

C I I シンタックスルール

1. 1 1 及び 1. 5 1

平成 8 年 7 月

(助) 日本情報処理開発協会

産業情報化推進センター

KEIRIN



この事業は、競輪の補助金を受けて実施したものです。





はじめに

昭和62年から、EDIの業界標準化作業を開始した(株)日本電子機械工業会(EIAJ)は、(財)日本情報処理開発協会 産業情報化推進センター(CII)と共同で、昭和63年に新しいEDI標準の開発を行った。このEDI標準では、流通業界に比べてデータ項目の多い電子機器業界特有のメッセージ構造を考慮し、既に米国で採用されていた可変長フォーマットを採用することになった。様々な検討の結果、シンタックスルール、標準メッセージおよび標準データ項目で構成するのが合理的であるとの結論になったが、このようなフォーマットの構成方法は、米国のANSI X.12や当時審議中であったEDIFACT(ISO9735)と同一である。

シンタックスルールは新たに開発することになり、CIIが提案したデータタグ方式シンタックスルールを、EIAJの改良提案にもとづいて、データ項目の区切りをデリミターからレンジスタグに変更して設計された。このシンタックスルールは、当初から漢字データの使用や他業界への適用も考慮されていたが、当面の単一業界(電子機器業界)への適用を前提に、一部の拡張機能は保留されることになり、電子機器業界での限定使用になることから、EIAJシンタックスルールと命名された。EIAJでは、EIAJシンタックスルールの使用を前提にした標準メッセージと標準データ項目の開発を進めたが、これらは、EIAJ標準メッセージ、EIAJ標準データ項目と呼ばれるようになった。

昭和63年秋に、実用性を確認するトライアル(試行)が行われ、好結果を得たため、平成元年4月に、EIAJシンタックスルール、EIAJ標準メッセージおよびEIAJ標準データ項目をまとめて、『EIAJ取引情報化対応標準1A』として、電子機器業界のEDI標準とした。これが、EIAJ標準である。

EIAJ標準は、その後順調に電子機器業界への普及が進んだが、電子機器業界での成果を他業界でも注目するようになり、EIAJ標準を使用したいという電子機器以外の業界が増加してきた。しかし、EIAJ標準のベースになっているEIAJシンタックスルールは、当面の電子機器業界への適用を前提に、一部の機能が保留されていたため、他業界では使用できないという問題が生じてきた。保留されているために障害となった機能とは、使用可能データ項目数を240種以上に拡張する機能である(EIAJ標準では、最大239種のデータ項目を使用できる)。

電子機器業界単独での使用であれば、239種のデータ項目でもメッセージの設計は不可能ではないが、例えば、製造業界全体での使用になった場合には、239種ではとうてい足りない状況になる。そこで、当初保留された拡張機能を復活することになったが、予想を遙に越えるEIAJ標準の普及を考慮し、既に表面化してきた不具合点も含めて、EIAJシンタックスルールを改良することになり、その検討は平成2年度から開始された。

この検討は、E I A J シンタックスルールの原提案者であるC I I において行われ、平成2年末から平成3年にかけて、E I A J との調整が行われた。E I A J シンタックスルールの改良した新しい標準は、『C I I シンタックスルール』と呼ぶことになったが、従来のE I A J 標準との互換性の確保（上方互換）について特に留意された。さらに、国際標準として制定されたE D I F A C T (I S O 9 7 3 5) との互換性についても真剣な検討が行われたが、メッセージの内部構造も含めた完全な互換性を確保するのは難しいため、メッセージ・グループの外部構造についての互換性を確保したオプションを、当面の対策として追加している。こうして、平成3年4月1日に、『C I I シンタックスルール試作仕様1.00』がまとめられ、さらに、C I I シンタックスルールの採用を決めた建設業界等からの要望などを追加し、平成3年8月28日に『C I I シンタックスルール試作仕様1.02』が発表された。平成3年度～4年度にかけて、トランスレーターの開発も行われ、実用化テストも実施された。平成4年7月1日からは、一般ユーザーに対してトランスレーターの供給も行われるようになり、平成4年8月28日付けで『C I I シンタックスルール1.10』として公開することになった。

C I I シンタックスルールは、このような経過で誕生した、わが国の国内取引に用いるE D I のための標準である。最大の特徴は、漢字の使用と効率の向上である。長期に渡って使用可能な標準であり、業界間や業際取引にも適用可能である。今後、C I I シンタックスルールをベースにした標準メッセージの開発が各業界で行われ、わが国のE D I 標準化の進行が期待される。

平成4年8月28日

産業情報化推進センター

『C I I シンタックスルール 1.1 1 及び 1.5 1』

1993. 3. 31
(産業情報化推進センター)

C I I シンタックスルールは、我が国の E D I の標準化を促進するために、新たに開発された構文規則である。このシンタックスルールは、(財)日本電子機械工業会で開発された E I A J シンタックスルール(電子機器業界の E I A J 標準)を、他の業界にも適用できるように拡張したシンタックスルールで、E I A J シンタックスルールに対して上方互換の機能設定になっている。

本仕様は、実用仕様として公開した「C I I シンタックスルール 1.1 0 (1992.8.28)」の改訂版で、『C I I シンタックスルール 1.1 1』と『C I I シンタックスルール 1.5 1』の 2 種類で構成されている。『C I I シンタックスルール 1.1 1』は、同 1.1 0 に Y 属性を追加した規格であり、それ以外は同一である。『C I I シンタックスルール 1.5 1』は同 1.1 1 にさらに設計画像データ伝送機能を追加した規格で、バージョン 1.5 1 がバージョン 1.1 1 を包含する形になっている。

バージョン 1.1 1 とバージョン 1.1 0 は、ほとんどの場合互換性をとることができ、通常は一緒に使うことができる(詳細は、付属資料の第 7 項を参照)。バージョン 1.5 1 についても、設計画像データ伝送機能を使う時には、専用のトランスレーターが必要になるが、そうでなければ、バージョン 1.1 0 や 1.1 1 と一緒に使うことができる。

本仕様の管理は、今後とも当センターである(財)日本情報処理開発協会 産業情報化推進センターで行う。また、著作権を含む工業所有権は当センターに帰属する。しかしながら、我が国の E D I の普及と標準化推進のため、当センターは本仕様の仕様を広く公開するとともに、無償での使用を認める。同時に、本仕様の一部を変更したような紛らわしいシンタックスルールの仕様は、標準化の乱れにつながる類似規格の発生を防止するため、排除の対象になることに注意されたい。

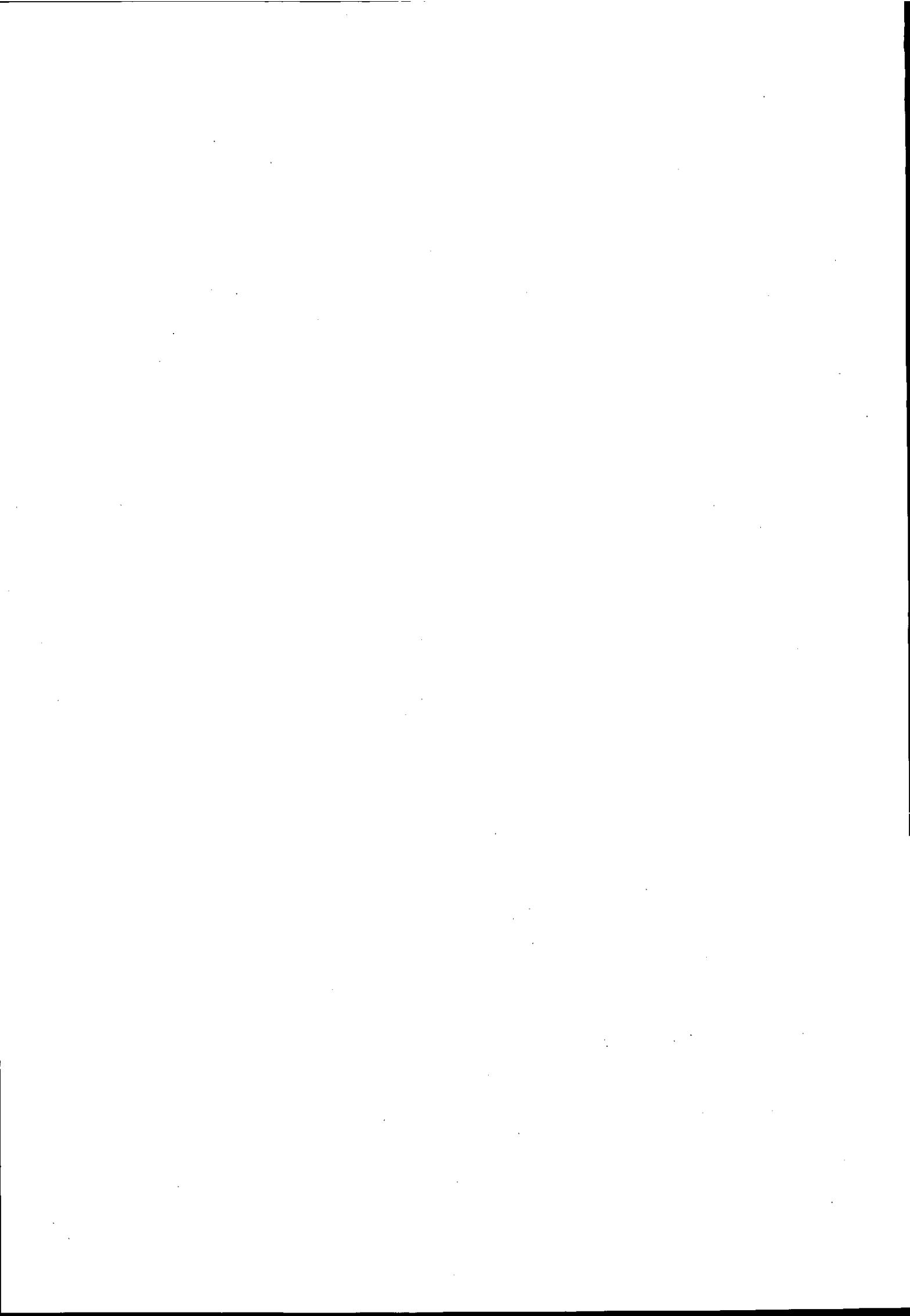
さらに、本仕様によってベンダーで開発されたトランスレーターについては、C I I シンタックスルール準拠表示を、製品に行うことを認める。但し、粗悪品等の発生があった場合、C I I シンタックスルール準拠表示の中止を求めることがある。当センターでは、このような事態を防止するため、当センターの示す特別な手続きを実施したトランスレーターに対して、『産業情報化推進センター推奨』表示を許可することとしている。

本仕様に関する問い合わせ先を、下記に示す。

〒105 東京都港区芝公園 3-5-8 機械振興会館内

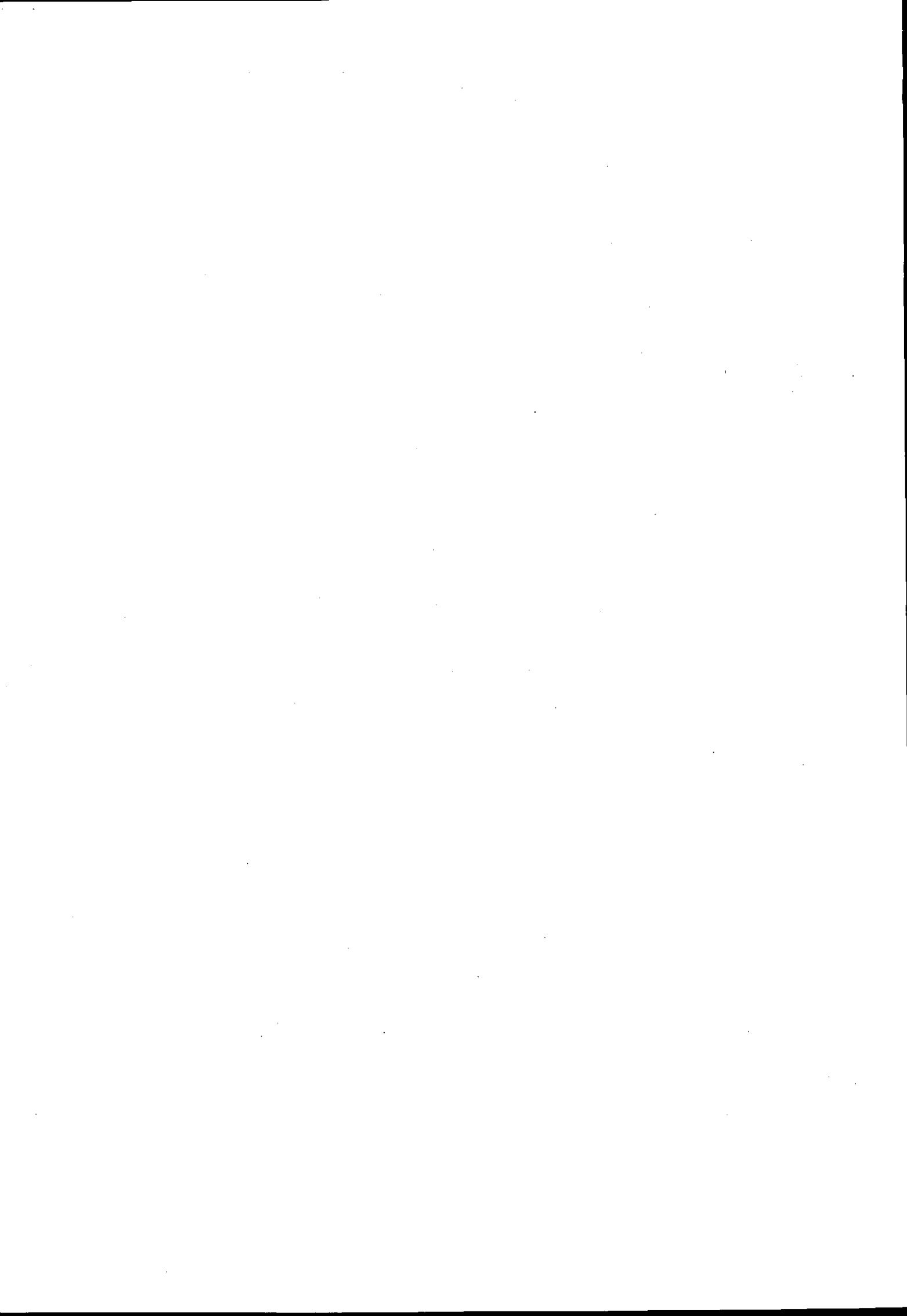
(財)日本情報処理開発協会 産業情報化推進センター

TEL 03-3432-9386 FAX 03-3432-9389



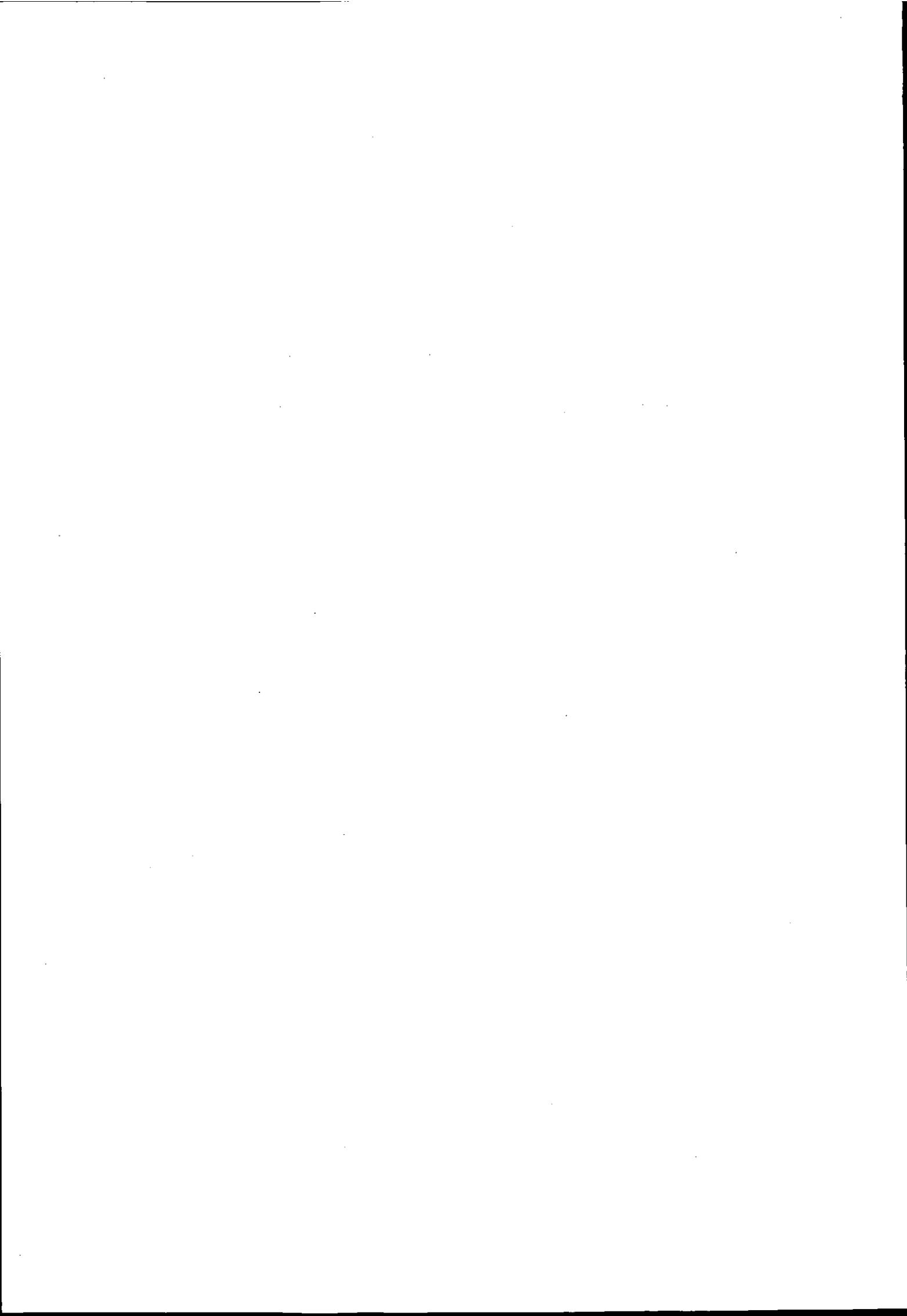
第5章	メッセージグループ・ヘッダーの構造	20
5.1	拡張モード (C 2 2)	20
5.2	トータル項目No.	20
5.3	分割モード (C 2 3)	20
5.4	非透過モード (C 2 6)	20
第6章	メッセージの構造	22
6.1	メッセージヘッダー	22
6.2	TFDエリア(Transfer Form Data Area)の詳細	22
6.3	TFDの省略	24
6.4	TFD形式データ2 (TFD制御子)の機能	24
6.5	データの繰り返し (繰り返しルール:表形式データ)	26
6.6	縮小/拡張モードとマルチ明細の関係	28
6.7	メッセージトレーラ	28
6.8	メッセージの格納構造 (分割モード)	29
第7章	バイナリーデータの構造	30
7.1	バイナリーデータの基本構造	30
7.2	バイナリーデータ・ヘッダー	30
7.3	バイナリーユニット	32
7.4	バイナリーデータ・トレーラ	32
7.5	バイナリーデータの格納構造	34
7.6	設計画像EDI専用データタグ番号	35
7.7	補 足 (トランスレーターにおける設計画像データ指示子の位置づけ)	37
第8章	特殊データの構造	38
8.1	受信確認メッセージ	38
8.2	エラー情報メッセージ	38
8.3	同報ヘッダー	42
第9章	メッセージグループ・トレーラの構造	44
9.1	トータルチェック	44
9.2	チェック用トータル数値の算出方法	44

第10章	メッセージグループの構造	46
10.1	業務処理電文	46
10.2	運用電文	47
10.3	同報電文	49
第11章	Zレベルオプション仕様	50
11.1	非透過モード・オプション	50
11.2	TYPE-Eオプション	53
Ⅲ	附属資料	61
附.1	マルチ明細に関する補足	61
附.2	外字とJIS補助漢字(JIS-X0212)について	62
附.3	ユニコード(JIS-X0221)について	63
附.4	EIAJシンタックスルールとの互換性	63
附.5	非透過モード・オプションについて	64
附.6	CIIシンタックスルールのバージョン番号について	64
附.7	Y属性について	64



I 概 要

『C I I シンタックスルールの概要』



1.1 特 徴

C I I シンタックスルールは、我が国の E D I に適合するよう使用可能文字と使用通信システムを特に考慮した体系である。以下に主な特徴を述べる。

- ① 製造業界など、項目数の多いメッセージのコーディングに適した、可変長フォーマットの採用。
- ② 英数字、カタカナ、ひらがな、漢字など、我が国の E D I で必要とする文字をサポート。
- ③ 複雑な業務処理に対応できる多機能なメッセージ構造（繰返明細の 9 重のネスティングが可能）。
- ④ 最小のメッセージ長を実現する効率的なメッセージ・コントロール構造（データタグ／制御タグ方式）。
- ⑤ 我が国の E D I の標準化に十分な、6 1 4 3 9 種のデータ項目を使用できる。
- ⑥ 通常の E D I には十分な、最大長 3 2 7 6 7 文字（漢字の場合は、1 6 3 8 3 文字）のデータ項目が取り扱える。
- ⑦ E I A J シンタックスルールと互換性がある。シンタックスレベルで、上方互換になっている。
- ⑧ 以上の特徴を十分に発揮させるトランスレーターのサポート（C I I トランスレーターは、様々なコンピュータに対して供給される）。

尚、ベンダーから実際に開発提供されるトランスレーターでは、取り扱い可能な最大データ項目長などは、上記の値よりも小さい場合があるので、ユーザーが導入する際には、注意が必要である。

1.2 基本構造

C I I シンタックスルールは、ファイル転送（バッチ型、リアルタイム型）で実現する E D I のために設計されており、その基本形式は、一つのメッセージグループ・ヘッダー、複数のメッセージ（複数のバイナリデータ）及び一つのメッセージグループ・トレーラで構成されるファイルである（図 I - 1 の上段を参照）。メッセージグループ・ヘッダー、メッセージ及びメッセージグループ・トレーラは、それぞれ一つのレコードに収容される（一つのメッセージを一つのレコードに収容）。この構造を基本形式として分割モードと称する一つのメッセージを複数の固定長レコードに収容した形式がある（図 I - 1 の下段を参照）。

(1) 基本形式

図 I - 1 の上段の構造で、C I I シンタックスルールの基本形式である。メッセージグループ・ヘッダーとメッセージグループ・トレーラは、それぞれ一つの 2 5 1 byte の固定長レコードに収容され、一つのメッセージが一つの可変長レコードに収容される。

(2) 分割モード

図 I - 1 の下段の構造であり、基本形式と基本的には同一で、交換の階層も同じである。しかし可変長レコードが取り扱えない通信システムに適合するよう、一つの可変長メッセージは複数の 2 5 1 byte の固定長レコードに収容される。このモードの縮小モード（後述）は E I A J シンタックスルールとの互換性が高い。

