Corso')

/* Crea il sottosistema.
CRTSBSD SBSD(CORSO/CNINTER) POOLS((1 *BASE) (2 *INTERACT)) +
TEXT('Prova per Corso')

/* Aggiunge voce stazione lavoro.
/* ADDWSE SBSD(CORSO/CNINTER) WRKSTNTYPE(*ALL) */
ADDWSE SBSD(CORSO/CNINTER) WRKSTN(CNPROVA)

/* Aggiunge la coda lavori al sottosistema.
ADDJOBQE SBSD(CORSO/CNINTER) JOBQ(CORSO/CNINTER) MAXACT(*NOMAX)

/* Aggiunge la specifica di instradamento al sottosistema.
ADDRTGE SBSD(CORSO/CNINTER) SEQNBR(9999) CMPVAL(*ANY) +
PGM(QSYS/QCMD) CLS(QGPL/QINTER)

//ENDBCHJOB
```

#### 35.5. Coda immissione.

35.5.1. La coda di immissione riceve la richiesta di esecuzione di un lavoro (lavoro sottomesso).

35.5.2. Il lavoro in coda viene eseguito se:

- 35.5.2.1. la coda è collegata ad un sottosistema,
- 35.5.2.2. il sottosistema è avviato,
- 35.5.2.3. la coda è rilasciata (non congelata),
- 35.5.2.4. il lavoro è pronto (non congelato in coda),
- 35.5.2.5. provenienti dalla stessa coda è in esecuzione contemporanea un numero di lavori inferiori al massimo permesso per la coda.

35.5.3. Solo i lavori interattivi nati dal collegamento di un terminale 5250 (o emulazione 5250) al video di signon fornito da un sottosistema "interattivo", non passano per una coda di immissione.

#### 35.6. Coda emissione.

35.6.1. La coda di emissione riceve le stampe di spool generate dall'esecuzione di un lavoro.

35.6.2. Risulta indispensabile perché è possibile ma non praticata la tecnica di stampa diretta sulle stampanti senza il passaggio dallo spool.

#### 35.7. Coda messaggi.

35.7.1. Esistono due tipi di code messaggi:

- 35.7.1.1. coda messaggi di programma,
- 35.7.1.2. oggetto coda messaggi.

35.7.2. Riguardo al primo tipo:

- 35.7.2.1. I programmi di un lavoro comunicano tra loro e con gli altri lavori principalmente ma non esclusivamente tramite messaggi.
- 35.7.2.2. L'insieme delle code messaggi dei programmi di un lavoro costituisce la coda esterna del lavoro che ha un ruolo essenziale nei confronti dei programmi che il sistema usa per eseguire il lavoro stesso, primo fra tutti il QCMD.
- 35.7.2.3. L'esecutore comandi prende dalla coda esterna i messaggi di richiesta e li esegue.
- 35.7.2.4. Il log di un lavoro ("joblog") si identifica con la lettura della coda esterna del lavoro con la fotografia di tutti i suoi messaggi.

35.7.3. Riguardo al secondo tipo:

- 35.7.3.1. Si tratta dell'oggetto coda messaggi che è un contenitore esterno al lavoro con il quale il lavoro può comunicare con la stessa tecnologia dei messaggi usati dentro il lavoro.
- 35.7.3.2. La coda messaggi può ricevere i messaggi generati dall'esecuzione di un lavoro.
- 35.7.3.3. Il lavoro può interrogare una coda messaggi per riceverne dati o scelte.
- 35.7.3.4. In particolare il programma può interagire con la coda messaggi dell'operatore QSYSOPR o con la coda messaggi dell'utente o del terminale lanciatore del lavoro.

## 36. Gestione dei lavori

36.1. Definizione e identificazione di un lavoro: Nome lavoro, Utente lavoro, Numero lavoro.

36.1.1. Un lavoro viene identificato sul sistema da tre parametri che sono

Job Name    Nome lavoro    Nome di 10 caratteri

Job User    Utente lavoro   Nome di 10 caratteri

Job Number   Numero lavoro   Alfanumerico contenente sempre 6 cifre

36.1.2. L'insieme dei tre campi identifica applicativamente in modo sufficientemente univoco il lavoro. Il numero lavoro è un progressivo incrementato ad ogni lavoro che nasce sul sistema e le sei cifre sono azzerate dopo il lavoro 999999 e la numerazione ricicla senza riguardo ad eventuali doppioni.

36.1.3. I lavori spariscono dalla tabella lavori dopo la cancellazione della loro ultima emissione in spool o, se mancano di spool, alla fine della loro esecuzione. Si tenga conto che le log dei job finiti diventano stampe e che le joblog si conservano per più tempo delle stampe applicative. Può quindi succedere che accidentalmente in un tempo abbastanza lungo si verifichi un identificativo doppio di job/user/jobnumber. Poco male. Il sistema distingue i job con una chiave sicuramente unica che si visualizza solo tramite certe API.

36.2. **WRKJOB** (Work with Job) permette di visualizzare le caratteristiche di un lavoro. Numerosi comandi di controllo dei lavori presentano degli elenchi sui quali è possibile individuare il lavoro che interessa per controllarlo tramite le opzioni.

36.2.1. Il comando fondamentale per il controllo di un certo lavoro è **WRKJOB** (Work with Job), documentato al link:

[https://www.ibm.com/support/knowledgecenter/ssw\\_ibm\\_i\\_73/cl/wrkjob.htm](https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_73/cl/wrkjob.htm)

Può essere chiamato senza parametri e allora presenta il lavoro corrente.

Può essere chiamato con l'identificazione di un lavoro e allora presenta il lavoro cercato:

> **WRKJOB** JOB(545965/UFNERONI/H1NERO)

Comunque chiamato presenta un menù a video che permette di chiedere specifici dettagli ed eseguire eventuali modifiche sul lavoro visualizzato.

```
Gestione lavoro
Lavoro: H1NERO Utente: UFNERONI Sistema: S65D69DA
Numero: 509389
Selezionare una delle seguenti:
 1. Visualizzazione attributi di stato del lavoro
 2. Visualizzazione attributi di definizione del lavoro
 3. Visualizzazione attributi di esecuzione del lavoro, se attivo
 4. Gestione dei file di spool
 10. Visualizz. registrazione lavoro, se attivo, in coda lavori o sospeso
 11. Visualizzazione stack di chiamata, se attivo
 12. Gestione dei blocchi, se attivi
 13. Visualizzazione elenco librerie, se attivo
 14. Visualizzazione file aperti, se attivi
 15. Visualizzazione sostituzioni di file, se attive
 16. Visualizzazione stato controllo convalida, se attivo
Scelta o comando
==>
F3=Fine F4=Richiesta F9=Duplic. F12=Ann.
```

```
Gestione lavoro
Lavoro: H1NERO Utente: UFNERONI Sistema: S65D69DA
Numero: 509389
Selezionare una delle seguenti:
 17. Visualizzazione stato comunicazioni, se attivo
 18. Visualizzazione gruppi di attivazione, se attivi
 19. Gestione esclusioni reciproche (di lavoro), se attive
 20. Gestione sottoprocessi, se attivi
 21. Gestione attributi libreria supporto magnetico, se attivi
 22. Visualizzazione variabili di ambiente, se attivo
 30. Tutte le opzioni precedenti
 40. Modifica lavoro
 41. Fine lavoro
 42. Congelamento lavoro
Scelta o comando
==>
F3=Fine F4=Richiesta F9=Duplic. F12=Ann.
```



```

Gestione lavoro
Lavoro: H1NERO Utente: UFNERONI Numero: 509389 Sistema: S65D69DA
Selezionare una delle seguenti:
 43. Rilascio lavoro
 45. Gestione di Java Virtual Machine, se attiva

Scelta o comando
==>
F3=Fine F4=Richiesta F9=Duplic. F12=Ann.

```

36.2.2. Chiamato con il solo nome, se presenti doppioni, ne presenta l'elenco per la scelta prima di visualizzare il menù di ingresso di quello cercato.

```

> WRKJOB JOB (JAMALOC)
È stato trovato un lavoro 062169/UFBA/JAMALOC doppio.
È stato trovato un lavoro 165670/UFBA/JAMALOC doppio.
È stato trovato un lavoro 388505/UFBA/JAMALOC doppio.
È stato trovato un lavoro 476086/UFBA/JAMALOC doppio.
Fine dei nomi duplicati.

```

```

Seleziona lavoro
27/11/20 19:17:33 CET S65D69DA
Immettere l'opzione e premere Invio.
1=Selezione

```

| Opz. | Lavoro  | Utente | Numero | Tipo  | -----Stato----- | Inoltrato nel sistema |
|------|---------|--------|--------|-------|-----------------|-----------------------|
| 1    | JAMALOC | UFBA   | 476086 | BATCH | OUTQ            | 21/11/20              |
| 2    | JAMALOC | UFBA   | 388505 | BATCH | OUTQ            | 14/11/20              |
| 3    | JAMALOC | UFBA   | 165670 | BATCH | OUTQ            | 07/11/20              |
| 4    | JAMALOC | UFBA   | 062169 | BATCH | OUTQ            | 31/10/20              |

```

F3=Fine F12=Ann.
Sono stati trovati dei lavori duplicati

```

36.2.3. Numerosi comandi di controllo dei lavori presentano degli elenchi sui quali è possibile individuare il lavoro che interessa per controllarlo tramite le opzioni.

36.2.4. La versione stampata dà informazioni più che abbondanti e in gran parte non importanti.

```

> WRKJOB OUTPUT(*PRINT)

```

```

5770SSI V7R3M0 160422 Gestione del lavoro 24/11/20 20:54:52 Pag. 1
Lavoro . : H1NERO Utente . : UFNERONI Numero . . . : 509389
Attributi stato lavoro
Stato del lavoro : ACTIVE
Profilo utente corrente : UFNERONI
Identità utente lavoro : UFNERONI
Impostato da : *DEFAULT
Imnesso nel sistema:
Data : 24/11/20
Ora : 14:28:02
Avviato:
Data : 24/11/20
Ora : 14:28:02
Sottosistema : QINTER
ID lotto sottosistemi : 2
Tipo di lavoro : INTER
Ambiente speciale : *NONE
Codice di ritorno programma : 1
Richiesta fine controllata : NO
Sistema : S65D69DA

```

```

5770SSI V7R3M0 160422 Gestione del lavoro 24/11/20 20:54:52 Pag. 2
Lavoro . : H1NERO Utente . : UFNERONI Numero . . . : 509389
Attributi di definizione del lavoro
Descrizione lavoro : JOBQ QDFTJOBQ
Libreria : QGPL
Unità ASP : *SYSBAS
Coda lavori : JOBQ
Libreria :
Unità ASP :
Priorità lavoro (sulla coda lavori) : JOBPTY 5
Priorità emissione (sulla coda di emissione) : OUTPTY 5
Severità di uscita : ENDSEV 30

```

```

Registrazione dei messaggi: LOG
Livello : 4
Severità : 0
Testo : *SECLVL
Registrazione comandi programma CL : LOGCLPGM *YES
Emissione registrazione lavoro : LOGOUTPUT *JOBEND
Unità di stampa : PRTDEV QSYSPRT
Coda di emissione predefinita : OUTQ UFNERONI
Libreria : QGPL
Data lavoro : DATE 24/11/2020
Formato data : DATFMT *DMY
Separatore data : DATSEP /
Separatore ora : TIMSEP :
Ora lavoro: DATETIME
Data corrente : 24/11/2020
Ora corrente : 20:54:52
Giorno della settimana : DAYOFWEEK *TUE
Fuso orario : TIMZON QP0100CET2
Scostamento : TIMOFFSET +01:00
Nome completo : TIMZONFULL Central European Time
Nome abbreviato : TIMZONABBR CET
Scostamento anno : YEAROFS 0
Switch lavoro : SWS 00000000
Risposta al messaggio di interrogazione . . . : INQMSGRPY *RQD
Codice account : ACGCDE
Testo stampa : PRTTXT '
COnversione DDM : DDMCNV *KEEP
Gestione interruzione messaggio : BRKMSG *NORMAL
Messaggio di stato : STSMMSG *NORMAL
Azione di ripristino unità : DEVRCYACN *MSG
Lotto per la fine della frazione di tempo . . : TSEPOOL *NONE
Formato per tasto Stampa : PRTKEYFMT *PRTHDR
Sequenza di ordinamento : SRTSEQ *HEX
Libreria :
Identificativo lingua : LANGID ITA
Identificativo paese o regione : CNTRYID IT
CCSID (Coded character set ID) : CCSID 280
CCSID predefinito : DFTCCSID 280
Dimensione massima coda messaggi lavoro . . : JOBMSGQMX 64
Azione completa coda messaggi lavoro . . . : JOBMSGQFL *PRTWRAP
Consentire più sottoprocessi : ALWMLTHD *NO
Formato decimale : DECFMT J
Controllo identificativo carattere : CHRIDCTL *DEVD
Gruppo ASP (Auxiliary storage pool) : ASPGRP *NONE

```

