

# コラム | 舞呉コボ郎のマイグレーション放浪記

## 第2回 メインフレーム to UNIX 編 / COBOL MOVEの深掘り

### 【後半】 [2024.7.1]



前回「第1回 マイグレ創成期 編 / COBOL 一番重要なのはMOVEとIF」はこちら

## COBOL編 ～「MOVEの深掘り」～

今回はMOVEとIFの概略を学んでいただきました。今回は、MOVEを深掘りします。IFの深掘りも今回記載予定でしたが、都合により次回まわしとさせていただきます。

### MOVEの基本的な考え方

MOVEは基本項目同士で互いの属性が異なる場合に何らかの形式変換が行われて転記されます。属性の組み合わせによっては転記が許されずにコンパイルエラーになる場合もあります。

一方または両方が集団項目の場合は英数字転記になり、転記元よりも転記先の桁数が長い場合に空白が補われますが、それ以外の形式変換は行われません。

### 属性のバリエーション

「属性が異なる場合」と記述しましたが、COBOLで扱う属性の種類についても今回整理します。たとえば、下記のように定義されたW-KOMOKUは何バイトの領域でどのようにメモリー配置されるのか？といったことが分かるように説明したいと思います。

```
01 W-KOMOKU PIC 9(4) USAGE COMP-5.
```

良く使う項目属性のメモリー配置がどうなるかは頭に入れておくべきですが、それ以外は使うときにマニュアルを調べることができれば良いです。

下記の表と、以降の説明は、MOVE命令の各項目属性間の転記可否と転記仕様を記載したものです。以降は「転記マトリックス」と記載します。

Micro Focus<sup>TM</sup> Visual COBOL 9.0J for Eclipse のマニュアル  
[リファレンス>COBOL 言語リファレンス>第 2 部：プログラムの定義>手続き部>文>MOVE 文](#) から引用

送信側項目の 項類	受信側データ項目の項類（関連する規則番号を各列に記載）								
	英字	英数字編集/ 英数字	整数/整数以 外	数字編集 外 部/内部浮動 小数点数	各国文字	NCHAR以外 の2バイト文 字	NCHAR2パイ ト文字	ブール	UTF-8
英字	適合/4c	適合/4a	非適合/3a	非適合/3a	適合	非適合/4e	適合	非適合/3f	適合
英数字	適合/4c	適合/4a	適合/4b	適合/4b	適合	非適合/4e	適合	適合/3e	適合
英数字編集	適合/4c	適合/4a	非適合/3a	非適合/3a	適合	非適合/4e	適合	非適合/3f	適合
整数	非適合/3c	適合/4a	適合/4b	適合/4b	適合	非適合/4e	適合	非適合/3f	非適合
整数以外	非適合/3c	非適合/3c	適合/4b	適合/4b	非適合	非適合/4e	非適合	非適合/3f	非適合
数字編集	非適合/3c	適合/4a	適合/4b	適合/4b	適合	非適合/4e	適合	非適合/3f	適合
各国文字	非適合	非適合	適合	適合	適合	適合	適合	非適合	適合
NCHAR以外 の2バイト文 字	非適合/4e	非適合/4e	非適合/4e	非適合/4e	適合	適合/4e	適合	非適合/3f	非適合

送信側項目の 項類	受信側データ項目の項類（関連する規則番号を各列に記載）								
	英字	英数字編集/ 英数字	整数/整数以 外	数字編集 外 部/内部浮動 小数点数	各国文字	NCHAR以外 の2バイト文 字	NCHAR2バイ ト文字	ブール	UTF-8
NCHAR2バイ ト文字	適合	適合	非適合	非適合	適合	適合/4e	適合	非適合	非適合
ブール	非適合/3c	適合/4a	非適合/3b	非適合/3b	非適合	非適合/3g	非適合	適合/4f	非適合
UFT-8	非適合	非適合	非適合	非適合	適合	非適合	非適合	非適合	適合

皆様がお使いのCOBOL製品のマニュアルにも上記と同様の転記マトリックスが記載されているはずですが。

こちらについても、良く使う属性間の転記の仕様は頭に入れて置き、普段使わない属性を含むMOVEをコーディングする際などは、転記マトリックスで仕様を確認しましょう。MOVEによって、転記元（送信側）項目の内部形式が、どのように変換されて、転記先（受信側）に設定されるかを知った上でコーディングすることが大切です。