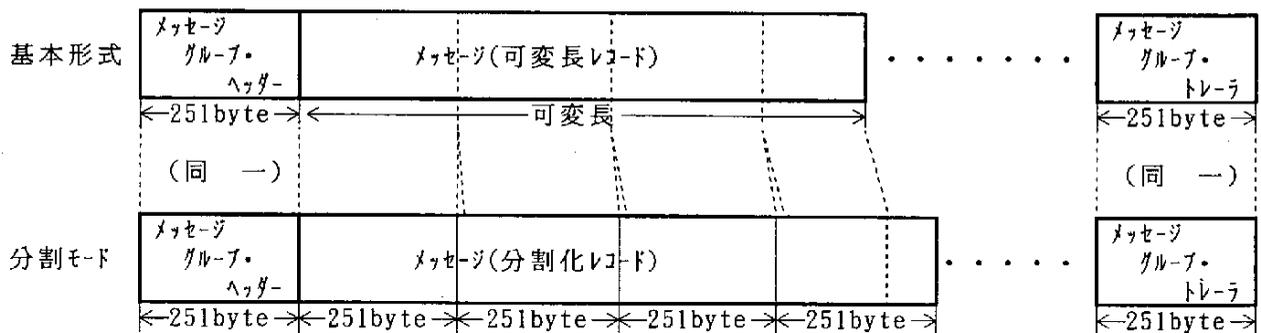


図 I - 1 基本形式と分割モードの伝送形態

1.3 縮小モードと拡張モード

C I I シンタックスルールでは、データエレメントはTFD (Transfer Form Data) 形式で、伝送される。TFDは、データエレメントに、データタグ及びデータ・レングスタグ (レングスタグ) が追加されたもので、図 I - 2 のフォーマットである。一つのメッセージは、このTFDを必要数集めて構成され、例えば、1 通の注文書/請求書を表す。

『データタグ』とは、データエレメントの意味や属性を表すIDであり、その値は、標準メッセージ及びデータエレメント・ディレクトリー (データ項目一覧表) 上で、項目No (整理番号) として表現される。データタグには縮小形式(239種のデータエレメントを表示できる)と拡張形式があり、C I I シンタックスルールでは、メッセージ長を短縮するために最適な使い分けが行われる。この使い分けをコントロールする制御子を、拡張モード指示子と呼ぶ。

TFDエリア中の拡張モード指示子 (X'F0') の左側を縮小モード領域、右側を拡張モード領域という (図 I - 3 を参照)。縮小モード領域中のTFDには、縮小モードのデータタグが使用され、拡張モード領域中のTFDには拡張モードのデータタグが使用される。TFDエリア中に拡張モード指示子がない時は、すべて縮小モード領域となり、TFDエリアはE I A J シンタックスルールとの互換性が高くなる。

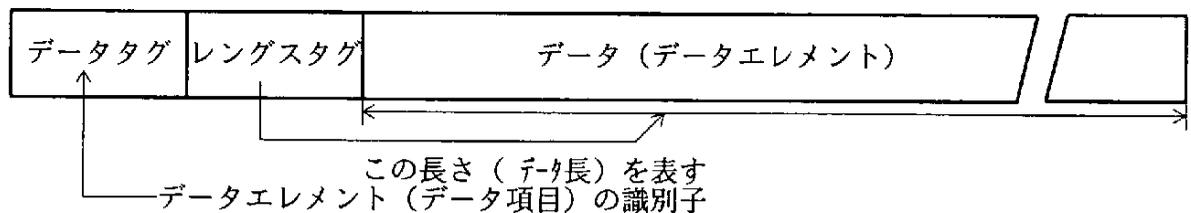


図 I - 2 T F D の 構 造

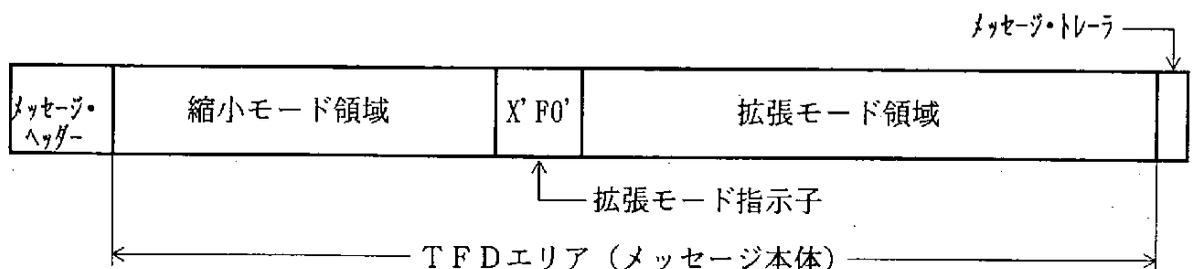


図 I - 3 縮小モードと拡張モード (メッセージ本体)

1.4 非透過モードオプション

このオプションは、J手順等の非透過モードの通信回線を使用する時に用いる。C I Iシンタックスルールは、透過モードの通信回線用に設計されているため、非透過モードの通信回線を使用すると一部のデータが通信制御のキャラクターと衝突しデータ伝送ができない。非透過モードオプションは通信制御キャラクターとの衝突を避けるための対策を追加するオプションである。但し、このオプションを設定するとデータ長が長くなる。したがって、透過モードの通信回線を用いる時は、このオプションを使用しない。

1.5 文字コード

C I Iシンタックスルールの基本仕様では、J I S - X 0 2 0 1及びJ I S - X 0 2 0 8 (J I S第1水準、第2水準漢字)を標準の文字コードとする。特に、メッセージグループ・ヘッダーとメッセージグループ・トレーラでは、英数字(ブランクと@を含む)以外の文字の使用は禁止され、文字コードもJ I S - X 0 2 0 1の使用が義務づけられる。ユーザーデータについては、ローカル協定に基づくローカル文字の使用が可能であるが、推奨できない。J I S - X 0 2 0 8は、メッセージ中のTFDのデータ部(データエレメント)でのみ使用できる。

また、TYPE - Eでは、すべてのセグメントについて、J I S - X 0 2 0 1, J I S - X 0 2 0 8の文字コードを使用しなければならない。J I S - X 0 2 0 8は、C I Iセグメント内でのみ使用できる。

1.6 トランスレーター

1.6.1 トランスレーターのアウトライン

送信用のトランスレーターは、オプションのサポートにより、様々な機能のものがある。しかし、すべての送信用のトランスレーターは、次の条件の内のどれか1つ(拡張モード)を検出するまで、縮小モードでメッセージを作成する。

① 240以上の値を持つデータ・タグの検出

② 明細番号付マルチ明細の検出

すなわち、縮小/拡張モードを自動的に設定する。一つのメッセージ中で以上の条件が成立しない時は、メッセージ全体が縮小モードで作成される。

受信用のトランスレーターも、オプションのサポートにより、様々な機能のものがある。しかしオプションサポートの範囲で、すべてのモードを自動検出する。

また、トランスレーターには文字コード変換機能が含まれ、漢字のJ I S第1、第2水準をカバーする。文字コードの変換は、必要に応じてバイパスできるオプションが設けられる。

1. 6. 2 モデル・トランスレーター

トランスレーターは、E D Iシステムにとって、最も重要な要素である。C I Iシンタックスルールの大きな特徴の一つは、トランスレーターをあらかじめ考慮して設計されていることで、本シンタックスルールが期待するトランスレーターの構造や機能があらかじめ決められている。この標準トランスレーターを、『モデル・トランスレーター』と呼ぶ。

トランスレーターの仕様を標準化する場合に、最も困難な事項は、一般的なA P I（アプリケーション・インタフェース）の標準化である。A P Iは、ユーザーのコンピュータ・システムの構成やE D Iシステムの構成に応じて様々な形態があり、通常、汎用的な仕様を定義することは難しい。そこで、業務処理システムとのデータの受渡しをすべてファイルで行い、トランスレーターへの起動／停止などの指令（コマンド）を手動で与える、実質的にA P Iのないトランスレーターを設定する。このトランスレーターは、いわば架空のトランスレーターであり、単なる設計ガイドであるが、ユーザー環境に応じた機能を追加することで（A P Iの変更等）、実用的なトランスレーターになる。従って、モデル・トランスレーターは、様々なベンダーがC I Iトランスレーターを開発する時見本となるものである。

モデル・トランスレーターを参考にしてベンダーがトランスレーターを開発することで、異なるベンダーが開発したトランスレーターの組合せでも問題なくE D Iのデータ交換が可能になる。

モデル・トランスレーターは、コンピュータの機種によらず同一であるのが原則であるが、メンフレームに実装されるトランスレーターとパーソナル・コンピュータ（パソコン）に実装されるトランスレーターとが、同一の性能／機能を保持するというわけにはいかない。そこで、E D I本来の目的であるデータ交換に問題が生じない範囲で、パソコン用トランスレーターでは、機能の簡略化が行われる。

従って、モデル・トランスレーターは4種存在する。すなわち、

① メンフレーム用モデル・トランスレーター

- a. データ送信用
- b. データ受信用

② パソコン用モデル・トランスレーター

- a. データ送信用
- b. データ受信用

である。図I-4は、メンフレーム用モデル・トランスレーター（送信用）の構成例である。

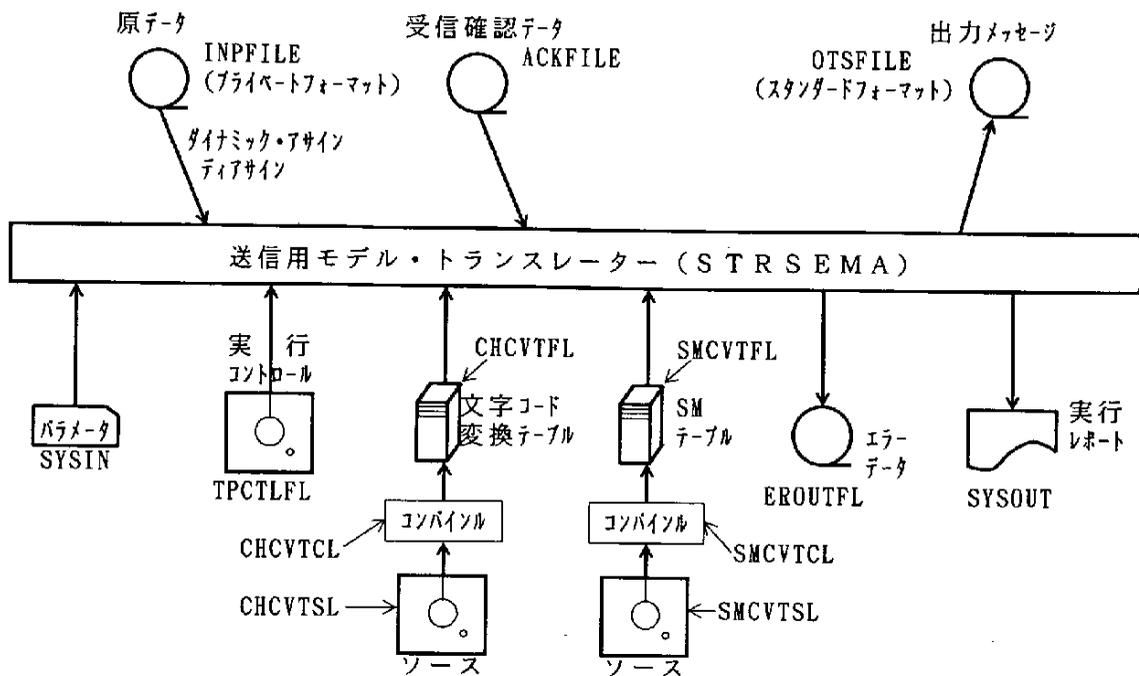
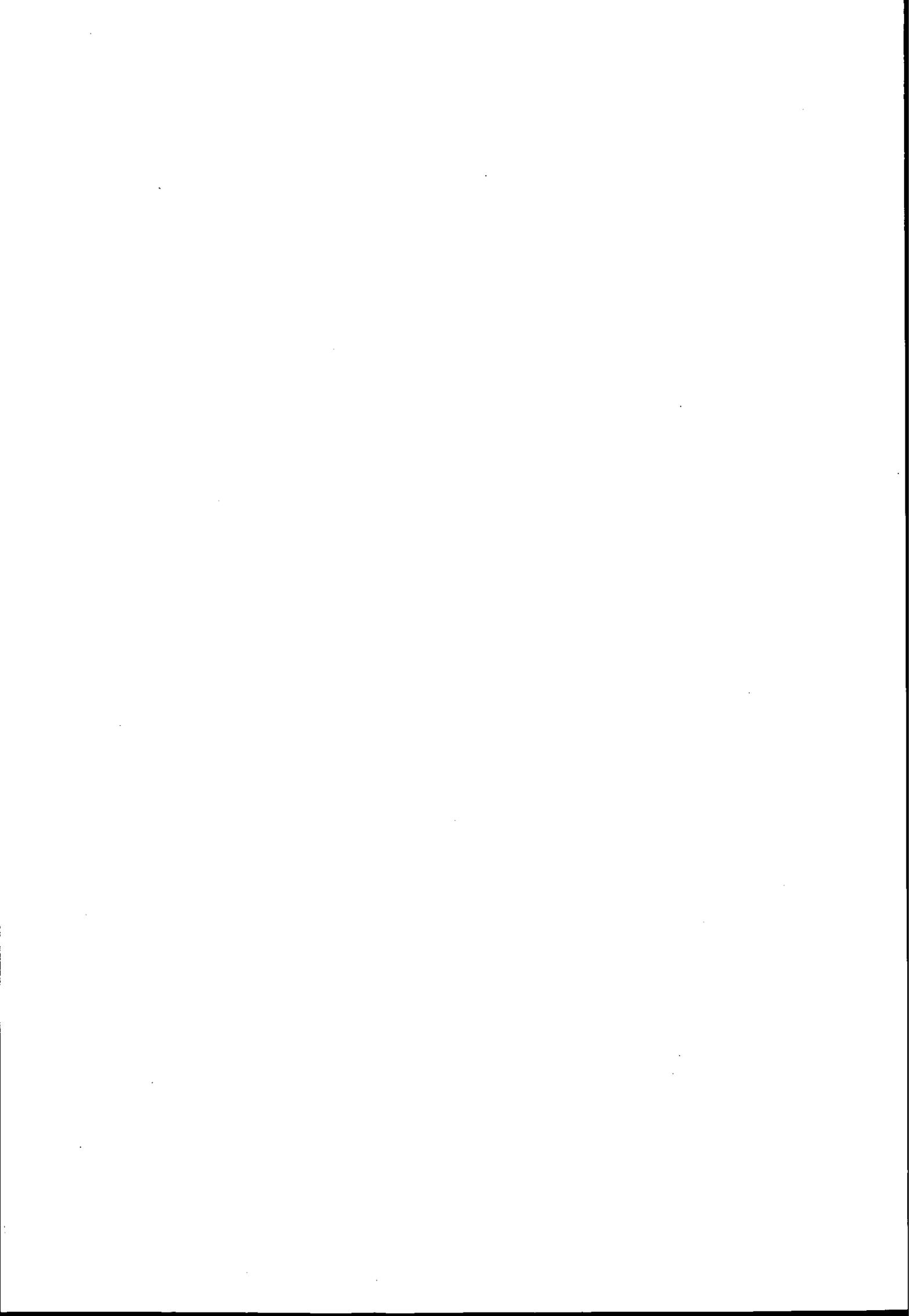


図1-4 送信用モデル・トランスレータの構成例 (メインフレーム用)

Ⅱ 詳 細

『C I I シンタックスルールの詳細』



第1章 キャラクターセットと文字コード

1.1 キャラクターセット

C I Iシンタックスルールでは、8bit文字と16bit文字（漢字）が使用できる。表1-1は、C I Iシンタックスルールで使用可能な8bit文字の一覧表である。これ以外の文字は原則として使用できないが、ローカル協定に基づくローカル文字の使用も不可能ではない。

16bit文字は、別途資料『C I Iシンタックスルール漢字コード表』を参照のこと。

1.2 文字コード

文字コードは、原則としてJ I S - X 0 2 0 1 (8bit文字、表1-1にコードを示す) とJ I S - X 0 2 0 8 (16bit文字) を用いる。しかしながら、ローカル協定により、ローカルな文字コードを用いることも不可能ではない。

表1-1 8bit文字コード表

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[¥]	^	_
6	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8																
9																
A		。	「	」	、	・	ヲ	ァ	ィ	ゥ	ェ	ォ	ャ	ュ	ョ	ッ
B	ー	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
C	タ	チ	ツ	テ	ト	ナ	ニ	ヌ	ネ	ノ	ハ	ヒ	フ	ヘ	ホ	マ
D	ミ	ム	メ	モ	ヤ	ユ	ヨ	ラ	リ	ル	レ	ロ	ワ	ン	ゝ	°
E																
F																

(注) コードは、Hex表示、縦軸は、上位4bit、横軸は、下位4bitである。
網掛けの部分は使用禁止である。

1.3 メッセージグループ・ヘッダーとメッセージグループ・トレーラのキャラクターセットと文字コード

メッセージグループ・ヘッダーでは、表1-2に示す8bit文字のみ使用可能である。また、メッセージグループ・トレーラも含めて、文字コードはJIS-X0201を用いなければならない。ローカル協定によるローカルコードの使用は許されない。

この処置は、オープンなEDIネットワークにおいて、確実なデータ交換（行き先のコントロール）を行うために必要である。

表1-2 メッセージグループ・ヘッダー及びメッセージグループ・トレーラの文字コード表 (8bit文字コード)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2	SP															
3	0	1	2	3	4	5	6	7	8	9						
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z					

(注) コードは、Hex表示、縦軸は、上位4bit、横軸は、下位4bitである。
網掛けの部分は使用禁止である。

第2章 構成要素の定義

以下の記述では、8 bitを1 byteと記述し、X '1A' は、Hex表示を表すものとする。又、誤解を避けるために、例えば、キャラクターCはC(X '43')のようにHex表示を併用する。又、図示されたフォーマット中の値は、すべてHex表示とする。さらに、特に断らないかぎり、バイナリーとは、符号無2進数とする。

2.1 交換

一つの交換とは一回の通信システム上の接続に相当し、一つ又は複数のメッセージグループを含むことができる。一つの交換に含まれるすべてのメッセージグループは、その交換では、通信システム上のある発信地から別の着信地に送信される。

2.2 メッセージグループ

一つのメッセージグループ・ヘッダーと一つのメッセージグループ・トレーラの対で構成する通信システム上のエンベロープをメッセージグループとする。このエンベロープは、複数のメッセージ、複数のバイナリーデータ及び複数の特殊情報を含むことができ、EDIで必要な交換電文を構成する。

2.3 メッセージグループ・ヘッダー

メッセージグループ・ヘッダーは、固定長データエレメントで構成される全長251 byteの固定フォーマットの文字列である。

2.4 メッセージグループ・トレーラ

メッセージグループ・トレーラは、固定長データエレメントで構成される全長251 byteの固定フォーマットのデータ列である。

2.5 メッセージ

一つのメッセージ・ヘッダー、一つのTFDエリア及びメッセージ・トレーラで構成する可変長のデータ列である。TFDエリアに、EDIユーザーのテキストデータが格納される。

2.6 高速応答I-EDI用メッセージ

短い応答時間が要求されるインタラクティブEDI (I-EDI) で、CIIシンタックスルーでは、専用のメッセージ形態が用意される。短い応答時間でなくてもよい場合は、2.2のエンベロープを用いる。

2.7 バイナリーデータ

一つのバイナリーデータ・ヘッダー、複数のバイナリーユニット及びバイナリーデータ・トレーラで構成する可変長のデータ列である。バイナリーユニットに、EDIユーザーの長大ビット列データが格納される。

2.8 特殊情報

特殊情報は、固定長データエレメントで構成される全長251byteの固定フォーマットのデータ列である。通信システムの運用上必要な情報などが格納される。

2.9 固定長データエレメント

データ長が固定されているデータエレメントである。数字や文字など3種類の属性がある。CIIシンタックスルールでは、メッセージグループ・ヘッダー、メッセージグループ・トレーラ、バイナリーデータ・ヘッダー、バイナリーデータ・トレーラ及び特殊情報などで用いられる。

2.10 可変長データエレメント

数値データの左側のゼロあるいは文字列データの右側のブランクの省略を行い、有効データのみで構成されるデータ長が可変のデータエレメントである。数字や文字など6種類の属性がある。このデータエレメントは、メッセージ内のTFDエリアの中で、TFD形式データの構成要素として用いられる。

2.11 データエレメントの記号名

固定長データエレメントの記号名は、1文字の英字と2桁の数字からなる3文字のコードで表す(例、A20など)。可変長データエレメントの記号名は、5～6桁の数字コードで表す(例、02088など)。

2.12 TFD

通信上で可変長データエレメントを扱う形式で、データタグ、レングスタグ及び可変長データエレメントでTFDを構成する。TFDには、TFD形式データ1とTFD形式データ2の2種類があり、TFD形式データ1にはEDIユーザーのデータを格納する。

2.13 データタグ

TFDの意味を表す数値で、1～2byteのビット列である。

2.14 レングスタグ

TFDを構成する可変長データエレメントの長さを表す数値で、1 byteか3 byteのビット列である。

2.15 ファイルとレコード

コンピュータ・システムにおける補助記憶装置の論理的管理単位をファイルとする。ファイルには、構造化ファイルと非構造化ファイルがある。

構造化ファイルにおける論理的分割単位をレコードとする。固定長ファイルは、長さが一定の固定長レコードの集合体であり、可変長ファイルは、長さが可変の可変長レコードの集合体である。

2.16 格納構造

メッセージグループ等のファイル/レコードへの格納方法を、格納構造と呼ぶ。

(1) 交換単位の格納構造

一つの交換は、一つのファイルに格納する。

(2) メッセージグループの格納構造

一つのメッセージグループを一つのファイルに格納してもよいし、複数のメッセージグループを一つのファイルに格納してもよい。しかし、一つのメッセージグループを複数のファイルに格納してはいけない。

第3章 データエレメントの種類と属性・長さの表記方法

データエレメントには、固定長データエレメントと可変長データエレメントがある。可変長データエレメントは、TFDの構成要素として使われる。

3.1 固定長データエレメントの種類と表記

(1) 文字列

8bit文字で構成されるデータ列である。『X(n)』（nは長さでbyte数）で表す。

(2) 数字列

8bit文字の「0～9」で構成されるデータ列である。『9(n)』（nは長さでbyte数）で表す。

(3) 2進数

長さが1～4byteの符号無2進数である。『Bin8』は1byte、『Bin16』は2byte、『Bin24』は3byteそして『Bin32』は4byteの符号無2進数表す。

3.2 可変長データエレメントの種類と表記

(1) 8bit文字列 (X属性)

最大長32767文字の8bit文字で構成される文字列データエレメントである。後述するTFD形式では、最も右側にある空白以外の文字よりもさらに右側にある空白を省略することができる。例えば、『△△ABC△△△△』と『△△ABC』は同一の意味である。

(2) 8bit単位ビット列 (B属性)

最大長32767byteの8bit単位のビット列で構成されるデータエレメントである。TFD形式では、最も右側にあるX'00'以外のビット列よりもさらに右側にあるX'00'を省略することができる。

(3) 16bit文字列 (K属性)

最大16383文字(32766byte)の16bit文字で構成される文字列データエレメントである。本文字列内に、8bit文字を含めることはできない。本文字列では、最も右側にある空白(16bit文字の空白、以下同じ)以外の文字よりもさらに右側にある空白を省略することができる。

例えば、『△△構文規則△△△』と『△△構文規則』は同一の意味である。

(4) 数値データエレメント (9属性)

最大30桁以下の数字列で、『0』～『9』までの数字だけで構成される数値データエレ

メントである。後述するTFD形式では、有効桁より左側にある『0』を省略することができる。すなわち『00123』と『123』は同一の意味とする。小数点は暗示的に示され、小数点以下の桁は、整数部に有効桁がある時は省略できない。

例えば、小数点以下が4桁である場合、『00123 (=0.0123)』は『123』と同一の意味になるが、『10000』は『1』と同一の意味にはならない。本データエレメントでは8bit文字を用いる。

(5) 数値データエレメント (N属性)

最大30桁以下の数字列で、『0』～『9』までの数字、正負符号（『+』及び『-』）及び小数点（『.』）で構成される数値データエレメントである。正負符号及び小数点は桁数に含めない。