```

5770SS1 V7R3M0 160422 Gestione del lavoro 24/11/20 20:54:52 Pag. 3
Lavoro . : H1NERO Utente . : UFNERONI Numero : 509389
Azione file di spool : SPLFACN *KEEP
Gruppo carico di lavoro : WLCGRP *NONE
Priorità risorse processore : PRCRSCTY *SYSCTL
Attributi di esecuzione del lavoro
Priorità di esecuzione : RUNPTY 20
Frazione di tempo in millisecondi : TIMESLICE 2000
Trasferibile in memoria ausiliaria : PURGE *YES
Tempo di attesa predefinito in secondi . . . : DFTWAIT 30
Tempo massimo CPU in millisecondi : CPUTIME *NOMAX
Tempo CPU utilizzato : 33
Memoria massima temporanea in megabyte . . : MAXTMPSTG *NOMAX
Memoria temporanea utilizzata : 4
Picco memoria temporanea utilizzata . . . : 4
Numero massimo sottoprocessi : MAXTHD *NOMAX
Sottoprocessi : 1
Affinità risorse sottoprocesso: THDRSCAFN
Gruppo : *NOGROUP
Livello : *NORMAL
Gruppo affinità risorse : RSCAFNGRP *NO

```

```

5770SS1 V7R3M0 160422 Gestione del lavoro 24/11/20 20:54:52 Pag. 4
Lavoro . : H1NERO Utente . : UFNERONI Numero : 509389
File di emissione in spool
Unità o Tot. Pag. Num.
File Coda Dati uten. Stato pag. cor. Copie Tipo mod. Ptà Data Ora file Coda Libreria
QPDSPJOB UFNERONI OPN 2 1 *STD 5 24/11/20 20:54:52 1 UFNERONI QGPL
Stack chiamata lavoro
Sottoproc. : 0000008A
Modulo o Ctl
Tipo Programma Id istruzioni Istruzione Gruppo attivazione Tipo esteso Bdy
QCMD QSYS 04FA *DFTACTGRP 0000000000000001 Y
1 QCMD QSYS 01C8 *DFTACTGRP 0000000000000001 N
Blocchi del lavoro
Stato lavoro : ACTIVE
Oggetto Libreria Tipo oggetto Blocco Stato Bloccchi Unità Ambito Sottoprocesso
H1NERO QSYS *DEVD *EXCLRD CONG *JOB
NERONI QSYS *LIB *SHRRD CONG *JOB
NERONI2 QSYS *LIB *SHRRD CONG *JOB
QDUI132 QSYS *FILE-DSP *SHRNUP CONG *JOB
QGMHGS D QSYS *PNLGRP *SHRNUP CONG *JOB
QGPL QSYS *LIB *SHRRD CONG *JOB
QHLP SYS QSYS *LIB *SHRRD CONG *JOB
QPDSPJOB QSYS *FILE-PRT *SHRNUP CONG *JOB
QSYS QSYS *LIB *SHRRD CONG *JOB

```

|            |         |           |        |      |     |                  |
|------------|---------|-----------|--------|------|-----|------------------|
| QSYS2      | QSYS    | *LIB      | *SHRRD | CONG |     | *JOB             |
| QTEMP      | QSYS    | *LIB      | *SHRRD | CONG |     | *JOB             |
| QUSRSYS    | QSYS    | *LIB      | *SHRRD | CONG |     | *JOB             |
| Q04079N029 | QSPL    | *FILE-PHY | *SHRRD | CONG | YES | *JOB             |
| UFGRP      | QSYS    | *USRPRF   | *SHRRD | CONG |     | *JOB             |
|            |         |           | *SHRRD | CONG |     | *JOB             |
|            |         |           | *SHRRD | CONG |     | *JOB             |
| UFNERONI   | QSYS    | *USRPRF   | *SHRRD | CONG |     | *JOB             |
|            |         |           | *SHRRD | CONG |     | *JOB             |
|            |         |           | *SHRRD | CONG |     | *JOB             |
| UFNERONI   | QUSRSYS | *MSGQ     | *EXCL  | CONG |     | *THREAD 0000008A |

|                                               |  |                         |  |                         |                   |                               |        |                             |
|-----------------------------------------------|--|-------------------------|--|-------------------------|-------------------|-------------------------------|--------|-----------------------------|
| 5770SS1 V7R3M0 160422                         |  | Gestione del lavoro     |  |                         | 24/11/20 20:54:52 |                               | Pag. 5 |                             |
| Lavoro . . : HINERO                           |  | Utente . . : UFNERONI   |  | Numero . . . . : 509389 |                   |                               |        |                             |
| Blocchi sui membri del lavoro                 |  |                         |  |                         |                   |                               |        |                             |
| File . . . . : Q04079N029                     |  | Libreria . . . . : QSPL |  | Tipo . . . . : PHY      |                   | Unità ASP. . . : *SYSBAS      |        |                             |
| Tipo                                          |  |                         |  |                         |                   |                               |        |                             |
| Membro                                        |  | blocco                  |  | Blocco                  |                   | Stato                         |        | Ambito                      |
| Q829307215                                    |  | MBR                     |  | *SHRRD                  |                   | CONG                          |        | *JOB                        |
|                                               |  | DATI                    |  | *SHRUPD                 |                   | CONG                          |        | *JOB                        |
| Blocchi sui record del lavoro                 |  |                         |  |                         |                   |                               |        |                             |
| Numero                                        |  |                         |  |                         |                   |                               |        |                             |
| File                                          |  | Libreria                |  | Membro                  |                   | Record                        |        | Stato                       |
| (Il lavoro non ha blocchi sui record)         |  |                         |  |                         |                   |                               |        |                             |
| Elenco delle librerie del lavoro              |  |                         |  |                         |                   |                               |        |                             |
| Unità                                         |  |                         |  |                         |                   |                               |        |                             |
| Libreria                                      |  | Tipo                    |  | ASP                     |                   | Testo                         |        |                             |
| NERONIQ                                       |  | SYS                     |  |                         |                   | Claudio Neroni                |        |                             |
| JFMTDTA                                       |  | SYS                     |  |                         |                   | Tarocco di FMTDTA per loggare |        |                             |
| QSYS                                          |  | SYS                     |  |                         |                   | Libreria di sistema           |        |                             |
| QSYS2                                         |  | SYS                     |  |                         |                   | Libreria di sistema per CPI   |        |                             |
| QHLPSYS                                       |  | SYS                     |  |                         |                   |                               |        |                             |
| QUSRSYS                                       |  | SYS                     |  |                         |                   | System Library for Users      |        |                             |
| QTEMP                                         |  | USR                     |  |                         |                   |                               |        |                             |
| NERONI2                                       |  | USR                     |  |                         |                   | Claudio Neroni Utility        |        |                             |
| QGPL                                          |  | USR                     |  |                         |                   | General Purpose Library       |        |                             |
| File aperti                                   |  |                         |  |                         |                   |                               |        |                             |
| Numero di percorsi ai dati aperti . . . . . : |  |                         |  | 2                       |                   |                               |        |                             |
| Record                                        |  |                         |  |                         |                   |                               |        |                             |
| File                                          |  | Libreria                |  | Membro/                 |                   | Formato                       |        | Tipo                        |
|                                               |  | Unità                   |  | record                  |                   | file                          |        | I/E                         |
| QDUI132                                       |  | QSYS                    |  | HINERO                  |                   | USRRCD                        |        | DSP                         |
| QPDSPJOB                                      |  | QSYS                    |  | *SPOOL                  |                   | DETAIL                        |        | PRT                         |
|                                               |  |                         |  |                         |                   | 145                           |        | O YES 1                     |
| Conteg. -Apertura--                           |  |                         |  |                         |                   |                               |        |                             |
|                                               |  |                         |  |                         |                   | 43                            |        | IO NO                       |
|                                               |  |                         |  |                         |                   | 145                           |        | O YES 1                     |
| Record relativo                               |  |                         |  |                         |                   |                               |        |                             |
|                                               |  |                         |  |                         |                   |                               |        | *ACTGRPDEFN                 |
|                                               |  |                         |  |                         |                   |                               |        | *DFTACTGRP 0000000000000002 |
| Ambito                                        |  |                         |  |                         |                   |                               |        |                             |
|                                               |  |                         |  |                         |                   |                               |        | *ACTGRPDEFN                 |
|                                               |  |                         |  |                         |                   |                               |        | *DFTACTGRP 0000000000000002 |
| Gruppo attivazione                            |  |                         |  |                         |                   |                               |        |                             |

|                                                   |  |                       |  |                     |  |                          |  |                         |  |
|---------------------------------------------------|--|-----------------------|--|---------------------|--|--------------------------|--|-------------------------|--|
| 5770SS1 V7R3M0 160422                             |  |                       |  | Gestione del lavoro |  | 24/11/20 20:54:52        |  | Pag. 6                  |  |
| Lavoro . . :                                      |  | HINERO                |  | Utente . . :        |  | UFNERONI                 |  | Numero . . . . : 509389 |  |
| Tutte le sovrascritture file                      |  |                       |  |                     |  |                          |  |                         |  |
| Livello chiamata . . . . . *                      |  |                       |  |                     |  |                          |  |                         |  |
| File                                              |  | Livello               |  | Tipo                |  | Specifiche parole chiave |  |                         |  |
| (Nessuna sovrascrittura file)                     |  |                       |  |                     |  |                          |  |                         |  |
| Stato controllo di commit                         |  |                       |  |                     |  |                          |  |                         |  |
| (Nessuna definizione di commit è attiva)          |  |                       |  |                     |  |                          |  |                         |  |
| Visualizzazione stato comunicazioni               |  |                       |  |                     |  |                          |  |                         |  |
| -----Comunicazioni----- Numero di operazioni----- |  |                       |  |                     |  |                          |  |                         |  |
| Identificativo                                    |  | Metodo                |  | Emissione           |  | Immissione               |  | Altro Operazione        |  |
| (Non è attivo alcun identificativo comunicazioni) |  |                       |  |                     |  |                          |  |                         |  |
| Gruppi attivazione lavoro                         |  |                       |  |                     |  |                          |  |                         |  |
| -----Gruppo attivazione-----                      |  |                       |  |                     |  |                          |  |                         |  |
| Nome                                              |  | Numero                |  | Indicatore In uso   |  | Programma                |  | Libreria                |  |
| *DFTACTGRP                                        |  | 000000000000000001    |  | Y                   |  |                          |  |                         |  |
| *DFTACTGRP                                        |  | 000000000000000002    |  | Y                   |  |                          |  |                         |  |
| QLGLOCAL                                          |  | 00000000000000011     |  | N                   |  | QLGLOCAL                 |  | QSYS                    |  |
| Esclusioni reciproche di lavoro                   |  |                       |  |                     |  |                          |  |                         |  |
| Stato                                             |  |                       |  |                     |  |                          |  |                         |  |
| Esclusione reciproca                              |  | Esclusione reciproca  |  |                     |  |                          |  |                         |  |
| (Il lavoro non ha esclusioni reciproche)          |  |                       |  |                     |  |                          |  |                         |  |
| Sottoprocessi                                     |  |                       |  |                     |  |                          |  |                         |  |
| Sottopr.                                          |  | Nome                  |  | Stato               |  | Totale I/E               |  | Priorità                |  |
| 0000008A                                          |  |                       |  | RUN                 |  | CPU Aus                  |  | esecuzione              |  |
|                                                   |  |                       |  |                     |  | 0,035 702                |  | 20                      |  |
| Attributi MLB                                     |  |                       |  |                     |  |                          |  |                         |  |
| Unità                                             |  | Priorità assegnazione |  | Tempo di attesa     |  | Tempo di attesa          |  |                         |  |
| libreria                                          |  | risorse               |  | montaggio iniziale  |  | montaggio fine volume    |  |                         |  |
| TAPMLBSE                                          |  | 20                    |  | 5                   |  | 5                        |  |                         |  |
| TAPMLBSE6                                         |  | 20                    |  | 5                   |  | 5                        |  |                         |  |

|                       |                       |                    |                 |                       |        |                   |        |
|-----------------------|-----------------------|--------------------|-----------------|-----------------------|--------|-------------------|--------|
| 5770SS1 V7R3M0 160422 | Gestione del lavoro   |                    |                 |                       |        | 24/11/20 20:54:52 | Pag. 7 |
| Lavoro . . :          | HINERO                | Utente . . :       | UFNERONI        | Numero . . . . :      | 509389 |                   |        |
| Attributi MLB         |                       |                    |                 |                       |        |                   |        |
| Unità                 | Priorità assegnazione | Tempo di attesa    |                 |                       |        |                   |        |
| libreria              | risorse               | montaggio iniziale | Tempo di attesa | montaggio fine volume |        |                   |        |
| TAPMLBSV              | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLBSV6             | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB11              | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB140             | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB141             | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB146             | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB147             | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB20              | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB30              | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB31              | 20                    | 5                  | 5               | 5                     |        |                   |        |
| TAPMLB94              | 20                    | 5                  | 5               | 5                     |        |                   |        |
| VTLB1                 | 20                    | 5                  | 5               | 5                     |        |                   |        |
| VTLB1DR               | 20                    | 5                  | 5               | 5                     |        |                   |        |

|                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |   |       |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|-------|
| VTLC1                 | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLC1DR               | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLH1                 | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLH1DR               | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLH2                 | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLS1                 | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLS1DR               | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLU1                 | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLU1DR               | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLV1                 | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLX1                 | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLX1DR               | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| VTLZ1                 | 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 5 | 5     |
| Variabili di ambiente |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |   |       |
| Nome                  | Valore                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |   | CCSID |
| CLASSPATH             | '.:./VisionSolutions/./VisionSolutions/lib/vsiJListener.jar:/VisionSolutions/1'ib/vsiJServer.jar:/VisionSolutions/lib/vsiGeneral.jar:/VisionSolutions/lib/vsiOMS.jar:/VisionSolutions/lib/vsiClusterJNI.jar:/VisionSolutions/lib/vsiEvent'Notification.jar:/VisionSolutions/lib/jdom.jar:/Visionsolutions/lib/xercesImp'1.jar:/VisionSolutions/lib/xmlParserAPIs.jar:/VisionSolutions/lib/xalan.jar:/QIBM/ProdData/Java400/ext/db2_classes.jar:/QIBM/ProdData/OS400/jt400/lib/jt40'0Native.jar' |   | 280   |
| EVSZIE                | '1000000'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |   | 280   |
| SKIPWASPLUGIN         | '1'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |   | 37    |
| QIBM_XJ9_TRACE_FILE   | '/home/j9trace/J9Trace%PID.txt'                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |   | 280   |
| QIBM_XJ9_TRACE        | '*ALL'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |   | 280   |
| JOBMAN NOROOT         | 'NO'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |   | 280   |
| UF_SYSTEM_LETTER      | 'H'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |   | 280   |
| SPCP                  | ' 280'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |   | 280   |
| SPOS                  | 'CEDLIB'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |   | 280   |
| QIBM_QSH_CMD_OUTPUT   | 'NONE'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |   | 280   |
| QIBM_MULTI_THREADED   | 'Y'                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |   | 280   |
| STACK_USRSPC_LIB      | 'NETTMP '                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |   | 280   |
| ARCAD_HOME            | '/ArcADIfsPrd/arcad_home'                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |   | 1144  |

|                                             |                        |                         |        |
|---------------------------------------------|------------------------|-------------------------|--------|
| 5770SS1 V7R3M0 160422                       | Gestione del lavoro    | 24/11/20 20:54:52       | Pag. 8 |
| Lavoro . : H1NERO                           | Utente . . : UFNERONI  | Numero . . . . : 509389 |        |
| Nome                                        | Valore                 |                         | CCSID  |
| MAIL_DEFAULT                                | 'noreply@objectway.it' |                         | 280    |
| LOCAL_TAPE_BRMS                             | 'TAPMLBSE'             |                         | 280    |
| REMOTE_TAPE_BRMS                            | 'TAPMLBSV'             |                         | 280    |
| PASE_USRGRP_LIMITED                         | 'N'                    |                         | 280    |
| QIBM_RPG_DISABLE_TIMESTAMP_MICROSEC         | 'Y'                    |                         | 280    |
| ILE_DEBUGGER_1                              | 'ALLOW WIDE_SCREEN'    |                         | 280    |
| * * * * F I N E D E L L A L I S T A * * * * |                        |                         |        |

36.3. WRKACTJOB (Work with Active Jobs) elenca i lavori attivi.

36.3.1. Il comando **WRKACTJOB** (Work with Active Jobs), documentato al link [https://www.ibm.com/support/knowledgecenter/ssw\\_ibm\\_i\\_73/cl/wrkactjob.htm](https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_73/cl/wrkactjob.htm) , chiamato senza parametri, presenta una videata con l'elenco dei lavori attivi sul sistema, un lavoro per riga e per ogni lavoro alcuni parametri significativi del lavoro stesso. I lavori possono essere esaminati tramite le opzioni presentate sulla testata. Il tempo dell'esame può essere allungato con F5 o fatto ripartire con F10.

|                                                                          |              |           |      |       |                |                       |
|--------------------------------------------------------------------------|--------------|-----------|------|-------|----------------|-----------------------|
| Gestione lavori attivi                                                   |              |           |      |       |                | S65D69DA              |
| % CPU: 26,2      Tempo trascorso: 00:00:28      Lavori attivi: 600       |              |           |      |       |                | 24/11/20 22:32:00 CET |
| Immettere opzioni, premere Invio.                                        |              |           |      |       |                |                       |
| 2=Modif.    3=Congelamento    4=Fine    5=Gest.    6=Ril.                |              |           |      |       |                |                       |
| 7=Visualiz. messaggio    8=Gestione file di spool    13=Scollegamento... |              |           |      |       |                |                       |
| Utente                                                                   |              |           |      |       |                |                       |
| Opz                                                                      | Sottosis/Lav | corrente  | Tipo | % CPU | Funzione       | Stato                 |
| ---                                                                      | ALAMAGIC     | QSYS      | SBS  | 0,0   |                | DEQW                  |
| ---                                                                      | C_DIFFER     | ALAMAGIC  | BCH  | 0,0   | PGM-NS0500CL   | DEQA                  |
| ---                                                                      | C_IMMED      | ALAMAGIC  | BCH  | 0,0   | PGM-NS0500CL   | DEQA                  |
| ---                                                                      | C_TEST_DIF   | ALAMAGIC  | BCH  | 0,0   | PGM-NS0500CL   | DEQA                  |
| ---                                                                      | C_TEST_IMM   | ALAMAGIC  | BCH  | 0,0   | PGM-NS0500CL   | DEQA                  |
| ---                                                                      | ARCAD        | QSYS      | SBS  | 0,0   |                | DEQW                  |
| ---                                                                      | ARCAD        | ARCAD_NET | ASJ  | 0,0   | COM-AMSGCMDE   | MSGW                  |
| ---                                                                      | WSSCMD       | ARCAD     | ASJ  | 0,0   | PGM-WSSDTAQMNG | DEQW                  |
| ---                                                                      | ASPSBS       | QSYS      | SBS  | 0,0   |                | DEQW                  |
| Parametri o comando                                                      |              |           |      |       |                | Segue...              |
| ====>                                                                    |              |           |      |       |                |                       |
| F3=Fine    F5=Rivisual.    F10=Riavvio statistiche                       |              |           |      |       |                |                       |
| F11=Visualizzazione dati scaduti    F12=Annull.    F24=Altri tasti       |              |           |      |       |                |                       |

Se si vuole sapere quale lavoro assorbe la maggior percentuale di CPU, si porta il cursore sulla colonna "% CPU" e si preme F16. La videata viene aggiornata e ordinata sul valore della colonna scelta, in questo caso in ordine discendente.

