

CIIシンタックスルール

試作仕様 1.02

平成 3 年 9 月

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

1. The first part of the paper discusses the importance of the study of the history of the United States. It is argued that a knowledge of the past is essential for a full understanding of the present and for the development of a sound policy for the future.

2. The second part of the paper discusses the importance of the study of the history of the United States. It is argued that a knowledge of the past is essential for a full understanding of the present and for the development of a sound policy for the future.

この資料は、日本自転車振興会から競輪収益の一部である機械工業振興資金の補助を受けて、平成3年度に実施した「ビジネスプロトコル等に関する調査研究」の一環としてとりまとめたものであります。

はじめに

昭和62年から、EDIの業界標準化作業を開始した(社)日本電子機械工業会(EIAJ)と(財)日本情報処理開発協会、産業情報化推進センター(CII)は、共同で、昭和63年に新しいEDI標準の開発を行った。この標準は、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、米国で採用されていた可変長フォーマットをベースにすることになった。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されることになったが、結果的に、米国のANSI X.12や当時審議中であったEDIFACT(ISO 9735)と、同一構成になっている。

シンタックスルールは、(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グ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った。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と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)が行われたが、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。保留された機能として、もっとも大きな問題を発生したのは、最大使用可能データ項目数が239種ということである。

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、なんとかメッセージの設計が可能であるが、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というのが、大きな問題になった。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E I A J標準の普及を考慮し、すでに表面化してきた不具合点も含めて、E I A J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を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、とくに留意された。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』として公開された。その後、C I Iシンタックスルールの採用を決めた建設業界からの要望などを追加し、現在『C I Iシンタックスルール試作仕様1.02』となっている。

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、業界間や業際の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。

C I Iシンタックスルールは、現在、実使用テストの準備が進められている。このテストは、平成4年3月末までに完了する予定であり、平成4年4月に『C I Iシンタックスルール1.10』として、商用使用可能なシンタックスルールとして全面公開される。

さらに、平成4年4月以後、複数のベンダーから、C I Iシンタックスルールをサポートする『トランスレーター』も一般販売される予定である。尚、通信ネットワークについては、E I A J標準をサポートするV A Nサービスが利用可能であり、平成4年4月以後は、C I IシンタックスルールをフルサポートするE D Iサービスも、開始されるであろう。

『C I I シンタックスルール試作仕様 1. 0 2』

1991. 8. 28

(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、(株)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。また、今後予想される国際標準 U N / E D I F A C T における I S O 9 7 3 5 の普及に対して、E D I ネットワーク上での共存が図れる機能も設定されている。

この試作仕様は、トランスレーターを試作するための仕様であり、シンタックスルールとしては暫定版である。但し、トランスレーターの試作とそれに続く試行使用において、不具合点がなければ、そのまま C I I シンタックスルール 1. 1 0 として、実運用に入ることが予定されている。

この試作仕様の管理は、当センターである(株)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、著作権も当センターに帰属する。当センターに断りなく本試作仕様の一部を変更した仕様は、変更の程度にもよるが、本試作仕様と同一の仕様と判断され、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、本試作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、当センターの許可なくして C I I シンタックスルール準拠の表示を製品に行うことを禁止する。

本試作仕様に基づく『C I I シンタックスルール 1. 1 0』の完全な公開は、1992年4月1日以降である。それまでは、予告のない変更が行われることに、注意されたい。

この試作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

(株)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

THE HISTORY OF THE UNITED STATES

OF THE

UNITED STATES OF AMERICA, FROM THE FIRST SETTLEMENTS TO THE PRESENT TIME. IN THREE VOLUMES. VOL. I.

BY JAMES OSGOOD, ESQ.

NEW-YORK: PUBLISHED BY J. OSGOOD, 15 NASSAU ST. 1854.

NEW-YORK: 1854.

THE HISTORY OF THE UNITED STATES, FROM THE FIRST SETTLEMENTS TO THE PRESENT TIME. IN THREE VOLUMES. VOL. I.

BY JAMES OSGOOD, ESQ.

NEW-YORK: PUBLISHED BY J. OSGOOD, 15 NASSAU ST. 1854.

NEW-YORK: 1854.

THE HISTORY OF THE UNITED STATES, FROM THE FIRST SETTLEMENTS TO THE PRESENT TIME. IN THREE VOLUMES. VOL. I.

BY JAMES OSGOOD, ESQ.

NEW-YORK: PUBLISHED BY J. OSGOOD, 15 NASSAU ST. 1854.

THE HISTORY OF THE UNITED STATES, FROM THE FIRST SETTLEMENTS TO THE PRESENT TIME. IN THREE VOLUMES. VOL. I.

BY JAMES OSGOOD, ESQ.

NEW-YORK: PUBLISHED BY J. OSGOOD, 15 NASSAU ST. 1854.

目 次

I 概 要	1
II 詳 細	10
1. キャラクターセットと文字コード	11
2. CIIシンタックスルールの詳細 (TYPE 12)	12
2-1 定 義	12
2-2 メッセージグループ・ヘッダー	18
2-3 メッセージ (可変長) の構造 (TYPE 12)	20
2-4 メッセージの格納構造 (TYPE 12 分割モード)	30
2-5 メッセージグループ・トレーラ	31
2-6 非透過モード時の文字コードの変換およびバイナリーデータの補正	32
3. CIIシンタックスルールの詳細 (TYPE-E)	33
3-1 交換の階層	33
3-2 UNAセグメントとUNBセグメント	33
3-3 メッセージの構造 (TYPE-E)	34
3-4 UNZセグメント	36
3-5 メッセージの格納構造 (TYPE-E)	36
4. システム運用電文フォーマット	37
III トランスレーターの概要	40

寒 風

「寒風」は、冬の風を指す。冬は一年の中で最も寒い季節で、

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

寒風が吹くのは、冬の始まりを告げるサインである。

I 概 要

『C I I シンタックスルールの概要』

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／制御タグ方式）
- ⑤ 我が国の E D I の標準化に十分な、最大 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 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 オプション機能の活用により、O S I 新手順（F 手順および M 手順）をはじめ、全銀手順、J C A 手順など、あらゆる通信システムに対応できる（図 1－1 参照）。
- ⑧ 将来普及が予想される I S O 9 7 3 5 との並行使用のためのオプションを最初から持っている。このオプションでは、I S O 9 7 3 5（UN／E D I F A C T）に合致する伝送形態になり、一つのネットワーク上で両方のメッセージを取り扱うことが可能。
- ⑨ 現在の 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。
- ⑩ 以上の特徴を十分に発揮させるモデルトランスレーターのサポート（シンタックスルール規格書の付属資料として、モデルトランスレーターの仕様がある）。

尚、今後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。例えば、本試作仕様により試作されるモデルトランスレーターでは、取り扱えるデータ項目の最大長は、4 0 0 0 字（漢字の場合は、2 0 0 0 字）の予定である。

2. 基本構造

C I Iシンタックスルールは、ファイル転送で実現するE D Iのために設計されており、その基本形態は、TYPE 1 2と称する一つのメッセージグループ・ヘッダー、複数のメッセージ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1-2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、2つのバリエーションがある。一つは、TYPE 1 2の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（図1-2の中段を参照）であり、もう一つは、TYPE-Eと称するISO 9 7 3 5構造を応用した形である（図1-2の下段を参照）。以下、順に概要を述べる。

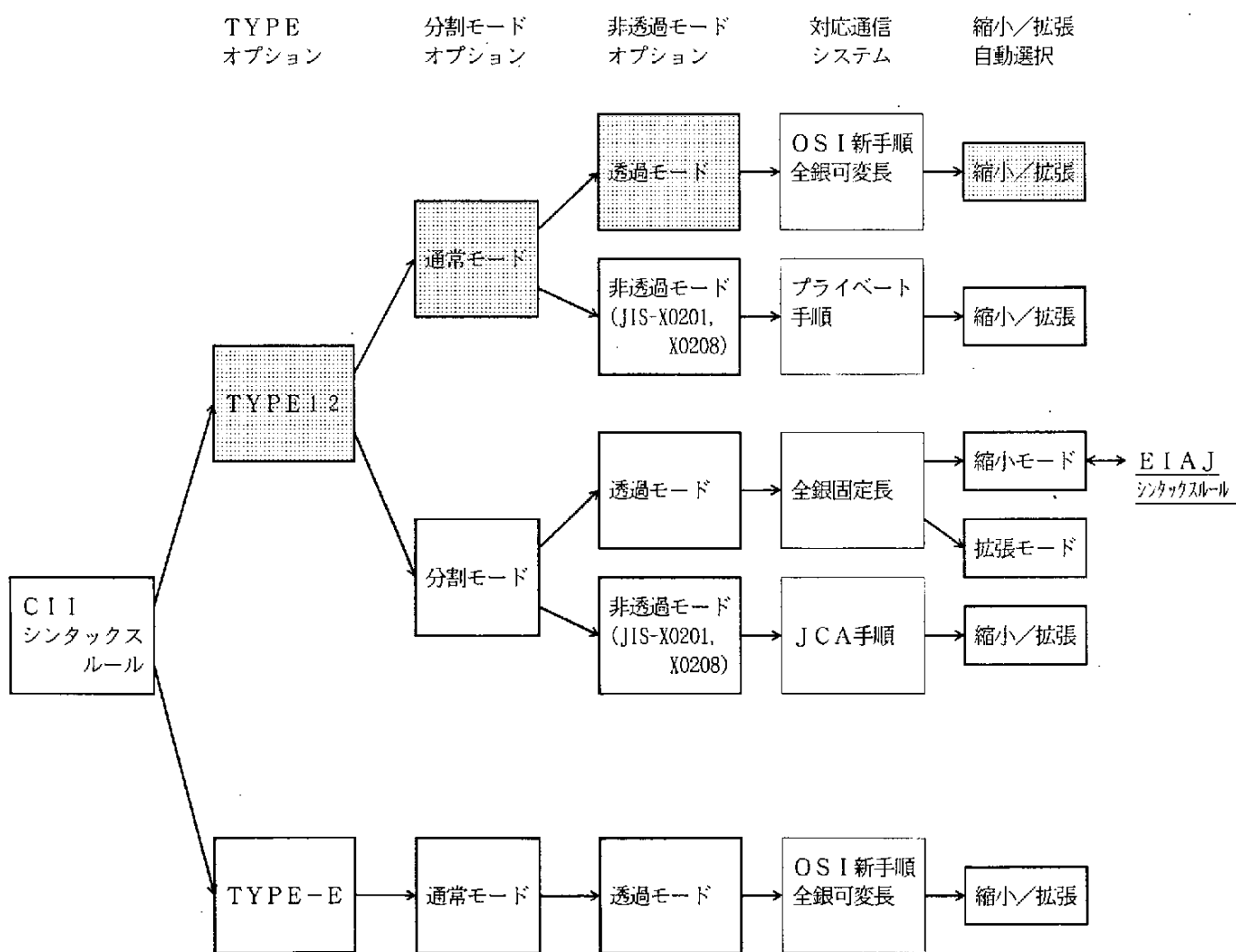


図1-1 C I Iシンタックスルールのオプションと通信システムとの関係
(注) 網かけ内が規定値（標準）

2-1 TYPE12 (通常モード)

図1-2の上段の構造で、CIIシンタックスルールの基本形である。その交換の階層は、図1-3で、示される。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251byte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。

2-2 TYPE12 (分割モード)

図1-2の中段の構造であり、基本形式(通常モード)と基本的には同一で、交換の階層も通常モードと同じく図1-3で表される。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、一つのメッセージは複数の251byte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)の時は、EIAJシンタックスルールと互換性(同一)がある。

2-3 TYPE-E

図1-2の下段の構造である。基本形式のメッセージグループ・ヘッダーが、ISO9735のUNA及びUNBセグメントに置き換えられ、メッセージグループ・トレーラが、UNZセグメントに置き換えられている。そして、メッセージは、UNH、CII、UNTセグメントに置き換えられているが、CIIセグメントのデータ部分は、基本形式のメッセージと同一である。このことにより、UN/EDIFACT電文(ISO9735でコーディングされている)と同一のネットワークでEDIを実施することができる。TYPE-Eの交換の階層は、ISO9735の交換の階層と同一である。UNAセグメントとUNBセグメントは一つの可変長レコードに格納され、他のセグメントは、一つのセグメントが一つの可変長レコードに格納される。