正負符号は、常に数字列の先頭（左側）になければならない。『+』符号は省略することができる。正負符号と最も左側にある有効桁との間にある『0』を省略することができる。すなわち、『-00123』と『-123』は同一の意味とする。小数点は、『.』で明示される。小数点以下の桁については、最も右側にある有効桁よりもさらに右側にある『0』を省略することができる。従って、『12.2100』と『12.21』は同一の意味とする。小数点が省略された時は、整数となる。『.123』や『-.012』等の表現も許される。本データエレメントでは8bit文字を用いる。

尚、受信用トランスレーターでは、ブランクとゼロを同等に扱うケースがあるが、これは、ローカルな拡張オプションである。

(6) 年月日 (6桁または8桁の数字列: Y属性)

西暦年月日を表す数値である。本規格は、CIIシンタックスルール1.11をサポートしているトランスレーターで用いることが可能であるが、CIIシンタックスルール1.10をサポートするトランスレーターでも、Y属性のかわりに9属性（変換テーブル上での定義）を用いることで、西暦1951～1999年までは、正常に動作する。2000年以後は、CIIシンタックスルール1.11以後をサポートしているトランスレーターでなければ、誤動作する可能性が大きい。

- ① 6桁表現と8桁表現がある。
- ② 6桁表現は Y(6) と表記し、1951年～2050年までの年月日を表す最大長6桁のYYMMDD形式8bit数字列である（小数点記号や正負記号を含まない）。伝送時に左側のゼロを省略することができる。
- ③ 8桁表現は Y(8) と表記し、0100年～9999年までの年月日を表す最大長8桁のYYYYMMDD形式8bit数字列である（小数点記号や正負記号を含まない）。伝送時に左側のゼロを省略することができる。

尚、0000年～0050年までは2000年～2050年と解釈され、0051年～0099年までは1951年～1999年と解釈される。すなわち、0025年=2025年とされ、0091年=1991年とされる。

- ④ 属性 Y(6) と Y(8) は、数字列であり、左側のゼロを省略することができる。すなわち、TFD形式においては、Y(6) 属性のデータ Y(8) 属性のデータ共に、最大長がそれぞれ 6 byte、8 byteの可変長である。
- ⑤ Y(6) と Y(8) は、1951年～2050年までの間を表現する時は、完全に互換性があり、同一の形式として扱うことができる。

(7) 標準メッセージ上での表記

標準メッセージ上では、(1)から(6)までのデータエレメントの属性と桁数の表示を、表3-1-1のように行う。

(8) 数値データにおけるブランクと『0』

数値データ内でのブランクは原則として使用禁止であるが、受信用トランスレーターでは、たとえ数値の途中に存在しても、エラー扱いにせず『0』と見なす。

表3-1 データエレメントの標準メッセージ上での表現方法

データ・タイプ		標準メッセージ、データエレメント・ディレクトリでの表現	データ例	備考
文字 データ	8 bit文字列 (JIS-X0201)	X属性 X (n) n:最大byte長	例) X(8)の時 ABCDEFGH	長さは、byte数で表す。 E I A Jシンタックス ルールと同一である。
	8 bit単位 ビット列	B属性 B (n) n:最大byte長	例) B(3)の時 X'F256AB'	長さは、byte数で表す。 X属性と違いX'00'が サプレスされる。
	16bit文字列 (JIS-X0208)	K属性 K (n) n:最大byte長	例) K(10)の時 産業と情報	長さは、byte数で表す。 漢字文字数の2倍になる。
数値 データ	固定小数点 正数 (JIS-X0201)	9属性 9 (n), 9 (n) V (m) n:整数部の桁数 m:小数部の桁数	例) 9(5)の時 23456 例) 9(3) V(2)の時 3456 (小数点は、4と5の間)	9(5)V(0)と9(5)は同一の 意味である。 E I A Jシンタックス ルールと同一である。
	浮動小数点数 (JIS-X0201)	N属性 N (n), N (n) V (m) n:整数部の桁数 m:小数部の桁数	例) N(5)の時 -23456 例) N(4) V(2)の時 -2345.6	N(5)V(0)とN(5)は同一の 意味である。
	日付 (JIS-X0201)	Y属性 Y (n) nは6又は8	例) Y(6)の時YYMMDD日付 930331 例) Y(8)の時YYYYMMDD日付 19930331	西暦日付である。

第4章 TFD (Transfer Form Data) の詳細

TFDは通信上で可変長データエレメントを扱う形式で、縮小モードと拡張モードがある。TFDは、常にデータタグで始まる。縮小モードの時、データタグは1 byteのバイナリー数値であり、拡張モードの時は1 byteか2 byteのバイナリー数値である。拡張モードの時のデータタグの長さは、データタグの先頭の1 byteの値で決まる。

TFDは、その内容(表す意味)によって、TFD形式データ1とTFD形式データ2があり、その区別も、データタグの先頭の1 byteの値で決まる。TFD形式データ1は、ユーザーデータを格納するTFDであり、TFD形式データ2はTFDエリアのモードや繰り返し構造のコントロールを行う。そこでTFD制御子とも呼ぶ。

すなわち、データタグは、C I Iシンタックスルールのもっとも重要な要素であり、このため、C I Iシンタックスルールはタグ方式と呼ぶ。

4.1 縮小モードと拡張モード

TFDには縮小モードと拡張モードがある。メッセージ内には、TFD形式のデータを格納するTFDエリアがあり、TFDエリア内の拡張モード指示子の左側が縮小モード、右側が拡張モードになる(図4-1を参照)。TFDエリア内に拡張モード指示子がない時は、TFDエリア全体が縮小モードとなる。

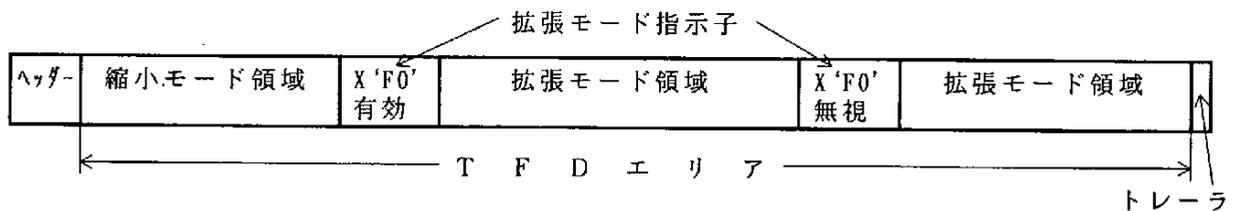


図4-1 メッセージ内のTFDエリアにおける拡張モード指示子の機能

4.2 縮小モード

4.2.1 縮小モードのTFD形式データ

(1) TFD形式データ1

図4-2に示すように、1 byteのデータタグ、1 byte又は3 byteのレンジタグ及び可変長データエレメント(EDIユーザーのデータを格納)で構成される。

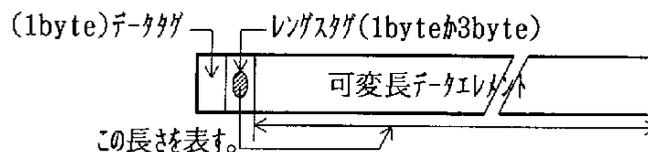


図4-2 縮小モードのTFD形式データ1

(2) TFD形式データ2

1 byteのデータタグだけで構成される。

4.2.2 縮小モードのデータタグ

縮小モードでは、データタグは1 byteの符号無2進数である。その値と意味を表4-1に示す。

0 ~ 239 (X'00' ~ X'BF')の範囲の数値でTFD形式データ1を構成するとともに、データタグの数値がデータエレメントの識別番号となる。240 ~ 255 (X'F0' ~ X'FF')の数値はTFD形式データ2を構成し、TFDエリアのコントロールを行う。

拡張モード指示子『X'F0'』で縮小モードは終わり、TFDエリアは拡張モードへ移行する。

表4-1 縮小モードのデータタグ

データタグ先頭 1byteの値	データ タグ長	フォーマット (Hex 表示)	タグの名前	意 味
X'00' ┆ X'BF'	1byte	\boxed{XX} X'00' ~ X'BF'	縮小データタグ	XXの部分で、0 ~ 239(X'00' ~ X'BF') のデータタグ番号を表す。
X'F0'	1byte	$\boxed{F0}$	拡張モード指示子	このタグの右側から拡張モードとなる。
X'F1' ┆ X'F9'	---	縮小モードでは使用禁止		
X'FA'	1byte	\boxed{FA}	マルチ明細ヘッダ	マルチ明細の開始を示す。
X'FB'	1byte	\boxed{FB}	改行マーク	マルチ明細の区切りを示す。
X'FC'	1byte	\boxed{FC}	マルチ明細トレーラ	マルチ明細の終了を示す。
X'FD'	---	縮小モードでは使用禁止		
X'FE'	1byte	\boxed{FE}	メッセージトレーラ	TFDエリアの終りとメッセージの終了を示す。
X'FF'	---	縮小モードでは使用禁止		

4.3 拡張モード

4.3.1 拡張モードのTFD形式データ

(1) TFD形式データ1

図4-3に示すように、2 byteのデータタグ、1 byte又は3 byteのレンジタグ及び可変長データエレメント（EDIユーザーのデータを格納）で構成される。

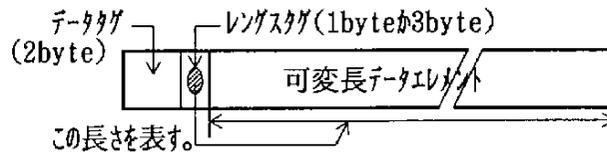


図4-3 拡張モードのTFD形式データ1

(2) TFD形式データ2

1 byteか2 byteのデータタグだけで構成される。

4.3.2 拡張モードのデータタグ

拡張モードでは、データタグは1 byteか2 byteの符号無2進数である。その値と意味を表4-2に示す。

(1) データタグの先頭1 byteが0~239 (X'00'~X'EF')の時

データタグは2 byteの符号無2進数で、0~61439 (X'0000'~X'EFFF')の範囲の数値でTFD形式データ1を構成するとともに、データタグの数値がデータエレメントの識別番号となる。

(2) データタグの先頭1 byteが240 (X'F0')の時

データタグは1 byteで、引き続き拡張モードを指示する。

(3) データタグの先頭1 byteが250 (X'FA')の時

データタグは2 byteの符号無2進数でマルチ明細ヘッダーを表し、その下位8 bit(後半の1 byte)がマルチ明細の明細番号となる。

表 4 - 2 拡張モードのデータタグ

データタグ先頭 1byteの値	データ タグ長	フォーマット (Hex 表示)	タグの名前	意 味												
X' 00' ┆ X' EF'	2byte	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>X</td><td>X</td><td>X</td><td>X</td></tr><tr><td>00</td><td>~</td><td>00</td><td>~</td></tr><tr><td>BF</td><td></td><td>FF</td><td></td></tr></table>	X	X	X	X	00	~	00	~	BF		FF		2byteデータタグ	0 ~ 61439(X' 0000' ~X' EFFF') のデータタグ番号を表す(XXXXの部分)。
X	X	X	X													
00	~	00	~													
BF		FF														
X' F0'	1byte	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>F</td><td>0</td></tr></table>	F	0	拡張モード指示子	拡張モードでは意味はない。										
F	0															
X' F1'	--	将来の拡張のためリザーブ														
X' F2'	--	将来の拡張のためリザーブ														
X' F3'	--	将来の拡張のためリザーブ														
X' F4' ┆ X' F7'	--	将来の拡張のためリザーブ														
X' F8'	1byte	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>F</td><td>8</td></tr></table>	F	8	バイトリデータ 補正子	特注) Zレベルオプション この2つのデータタグは、次期リリースで廃止される予定										
F	8															
X' F9'	2byte	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>F</td><td>9</td><td>X</td><td>X</td></tr></table>	F	9	X	X	インターナルデータセグメント									
F	9	X	X													
X' FA'	2byte	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>F</td><td>A</td><td>X</td><td>X</td></tr></table>	F	A	X	X	マルチ明細ヘッダ-A (1byte識別子)	XXで1文字の明細番号を表す。 (文字コードは JIS-X0201 とする。)								
F	A	X	X													
X' FB'	1byte	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>F</td><td>B</td></tr></table>	F	B	改行マーク	マルチ明細の区切りを示す。										
F	B															
X' FC'	1byte	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>F</td><td>C</td></tr></table>	F	C	マルチ明細トレーラ	マルチ明細の終りを示す。										
F	C															
X' FD'	--	将来の拡張のためリザーブ														
X' FE'	1byte	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>F</td><td>E</td></tr></table>	F	E	メッセージトレーラ	TFDエリアの終りとメッセージの終りを示す。										
F	E															
X' FF'	--	将来の拡張のためリザーブ														

4.4 レングスタグ (データ長)

レングスタグは、データタグの直後に位置し、直後の可変長データエレメントのデータ長をbyte数で表す。1～239byteまでの長さは、1byteの符号無バイナリー数値で表し、240～32767byteまでの長さは、X'F2'と2byteの符号無バイナリー数値を並べた3byteで表す(表4-3を参照)。

表4-3 レングスタグの構造

レングスタグ先頭 1byteの値	レングスタグ長	フォーマット	表す長さ (Hex表示)	表す長さ (Dec表示)	備 考
X'00' } X'EF'	1byte	XX	X'0000' } X'00EF'	00000 } 00239	8bitの2進数(XXの部分)で表す。
X'F2'	3byte	F,2,XX,XX,XX	X'00F0' } X'7FFF'	00240 } 32767	16bitの2進数(XXXXの部分)で表す。

4.5 TFDエリア

一つ又は複数のTFD形式データ1(データエレメント)と一つ又は複数のTFD形式データ2(TFD制御子)で、TFDエリアを構成する。TFDエリアは可変長データエレメントの集合体である。TFDエリア内では、TFDは隙間なく並べられなければならない。

TFDエリアの最後部(TFDエリアの終わり)には、TFD-END指示制御子『X'FE』を置かなければならない。

TFDエリア内では、任意のTFD形式データ1のTFDが省略可能である。TFD形式データ2のTFDは、条件により省略可能である。

第 5 章 メッセージグループ・ヘッダーの構造

メッセージグループ・ヘッダーは、メッセージグループの開始を表すヘッダーで、図 5-1 及び表 5-1 で示す 2 5 1 byte 固定長の固定フォーマットである。メッセージグループ・ヘッダーは、一つの 2 5 1 byte のレコードに格納される。

5.1 拡張モード (C 2 2)

メッセージグループに含まれるすべてのメッセージを縮小モードにする時、X' 20' か X' 53' をセットする。これ以外のケースでは、X' 45' をセットする。

5.2 トータル項目 N_a

拡張モード (C 2 2) が X' 45' (拡張を意味) の時、第 2 トータル項目 $N_a - 1, 2$ (C 2 7, C 2 8) を用い、そうでない時、第 1 トータル項目 $N_a - 1, 2$ (C 1 5, C 1 6) を用いる。

C 2 2 が縮小モードの時は、第 2 トータル項目 $N_a - 1, 2$ が無視され、C 2 2 が拡張モードの時は、第 1 トータル項目 $N_a - 1, 2$ が無視されなければならない。

5.3 分割モード (C 2 3)

分割モード (C 2 3) が X' 20' か X' 4D' の時、分割モードと呼び、メッセージグループのすべてが 2 5 1 byte の固定長レコードに格納される。X' 53' の時、通常モード (非分割モードとも呼ぶ) と呼び、メッセージグループは可変長レコードに格納される。

5.4 非透過モード (C 2 6)

非透過モードは、Z レベルオプションである。このオプションを使う時は、X' 4D' をセットする。

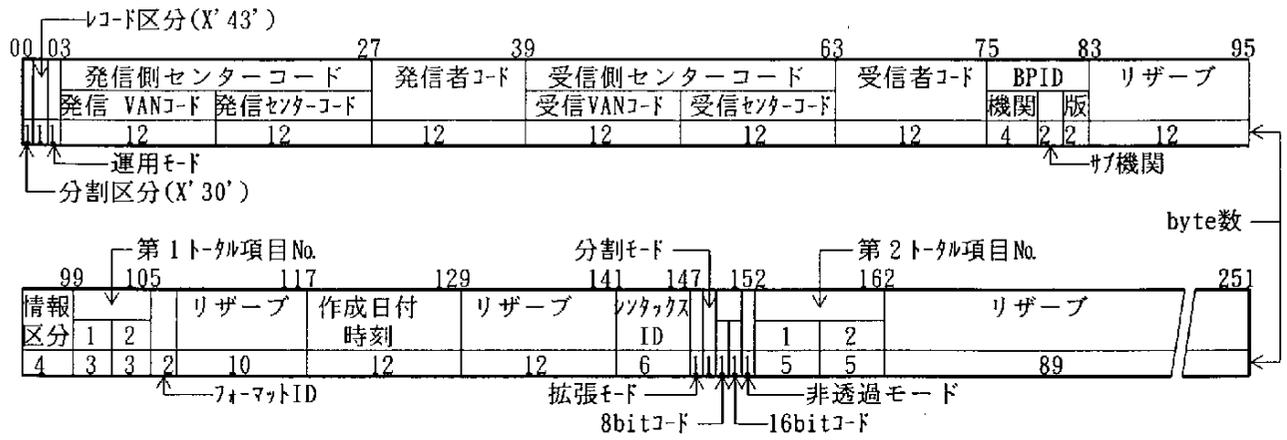


図5-1 メッセージグループ・ヘッダーのフォーマット

表5-1 メッセージグループ・ヘッダーのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	X'30' 固定, メッセージグループ・ヘッダー・カレラを示す。
C02	X(1)	ロード区分	X'43' 固定, メッセージグループ・ヘッダーであることを示す。
C03	X(1)	運用モード	X'31'の時テスト電文、X'30'又はX'20'の時通常電文を表す。
C04	X(12)	発信VANコード	英数字12文字のコード(文字コード JIS-X0201)。
C05	X(12)	発信センターコード	" " (" ")。
C06	X(12)	発信者コード	" " (" ")。
C07	X(12)	受信VANコード	" " (" ")。
C08	X(12)	受信センターコード	" " (" ")。
C09	X(12)	受信者コード	" " (" ")。
C10	X(4)	BPID機関	英数字4文字のコード(" ")で標準メッセージ管理機関を表す。
C11	X(2)	BPIDサブ機関	英数字2文字のコード(" ")で標準メッセージ管理機関が管理するコード。
C12	X(2)	BPID版	" " (" ") " " " "。
F13	X(12)	リザーブ	将来の拡張のためリザーブ (allX'20' をセット)。
C14	X(4)	情報区分コード	英数字4文字のコード(文字コード JIS-X0201)で、情報の種類を表す。
C15	9(3)	第1トータル項目No.-1	「拡張モード(C22)」がX'20'かX'53'の時、ハッシュ・カレラに使う9属性データ項目のデータタグ番号をセット(文字コード JIS-X0201)。allX'20'をセットするとハッシュ・カレラをしない。
C16	9(3)	第1トータル項目No.-2	同上の2つめのデータタグ番号をセットした。
C17	X(2)	フォーマットID	通常モード(非分割モード)のメッセージグループの時X'3130', 分割モードのメッセージグループの時X'3131', 受信確認電文の時X'3230'をセット。
C18	X(10)	リザーブ	将来の拡張のためリザーブ (allX'20' をセット)。
C19	X(12)	作成日付時刻	メッセージグループの作成日付時刻 YYMMDDHHMMSS(文字コード JIS-X0201)。
F20	X(12)	リザーブ	将来の拡張のためリザーブ (allX'20' をセット)。
C21	X(6)	シンタックスIDバージョン	シンタックス・カレラ管理機関とバージョン(文字コード JIS-X0201)を表す。
C22	X(1)	拡張モード	EIAJシンタックス・カレラ互換に於ける時、X'20'かX'53'をセット、それ以外の時X'45'をセット。
C23	X(1)	分割モード	通常モードの時X'53'をセット、分割モードの時X'20'かX'4D'をセット。
C24	X(1)	文字コード8bit	X属性、9属性、N属性の文字コードがJIS-X0201の時X'20'かX'53'をセット。その他の時X'50'をセット。
C25	X(1)	文字コード16bit	K属性の文字コードがJIS-X0208, JIS-X0212の時X'20'かX'53'をセット。K属性の文字コードがJIS-X0221の時X'55'をセット、その他の時X'50'をセット。
C26	X(1)	非透過モード	X'20'かX'53' (透過モード)をセット。
C27	9(5)	第2トータル項目No.-1	「拡張モード(C22)」が45'の時、ハッシュ・カレラに使う9属性かN属性データ項目のデータタグ番号をセット(文字コード JIS-X0201)。allX'20'をセットするとハッシュ・カレラをしない。
C28	9(5)	第2トータル項目No.-2	同上の2つめのデータタグ番号をセットした。
F29	X(89)	リザーブ	将来の拡張のためリザーブ (allX'20' をセット)。

第6章 メッセージの構造

メッセージは、メッセージヘッダー、TFDエリアそしてメッセージトレーラで構成され、メッセージヘッダーには、AタイプとBタイプ（拡張型）がある。この状況を図6-1、表6-1及び表6-2に示す。

6.1 メッセージヘッダー

図6-1に示すように、AタイプとBタイプがある。メッセージ長(D04)がX'0009'~X'7FFF'の時、Aタイプを示し、メッセージ長(D04)がX'8080'の時、Bタイプを示す。

① Aタイプ・メッセージヘッダー

Aタイプは、TFDエリアの長さが32758byte以下（メッセージ長(D04)では32767byte以下）の時用いることができる。メッセージ長(D04:16bit2進数)がメッセージ長を表す。

② Bタイプ・メッセージヘッダー

Bタイプは、TFDエリアの長さが32759byte以上（拡張メッセージ長(D06)の値では32776byte以上）の時用いる。拡張メッセージ長(D06:十進数)がメッセージ長を表す。しかし、Bタイプ・メッセージヘッダーは、TFDエリアの長さが32758byte以下の時でも用いることができる。

すなわち、TFDエリア長が32758byte以下の時は、AタイプでもBタイプでも使うことができるが、TFDエリア長が32759byte以上の時は、Bタイプを使わなければならない。

メッセージ長(D04,D06)には、メッセージトレーラの長さが含まれないので、注意する。

6.2 TFDエリア (Transfer Form Data Area)の詳細

TFDエリアは、TFD形式データ1及びTFD形式データ2（それぞれ第4章参照）を隙間なく並べることによって構成する可変長のエリアである。

TFDエリアは常に縮小モードで始まり、拡張モード指示子(X'F0')が出現した時点で拡張モードに切り換る。この拡張モードは、同一メッセージグループ内の次のメッセージのTFDエリアに影響しない。拡張モードで終わったメッセージの次のメッセージのTFDエリアは、再び縮小モードで始まる。

拡張モード指示子がTFDエリアにない時は、TFDエリア全体が縮小モードである。メッセージグループ・ヘッダーの「拡張モード(C22)」を、「X'20'」か「X'53'」にセットした時は、メッセージグループ内のすべてのメッセージのTFDエリア全体を縮小モードにしななければならない。逆に、「拡張モード(C22)」を、「X'45'」をセットした