```

Gestione lavori attivi
S65D69DA
24/11/20 22:34:23 CET
% CPU: 19,6 Tempo trascorso: 00:02:52 Lavori attivi: 600

Immettere opzioni, premere Invio.
2=Modif. 3=Congelamento 4=Fine 5=Gest. 6=Ril.
7=Visualiz. messaggio 8=Gestione file di spool 13=Scollegamento...

Opz Sottosis/Lav Utente corrente Tipo % CPU Funzione Stato
CEDAUTLST QSECOFR BCH 12,6 PGM-CEDAUTLST RUN
BSFFILUSE ENFORCE BCH 2,3 PGM-LIBUSEFL RUN
JSRCLSTLU UFBA BCH 1,7 COM-JSRCLSTLU RUN
BSFQHSTNEW BSAFE BCH 0,2 DLY-3 DLYW
XEROX QSPLJOB WTR 0,0 EVTW
WSSCMD ARCAD ASJ 0,0 PGM-WSSDTAQMNG DEQW
WEBGOV QTMHHTTP BCI 0,0 PGM-QZSRHTTP DEQW
WEBGOV QTMHHTTP BCI 0,0 PGM-QZSRHTTP SIGW
WEBGOV QTMHHTTP BCH 0,0 PGM-QZHBMAIN SIGW

Segue...

Parametri o comando
====>
F3=Fine F5=Rivisual. F10=Riavvio statistiche
F11=Visualizzazione dati scaduti F12=Annull. F24=Altri tasti

```

Se sulla videata di ingresso si batte F11, i dati esposti sulla riga cambiano.

```

Gestione lavori attivi
S65D69DA
24/11/20 22:47:33 CET
% CPU: 29,2 Tempo trascorso: 00:16:02 Lavori attivi: 599

Immettere opzioni, premere Invio.
2=Modif. 3=Congelamento 4=Fine 5=Gest. 6=Ril.
7=Visualiz. messaggio 8=Gestione file di spool 13=Scollegamento...

Opz Sottosis/Lav Tipo Lot. Ptà CPU Int Rsp IOaus % CPU
ALAMAGIC SBS 2 0 0,0 0 0 0 0,0
C_DIFFER BCH 2 50 8,8 2 0 0 0,0
C_IMMED BCH 2 50 3,3 3 0 0 0,0
C_TEST_DIF BCH 2 50 2,4 6 0 0 0,0
C_TEST_IMM BCH 2 50 2,9 201 0 0 0,0
ARCAD SBS 2 0 0,0 0 0 0 0,0
ARCAD ASJ 2 60 0,0 0 0 0 0,0
WSSCMD ASJ 2 60 0,0 0 0 0 0,0
ASPSBS SBS 2 0 0,0 0 0 0 0,0

Segue...

Parametri o comando
====>
F3=Fine F5=Rivisual. F7=Ricerca F10=Riavvio statistiche
F11=Vis. dati sottopr. F12=Annull. F23=Altre opz. F24=Altri tasti

```

Se si porta il cursore sulla colonna "Int" e si batte F16 si ottiene l'ordinamento discendente sul numero di interazioni tra l'utente e il video.

```

Gestione lavori attivi
S65D69DA
24/11/20 22:49:43 CET
% CPU: 31,1 Tempo trascorso: 00:18:12 Lavori attivi: 600

Immettere opzioni, premere Invio.
2=Modif. 3=Congelamento 4=Fine 5=Gest. 6=Ril.
7=Visualiz. messaggio 8=Gestione file di spool 13=Scollegamento...

Opz Sottosis/Lav Tipo Lot. Ptà CPU Int Rsp IOaus % CPU
VALERIOH1 INT 4 20 2,5 400 0,0 3878 0,0
H1NERO INT 4 20 0,2 41 0,0 127 0,0
SENIO01 INT 4 20 0,0 0 0,0 0 0,0
H2ROMANO INT 4 20 0,1 0 0,0 0 0,0
HICOLOMBO INT 4 20 2,2 0 0,0 0 0,0
H1AGRON INT 4 20 13,1 0 0,0 0 0,0
H1 INT 4 20 0,6 0 0,0 0 0,0
FAZIO02 INT 4 20 0,0 0 0,0 0 0,0
DACCIO2 INT 4 20 0,1 0 0,0 0 0,0

Segue...

Parametri o comando
====>
F3=Fine F5=Rivisual. F7=Ricerca F10=Riavvio statistiche
F11=Vis. dati sottopr. F12=Annull. F23=Altre opz. F24=Altri tasti

```

36.4. Per valutare l'uso dei successivi comandi di controllo lavori, si eseguono due sottomissioni al solo scopo di mostrarne i job che nascono nelle videate generate dai comandi successivi.

> SBMJOB CMD (DSPJOB) JOB (PRIMO) HOLD (\*YES)

Il lavoro 546258/UFNERONI/PRIMO è stato sottomesso alla coda lavori QBATCH nella libreria QGPL.

> SBMJOB CMD (JPAUSE MSGQ (H1NERO)) JOB (SECONDO)

Il lavoro 546259/UFNERONI/SECONDO è stato sottomesso alla coda lavori QBATCH nella libreria QGPL.

36.5. WRKSBMJOB (Work with Submitted Jobs) elenca i lavori sottomessi dall'utente corrente.

36.5.1. Il comando **WRKSBMJOB** (Work with Submitted Jobs), documentato al link [https://www.ibm.com/support/knowledgecenter/ssw\\_ibm\\_i\\_73/cl/wrksbmjob.htm](https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_73/cl/wrksbmjob.htm) ,

chiamato senza parametri, presenta una videata con l'elenco dei lavori sottomessi dall'utente corrente, un lavoro per riga e per ogni lavoro alcuni parametri significativi del suo stato. La visualizzazione può essere rinfrescata con F5.

Le opzioni permettono di interagire con i lavori. Ad esempio:

Opzione 6 + Invio su PRIMO rilascia in esecuzione il lavoro nato congelato.

```

Gestione lavori inoltrati
S65D690A
27/11/20 11:27:33 CET

Inoltrato da : *USER

Immettere le opzioni e premere Invio.
2=Modifica 3=Congelamento 4=Fine 5=Gest. 6=Ril.
7=Visualiz. messaggio 8=Gestione file di spool

Opz Lavoro Utente Tipo -----Stato----- Funzione
PRIMO UFNERONI BATCH JOBQ HELD
SECONDO UFNERONI BATCH MSGW COM-JPAUSE

Parametri o comando
==>
F3=Fine F4=Richiesta F5=Rivis. F9=Duplic.
F11=Visualizzazione dati di pianificazione F12=Annull F24=Altri tasti

```

36.7. Il comando **WRKUSRJOB** (Work with User Jobs), documentato al link [https://www.ibm.com/support/knowledgecenter/ssw\\_ibm\\_i\\_73/cl/wrkusrjob.htm](https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_73/cl/wrkusrjob.htm) , chiamato senza parametri, presenta una videata con l'elenco dei lavori dell'utente corrente, un lavoro per riga e per ogni lavoro alcuni parametri significativi del suo stato. La visualizzazione può essere rinfrescata con F5.

