

RPG

strutture dati e I/O



Marco Riva



www.markonetools.it



Ultimo aggiornamento: 26/11/2020

Strutture dati

- ▶ le strutture dati (DS) sono una importante componente del linguaggio di programmazione RPG
- ▶ aiutano i programmatori a scrivere codice
 - ▶ più chiaro
 - ▶ più solido
- ▶ scopo
 - ▶ organizzare e raggruppare in maniera logica i sottocampi
 - ▶ rendere evidente la correlazione dei sottocampi (perché appartenenti alla stessa tabella, stesso formato video, ...)



2

Strutture dati e I/O

- ▶ le strutture dati sostituiscono le vecchie KLIST
- ▶ i codici operativi di I/O prevedono la possibilità di specificare una **struttura dati** nel campo risultato
- ▶ garantiscono migliori prestazioni soprattutto in applicazioni batch con I/O intenso
- ▶ evitano conflitti per nomi campi uguali in file diversi senza usare keyword `prefix`, avendo la comodità di mantenere i nomi dei sottocampi uguali nelle diverse strutture dati e sfruttando appieno il codice operativo `eval-corr`.
- ▶ la struttura dati deve essere definita con keyword `likerec` (da preferire!) o `extname`
- ▶ con la keyword `likerec` posso specificare se la DS verrà usata come `*input` o `*output` o `*key` (ovverosia solo i campi chiave del file)



3

Strutture dati qualificate

- **qualificate**: keyword `qualified`
 - consentono di avere sottocampi con lo stesso nome presenti in diverse DS
 - codice operativo **eval-corr** che consente di copiare i campi corrispondenti (in base al nome e al tipo dati compatibile) tra due DS
- altre keyword (p.es. `likerec`) prevedono che la struttura dati sia **implicitamente qualificata**



Alternative a KLIST

- Invece di KLIST si può usare una ds definita con `likerec` e la bif `%kds`

```
dcl-ds KeyTest likerec(PROJACT:*key);
```

implicitamente
qualified

```
KeyTest.PROJNO = 'AD3110';
```

```
KeyTest.ACTNO = 10;
```

```
KeyTest.ACSTDAT = '1982-01-01';
```

chiave
completa

```
chain %kds(KeyTest) XPROJAC1;
```

```
...
```

```
setll %kds(KeyTest) XPROJAC1;
```

chiave
parziale

```
do....;
```

```
    reade %kds(KeyTest:1) XPROJAC1;
```

```
    ...
```

```
enddo;
```

il numero di campi
chiave da utilizzare
può anche essere una
variabile



5

Strutture dati e I/O su video/printer files

- L'utilizzo di strutture dati con le operazioni di I/O non è limitata ai file di database, ma si applica anche a file video e printer
- Si semplifica molto la gestione del trasferimento dei campi dal database al video e viceversa
- Il codice operativo **eval-corr** consente di trasferire facilmente il contenuto di due strutture dati anche se non sono identiche
 - la DS del file video potrebbe contenere anche campi provenienti da tabelle diverse
 - più istruzioni eval-corr possono trasferire il contenuto della DS del video nelle diverse DS dei file



➤ File video

➤ File video

➤ File video

➤ File video

Test / FMT01

Test / FMT01

Test / FMT01

Test / FMT01

Test / FMT01

Test / FMT01

Test / FMT01

Test / FMT01

Gestire I/O con strutture dati/2

IMPIEGATI	
Codice	000010
Dati anagrafici	
Iniziale	
Reparto	
Mansione	

INVIO

IMPIEGATI	
Codice	000010
Dati anagrafici	
Iniziale	I
Reparto	A00
Mansione	PRES

```
dcl-f HINT21V workstn inddds(DspFI);
dcl-f EMPLOYEE keyed usage(*update) rename(employee:emprec);

dcl-ds DspFI qualified;
  F3_Fine ind pos(3);
  F6_Aggiorna ind pos(6);
end-ds;

dcl-ds kEmp likerec(emprec:*key);
dcl-ds iEmp likerec(emprec:*input);
dcl-ds oEmp likerec(emprec:*output);
dcl-ds rEmp likerec(emprec:*input);
dcl-ds Video likerec(FMT01:*all);
dcl-s LastEmpNo like(EMPNO);
dcl-s ForEverTrue ind inz(*on);
```

```
dow ForEverTrue;

  exfmt FMT01 Video;
  clear Video.wsmsge;

  select;
  [...];
  // Invio
  other;
  // se specificato codice impiegato visualizzo i dati
  if Video.EMPNO <> *blanks and Video.EMPNO <> LastEmpNo;
    eval-corr kEmp = Video;
    chain(n) %kds(kEmp) employee iEmp;
    eval-corr Video = iEmp;
    eval-corr oEmp = iEmp;
    LastEmpno = Video.Empno;
  endif;
endsl;
```

input/output del formato video avviene tramite la DS Video

tutti i campi del formato FMT01 sono qualificati dalla DS Video.

la DS kEmp per la chiave della tabella viene valorizzata dai campi corrispondenti del video

non alloco record della tabella

il record letto dalla tabella valorizza la DS iEmp

i campi del formato video vengono popolati da iEmp

preparo la DS di output

Gestire I/O con strutture dati/3

valorizzo la DS per l'update della tabella dai campi del video

se ci sono dei campi variati rispetto al record letto dalla tabella

verifico se il record della tabella letto ora è diverso da quello letto inizialmente ovvero se un altro utente ha modificato lo stesso record

```
select;
[...];
when DspfI.F6_Aggiorna;
  eval-corr oEmp = Video;
  if iEmp <> oEmp;
    // leggo il record allocandolo e valorizzo un'altra DS di input
    chain %kds(kEmp) employee rEmp;
    if rEmp <> iEmp;
      Video.wsmsge = 'Modifiche in conflitto con altri utenti';
    else;
      update emprec oEmp;
    endif;
  endif;
[...];
endsl;
```

alla pressione di F6

leggo record dalla tabella e popolo DS rEmp

alloco record della tabella

aggiorno il record con i dati della DS oEmp

IMPIEGATI	
Codice	000010
Dati anagrafici	
Iniziale	I
Reparto	B01
Mansione	MANAGER

F6=Aggiorna



9

Riferimenti



➤ E-mail aziendale: mriva@sirio-is.it



➤ Blog: www.markonetools.it



➤ Facebook: <https://www.facebook.com/markonetools/>



➤ YouTube: <https://www.youtube.com/channel/UCb47YJQJCzU-5x4nnGzDu-w>



➤ E-mail blog: info@markonetools.it



➤ LinkedIn: www.linkedin.com/in/marcoriva-mk1