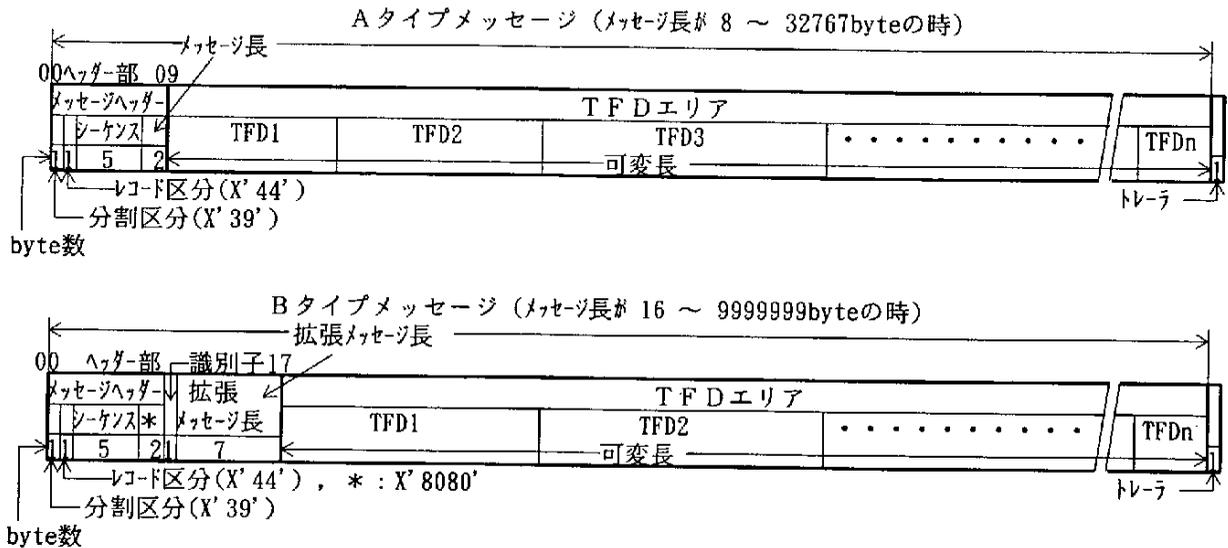


図 6-1 メッセージのフォーマット

表 6-1 Aタイプメッセージヘッダーを持つメッセージのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	非分割の時 X' 39'。但し分割レコードの場合は、X' 31'。最終分割の時 X' 39'。
C02	X(1)	レコード区分	X' 44' 固定、メッセージであることをしめす。
D03	9(5)	シーケンス番号	同一メッセージグループ内のメッセージの順番を表す。メッセージグループヘッダー直後のメッセージのシーケンス番号をとし、以後1、づつ増加する(文字コード JIS-X0201)。
D04	Bin16	メッセージ長	メッセージ長が9~32767byteの時用いる。 分割区分の左側からTFDエリアの右側までの長さ(メッセージ長)を2進数で表す。
D07	可変	TFDエリア	最大 32758byteまでの可変長のTFD格納エリア。
D08	X(1)	メッセージトレー	X' FE' 固定、メッセージの終わりを示す。

表 6-2 Bタイプメッセージヘッダーを持つメッセージのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	非分割の時 X' 39'。但し分割レコードの場合は、X' 31'。最終分割の時 X' 39'。
C02	X(1)	レコード区分	X' 44' 固定、メッセージであることをしめす。
D03	9(5)	シーケンス番号	同一メッセージグループ内のメッセージの順番を表す。メッセージグループヘッダー直後のメッセージのシーケンス番号をとし、以後1、づつ増加する(文字コード JIS-X0201)。
D04	Bin16	メッセージ長	X' 8080' 固定、拡張メッセージ長使用を表す。メッセージ長が32776~9999999byteの時用いる。
D05	X(1)	識別子	X' F7' 固定 (但し、メッセージ長が、9~32767byteの時も用いることができる。)
D06	9(7)	拡張メッセージ長	分割区分の左側からTFDエリアの右側までの長さ(メッセージ長)を7桁の数値で表す。 (文字コード JIS-X0201)。
D07	可変	TFDエリア	最大 9999999byteまでの可変長のTFD格納エリア。
D08	X(1)	メッセージトレー	X' FE' 固定、メッセージの終わりを示す。

時は、縮小・拡張どちらのモードも許される。通常のC I Iトランスレーター（送信用）では、E I A J互換にする時のみ「拡張モード（C22）」に、「X'20'」（縮小の意味）をセットする。

T F D形式データは、常にデータタグで始まる。縮小モードの時、データタグは1 byteの2進数値であり、拡張モードの時は、1 byteか2 byteの2進数値である。拡張モード時のデータタグの長さやT F D形式データ1か2かの区別は、データタグの先頭の1 byteの値で決まる。

したがって、T F Dエリアを参照する時は、必ずエリアの先頭（左端）から参照しなければならない。T F Dエリアの左端には、必ず、データタグが存在する。

6.3 T F Dの省略

一つのメッセージに含めなければならないT F D（データ項目）は、標準メッセージによって規定される。しかしながら、実際の業務で必要とするT F D（データ項目）が、標準メッセージによって規定されるT F D（データ項目）よりも少ないことがある。このような時、不要なT F D（データ項目）を省略することができる。T F D形式データ1（一般データ項目）の省略については何らの制限もない。さらに、数値がゼロの数値T F Dや内容がallブランクの文字列T F Dは、一般的に省略可能である。

T F D形式データ2（T F D制御子）については、一般的に省略できない。しかしながら、マルチ明細制御子（マルチ明細ヘッダー、改行マーク及びマルチ明細トレーラ）は、条件により省略できることがある（6.5を参照）。

6.4 T F D形式データ2（T F D制御子）の機能

T F D形式データ2は、T F D制御子と呼ばれ、T F Dエリアのコントロールを行う。

(1) 拡張モード指示子（X'F0' = 2 4 0）

T F Dエリアに、この制御タグが出現すると、それより右側のT F Dエリアは、拡張モードになり、拡張モードのT F Dが存在すると見なす。拡張モード指示子が同一のT F Dエリアに複数個存在する場合は一番左側の拡張モード指示子が有効で、それ以外は無視する（図6-2を参照）。

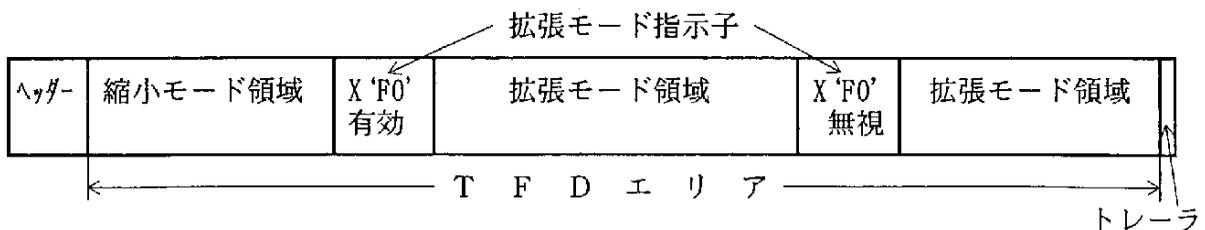


図6-2 T F Dエリアにおける拡張モード指示子の機能

(2) マルチ明細ヘッダー (X 'FA' = 2 5 0)

このTFDは、マルチ明細ヘッダーを表し、マルチ明細の始まりを示す。

拡張モードの時、マルチ明細ヘッダーは1byteの制御データを保持し(図6-3)、マルチ明細の番号を明示する。一つの標準メッセージ内に2つ以上のマルチ明細がある時、この番号を用いて、明示的に、それぞれのマルチ明細を区分する。マルチ明細の明細番号は、通常、1 (X '31'), 2 (X '32'), ..., 9 (X '39'), A (X '41'), B (X '42'), ..., Z (X '5A') (1文字で表す) までの35種である。

尚、縮小モードの時は、マルチ明細の番号の明示はない。又、縮小モードの時にマルチ明細が始まり(縮小モードマルチ明細)、その明細が終わる前に拡張モードへ移行した時も、このマルチ明細は縮小モードマルチ明細である。

通常の送信用トランスレータは、明示的明細番号付マルチ明細をTFDエリアに組み込む時、縮小モードであれば、拡張モードに移行させる。

(3) マルチ明細改行マーク (X 'FB' = 2 5 1)

マルチ明細の1行の終わりを示す(6.5を参照)。

(4) マルチ明細トレーラ (X 'FC' = 2 5 2)

マルチ明細の終わりを示す(6.5を参照)。

(5) トレーラ (メッセージ・トレーラ) (X 'FE' = 2 5 4)

TFDエリアの終わりを示す。メッセージの終わりでもある。メッセージ・トレーラを兼ねる。

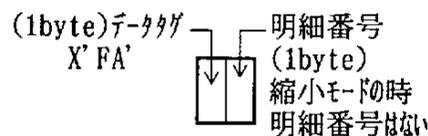


図6-3 マルチ明細ヘッダーのフォーマット

6.5 データの繰り返し（繰り返しルール：表形式データ）

同一メッセージ内のTFDエリアでは、同一のデータタグ値を持つTFD形式データ1のTFD（データ項目）を2個以上含めることは原則としてできない。同一のデータタグ値を持つTFD形式データ1を複数個含めるためには、繰り返しルールを用いる。

(1) 単独項目の暗示的繰り返し

同一のデータタグ値のTFDを単純に複数個並べることで、同一メッセージ上で、同一項目を複数個伝送できる。この場合、同一項目（TFD）を隙間なく隣合うように並べ、送信側と受信側で、メッセージ内での項目の順番の管理を厳密に行わなければならない。この方式は運用ミスを起こし易く推奨できない。なるべくセグメント化する。

(2) マルチ明細

いくつかのTFD（データタグ値は、それぞれ異なる）を集めてセグメントとし、セグメント単位で明示的に繰り返しを行うもので、表形式データの伝送に適用する。セグメントは一つのTFDで構成してもよい。

図6-4のように、最初のセグメントの先頭にマルチ明細ヘッダー、セグメントとセグメントの間に、改行マーク、最後のセグメントの終わりに、マルチ明細トレーラをセットする。また、必要に応じて各セグメント内の任意のTFDを省略できる。同一セグメント内のすべてのTFDが省略された時は、改行マークだけが残される（図6-5）。

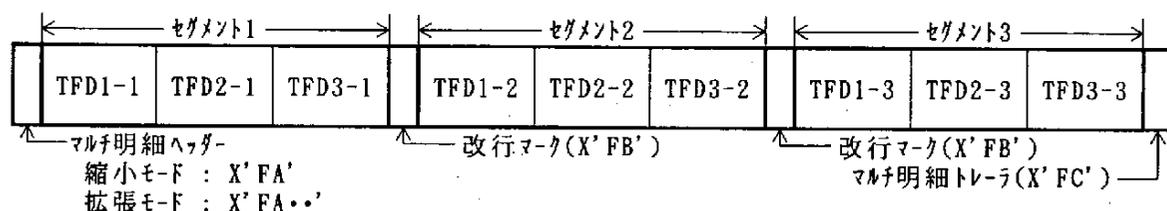


図6-4 マルチ明細

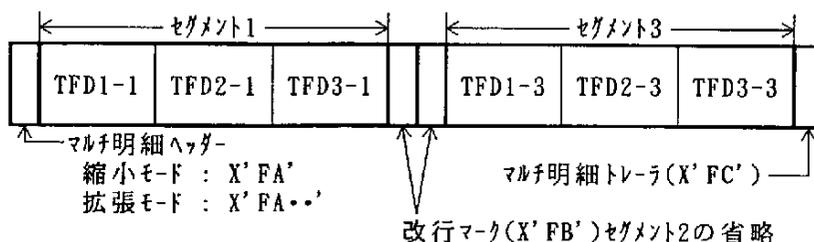


図6-5 マルチ明細（図6-4のTFD1-2、TFD2-2、TFD3-2が省略された時）

(3) マルチ明細の複数化

縮小モードでは、暗示的にマルチ明細の複数化ができる。この場合、図6-6に示すように、左側のマルチ明細の省略を行う時は、少なくともマルチ明細ヘッダーとマルチ明細トレーラは残すことが望ましい。

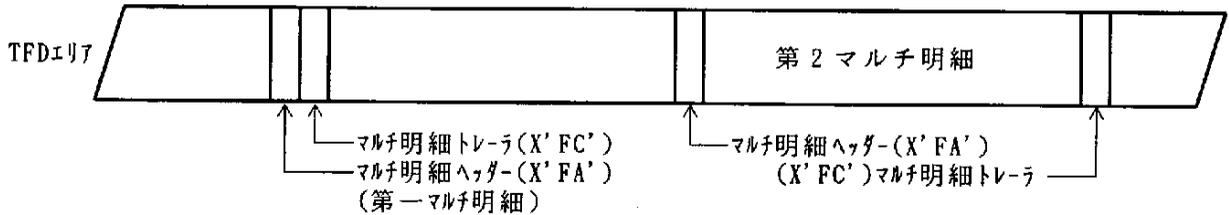


図6-6 縮小モードにおけるマルチ明細の複数化

拡張モードでは、マルチ明細の番号によって明示的にマルチ明細の複数化を行う。この場合、任意のマルチ明細が省略可能である(図6-7参照)。

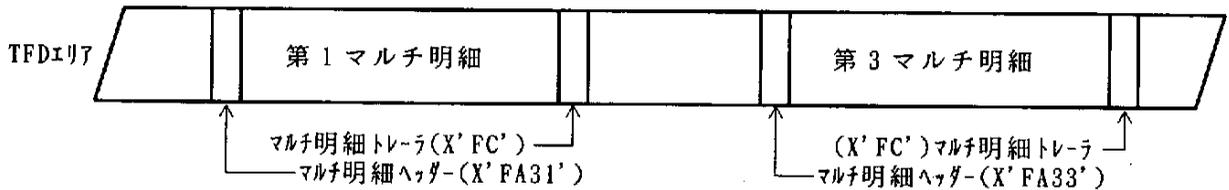


図6-7 拡張モードにおけるマルチ明細の複数化(第2マルチ明細の省略)

(4) マルチ明細のネスト化(拡張モード)

拡張モードでは、マルチ明細のネスト化ができる。図6-8に示すように、2次元以上の表型式データで用いる。各レベルでのマルチ明細ヘッダーの明細番号は、異なる番号を用いる。また、任意のTFD、セグメントおよびマルチ明細の全体を省略できる。しかしながら、レベル2のセグメントが省略されない場合、レベル1のセグメントのマルチ明細ヘッダーとマルチ明細トレーラは省略することも可能であるが、このことによって誤動作する受信用トランスレータもある。一般的に $n < m$ の時、レベル m のセグメントが省略されない場合、

レベルnのマルチ明細ヘッダーとマルチ明細トレーラを省略する時は、注意を要する。

縮小モードでのマルチ明細のネスト化は禁止する。多くの送信用トランスレーターは、縮小モードでマルチ明細のネスト化を検出した場合、自動的にエラー処理を行う。

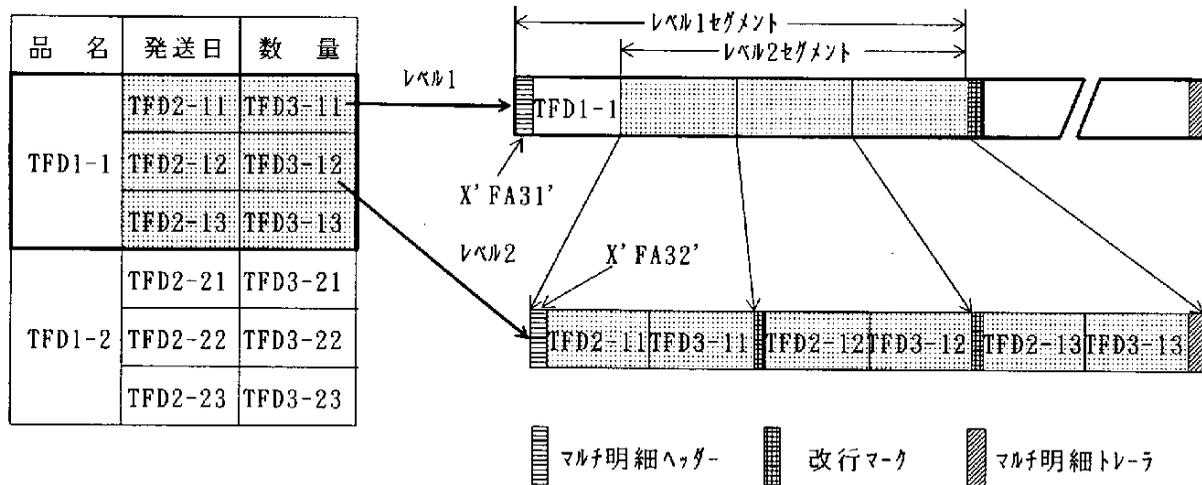


図 6 - 8 マルチ明細のネスティング

6.6 縮小/拡張モードとマルチ明細の関係

(1) マルチ明細中での拡張モードへの移行

縮小モード中の明細番号無マルチ明細の途中で拡張モードへ移行した時でも、そのマルチ明細は縮小モードのマルチ明細であり、明細番号はない。

(2) 拡張モード中の明細番号無マルチ明細

拡張モード中では、明細番号無マルチ明細ヘッダーの使用は許されない。従って、拡張モード中に明細番号無マルチ明細ヘッダーが出現した時（トランスレーターの変換テーブルでこのような指定のケースがある）は、エラー処理にしなければならない。

6.7 メッセージ・トレーラ

TFDエリアの終わりを示すTFD制御子（1byteの数値（254=X'FE'））で、一つのメッセージの終わりを示す。メッセージ・トレーラを兼ねる。

6.8 メッセージ格納構造 (分割モード)

通常、一つの変長メッセージを一つの変長レコードに格納する。しかし、固定長形式のレコード型ファイルにメッセージを格納するためには、変長メッセージを適当に分割しなければならない。このためにあるのが、分割モードである。分割モードでは一つの変長メッセージを250byteずつに分割して、1個あるいはn個の251byteの固定長レコードに収容する。レコードの分割は、表6-3のように行われる。

表6-3 分割モードにおけるメッセージの分割

メッセージ長(D04) 又は拡張メッセージ長(D06) の値	レコード数
1 ~ 250	1レコード
251 ~ 500	2レコード
501 ~ 750	3レコード
⋮	⋮
$250 * (n - 1) + 1 \sim 250 * n$	nレコード

最初のレコード用のみ251byteのデータを切り取り、分割区分をX'31'とする。以後、250byteずつ切り取ったデータの左先頭に分割区分(1byte)を追加し、251byteとしたデータを分割化レコードとする。分割区分は、レコードの順番チェックと最終レコードの検出のために、次のように付番する。

X'31' ⇒ X'32' ⇒ …… ⇒ X'38' ⇒ X'31' ⇒ X'32' ⇒ …… ⇒ X'35' ⇒ X'39' (最終レコード)

つまり、X'31', X'32' ……の順でX'38'まで行き、ふたたびX'31'からX'38'まで番号を付ける。X'31' ~ X'38'までの繰り返しである。そして、最終レコードは、常にX'39'を付ける。図6-9は、4つに分割された例を示す。最終レコードの右余白には、ブランク(X'20')を満たす。

尚、メッセージ長(D04又はD06)の値が250以下の時は、1レコードに格納され、分割区分は、X'39'である。

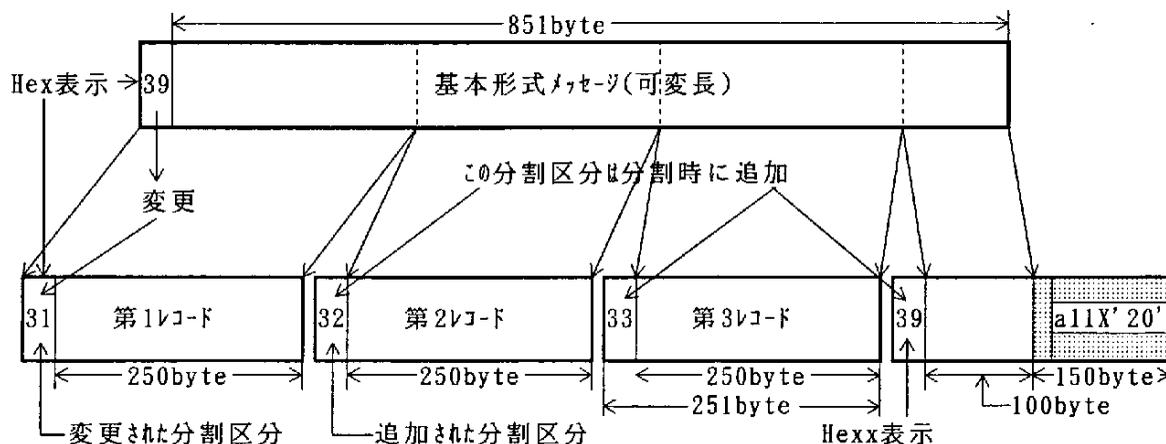


図6-9 分割化固定長レコード (4分割の例)

第7章 バイナリーデータの構造

本仕様はBレベルオプションである。バイナリーデータとは、EDIユーザーの長大BIT列データを格納するものである。CAD/CAMデータや画像データがこれに相当する。

7.1 バイナリーデータの基本構造

一つのバイナリーデータは一つのバイナリーデータ・ヘッダー、複数のバイナリーユニットそして、一つのバイナリーデータ・トレーラで構成される(図7-1参照)。バイナリーデータ・ヘッダーとバイナリーデータ・トレーラは、251byteの固定フォーマットであるが、バイナリーユニットは、メッセージグループ・ヘッダーの分割モード(C23)の値に対応し、X'53'(通常モード)の時、32001byteの固定フォーマットとなり、X'20'かX'4D'(分割モード)の時、251byteの固定フォーマットとなる。

7.2 バイナリーデータ・ヘッダー

バイナリーデータ・ヘッダーは、251byteの固定フォーマットのデータで、メッセージグループ内において一つのバイナリーデータの始まりを表す(図7-2及び表7-1参照)。

① リンク番号(H04)

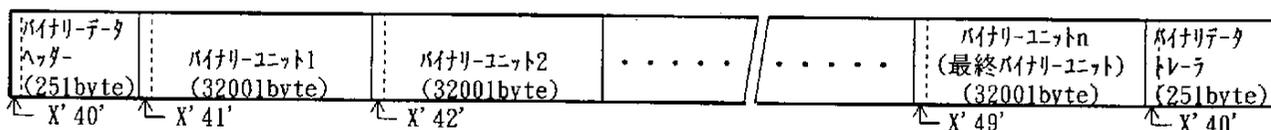
同一のメッセージグループ内で、バイナリーデータとメッセージの論理的関係を示す。同一の番号を持つバイナリーデータとメッセージが論理的関係にあることを表す。

尚、メッセージ内では、タグ番号がX'BF00'(=61184)のTFDが、このリンク番号の項目になる。リンク番号の値そのものは運用で決めるが、通常必要な範囲においてユニークでなければならない。

② ファイル識別名(H05), フォーマット識別名(H06), 圧縮識別名(H07)