```

Gestione lavori utente S65D690A
 27/11/20 12:20:04 CET

Immettere le opzioni e premere Invio.
 2=Modifica 3=Congelamento 4=Fine 5=Gest. 6=Ril.
 7=Visualiz. messaggio 8=Gestione file di spool 13=Scollegamento

Opz Lavoro Utente Tipo -----Stato----- Funzione

 HINERO UFNERONI INTER ACTIVE COM-WRKUSRJOB
 PRIMO UFNERONI BATCH JOBQ HELD
 SECONDO UFNERONI BATCH MSGW
 COM-JPAUSE

Parametri o comando
====>
F3=Fine F4=Richiesta F5=Rivis. F9=Duplic.
F11=Visualizzazione dati di pianificazione F12=Annull F24=Altri tasti

```

```

Gestione lavori utente
S65D69DA
27/11/20 12:30:01 CET
Immettere le opzioni e premere Invio.
2=Modifica 3=Congelamento 4=Fine 5=Gest. 6=Ril.
7=Visualiz. messaggio 8=Gestione file di spool 13=collegamento

Opz Lavoro Utente Tipo -----Stato----- Funzione
UFCOPPI01 UFMGOPPI INTER ACTIVE PGM-ASPMAN
UFCOPPI02 UFMGOPPI INTER SYSREQ PGM-QMNSYSRQ
UFCOPPI02 UFMGOPPI INTER ACTIVE COM-STRSQL

Parametri o comando
==>
F3=Fine F4=Richiesta F5=Rivis. F9=Duplic.
F11=Visualizzazione dati di pianificazione F12=Annull F24=Altri tasti

```

07/02/2023 18:51:00

- 36.9. Diverso dai precedenti, il comando **WRKSYSSTS** (Work with System Status), documentato al link [https://www.ibm.com/support/knowledgecenter/ssw\\_ibm\\_i\\_73/cl/wrksyssts.htm](https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_73/cl/wrksyssts.htm), chiamato senza parametri, presenta una videata con l'elenco dei lotti di memoria attivi sul sistema, un lotto per riga e per ogni lotto alcuni parametri significativi del funzionamento. Il tempo dell'esame può essere allungato con F5 o fatto ripartire con F10.

```

Gestione stato sistema
S65D69DA
24/11/20 23:50:26 CET
% CPU utilizzata : 11,4 Memoria ausiliaria:
Tempo trascorso : 00:00:01 Sistema ASP : 31534 G
Lavori nel sistema . . . : 122436 % sistema ASP utiliz . : 94,2649
% indirizzi permanenti . : 1,064 Totale : 31534 G
% indirizzi temporanei . : 34,047 Tempor. corrente
 utilizzata : 23980 M
 Tempor. max utilizzata : 28947 M

Immettere modifiche (se consentito), premere Invio.

Lotto Dim (M) Dim (M) Attivo -----DB----- -----Non-DB-----
sist lotto riservata max Errori Pag Errori Pag
1 9677,82 5037,56 +++++ 0,0 0,0 0,0 0,0
2 16536,58 45,46 1427 0,0 0,0 1,8 1,8
3 3276,79 0,00 76 0,0 0,0 0,0 0,0
4 3276,79 0,00 820 0,0 0,0 0,9 1,8

Comando
==>
F3=Fine F4=Richiesta F5=Rivisualiz. F9=Duplicaz. F10=Riavvio
F12=Ann. F19=Stato sistema esteso F24=Altri tasti

```

Premendo F11, si commuta la visualizzazione di tutte le informazioni sui lotti.

| Lotto | Dim (M)  | Dim (M)   | Attivo | Attivo-> | Attesa-> | Attivo-> |
|-------|----------|-----------|--------|----------|----------|----------|
| sist  | lotto    | riservata | max    | Attesa   | Inel     | Inel     |
| 1     | 9677,82  | 5037,56   | +++++  | 55,4     | 0,0      | 0,0      |
| 2     | 16536,58 | 45,46     | 1427   | 3767,3   | 0,0      | 0,0      |
| 3     | 3276,79  | 0,00      | 76     | 0,0      | 0,0      | 0,0      |
| 4     | 3276,79  | 0,00      | 820    | 55,4     | 0,0      | 0,0      |

| Lotto | Dim (M)  | Dim (M)   | Attivo | Lot.      | Sottosis. | Libreria |
|-------|----------|-----------|--------|-----------|-----------|----------|
| sist  | lotto    | riservata | max    |           |           |          |
| 1     | 9677,82  | 5037,56   | +++++  | *MACHINE  |           |          |
| 2     | 16536,58 | 45,46     | 1427   | *BASE     |           |          |
| 3     | 3276,79  | 0,00      | 76     | *SPOOL    |           |          |
| 4     | 3276,79  | 0,00      | 820    | *INTERACT |           |          |

| Lotto | Dim (M)  | Dim (M)   | Attivo | Opz    |
|-------|----------|-----------|--------|--------|
| sist  | lotto    | riservata | max    | paging |
| 1     | 9677,82  | 5037,56   | +++++  | *FIXED |
| 2     | 16536,58 | 45,46     | 1427   | *CALC  |
| 3     | 3276,79  | 0,00      | 76     | *CALC  |
| 4     | 3276,79  | 0,00      | 820    | *FIXED |



## 37. Integrated file system (IFS)

- 37.1. Sul sistema esistono due modalità di indirizzamento degli oggetti.
- 37.2. Modalità "QSYS.LIB".
- 37.2.1. La prima è quella nativa per librerie e per oggetti, detta "QSYS.LIB", vista fino ad ora.
  - 37.2.2. Utilizza solo una parte del contenuto dei dischi.
  - 37.2.3. Accede agli oggetti in maniera sofisticata e complessa.
  - 37.2.4. Gli oggetti contenuti hanno tutti una tipologia e incorporano la propria definizione funzionale.
  - 37.2.5. Permette l'utilizzo del database nativo e dell'SQL.
  - 37.2.6. Permette le elaborazioni basate su linguaggi di programmazione con accesso diretto al database.
- 37.3. Modalità "IFS".
- 37.3.1. La seconda è nata molto più tardi per compatibilità col mondo elaborativo PC e UNIX, detta "IFS".
  - 37.3.2. Accede agli oggetti trattandoli come semplici file senza sfruttarne le caratteristiche native.
  - 37.3.3. Vede però tutto il contenuto dei dischi.
  - 37.3.4. Per alcune attività permette di indirizzare anche gli oggetti appartenenti a QSYS.LIB.
  - 37.3.5. Lo spazio IFS è organizzato come quello di un PC: radice, indirizzari e sottoindirizzari a più livelli.
  - 37.3.6. I nomi dei file IFS sono composti da nome propriamente detto, punto, estensione. Se manca l'estensione, manca anche il punto separatore.
  - 37.3.7. L'univocità dei nomi in un indirizzario NON tiene conto di maiuscole/minuscole.
  - 37.3.8. L'ordinamento della visualizzazione dei nomi è sensibile invece a maiuscole/minuscole.
- 37.4. Creare un indirizzario in radice con MKDIR.
- 37.4.1. Crea un indirizzario in radice col nome minuscolo poi ritenta la creazione col nome maiuscolo.  
> MKDIR DIR('/corso')  
Indirizzario creato.  
> MKDIR DIR('/CORSO')  
L'oggetto esiste già. L'oggetto è /CORSO.
- 37.5. Visualizzare gli indirizzari IFS con WRKLNK e usare le opzioni sulle righe dell'elenco.
- 37.5.1. L'interfaccia proposta da WRKLNK scimmietta il PDM ma è molto più povera e, in particolare, non dispone di opzioni personalizzabili.
  - 37.5.2. Visualizza il segmento di indirizzario radice contenente i nomi che iniziano con "c" o con "C".

```
Gestione collegamenti oggetto

Indirizzario . . . : /
Immettere le opzioni e premere Invio.
 2=Modifica 3=Copia 4=Rimozione 5=Visualizz. 7=Ridenom.
 8=Attributi vis. 11=Modifica indirizzario corrente ...

Opz. Colleg. oggetto
c:
cedlib
cgidev
cgidev_version1
cgidevexthtml
cgidev2aabcash_mio.TXT
cgi400c2
com
core
Segue...

Parametri o comando
====>
F3=Fine F4=Richiesta F5=Rivisualizzazione F9=Duplicaz. F12=Annull.
F17=Inizio elenco da F22=Visualizzazione campo intero F23=Altre opz.
```

```
Gestione collegamenti oggetto

Indirizzario . . . : /
Immettere le opzioni e premere Invio.
 2=Modifica 3=Copia 4=Rimozione 5=Visualizz. 7=Ridenom.
 8=Attributi vis. 11=Modifica indirizzario corrente ...

Opz. Colleg. oggetto
corso
cumulpower
cumul2016
cum201901
ComandiModificativ07R01M00
CALEND_000.xml
CERTIFICATO
COWEBBDPOBJ
COWEBUFNETF
Segue...

Parametri o comando
====>
F3=Fine F4=Richiesta F5=Rivisualizzazione F9=Duplicaz. F12=Annull.
F17=Inizio elenco da F22=Visualizzazione campo intero F23=Altre opz.
```



```

Gestione collegamenti oggetto

Indirizzario . . . : /

Immettere le opzioni e premere Invio.
 2=Modifica 3=Copia 4=Rimozione 5=Visualizz. 7=Ridenom.
 8=Attributi vis. 11=Modifica indirizzario corrente ...

Opz. Colleg. oggetto
 COWEBUF0BJ

Parametri o comando
==>
F3=Fine F4=Richiesta F5=Rivisualizzazione F9=Duplicaz. F12=Annull.
F17=Inizio elenco da F22=Visualizzazione campo intero F23=Altre opz.

```

37.5.3. L'opzione 5 su un indirizzario permette di accedere alla visualizzazione dei suoi sottoindirizzari.

```

Gestione collegamenti oggetto

Indirizzario . . . : /

Immettere le opzioni e premere Invio.
 2=Modifica 3=Copia 4=Rimozione 5=Visualizz. 7=Ridenom.
 8=Attributi vis. 11=Modifica indirizzario corrente ...

Opz. colleg. oggetto
 core
 corporate.txt
 5 corso
 ctrbkp.txt
 Cambi_ARCOTITF
 Cambi_BDMSTITF
 Capitaliz_BDMSTITF
 Capitaliz_UFBDTITF
 Circolante_BDMSTITF

Parametri o comando
==>
F3=Fine F4=Richiesta F5=Rivisualizzazione F9=Duplicaz. F12=Annull.
F17=Inizio elenco da F22=Visualizzazione campo intero F23=Altre opz.

```

```

Gestione collegamenti oggetto

Indirizzario . . . : /corso

Immettere le opzioni e premere Invio.
 2=Modifica 3=Copia 4=Rimozione 5=Visualizz. 7=Ridenom.
 8=Attributi vis. 11=Modifica indirizzario corrente ...

Opz. Colleg. oggetto
 .
 ..
 JSIGNF.csv

Parametri o comando
==>
F3=Fine F4=Richiesta F5=Rivisualizzazione F9=Duplicaz. F12=Annull.
F17=Inizio elenco da F22=Visualizzazione campo intero F23=Altre opz.

```

37.6. Visualizzare il contenuto di un file IFS con DSPF.

37.6.1. L'opzione 5 su un file IFS di tipo testo ne visualizza il contenuto.

```

Esame : /corso/JSIGNF.csv
Record : 1 di 4 da 18 Colon.: 1 82 da 131
Control.:

.....1.....2.....3.....4.....5.....6.....7.....8.....9.....0
*****Inizio dei dati*****
"FLAG";"SIGN";"DEBIT";"CREDIT"
"Sign flag";"Sign multiplier";"Debit column multiplier";"Credit column multiplier"
"D";-1;1;0
"A";1;0;1
*****Fine dati*****

F3=Fine F10=Visual. esa F12=Annull. F15=Servizi F16=Ricerca rip. F19=Sinis. F20=Destra

```

37.6.2. L'opzione 2 su un file chiama l'editor EDTF sul suo contenuto. Può essere stato impedito per forzare ad usare altri editor. Somiglia molto al SEU. Agisce correttamente su file che contengono normale testo.

```

Mod. file: /corso/JSIGNF.CSV
Record : 1 di 4 da 10 Colon.: 1 82 da 126
Control.:

COM1.....2.....3.....4.....5.....6.....7.....8.....9.....0.....
*****Inizio dei dati*****
"FLAG";"SIGN";"DEBIT";"CREDIT"
"Sign flag";"Sign multiplier";"Debit column multiplier";"Credit column multiplier"
"D";-1;1;0
"A";1;0;1
*****Fine dati*****

F2=Salv. F3=Salv./Fine F12=Fine F15=Servizi F16=Ripet. ric. F17=Ripet. modif. F19=Sin. F20=Destra

```

37.7. Copiare dati fa file di database a file IFS con CPYTOIMPF (Copy To Import File).

37.7.1. Copia il file CNU2 in forma testo nel nuovo indirizzario.

```

> CPYTOIMPF FROMFILE(CNU2) TOSTMF('/corso/butta.txt') RCDDL(*CRLF)
Tutti i record copiati dal file CNU2 in CORSO

```

37.7.2. Visualizza il file generato.