特に、転記先項目をREDEFINESしたり、部分参照したり、集団項目転記で他の項目へ転記する場合は、物理的なメモリー配置が分かっていると思わぬ不具合を引き起こすことがあるので注意しましょう。

転記マトリックスを使う前にUSAGE句について理解しましょう。下記のように定義された、「COMP-3」とかが何を意味しているかです。

（COMPはCOMPUTIONALの省略形なので、USAGE COMP-3は、USAGE COMPUTIONAL-3と同意です）

```
01 W-KOMOKU1 PIC 9(04)V9(03) USAGE COMP-3.
```

なお、COBOL製品やコンパイル指令によって、COMP-nの意味合いが下記説明内容と異なる場合がありますので、お使いのCOBOL製品で確認してください。例えば、COMP-4が単精度の内部浮動小数点となるCOBOL製品があります。

下表は Micro Focus<sup>TM</sup> Visual COBOL 9.0J を例として、数値として使用される「USAGE句」毎に、「PIC句の例」に記載した定義を行った項目に値12を設定した場合のメモリー配置を示しています。同じ12であってもメモリー配置はさまざまですね。

符号についても勉強しておきましょう。

「バイナリ」項目で符号をもつ形式と浮動小数点は、負数が格納された場合は符号部のビットが「1」となります。

「内部10進数」項目は、PIC句にS(符号)を含む場合は、右端の4ビットに符号が設定されます。

正の場合は16進で「C」、負の場合は16進で「D」が通常は設定されます。

PIC句にS(符号)を含まない場合は、右端の4ビットに16進で「F」が設定されます。

※内部10進数の符号は、COBOL製品によって異なる場合がありますので、お使いのCOBOL製品のマニュアルで確認してください。

「外部10進数」項目は、PIC句にS(符号)を含む場合に、分離符号か重ね符号かをSIGN句で指定することができます。

SEPARATEを指定すると分離符号となり、値が正の場合は文字「+」が、負の場合は文字「-」が符号に設定されます。

SEPARATEが指定されない重ね符号となり、SIGN句の指定によって、項目の左端か右端のバイトに数値と符号を合わせた意味を持つコードが設定されます。

コボ郎の経験からは、99%以上はSIGN句を省略しているので、項目の右端が重ね符号の形式で使用されています。稀に項目の左端が分離符号の形式で使用されています。

それ以外の2つの形式は実物を見たことないですが、すべての形式について具体的に見ていきましょう。

上表のメモリー配置では、「0」と「2」の重ね符号を記載しましたが、重ね符号の一覧を記載しておきます。

符号を付加する前の左端または右端の値	符号付の値を表す文字					
	正の値			負の値		
	文字集合(ASCII)		文字集合(EBCDIC)	文字集合(ASCII)		文字集合(EBCDIC)
	符号(ASCII)	符号(EBCDIC)	符号(EBCDIC)	符号(ASCII)	符号(EBCDIC)	符号(EBCDIC)
0	0(30)	{(7B)	{{(C0)	p(70)	}(7D)	}(D0)
1	1(31)	A(41)	A(C1)	q(71)	J(4A)	J(D1)
2	2(32)	B(42)	B(C2)	r(72)	K(4B)	K(D2)
3	3(33)	C(43)	C(C3)	s(73)	L(4C)	L(D3)
4	4(34)	D(44)	D(C4)	t(74)	M(4D)	M(D4)
5	5(35)	E(45)	E(C5)	u(75)	N(4E)	N(D5)
6	6(36)	F(46)	F(C6)	v(76)	O(4F)	O(D6)
7	7(37)	G(47)	G(C7)	w(77)	P(50)	P(D7)
8	8(38)	H(48)	H(C8)	x(78)	Q(51)	Q(D8)
9	9(39)	I(49)	I(C9)	y(79)	R(52)	R(D9)