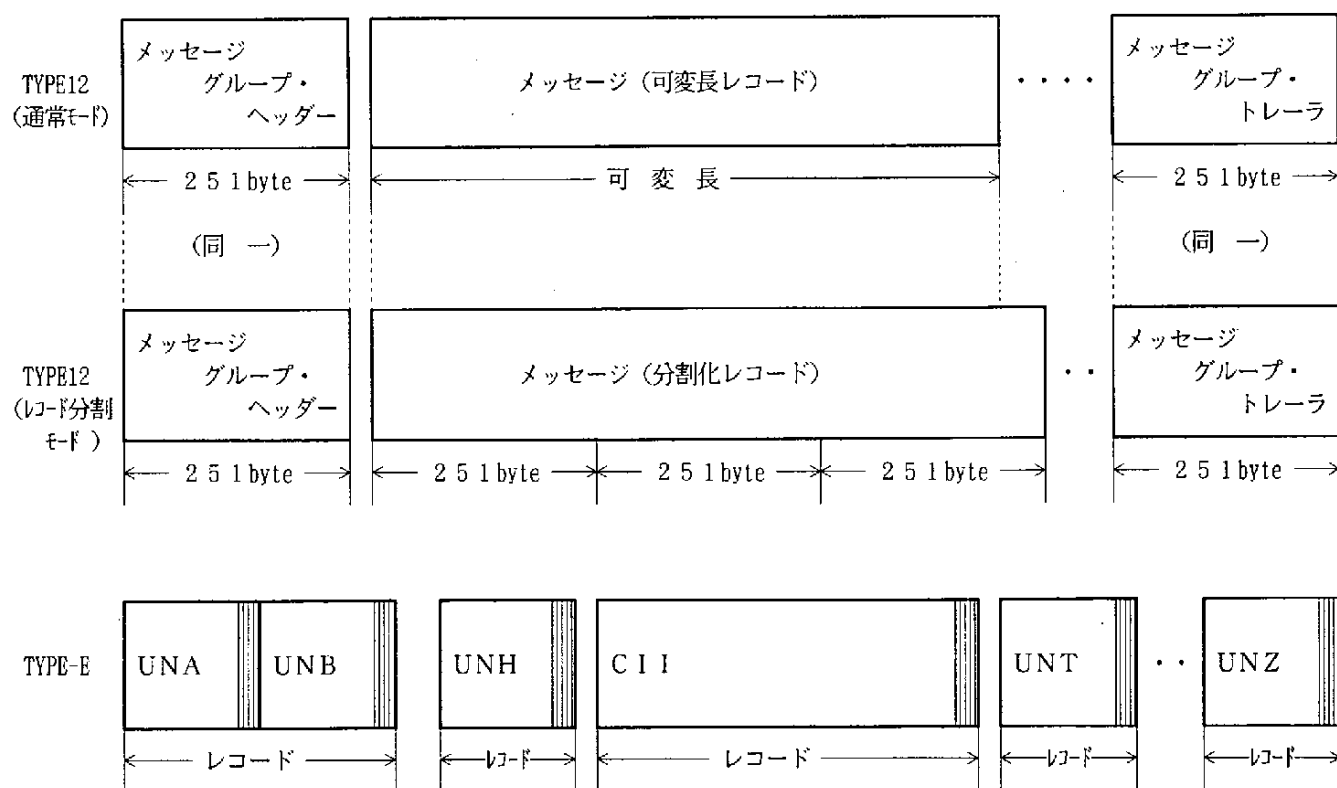


図1-2 TYPE12 (基本形、分割モード) 及びTYPE-Eの伝送形態

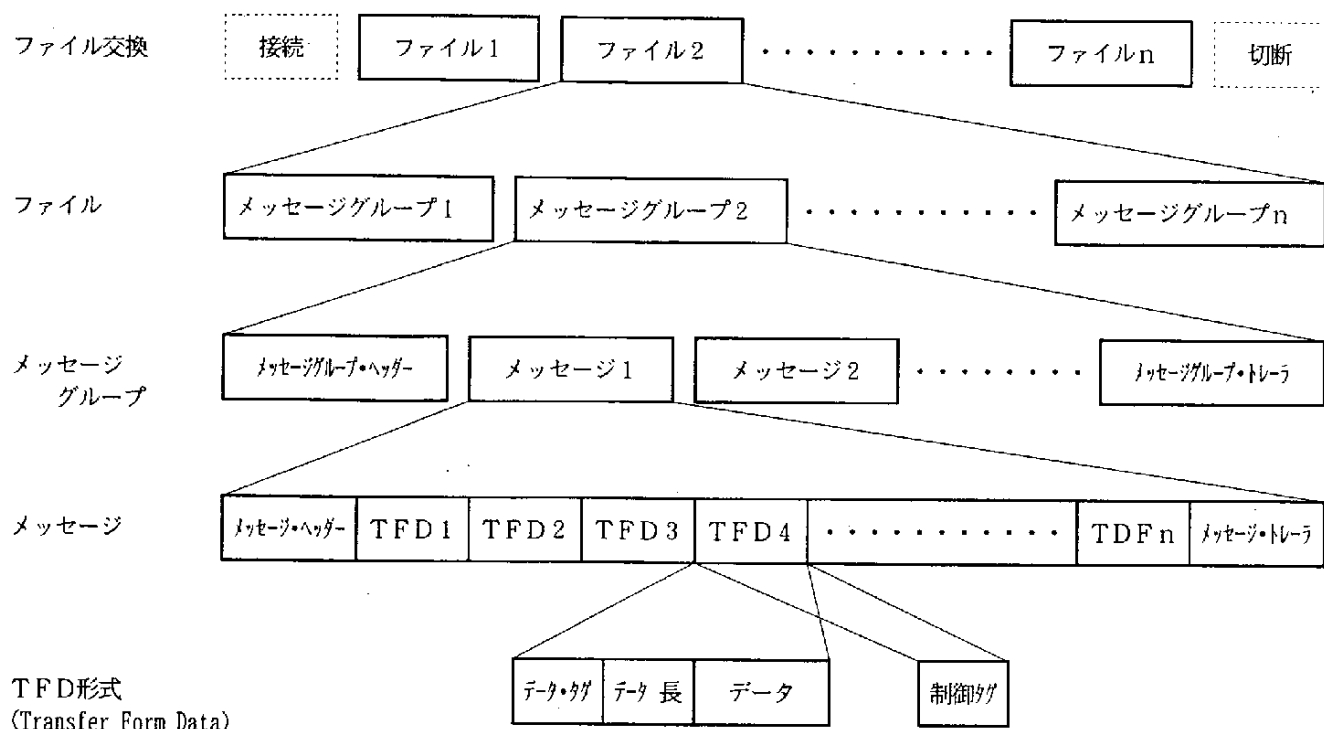


図1-3 交換の階層 (TYPE 12)

2-4 TYPE 12とTYPE-Eの違い

TYPE-Eは、TYPE 12のメッセージグループ・ヘッダがISO 9735のUNA, UNBセグメントに置き換えられ、メッセージグループ・トレーラがUNZセグメントに置き換えられる。一つのメッセージは、メッセージ・ヘッダがUNHセグメントに置き換えられ、メッセージ本体はセグメント・タグ (CII) がセットされる以外はTYPE 12のメッセージがそのまま用いられ、UNTセグメントが追加される (図1-4参照)。

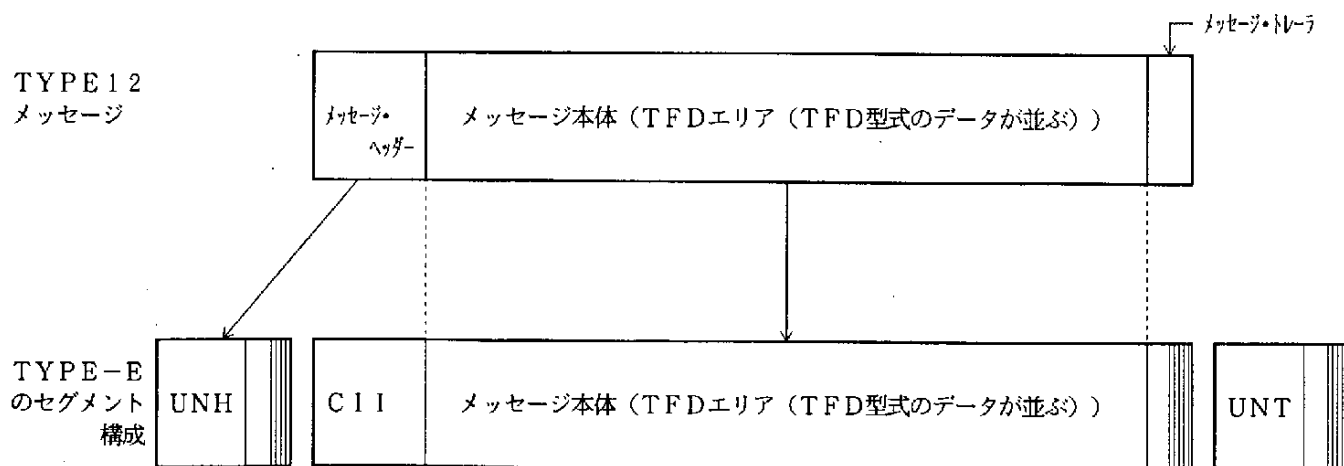


図1-4 TYPE 12メッセージのセグメント化

3. 縮小モードと拡張モード

CII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1-5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。

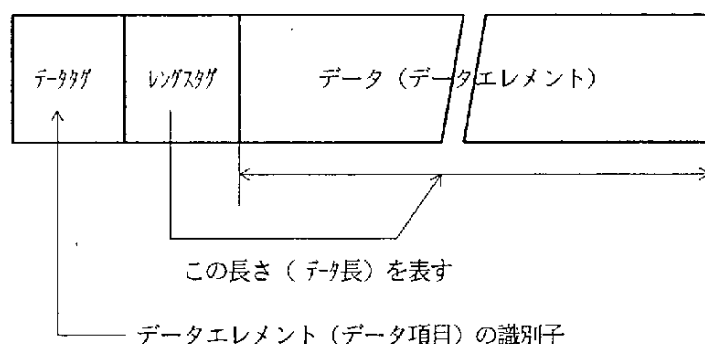


図1-5 TFDの構造

『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(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(61439種のデータエレメントを表示できる)があり、CII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1-6参照)。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EIAJシンタックスルールと同一になり、互換性の確保にも活用される。

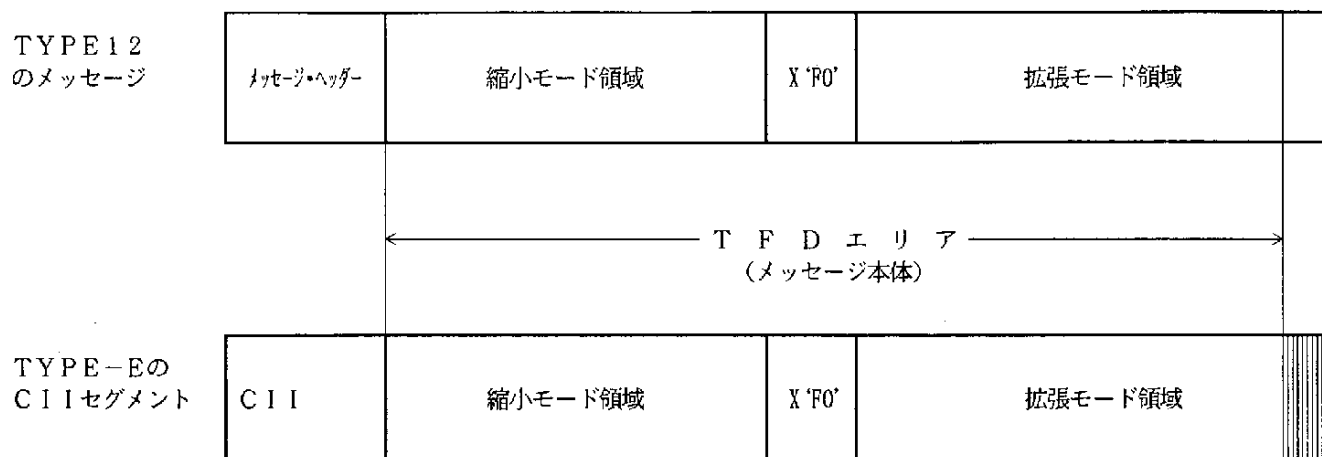


図1-6 縮小モードと拡張モード(メッセージ本体)

4. 非透過モードオプション

このオプションは、非透過モードの通信回線を使用する時に用いる。CII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。

5. 文字コード

CIIシンタックスルールでは、JIS-X0201及びJIS-X0208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・トレーラーでは、英数字とブランク以外の文字の使用は禁止され、文字コードもJIS-X0201の使用が義務づけられる。TYPE12のメッセージについては、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。JIS-X0208（JIS第1水準、第2水準漢字）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、メッセージ（CIIセグメント）も含めてすべてのセグメントについて、JIS-X0201、JIS-X0208の文字コードを使用しなければならない。JIS-X0208（JIS第1水準、第2水準漢字）は、CIIセグメント内でのみ使用できる。

6. トランスレーター

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、オプションの設定で、TYPE12、TYPE-Eのどのメッセージも作成することができる。

また、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 明示的マルチ明細の検出

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。この時、EIAJシンタックスルールと完全に互換性のあるメッセージになる（TYPE12の時）。

受信用のトランスレーターは、TYPE12及びTYPE-Eのメッセージに加えてEIAJシンタックスルールのメッセージでも、すべて自動的に検出し、同一の変換テーブルを用いてメッセージの解釈を完全にできる。

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JIS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。

6-2 モデル・トランスレーター

トランスレーターは、EDIシステムにとって、最も重要な要素である。CII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API（アプリケーション・インターフェース）の標準化である。APIは、ユーザーのコンピュータ・システムの構成やEDI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API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、もちろん実システムに適用可能であるが、運用効率（特に、使いがって）が悪く、実用的システムとしては問題が残る。しかしながら、トランスレーター本来の機能はすべて備えており、ユーザー環境に応じた若干の改造（APIの変更）を行うことで、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CII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・トランスレーターは、コンピュータの機種によらず同一の機能を保持す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、EDI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 Smart TRanSlator for Encoding standard format (STRSEMA)
- b. データ受信用 Smart TRanSlator for Decoding standard format (STRSDMA)

② パソコン用モデル・トランスレーター

- a. データ送信用 Smart TRanSlator for Encoding standard format (STRSEPA)
- b. データ受信用 Smart TRanSlator for Decoding standard format (STRSDPA)