これらの識別名は、ある範囲でユニークでなければならないが、識別名そのものは、運用で決める。

通常モード（非分割モード）時のバイナリーデータの構造



分割モード時のバイナリーデータの構造

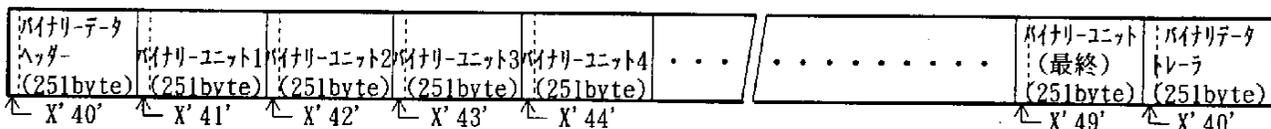


図7-1 バイナリーデータの構造

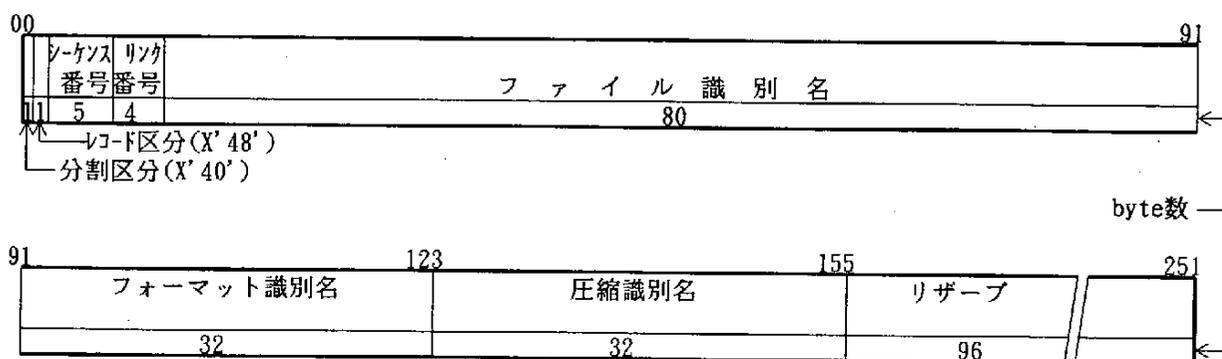


図7-2 バイナリーデータ・ヘッダーのフォーマット

表7-1 バイナリーデータ・ヘッダーのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	X'40' 固定, バイナリーデータ・ヘッダ・トレーラを示す。
C02	X(1)	コード区分	X'48' 固定, バイナリーデータ・ヘッダであることを示す。
D03	9(5)	シーケンス番号	メッセージヘッダのシーケンス番号と同一の意味を持つ。メッセージヘッダとバイナリーデータを区別でき、1から17の順に付番する(文字コード JIS-X0201)。
H04	9(4)	リンク番号	同一のメッセージグループ内で、メッセージとバイナリーデータとの論理的関係を表す。同一のリンク番号を持つメッセージとバイナリーデータが論理的関係にあることを表す。 文字コードは、メッセージグループ・ヘッダの文字コード8bit(C24)の指示に従う。
H05	X(80)	ファイル識別名	バイナリーデータをローカルに保存する時のファイル名で、左詰めにセットし余白は空白で満たす。 文字コードは、メッセージグループ・ヘッダの文字コード8bit(C24)の指示に従う。
H06	X(32)	フォーマット識別名	バイナリーデータそのもののフォーマットを表す識別名で、左詰めにセットし余白は空白で満たす。 文字コードは、メッセージグループ・ヘッダの文字コード8bit(C24)の指示に従う。
H07	X(32)	圧縮識別名	バイナリーデータそのもののデータ圧縮方法を表す識別名で、左詰めにセットし余白は空白で満たす。 文字コードは、メッセージグループ・ヘッダの文字コード8bit(C24)の指示に従う。
F08	X(96)	リザーブ	将来の拡張のためリザーブ (allX'20' をセット)。

7.3 バイナリーユニット

バイナリーユニットは、CAD/CAMデータ、イメージデータそのもので、byte単位のビット列とする。251bitのような8の倍数でないビット列は扱わない。バイナリーユニットのフォーマットは、通常モード時と分割モード時で異なっており、通常モード時は32001byte、分割モード時は251byteの固定長フォーマットである。EDIユーザーのビット列32000byte、250byteを、それぞれ収容する。但し、通常モードにおいて32000byteあるいは分割モードにおいて250byteを越えるビット列を格納する場合は、格納構造の規定にしたがって分割される（図7-3及び表7-2参照）。

尚、ビット列データのフォーマット、すなわちCAD/CAMデータ、イメージデータそのもののフォーマットについては、本シンタックスルールでは規定されない。

7.4 バイナリーデータ・トレーラ

バイナリーデータ・トレーラは、一つのバイナリーデータの終わりを表す（図7-4及び表7-3参照）。

① リンク番号(H04)

バイナリーデータ・ヘッダーの「リンク番号」と同一の番号とする。

② 最終ユニット有効長(T05)

バイナリーデータは、格納構造にしたがって複数の固定長のバイナリーユニットに分割して格納される。この場合、一般的に最後のバイナリーユニットに端数のデータが発生する。有効データ長は、その端数の長さを表している。数値の1が1byteに相当し、以下、2が2byteに、nがnbyteに相当する。

③ 全ユニット数(T06)

バイナリーデータは、格納構造にしたがって複数の固定長のバイナリーユニットに分割して格納する。全ユニット数は、そのユニット数を表す（バイナリーデータ・ヘッダーとバイナリーデータ・トレーラを含む）。

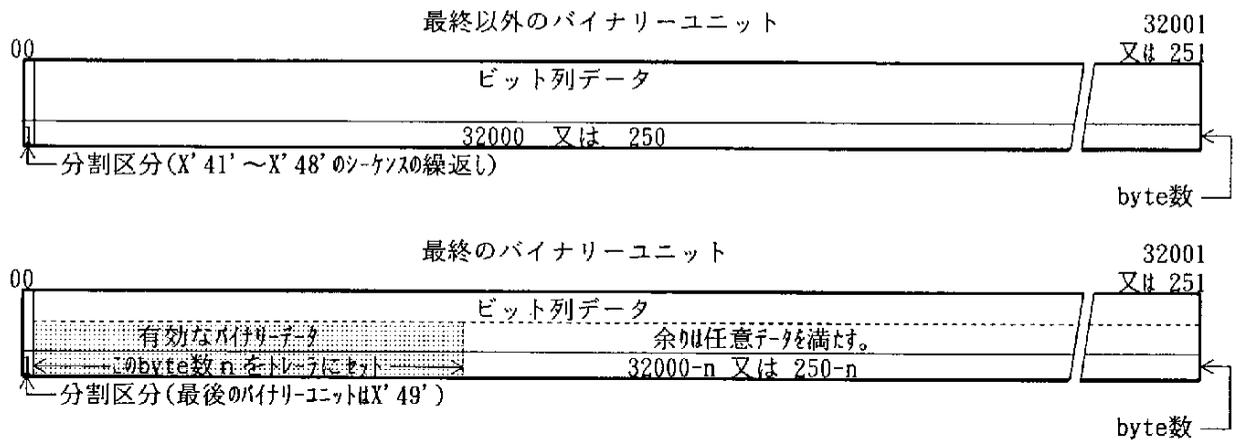


図7-3 バイナリユニットのフォーマット

表7-2 バイナリユニットのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	バイナリユニットを示す。X' 41' ~ X' 48' のシーケンスの繰り返し、最後のバイナリユニットはX' 49'。
D10	B(32000) 又は B(250)	バイナリデータ	長大ビット列データを格納する。 通常モードでは、長大ビット列データを32000byteの単純に分割して格納する。 分割モードでは、長大ビット列データを250byteの単純に分割して格納する。 最終のバイナリユニット(分割区分(C01)=X' 49')の時は、端数のビット列データを左詰めにセットする。この有効なビット列データの長さ(byte数)をバイナリデータ・トレーラの最終ユニット有効長(T05)にセットする。余白は、任意データで満たす。

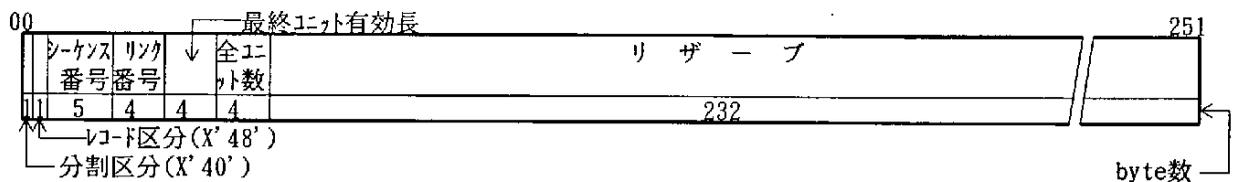


図7-4 バイナリデータ・トレーラのフォーマット

表7-3 バイナリデータ・トレーラのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	X' 40' 固定、バイナリデータ・ヘッダ・トレーラを示す。
C02	X(1)	レコード区分	X' 54' 固定、バイナリデータ・トレーラであることを示す。
D03	9(5)	シーケンス番号	バイナリデータ・ヘッダのシーケンス番号と同一の値 (文字コード JIS-X0201)。
H04	9(4)	リンク番号	同一のメッセージグループ内で、メッセージとバイナリデータとの論理的関係を表す。同一のリンク番号を持つメッセージとバイナリデータが論理的関係にあることを表す。文字コードは、メッセージグループ・ヘッダの文字コード8bit(C24)の指示に従う。バイナリデータ・ヘッダのリンク番号と同一の値。
T05	Bin32	最終ユニット有効長	バイナリデータは、格納構造に基いて複数の固定長のバイナリユニットに分割して格納される。この場合、一般的に最終のバイナリユニットに端数のデータが発生する。有効データ長は、その端数の長さを表している。数値0が1byteに相当し、以下、2が2byte、nがnbyteに相当する。
T06	Bin32	全ユニット数	バイナリデータは、格納構造に基いて複数の固定長のバイナリユニットに分割して格納される。全ユニット数は、その全バイナリユニット数を表す。尚、全バイナリユニット数とは一つのバイナリデータのユニット数で、バイナリデータ・ヘッダとバイナリデータ・トレーラを含む。
F07	X(232)	リザーブ	将来の拡張のためリザーブ (all X' 20' をセット)。

7.5 バイナリーデータの格納構造

バイナリーデータは、通常モード時も分割モード時も固定長であり、下記のように格納する(図7-5参照)。

(1) 通常モード時 (メッセージ・ヘッダの分割モード(C23=X' 53'))

バイナリーデータ・ヘッダ (2 5 1 byte) とバイナリーデータ・トレーラ (2 5 1 byte) は、それぞれ一つの可変長レコードに格納する。そして、バイナリーユニット (3 2 0 0 1 byte) を一つの可変長レコードに格納する。

(2) 分割モード時 (メッセージ・ヘッダの分割モード(C23=X' 20' or X' 4D'))

バイナリーデータ・ヘッダ (2 5 1 byte) とバイナリーデータ・トレーラ (2 5 1 byte) は、それぞれ一つの固定長レコードに格納する。そして、バイナリーユニット (2 5 1 byte) を一つの固定長レコードに格納する。

(3) 分割区分のシーケンス (各レコードの最初の1 byte)

- ① バイナリーデータ・ヘッダ X' 40'
- ② バイナリーユニット X' 41' ~X' 48' の繰り返しシーケンス, 最後のレコード (バイナリーデータ・トレーラ直前のレコード) は、X' 49'
- ③ バイナリーデータ・トレーラ X' 40'

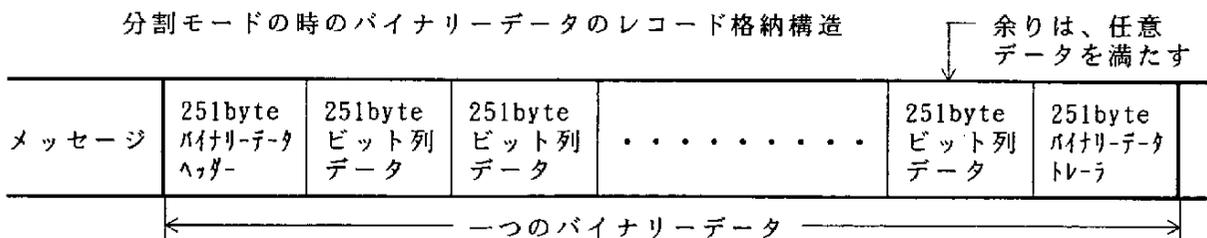
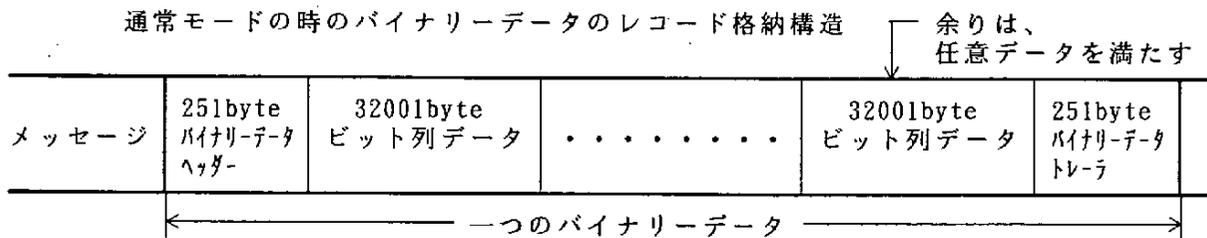


図7-5 バイナリーデータの格納構造

7.6 設計画像EDI専用データタグ番号

タグ番号『X'EF00' (=61184)』～『X'EF0F' (=61199)』までの16種のTFDを設計画像EDI専用特殊TFDとして、リザーブする。これらの特殊TFDは設計画像データ指示子として、メッセージ内のTFD(データ項目)として用いる。この内、図7-6に示す8種について、その内容を規定する。残りは、将来の拡張に備えてのリザーブとする。

EF00	××	リンク番号	9属性 Max 4byte
EF01	××	ファイル識別名	X属性 Max 80byte
EF02	××	フォーマット識別名	X属性 Max 32byte
EF03	××	圧縮識別名	X属性 Max 32byte
EF0C	××	ファイル名メモ	X属性 Max 250byte
EF0D	××	ファイル名メモ	K属性 Max 250byte
EF0E	××	任意メッセージ	X属性 Max 250byte
EF0F	××	任意メッセージ	K属性 Max 250byte

図7-6 予約された専用データタグ(設計画像データ指示子)

- ① X'EF00'～X'EF0F'はセットで用いる。但し、不要なものは省略できる。
- ② X'EF00'～X'EF0F'はメッセージ中で用いる。
- ③ 『リンク番号(X'EF00')』は、同一メッセージグループ内のn個のバイナリデータを区別するとともに、特定のメッセージに対応するバイナリデータを特定する(図7-7参照)。

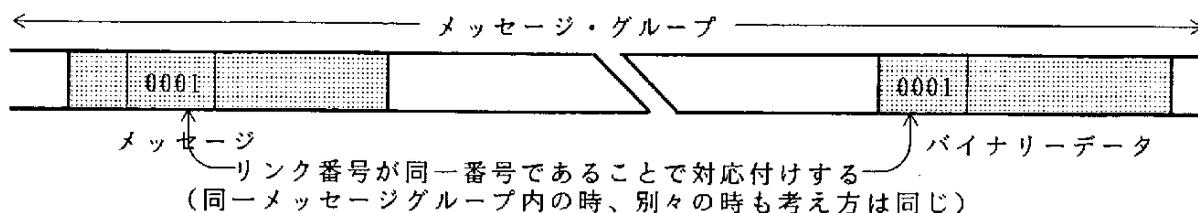


図7-7 リンク番号によるメッセージとバイナリデータのリンク

- ④ 『ファイル識別名 (X' EF01') 』は③の『リンク番号』で特定されるバイナリーデータ・ヘッダー内の『ファイル識別名』と同一の値でなければならない。『ファイル識別名』は、通常は受け側の社内システムで、メッセージと関連設計画像データファイルをリンクするために用いる。
- ⑤ 『フォーマット識別名 (X' EF02') 』は、設計画像データそのもののフォーマットを表すデータコードである。③の『リンク番号』で特定されるバイナリーデータ・ヘッダー内の『フォーマット識別名』と同一の値でなければならない。このデータコードは登録方式とする必要があるが、本シンタックスルールでは規定しない。運用で決定する。
- ⑥ 『圧縮識別名 (X' EF03') 』は、設計画像データそのものの圧縮方法を示すデータコードである。③の『リンク番号』で特定されるバイナリーデータ・ヘッダー内の『圧縮識別名』と同一の値でなければならない。このデータコードは『フォーマット識別名』と同じく登録方式とする必要があるが、本シンタックスルールでは規定しない。運用で決定する。
- ⑦ 『ファイル名メモ (X' EF0C' はX属性、X' EF0D' はK属性) 』は、ユーザーが使用する任意のメモである。このファイル名メモは、関連するバイナリーデータヘッダーにはない。
- ⑧ 『任意メッセージ (X' EF0E' はX属性、X' EF0F' はK属性) 』は、ユーザーが使用する任意のメッセージである。このメッセージは、関連するバイナリーデータヘッダーにはない。
- ⑨ 複数組の設計画像データ指示子をメッセージ内で用いる時は、マルチ明細とする。
- ⑩ 『リンク番号』『ファイル識別名』『フォーマット識別名』『圧縮識別名』については、関連する(『リンク番号』でリンクされる)バイナリーデータ・ヘッダー内にも同一の値のデータがある。しかし、『ファイル名メモ』と『任意メッセージ』については、バイナリーデータヘッダー内には、同一のデータはない。

7.7 補 足 (トランスレーターにおける設計画像データ指示子の位置づけ)

7.7.1 送信側トランスレーター

- (1) 変換テーブルには、設計画像特殊TFDが通常データ項目と同様に定義される。
- (2) 変換テーブルで示されるローカルエリアには、設計画像データに関する情報が記述される。
- (3) 設計画像データは、一つのデータで一つのファイルを構成しているのが望ましい。この場合は、『ファイルアクセスに関する記述』は、ファイル名のみでよい。そうでない場合は、ファイル名とアクセスキーの記述が必要になる。
- (4) 通常のトランスレーターでは『リンク番号』を自動採番(昇順)する。『ファイル識別名』『フォーマット識別名』『圧縮識別名』『ファイル名メモ』『任意メッセージ』は、ユーザーが適当な値を設定しなければならない。
- (5) 変換テーブルには、メッセージパス機能が必要である。即ち、通常はメッセージとバイナリーデータ混在で生成するが(最小のメッセージは、設計画像特殊TFDのみとなる)、パス機能によりメッセージの作成を抑止する。

7.7.2 受信側トランスレーター

- (1) 変換テーブルには、設計画像データ指示子が通常データ項目と同様に定義される。
- (2) 変換テーブルで示されるローカルエリアには、設計画像データに関する情報が、受信メッセージから展開される。この情報は通常メッセージの方から展開される。
- (3) 『ファイルアクセスに関する記述』は、トランスレーターが自動発生し、ユーザー側にその値を通知するエリアとなる。
- (4) 『ファイルアクセスに関する記述』は、受信した設計画像データを書き込むファイルを指示する。
- (5) トランスレーターが『ファイルアクセスに関する記述』を自動発生する方法として、以下に示すような複数の方法の中から、ユーザーが選択できるようにする。
 - ① あらかじめ、ファイル名列をトランスレーターに与え、その中からトランスレーターが自動選択する。
 - ② プレフィックス(数文字)をトランスレーターに与え、トランスレーターがそれに4桁の番号を追加し、ファイル名とする。
 - ③ ファイル名をトランスレーターに与え、アクセスキー(数字6桁程度)をトランスレーターが『000001』から昇順に付番する方式(メインフレーム向き)。
- (6) 設計画像データをユーザーのファイルへ書き込むとき、メッセージ関係とは別のユーザーインタフェースエリアを用意しておき、ここへ、ハンドリングしたバイナリーデータのヘッダー情報をおとす。

第 8 章 特殊データの構造

特殊データには、システム運用電文等の要素として使われる 251 byte 固定長固定フォーマットの特殊メッセージと EDI サービス事業者へ同報を指示するための 251 byte 固定長固定フォーマットの特殊ヘッダーがある。特殊データは一つの 251 byte のレコードに格納される。

8.1 受信確認メッセージ

(1) 受信確認メッセージの機能

受信確認電文に使われるメッセージで、受信者が解釈可能なメッセージを受信できたか否かを、発信者に通知する。

(2) 受信確認メッセージの構造

受信確認メッセージ構造を図 8-1 及び表 8-1 に示す。エラーフラグ 1~5 (E11~E15) にセットする値を表 8-3 に示す。

8.2 エラー情報メッセージ

(1) エラー情報メッセージの機能

エラー情報電文に使われるメッセージで、EDI サービス事業者が、交換処理に際しエラーが発生したことを、電文発信者に通知する。

(2) エラー情報メッセージの構造

エラー情報メッセージの構造を図 8-2 及び表 8-2 に示す。エラーフラグ 1~5 (E11~E15) にセットする値を表 8-3 に示す。



図 8-2 エラー情報メッセージのフォーマット



図 8-1 受信確認メッセージのフォーマット

表 8-1 受信確認メッセージのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	X' 39' 固定。
C02	X(1)	ワード区分	X' 44' 固定。
D03	9(5)	シーケンス番号	同一メッセージグループ内で1 から順に1 から昇順に付与(文字コード JIS-X0201)。
C01	X(1)	分割区分	(受信メッセージグループ・ヘッダ前半の内容)
C02	X(1)	ワード区分	
C03	X(1)	運用モード	
C04	X(12)	発信VANコード	
C05	X(12)	発信センターコード	
C06	X(12)	発信者コード	
C07	X(12)	受信VANコード	
C08	X(12)	受信センターコード	
C09	X(12)	受信者コード	
C10	X(4)	BPID機関	
C11	X(2)	BPIDサブ機関	
C12	X(2)	BPID版	
F13	X(12)	リザーブ	
C14	X(4)	情報区分コード	
C15	9(3)	第1トータル項目No-1	(受信メッセージグループ・トレー前半の内容)
C16	9(3)	第1トータル項目No-2	
C17	X(2)	フォーマットID	
C18	X(10)	リザーブ	受信に成功したメッセージグループ・ヘッダの、分割識別子 (C01) からハッシュトータル2 (E05) までの37byteのコピー。
C19	X(12)	作成日付時刻	
C01	X(1)	分割区分	
C02	X(1)	ワード区分	エラーコードをセット
D03	9(5)	シーケンス番号	
E04	9(15)	ハッシュトータル1	
E05	9(15)	ハッシュトータル2	
E11	X(2)	エラーフラグ1	
E12	X(2)	エラーフラグ2	受信側で、受信用トランスレータでメッセージグループを処理した時発生したエラーのエラーコードをセットする。受信用トランスレータで検出したエラーのエラーコードを検出順に最大5個までセットできる。エラーコードを、表 8-1に示す(文字コード JIS-X0201)。allX' 20' かつallX' 30' の時、エラー無しとする。
E13	X(2)	エラーフラグ3	
E14	X(2)	エラーフラグ4	
E15	X(2)	エラーフラグ5	
E20	X(12)	日付時刻	
F21	9(56)	リザーブ	将来の拡張のためリザーブ (allX' 20' をセット)。

表8-2 エラー情報メッセージのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)	
C01	X(1)	分割区分	X' 39' 固定。	
C02	X(1)	レコード区分	X' 44' 固定。	
D03	9(5)	シケンス番号	同一メッセージグループ内で1から順に1の昇順に付与(文字コード JIS-X0201)。	
C01	X(1)	分割区分	(エラー・メッセージグループ・ヘッダ前半の内容) エラーしたメッセージグループ・ヘッダの、分割識別子 (C01) から 作成日付時刻 (C28) までの 162byte のコピー。	
C02	X(1)	レコード区分		
C03	X(1)	運用モード		
C04	X(12)	発信VANコード		
C05	X(12)	発信センターコード		
C06	X(12)	発信者コード		
C07	X(12)	受信VANコード		
C08	X(12)	受信センターコード		
C09	X(12)	受信者コード		
C10	X(4)	BPID機関		
C11	X(2)	BPIDサブ機関		
C12	X(2)	BPID版		
F13	X(12)	リザーブ		
C14	X(4)	情報区分コード		
C15	9(3)	第1トータル項目No.-1		
C16	9(3)	第1トータル項目No.-2		
C17	X(2)	フォーマットID		
C18	X(10)	リザーブ		
C19	X(12)	作成日付時刻		
F20	X(12)	リザーブ		
C21	X(6)	シケンスIDバージョン		
C22	X(1)	拡張モード		
C23	X(1)	分割モード		
C24	X(1)	文字コード8bit		
C25	X(1)	文字コード16bit		
C26	X(1)	非透過モード		
C27	9(5)	第2トータル項目No.-1		
C28	9(5)	第2トータル項目No.-2		
C01	X(1)	分割区分	(エラー・メッセージグループ・トレー前半の内容) エラーしたメッセージグループ・ヘッダの、分割識別子 (C01) から ハッシュトータル2 (E05) までの37byte のコピー。	
C02	X(1)	レコード区分		
D03	9(5)	シケンス番号		
E04	9(15)	ハッシュトータル1		
E05	9(15)	ハッシュトータル2		
E11	X(2)	エラーフラグ1	エラーコードをセット	受信側で、受信用トランスレータでメッセージグループを処理した時発生したエラーの エラーコードをセットする。受信用トランスレータで検出したエラーのエラーコードを検出順に 最大5個までセットできる。エラーコードを、表 - に示す(文字コード JIS-X0201)。 allX' 20' か allX' 30' の時、エラー無しとする。
E12	X(2)	エラーフラグ2	"	
E13	X(2)	エラーフラグ3	"	
E14	X(2)	エラーフラグ4	"	
E15	X(2)	エラーフラグ5	"	
E20	X(12)	日付時刻	確認メッセージを作成した日付と時刻をセット(文字コード JIS-X0201)。	
F22	X(23)	リザーブ	将来の拡張のためリザーブ (allX' 20' をセット)。	

