

Managing the order of the alternate keys in an indexed file

Author: Veryant Support

Saved From: <http://support.veryant.com/support/phpkb/question.php?ID=341>

Sometimes when a user accesses a file migrated from Acucobol or RM/COBOL to either Jisam or c-tree file systems, the order of the alternate keys can change from the previous COBOL dialect used.

In these cases or even when we created a brand new file, it's important to use the appropriate compiler options for handling alternate keys.

When migrating from ACU to Jisam or c-tree or accessing an original file by using Visionj file connector, make sure you compile the program with ';-cko'; option.

This compiler option lists keys in offset order. Without this option keys are listed following the order they're declared in the FILE-CONTROL paragraph.

There is a similar compiler option, ';-crko';, specific to RM/COBOL data files. Just like ';-cko';, without this option keys are listed following the order they're declared in the FILE-CONTROL paragraph. But ';-crko'; handles split keys with multiple segments differently.

The -crko will evaluate all segments of the key for the offset, while -cko only evaluates the first segment for the offset.

Note that the -cko and -crko options change the order in which keys are registered in the physical file (you can verify this with file management utilities such as jutil and ctutil). As a consequence, this option affects XML/ISS dictionaries as well as the I\$IO and file interfaces where keys are indicated by ordinal number.

- When compiled without any option, the order of the keys would be: **sr-key, key2, key3, sr-title**.
- With -cko, the order is changed to follow the order of the fields in the FD: **sr-key, sr-title, key2, key3**.
- With -crko, the order is changed to follow the order of the fields in the FD also for segments: **sr-key, sr-title, key2, key2**.

To see this in detail, compile the program with -cko or -crko and without any option, run it to create the file songs, and look at the file with jutil, using the -x option to display the offsets:

```
jutil -info songs -x
```

The structure of the physical file including all the alternate keys with the length and the offset of each alternate key are:

With no compiler options:

Key	Dups	Segments	Seg.	Offset	Size
1	N	1	1	0	5
2	Y	3	1	40	20
			2	90	15
			3	60	30
3	Y	3	1	40	20
			2	60	30
			3	90	15
4	Y	1	1	5	30

With -cko:

Key	Dups	Segments			
1	N	1	Seg.	Offset	Size
			1	0	5
2	Y	1	Seg.	Offset	Size
			1	5	30
3	Y	3	Seg.	Offset	Size
			1	40	20
			2	90	15
			3	60	30
4	Y	3	Seg.	Offset	Size
			1	40	20
			2	60	30
			3	90	15

With -crko:

Key	Dups	Segments			
1	N	1	Seg.	Offset	Size
			1	0	5
2	Y	1	Seg.	Offset	Size
			1	5	30
3	Y	3	Seg.	Offset	Size
			1	40	20
			2	60	30
			3	90	15
4	Y	3	Seg.	Offset	Size
			1	40	20
			2	90	15
			3	60	30