項目定義例	備考	値の例	Micro Focus™ Visual COBOL 9.0J を使用した、文字集合がASCIIの場合のメモリ配置	文字集合がEBCDICの場合のメモリ配置	
01 W-KOMOKU-S1 PIC S9(04). 01 W-KOMOKU-S2 PIC S9(04) SIGN IS TRAILING. ※項目の右端が重なり符号	SIGN句が省略されると下行と同様 ISは省略可能	+12 -12	<p>■ コンパイル指令がSIGN"ASCII"の場合</p> <p>(16進) 3 0 3 0 3 1 3 2 (文字) 0 0 1 2</p> <p>(16進) 3 0 3 0 3 1 7 2 (文字) 0 0 1 r</p>	<p>■ コンパイル指令がSIGN"EBCDIC"の場合</p> <p>(16進) 3 0 3 0 3 1 4 2 (文字) 0 0 1 B</p> <p>(16進) 3 0 3 0 3 1 4 B (文字) 0 0 1 K</p>	<p>■ メインフレーム等の場合</p> <p>(16進) F 0 F 0 F 1 C 2 (文字) 0 0 1 B</p> <p>(16進) F 0 F 0 F 1 D 2 (文字) 0 0 1 K</p>
01 W-KOMOKU-S3 PIC S9(04) SIGN IS TRAILING SEPARATE CHARACTER. ※項目の右端が分離符号	ISとCHARACTERは省略可能	+12 -12	<p>■ コンパイル指令がSIGN"ASCII"の場合</p> <p>(16進) 3 0 3 0 3 1 3 2 B (文字) 0 0 1 2 +</p> <p>(16進) 3 0 3 0 3 1 3 2 D (文字) 0 0 1 2 -</p>	<p>■ コンパイル指令がSIGN"EBCDIC"の場合</p> <p>(16進) 3 0 3 0 3 1 3 2 B (文字) 0 0 1 2 +</p> <p>(16進) 3 0 3 0 3 1 3 2 D (文字) 0 0 1 2 -</p>	<p>■ メインフレーム等の場合</p> <p>(16進) F 0 F 0 F 1 F 2 4 E (文字) 0 0 1 2 +</p> <p>(16進) F 0 F 0 F 1 F 2 6 0 (文字) 0 0 1 2 -</p>
01 W-KOMOKU-S4 PIC S9(04) SIGN IS LEADING. ※項目の左端が重なり符号	ISは省略可能	+12 -12	<p>■ コンパイル指令がSIGN"ASCII"の場合</p> <p>(16進) 3 0 3 0 3 1 3 2 (文字) 0 0 1 2</p> <p>(16進) 7 0 3 0 3 1 3 2 (文字) p 0 1 2</p>	<p>■ コンパイル指令がSIGN"EBCDIC"の場合</p> <p>(16進) 7 B 3 0 3 1 3 2 (文字) [ 0 1 2</p> <p>(16進) 7 D 3 0 3 1 3 2 (文字) ] 0 1 2</p>	<p>■ メインフレーム等の場合</p> <p>(16進) C 0 F 0 F 1 F 2 (文字) [ 0 1 2</p> <p>(16進) D 0 F 0 F 1 F 2 (文字) ] 0 1 2</p>
01 W-KOMOKU-S5 PIC S9(04) SIGN IS LEADING SEPARATE CHARACTER. ※項目の左端が分離符号	ISとCHARACTERは省略可能	+12 -12	<p>■ コンパイル指令がSIGN"ASCII"の場合</p> <p>(16進) 2 B 3 0 3 0 3 1 3 2 (文字) + 0 0 1 2</p> <p>(16進) 2 D 3 0 3 0 3 1 3 2 (文字) - 0 0 1 2</p>	<p>■ コンパイル指令がSIGN"EBCDIC"の場合</p> <p>(16進) 2 B 3 0 3 0 3 1 3 2 (文字) + 0 0 1 2</p> <p>(16進) 2 D 3 0 3 0 3 1 3 2 (文字) - 0 0 1 2</p>	<p>■ メインフレーム等の場合</p> <p>(16進) 4 E F 0 F 0 F 1 F 2 (文字) + 0 0 1 2</p> <p>(16進) 6 0 F 0 F 0 F 1 F 2 (文字) - 0 0 1 2</p>

Micro Focus™ Visual COBOL 9.0J for Eclipse のマニュアル

リファレンス>COBOL 言語リファレンス>第 1 部 : 概念>COBOL 言語の概念>データの字類および項類

Micro Focus™ Visual COBOL 9.0J for Eclipse のマニュアル

リファレンス>COBOL 言語リファレンス>第 2 部 : プログラムの定義>データ部>ファイル記述項およびデータ記述項>データ記述項の骨組み>PICTURE 句