```

> DSPF STMF('/corso/butta.txt')

```

```

Esame : /corso/butta.txt
Record : 1 di 31 da 18 Colon.: 1 64 da 131
Control.:

.....1.....2.....3.....4.....5.....6.....7.....8.....9.....0
*****Inizio dei dati*****
1 ;10 ;PRIMA riga MODIFICATA
1 ;20 ;seconda RIGA
2 ;1 ;Tanto
2 ;2 ;va la
2 ;3 ;gatta
2 ;4 ;al lardo
4 ;1 ;Riga senza testata
5 ;10 ;Nel
5 ;20 ;mezzo
5 ;30 ;del
5 ;40 ;cammin
5 ;50 ;di
5 ;60 ;nostra
5 ;70 ;vita
5 ;80 ;mi
5 ;90 ;ritrovai
5 ;100 ;per
5 ;110 ;una

F3=Fine F10=Visual. esa F12=Annull. F15=Servizi F16=Ricerca rip. F19=Sinis. F20=Destra
Esame : /corso/butta.txt
Record : 18 di 31 da 18 Colon.: 1 64 da 131
Control.:

.....1.....2.....3.....4.....5.....6.....7.....8.....9.....0
5 ;110 ;una
5 ;120 ;selva
5 ;130 ;oscura
5 ;140 ;che
5 ;150 ;la
5 ;160 ;diritta
5 ;170 ;via
5 ;180 ;era
5 ;190 ;smarrita
5 ;200 ;Ahi
5 ;210 ;quanto
5 ;220 ;a
5 ;230 ;dir ...
2 ;5 ;che ci lascia lo zampino.
*****Fine dati*****

F3=Fine F10=Visual. esa F12=Annull. F15=Servizi F16=Ricerca rip. F19=Sinis. F20=Destra

```

37.8. Copiare dati da file IFS a file di database con CPYFRMIMPF (Copy From Import File).

37.8.1. Da scrivere.

37.9. I detti comandi del sistema operativo CPYTOIMPF e CPYFRMIMPF, destinati allo scambio di dati tra file di database e file di IFS, sono purtroppo delicatucci nella parametrizzazione e richiedono ogni volta una faticosa messa a punto.

37.10. In luogo di quelli, piuttosto debolucci, è possibile usare due coppie di tool:

```
JTOTXT (Unload Database to TXT)
JFROMTXT (Load TXT to database)
JTOSCV (Unload database to CSV)
JFROMCSV (Load CSV to database).
```

37.11. La seconda coppia di tool è pensata per risolvere il caso frequente di:

scarico dati su PC,  
correzione su PC tramite Excel,  
ricarico su un altro file con lo stesso tracciato dell'originale.

37.12. JTOSCV (Unload database to CSV)

37.12.1. Copia il file di database CNU2 in un file IFS in formato csv.

```
> JTOSCV FROMFILE(CNU2) TOCSV(CNU2) TODIR('/corso') RMVBLANK(*BOTH)
 Copiato il file di database datore CORSO/CNU2 nel file IFS ricevente
 /corso/CNU2.csv
> /**/DSPF STMF('/corso/CNU2.csv')
```

37.12.2. Visualizza il file IFS generato.

```
> DSPF STMF('/corso/CNU2.csv')
"N2DOC";"N2RIG";"N2TXT"
"Numero documento";"Numero riga";"Testo riga"
1;10;"PRIMA riga MODIFICATA"
1;20;"seconda RIGA"
2;1;"Tanto"
2;2;"va la"
2;3;"gatta"
2;4;"al lardo"
4;1;"Riga senza testata"
5;10;"Nel"
5;20;"mezzo"
5;30;"del"
5;40;"cammin"
5;50;"di"
5;60;"nostra"
5;70;"vita"
5;80;"mi"
5;90;"ritrovai"
5;100;"per"
5;110;"una"
5;120;"selva"
5;130;"oscura"
5;140;"che"
5;150;"la"
5;160;"diritta"
5;170;"via"
5;180;"era"
5;190;"smarrita"
5;200;"Ahi"
5;210;"quanto"
5;220;"a"
5;230;"dir ..."
2;5;"che ci lascia lo zampino."
```

37.13. JFROMCSV (Load CSV to database)

37.13.1. Copia il file IFS in formato csv in un file di database estemporaneo CNU2RST con il tracciato di CNU2.

```
> JFROMCSV PROTOFILE(CNU2) TOFILE(CORSO/CNU2RST) FROMCSV(CNU2) FROMDIR('/corso')
RPLTOFILE(*YES) FROMRCD(3)
 Copiato il file datore IFS /corso/CNU2.csv nel file di database ricevente
 CORSO/CNU2RST, prototipo CORSO/CNU2
> /**/RUNQRY QRYFILE((CORSO/CNU2RST)) RCDSLTL(*YES)
```

37.13.2. Visualizza il file di database ricaricato.

```
> RUNQRY QRYFILE((CORSO/CNU2RST)) RCDSLTL(*YES)
```

| Riga   | .....1.....      | .....2..... | .....3.....           | .....4..... | .....5..... | .....6..... |
|--------|------------------|-------------|-----------------------|-------------|-------------|-------------|
|        | Numero documento | Numero riga | Testo riga            |             |             |             |
| 000001 | 1                | 10          | PRIMA riga MODIFICATA |             |             |             |
| 000002 | 1                | 20          | seconda RIGA          |             |             |             |
| 000003 | 2                | 1           | Tanto                 |             |             |             |
| 000004 | 2                | 2           | va la                 |             |             |             |
| 000005 | 2                | 3           | gatta                 |             |             |             |
| 000006 | 2                | 4           | al lardo              |             |             |             |
| 000007 | 4                | 1           | Riga senza testata    |             |             |             |
| 000008 | 5                | 10          | Nel                   |             |             |             |

|        |   |     |                           |
|--------|---|-----|---------------------------|
| 000009 | 5 | 20  | mezzo                     |
| 000010 | 5 | 30  | del                       |
| 000011 | 5 | 40  | cammin                    |
| 000012 | 5 | 50  | di                        |
| 000013 | 5 | 60  | nostra                    |
| 000014 | 5 | 70  | vita                      |
| 000015 | 5 | 80  | mi                        |
| 000016 | 5 | 90  | ritrovai                  |
| 000017 | 5 | 100 | per                       |
| 000018 | 5 | 110 | una                       |
| 000019 | 5 | 120 | selva                     |
| 000020 | 5 | 130 | oscura                    |
| 000021 | 5 | 140 | che                       |
| 000022 | 5 | 150 | la                        |
| 000023 | 5 | 160 | diritta                   |
| 000024 | 5 | 170 | via                       |
| 000025 | 5 | 180 | era                       |
| 000026 | 5 | 190 | smarrita                  |
| 000027 | 5 | 200 | Ahi                       |
| 000028 | 5 | 210 | quanto                    |
| 000029 | 5 | 220 | a                         |
| 000030 | 5 | 230 | dir ...                   |
| 000031 | 2 | 5   | che ci lascia lo zampino. |

#### 37.14. Impostazione dell'indirizzario corrente.

37.14.1. Serve a predisporre un indirizzario di default per i comandi IFS a partire da **WRKLNK**.

37.14.2. Opzione 11 su un indirizzario elencato in una riga dell' interfaccia **WRKLNK**.

37.14.3. Mediante il comando **CHGCURDIR** (Change Current Directory).

## 38. Sorgenti e IFS

- 38.1. In corso di messa a punto due coppie di tool da collaudare tra le due macchine DEV720 e CORSO di nome JSRCSTM () - JSTMSRC () e le varianti JSRXSTM () - JSTMSRX () che conservano numero di sequenza e data di aggiornamento delle righe.
- 38.2. Passaggio di un sorgente da membro di file sorgente nel sistema di indirizzamento QSYS a file di testo nel sistema di indirizzamento IFS.
- 38.2.1. I due comandi qui suggeriti, CPYTOIMPF e CPYFROMIMPF, hanno il difetto di perdere i campi di servizio dei file sorgenti: data aggiornamento e numero di sequenza della riga. Per evitarlo occorrerebbe sostituire i due comandi con altrettanti rpgle e opportuni comandi e ccle di contorno.
- 38.2.2. Il comando di sistema CPYTOIMPF, opportunamente parametrizzato, permette di trascrivere il sorgente scelto dal membro di file fisico source, suo contenitore tradizionale, al file di testo IFS.
- ```
> CPYTOIMPF FROMFILE(CORSO/CORSO CNA1GD) TOSTMF('/home/NERONI/CORSO-CORSO-CNA1GD.txt') MBROPT(*REPLACE) RCDDL(*CRLF) STRDLM(*NONE)
Tutti i record copiati dal file CORSO in CORSO
```
- 38.2.3. Si osservino i due parametri finali che risultano cruciali per l'esito della copia.
- 38.2.4. Il risultato sarà visibile con il comando seguente.
- ```
> DSPF STMF('/home/NERONI/CORSO-CORSO-CNA1GD.txt')
```
- 38.2.5. E modificato con il comando seguente.
- ```
> EDTF STMF('/home/NERONI/CORSO-CORSO-CNA1GD.txt')
```
- 38.3. Passaggio di un sorgente da file di testo nel sistema di indirizzamento IFS a membro di file sorgente nel sistema di indirizzamento QSYS.
- 38.3.1. Il comando di sistema CPYFROMIMPF, opportunamente parametrizzato, permette di trascrivere un sorgente da un file di testo IFS ad un membro di file fisico source.
- 38.3.2. Si suppone di ricaricare, con un nome diverso (RICARICATO) e in un file source diverso, il sorgente scaricato nel paragrafo precedente tramite il CPYTOIMPF.
- ```
> CPYFROMIMPF FROMSTMF('/home/NERONI/CORSO-CORSO-CNA1GD.txt') TOFILE(CORSO/DUE
RICARICATO) MBROPT(*REPLACE) RCDDL(*CRLF) STRDLM(*NONE) RMVBLANK(*NONE)
651 record copiati nel membro RICARICATO.
```

## 39. Ulteriori nozioni di database>NC

- 39.1. File join.
- 39.2. Posizionamento dei logici rispetto ai fisici di base.
- 39.3. Puntamenti logici-fisici dopo ripristino.
- 39.4. Trigger.
- 39.5. Documentazione del database.

## 40. Elenco istruzioni di calcolo in RPGLE>NC

- 40.1. Lettura e scrittura del database.
- 40.2. Lettura e scrittura del video.
- 40.3. Calcolo numerico.
- 40.4. Gestione stringhe.
- 40.5. Date
- 40.6. Built in function.
- 40.7. Subroutine.
- 40.8. Procedure.

## 41. Tecniche di modularizzazione della programmazione su IBMi>NC

### 41.1. Premesse.

- 41.1.1. I concetti qui esposti sono eviscerati per quanto riguarda il linguaggio Rpg Ile ma sono applicabili a tutto l'universo Ile che abbraccia anche altri linguaggi IBMi e, in particolare, il CL (Control language) Ile.
- 41.1.2. Per l'Rpg, Ile è non solo la forma delle istruzioni (fixed o free) ma anche la struttura di assemblaggio delle istruzioni eseguibili, utilizzabile o no più per ragioni di stile che per necessità.

### 41.2. Introduzione agli esempi illustrativi.

- 41.2.1. Gli esempi riportati nel capitolo sono reperibili tra i sorgenti di corredo al corso tramite i corrispondenti link. Tutti gli esempi sono varianti possibili dello stesso procedimento.
- 41.2.2. Gli esempi qui di seguito illustrati si basano su due schemi ben distinti.
- 41.2.3. Il primo basato su un programma creato da CRTBNDRPG e sulle call dinamiche (CALL) che il programma di ingresso fa a programmi della stessa specie.
- 41.2.4. Il secondo basato su un programma creato da CRTPGM che riunisce moduli precedentemente creati da CRTMODRPG e in cui il modulo di ingresso esegue call statiche (CALLB) agli altri moduli assemblati dal CRTPGM.
- 41.2.5. I due schemi possono intersecarsi ed, in particolare è permesso:
  - 41.2.5.1. in un modulo eseguire anche call dinamiche;
  - 41.2.5.2. in un programma o in un modulo eseguire call dinamiche a programmi comunque creati.
- 41.2.6. Gli esempi seguenti riguardano solo i due schemi elementari fondamentali e non le loro intersezioni.

### 41.3. Programma CRTBNDRPG solo con main routine.

- 41.3.1. Un programma eseguibile deve obbligatoriamente essere dotato di una routine principale.
- 41.3.2. La main routine può eseguire direttamente anche tutte le istruzioni che realizzano l'elaborazione richiesta.
- 41.3.3. Come esempio, vedi i tre seguenti sorgenti di corredo al corso.
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXA..txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXA..txt)
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXA1.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXA1.txt)
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXA2.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXA2.txt)

### 41.4. Programma CRTBNDRPG con subroutine.

- 41.4.1. Più frequentemente, però, la main routine chiama altre istruzioni, presenti in sezioni del programma dette subroutine, tramite l'istruzione EXSR.
- 41.4.2. Le subroutine contengono istruzioni, confinate tra BEGSR ed ENDSR, che possono essere chiamate in esecuzione da punti diversi del programma stesso evitando di ripetere più volte la minutazione delle istruzioni stesse.
- 41.4.3. Una subroutine condivide con la main routine e con le altre subroutine tutte le variabili contenute nel programma.
- 41.4.4. Una subroutine ben fatta dichiara nei propri commenti la variabili ricevute e le variabili restituite dalla subroutine stessa. Assoggettarsi a questa regola permette di non schiantarsi contro variabili guastate da una manipolazione nascosta eseguita dalla subroutine chiamata.
- 41.4.5. Come esempio, vedi i tre seguenti sorgenti di corredo al corso.
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXB..txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXB..txt)
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXB1.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXB1.txt)
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXB2.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXB2.txt)

### 41.5. Programma CRTBNDRPG con procedure interne.

- 41.5.1. La procedura interna è la forma più semplice di procedura e permette di sostituire una subroutine.
- 41.5.2. La procedura viene minutata in fondo al programma e può contenere la definizione di variabili locali che non sono leggibili dal resto del programma.
- 41.5.3. La procedura può tuttavia leggere ed anche modificare le variabili generali.
- 41.5.4. Nella procedura viene definita anche la modalità di chiamata che restituisce un solo parametro di ritorno e permette di usare la procedura come se fosse una funzione:  
`eval ValoreDiRitorno=Procedura (Parametro1:Parametro2)`
- 41.5.5. Come esempio, vedi i tre seguenti sorgenti di corredo al corso.
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXC..txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXC..txt)
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXC1.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXC1.txt)
  - [www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXC2.txt](http://www.neroni.it/CORSO2023/A-CORSO.CORSO/CNXC2.txt)

### 41.6. Programma CRTBNDRPG con call dinamiche.



- 41.6.1. Un programma può eseguire la chiamata diretta di un altro programma eseguibile reperendolo sul sistema tramite il comando `CALL` seguito da nessuna, una o più `PARM`. La `CALL` vale solo in fixed form. L'equivalente `free`, illustrata più avanti, prevede l'intermediazione di una "procedure" con uso di "external program".
- 41.6.2. Una call ben fatta dichiara nei propri commenti le variabili ricevute e le variabili restituite dalla call stessa. I commenti vanno ripetuti sulla `*ENTRY PLIST` del chiamato. Stessi commenti se si usano le procedure, sia nel chiamante che nel chiamato.
- 41.6.3. La chiamata si chiama dinamica perché viene chiamato il programma eseguibile attualmente presente sul sistema. Il chiamante cerca col nome ma poi memorizza un puntatore al programma trovato e chiamato.
- 41.6.4. La seconda chiamata e le successive allo stesso programma senza che il chiamante venga mandato a fine, tuttavia, non utilizzano più il nome ma il puntatore. Perciò, se tra la fine della prima chiamata il programma viene rinominato e/o spostato di libreria e quindi conserva il proprio indirizzo fisico sul sistema, il programma chiamato sarà sempre quello della prima chiamata.
- 41.6.5. Il programma chiamante condivide con il programma chiamato solo la stessa area di memoria dei parametri esplicitamente passati dal chiamante e ricevuti dal chiamato. Si noti che i parametri del chiamante devono essere dichiarati dal chiamato con le stesse caratteristiche con cui li dichiara il chiamante perché l'unica reale condivisione di informazioni tra i due è l'indirizzo di inizio in memoria delle variabili stesse. Dichiararli di misura diversa autorizza i programmi a pasticciare fuori dalle variabili nel programma controparte provocando errori imprevedibili e quindi difficili da capire. La regola migliore in caso di errori inaspettati e comportamenti strani è: verificare le dimensioni dei parametri tra chiamanti e chiamati in tutto il processo che manifesta il problema.
- 41.7. Moduli.
  - 41.7.1. Oltre allo schema Un programma eseguibile deve obbligatoriamente essere dotato di una routine principale.
- 41.8. Binding directory
- 41.9. Create program.

## 42. SQL>NC

42.1. Interfaccia **STRSQL** (Start SQL).

42.2. Interfaccia **RUNSQLSTM** (Run SQL Statements).

## 43. SQL in RPGLE>NC

### 43.1. Utilizzo di istruzioni SQL in RPGLE.

## 44. OPNQRYF (Open Query File)>NC

44.1. Ordinamento estemporaneo dei dati in CLLE.

## 45. RPG Free

- 45.1. L'ultima versione del linguaggio RPG non è più posizionale ma usa istruzioni composte da frasi che iniziano con un codice istruzione.
- 45.2. Formato libero dell'RPGL delle specifiche di calcolo.
- 45.2.1. Il codice istruzione determina caso per caso la sintassi della riga stessa.
- 45.2.2. Il codice istruzione può essere corredato da opzioni tra parentesi.
- 45.2.3. Il codice istruzione può essere seguito da parametri sotto forma di nomi di campi, file o record.
- 45.2.4. I parametri possono essere multipli e in tal caso chiusi tra parentesi e separati da due punti.
- 45.2.5. La riga termina sempre con il punto e virgola.
- 45.2.6. L'istruzione può continuare su più righe fisiche senza simboli di "a capo".
- 45.2.7. Ad esempio:
- ```
chain chiave record;
chain(n) (chiave1:chiave2) record;
chain(n)
    (chiave1:chiave2)
    record;
```
- 45.2.8. I parametri possono essere anche stringhe alfanumeriche tra apici, o numeriche, o parole riservate che iniziano con "*".
- ```
setll *start file;
```
- 45.3. Formato libero dell'RPGL delle specifiche di definizioni dei file e dei dati.
- 45.3.1. Regole
- 45.4. Seguono esempi di free di qualcuno degli schemi di programma già visti in fixed form. Sono perfettamente equivalenti agli originali fixed form.
- 45.5. Conversione del programma RPGL fixed form CNA1GD di Gestione anagrafica completa in RPGL free con il nome CNA1GDF.