である。図1-7は、メンフレーム用モデル・トランスレーター（送信用）の構成である。

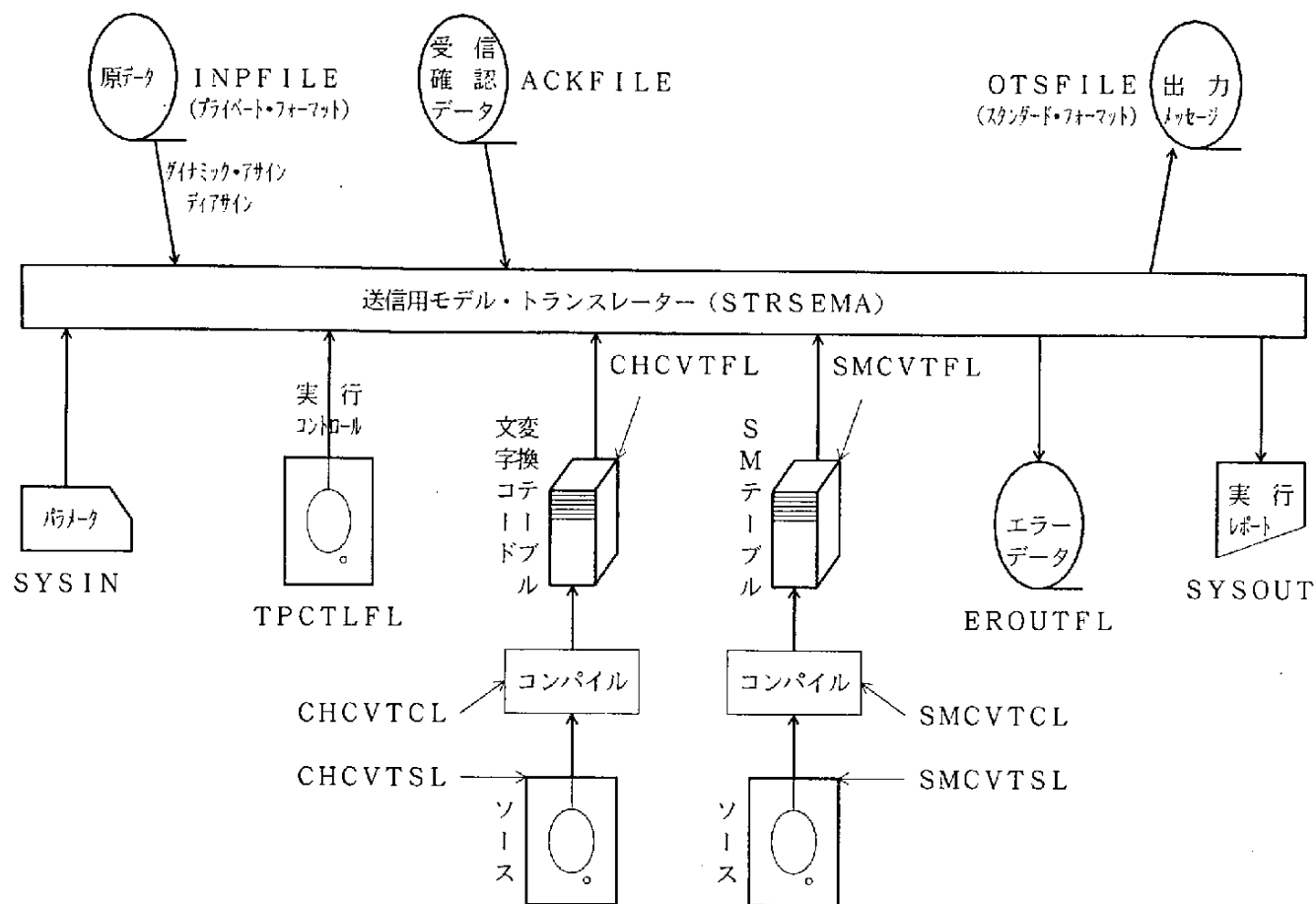


図1-7 送信用モデル・トランスレーターの構成 (メインフレーム用)

8 C I I シンタックスルールによるシステムの構築

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の上位互換の規格であるため、E I A J 標準によるシステムとの共存を図るための特別な工夫は、通常、不要である。以下では、この状況について説明を行う。この説明のために、一つの仮定を行う。それは、E I A J 標準メッセージには、当分の間、項目No.240以上の項目と、N、K属性のデータエレメントの導入が行われないということである。もしこの仮定が成り立たない時は、E I A J シンタックスルールそのものがC I I シンタックスルールに改訂される時であるから、特別な工夫は、当然、不要になる。

図1-8の状況を設定する。E I A J 標準を用いている業界 (E業界) の他にA、Bの二つの業界がある。E業界は、早くからE I A J 標準による業界内E D Iを実施している。A業界が新たにE業界とE D Iを実施することになったが、そのための標準メッセージ (SM) は、項目No.240以上の項目と、N、K属性のデータエレメントは導入しないで作成された。E業界側の企業は、既に使っているE I A J トランスレーターを使用することになったが、A業界側の企業は、新規のシステムなので、C I I トランスレーターを使用することになった。

その後さらに、B業界も新たにE業界とE D Iを実施することになったが、こちらのSMは、B業界の強い要望もあり、K属性 (漢字) のデータエレメントを導入することになった。幸いなことに、B業界とE D Iを行うE業界側の企業は比較的小数の大企

業だったので、既に使用していたEIAJトランスレーターに加えてCIIトランスレーターを追加することにした。ごく一部の企業では、すべてCIIトランスレーターに変更してしまった。B業界はCIIトランスレーターを導入した。

このようなEDIの進展が、さらに、A業界とB業界のEDIも構築させることになったが、両業界ともCIIトランスレーターを導入済みであり、スムーズにEDIが構築された。

以上の過程は、CIIシンタックスルールの上位互換性が完全にカバーしており、ユーザー（各企業）は、シンタックスルールの違いに関する考慮は、一切不要である。

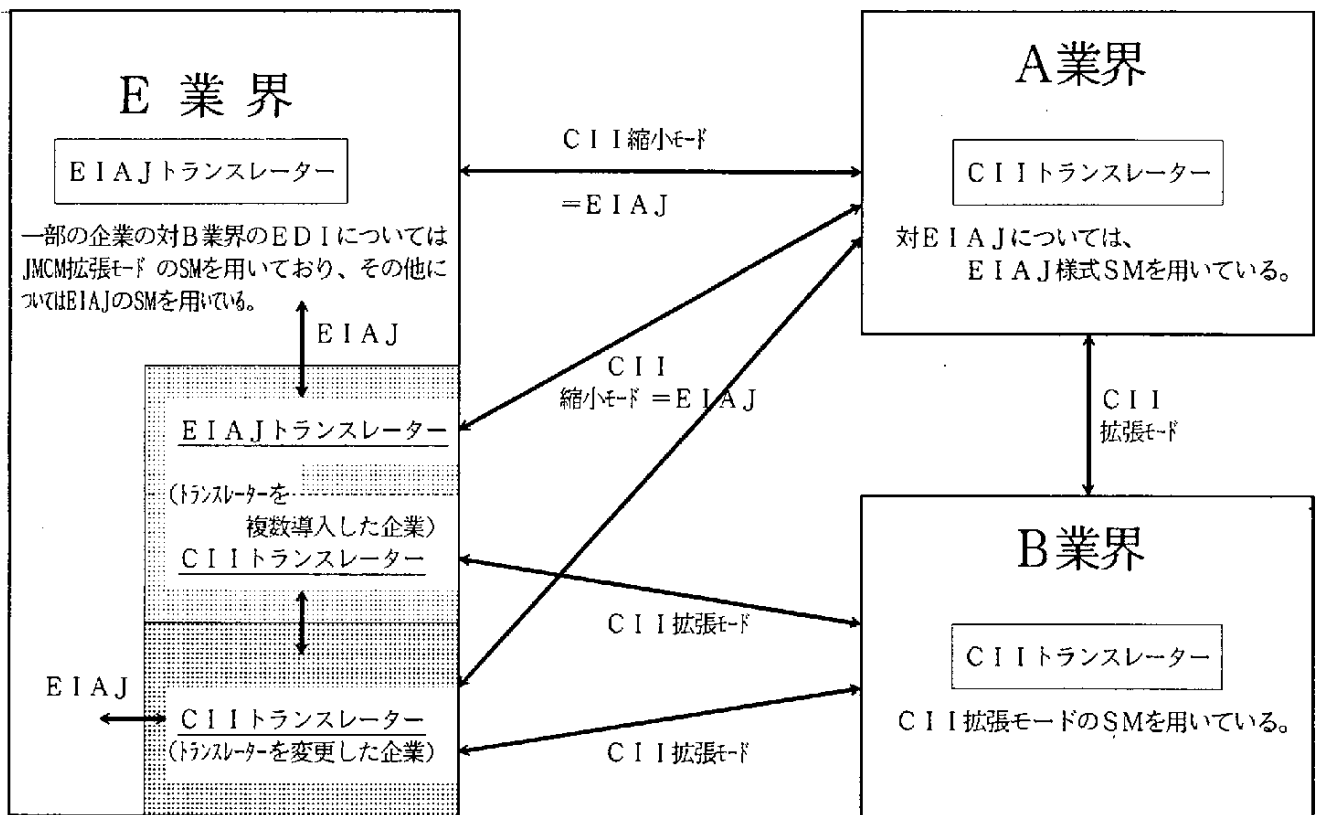
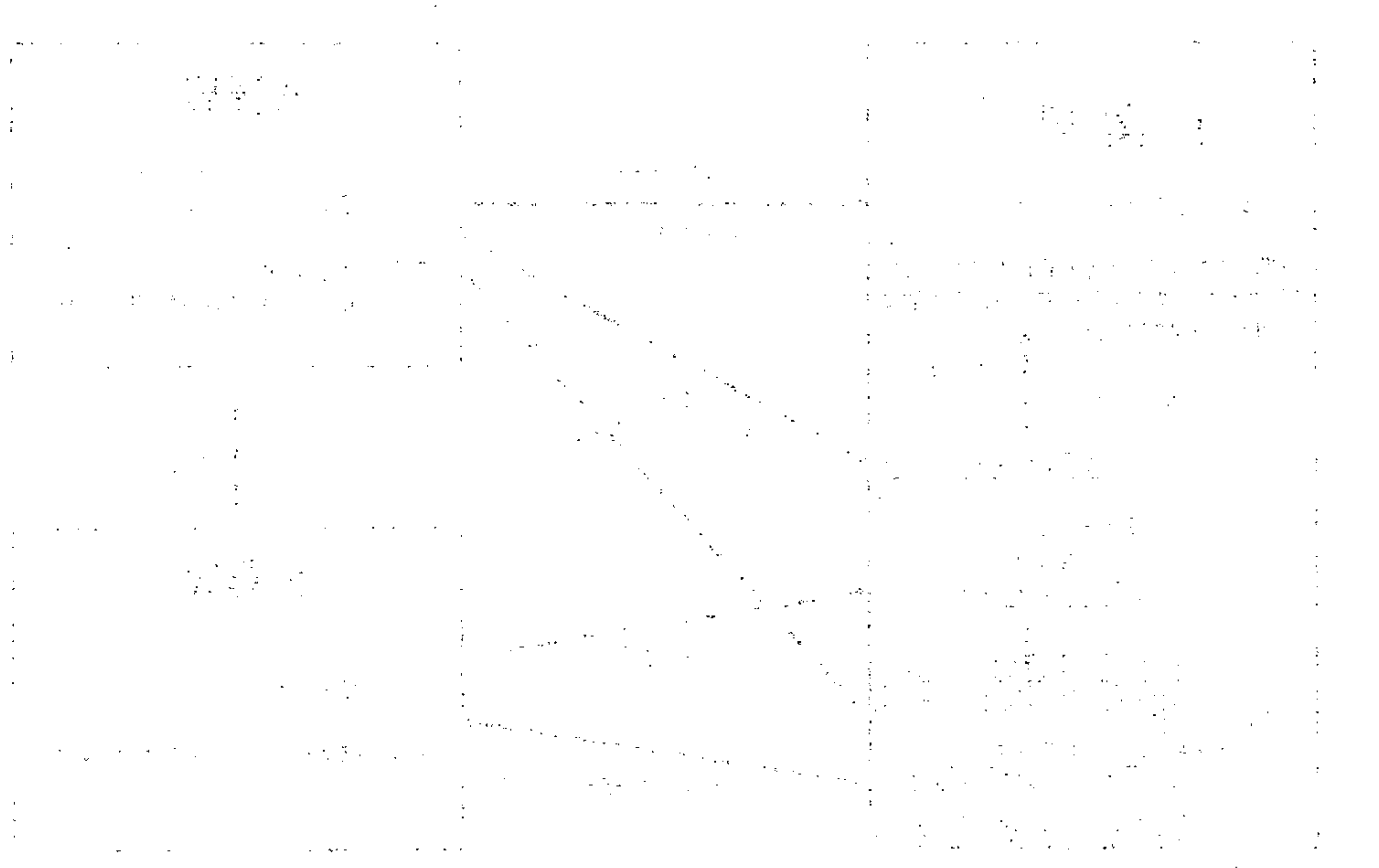


図1-8 EIAJシンタックスルールとCIIシンタックスルールの共存

9 TYPE-Eの応用

TYPE-Eは、将来のPEDI (EDI-MHS) 網のための形式である。但し、TYPE-EはISO 9735に準拠したメッセージではあるが、国連の標準メッセージ (UNSM) には準拠していない。TYPE-EからUNSM準拠のメッセージを生成するために、CII-UNSMコンバーターが必要である。現実の業務では、そのような必要性はないと考えられるが、もし必要であれば、技術的に可能であり、UNSMの第1バージョンが確定し日本国内サブセット制定後に、CII (TYPE-E) -UNSMコンバーターが開発されることになろう。但し、この日本国内サブセットは、漢字項目を始めとする多くのローカル規定が含まれた通常のUNSMサブセットとは、かなり異なるものとなることが予想される。上記のコンバーターは、単なるセグメントの再構成が主な機能となる。

The first part of the report deals with the general situation of the country and the progress of the work. It is followed by a detailed account of the work done during the year, and a summary of the results. The report is divided into two main parts, the first of which deals with the general situation of the country and the progress of the work, and the second with a detailed account of the work done during the year, and a summary of the results.



The second part of the report deals with the detailed account of the work done during the year. It is divided into two main parts, the first of which deals with the general situation of the country and the progress of the work, and the second with a detailed account of the work done during the year, and a summary of the results. The report is divided into two main parts, the first of which deals with the general situation of the country and the progress of the work, and the second with a detailed account of the work done during the year, and a summary of the results.