列Aまたは行1の記述内容	識別方法の概要				
英字	PICTURE文字列に「A」が指定されている場合。 表意定数 SPACE				
英数字	PICTURE文字列に「X」が指定されている場合。 表意定数 SPACE,QUOTE「"」,LOW-VALUE,HIGH-VALUE,ZERO(転記先がD列とE列以外の場合),英数字文字列 ※英数字転記で表意定数が用いられる場合は、転記先項目のバイト数分だけ該当文字が設定される。 01 W-KOMOKU-ZERO PIC X(04). MOVE ZERO TO W-KOMOKU-ZERO この命令でW-KOMOKU-ZEROに4桁の「0」が下記のように設定される。 (文字) <table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0
0	0	0	0		
英数字編集	PICTURE文字列に「A」や「X」、編集記号（「B」、「0」、「/」）を含む場合。 ※厳密な条件はマニュアル参照				
整数	PICTURE文字列に「9」が含まれ、かつ、「V(仮想小数点)」を含まない場合。表意定数ZERO(転記先がD列またはE列の場合)。整数の数字文字列。 ※USAGEはDISPLAY,COMP,COMP-4,BINARY,COMP-3,PACKED-DECIMAL,COMP-5,COMP-X				
整数以外	PICTURE句が存在する場合は、PICTURE文字列に、「9」と「V(仮想小数点)」を含む場合。整数以外の数字文字列。 ※USAGEはDISPLAY,COMP-1,FLOAT-SHORT,COMP-2,FLOAT-LONG,COMP-3,PACKED-DECIMAL				
数字編集	PICTURE文字列に「9」や編集記号（0、「B」、「/」、「Z」、「*」、「+」、「,」、「.」、「-」、「CR」、「D B」、「¥」）を含む場合。 ※厳密な条件はマニュアル参照				
各国文字	PICTURE文字列に「N」が指定され、かつ、USAGE句に「NATIONAL」が指定されるか、USAGE句を省略してコンパイル指令NSYMBOL"NATIONAL"が指定される場合。 ※UTF-16のBMP(基本多言語面) がエンコードに用いられ、1文字が2バイトで表現される				
NCHAR以外の2バイト文字	コンパイラ指令DBCS=3が指定され、かつ、PICTURE文字列に「G」が指定され、かつ、USAGE句に「DISPLAY-1」が指定される場合。 ※コンパイラ指令で指定したCOBOL方言によってはUSAGE句省略時の既定値がDISPLAY-1となる場合がある				
NCHAR2バイト文字	NCHARコンパイラ指令NCHAR=2またはパラメータを省略されたNCHARが指定され、かつ、PICTURE文字列に「N」が指定され、かつ、USAGE句に「NCHAR」が指定されるかUSAGE句が省略された場合。				
ブール	PICTURE文字列に「1」が指定されている場合。				
UTF-8	PICTURE文字列にU(n)が指定されている場合。				
数字編集の外部浮動小数点	PICTURE文字列に「E」が含まれ、「+」、「-」、「9」、「.」、「V」で構成されている場合。 (定義例) 01 01 W-KOMOKU-6 PIC +9999.99E+999.				

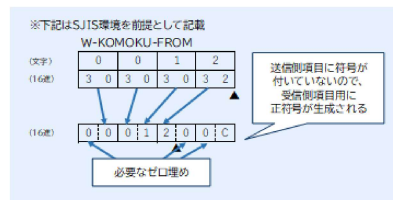
下記のMOVEを例に転記マトリックスを使ってみます。

01 W-KOMOKU-FROM PIC 9(4) VALUE 12.  
 01 W-KOMOKU-TO PIC S9(5)V9(02) USAGE COMP-3.  
 MOVE W-KOMOKU-FROM TO W-KOMOKU-TO.

