

## Query ricorsive *connect by*



Marco Riva



[www.markonetools.it](http://www.markonetools.it)

IBMCHAMPION

2021

Power Systems area



Ultimo aggiornamento: 11/10/2021

# Tabella di esempio DEPARTMENT

- ▶ creare il database di esempio di IBM

```
call create_sql_sample('SCHEMA-NAME');
```

- ▶ prendiamo in esame la tabella DEPARTMENT

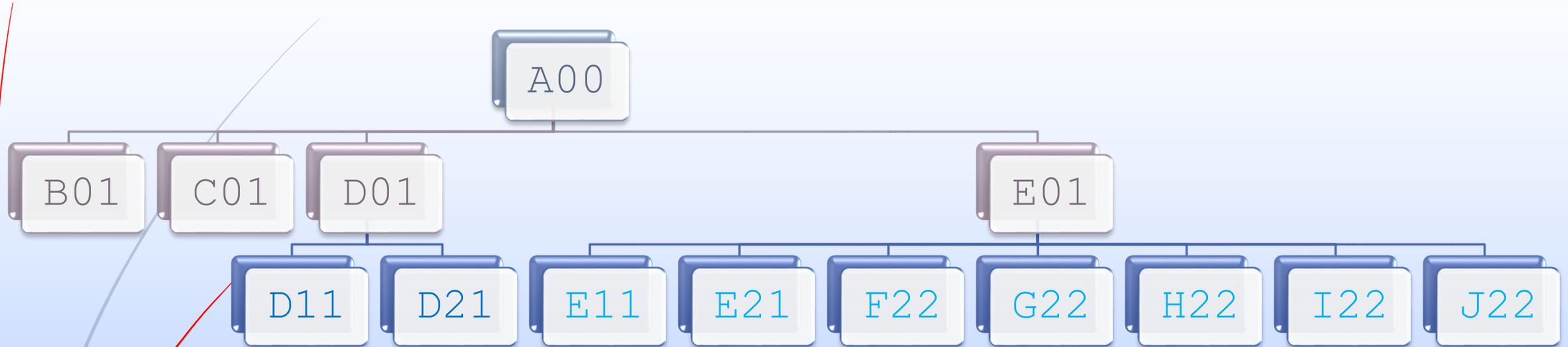
```
select * from DEPARTMENT order by DEPTNO;
```

per ogni ufficio è  
definito nel campo  
ADMRDEPT l'ufficio di  
riferimento

DEPTNO	DEPTNAME	MGRNO	ADMRDEPT	LOCATION
A00	SPIFFY COMPUTER SERVICE DIV.	000010	A00	<NULL>
B01	PLANNING	000020	A00	<NULL>
C01	INFORMATION CENTER	000030	A00	<NULL>
D01	DEVELOPMENT CENTER	<NULL>	A00	<NULL>
D11	MANUFACTURING SYSTEMS	000060	D01	<NULL>
D21	ADMINISTRATION SYSTEMS	000070	D01	<NULL>
E01	SUPPORT SERVICES	000050	A00	<NULL>
E11	OPERATIONS	000090	E01	<NULL>
E21	SOFTWARE SUPPORT	000100	E01	<NULL>
F22	BRANCH OFFICE F2	<NULL>	E01	<NULL>
G22	BRANCH OFFICE G2	<NULL>	E01	<NULL>
H22	BRANCH OFFICE H2	<NULL>	E01	<NULL>
I22	BRANCH OFFICE I2	<NULL>	E01	<NULL>
J22	BRANCH OFFICE J2	<NULL>	E01	<NULL>



# Relazioni gerarchiche tra uffici



A00 è l'ufficio di riferimento per gli uffici B01, C01, D01, E01  
a sua volta D01 è l'ufficio di riferimento per D11 e D21  
e E01 è l'ufficio di riferimento per E11, E21, F22, G22, H22, I22, J22