表 8 - 3 受信確認メッセージ、エラー情報メッセージで表示されるエラーコード

エラーコード	エラー内容
(7桁)	エラーなし
00	エラーなし
01	取決め以外の情報区分コード（トランスレーターの構造によっては、出力されない）
02	メッセージグループ・ヘッダーが見つからない
03	メッセージグループ・トレーラが見つからない
04	シンタックスIDの不正
05	分割区分シーケンスエラー
10	未定義制御タグの検出
11	不正データタグ検出
12	マルチ明細ヘッダーが実行形式SMテーブル上にない
13	マルチ明細トレーラが実行形式SMテーブル上にない
14	固有側（標準側）繰り返しが標準側（固有側）繰り返しを超えた
15	データのレングスが最大値を超えた
16	チェック・サム項目の値が数値でない
17	数値変換の項目の値が数値でない
18	標準側データ長が固有側データ長より大きい
19	レコード区分がD (X'44')でない（メッセージが見つからない）
20	過大レコード長
21	メッセージ・トレーラ (X'FE')がない
22	負のデータ有り（9属性の時など）
30	シーケンスNoが昇順ではない
31	チェックサムの数値がメッセージグループ・トレーラ上の数値と一致しない
32	実行形式SMテーブルサーチ不能
33	不正文字コードの検出
34	非透過モード時のメッセージ長不正
35	縮小モード中のマルチ明細のネスト、もしくは拡張モードマルチ明細ヘッダー有り
36	Y属性（日付）データの不正
40	UNAセグメントの不正
41	UNBセグメント無し
42	UNHセグメント無し
43	UNTセグメント無し
44	UNZセグメント無し
81	交換エラー
82	同報エラー
99	その他のエラー

8.3 同報ヘッダー

(1) 同報ヘッダーの機能

同報ヘッダーは、メッセージグループ・ヘッダーの直前に付加することで、直後の1メッセージ・グループが同報電文(メッセージグループ)であることを、EDIサービス事業者へ指示する。

(2) 同報ヘッダーの構造

同報ヘッダーの構造を図8-3及び表8-4に示す。

一つの同報ヘッダーで、同報通知先6カ所まで指定できる。7カ所以上の時は、継続区分(B04)を、X'43'(継続有り)として、さらに同報ヘッダーを並べる。最後の同報ヘッダーの継続区分(B04)は、X'45'(継続無し)とする。

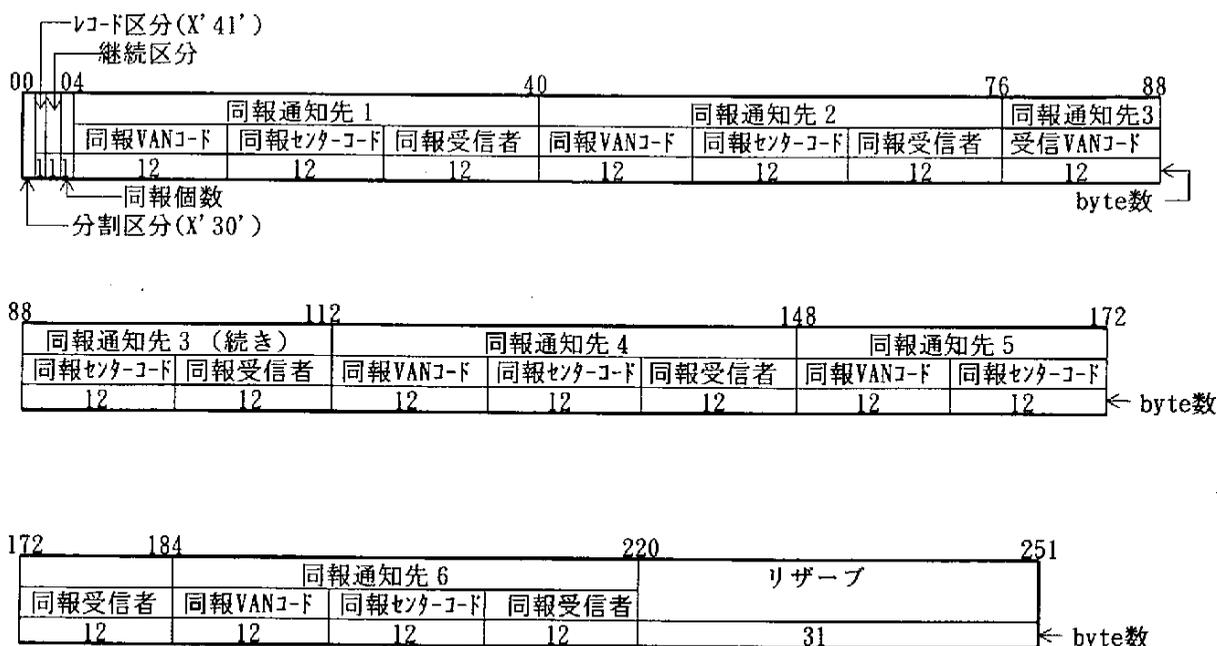


図8-3 同報ヘッダーのフォーマット

表8-4 同報ヘッダーのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	X' 30' 固定, 同報ヘッダーを表す。
C02	X(1)	コード区分	X' 41' 固定, 同報ヘッダーを表す。
B03	X(1)	継続区分	継続無し: X' 45', 継続有り: X' 43'。
B04	9(1)	同報個数	このヘッダー内の同報宛先の個数をあらわす(文字コード JIS-X0201)。
B05	X(12)	同報先VANコード1	同報通知先 1 (各コード英数字12byte(文字コード JIS-X0201))。
B06	X(12)	同報先センターコード1	
B07	X(12)	同報先受信者1	
B08	X(12)	同報先VANコード2	同報通知先 2 (" ("))。
B09	X(12)	同報先センターコード2	
B10	X(12)	同報先受信者2	
B11	X(12)	同報先VANコード3	同報通知先 3 (" ("))。
B12	X(12)	同報先センターコード3	
B13	X(12)	同報先受信者3	
B14	X(12)	同報先VANコード4	同報通知先 4 (" ("))。
B15	X(12)	同報先センターコード4	
B16	X(12)	同報先受信者4	
B17	X(12)	同報先VANコード5	同報通知先 5 (" ("))。
B18	X(12)	同報先センターコード5	
B19	X(12)	同報先受信者5	
B20	X(12)	同報先VANコード6	同報通知先 6 (" ("))。
B21	X(12)	同報先センターコード6	
B22	X(12)	同報先受信者6	
F23	X(31)	リザーブ	将来の拡張のためリザーブ (all X' 20' をセツト)。

第9章 メッセージグループ・トレーラの構造

メッセージグループ・トレーラは、メッセージグループの終りを示すトレーラで、図9-1及び表9-1で示す251byteの固定長固定フォーマットである。メッセージグループ・トレーラは、一つの251byteのレコードに格納される。

9.1 トータルチェック

メッセージグループ・ヘッダーのトータル項目No指定(C15, C16, C24, C25)で、blank(X'20')かゼロ(X'30')が指定されている時は、このエリアを無視する(トータル・チェックをしない)。

9.2 チェック用トータル数値の算出方法

メッセージグループ・ヘッダーで指定された項目No(データ・タグ値)の数値データをメッセージ中から取り出し、全体の桁数が15桁より小さい時は、左側に0を必要な数だけ追加し全体の桁数が16桁以上の時は、16桁以上をカットして15桁にして、加算する。和が16桁以上になる時は、その都度16桁以上をカットして15桁にする。一つのメッセージグループ内にある指定されたTFDの数値(データ部)の総計を、チェック用トータル数値とする。項目1と項目2それぞれ独立に計算する。

メッセージグループ・ヘッダーの拡張モード指示子で、縮小モード(X'20' or X'53')が設定されている場合には、第1トータル項目Noエリアに設定された3桁の項目No(データタグ)で示されるTFDのデータ部の数値を用いてチェック値を計算し、拡張モード(X'45)が設定されている場合には、第2トータル項目Noエリアに設定された5桁の項目No(データタグ)で示されるTFDのデータ部の数値を用いてチェック値を計算する。

計算に用いる数値は、TFD形式における数値データをそのまま用いる。N形式の数値データの時は、正負の符号と小数点を無視して用いる。特に、N形式において送信用のトランスレーターの不具合により、省略可能な右側のLSBにゼロが存在する可能性がある(図9-2)。この場合はゼロを無視しないで計算を行う。

blank(X'20')または0(X'30')の項目Noが指定された場合には、チェック用トータル数値を算出せず、項目合計を0(X'30')とする。

CIIシンタックスルールにおけるチェック用トータル数値の算出方法は、EIAJシンタックスルールにおけるそれとは異なるので、縮小モードにおいて互換性を持たせるためには、9属性で小数点以下の桁数が3桁で定義されているデータ項目を、第1トータル項目Noエリアに設定しなければならない。

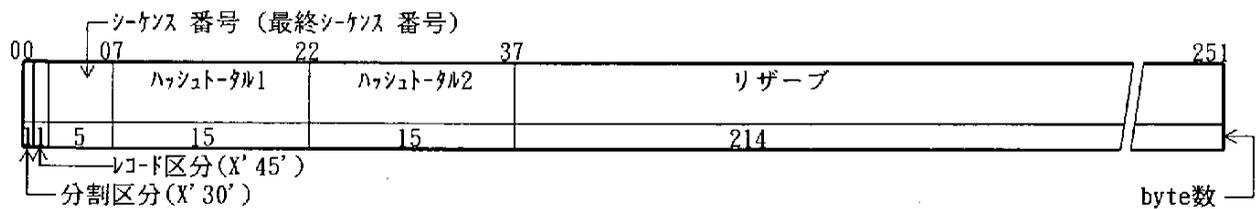


図 9-1 メッセージグループ・トレーラのフォーマット

表 9-1 メッセージグループ・トレーラのデータ項目

識別	属性	データ項目名	説明 (設定すべき値)
C01	X(1)	分割区分	X'30' 固定, メッセージグループ・ヘッダ・トレーラを示す。
C02	X(1)	レコード区分	X'45' 固定, メッセージグループ・トレーラであることを示す。
D03	9(5)	シーケンス番号 (最終シーケンス番号)	直前のメッセージ又はバイナリデータのシーケンス番号。つまり、メッセージ及びバイナリデータの合計数を表す(文字コード JIS-X0201)。合計が 0 件の時は、X'30' となる。
E04	9(15)	ハッシュトータル 1	チェック用ハッシュトータル数値 1 (文字コード JIS-X0201)。
E05	9(15)	ハッシュトータル 2	チェック用ハッシュトータル数値 2 (文字コード JIS-X0201)。
R06	X(214)	リザーブ	将来の拡張のためリザーブ (all X'20' をセット)。

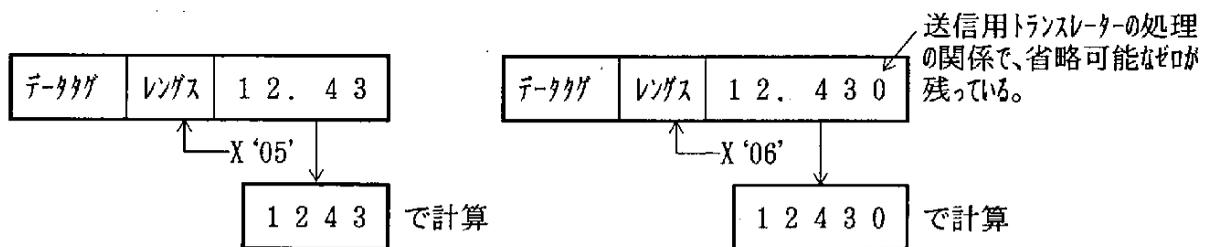


図 9-2 N形式データにおけるトータルチェック値計算用の数値

第10章 メッセージグループの構造

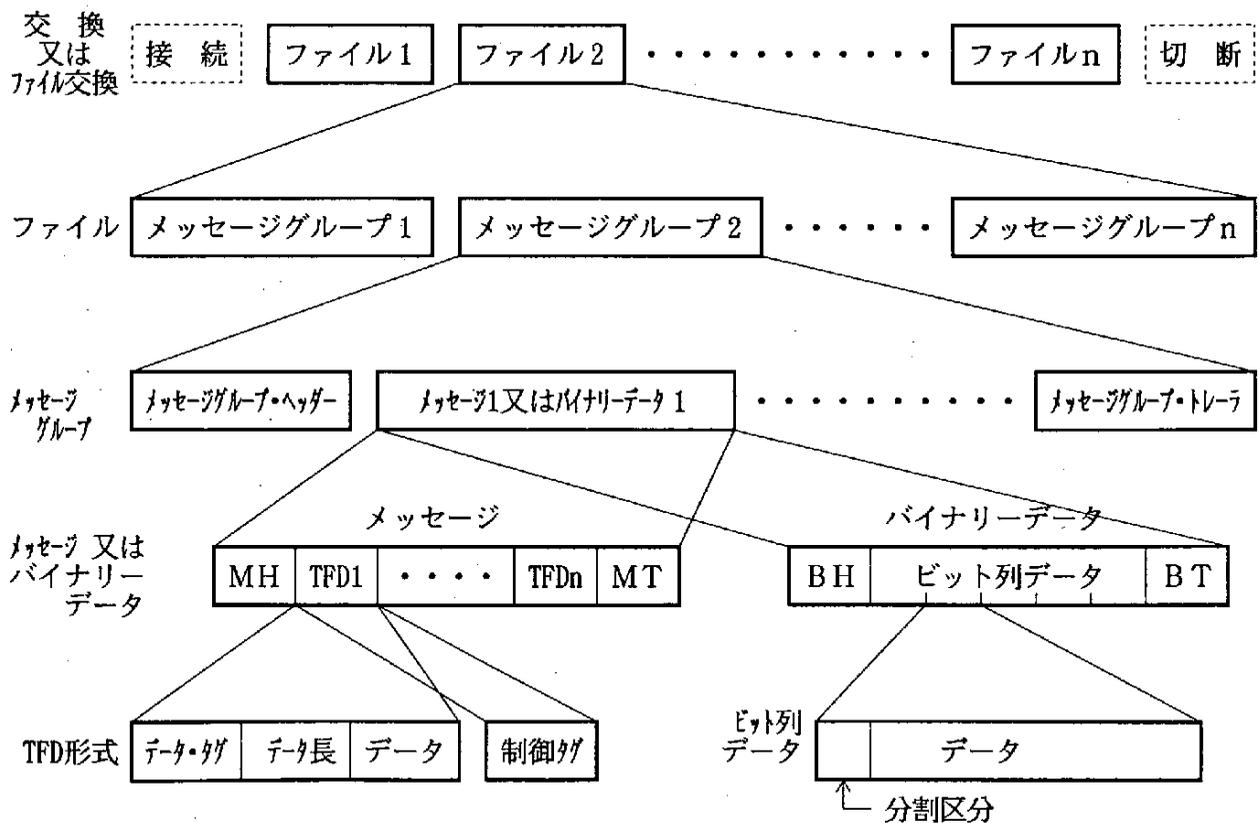
メッセージグループ・ヘッダー、一つ又は複数のメッセージか特殊メッセージ、一つ又は複数のバイナリーデータ及びメッセージグループ・トレーラで、メッセージグループを構成する。

メッセージグループはEDIネットワークにおける最小の交換単位である。ファイル転送システムによって交換を行う場合、一つのファイルに、複数のメッセージ・グループを格納することができる。しかし、一つのメッセージを複数のファイルにまたがって格納することはできない。

メッセージグループが持つ機能によって、業務処理電文と運用電文がある。又、同報ヘッダーを用いることで、両電文とも同報電文とすることができる。

10.1 業務処理電文

図10-1に示すように、メッセージグループ・ヘッダー、メッセージ、バイナリーデータ及びメッセージグループ・トレーラで構成される(図10-1参照)。



注) MH: メッセージ・ヘッダー MT: メッセージ・トレーラ
 BH: バイナリーデータ・ヘッダー BT: バイナリーデータ・トレーラ

図10-1 業務処理電文(メッセージグループ)の構成

10.2 運用電文

運用電文は、データ交換に係わるシステム運用を円滑に行うためにある。受信確認電文、エラー情報電文及びゼロ件メッセージ電文等がある。

運用電文では、メッセージグループ・ヘッダー及びメッセージグループ・トレーラの中のトータル項目Noを、すべてブランク (X'20')としてハッシュ・トータル・チェックをバイパスする。

10.2.1 受信確認電文

受信確認電文とは、解釈可能な通常電文（業務処理メッセージ）を受信できたかどうかを、業務メッセージの受信者から送信者へ知らせる電文である。この電文は、業務上の約束の成立等とは無関係な電文である。例えば、発注メッセージの受信済ステータスをこのメッセージで送信者へ通知しても、発注契約は成立しない。受信確認電文は、システム上の電文受信が成功したかどうかを、送信者に伝達するものである。

受信確認電文は、一对のメッセージグループ・ヘッダーとメッセージグループ・トレーラの中に、一つ又は複数個の受信確認データを入れた構成である（図10-2参照）。メッセージグループ・ヘッダーの情報区分コード(C14)を「9001」とする。

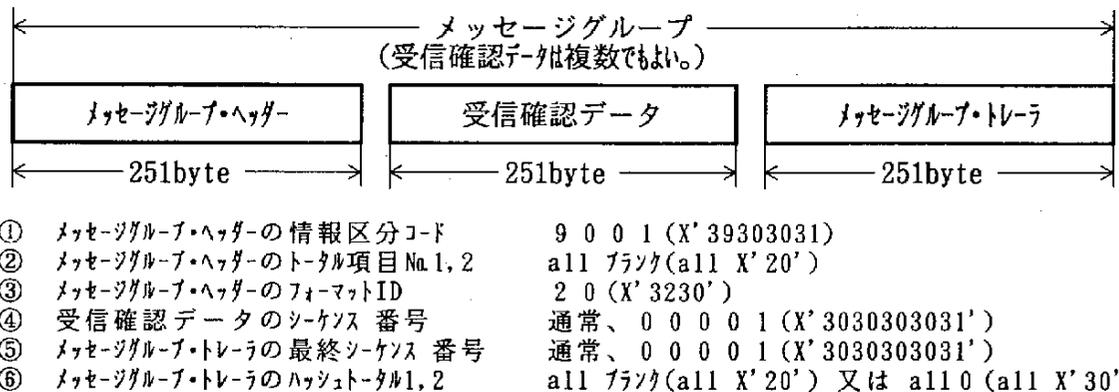
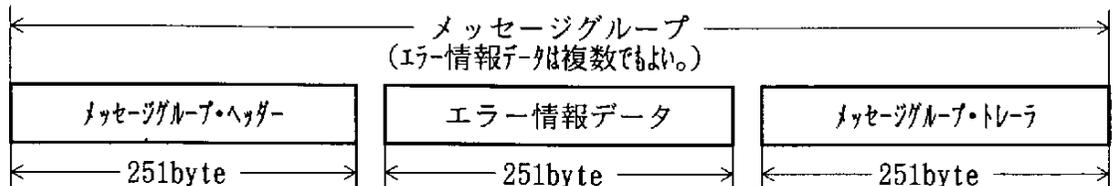


図10-2 受信確認電文の構成

10.2.2 エラー情報メッセージ電文

エラー情報メッセージは、EDIサービス事業者がユーザーへ、メッセージ・グループの交換エラーを伝えるために用いる。

エラー情報メッセージは、一對のメッセージグループ・ヘッダーとメッセージグループ・トレーラの中に、一つ又は複数個のエラー情報データを入れた構成である（図10-3参照）。メッセージグループ・ヘッダーの情報区分コード(C14)を「9201」とする。



- | | | |
|---|-------------------------------|--|
| ① | メッセージグループ・ヘッダーの情報区分コード | 9 2 0 1 (X' 39323031) |
| ② | メッセージグループ・ヘッダーのトータル項目 No.1, 2 | all フラック(all X' 20') |
| ③ | メッセージグループ・ヘッダーのフォーマットID | 2 0 (X' 3230') |
| ④ | メッセージグループ・トレーラのハッシュトータル1, 2 | all フラック(all X' 20') 又は all 0 (all X' 30') |

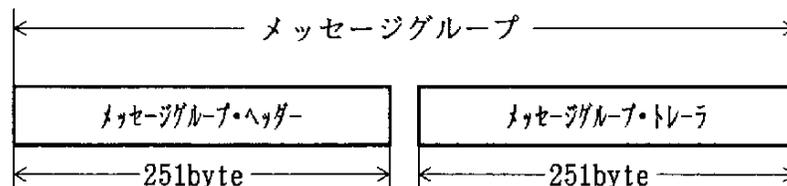
図10-3 エラー情報メッセージ電文の構成

10.2.3 ゼロ件メッセージ電文

ゼロ件メッセージ電文は、受信者に対して、送信すべきメッセージ件数がゼロ（無）であることを伝達する電文である。

ゼロ件メッセージ電文は、図10-4に示すように、メッセージのない（0件）メッセージグループ・ヘッダーとメッセージグループ・トレーラの対である。図10-4のメッセージグループ・ヘッダーの情報区分コード(C14)は、「9101」とする。

メッセージグループ・トレーラの中の最終シーケンス番号をゼロ（メッセージ無）とする。



- | | | |
|---|-------------------------------|--|
| ① | メッセージグループ・ヘッダーの情報区分コード | 9 1 0 1 (X' 39313031) |
| ② | メッセージグループ・ヘッダーのトータル項目 No.1, 2 | all フラック(all X' 20') |
| ③ | メッセージグループ・ヘッダーのフォーマットID | 2 0 (X' 3230') |
| ④ | メッセージグループ・トレーラの最終シーケンス番号 | 0 0 0 0 (X' 3030303030') |
| ⑤ | メッセージグループ・トレーラのハッシュトータル1, 2 | all フラック(all X' 20') 又は all 0 (all X' 30') |

図10-4 ゼロ件メッセージ電文の構成

10.3 同報電文

メッセージグループの直前に、同報ヘッダーを追加することで、ヘッダー直後の一つのメッセージグループを同報電文とすることができる（図10-5参照）。同報ヘッダーを二つ以上にし、同報宛て先を増やすことができる。

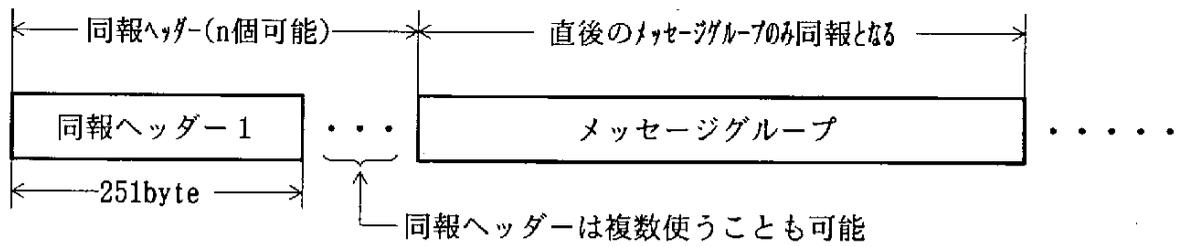


図10-5 同報電文の構成

第11章 Zレベルオプション仕様

Zレベルオプションは、将来廃止が予定されているオプションである。非透過モードオプションとTYPE-Eオプションがある。

11.1 非透過モード・オプション

非透過モード・オプションは、J手順などの非透過モードの通信システムを活用する時利用するオプションである。

メッセージグループ・ヘッダーの非透過モード指示子(C26)で、非透過モード(X'4D')が設定された場合、以下のように、文字コードの変換およびバイナリデータの補正が行われる。