Ⅱ 詳 細

『C I I シンタックスルールの詳細』

1. 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

CII 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2-1は、CII シンタックスルールで
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、ローカル協定によるこれ以外の文字の使用も
不可能ではない。

16bit 文字は、別途資料『CII 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JIS-X0201（8bit 文字、表1-1にコードを示す）とJIS-X0208（16bit 文字）
を用いる。しかしながら、ローカル協定により、これ以外の文字コードを用いることも不可能ではない。

表2-1 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	°	。
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・ヘッダーとメッセージグループ・トレーラでは、表2-2に示す8bit 文字のみ使用可能で、かつ、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。

表2-2 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
網掛けの部分は使用禁止

2. CII シンタックスルールの詳細 (TYPE 12)

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。

2-1 定 義

2-1-1 固定長データエレメント

右側のブランクの省略を行わない、データ長が常に固定されている文字列データエレメントである。メッセージグループ・ヘッダーとメッセージグループ・トレーラで用いられ、使用可能な文字セットと文字コードは、表2-2で示される。

2-1-2 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うデータエレメントである。このデータエレメントは、メッセージ内のTFDエリア中で、TFD形式データ構成要素として用いられる。以下の4種がある。

(1) 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、整数部に有効桁が有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。尚、本データエレメントでは8bit 文字を用いる。

(2) 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。したがって、『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。尚、本データエレメントでは8bit 文字を用いる。

(3) 8bit 文字列 (X属性)

最大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blank以外の文字よりもさらに右側にあるblank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。

(4) 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blank（16bit 文字のblank、以下同じ）以外の文字よりもさらに右側にあるblank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。

(5) 補足1

標準メッセージ上では、(1)から(4)までのデータエレメントの属性と桁数の表示を、表2-3のように行う。

(6) 補足2

数値データ内でのblankは原則として使用禁止であるが、受信側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2-3 C I Iシンタックスルールの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:最大長 (byte数)	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。
	16 bit文字列 (JIS X0208)	K属性	K (n) n:最大長 (byte数)	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 23456 (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。

2-1-3 TFD形式データ1 (データエレメント)

メッセージ内の可変長データエレメントはTFD(Transfer Form Data)形式で表現しなければならない。

データタグ (データ識別ID)、データエレメント長 (データ長) 及び可変長データエレメントでTFD形式データを構成する (図2-1参照)。

- ① データタグは、メッセージ内でのデータエレメントを識別する。通常、標準メッセージ及びデータエレメント・ディレクトリの整理番号が用いられる。
- ② データエレメント長は、直後のデータエレメントの長さをbyte数で表す。
- ③ データエレメントには、2-1-1で示した省略ルールが適用できる。

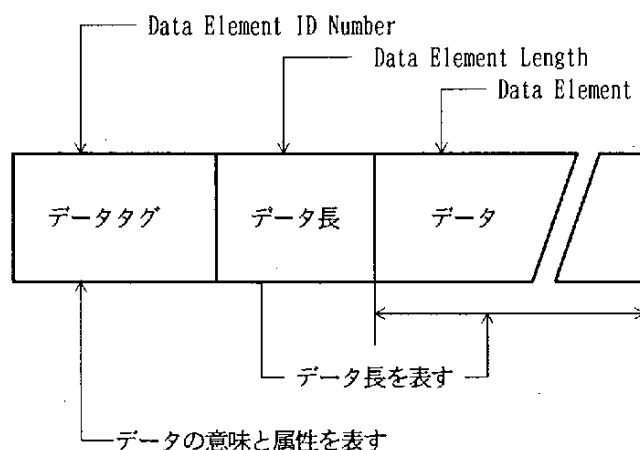


図2-1 TFD形式データ

(1) データタグ

データタグには、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。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2-2を参照）。縮小モードでは、データタグは1 byteの符号無バイナリーの数値となり、1～239の範囲の数値でデータエレメントの識別子を表す。拡張モードでは、データタグは2 byteの符号無バイナリーの数値となり、1～61439の範囲の数値でデータエレメントの識別子を表す。この数値（データタグの値）には、通常、標準メッセージやデータエレメント・ディレクトリーの整理番号（項目No.）を用いる。

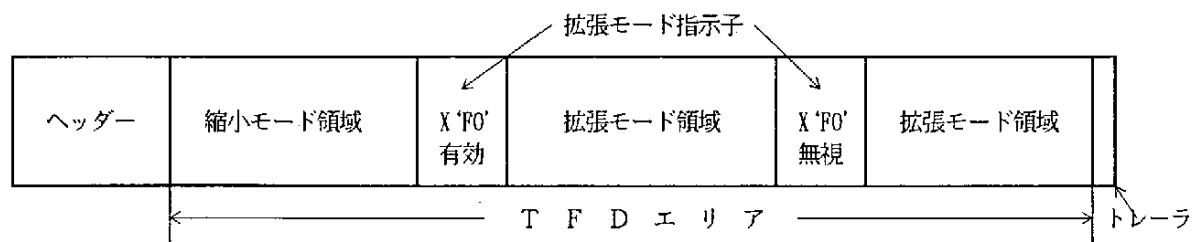


図2-2 メッセージ内のTFDエリアにおける拡張モード指示子の機能

(2) データ長（データエレメント長）

データ長は、直後のデータエレメントのデータ長をbyte数で表す。1～239 byteまでのデータエレメントの長さは、1 byteの符号無バイナリー数値で表す。240～32767 byteまでのデータエレメントの長さは、データ長拡張子と2 byteの符号無バイナリー数値を用いて表す（図2-3を参照）

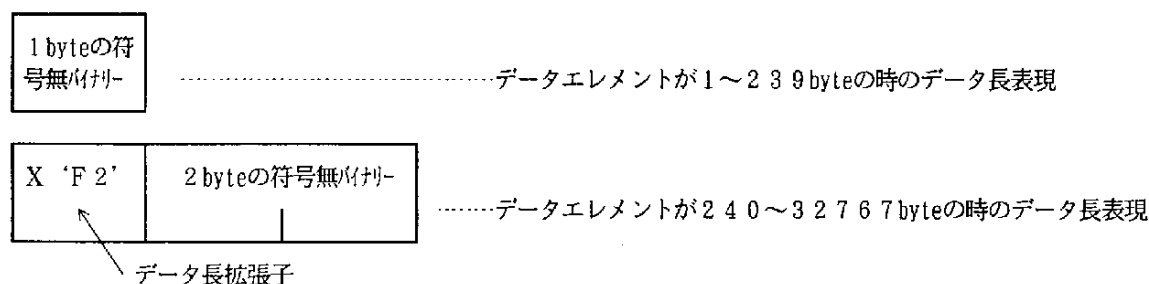


図2-3 データ長（データエレメント長）の形式

(3) TFD形式データ1のまとめ

TFD形式データ1は、データタグが縮小モードと拡張モードの2種類あり、データ長も通常形式と拡張形式の2種類があるので、両者を組み合わせることにより、合計4種類ある。

- ① 縮小モードデータタグ + 通常形式データ長..... (01型TFDである(図2-4の01型))
- ② 縮小モードデータタグ + 拡張形式データ長..... (02型TFDである(図2-4の02型))
- ③ 拡張モードデータタグ + 通常形式データ長..... (11型TFDである(図2-4の11型))
- ④ 拡張モードデータタグ + 拡張形式データ長..... (12型TFDである(図2-4の12型))

2-1-7 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。

2-1-8 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文字列である。

2-1-9 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、一つ又は複数のメッセージ及び一つの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・ヘッダーは、常に、メッセージグループの先頭に置き、次いでメッセージを置き、最後にメッセージグループ・トレーラを置かなければならない。

2-1-10 交換

一つ又は複数のメッセージグループで、一つの交換を構成する。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、通信システム上の一つの発信地から一つの着信地に送信される。

2-1-11 レコードとファイル

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呼び、個々のファイル内の論理的かつ機械的分割単位をレコードと呼ぶ。

2-1-12 格納構造

メッセージグループのファイル／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。

(3) メッセージグループ・ヘッダーとメッセージグループ・トレーラの格納構造

一つのメッセージグループ・ヘッダー及び一つのメッセージグループ・トレーラは、それぞれ一つの251 byteのレコードに格納しなければならない。

(4) メッセージの格納構造

一つのメッセージは、一つの可変長レコードに格納するか、一つ又は複数の251 BYTEの固定長レコードに格納しなければならない。前者を通常モード、後者を分割モードと呼ぶ。分割モードにおいて、メッセージの長さとレコードの長さが合わない時レコード上の余りの部分は、ブランクで満たさなければならない。

2-2 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、TYPE 12で用いられるヘッダーで、図2-6の251byte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、1-3項で示すキャラクターセットと文字コードで表現しなければならない。

一つのメッセージグループ・ヘッダーは、一つの251BYTEのレコードに格納しなければならない。

TYPE 12 (251 byte)

分割 区分	レ コ ド 区 分		発信センタ-コード		発信者 コード	受信センタ-コード		受信者 コード	BPID			リザ-ブ	情報 区分	第1トータ ル項目No.		フォーマ ット ID	リザ-ブ
			所属VA Nコード	発信セン タ-コード		所属VA Nコード	受信セン タ-コード		機関		版			1	2		
1	1	1	12	12	12	12	12	12	4	2	2	12	4	3	3	2	10

byte数

運用モード

サブ機関

(メッセージグループ・ヘッダーの続き)

作成日 付時刻	リザーブ	シンタックス IDバージョン	拡張 モード	分割 モード	文字 コード		非 透 過	第2トータル 項目No.		リザーブ
					8	16		1	2	
12	12	6	1	1	1	1	1	5	5	89

byte数

(E I A Jシンタックスルールでは、
定義されていない部分)

図2-6 メッセージグループ・ヘッダーのフォーマット

- ① 分割区分(0=X'30') (文字1 byte) TYPE 12のヘッダーかトレーラーであることを示す。
- ② レコード区分 (文字1 byte) TYPE 12のメッセージグループ・ヘッダーであることを示す。
(C=X'43')
- ③ 運用モード (文字1 byte) (=X'31')の時テスト・データを表す。(0=X'30', △=X'20')の時通常のデータを表す。
- ④ 発信センターコード
 所属VANコード (文字12byte) キャラクターセットと文字コードは、表2-2を参照。
 発信センターコード (文字12byte) " " " "。
- ⑤ 発信者コード (文字12byte) キャラクターセットと文字コードは、表2-2を参照。

⑥ 受信センターコード

所属VANコード (文字12byte)
受信センターコード (文字12byte)

キャラクターセットと文字コードは、表2-2を参照。
" "

⑦ 受信者コード (文字12byte)

キャラクターセットと文字コードは、表2-2を参照。

⑧ BPID (キャラクターセットと文字コードは、表2-2を参照)

機 関 (文字4 byte)
サブ機関 (文字2 byte)
版 (文字2 byte)

ユニークな、標準メッセージを管理する機関を表すコード
大分類管理コード (標準メッセージ管理機関が管理するコード)
バージョン番号 (標準メッセージ管理機関が管理するコード)

⑨ リザーブ (文字12byte)

将来の拡張エリア (all フラグ =X'20')

⑩ 情報区分コード (文字4 byte)

情報の種類を表す。キャラクターセットと文字コードは、表2-2を参照。

⑪ 第1トータル項目ナンバー (文字3 byte * 2)

縮小モードの時、ハッシュ・トータルに使う項目を指定する。項目No. (デフォルト値) が999以下の9属性の項目を指定する。2つの項目を指定できる。フラグを指定すると、ハッシュ・トータルを行なわない。キャラクターセットと文字コードは、表2-2を参照。

⑫ フォーマットID (文字2 byte)

E I A J シンタックスルールとの互換を保つため、10=X'3130' 固定とする。

⑬ リザーブ (文字10byte)

将来の拡張エリア (all フラグ =X'20')

⑭ 作成日付時刻 (文字12byte)

メッセージグループの作成日付時刻 YYMMDDHHMMSS。文字コードは、表2-2を参照。

⑮ リザーブ (文字12byte)

将来の拡張エリア (all フラグ =X'20')

⑯ シンタックスID/バージョン 番号 (文字6 byte)

シンタックスルール管理機関およびリリース・バージョン番号 (C I I 1 0 0) を表す。キャラクターセットと文字コードは、表2-2を参照。

⑰ 拡張モード (文字1 byte)

フラグ又はSの時、縮小モード (E I A J シンタックスルール互換モード) を表し、Eの時拡張モードを表す。文字コードは、表2-2を参照。

⑱ 分割モード (文字1 byte)

フラグ又はMの時、分割モードを表し、Sの時、通常モード (レコード分割無し) を表す。文字コードは、表2-2を参照。

⑲ 文字コード

メッセージをコーディングしている文字コードの種類を表す。文字コードは、表2-2を参照。

8 b i t 系 (文字1 byte)
16 b i t 系 (文字1 byte)

フラグ又はSの時、8 b i t 系文字がJIS-X0201 であり、Pの時、それ以外を表す。
フラグ又はSの時、16 b i t 系文字がJIS-X0208 であり、Pの時、それ以外を表す。

⑳ 非透過モード (文字1 byte)

フラグ又はSの時、透過モードを表し、Mの時、非透過モードを表す。文字コードは、表2-2を参照。非透過モードについては、2-6を参照。

㉑ 第2トータル項目ナンバー (文字5 byte * 2)

拡張モードの時、ハッシュ・トータルに使う項目を指定する。9属性の項目を指定する。2つの項目を指定できる。フラグを指定すると、ハッシュ・トータルを行なわない。文字コードは、表2-2を参照。尚、縮小モードの時は、このエリアをブランクにしなければならない。拡張モードの時は、㉑の第1トータル項目ナンバーは指定しても、無視する。

㉒ リザーブ (文字89byte)

将来の拡張エリア (all フラグ =X'20')

2-3 メッセージ（可変長）の構造（TYPE 12）

一つのメッセージは、メッセージ・ヘッダー、TFDエリア、メッセージ・トレーラで構成され、一つのトランザクション、例えば1通の発注書に相当する。又、通常モードでは、一つのメッセージは一つの変長レコードに格納し、分割モードでは、250 byte（分割識別子を除く）ずつ単純に分解して251byteの固定長レコード（250byte以下の時は1レコード）に格納する。レコードへの格納方法を除けば、通常モードと分割モードは同一の構造である。以下では、通常モードのメッセージの構造を説明する。