3

# Query gerarchiche (*Hierarchical queries*)

- ▶ disponibili da IBM i **7.1** (SF99701 liv. 9)
- ▶ descritte nel manuale [DB2 for i SQL reference](#) al cap. 6 *Queries*, par. *Hierarchical queries*
- ▶ le query gerarchiche sono una forma di query ricorsive "semplificate"
- ▶ un altro metodo per costruire query ricorsive consiste nell'utilizzare le CTE (Common Table Expression). Un esempio lo trovate nell'articolo <https://www.markonetools.it/query-ricorsive/>



# Perché "semplificate"?

le query gerarchiche con la clausola `connect by` utilizzano la medesima subquery come "seme"



e come step ricorsivi



# Le 3 clausole fondamentali

step 1: query "seme"

step successivi:  
query ricorsiva

ordinamento in base a  
gerarchia della ricorsività

```
select DEPTNO, DEPTNAME, ADMRDEPT
from DEPARTMENT
start with DEPTNO = 'A00'
connect by nocycle prior DEPTNO = ADMRDEPT
order siblings by DEPTNO;
```

- Dobbiamo immaginare il DB2 che scompone e riscrive questa query in  $n$  subquery che esegue in sequenza e alla fine restituisce come risultato globale l'unione di tutti i risultati parziali



# Il risultato intermedio R

- Prima di capire come viene eseguita la ricorsività dobbiamo immaginare che dalla subquery venga estratto il **risultato R** ottenuto dalla subquery "principale" senza le clausole della query gerarchica

```
select DEPTNO, DEPTNAME, ADMRDEPT  
from DEPARTMENT  
start with DEPTNO = 'A00'  
connect by nocycle prior DEPTNO = ADMRDEPT  
order siblings by DEPTNO;
```

risultato intermedio R

DEPTNO	DEPTNAME	ADMRDEPT
A00	SPIFFY COMPUTER SERVICE DIV.	A00
B01	PLANNING	A00
C01	INFORMATION CENTER	A00
D01	DEVELOPMENT CENTER	A00
D11	MANUFACTURING SYSTEMS	D01
D21	ADMINISTRATION SYSTEMS	D01
E01	SUPPORT SERVICES	A00
E11	OPERATIONS	E01
E21	SOFTWARE SUPPORT	E01
F22	BRANCH OFFICE F2	E01
G22	BRANCH OFFICE G2	E01
H22	BRANCH OFFICE H2	E01
I22	BRANCH OFFICE I2	E01
J22	BRANCH OFFICE J2	E01

N.B. nella subquery può anche essere presente una clausola WHERE.

In questo esempio non ci occorre

# Query "seme": start with

- Il **primo step** (la query seme) la dobbiamo immaginare come l'esecuzione della subquery principale (*in verde*) con aggiunta alla sua clausola *where* la *search-condition* (`DEPTNO = 'A00'`) specificata nella clausola *start with*

```
select DEPTNO, DEPTNAME, ADMRDEPT
from DEPARTMENT
start with DEPTNO = 'A00'
connect by nocycle prior DEPTNO = ADMRDEPT
order siblings by DEPTNO;
```

risultato intermedio  $H_1$

DEPTNO	DEPTNAME	ADMRDEPT
A00	SPIFFY COMPUTER SERVICE DIV.	A00

```
select DEPTNO, DEPTNAME, ADMRDEPT
from DEPARTMENT
where DEPTNO = 'A00';
```



8

# Query "ricorsiva": connect by

l'operatore **prior** può essere anteposto a un nome di campo per referenziare il valore del risultato intermedio precedente