```
*FREE //Claudio Neroni 03-04-2021 Creato per Corso.
// CNA1GDF
// Anagrafica persone dsp-chg-dlt-add-sfl-rfr FREE

//Anagrafica persone logico codice
dcl-f CNA11 keyed rename(A1:A11) infds(FIDS1);
//Anagrafica persone logico descrizione
dcl-f CNA12 keyed rename(A1:A12) infds(FIDS2);
//Tabella provincie
dcl-f CNT11 keyed;
//Anagrafica persone dsp-chg-dlt-add-sfl-rfr
//con Display Specific Feedback Information
dcl-f CNA1GW workstn infds(DSPFBK);
//Anagrafica persone fisico
dcl-f CNA1 usage(*update:*output:*delete) rename(A1:UPD) usroprn;

//Struttura dati del file dati.
dcl-ds A1D extname('CNA1':'A1') end-ds;
//Salvataggio della Struttura dati del file dati.
dcl-ds A1DSV extname('CNA1':'A1') prefix(SV) end-ds;

//Struttura dati del video dati.
dcl-ds W1D extname('CNA1GW':'W1') end-ds;
//Salvataggio della Struttura dati del video dati.
dcl-ds W1DSV extname('CNA1GW':'W1') prefix(SV) end-ds;

//File information data structure
dcl-ds FIDS1;
 //Relative record number in data member
 FIDS1RR bindec(9:0) pos(397);
end-ds;
//File information data structure
dcl-ds FIDS2;
 //Relative record number in data member
 FIDS2RR bindec(9:0) pos(397);
end-ds;

//Costanti
dcl-c NASCITA const('Provincia di Nascita');
dcl-c RESIDENZA const('Provincia di Residenza');

//Display Specific Feedback Information
/copy CORSO,DSPFBKCL
//Rename indicators
/copy CORSO,CNA1GDI

// Dichiaro il prototipo della procedura Finestra provincia,
```

```

dcl-pr CNT1WIN extpgm('CNT1WIN');
 CDPR char(2) ; // U Codice provincia
 INTE char(50) const; // I Intestazione
end-pr;
dcl-s CDPR char(2);

//-----
//Predispone chiusura.
*inlr=*on;
//Assume di usare video guida per codice.
CurByCod=*on;
CurByDes=*off;
//Predispone protezione campi.
CurDsp=*on;
//Gestisce video guida.
exsr SRW0;

//-----
//Gestisce video guida.
begsr SRW0;

//Pulisce video guida.
clear W01;
clear W02;

//Cicla intorno al video guida.
dow 1=1;

 //Annota non aggiunta.
 CurAdd=*off;
 //Prepara modo.
 if CurDsp;
 WWMOD='DSP';
 else;
 WWMOD='CHG';
 endif;

 //Emette video guida.
 if CurByCod;
 exfmt W01;
 endif;
 if CurByDes;
 exfmt W02;
 endif;

 //Se richiesta fine, abbandona.
 if ReqEnd;
 leave;
 endif;

 //Se richiesto switch tra visualizzazione e modifica, esegue e ricicla.
 if ReqSwiMod;
 if CurDsp;
 CurDsp=*off;
 else;
 CurDsp=*on ;
 endif;
 iter;
 endif;

 //Se richiesta aggiunta.
 if ReqAdd;
 //Esegue aggiunta.
 exsr SRW1A;
 //Se richiesta fine, abbandona.
 if ReqEnd;
 leave;
 endif;
 //Ricicla.
 iter;
 //Se richiesta aggiunta.
 endif;

 //Se richiesto switch tra guida per codice e guida per descrizione,
 //commuta la scelta del video guida e ricicla.
 if ReqSwiGui;
 if CurByCod;
 CurByCod=*off;
 CurByDes=*on;
 else;
 CurByCod=*on;
 CurByDes=*off;
 endif;
 iter;

```

```

endif;

//Se richiesto PagUp.
if ReqPagUp;

 //Se corre elaborazione per codice.
 if CurByCod;
 //Si posiziona prima della chiave di ricerca.
 setll (W0CDPE) A11;
 //Legge all'indietro.
 readp A11;
 //Se non trova record, ritenta a partire dal fondo.
 if %eof;
 setgt (*hival) A11;
 readp A11;
 endif;
 //Se corre elaborazione per codice.
 endif;

 //Se corre elaborazione per descrizione;
 if CurByDes;
 //Si posiziona prima della chiave di ricerca.
 setll (W0DEPE:W0CDPE) A12;
 //Legge all'indietro.
 readp A12;
 //Se non trova record, ritenta a partire dal fondo.
 if %eof;
 setgt (*hival) A12;
 readp A12;
 endif;
 //Se corre elaborazione per descrizione;
 endif;

//Se richiesto PagDw o Enter.
else;

 //Se corre elaborazione per codice.
 if CurByCod;
 //Si posiziona prima della chiave di ricerca.
 setll (W0CDPE) A11;
 //Legge in avanti.
 read A11;
 //Se corre elaborazione per codice.
 endif;

 //Se corre elaborazione per descrizione;
 if CurByDes;
 //Si posiziona prima della chiave di ricerca.
 setll (W0DEPE:W0CDPE) A12;
 //Legge in avanti.
 read A12;
 //Se corre elaborazione per descrizione;
 endif;

//Se richiesto PagDw o Enter.
endif;

//Se non trova record, ricicla.
if %eof;
 iter;
endif;

//Gestisce video dati.
exsr SRW1;

//Se richiesta fine, abbandona.
if ReqEnd;
 leave;
endif;

//Cicla intorno al video guida.
enddo;

endsr;

//-----
//Gestisce video dati.
//Va chiamata con un record gia' in canna.
begsr SRW1;

//Cicla intorno al video dati.
dow 1=1;

//Se non corre errore.

```

```

if not ErrGen;
 //Trascrive dati dal record di database al video.
 exsr DBW1;
 //Decodifica video dati.
 exsr DEW1;
//Se non corre errore.
endif;

//Se corre errore, scrive base per errore.
if ErrGen;
 ErrGen=*off;
 write W1;
 ErrGen=*on;
endif;

//Prepara modo.
if CurDsp;
 WWMOD='DSP';
else;
 WWMOD='CHG';
endif;

//Emette video dati.
exfmt W1;
//Elimina posizionamento cursore.
exsr CURZ;
//Spegne tutti gli errori.
exsr OFF;

//Se richiesta fine, abbandona.
if ReqEnd;
 leave;
endif;

//Se richiesto video precedente, abbandona.
if ReqPrv;
 leave;
endif;

//Se richiesto switch tra visualizzazione e modifica, esegue e ricicla.
if ReqSwiMod;
 if CurDsp;
 CurDsp=*off;
 else;
 CurDsp=*on ;
 endif;
 iter;
endif;

//Se richiesto rinfrescamento per chiave.
if ReqRfrKey;

 //Se corre elaborazione per codice.
 if CurByCod;
 //Rilegge l'ultimo record messo a video.
 chain (W0CDPE) A11;
 //Se corre elaborazione per codice.
 endif;

 //Se corre elaborazione per descrizione;
 if CurByDes;
 //Rilegge l'ultimo record messo a video.
 chain (W0DEPE:W0CDPE) A12;
 //Se corre elaborazione per descrizione;
 endif;

 //Se il record non esiste, accende
 //Errore rinfrescamento per chiave.
 if not %found;
 ErrGen =*on;
 ErrRfrKey=*on;
 endif;

 //Ricicla.
 iter;

//Se richiesto rinfrescamento per chiave.
endif;

//Se richiesto rinfrescamento per numero relativo di record.
if ReqRfrRrn;

 //Apri database.
 exsr OPDB;

```



```

//Se errore in apertura, ricicla.
if ErrGen;
 iter;
endif;

//Se corre elaborazione per codice.
if CurByCod;
 //Rilegge per sola lettura l'ultimo record messo a video.
 chain(n) FIDS1RR UPD;
//Se corre elaborazione per codice.
endif;

//Se corre elaborazione per descrizione;
if CurByDes;
 //Rilegge per sola lettura l'ultimo record messo a video.
 chain(n) FIDS2RR UPD;
//Se corre elaborazione per descrizione;
endif;

//Se il record non esiste, accende
//Errore rinfrescamento per numero relativo di record
if not %found;
 ErrGen =*on;
 ErrRfrRrn=*on;
endif;

//Ricicla.
iter;
//Se richiesto rinfrescamento per numero relativo di record.
endif;

//Se richiesto elenco.
if ReqLst;
 //Trascrive cursore per la riemissione.
 exsr CURT;
 //Chiama elenco su provincia nascita.
 if FLD='W1PRNA';
 CDPR=W1PRNA;
 CNT1WIN(CDPR:NASCITA);
 if CDPR<>*blank;
 W1PRNA=CDPR;
 endif;
 endif;
 //Chiama elenco su provincia residenza.
 if FLD='W1PRRE';
 CDPR=W1PRRE;
 CNT1WIN(CDPR:RESIDENZA);
 if CDPR<>*blank;
 W1PRRE=CDPR;
 endif;
 endif;
 //Decodifica video dati.
 exsr DEW1;
////////Controlla video dati.
////////exsr CTW1;
 //Finge errore per non ricaricare da database.
 ErrGen=*on;
 //Ricicla.
 iter;
//Se richiesto elenco.
endif;