2-3-1 メッセージの全体構造

図2-7の構造である。TFDエリア（Transfer Form Data Area）は、可変長であり、この中にTFD形式のデータが収容される。TFD形式データの構造は、2-1-3及び2-1-4を参照。

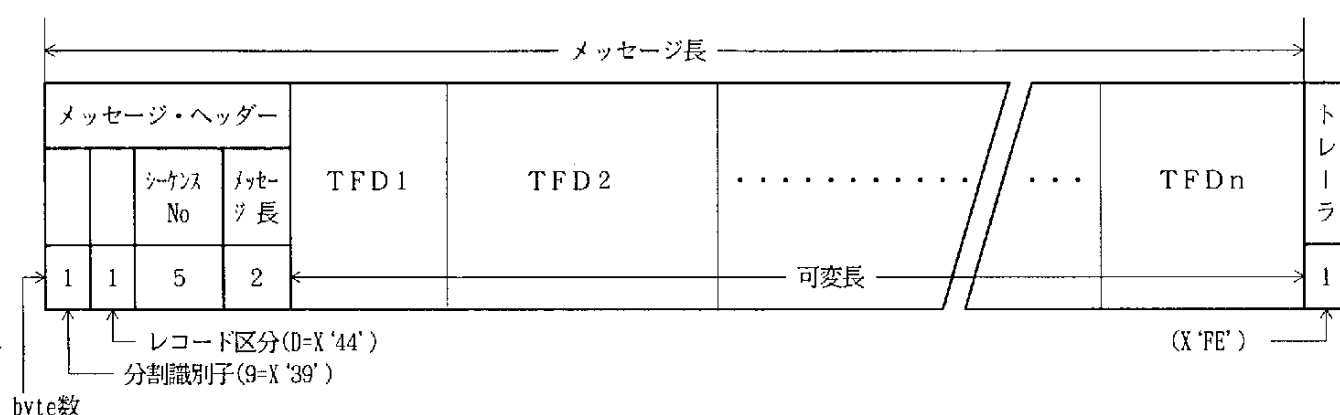


図2-7 メッセージの全体構造

2-3-2 メッセージ・ヘッダーの詳細

- | | |
|----------------------------|--|
| ① 分割識別子 (1=X'39') | メッセージを収容したレコードの分割識別子 |
| ② レコード区分 (0=X'44') | メッセージを収容したレコードであることを表す。 |
| ③ シーケンスNo. (数値5桁アンパック 型式) | メッセージを収容したレコードの昇順の5桁のシーケンス番号である。メッセージグループ・ヘッダー・レコード直後のレコードの値は1で、以後1ずつ増加する。文字コードは、表2-1を参照。 |
| ④ メッセージ長 (数値2byteパッキング 型式) | メッセージを収容したレコードの先頭から、トレーラの手前までの全長を表す。トレーラはこの長さに含まれないことに注意する。
数値の最大値は、32767 であるが、これより長いメッセージの時は、このエリアの値を 32896(=X'8080')とし、『2. 3. 5』で示すメッセージ長指示子によって、メッセージ長を表す。メッセージ長指示子を用いる場合は、メッセージ長指示子は図2-7のTFD 1の位置におかなければならない。 |

2-3-3 TFDエリア（Transfer Form Data Area）の詳細

TFDエリアは、TFD形式データ1（2-1-3参照）及びTFD形式データ2（2-1-4参照）で構成する。

TFDエリアは縮小モードで始まり、拡張モード指示子が出現した後、拡張モードに切り換わる。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り、さらに、メッセージグループ内のすべてのメッセージのTFDエリア全体が縮小モードの時メッセージグループ・ヘッダーの『拡張モード』が『S』になる。

TFD形式データは、常にタグ（データタグ又は制御タグ）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、データタグは2 byteのバイナリー数値である。また制御タグは常に1 byteのバイナリーの数値である。データタグか制御タグかは、タグの値によって決定される。また、制御タグは、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。すなわち、データタグや制御タグは、C I Iシンタックスルールの最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。TFD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。エリアの左端には、必ず、データタグか制御タグが存在する。

2-3-4 TFDの省略

一つのメッセージに含めなければならないTFD（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TFD（データ項目）が、標準メッセージによって規定されるTFD（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TFD（データ項目）を省略することができる。TFD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TFDや内容がallブランクの文字列TFDは、一般的に省略可能である。

TFD形式データ2（TFD制御子）については、一般に省略できない。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（2-3-7参照）

2-3-5 TFD制御子の機能

TFD形式データ2は、TFD制御子と呼ばれ、TFDエリアのコントロールを行う。TFD制御子は、1 byteのバイナリーの数値であるが、ある種のTFD制御子は制御データを保持する。制御データは固定長である（図2-8を参照）

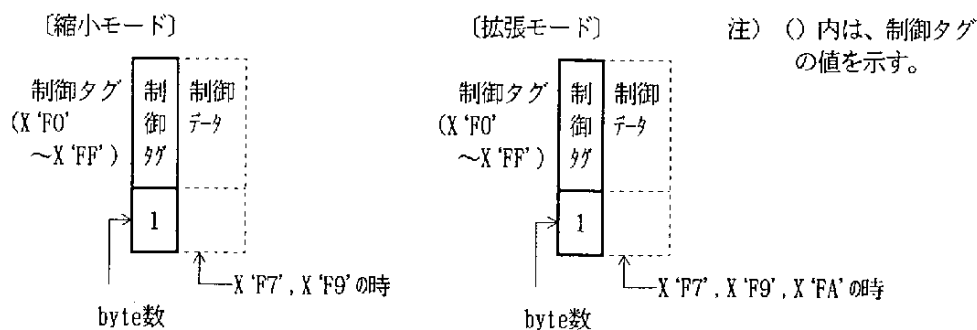


図2-8 TFD制御子の構造

(1) 拡張モード指示子(240=X'F0')

TFDエリアに、この制御タグが出現すると、それより右側のTFDエリアは、拡張モードになり、拡張モードのTFDが存在すると見なす。拡張モード指示子が同一のTFD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2-9を参照)。

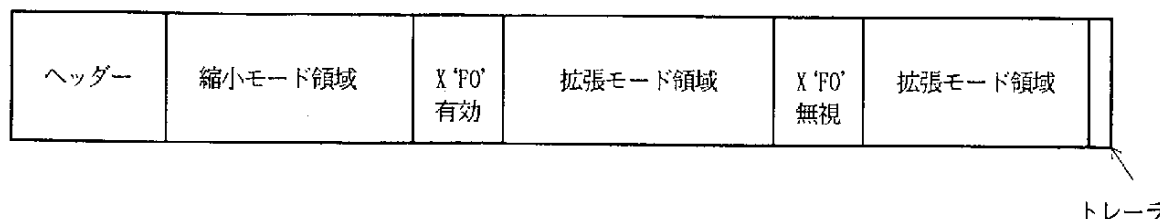


図2-9 TFDエリアにおける拡張モード指示子の機能

(2) データ長拡張子(242=X'F2')

データ長が240byte以上の時、これを付加することで、データ長が2byteで表現されていることを示す。この拡張子は、常に、データタグの直後でデータ長エリアの直前になければならない。

データ長の直前にデータ長拡張子(X'F2')が付加することで、データ長の最大長が、32767byteまで拡張される。通常のトランスレータは、データ長が240byte以上になると自動的にデータ長拡張子(X'F2')を付加し、データ長を2byteで表現する。

尚、単一項目のデータ長が240byte以上の時は、縮小モードでも、EIAJシンタックスルールとは互換性がない。

(3) メッセージ長指示子(247=X'F7')

メッセージ長指示子はメッセージ長を表す制御タグであり、制御データとして7byteのメッセージ長を保持する。この制御タグは、TFDエリアの先頭(左端:図2-7のTFD1の位置)にある時のみ有効で、その他の場所に存在する時はエラーである。7byteのメッセージ長は、メッセージ・トレーラ(データセグメント・セパレーター(TYPE-Eの時))を除く、メッセージ長を表す。メッセージ長は、十進キャラクター表現である。

メッセージ長指示子を用いる時は、メッセージ・ヘッダーのメッセージ長エリアは、X'8080'にする。

メッセージ長指示子は、以下の時に用いる

① 透過モードの時(TYPE12)

メッセージ長が32768(byte)~999999(byte)の時、図2-10に示すように用いる(メッセージ長が9(byte)~32767(byte)の時は、メッセージ・ヘッダーのメッセージ長で表す)。

② 非透過モードの時 (TYPE 1 2)

メッセージ・ヘッダーのメッセージ長エリアを、X'8080' にし、図2-10に示すように用いる。メッセージ長は9 (byte) ~ 9999999 (byte) が許される。

メッセージ長指示子を含むTYPE 1 2メッセージは、縮小モードであっても、EIA Jトランスレーターでは解読不能になる。

(4) バイナリーデータ補正子(248=X'F8')

TYPE 1 2の非透過モードの時、この補正子の直後の1 byteのバイナリーデータが、補正されていることを示す。補正の方法は、2-6を参照。バイナリーデータ補正子は、以下のケースの時のみ有効である。

- ① 縮小モードで、データタグ (1 byte) の直前にある時
- ② 拡張モードで、データタグ (2 byte) の直前、データタグ第2 byteの直前にある時
- ③ 縮小/拡張モードで、レングスタグ (1 byte) の直前、データ長拡張子付レングスタグ (2 byte) の直前と第2 byteの直前にある時

(5) インターナルセグメント指示子(249=X'F9')

メッセージ本体内のTFDエリア内の任意のTFDの集合を『インターナルセグメント』と定義する。インターナルセグメント指示子は、インターナルセグメントの区切りを示す。

- ① インターナルセグメントは、インターナルセグメント指示子 (X'F9'), マルチ明細ヘッダー (X'FA'), 改行マーク (X'FB')およびマルチ明細トレーラ (X'FC')で区切られるTFDの集合である。
- ② マルチ明細におけるセグメント

マルチ明細のセグメントは、インターナルセグメントであり、一つの明細内にインターナルセグメント・セパレーターを設けることで、二つのインターナルセグメントに分割することも可能である。

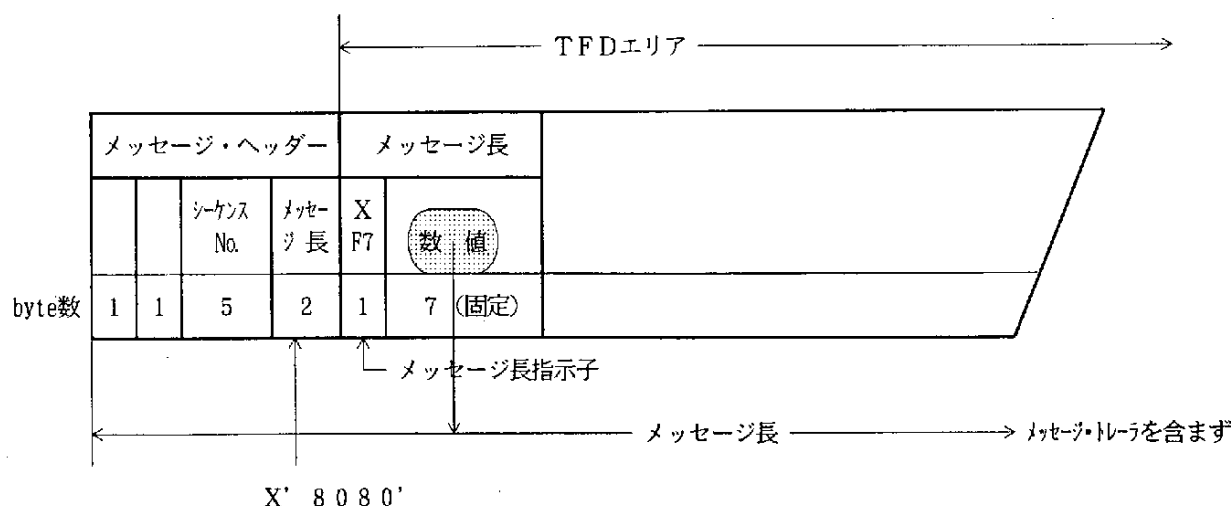


図2-10 TYPE 1 2におけるメッセージ長指示子の使い方

この制御タグは、1 byteの明細番号を持つ。明細番号は、1～9とA～Zまでの35種類である。すなわち、一つのメッセージ内に最大35種類のインターナルセグメントを設定できる。インターナルセグメントは、オプションであり、トランスレーターの処理になんらの影響も与えない。送信用トランスレーターではオプション指定によって、インターナルセグメント指示子を挿入する。一方、受信用トランスレーターでは、インターナルセグメント指示子を、単純に読み飛ばす（無視する）。

標準メッセージ上でインターナルセグメントを設定し、このセグメントとUN/FDIFACTにおけるデータセグメントの構造を一致させることにより、TYPE-EとUN/EDIFACTとの間の相互変換が可能になる。

尚、インターナルセグメントの明細番号は、一つのメッセージ内でユニークでなければならない。

インターナルセグメント指示子を含むTYPE 12メッセージは、縮小モードであってもEIAJトランスレーターでは解読不能になる。

(6) マルチ明細ヘッダー(250=X'FA')

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（2-3-7を参照）。

拡張モードの時は、このTFD制御子は1 byteの制御データを保持し（図2-8）、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別する。明細番号は、1=X'31', 2=X'32', ..., 9=X'39', A=X'41', B=X'42', ..., Z=X'5A'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く、明細番号は常に『0』として扱う。例えば、縮小モードの時にマルチ明細が始まり、その明細が終わる前に拡張モードへ移行した時は、ゼロ番のマルチ明細として認識する。

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

(7) マルチ明細改行マーク(251=X'FB')