- Gli step successivi producono altri risultati intermedi  $H_2, H_3, \dots$
- Il risultato intermedio  $H_{n+1}$  deriva dal join tra il risultato precedente  $H_n$  con il risultato  $R$ . La *search-condition* (**prior DEPTNO = ADMRDEPT**) per eseguire il join è specificata dopo la clausola **connect by**

```
select DEPTNO, DEPTNAME, ADMRDEPT
from DEPARTMENT
start with DEPTNO = 'A00'
connect by nocycle prior DEPTNO = ADMRDEPT
order siblings by DEPTNO;
```

risultato intermedio  $H_1$

DEPTNO	DEPTNAME	ADMRDEPT
A00	SPIFFY COMPUTER SERVICE DIV.	A00

risultato intermedio R

DEPTNO	DEPTNAME	ADMRDEPT
A00	SPIFFY COMPUTER SERVICE DIV.	A00
B01	PLANNING	A00
C01	INFORMATION CENTER	A00
D01	DEVELOPMENT CENTER	A00
D11	MANUFACTURING SYSTEMS	D01
D21	ADMINISTRATION SYSTEMS	D01
E01	SUPPORT SERVICES	A00
E11	OPERATIONS	E01
E21	SOFTWARE SUPPORT	E01
F22	BRANCH OFFICE F2	E01
G22	BRANCH OFFICE G2	E01
H22	BRANCH OFFICE H2	E01
I22	BRANCH OFFICE I2	E01
J22	BRANCH OFFICE J2	E01

# Esempio: secondo step di ricorsività

```
select R.DEPTNO, R.DEPTNAME, R.ADMRDEPT  
from DEPARTMENT H1 inner join DEPARTMENT R  
on H1.DEPTNO = 'A00' and H1.DEPTNO = R.ADMRDEPT;
```

**nocycle**: se la query intermedia produce tra i record del risultato un record ripetuto di uno step precedente il record viene ignorato

risultato intermedio H<sub>2</sub>

DEPTNO	DEPTNAME	ADMRDEPT
<del>A00</del>	<del>SPIFFY COMPUTER SERVICE DIV.</del>	<del>A00</del>
B01	PLANNING	A00
C01	INFORMATION CENTER	A00
D01	DEVELOPMENT CENTER	A00
E01	SUPPORT SERVICES	A00



10

# Esempio: terzo step di ricorsività

```
select R.DEPTNO, R.DEPTNAME, R.ADMRDEPT
  from DEPARTMENT H2 inner join DEPARTMENT R
    on H2.DEPTNO in ('B01', 'C01', 'D01', 'E01') and H2.DEPTNO = R.ADMRDEPT;
```

risultato intermedio H<sub>3</sub>

DEPTNO	DEPTNAME	ADMRDEPT
D11	MANUFACTURING SYSTEMS	D01
D21	ADMINISTRATION SYSTEMS	D01
E11	OPERATIONS	E01
E21	SOFTWARE SUPPORT	E01
F22	BRANCH OFFICE F2	E01
G22	BRANCH OFFICE G2	E01
H22	BRANCH OFFICE H2	E01
I22	BRANCH OFFICE I2	E01
J22	BRANCH OFFICE J2	E01



# Esempio: quarto step di ricorsività

```
select R.DEPTNO, R.DEPTNAME, R.ADMRDEPT
  from DEPARTMENT H3 inner join DEPARTMENT R
    on H3.DEPTNO in
      ('D11', 'D21', 'E11', 'E21', 'F22', 'G22', 'H22', 'I22', 'J22')
   and H3.DEPTNO = R.ADMRDEPT;
```

risultato intermedio H<sub>4</sub>

DEPTNO	DEPTNAME	ADMRDEPT

nessun record restituito,  
quindi la query ricorsiva è  
terminata e il DB2 può  
restituire il risultato finale...