//Se richiesta decodifica.
if ReqDec;
 //Decodifica video dati.
 exsr DEW1;
 //Controlla video dati.
 exsr CTW1;
 //Finge errore per non ricaricare da database.
 ErrGen=*on;
 //Ricicla.
 iter;
//Se richiesta decodifica.
endif;

//Se Richiesta cancellazione record.
if ReqDel;
 //Esegue cancellazione.
 exsr DLDB;
 //Se errore, ricicla.
 if ErrGen;
 iter;
 endif;
endif;

```

```

//Se Richiesta cancellazione record.
endif;

//Se corre modifica.
if not CurDsp;
 //Decodifica video dati.
 exsr DEW1;
 //Controlla video dati.
 exsr CTW1;
 //Se errore, ricicla.
 if ErrGen;
 iter;
 endif;
 //Se il video e' stato modificato.
 if W1D<>W1DSV;
 //Esegue update.
 exsr UPDB;
 //Se errore, ricicla.
 if ErrGen;
 iter;
 endif;
 //Se il video e' stato modificato.
 endif;
//Se corre modifica.
endif;

//Se Richiesta pagina su.
if ReqPagUp;

 //Se corre elaborazione per codice.
 if CurByCod;
 //Si posiziona prima della chiave ultima vista.
 setll (W0CDPE) A11;
 //Legge all'indietro.
 readp A11;
 //Se corre elaborazione per codice.
 endif;

 //Se corre elaborazione per descrizione;
 if CurByDes;
 //Si posiziona prima della chiave ultima vista.
 setll (W0DEPE:W0CDPE) A12;
 //Legge all'indietro.
 readp A12;
 //Se corre elaborazione per descrizione;
 endif;

//Se Richiesta pagina giu' o Invio.
else;

 //Se corre elaborazione per codice.
 if CurByCod;
 //Si posiziona oltre la chiave ultima vista.
 setgt (W0CDPE) A11;
 //Legge in avanti.
 read A11;
 //Se corre elaborazione per codice.
 endif;

 //Se corre elaborazione per descrizione;
 if CurByDes;
 //Si posiziona oltre la chiave ultima vista.
 setgt (W0DEPE:W0CDPE) A12;
 //Legge in avanti.
 read A12;
 //Se corre elaborazione per descrizione;
 endif;

//Se Richiesta pagina giu' o Invio.
endif;

//Se non trova record, pulisce chiave accantonata e abbandona.
if %eof;
 clear W0CDPE;
 clear W0DEPE;
 leave;
endif;

//Cicla intorno al video dati.
enddo;

endsr;

//-----

```

```

//Trascrive dati dal record di database al video.
begsr DBW1;

//Accantona la chiave nel video guida.
exsr DBW1KK;
//Trascrive i campi dal record di database al video dati.
clear W1;
W1CDPE=A1CDPE;
W1DEPE=A1DEPE;
W1PRNA=A1PRNA;
W1PRRE=A1PRRE;
//Salva i dati del record letto.
A1DSV=A1D;
//Salva i dati del video.
W1DSV=W1D;

endsr;

//-----
//Decodifica video dati.
begsr DEW1;

//Decodifica provincia di nascita.
clear W1PRNAD;
if W1PRNA<>*blank;
 W1PRNAD=*all'?' ;
 chain (W1PRNA) T1;
 if %found;
 W1PRNAD=T1DEPR;
 endif;
endif;

//Decodifica provincia di residenza.
clear W1PRRED;
if W1PRRE<>*blank;
 W1PRRED=*all'?' ;
 chain (W1PRRE) T1;
 if %found;
 W1PRRED=T1DEPR;
 endif;
endif;

endsr;

//-----
//Spegne tutti gli errori.
begsr OFF;

eval ErrAll=*off;

endsr;

//-----
//Controlla video dati.
begsr CTW1;

//Spegne tutti gli errori.
exsr OFF;

//Esegue.
dou 1=1;

 //Codice obbligatorio.
 if W1CDPE=*blank;
 ErrGen =*on;
 ErrCodBla=*on;
 leave;
 endif;

 //Descrizione obbligatoria.
 if W1DEPE=*blank;
 ErrGen =*on;
 ErrDesBla=*on;
 leave;
 endif;

 //Descrizione piu' corta del codice.
 if %len(%trim(W1DEPE))<=%len(%trim(W1CDPE));
 ErrGen =*on;
 ErrDesShr=*on;
 leave;
 endif;

 //Provincia di nascita obbligatoria.

```

```

 if W1PRNA=*blank;
 ErrGen =*on;
 ErrNasBla=*on;
 leave;
 endif;

 //Provincia di nascita manca in tabella.
 if W1PRNAD=*all'?';
 ErrGen =*on;
 ErrNasTab=*on;
 leave;
 endif;

 //Provincia di residenza manca in tabella.
 if W1PRRED=*all'?';
 ErrGen =*on;
 ErrResTab=*on;
 leave;
 endif;

 //Esegue.
enddo;

endsr;

//-----
//Trascrive dati dal video al record di database
begsr W1DB;

A1CDPE=W1CDPE;
A1DEPE=W1DEPE;
A1PRNA=W1PRNA;
A1PRRE=W1PRRE;

endsr;

//-----
//Ricalca database.
begsr UPDB;

//Esegue.
dou 1=1;

 //Apri database.
 exsr OPDB;

 //Se errore in apertura, abbandona.
 if ErrGen;
 leave;
 endif;

 //Se corre elaborazione per codice.
 if CurByCod;
 //Recupera il record fisico per ricalcarlo.
 chain(e) FIDS1RR UPD;
 //Se corre elaborazione per codice.
 endif;

 //Se corre elaborazione per descrizione;
 if CurByDes;
 //Recupera il record fisico per ricalcarlo.
 chain(e) FIDS2RR UPD;
 //Se corre elaborazione per descrizione;
 endif;

 //Se il record e' occupato, accende errore e abbandona.
 if %error;
 ErrGen =*on;
 ErrOcc =*on;
 leave;
 endif;

 //Se il record non esiste, accende errore e abbandona.
 if not %found;
 ErrGen =*on;
 ErrExi =*on;
 leave;
 endif;

 //Se i dati del record sono cambiati, accende errore e abbandona.
 if A1D<>A1DSV;
 ErrGen =*on;
 ErrChg =*on;
 leave;

```

```

endif;

//Trascrive dati dal video al record di database
exsr W1DB;
//Ricalca il record.
update(e) UPD;

//Se la chiave del record esiste gia', accende errore e abbandona.
if %error;
 ErrGen =*on;
 ErrKey =*on;
 leave;
endif;

//Esegue.
enddo;

endsr;

//-----
//Gestisce video dati in aggiunta record.
begsr SRW1A;

//Assume campi agibili.
CurDsp=*off;
//Assume aggiunta.
CurAdd=*on;

//Cicla intorno a video dati.
dow 1=1;

 //Se non corre errore.
 if not ErrGen;
 //Pulisce video dati.
 clear W1;
 //Decodifica video dati.
 exsr DEW1;
 //Se non corre errore.
 endif;

 //Se corre errore, scrive base per errore.
 if ErrGen;
 ErrGen=*off;
 write W1;
 ErrGen=*on;
 endif;
 //Prepara modo.
 WWMOD='ADD';
 //Emette video dati.
 exfmt W1;
 //Elimina posizionamento cursore.
 exsr CURZ;
 //Spegne tutti gli errori.
 exsr OFF;

 //Se richiesta fine, abbandona.
 if ReqEnd;
 leave;
 endif;

 //Se richiesta Guida, abbandona.
 if ReqPrv;
 leave;
 endif;

 //Se richiesto rinfrescamento, ricicla.
 if ReqRfrKey;
 leave;
 endif;

 //Se corre elenco.
 if ReqLst;

 //Trascrive cursore per la riemissione.
 exsr CURT;
 //Chiama elenco su provincia nascita.
 if FLD='W1PRNA';
 CDPR=W1PRNA;
 CNT1WIN(CDPR:NASCITA);
 if CDPR<>*blank;
 W1PRNA=CDPR;
 endif;
 endif;
 endif;

```

```

 //Chiama elenco su provincia residenza.
 if FLD='W1PRRE';
 C DPR=W1PRRE;
 CNT1WIN(C DPR:RESIDENZA);
 if C DPR<>*blank;
 W1PRRE=C DPR;
 endif;
 endif;

 //Decodifica video dati.
 exsr DEW1;
 ///// //Controlla video dati.
 ///// exsr CTW1;
 //Finge errore per non ricaricare da database.
 ErrGen=*on;
 //Ricicla.
 iter;

 //Se corre elenco.
 endif;

 //Se corre decodifica.
 if ReqDec;

 //Decodifica video dati.
 exsr DEW1;
 //Controlla video dati.
 exsr CTW1;
 //Finge errore per non pulire video.
 ErrGen=*on;
 //Ricicla.
 iter;

 //Se corre decodifica.
 endif;

 //Decodifica video dati.
 exsr DEW1;
 //Controlla video dati.
 exsr CTW1;
 //Se errore, ricicla.
 if ReqEnd;
 iter;
 endif;

 //Scrive database.
 exsr WRDB;
 //Se errore, ricicla.
 if ReqEnd;
 iter;
 endif;

 //Cicla intorno a video dati.
 enddo;

 endsr;

 //-----
 //Scrive database.
 begsr WRDB;

 //Esegue.
 dou 1=1;

 //Apri database.
 exsr OPDB;
 //Se errore in apertura, abbandona.
 if ErrGen;
 leave;
 endif;
 //Trascrive dati dal video al record di database.
 exsr W1DB;
 //Accantona la chiave database nei campi dei video guida.
 exsr DBW1KK;
 //Aggiunge il record.
 write(e) UPD;
 //Se la chiave del record esiste gia', accende errore e abbandona.
 if %error;
 ErrGen =*on;
 ErrKey =*on;
 leave;
 endif;

 //Esegue.

```

```

 enddo;

 endsr;

//-----
//Apre database.
begsr OPDB;

//Se il file fisico per update non e' aperto.
if not %open(CNA1);
 //Tenta l'apertura del file.
 open(e) CNA1;
 //Se errore in apertura, accende
 //File non disponibile per modifica e aggiunta.
 if %error;
 ErrGen =*on;
 ErrNotAvl=*on;
 endif;
//Se il file fisico per update non e' aperto.
endif;

endsr;

//-----
//Accantona la chiave database nei campi dei video guida.
begsr DEW1KK;

W0CDPE=A1CDPE;
W0DEPE=A1DEPE;

endsr;

//-----
//Cancella database.
begsr DLDB;

//Esegue.
dou 1=1;
 //Apre database.
 exsr OPDB;
 //Se errore in apertura, abbandona.
 if ErrGen;
 leave;
 endif;
 //Se corre elaborazione per codice.
 if CurByCod;
 //Recupera il record fisico per cancellarlo.
 chain(e) FIDS1RR UPD;
 //Se corre elaborazione per codice.
 endif;

 //Se corre elaborazione per descrizione;
 if CurByDes;
 //Recupera il record fisico per cancellarlo.
 chain(e) FIDS2RR UPD;
 //Se corre elaborazione per descrizione;
 endif;

 //Se il record e' occupato, accende errore e abbandona.
 if %error;
 ErrGen =*on;
 ErrOcc =*on;
 leave;
 endif;

 //Se il record non esiste, accende errore e abbandona.
 if not %found;
 ErrGen =*on;
 ErrExi =*on;
 leave;
 endif;

 //Se i dati del record sono cambiati, accende errore e abbandona.
 if A1D<>A1DSV;
 ErrGen =*on;
 ErrChg =*on;
 leave;
 endif;

 //Cancella il record.
 delete(e) UPD;
 //Se la cancellazione fallisce, accende errore e abbandona.
 if %error;
 ErrGen =*on;

```

```

 ErrNotDlt=*on;
 leave;
 endif;

//Esegue.
enddo;

endsr;

//-----
//Elimina posizionamento cursore.
begsr CURZ;

clear LIN;
clear POS;

endsr;

//-----
//Trascrive cursore per la riemissione.
begsr CURT;

LIN=ROW;
POS=COL;

endsr;

//-----

```

#### 45.6. Utilizza il seguente /COPY di ridenominazione degli indicatori numerici.