マルチ明細の1行の終わりを示す（2-3-7を参照）。

(8) マルチ明細トレーラ(252=X'FC')

マルチ明細の終わりを示す（2-3-7を参照）。

(9) トレーラ（メッセージ・トレーラ）(254=X'FE')

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

(10) そ の 他

TFD形式データ1のデータエレメントのデータ長は、縮小形式の時、最小1 byte、最大239 byteであり、拡張形式の時、最小240 byte、最大32767 byteである。いずれの場合も、0 byteは許されない。

表 2 - 4 T F D制御子の制御タグの値と機能

制 御 タ グ の 値						デ 自 身 タ グ の 長 さ	項 目 No. の 割 当 (意 味 及 び 機 能)
拡張モード			縮小モード				
Hex	Dec	長さ	Hex	Dec	長さ		
F0	240	1	F0	240	1	0	拡張モード指示子。このデータ・タグが出現するまでは、縮小モードであり、このデータ・タグが出現した後は、拡張モードになる。拡張モードになった後、このデータ・タグが出現しても、無視される。
F1	241	1	無効	無効	無効	0	使用禁止（リザーブ），（制御タグ、未定義）
F2	242	1	F2	242	1	0	データ長拡張子、データ長17の直前になければならない。データ長が2 byteになる。
F3 }	243 }	1	無効	無効	無効	0	使用禁止（リザーブ），（制御タグ、未定義）
F6	246	1					
F7	247	1	F7	247	1	0	メッセージ長指示子、メッセージヘッダのメッセージ長でメッセージの長さが表せない時、メッセージ長の値を'8080'とし、この制御タグに続く7文字の数値でメッセージ長を表す。この制御タグは、TFDエリアの先頭になければならない。
F8	248	1	F8	248	1	0	非透過モードの時のパリティデータ補正指示子、直後の1 byteのパリティデータの補正を表す。
F9	249	1	F9	249	1	0	インターレグメント指示子を表す。第2 byteがインターレグメントの番号を表し、1～Zの35個のインターレグメントが使える。
FA	250	1	FA	250	1	0	マルチ明細のヘッダーを表す。第2 byteがマルチ明細の番号を表し、1～Zの35個の明示的複数マルチ明細が使える。 縮小モードでは、'FA30'は'FA'で表せられ'FA32'～'FA5A'は使用できない。
FB	251	1	FB	251	1	0	マルチ明細改行マーク（マルチ明細の改行を表す。）
FC	252	1	FC	252	1	0	マルチ明細トレーラー（マルチ明細の終わりを表す。）
FD	253	1	無効	無効	無効	0	使用禁止（リザーブ），（制御タグ、未定義）
FE	254	1	FE	254	1	0	メッセージのトレーラー、メッセージの終わりを表す。
FF	255	1	無効	無効	無効	0	使用禁止（リザーブ），（制御タグ、未定義）

2-3-6 データタグと項目No.

標準メッセージやデータエレメント・ディレクトリーでは、設計やメンテナンスの合理化のために、データエレメントに整理番号を付け、項目Noとする。C I I シンタックスルールでは、この項目NoとT F D形式データ1のデータタグの値を一致させることにより、トランスレータの変換テーブルの運用を効率化させる。従って、新しい項目を標準メッセージやデータエレメント・ディレクトリーに追加する時は、ある規則に従って、その追加項目の項目Noを定める。この規則を、項目No割当計画と呼ぶ。表2-5は、産業情報化推進センターが推奨する割当計画である。

表2-5 項目No割当計画

データタグの値						デ 自 身 タ グ の 長 さ	項 目 No. の 割 当 (意 味 及 び 機 能)
拡張モード			縮小モード				
Hex	Dec	長さ	Hex	Dec	長さ		
0000	0	2	00	000	1	1	使 用 禁 止
0001 } 00EF	1 } 239	2 	01 } EF	001 } 239	1 	1 又は 2	一般項目（8 bit文字, 16 bit文字）の項目Noを表す。（9, X属性のデータのみ使用し、データ長エリアが1 byte（項目データ長が 239byte以下）の時、E I A J 標準との互換性領域）
00F0 } EA5F	240 } 59999	2 	無効	無効	無効	1 又は 2	一般項目（8 bit文字, 16 bit文字）の項目Noを表す。
EE48 } EFFF	61000 } 61439	2 	無効	無効	無効	1 又は 2	使用禁止（リザーブ, 未定義）

2-3-7 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やすく、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2-11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ーをセットする。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2-12）。

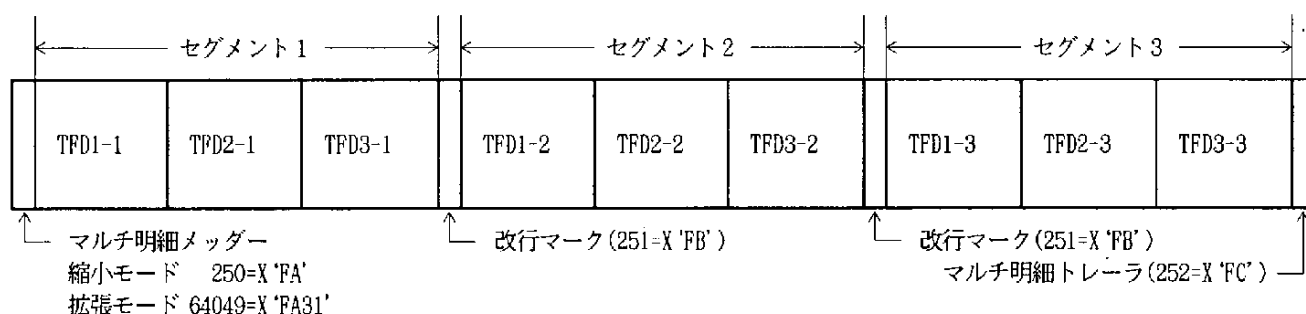


図2-11 マルチ明細

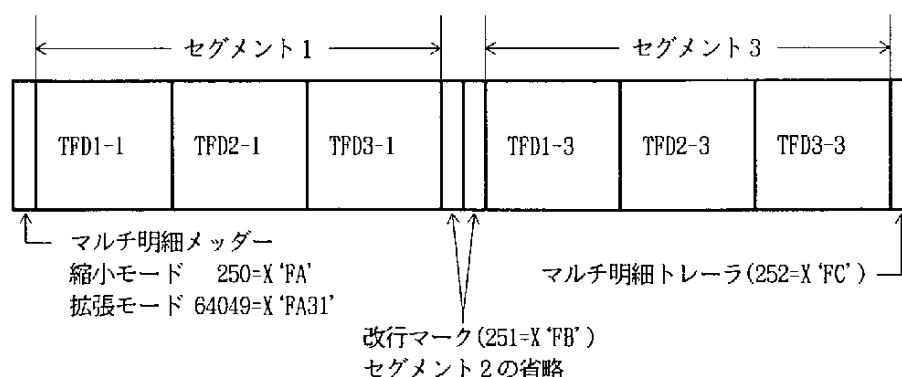


図2-12 マルチ明細（図2-11 のTFD1-2, TFD2-2, TFD3-2が省略された時）

(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2-13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ーは残すことが望ましい。

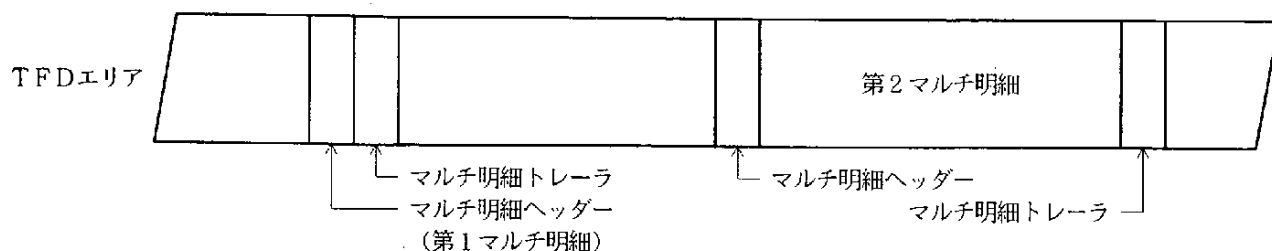


図2-13 縮小モードにおけるマルチ明細の複数化

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が可能である（図2-14参照）

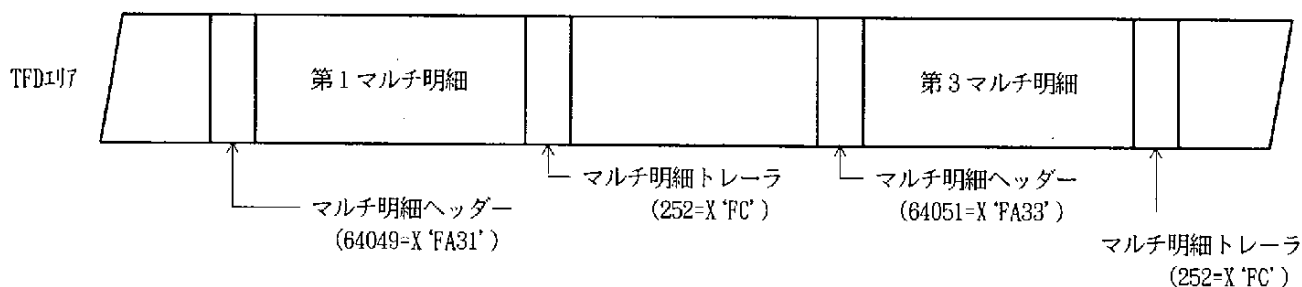


図2-14 拡張モードにおけるマルチ明細の複数化（第2マルチ明細の省略）

(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2-15に示すように、3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ーは省略しない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ーを省略しない時は、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は、明細番号無のマルチ明細によるネスト化を検出した場合、自動的にエラー処理を行う。

2-3-8 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はそのマルチ明細は、『0』番の明細番号付マルチ明細と見なす。

(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が出現した時（トランスレータの変換テーブルでこのような指定のケースがある）は、強制的に『0』番の明細番号を付加する。

従って、拡張モードの『0』番マルチ明細と縮小モードにおけるマルチ明細は、同一明細として取り扱う。

2-3-9 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。

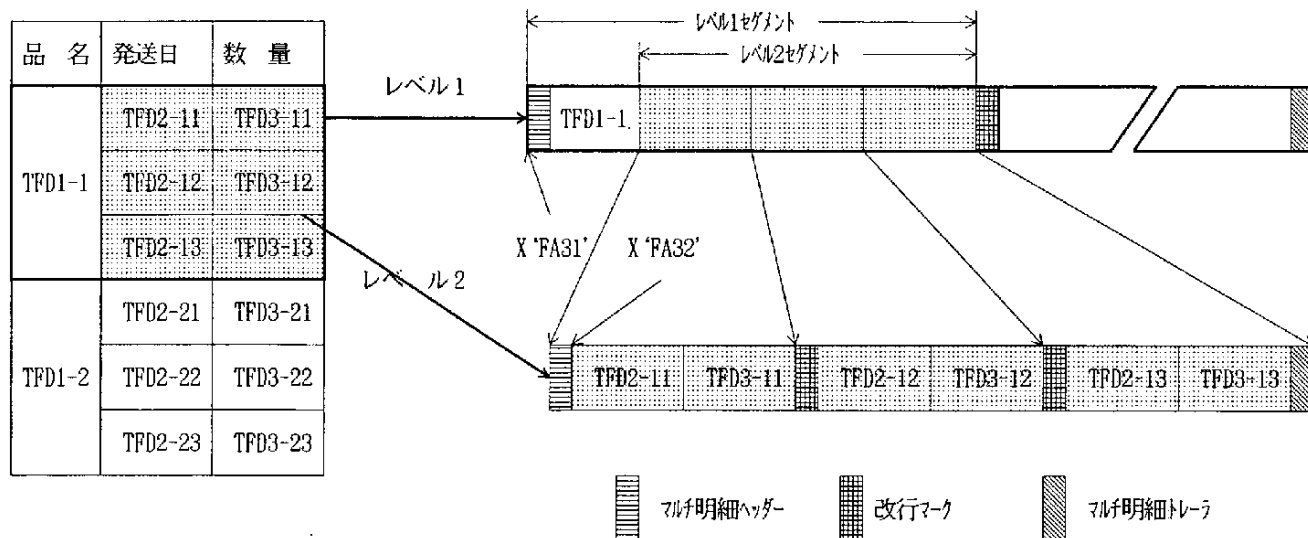
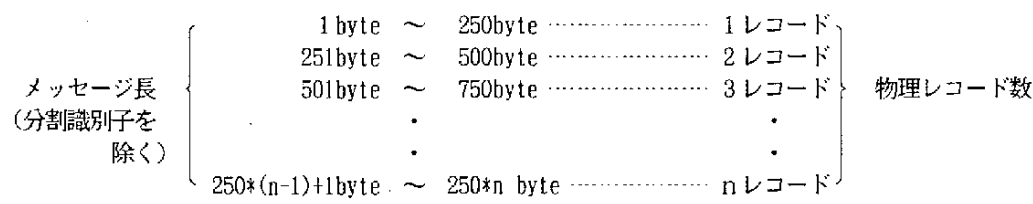


図2-15 マルチ明細のネスティング

2-4 メッセージの格納構造 (TYPE 1 2 分割モード)

TYPE 1 2 の通常モードでは、一つの可変長のメッセージを一つの可変長レコードに格納する。一方、分割モードでは一つの可変長メッセージを 250 byte ずつに分割して、1 個あるいは n 個の 251 byte の固定長レコードに収容する。すなわち、分割モードの時はメッセージの内容自体は同一で、論理レコードへの格納方法が異なる。

レコードの分割は、以下のように行われる。



分割されたメッセージ(250 byte) の左先頭に、分割識別子 (1 byte) を追加した 251 byte のデータが分割化レコードとなる。分割識別子は、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。文字コードは、原則として JIS-X0201 である。