12

# Risultato finale

- Il risultato finale è la UNION ALL di tutti i risultati intermedi  $H_n$

risultato intermedio  $H_1$

DEPTNO	DEPTNAME	ADMRDEPT
A00	SPIFFY COMPUTER SERVICE DIV.	A00



risultato intermedio  $H_2$

B01	PLANNING	A00
C01	INFORMATION CENTER	A00
D01	DEVELOPMENT CENTER	A00
E01	SUPPORT SERVICES	A00



risultato intermedio  $H_3$

D11	MANUFACTURING SYSTEMS	D01
D21	ADMINISTRATION SYSTEMS	D01
E11	OPERATIONS	E01
E21	SOFTWARE SUPPORT	E01
F22	BRANCH OFFICE F2	E01
G22	BRANCH OFFICE G2	E01
H22	BRANCH OFFICE H2	E01
I22	BRANCH OFFICE I2	E01
J22	BRANCH OFFICE J2	E01



DEPTNO	DEPTNAME	ADMRDEPT
A00	SPIFFY COMPUTER SERVICE DIV.	A00
E01	SUPPORT SERVICES	A00
J22	BRANCH OFFICE J2	E01
I22	BRANCH OFFICE I2	E01
H22	BRANCH OFFICE H2	E01
G22	BRANCH OFFICE G2	E01
F22	BRANCH OFFICE F2	E01
E21	SOFTWARE SUPPORT	E01
E11	OPERATIONS	E01
D01	DEVELOPMENT CENTER	A00
D21	ADMINISTRATION SYSTEMS	D01
D11	MANUFACTURING SYSTEMS	D01
C01	INFORMATION CENTER	A00
B01	PLANNING	A00



13

# Infine mettiamo ordine

- Nelle query gerarchiche si può specificare la clausola `order siblings by` per indicare che si desidera l'ordinamento rispettando la gerarchia

```
select DEPTNO, DEPTNAME, ADMRDEPT
from DEPARTMENT
start with DEPTNO = 'A00'
connect by nocycle prior DEPTNO = ADMRDEPT
order siblings by DEPTNO;
```



DEPTNO	DEPTNAME	ADMRDEPT
A00	SPIFFY COMPUTER SERVICE DIV.	A00
B01	PLANNING	A00
C01	INFORMATION CENTER	A00
D01	DEVELOPMENT CENTER	A00
D11	MANUFACTURING SYSTEMS	D01
D21	ADMINISTRATION SYSTEMS	D01
E01	SUPPORT SERVICES	A00
E11	OPERATIONS	E01
E21	SOFTWARE SUPPORT	E01
F22	BRANCH OFFICE F2	E01
G22	BRANCH OFFICE G2	E01
H22	BRANCH OFFICE H2	E01
I22	BRANCH OFFICE I2	E01
J22	BRANCH OFFICE J2	E01



# Identificare i livelli di ricorsività

- Il numero massimo di livelli di ricorsività è 250
- La pseudo-colonna **LEVEL** identifica il livello di ricorsività

```
select LEVEL, DEPTNO, DEPTNAME, ADMRDEPT
from DEPARTMENT
start with DEPTNO = 'A00'
connect by nocycle prior DEPTNO = ADMRDEPT
order siblings by DEPTNO;
```

LEVEL	DEPTNO	DEPTNAME	ADMRDEPT
1	A00	SPIFFY COMPUTER SERVICE DIV.	A00
2	B01	PLANNING	A00
2	C01	INFORMATION CENTER	A00
2	D01	DEVELOPMENT CENTER	A00
3	D11	MANUFACTURING SYSTEMS	D01
3	D21	ADMINISTRATION SYSTEMS	D01
2	E01	SUPPORT SERVICES	A00
3	E11	OPERATIONS	E01
3	E21	SOFTWARE SUPPORT	E01
3	F22	BRANCH OFFICE F2	E01
3	G22	BRANCH OFFICE G2	E01
3	H22	BRANCH OFFICE H2	E01
3	I22	BRANCH OFFICE I2	E01
3	J22	BRANCH OFFICE J2	E01



# I record "foglia"

- La pseudo-colonna **CONNECT\_BY\_ISLEAF** identifica se si tratta di un record "foglia" ovvero un record da cui non derivano record a un livello di ricorsività successivo

```
select LEVEL, CONNECT_BY_ISLEAF "IsLeaf",  
       DEPTNO, DEPTNAME, ADMRDEPT  
from DEPARTMENT  
start with DEPTNO = 'A00'  
connect by nocycle prior DEPTNO = ADMRDEPT  
order siblings by DEPTNO;
```