尚、非透過モードでは、JIS-X0201とJIS-X0208の文字コードが使用されることを前提に、非透過モード通信系による不具合を回避する対策を行うので、これ以外の文字コードのデータを用いることは許されない。

11.1.1 メッセージグループ・ヘッダーとメッセージグループ・トレーラでの対策

(1) 分割識別子

メッセージグループ・ヘッダーとメッセージグループ・トレーラの分割区分は、特に変換をしない(0(X'30'))とする。これは、受信用トランスレーターが分割区分によって、メッセージグループ・ヘッダーを識別するために必要な処置である。

(2) 分割区分以外

X'20'～X'3F'を、X'80'～X'9F'に変換し、伝送する。この変換は、トランスレーターで行われる。(表11-1を参照)。

受信用トランスレーターでは、透過モードか非透過モードかを判別してメッセージグループを分析しなければならない。メッセージグループ・ヘッダーとメッセージグループ・トレーラについては次のようにして行う。メッセージグループ・ヘッダーの非透過モード表示は、透過モードの時は空白(X'20')かS(X'53')となり、非透過モードの時はM(X'4D')：非透過モードへの変換後もM(X'4D')となる。従って、M(X'4D')であるかどうかを判定する。非透過モードの時は、メッセージグループ・ヘッダーとメッセージグループ・トレーラの分割区分を除いて、表13-1に従って逆変換を行ってから分析を行う。

11.1.2 メッセージの対策

(1) メッセージ・ヘッダー

メッセージ・ヘッダーの分割区分、レコード区分、シーケンスNoは、表11-1に従って変換を行う。そして、Bタイプのメッセージヘッダーを用いるが、メッセージ長を表す文字デ

ータの文字コードのX'30'～X'39'は表11-1に従って、X'90'～X'99'に変換して表さなければならない(Aタイプのメッセージヘッダーは使えない)。

受信用トランスレータでは、メッセージグループ・ヘッダーのチェックにより非透過モードを検出した時は、メッセージヘッダーの分割区分を表11-1に従って逆変換して透過モードの時と同様な方法でメッセージを認識するか、X'99'(但し、分割モード時はX'91'～X'98'で最終レコードはX'99'のシーケンス)を検出することで、メッセージを認識する。

(2) TFD形式データ1のバイナリーデータ(データタグおよびレンジスタグ)の補正

表11-2に従って補正を行う。補正を行うバイナリーデータの直前には非透過モード・補正指示子(X'F8')を追加する。表11-2で、下線が付いているデータは必ず補正しなければならない(例えば、X'01'→X'F881')。一方()が付いているデータの補正は任意である。例えば、X'14'は、無補正でもX'14'→X'F894'のように補正しても、どちらでもよい。X'40'のように無補正が指示されているデータは、補正してはいけない。

例) X'00' → X'F880' , X'F8' → X'F8C8'
X'40' → X'40' , X'FF' → X'F8CF'

バイナリーデータの補正を行った時は、メッセージ長が増加する。非透過モード・バイナリーデータ補正指示子(X'F8')を一つ使うごとに、1 byteずつ増加する。

(3) TFD形式データ1のデータ部の変換

表11-1に従って変換する。この変換は、変換前の文字コードがJIS-X0201かJIS-X0208になっていることが前提になっている。すなわち、非透過モードではこれ以外の文字コードを使うことはできない。

(4) TFD形式データ2(TFD制御子の変換)

制御タグはバイナリーデータであるので表11-2に従って補正するが、X'F0'～X'FF'(X'F8'を除く)の範囲にあり、X'FF'をX'CF'に補正する以外は、無補正である。但し、現在制御タグX'FF'は定義されていないので、実質的に補正無しとしてよい。X'F8'は、この文字がTFD制御子(バイナリーデータ補正子)として使用される時は無補正であり、バイナリーデータ(データタグまたはレンジスタグ)として使用される時は、表11-2に従って、X'F8'を追加してX'F8C8'にしなければならない(メッセージ長も1増加する)。

TFD制御子の制御データについては、以下のようになる。

① インターナルセグメント指示子(X'F9')

この制御タグの制御データ(インターナルセグメント番号)は表11-1に従って変換する。

② マルチ明細ヘッダー(X'FA')

この制御タグの制御データ(明細番号)は表11-1に従って変換する。

(5) メッセージグループ・トレーラ (X'FB')

変換も補正も特に必要としない (制御タグと見なす)。

表11-1 非透過モード時の文字コードの変換

数値	変換値	数値	変換値	数値	変換値
2 0	→ 8 0	3 0	→ 9 0	4 0 } ↓ 7 F } A 0 } ↓ D F }	変換無
2 1	→ 8 1	3 1	→ 9 1		
2 2	→ 8 2	3 2	→ 9 2		
2 3	→ 8 3	3 3	→ 9 3		
2 4	→ 8 4	3 4	→ 9 4		
2 5	→ 8 5	3 5	→ 9 5		
2 6	→ 8 6	3 6	→ 9 6		
2 7	→ 8 7	3 7	→ 9 7		
2 8	→ 8 8	3 8	→ 9 8		
2 9	→ 8 9	3 9	→ 9 9		
2 A	→ 8 A	3 A	→ 9 A		
2 B	→ 8 B	3 B	→ 9 B		
2 C	→ 8 C	3 C	→ 9 C		
2 D	→ 8 D	3 D	→ 9 D		
2 E	→ 8 E	3 E	→ 9 E		
2 F	→ 8 F	3 F	→ 9 F		

(注) メッセージグループ・ヘッダとトレーラの分割識別子
及びバイナリデータ以外に適用,
表中の値は、すべてHex 表示

表11-2 非透過モード時のバイナリデータの補正

数値	補性値	数値	補性値	数値	補性値	数値	補性値	数値	補性値
00	→ 80	10	→ 90	(20	→ A0)	(30	→ B0)	4 0 } ↓ F 7 } F 8 → C 8 F 9 } ↓ F E } F F → C F }	補正無
01	→ 81	(11	→ 91)	(21	→ A1)	(31	→ B1)		
02	→ 82	(12	→ 92)	(22	→ A2)	32	→ B2		
03	→ 83	(13	→ 93)	(23	→ A3)	(33	→ B3)		
04	→ 84	(14	→ 94)	(24	→ A4)	(34	→ B4)		
05	→ 85	15	→ 95	(25	→ A5)	(35	→ B5)		
06	→ 86	16	→ 96	26	→ A6	(36	→ B6)		
(07	→ 87)	17	→ 97	(27	→ A7)	37	→ B7		
(08	→ 88)	(18	→ 98)	(28	→ A8)	(38	→ B8)		
(09	→ 89)	(19	→ 99)	(29	→ A9)	(39	→ B9)		
(0A	→ 8A)	(1A	→ 9A)	(2A	→ AA)	(3A	→ BA)		
(0B	→ 8B)	(1B	→ 9B)	(2B	→ AB)	(3B	→ BB)		
(0C	→ 8C)	(1C	→ 9C)	(2C	→ AC)	(3C	→ BC)		
(0D	→ 8D)	(1D	→ 9D)	2D	→ AD	3D	→ BD		
(0E	→ 8E)	(1E	→ 9E)	2E	→ AE	(3E	→ BE)		
(0F	→ 8F)	(1F	→ 9F)	(2F	→ AF)	(3F	→ BF)		

(注) データタグ、制御タグ及びレングスタグのバイナリデータに適用
表中の値は、すべてHex 表示

11.2 TYPE-Eオプション

TYPE-Eは、将来、UN/EDIFACTと並行使用する時に備えて用意されるオプションで、ヘッダーとトレーラにISO 9735のシンタックスルールを取り入れ、電文全体がISO 9735と同一のセグメント構成ルールに従って、コーディングされる。文字コードは、JIS-X0201(8bit文字)とJIS-X0208(16bit文字)の使用を義務づける。

CCIシンタックスルールの基本形は、メッセージグループを構成するメッセージグループ・ヘッダー、メッセージそしてメッセージグループ・トレーラ等の構成要素の外形が、ISO 9735とは異なるため、ISO 9735をサポートするEDIネットワークを利用することができない。TYPE-Eは、構成要素の外形がISO 9735と同一になっているので、ISO 9735をサポートするEDIネットワークを利用することができる。但し、メッセージ内部の論理構造はUN/EDIFACTとは異なるため、UN/EDIFACTメッセージとして用いることはできない。

以下の記述では、8bitを1byteと記述し、X'1A'は、Hex表示を表すものとする。又、誤解を避けるために、例えば、キャラクターCは、C(X'43')のようにHex表示を併用する。又、図示されたフォーマット中の値は、すべてHex表示とする。さらに、特に断らないかぎり、バイナリーとは、符号無2進数とする。

11.2.1 交換の階層

TYPE-Eの交換の階層は、ISO 9735と同一である。詳細は、ISO 9735の規格書を参照されたい。尚、機能グループは用いない。従ってUNGセグメントとUNEセグメントは用いない。

11.2.2 UNAセグメントとUNBセグメント

UNA、UNBセグメントの構造は、ISO 9735と同一で、基本形におけるメッセージグループ・ヘッダーに相当する。UNAセグメントはオプションであるが、使用を義務づける。

(1) UNAセグメント

- | | |
|--------------------------------|-------|
| ① コンポーネント・データエレメント・セパレータ | X'FC' |
| ② データエレメント・セパレータ | X'FF' |
| ③ 小数点記号文字 | X'2E' |
| ④ リリース記号 | X'FD' |
| ⑤ セグメント・セパレータ | X'FE' |

(2) UNBセグメント

以下に、TYPE-Eと基本形のメッセージグループ・ヘッダーとの対応を示す。文字コードは、JIS-X0201を使用する。尚、TYPE-Eでは、発信者コード、受信者コード、トータル項目NoおよびフォーマットIDは使用しない。

S001	シタックス 識別記号		
0001	シタックス識別記号	(文字 4byte)	C I I B ('43494942')
0002	シタックスバージョン番号	(数値 1byte)	1 (X'31')
S002	交換発信者		
0004	送信者識別コード	(文字35byte-Max)	発信センターコード (12文字)
0007	識別コード修飾子	(文字 4byte-Max)	使用しない。
0008	戻り経路用アドレス	(文字14byte-Max)	(発信者) 発信VANコード (12文字)
S003	交換受信者		
0010	受信者識別コード	(文字35byte-Max)	受信センターコード (12文字)
0007	識別コード修飾子	(文字 4byte-Max)	使用しない。
0014	経路アドレス	(文字14byte-Max)	(発信者) 受信VANコード (12文字)
S004	作成日付		
0017	日付	(数値 6byte)	作成日付
0014	時間	(数値 4byte)	作成時刻 (秒は使用しない)
0020	交換コントロール参照番号	(文字14byte-Max)	任意の値とする。受信確認 要求をするときは all 0 (X'30')等を使用しない。
S005	受信者参照番号, パスワード		
0022	受信者参照 パスワード	(文字14byte-Max)	使用しない。
0025	パスワード 修飾子	(文字 2byte)	使用しない。
0026	アプリケーション参照番号	(文字14byte-Max)	B P I D (8桁) + 情報 区分 (4桁)
0029	処理優先順コード	(文字 1byte)	使用しない。
0031	受信確認要求	(数値 1byte)	受信確認要求をする時は、1 (X'31')とし要求しない時は 省略か0(X'30')とする。
0032	通信協定識別	(文字35byte-Max)	使用しない。
0035	テスト指示記号	(数値 1byte)	運用モード (通常モードは、 0 (X'30')を使用)

11.2.3 メッセージの構造 (TYPE-E)

基本形におけるメッセージ・ヘッダーがUNHセグメントに、TFDエリアがCIIセグメントに、メッセージ・トレーラがUNTセグメントに、それぞれ置き換えられる (図11-1 参照)。

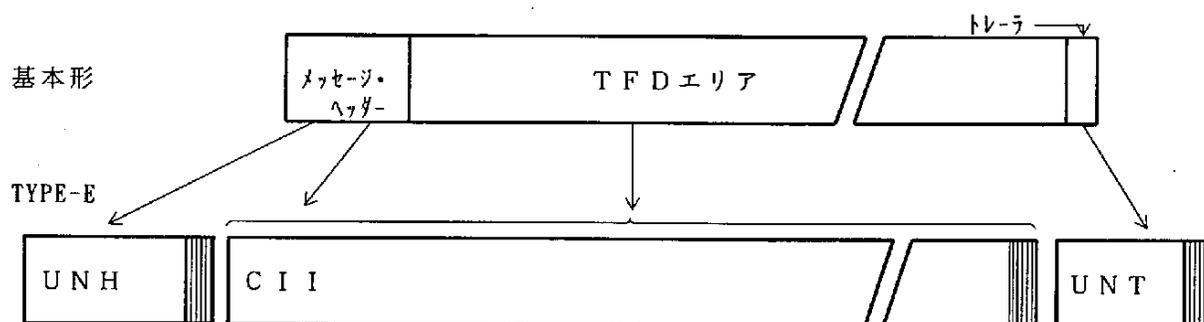


図11-1 基本形とTYPE-Eとの対応

(1) UNHセグメント

UNHセグメントは、ISO 9735と同一である。以下に、基本形のメッセージ・ヘッダーとの対応を示す。文字コードは、JIS-X0201を使用する。

0062	メッセージ参照番号	(文字14byte-Max)	シーケンスNo (5桁)
S009	メッセージ修飾子		
0065	メッセージタイプ	(文字 6byte-Max)	情報区分 (4文字)
0052	メッセージバージョン番号	(数値 3byte-Max)	版 (2文字)
0054	メッセージリリース番号	(数値 3byte-Max)	使用しない。
0051	管理機関コード	(文字 2byte)	使用しない。
0057	協会指定コード	(文字 6byte-Max)	使用しない。
0068	共通アクセス参照番号	(文字35byte-Max)	使用しない。
S010	数次更新型データの伝送状況		
0070	伝送のシーケンス	(数値 2byte-Max)	使用しない。
0073	初回伝送か最終伝送か	(文字 1byte)	使用しない。

(2) C I Iセグメント

Aタイプメッセージ（非透過モードは不可、透過モードに限る）のメッセージ・ヘッダーを変更したメッセージを、そのままC I Iセグメントとして用いる。

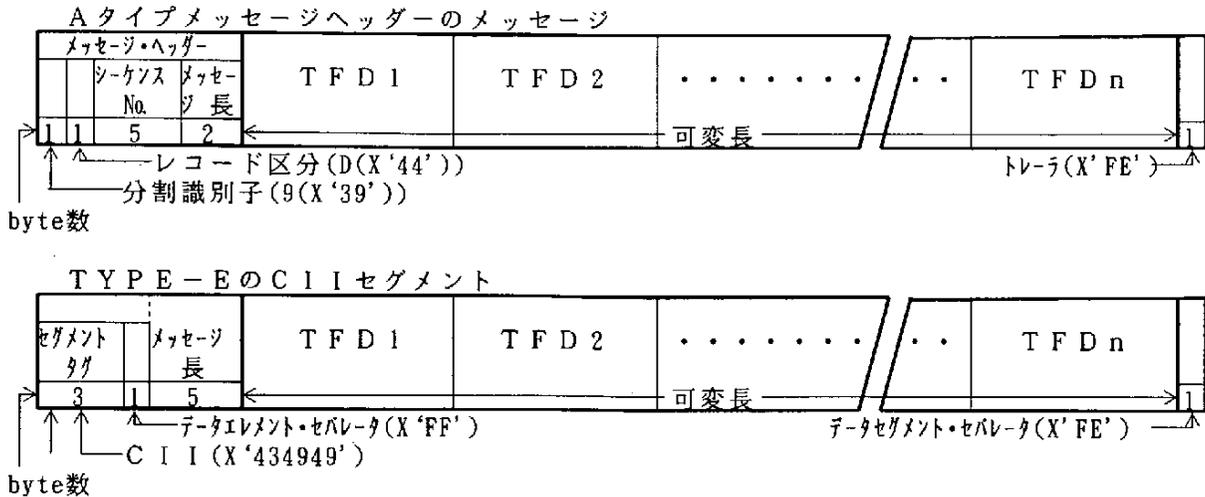


図11-2 TYPE-EのC I Iセグメント

① セグメントタグ（文字 3 byte）

C I I (X'434949')

② メッセージ長（文字 5 byte）

メッセージ長は、アンパック形式の数字 5 桁が用いられる。文字コードは、J I S - X 0 2 0 1を用いる。メッセージ長が、3 2 7 6 7 byteよりも長い時は、メッセージ長エリアを 9 9 9 9 9 とし、タイプ B メッセージの拡張メッセージ長と同等の方法で、アンパック形式の数字 7 桁で表す。この時、識別子と拡張メッセージ長を合わせて 8 byte メッセージ長が増加する。

尚、TFD内のバイナリーデータで、デリミターと衝突するデータについては、リリース文字 (X'FD')の付加を行う。拡張モードのデータタグの第 2 byte (X'FC' 以上の時)、拡張形式のレングスタグ (データ長) の第 2 byte (X'FC' 以上の時) およびマルチ明細トレーラ (X'FC')が、衝突する。

例) X '0 2 F E' → X '0 2 F D F E'
X 'F C F E' → X 'F D F C F D F E'

(3) UNTセグメント

UNTセグメントは、ISO 9735と同一である。TYPE 12のメッセージ・トレーラ (X'FE')に対応する。文字コードはJIS-X0201を使用する。

0074	メッセージ内のセグメント数	(数値 6byte-Max)	UNHとUNTを含むセグメント数
0062	メッセージ参照番号	(文字14byte-Max)	UNHの0062と同一の値 (シーケンスNo. (5桁))

11.2.4 UNZセグメント

UNZセグメントは、ISO 9735と同一である。基本形におけるメッセージグループ・トレーラに対応する。文字コードは、JIS-X0201を使用する。尚、TYPE-Eでは、トータル項目No. (項目合計1, 項目合計2) は、使用しない。

0036	交換コントロール・カウント	(数値 6byte-Max)	メッセージの総数 (最終シーケンス番号 (5桁))
0020	交換コントロール参照番号	(文字14byte-Max)	UNBの0020と同一とする。

11.2.5 メッセージの格納構造 (TYPE-E)

TYPE-Eでは、UNAセグメントとUNBセグメントを同一のレコードに格納し、その他のセグメントについては、一つのセグメントを一つの可変長レコードに格納する。

11.2.6 TYPE-Eとインターナルセグメント

CIIシンタックスルールには、ISO 9735のセグメントに対応するものとして、マルチ明細がある。通常、TFDエリアはマルチ明細がない時一つのセグメントで構成され、マルチ明細がある時は、マルチ明細とそうでない部分（名無しセグメント）で構成される（図11-3参照）。

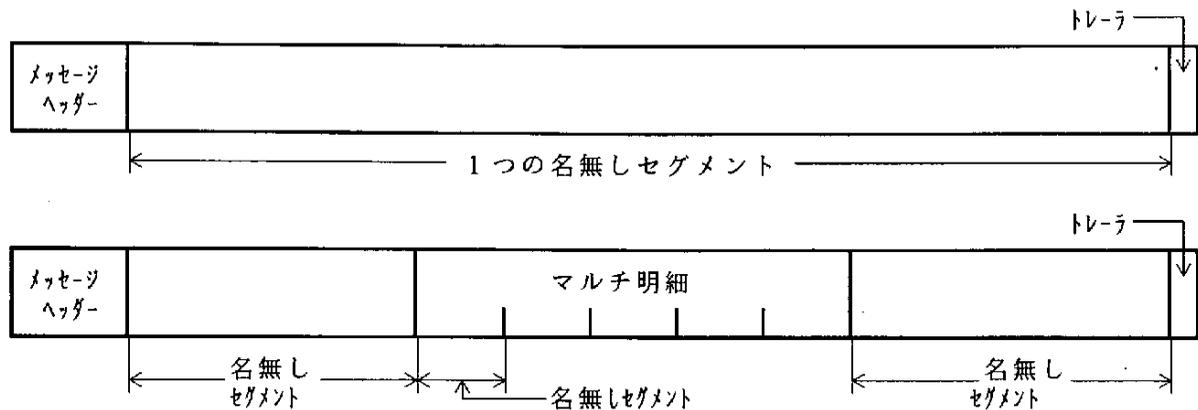


図11-3 通常のCIIシンタックスルールによるTFDエリアの構造

これに対し、UN/EDIFACT電文への変換を対象にしたTFDエリアは、ISO 9735と同等のセグメント構造が必要になる。このような時、インターナルセグメント指示子を用いて、TFDエリアをセグメント化する。

インターナルセグメント指示子を用いてセグメント化したTFDエリアを図11-4に示す。図11-4のインターナルセグメントA、B、C及び、マルチ明細セグメント1をUN/EDIFACT電文におけるセグメントに対応させる。UN/EDIFACT電文では、セグメント内のデータ項目の順番は固定化されているが、CIIシンタックスルールにおけるセグメント内のデータ項目（TFD）の順番は可変である。インターナルセグメントは、通常、受信用のCIIトランスレーターに対して何らの影響も与えない（無視される）。

尚、インターナルセグメントの名は、1～9とA～Zの35種類の使用が規定されているが、表1-1の8bit文字すべての使用に拡張してもよい。この場合、156種類（blankとゼロを除く）のセグメントをアプリケーションで設定できる。

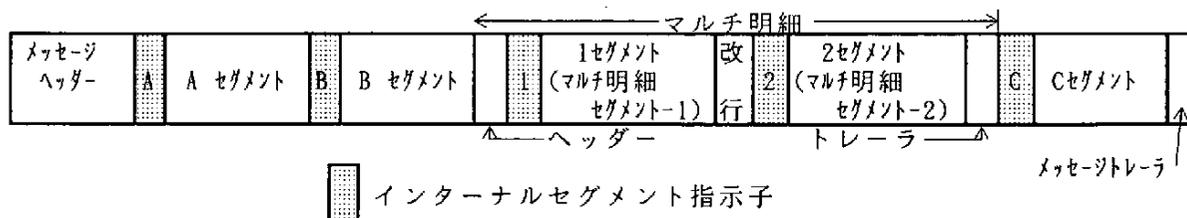


図11-4 インターナルセグメント指示子によるセグメント化

11.2.7 システム運用電文フォーマット

(1) 受信確認メッセージの構造 (TYPE-E)

TYPE-Eの受信確認メッセージは、図11-5に示すように、UNA, UNB, UNH, UNTおよびUNZセグメントで構成される。文字コードは、JIS-X0201を使用する。尚、一組のUNA, UNB, UNZセグメントの中に、複数のUNH, UNTセグメントの対を入れることができる。

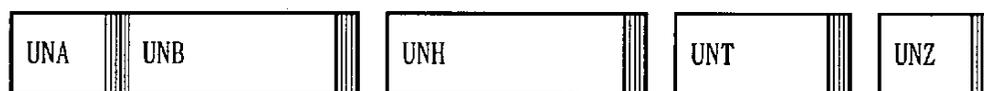


図11-5 TYPE-E受信確認メッセージの構造

- ① UNAセグメント (11.2.2-(1)を参照)
- ② UNBセグメント
 - a. 0026 アプリケーション参照番号
BPID (8桁) + 情報区分 (9001 (X'39303031'))
 - b. 0031 受信確認要求
使用しない (省略か0 (X'30'))。受信確認要求を、してはいけない。
 - c. その他の項目は、11.2.2-(2)を参照。