1=X'31' → 2=X'32' → → 8=X'38' → 1=X'31' → 2=X'32' → → 5=X'35' → 9=X'39' (最終レコード)

つまり、1, 2, 3, の順で 8 まで行き、ふたたび 1 から 8 まで番号を付ける。1 ~ 8 までの繰り返しである。そして、最終レコードは、常に 9 を付ける。図 2-16 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、メッセージ長が 250 byte 以下の時 (分割識別子を除く) は、1 レコードに格納され、分割識別子は、9=X'39' である。

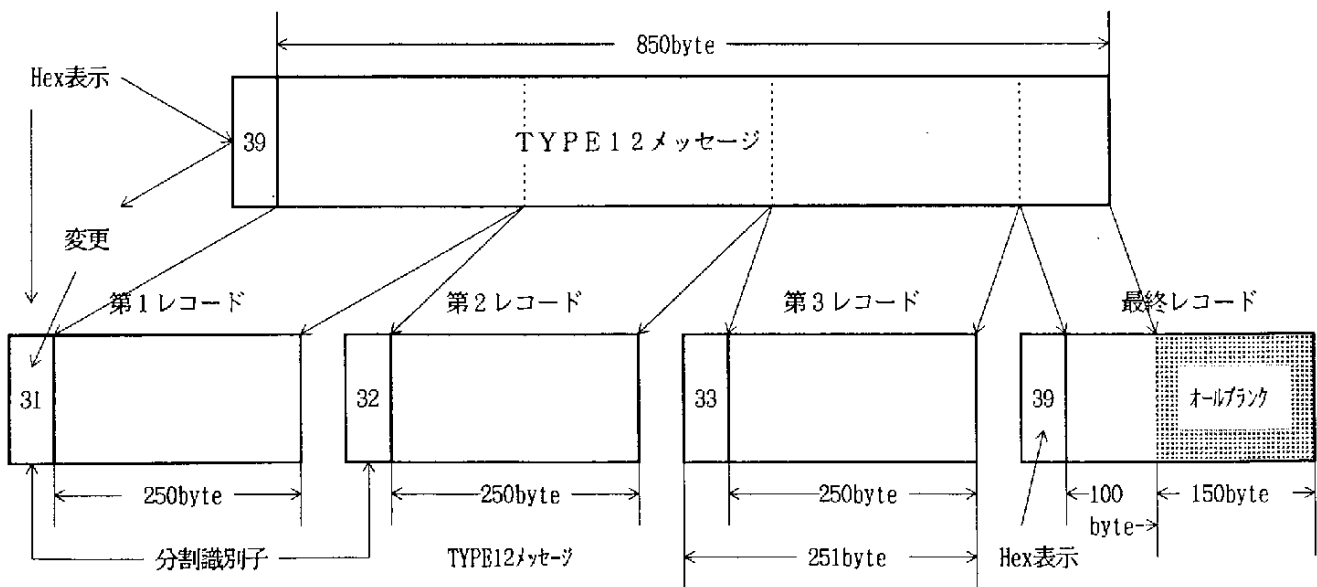


図 2-16 TYPE 1 2 分割化固定長レコード (4 分割の例)

2-5 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、TYPE 12 で用いられるトレーラで、図2-17で示す251byteの固定フォーマットである。一つのメッセージグループ・トレーラは、一つの251byte長のレコードに格納される。

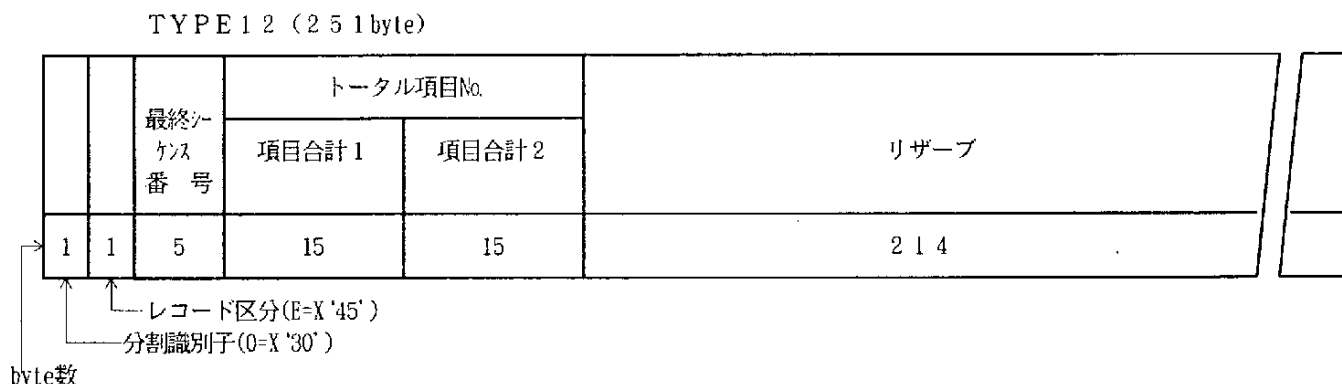


図2-17 メッセージグループ・トレーラ (TYPE 12)

2-5-1 メッセージグループ・トレーラ・レコードの詳細

- ① 分割識別子(0=X'30') ヘッダーもしくはトレーラ・レコードであることを示す。
- ② レコード区分(E=X'45') メッセージグループ・トレーラであることを示す。
- ③ 最終シーケンスNo. (数値15桁7桁型 型式) 直前のメッセージのシーケンスNo.である。すなわち、メッセージの数を表す。
メッセージが0件の時は、0になる。文字コードはJIS-X0201 を用いる。
- ④ トータル項目No.

項目合計1 (数値15桁7桁型 型式)	チェック用トータル数値1、文字コードは、JIS-X0201 を用いる。
項目合計2 (数値15桁7桁型 型式)	チェック用トータル数値2、文字コードは、JIS-X0201 を用いる。
- ⑤ リザーブ (文字 214byte) 将来の拡張エリア(all桁=X'20')

2-5-2 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、チェック値を計算する。

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 (F又はS) が設定されている場合には、第1トータル項目ナンバー・エリアで指示された項目No. (3桁) の値でチェック値を計算し、拡張モード (E) が設定されている場合には、第2トータル項目ナンバー・エリアで指示された項目No. (5桁) の値でチェック値を計算する。

尚、blank (=X'20') 又は0 (=X'30') の項目No.が指定されて場合には、項目合計については、チェック用トータル数値を算出せず 0=X'30' とする。

2-6 非透過モード時の文字コードの変換およびバイナリーデータの補正

メッセージグループ・ヘッダーの非透過モード指示子で、非透過モード(=M)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ー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以外の文字をデータに用いることは許されない。尚、詳細は、文字コード表を参照すること。

2-6-1 文字(JIS-X0201とJIS-X0208)の変換

X'20' ~X'3F' を、X'80' ~X'9F' に変換し、伝送する。この変換は、トランスレータで行われる。尚、メッセージグループ・ヘッダーとメッセージグループ・トレーラの分割識別子については、この変換から除外される(表2-6参照)。

2-6-2 バイナリーデータの補正

(1) メッセージ・ヘッダーのメッセージ長

メッセージ・ヘッダーのメッセージ長は、X'8080' に置き換え、メッセージ長指示子によって、メッセージ長を表す。

(2) データタグおよびレングスタグの補正

表2-7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・バイナリーデータ補正指示子(X'F8')を追加する。表2-7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'

表2-6 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
20 → 80	30 → 90	40	変換無	↓	変換無
21 → 81	31 → 91				
22 → 82	32 → 92				
23 → 83	33 → 93				
24 → 84	34 → 94				
25 → 85	35 → 95				
26 → 86	36 → 96				
27 → 87	37 → 97	7F	変換無	↓	変換無
28 → 88	38 → 98	A0			
29 → 89	39 → 99				
2A → 8A	3A → 9A				
2B → 8B	3B → 9B				
2C → 8C	3C → 9C				
2D → 8D	3D → 9D				
2E → 8E	3E → 9E				
2F → 8F	3F → 9F	DF			

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

表2-7 非透過モード時のバイナリーデータの補正

数値	補正値	数値	補正値	数値	補正値	数値	補正値	数値	補正値
00 → 80	10 → 90	(20 → A0)	(30 → B0)	40	補正無	↓	補正無	F7	F8 → C8
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		補正無	↓	補正無	FE	FF → CF
(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ーデータに適用
表中の値は、すべてHex表示

3. C I I シンタックスルールの詳細 (TYPE-E)

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。

以下の記述では、8bit を1byteと記述し、X'1A'は、Hex表示を併用する。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表すものとする。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。

3-1 交換の階層

TYPE-Eの交換の階層は、ISO9735と同一である。詳細は、ISO9735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。

3-2 UNAセグメントとUNBセグメント

UNA、UNBセグメントの構造は、ISO9735と同一で、TYPE 12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'2F' |
| 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TYPE 12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=X'43494942'
0002	シタックスバージョン番号	(数値 1byte)	1=X'31'
S002	交換送信者		
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)所属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)	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。
0014	経路アドレス	(文字14byte-Max)	(発信者)所属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	処理優先順位コード	(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'、しない時は、省略か 0=X'30'
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)

3-3 メッセージの構造 (TYPE-E)

TYPE 1 2 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 C I I 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 2-18 参照)。

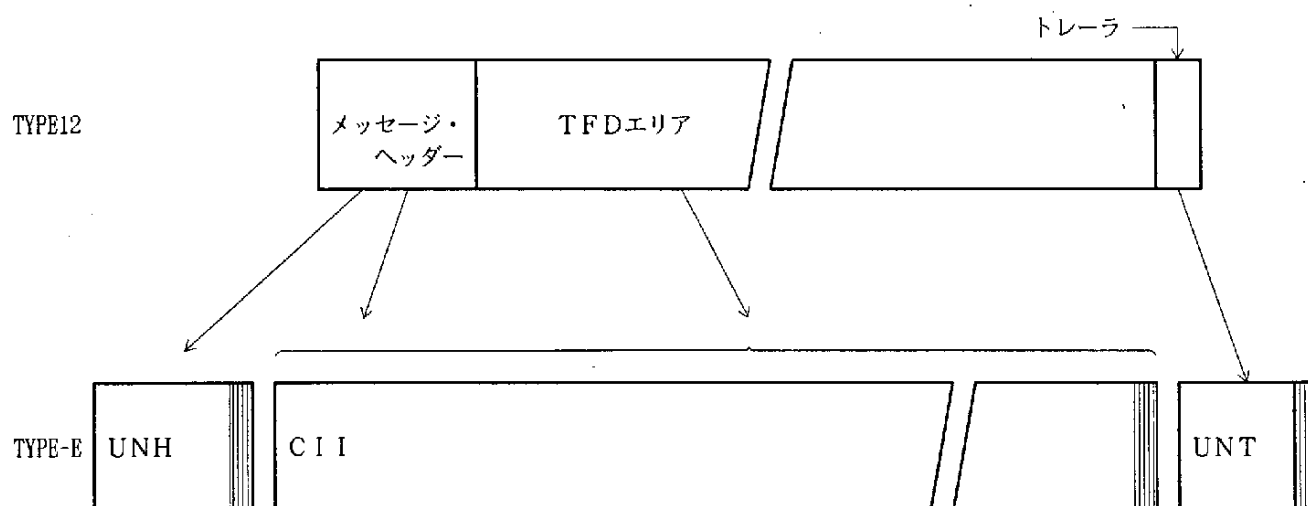


図 2-18 TYPE 1 2 と TYPE-E との対応

3-3-1 UNHセグメント

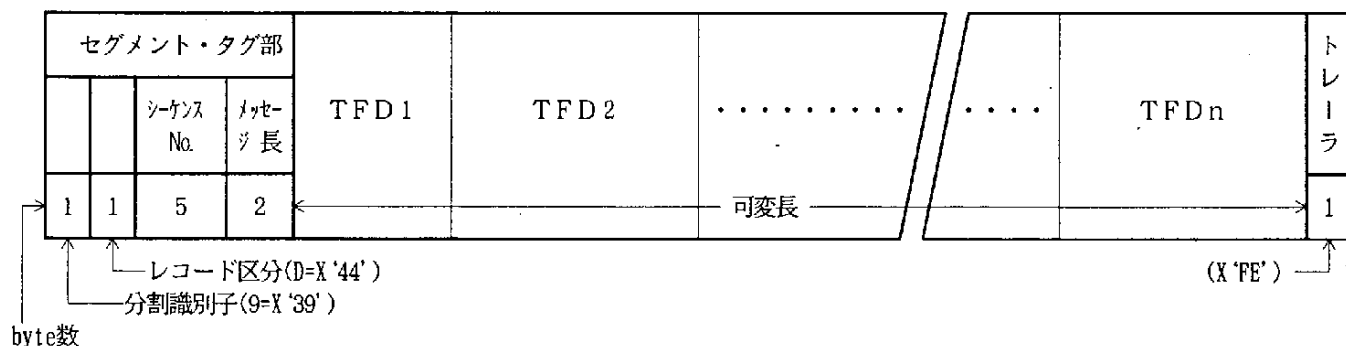
UNHセグメントは、ISO 9735と同一である。以下に、TYPE 12のメッセージ・ヘッダーとの対応を示す。文字コードは、JIS-X0201 を使用する。