1. 転記元は整数、転記先は非整数なので、交差箇所を見ると、「適合/4b」となっていることがわかります。
2. 後述説明の4.bに記載されている内容のうち、条件に該当する箇所赤点線で囲んだ箇所を下記に抽出し、それに従った転記を図示します。

転記マトリックスとそれが示す引用箇所を記載します。

	A列	B列	C列	D列	E列	F列	G列	H列	I列	J列
行1	送信側項目の項類	英字	英数字編集/英数字	整数/整数以外	数字編集外部/内部浮動小数点数	各国文字	NCHAR以外の2バイト文字	NCHAR 2バイト文字	ブール	UTF-8
行2	英字	適合/4c	適合/4a	非適合/3a	非適合/3a	適合	非適合/4e	適合	非適合/3f	適合
行3	英数字	適合/4c	適合/4a	適合/4b	適合/4b	適合	非適合/4e	適合	適合/3e	適合
行4	英数字編集	適合/4c	適合/4a	非適合/3a	非適合/3a	適合	非適合/4e	適合	非適合/3f	適合
行5	整数	非適合/3c	適合/4a	適合/4b	適合/4b	適合	非適合/4e	適合	非適合/3f	非適合
行6	整数以外	非適合/3c	非適合/3c	適合/4b	適合/4b	非適合	非適合/4e	非適合	非適合/3f	非適合
行7	数字編集	非適合/3c	適合/4a	適合/4b	適合/4b	適合	非適合/4e	適合	非適合/3f	適合
行8	各国文字	非適合	非適合	適合	適合	適合	適合	適合	非適合	適合
行9	NCHAR以外の2バイト文字	非適合/4e	非適合/4e	非適合/4e	非適合/4e	適合	適合/4e	適合	非適合/3f	非適合
行10	NCHAR 2バイト文字	適合	適合	非適合	非適合	適合	適合/4e	適合	非適合	非適合
行11	ブール	非適合/3c	適合/4a	非適合/3b	非適合/3b	非適合	非適合/3g	非適合	適合/4f	非適合
行12	UTF-8	非適合	非適合	非適合	非適合	適合	非適合	非適合	非適合	適合



Micro Focus™ Visual COBOL 9.0J for Eclipse のマニュアル  
 リファレンス>COBOL 言語リファレンス>第 2 部：プログラムの定義>手続き部>文>MOVE 文 から引用

※ 下記のMicro Focus™ Visual COBOL 9.0J for Eclipseのマニュアルに記載されている欄内のEnt/MFなどの記述は対応するコンパイル指令で設定するCOBOLの方言に対応しています。