LEVEL	IsLeaf	DEPTNO	DEPTNAME	ADMRDEPT
1	0	A00	SPIFFY COMPUTER SERVICE DIV.	A00
2	1	B01	PLANNING	A00
2	1	C01	INFORMATION CENTER	A00
2	0	D01	DEVELOPMENT CENTER	A00
3	1	D11	MANUFACTURING SYSTEMS	D01
3	1	D21	ADMINISTRATION SYSTEMS	D01
2	0	E01	SUPPORT SERVICES	A00
3	1	E11	OPERATIONS	E01
3	1	E21	SOFTWARE SUPPORT	E01
3	1	F22	BRANCH OFFICE F2	E01
3	1	G22	BRANCH OFFICE G2	E01
3	1	H22	BRANCH OFFICE H2	E01
3	1	I22	BRANCH OFFICE I2	E01
3	1	J22	BRANCH OFFICE J2	E01



# Elemento padre

- **CONNECT\_BY\_ROOT** DEPTNO: restituisce il valore dell'elemento "padre"

LEVEL	Padre	DEPTNO	DEPTNAME	ADMRDEPT
1	A00	A00	SPIFFY COMPUTER SERVICE DIV.	A00
2	A00	B01	PLANNING	A00
2	A00	C01	INFORMATION CENTER	A00
2	A00	D01	DEVELOPMENT CENTER	A00
3	A00	D11	MANUFACTURING SYSTEMS	D01



# Il percorso

- La funzione **SYS\_CONNECT\_BY\_PATH** consente di costruire una stringa (tipo dati CLOB(1M)) che rappresenta il "percorso" dall'elemento root all'elemento corrente

```
select LEVEL, SYS_CONNECT_BY_PATH(DEPTNO, '/') "Path",  
DEPTNO, DEPTNAME, ADMRDEPT  
from DEPARTMENT  
start with DEPTNO = 'A00'  
connect by nocycle prior DEPTNO = ADMRDEPT  
order siblings by DEPTNO;
```

LEVEL	Path	DEPTNO	DEPTNAME	ADMRDEPT
1	/A00	A00	SPIFFY COMPUTER SERVICE DIV.	A00
2	/A00/B01	B01	PLANNING	A00
2	/A00/C01	C01	INFORMATION CENTER	A00
2	/A00/D01	D01	DEVELOPMENT CENTER	A00
3	/A00/D01/D11	D11	MANUFACTURING SYSTEMS	D01
3	/A00/D01/D21	D21	ADMINISTRATION SYSTEMS	D01
2	/A00/E01	E01	SUPPORT SERVICES	A00
3	/A00/E01/E11	E11	OPERATIONS	E01
3	/A00/E01/E21	E21	SOFTWARE SUPPORT	E01
3	/A00/E01/F22	F22	BRANCH OFFICE F2	E01
3	/A00/E01/G22	G22	BRANCH OFFICE G2	E01
3	/A00/E01/H22	H22	BRANCH OFFICE H2	E01
3	/A00/E01/I22	I22	BRANCH OFFICE I2	E01
3	/A00/E01/J22	J22	BRANCH OFFICE J2	E01



# Riferimenti



➤ E-mail aziendale: [mriva@sirio-is.it](mailto:mriva@sirio-is.it)



➤ Blog: [www.markonetools.it](http://www.markonetools.it)



➤ E-mail blog: [info@markonetools.it](mailto:info@markonetools.it)



➤ LinkedIn: [www.linkedin.com/in/marcoriva-mk1](http://www.linkedin.com/in/marcoriva-mk1)



➤ Twitter: [@MarcoRiva73](https://twitter.com/MarcoRiva73)



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



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

Power coffee - MK1



19