```

*FREE // https://www.nicklitten.com/named-indicators-in-rpg/

dcl-s RENIND pointer inz(%addr(*IN));
dcl-ds INDICATORS based(RENIND);
 IN01 ind pos(01); //
 ReqLst ind pos(02); // Richiesto elenco
 ReqEnd ind pos(03); // Richiesta fine
 ReqDec ind pos(04); // Richiesta decodifica
 ReqRfrKey ind pos(05); // Richiesto rinfrescamento per chiave
 IN06 ind pos(06); //
 ReqSwiMod ind pos(07); // Richiesta commutazione modo visualizzazione/modifica
 IN08 ind pos(08); //
 ReqAdd ind pos(09); // Richiesta aggiunta
 IN10 ind pos(10); //
 ReqSwiGui ind pos(11); // Richiesta commutazione guida codice/descrizione
 ReqPrv ind pos(12); // Richiesto video precedente
 IN13 ind pos(13); //
 IN14 ind pos(14); //
 IN15 ind pos(15); //
 IN16 ind pos(16); //
 ReqRfrRrn ind pos(17); // Richiesto rinfrescamento per numero relativo di record
 IN18 ind pos(18); //
 IN19 ind pos(19); //
 IN20 ind pos(20); //
 IN21 ind pos(21); //
 IN22 ind pos(22); //
 ReqDel ind pos(23); // Richiesta cancellazione record
 IN24 ind pos(24); //
 ReqPagUp ind pos(25); // Richiesta pagina su (indietro)
 ReqPagDw ind pos(26); // Richiesta pagina giu' (avanti)
 IN27 ind pos(27); //
 IN28 ind pos(28); //
 IN29 ind pos(29); //
 IN30 ind pos(30); //
 IN31 ind pos(31); //
 IN32 ind pos(32); //
 IN33 ind pos(33); //
 IN34 ind pos(34); //
 IN35 ind pos(35); //
 IN36 ind pos(36); //
 IN37 ind pos(37); //
 IN38 ind pos(38); //
 IN39 ind pos(39); //
 IN40 ind pos(40); //
 CurByCod ind pos(41); // Corre elaborazione per codice
 CurByDes ind pos(42); // Corre elaborazione per descrizione
 IN43 ind pos(43); //
 CurDsp ind pos(44); // Corre protezione campi (Corre visualizzazione)
 CurAdd ind pos(45); // Aggiunta
 IN46 ind pos(46); //
 IN47 ind pos(47); //
 IN48 ind pos(48); //
 IN49 ind pos(49); //
 IN50 ind pos(50); //

```



```

IN51 ind pos(51); //
IN52 ind pos(52); //
IN53 ind pos(53); //
IN54 ind pos(54); //
IN55 ind pos(55); //
IN56 ind pos(56); //
IN57 ind pos(57); //
IN58 ind pos(58); //
IN59 ind pos(59); //
ErrGen ind pos(60); // Errore generico
ErrOcc ind pos(61); // Il record e' occupato
ErrExi ind pos(62); // Il record non esiste
ErrChg ind pos(63); // I dati del record sono cambiati
ErrKey ind pos(64); // La chiave del record esiste gia'
ErrNotAvl ind pos(65); // File non disponibile per modifica e aggiunta
ErrNotDlt ind pos(66); // Record non cancellabile
ErrRfrKey ind pos(67); // Errore rinfrescamento per chiave
ErrRfrRrn ind pos(68); // Errore rinfrescamento per numero relativo di record
IN69 ind pos(69); //
IN70 ind pos(70); //
ErrCodBla ind pos(71); // Codice obbligatorio
ErrDesBla ind pos(72); // Descrizione obbligatoria
ErrDesShr ind pos(73); // Descrizione piu' corta del codice
ErrNasBla ind pos(74); // Provincia di nascita obbligatoria
ErrNasTab ind pos(75); // Provincia di nascita manca in tabella
ErrResTab ind pos(76); // Provincia di residenza manca in tabella
IN77 ind pos(77); //
IN78 ind pos(78); //
IN79 ind pos(79); //
IN80 ind pos(80); //
IN81 ind pos(81); //
IN82 ind pos(82); //
IN83 ind pos(83); //
IN84 ind pos(84); //
IN85 ind pos(85); //
IN86 ind pos(86); //
IN87 ind pos(87); //
IN88 ind pos(88); //
IN89 ind pos(89); //
IN90 ind pos(90); //
IN91 ind pos(91); //
IN92 ind pos(92); //
IN93 ind pos(93); //
IN94 ind pos(94); //
IN95 ind pos(95); //
IN96 ind pos(96); //
IN97 ind pos(97); //
IN98 ind pos(98); //
IN99 ind pos(99); //
ErrAll char(31) pos(60); // Tutti gli indicatori di errore
end-ds INDICATORS;

```

45.7. Utilizza il seguente /COPY per la lettura della posizione del cursore video 5250. .

```

dcl-ds DSPFBK; // Display Specific Feedback Information
DSP_FLAG1 CHAR(2) POS(367); // Display flags
DSP_AID CHAR(1) POS(369); // AID byte
CURSOR CHAR(2) POS(370); // Cursor location
ROW INT(3) POS(370); // Cursor location Row
COL INT(3) POS(371); // Cursor location Column
DATA_LEN INT(10) POS(372); // Actual data len
SF_RRN INT(5) POS(376); // Subfile rrn
MIN_RRN INT(5) POS(378); // Subfile min rrn
NUM_RCDS INT(5) POS(380); // Subfile num rcds
ACT_CURS CHAR(2) POS(382); // Active window cursor location
DSP_MAJOR CHAR(2) POS(401); // Major ret code
DSP_MINOR CHAR(2) POS(403); // Minor ret code
end-ds;

```

## 46. Strutture dei programmi interattivi>NC

- 46.1. Attività presenti solitamente in programmi che usano il video.
- 46.2. Anagrafica.
- 46.3. Gestione documento.
- 46.4. Elenco con scelte.
- 46.5. Lancio di un batch.

## 47. Strutture dei programmi batch>NC

- 47.1. Attività presenti solitamente in programmi che non usano il video.
- 47.2. Acquisizione movimenti.
- 47.3. Stampa.
- 47.4. Estrazione dati.

## 48. Ftp (File Transfer Protocol)>NC

- 48.1. Traslato dati via FTP tra sistemi (PC-IBMi, IBMi-PC, IBMi-PC-IBMi, IBMi-IBMi).
- 48.2. Automatizzazione degli scambi via FTP.

## 49. Aree dati > NC

49.1. Uso e aggiornamento di aree dati.

49.2. Local data area.

## 50. Cenni ad altri tipi di oggetto>NC

- 50.1. File messaggi.
- 50.2. Code dati.
- 50.3. User space.
- 50.4. Device.

## 51. Reperimento di Tool dal web>NC

- 51.1. Tool open source con interfaccia comandi.
- 51.2. Tool in forma solo oggetto.
- 51.3. Tool della libreria NERONI2 da [www.neroni.it](http://www.neroni.it)

## 52. Cobol>NC

- 52.1. Struttura del linguaggio, visione generale della suddivisione statica di ogni programma.
- 52.2. Area A, Area B e livelli.
- 52.3. Metodi di riferimento ai dati, definizioni delle variabili.
- 52.4. Identification Division e Environment Division: principali codici del linguaggio in queste "divisioni".
- 52.5. Data Division: definizione dettagliata dei principali tipi di dati.
- 52.6. Procedure Division: le istruzioni di programma maggiormente utilizzate.
- 52.7. Procedure Division: le istruzioni obsolete da evitare che si trovano nei programmi più datati.
- 52.8. COBOL SQL: differente modalità di accesso ai dati rispetto a quella nativa del sistema.
- 52.9. Le principali funzioni intrinseche che hanno una corrispondenza con le funzioni RPG free.
- 52.10. Differenze principali e problemi da tener presente nella conversione di un programma da COBOL a RPG free.
- 52.11. Debug in Cobol.



## 53. Link

### 53.1. Marco Riva su RPG

[www.markonetools.it/liberiamo-lrpg](http://www.markonetools.it/liberiamo-lrpg)

<https://docplayer.it/204478327-Rpg-free-forever-marco-riva-ultimo-aggiornamento-4-12-2020.html>

### 53.2. Esempi IBM su RPG

[www.ibm.com/support/pages/coding-rpg-iv-beginners-tutorial](http://www.ibm.com/support/pages/coding-rpg-iv-beginners-tutorial)

## 54. Esercitazione finale 1: Convogli ferroviari

### 54.1. Analisi tecnica.

- 54.1.1. Si abbia un deposito di materiale rotabile ferroviario composto da 5 motrici e 20 vagoni passeggeri.
- 54.1.2. Costruire un reference file **CNE0** contenente tutti i campi usati nei file seguenti. Il nome di ciascun campo di quattro caratteri. I nomi dei campi dei file fisici del processo sono di sei caratteri e fanno riferimento ai campi del reference ma aggiungono come prefisso i due caratteri del record in cui si trovano.
- 54.1.3. Costruire una anagrafica delle motrici **CNE1** (record E1) con  
codice rotabile motrice (E1**CDRO** con valori M1, M2, M3, M4, M5),  
peso motrice in tonnellate (E1**PEMO** con valori da 60 a 80),  
peso trainabile in tonnellate (E1**PETR** da 200 a 400),  
priorità di utilizzo motrice (E1**PRUT** da 1 a 5),  
codice convoglio che usa la motrice (E1**CDCV**).
- 54.1.4. Costruire una anagrafica dei vagoni **CNE2** (record E2) con  
codice rotabile vagone (E2**CDRO** con valori da V01 a V20),  
peso vagone in tonnellate (E2**PEVG** da 30 a 50),  
numero massimo passeggeri (E2**PAMX** da 50 a 100),  
priorità di utilizzo vagone (E2**PRUT** da 1 a 20).  
codice convoglio che usa il vagone (E2**CDCV**).
- 54.1.5. Costruire una anagrafica dei convogli **CNE3** (record E3) con  
codice convoglio (E3**CDCV** da C1 a C5),  
codice rotabile motrice (E3**CDRO**),  
5 codici rotabile vagone (E3**CDRO1**, E3**CDRO2**, E3**CDRO3**, E3**CDRO4**, E3**CDRO5**, tutti con  
reference field **CDRO**),  
peso totale vagoni (E3**PEVGT**),  
numero totale passeggeri (E3**PAMXT**).
- 54.1.6. Costruire le viste logiche "1" (**CNE11**, **CNE21**, **CNE31**) a chiave unica (codice) su ciascuno dei tre file.
- 54.1.7. Costruire le viste logiche "2" (**CNE12**, **CNE22**) a chiave virtualmente unica (priorità + codice) su ciascuno dei tre file.
- 54.1.8. Costruire un programma rpg batch **CNEP1** (se necessario con un cappello CL **CNEP1C**) che permette di creare nel file vuoto CNE3 i record convoglio già completamente compilati. Il programma:
  - 54.1.8.1. Riceve un solo numero passeggeri massimo valido per tutti i convogli.
  - 54.1.8.2. Assegna al primo convoglio la motrice con la priorità massima (valore minore).
  - 54.1.8.3. Assegna i vagoni al primo convoglio secondo la loro priorità fino ad eguagliare o superare il numero massimo passeggeri.
  - 54.1.8.4. Genera i convogli successivi fino ad esaurire le motrici o i vagoni.
- 54.1.9. Costruire un programma interattivo **CNEP2** (con dspf **CNEP2W**) che permette di modificare i record convoglio. Il programma:
  - 54.1.9.1. Presume che i record convoglio esistano già.
    - Riempire eventualmente il file con un dfu scrivendo su ogni record solo il codice convoglio.
    - Oppure eseguire il programma già scritto CNEP1.
  - 54.1.9.2. Modifica a video un convoglio permettendo la scelta di motrice e vagoni.
  - 54.1.9.3. Una certa motrice e un certo vagone possono essere usati in un solo convoglio.
  - 54.1.9.4. Controlla che il peso dei vagoni del convoglio non superi il peso trainabile dalla motrice.
  - 54.1.9.5. Calcola e scrive negli appositi campi del record convoglio:
    - il peso totale dei vagoni del convoglio;
    - il numero di persone trasportabili con il convoglio.

## 55. Esercitazione finale 2: Orario scolastico

### 55.1. Analisi tecnica.

55.1.1. Si ha, per una piccola scuola media, un elenco di insegnanti e un elenco di classi.

55.1.2. Si vuole costruire un orario settimanale delle lezioni.

55.1.3. Si crea un unico file fisico CNF1 con il seguente tracciato:

55.1.3.1. codice classe (F1**CLAS** con valori PRI, SEC, TER, QUA, QUI,...),

55.1.3.2. codice insegnante (F1**INSE** con valori ITA, ENG, MAT, GEO, STO,...),

55.1.3.3. codice ora di lezione (F1**ORAS**

con valori 01-08 per il lunedì,

con valori 09-16 per il martedì,

con valori 17-24 per il mercoledì,

con valori 25-32 per il giovedì,

con valori 33-40 per il venerdì).

55.1.4. Creare la vista logica unica 1 per classe-ora.

55.1.5. Non è necessario creare l'anagrafica classi né l'anagrafica insegnanti per creare il programma richiesto qui di seguito.

55.1.6. Si crea il programma interattivo CNFP1 "Crea orario" con display file CNFP1W.

55.1.6.1. Il pgm presenta un video W0 che permette di scegliere la classe di cui si vuole gestire l'orario settimanale, ad esempio "PRI".

55.1.6.2. Il pgm presenta poi un video W1 che porta un'immagine dell'orario settimanale di una singola classe come segue:

Classe: PRI

H Lun Mar Mer Gio Ven

1 xxx xxx xxx xxx xxx

2 xxx xxx xxx xxx xxx

3 xxx xxx xxx xxx xxx

4 xxx xxx xxx xxx xxx

5 xxx xxx xxx xxx xxx

6 xxx xxx xxx xxx xxx

7 xxx xxx xxx xxx xxx

8 xxx xxx xxx xxx xxx

55.1.6.3. La videata in sostanza viene riempita leggendo i record del file CNF1 di una singola classe.

55.1.6.4. I campi **xxx** sono di input/output.

55.1.6.5. Caricando nei campi xxx il codice insegnante opportuno, occorre controllare che lo stesso insegnante non sia già occupato nella stessa ora in un'altra classe. Per farlo, creare la vista logica unica 2 (per insegnante-ora).

55.1.6.6. La trascrizione della videata nel database avviene dopo il controllo: prima si cancellano tutti i record che sono stati usati per presentare la videata e solo allora si ricaricano tutti i record che corrispondono alla nuova versione.

55.1.6.7. Non si gestisce la modifica contemporanea di più utenti.