- b. 数字編集データ項目は、数値または数字編集
- ⑤0090 MF またはプールデータ項目に転記できない。
  - ⑤545 現在、この制限は廃止されている。
- c. 数値定数、表意定数 ZERO、数値データ項目、数字編集データ項目
- ⑤0090 MF、またはプールデータ項目は、英字データ項目に転記できない。
- d. 整数でない数値定数または整数でない数値データ項目は、英数字または英数字編集のデータ項目
- ⑤0090 MF またはプールデータ項目に転記できない。
- e. ⑤0090 MF identifier-2 がプールデータ項目である場合、受け取ることができるのは、プール文字が関連付けられた表意定数のみである。
- f. ⑤0090 MF 英字、英数字編集、数値、または各国文字のデータ項目は、プールデータ項目に転記できない。
- g. ⑤0090 MF プールデータ項目は、各国文字データ項目に転記できない。
- h. Ent MF UTF-8 以外の項目を UTF-8 項目に転記する場合、送信側の文字は UTF-8 に変換される。この変換は、埋め込みも切り詰めも引き起こさない十分な長さを持つ項類が UTF-8 の一時データ項目に転記された場合と同様である。変換後の項類が UTF-8 の項目が送信側のデータ項目として扱われる。
- i. Ent MF 字類 UTF-8 の項目を字類 UTF-8 の別の項目に送信する場合、転記時に変換は行われない。
- j. Ent MF UTF-8 の転記時に発生する可能性がある桁寄せおよび埋め込みまたは切り詰めについて、グラフィック文字を形成する複数の符号化方式単位が切り詰めによって分けられないようにすることはプログラマーの責任である。
- k. Ent MF 受信側項目が動的長基本項目である場合、その長さは、送信側の長さの最小値、または LIMIT 指定で明示的か暗黙的に示された値に設定される。送信側項目の長さが受信側項目の長さよりも長い場合、右側の超過した文字が切り詰められる。
- l. 上記以外の基本項目の転記は文法的に正しく、下記の一般規則 5 に示す規則に従って実行される。
4. 文法的に正しい基本項目の転記の実行中に、必要に応じて、ある形式の内部表現から別の形式の内部表現にデータが変換される。これは、受信側項目に対して指定された編集処理
- ⑤555、または受信側項目によって暗黙指定された逆編集とあわせて行われる。
- a. 受信側項目が英数字編集または英数字の項目である場合、「COBOL 言語の概念」の章の「標準桁寄せ規則」トピックに記載のとおり、桁寄せおよび必要な空白の穴埋めが行われる。送信側項目のサイズが受信側項目のサイズよりも大きい場合は、受信側項目が満たされた後で、右側の超過文字が切り捨てられる。送信側項目が符号付き数値として記述されている場合、演算符号は転記されない。演算符号が別個の文字位置を占める場合（「SIGN 句」トピックを参照）、その演算符号は転記されず、送信側項目は実際のサイズ（標準データ形式の文字でのサイズ）よりも 1 文字小さいものと見なされる。
- b. 受信側項目が数値または数字編集の項目である場合、「COBOL 言語の概念」の章の「標準桁寄せ規則」セクションに記載のとおり、小数点による桁寄せおよび必要なゼロの穴埋めが行われる。ただし、編集要件によりゼロが置き換えられている場合は除く。
- ⑤545 送信側の作用対象が数字編集である場合、その作用対象の編集されていない数値（符号付きの場合がある）を取得するために、逆編集が暗黙指定される。その後、編集されていない数値が受信側項目に転記される。逆編集の効果は、DE-EDIT コンパイラ指令の設定に依存する。
- 受信側項目が符号付き数値である場合、送信側の符号は受信側項目に転記される。（「SIGN 句」のトピックを参照）。必要に応じて、符号の表現形式の変換が行われる。送信側項目に符号が付いていない場合、受信側項目用に正符号が生成される。
- 受信側項目が符号なしの数値である場合、送信側項目の絶対値が受信側項目に転記され、受信側項目用の演算符号は生成されない。
- 送信側項目が英数字として記述されたデータ項目である場合、送信側項目が符号なしの整数として記述された場合と同様にデータが転記される。
- 受信側項目が数値または数字編集であり、送信側項目が英数字である際に、送信側項目の内容が整数でない場合、転記結果は不定となる。送信側項目が英数字の定数であり、その内容が整数でない場合は、エラーが報告されて、ゼロがその対象に転記される。「手続き部」の章の「矛盾するデータ」トピックを参照。
- ⑤555 送信側項目が数値データ項目である場合、転記し戻すと数字編集項目に同じ値が現れるように転記処理が行われる（ただし、切り捨てが発生した場合は除く）。データ項目内のデータが、編集された PICTURE 文字列に準拠しない場合、ゼロがその対象に転記される。
- c. 受信側項目が英字である場合、「COBOL 言語の概念」の章の「標準桁寄せ規則」トピックに記載のとおり、桁寄せおよび必要な空白の穴埋めが行われる。送信側項目のサイズが受信側項目のサイズよりも大きい場合は、受信側項目が満たされた後で、右側の超過文字が切り捨てられる。
- d. ⑤515 ⑤520 MF 受信側項目が浮動小数点数である場合、最初に送信側項目が内部浮動小数点数に変換された後に転記される。外部浮動小数点数項目へ、または外部浮動小数点数項目からデータを転記する場合、最初に内部浮動小数点数値のデータ変換が行われる。
- e. ⑤520 MF 受信側項目が NCHAR でない 2 バイト文字である場合、送信側項目も 2 バイト文字または各国文字とする。変換は行われない。送信側項目と受信側項目のサイズが異なる場合、データ項目の右側が切り捨てられるか 2 バイト文字の空白で穴埋めされる。

※ なお 2 バイト文字を含む転記については、Micro Focus<sup>TM</sup> Visual COBOL 9.0J マニュアルの下記箇所を参照してください。

Micro Focus<sup>TM</sup> Visual COBOL 9.0J for Eclipse のマニュアル

リファレンス>COBOL 言語リファレンス>第 3 部：追加トピック 2 バイト文字サポートの Micro Focus 拡張>手続き部>MOVE 文 (DBCS)

今回は細かい説明が多かったのですが、マニュアルを読み解くために役立つ情報が提供できたのではないかと思います。項目属性をしっかり押さえ、マニュアルに記載されている転記規則が理解できるようになることが大切です。

⇒次回へ続く

※記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

※免責事項

- ・当コラムに記載した内容に関して、いかなる保証もするものではありません。
- ・記載内容は予告なく変更する可能性があり、記載内容に誤りがあった場合でも一切責任を負いかねます。