③ UNHセグメント

a. 0065 メッセージ・タイプ
情報区分

b. 0068 共通アクセス参照番号

受信に成功したUNBセグメントの0020交換コントロール参照番号（文字14byte-Max）をセットする。

c. その他の項目は、11.2.3-(1)を参照

④ UNTセグメント（11.2.3-(3)を参照）

⑤ UNZセグメント（11.2.4を参照）

(2) ゼロ件メッセージの構造

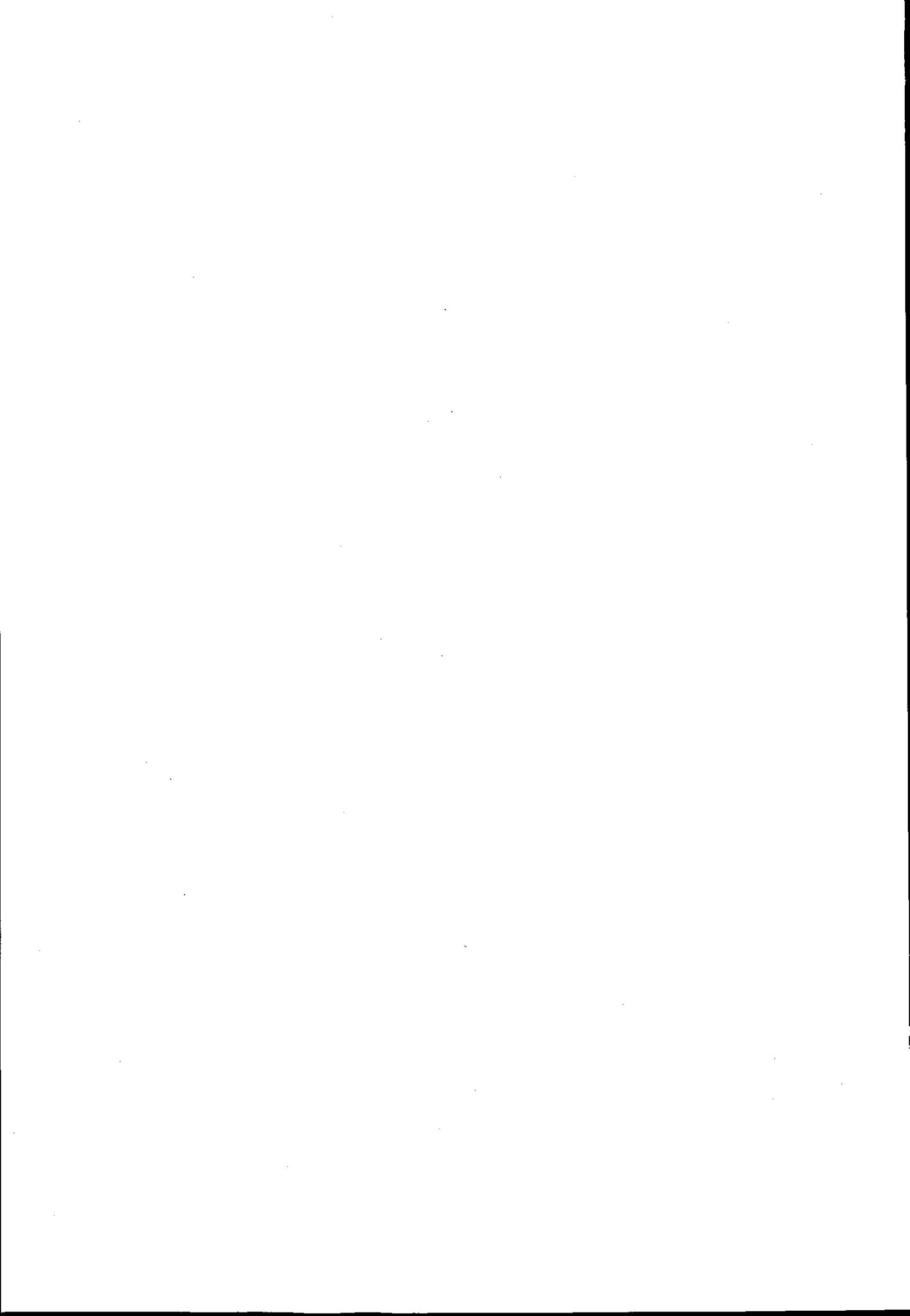
ゼロ件メッセージは、TYPE-Eについては、UNA、UNBおよびUNZセグメントだけで構成される電文（UNH、UNTセグメントの対が0件）である。

TYPE-Eのゼロ件メッセージでは、UNZセグメントの中の交換コントロール・カウンタは、ゼロ（メッセージ無）とする。



図11-6 ゼロ件メッセージの構造

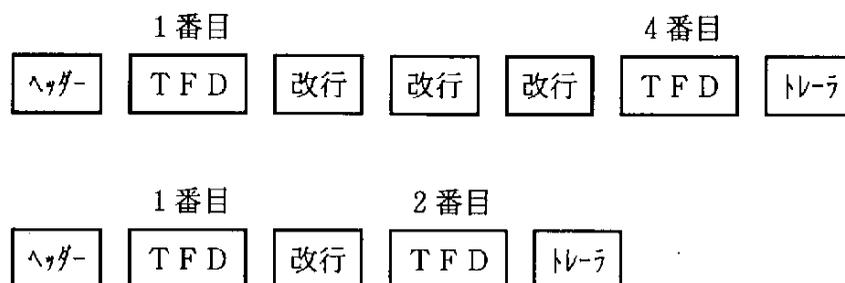
III 附 属 资 料



附.1 マルチ明細に関する補足

附.1.1 改行マークの省略

図A-1の上段と下段では、改行マークの省略以外は同一の構成であるが、意味は違っている。図A-1上段では、TFDの2番目と3番目が省略されたことを表すが、図A-1の下段では、TFDは1番目と2番目の2つしかないことを表している。すなわち、改行マークは通常省略できないことになる。しかしながら、図A-1の上段の構成で2番目と3番目のデータがなく、4番目のTFDが2番目として認識されてもかまわない時は、図A-1の下段の構成にすることが可能である。



図A-1 マルチ明細における改行マークの意味

附.1.2 マルチ明細全体の省略

一つのマルチ明細全体が空の時は、そのマルチ明細全体を省略できる。ここでいうマルチ明細全体とは、マルチ明細ヘッダーからそれに対応するマルチ明細トレーラまでをいう（マルチ明細の次のトレーラではない）。また空という意味は、マルチ明細ヘッダーからそれに対応するマルチ明細トレーラまでの間に、TFD制御子しかなかった場合である（有効データがない）。この一般則は、マルチ明細のネスティングがある時でも有効である。

省略可能であるというのは、省略しなければならないという意味ではない。受信用のトランスレータでは、省略の有無にかかわらず正しくメッセージを解析できなければならない。

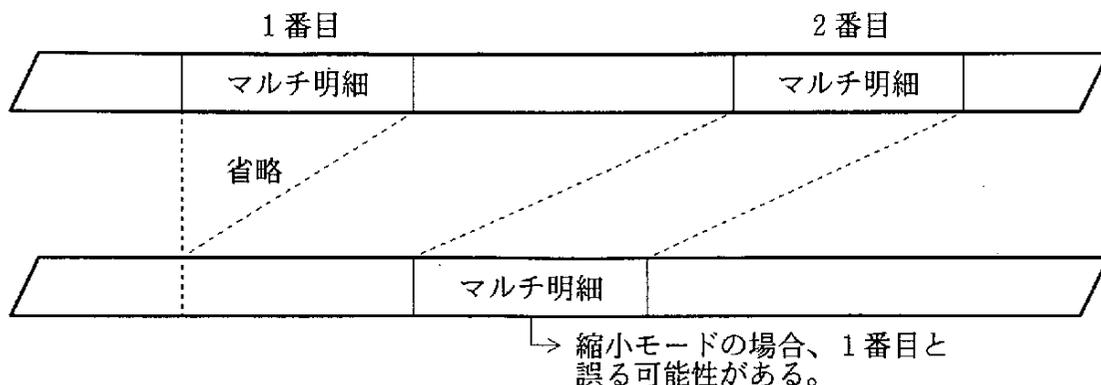
附.1.3 マルチ明細ヘッダーの明細番号の意味

拡張モードのマルチ明細ヘッダーには、明細番号がある。縮小モード(E I A J 互換モード)では、明細番号がない。

図A-2に示すように、一つのメッセージ内に2つの独立したマルチ明細がある場合、縮小モードでは、マルチ明細内の適当なTFDのタグ番号を用いて、マルチ明細の認識を行う。なぜなら、図A-2の左側のマルチ明細が省略された場合、右側のマルチ明細を左側のマルチ明

細として認識してしまう可能性があるからである。従って、右側のマルチ明細と左側のマルチ明細に同一のタグ番号を持つTFDを入れてはいけない。

拡張モードでは、明細番号を用いてマルチ明細を区別するので、このような制限はない。明細番号は、同一のメッセージ内のマルチ明細を区別するために用いるので、同一のメッセージ内でユニークな番号を付ければよい。



図A-2 マルチ明細全体の省略

附. 2 外字とJIS補助漢字 (JIS-X0212) について

CIIシンタックスルール1.11及び1.51では、JIS-X0201とJIS-X0208が標準として規定してある。実際の業務では、JIS-X0208では規定されていない漢字も少なくない。そこで、外字の設定が必要になるが、外字の文字コードはコンピュータの機種によりまちまちでm:nを想定したEDIのデータ交換には馴染まない。

そこで、外字の代わりにJIS-X0212 (補助漢字) を導入する。JIS-X0212と、JIS-X0208は、同一の文字コードになっているので、そのまま混用することはできない。CIIシンタックスルール2シリーズCオプションでは、以下のようにして、JIS-X0212を導入する予定である。

JIS-X0212 (補助漢字) の第1byteのMSBbitを1にする (第1byteにX'80'をORする)。

附. 3 ユニコード (J I S - X 0 2 2 1) について

ユニコードは、C I I シンタックスルール 2 シリーズの D オプションで導入される予定である。ユニコード導入時には、メッセージグループ・ヘッダーの 16bit 文字コード指定エリアにユニコードを表す識別子『U (X '55')』が、新たに追加になる予定である。

附. 4 E I A J シンタックスルールとの互換性

C I I シンタックスルールは E I A J シンタックスルールとの互換性を可能なかぎり配慮してあるが、C I I トランスレータと E I A J トランスレータとを組み合わせる場合には、以下の点について考慮しなければならない。以下の制限を守れば、通常、ユーザーは特別な考慮なしに C I I トランスレータと E I A J トランスレータとを混用できる。

(1) メッセージにおける制限 (縮小モード) を用いる。

- ① N 属性のデータ項目を使わない。
- ② 240 以上のタグ番号 (項目 No) を持つデータ項目を使わない。
- ③ 文字コードは、標準を用いる (J I S - X 0 2 0 1 と J I S - X 0 2 0 8)。C I I トランスレータでは標準を指定することで、この条件が満足される。E I A J トランスレータで文字コード変換のサポートがない場合は、ユーザー側で変換する。

受信用の C I I トランスレータでは、E I A J トランスレータで作成されたメッセージは無条件で標準文字コードを使用していると判断する。

- ④ 240 byte 以上の長さを持つデータ項目を使用しない。
- ⑤ 明細番号付マルチ明細を使わない。
- ⑥ マルチ明細のネスティングを使わない。
- ⑦ 最大メッセージ長を 32767 byte 以下にする。
- ⑧ インターナルセグメントを使わない。

(2) メッセージグループ・ヘッダーに関する考慮事項

トータル項目 No を指定する時は、必ず、小数点以下が 3 桁の 9 属性のデータ項目を指定する。

(3) 通信系に関する考慮事項

- ① 透過モードを使用する (透過モードの通信システムを使用する。)
- ② 分割モードを使用する。非分割モードを使用すると、分割区分の分だけ (1 byte) ずれを生ずる (C I I シンタックスルールでは、非分割モードの時も分割区分が付いているが、E I A J シンタックスルールでは、非分割モードの時は、分割区分がない)。

(4) その他

E I A J シンタックスルールの使用では、メッセージグループ・ヘッダーやトレーラも含

めて、伝送電文全体の文字コード（バイナリー部分を除く）をEBCDICコードとして運用している例がある。このケースでは、特別のEIAJ対応CIIトランスレーターでなければ、CIIシンタックスルールとの共通運用ができないので、注意を要する。

附. 5 非透過モード・オプションについて

非透過モードは、J手順等の古い非透過モードの通信システムのために設けられたオプションで、可能なかぎり、全銀手順やOSI新手順でネットワークを構成し、非透過モードは使用しないことが望ましい。CIIシンタックスルール2シリーズでは、廃止される予定である。

附. 6 CIIシンタックスルールのバージョン番号について

CIIシンタックスルールのバージョン番号は、3桁で表現される。

- ① 一番右側の桁は、トランスレーターには影響を及ぼさない修正があった時、変更される。主にシンタックスルールの矛盾の修正や本書（規格書）の記述の誤り修正の場合に変更される。
- ② 中央の桁は、トランスレーターのマイナーチェンジが必要になるようなシンタックスルールの修正があった時、変更される。
- ③ 一番左側の桁は、シンタックスルールの大変更があった時、変更される。新しいバージョン対応のトランスレーターが必要になる。

下位2桁の変更では、新バージョンが旧バージョンを包含する方式をとる。しかしながら、上位1桁の変更では、必ずしも新バージョンが旧バージョンを包含しない場合もある。

インタラクティブEDI用シンタックスルールとバッチ処理用シンタックスルールが包含関係にならない場合、2つのバージョンのシンタックスルールが同時運用されることになる。

附. 7 Y属性について

附. 7.1 目 的

従来型のシステム設計では、年月日は、数字6桁で設計するのが普通であったが、最近、2000年対策のために、数字8桁で設計するケースが増加している。

このことは、業界標準化における標準メッセージの設計にも影響し始めており、最近のメッセージ設計では、数字8桁とするのが多い。

6桁表示の年月日と8桁表示の年月日は、そのままでは互換性がないので、ユーザーは特別の対策が必要となるし、2000年対策のためだけに、8桁表示が必要になるというのも通常の事務処理システムでは無駄が多い。歴史のように、数千年レンジの処理が必要な場合はともかくとして、事務処理システムのように高々数十年の処理で済む場合には、4桁表示は、いか

にも冗長である。本規格の追加は、これらの問題の解決を目的としている。

附. 7. 2 基本的な考え方

現在1995年であるから、通常のシステムの寿命が20年程度とすれば、これから20年以内に構築するシステムでは、2030年まで有効な日付表現が設定されていれば、十分であると結論できる。

少し余裕を見て、2050年まで有効な日付表現を採用すれば、業務処理上の問題点はないとすることができる。一方、C I I シンタックスルールの寿命は、永遠であるが、2050年という60年後以後も有効かどうかを議論することは、あまり意味がないだろう。コンピュータ実用化以来まだ40年程の歴史しかないし、この間にも、何回かシステムの技術的体系は変化してきた。この面からも2050年まで有効な方式によりシステムを構築することは、妥当な選択と言える。

附. 7. 3 基本設計

日付表現の有効範囲を、1951年～2050年までとし、6桁表現と8桁表現の互換性を図る特別な変換をトランスレータに設けることにより、6桁表現のシステムと8桁表現のシステム間のEDIを実現するものである。

日付の6桁表現と8桁表現を以下のように、対応させる。

6桁表現		8桁表現
5 1 0 1 0 1	←————→	1 9 5 1 0 1 0 1
}		}
9 9 1 2 3 1	←————→	1 9 9 9 1 2 3 1
0 0 0 1 0 1	←————→	2 0 0 0 0 1 0 1
}		}
5 0 1 2 3 1	←————→	2 0 5 0 1 2 3 1

図A-3 6桁表現と8桁表現の対応

図A-3から、8桁から6桁への変換は、単純に、上位2桁をカットすればよいことが分かる。逆に6桁から8桁への変換では、

51年～99年の時（51以上の時）……… 19を追加する
00年～50年の時（50以下の時）……… 20を追加する

のようにする必要のあることが分かる。

このような取扱をする属性をY属性として新たに定義し、シンタックスルールに追加した。

附. 7. 4 トランスレーターでの対応

(1) 送信用トランスレーター

Y属性は基本的に数値（9属性の特別ケース）であるから、9属性と同一の変換を行う。

- ① ローカルエリアは一般的に、整数で、6桁か8桁でなければならない。
- ② ローカルエリアが2進表現の場合には、等価的に6桁か8桁でなければならない。

変換は、9属性と同一の変換である。したがって、送信側ではY属性を9属性で定義してもよい。

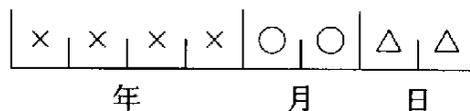
(2) 受信用トランスレーター

Y属性は、基本的に数値（9属性の特別ケース）であるから同一の変換を行えばよいが、Y(6)とY(8)の互換性を保つ処理が必要になる。

- ① ローカルエリアは一般的に、整数で、6桁か8桁でなければならない。
- ② ローカルエリアが2進表現の場合には、等価的に6桁か8桁でなければならない。

変換は、以下のように行う。

- ① 必要であれば、文字コードを変換する。
- ② 受信したY属性のデータは、8 byteに変換する。この時8 byteに満たないデータは、上位桁（左側）にゼロを追加して、8 byteにする。
- ③ ②の処理の結果、年相当部分が、0000～0050の時は、上位1桁を2として、2000～2050とする。年相当部分が、0051～0099の時は、上位2桁を19として、1951～1999とする。
- ④ ③の処理の結果、ローカルエリアが6桁の時でも8桁の時でも、以下のような形になる。



- ⑤ ローカルエリアの形式に変換する。

この時、ローカルエリアが6桁の時は、左側2桁をカットする。

(3) その他のトランスレーターでの対策事項

- ① 拡張モード指示

拡張モード指示は、Y変換の存在によって、特に影響を受けない。

相手側のトランスレーターがEIAJトランスレーターの場合やその他様々なケースでの互換性は、図A-4のようになる。

② エラーコードについて

『36 : 不正日付の検出 (Y変換で検出)』が新たに追加された。

○印は可、×は互換性なし

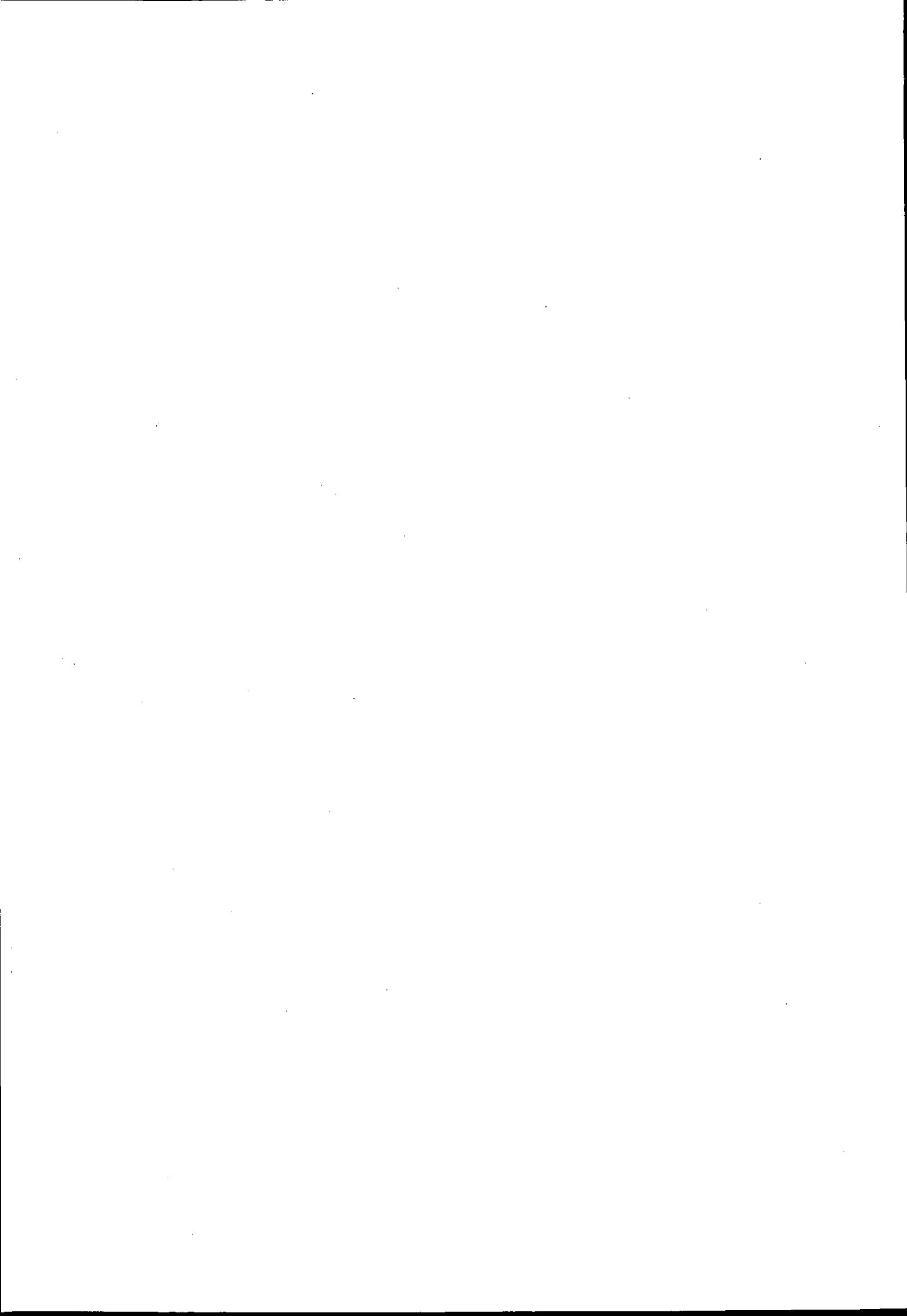
	C I I		E I A J		C I I		C I I
○	9 (6)	→	9 (6)	○	9 (6)	→	9 (6)
×	9 (6)	→	9 (8)	×	9 (6)	→	9 (8)
○	Y (6)	→	9 (6)	○	9 (6)	→	Y (6)
×	Y (6)	→	9 (8)	○	9 (6)	→	Y (8)
△注1	9 (8)	→	9 (6)	○	Y (6)	→	9 (6)
○	9 (8)	→	9 (8)	×	Y (6)	→	9 (8)
△注1	Y (8)	→	9 (6)	○	Y (6)	→	Y (6)
○	Y (8)	→	9 (8)	○	Y (6)	→	Y (8)
○	9 (6)	←	9 (6)	○注2	9 (8)	→	9 (6)
×	9 (8)	←	9 (6)	○	9 (8)	→	9 (8)
○	Y (6)	←	9 (6)	○	9 (8)	→	Y (6)
○	Y (8)	←	9 (6)	○	9 (8)	→	Y (8)
○注2	9 (6)	←	9 (8)	○注2	Y (8)	→	9 (6)
○	9 (8)	←	9 (8)	○	Y (8)	→	9 (8)
○	Y (6)	←	9 (8)	○	Y (8)	→	Y (6)
○	Y (8)	←	9 (8)	○	Y (8)	→	Y (8)

注1) E I A J トランスレーターに強制カットの機能があれば、○

注2) 強制カットを指定する。

注3) 9タイプは、いずれの場合も日付を表すとする。

図A-4 Y属性をサポートしていないトランスレーターとの互換性



禁 無 断 転 載

平成 8 年 7 月 発行

発行所 財団法人 日本情報処理開発協会
産業情報化推進センター
東京都港区芝公園 3 丁目 5 番 8 号
機 械 振 興 会 館 内
TEL(3432) 9 3 8 6

印刷所 株式会社 タ ケ ミ 印 刷
東京都千代田区神田司町 2 - 16
TEL(3254) 5 8 4 0 (代)