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。

3-3-2 CIIセグメント

TYPE 12のメッセージのメッセージ・ヘッダーが変更されたメッセージが、そのままCIIセグメントとして用いられる。

TYPE 12のメッセージ



TYPE-EのCIIセグメント

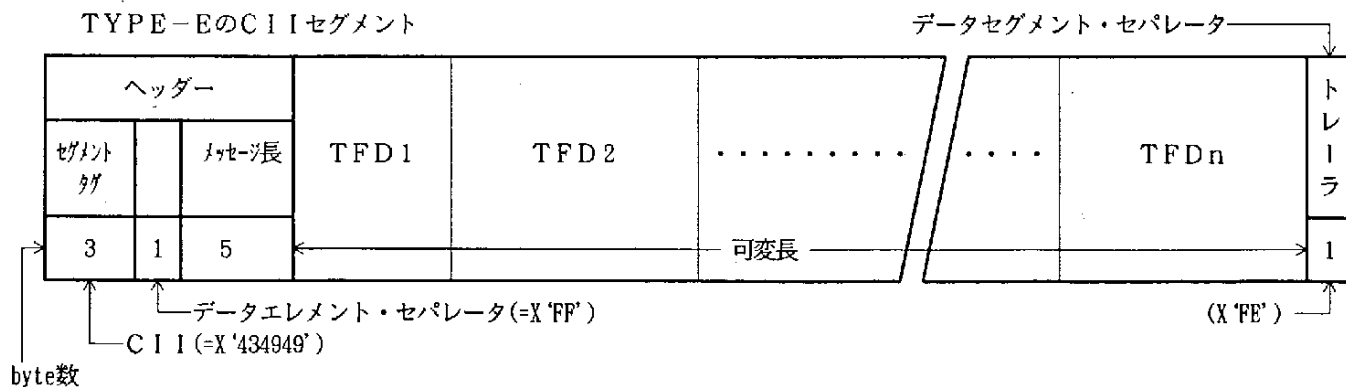


図2-19 TYPE-EのCIIセグメント

- 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、メッセージ長指示子により、図 2-19 の T F D
 1 の位置で、アンパック形式の数字 7 桁で表す。

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'PC') 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'

3-3-3 UNTセグメント

UNTセグメントは、ISO 9 7 3 5 と同一である。TYPE 1 2 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))

3-4 UNZセグメント

UNZセグメントは、ISO 9 7 3 5 と同一である。TYPE 1 2 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	(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。

3-5 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変長レコードに格納する。

4. システム運用電文フォーマット

システム運用電文は、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メッセージであり、受信確認メッセージとゼロ件メッセージの2種類がある。

受信確認メッセージとは、業務メッセージの受信済ステータスを、業務メッセージの受信者から送信者へ知らせるメッセージであり、このメッセージの発信は、業務上の約束の成立を意味しない。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。受信確認メッセージは、システム上の電文受信が成功したことを、送信者に伝達するものである。

ゼロ件メッセージは、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ために用いるメッセージである。

4-1 受信確認メッセージの構造（TYPE 12）

TYPE 12の受信確認メッセージの構造を、図2-20に示す。

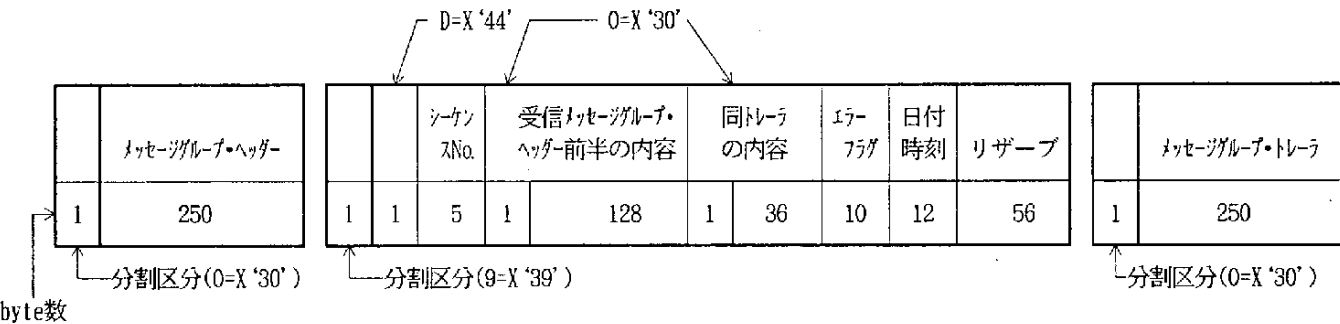


図2-20 TYPE 12の受信確認メッセージ

以下に、TYPE 12の受信確認メッセージの詳細を示す。文字コードは、JIS-X0201 を使用する。

- ① メッセージグループヘッダ (文字251byte) 図2-6と同一の固定フォーマット
- i) 情報区分コード (文字4byte) 9 0 0 1 = (X'99303031')
 - ii) トータル項目ナンバ (文字3byte*2) all フラグ =X'20' (ハッシュ・トータル無)
 - iii) フォーマットID (文字2byte) 縮小モード =(10=X'3130')
 - iv) その他の項目 図-5と同一内容
- ② 受信確認メッセージ (文字251byte)
- i) レコード区分 (文字1byte) データレコード (受信確認メッセージ) であることを示す (D=x'44')。
 - ii) シーケンスNo (数値5byte) 通常、0 0 0 0 1=x'3030303031'
 - iii) 受信メッセージグループヘッダの内容 (文字129byte) 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～作成日付時刻までの 129byteのコピー
 - iv) 同トレラの内容 (文字37byte) 受信に成功したメッセージグループトレーラの分割識別子～トータル項目No (項目合計2) までの 37byte のコピー
 - v) エラーフラグ (文字 2byte*5) 受信データ処理時にエラーが発生した時、エラーコードをセットする。all フラグ =X'20' の時、エラー無とする。
 - vi) (作成) 日付時刻 (文字12byte) 当該受信確認メッセージの作成時刻 (YYMMDDHHMMSSタイプ)
 - ii) リザーブ (文字56byte) 将来の拡張エリア (all フラグ =X'20')

③ メッセージグループ・トレーラ (文字251byte) 図2-16と同一の固定フォーマット

- i) 最終シーケンス番号 (数値5 byte) 通常、0 0 0 0 1 =x '3030303031'
- ii) トータル項目No. (数値15byte*2) all フラグ =X '20' (ハッシュ・トータル無)
- iii) その他の項目 図2-16と同一内容

尚、受信確認メッセージは、一組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、複数個入れることも可能である。

4-2 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2-21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。

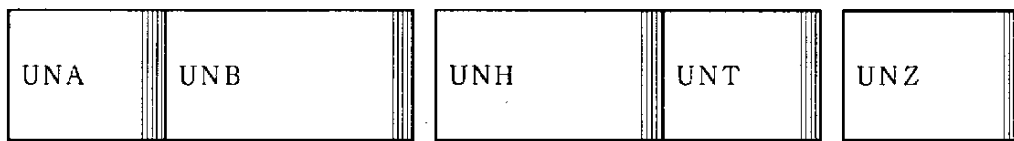


図2-21 TYPE-E受信確認メッセージの構造

(1) UNAセグメント 3-2を参照

(2) UNBセグメント

- ① 0026 アプリケーション参照番号 B P I D (8桁) + 情報区分 (9 0 0 1 = (X '39303031'))
- ② 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。
- ③ その他の項目は、3-2を参照

(3) UNHセグメント

- ① 0065 メッセージタイプ 情報区分
- ② 0068 共通7桁参照番号 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。
- ③ その他の項目は、3-2を参照

(4) UNTセグメント 3-3-1を参照

(5) UNZセグメント 3-4を参照

4-3 ゼロ件メッセージの構造

ゼロ件メッセージは、図2-22に示すように、TYPE 12については、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・トレーラの対であり、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。

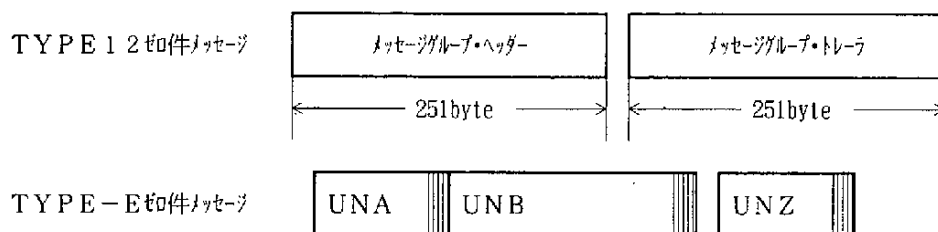


図2-22 ゼロ件メッセージの構造

TYPE 12のゼロ件メッセージでは、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し、さらに、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。

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・カウントは、ゼロ（メッセージ無）とする。

At the meeting of the Board of Directors of the
 University of California, held at Berkeley, California,
 on December 13, 1911, the following resolution was
 adopted:

Resolved, That the sum of \$10,000 be appropriated
 for the purchase of books for the University of California
 Library, to be paid out of the fund for the purchase of
 books for the University of California Library.

At the meeting of the Board of Directors of the
 University of California, held at Berkeley, California,
 on December 13, 1911, the following resolution was
 adopted:

RESOLUTIONS OF THE BOARD OF DIRECTORS

At the meeting of the Board of Directors of the
 University of California, held at Berkeley, California,
 on December 13, 1911, the following resolution was
 adopted: (79)
 Resolved, That the sum of \$10,000 be appropriated
 for the purchase of books for the University of California
 Library, to be paid out of the fund for the purchase of
 books for the University of California Library.

Ⅲ トランスレーターの概要

トランスレーターの仕様は暫定仕様である

1. 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・インターフェース) 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。この種のトランスレーターは、実用的ではないが、トランスレーター本来の機能はすべて備えており、ユーザー環境に応じた改造(A P I の変更)を行うことで、実用的なトランスレーターになる。従って、標準トランスレーターとして機能評価を行う上で、価値あるものである。これが、モデル・トランスレーターの実態である。

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、同一の機能を保持する。しかしながら、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 Smart TRanSlator for Encoding standard format (STRSEMA)
- b. データ受信用 Smart TRanSlator for Decoding standard format (STRSDMA)

② パソコン用モデル・トランスレーター

- a. データ送信用 Smart TRanSlator for Encoding standard format (STRSEPA)
- b. データ受信用 Smart TRanSlator for Decoding standard format (STRSDPA)

である。

2. モデル・トランスレーターの基本構成

以下の説明は、すべて、メンフレーム用モデル・トランスレーターについて、行う。

2-1 送信用トランスレーター (STRSEMA) の構成

図3-1に、送信用トランスレーターの構成を示す。以下に、各ファイルの概要について、説明する。

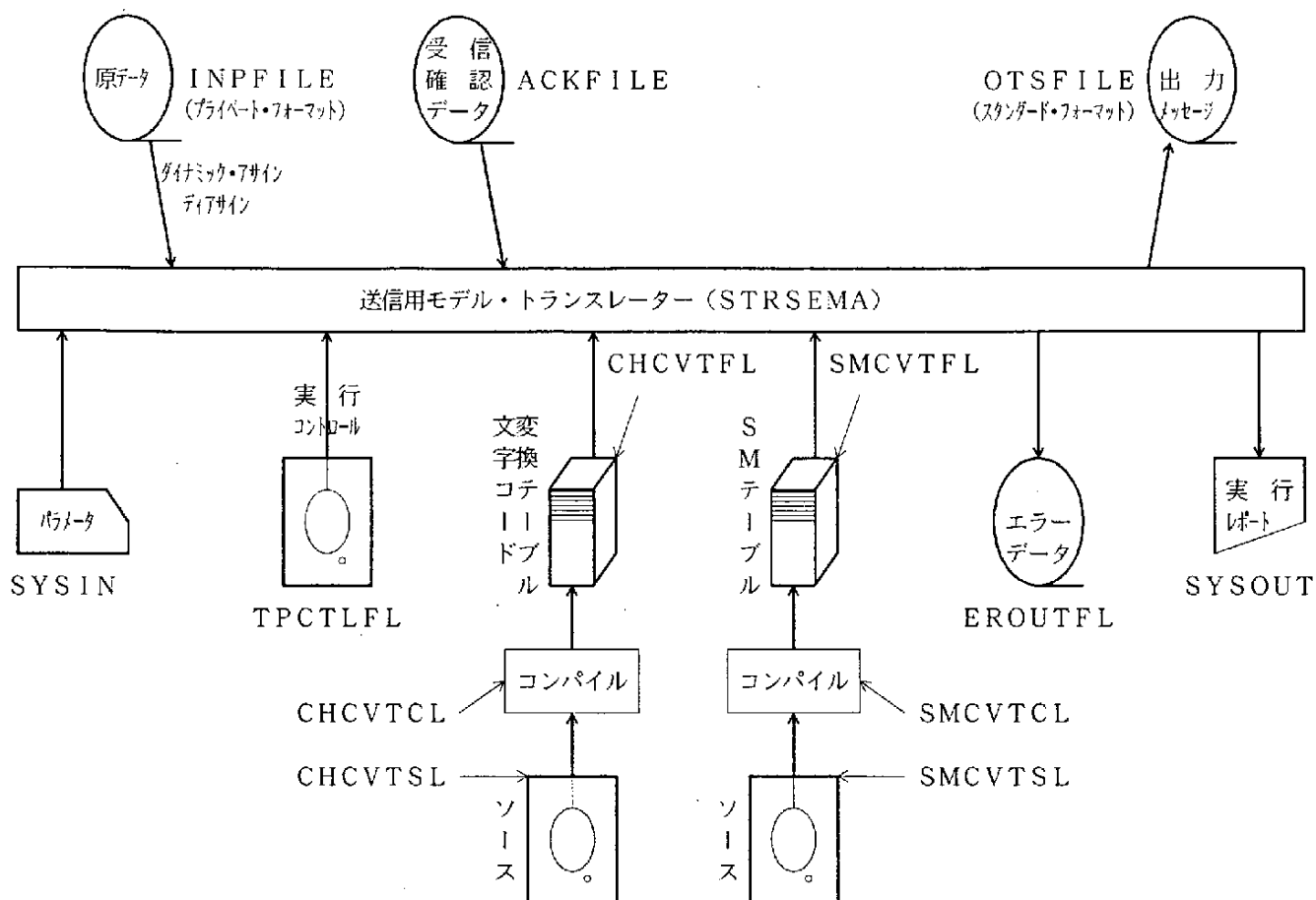


図 3-1 送信用トランスレータの構成

(1) INPFIL E (Input Private Format File)

ユーザー・フォーマット (固定) の送信データを収容したシーケンシャル編成のファイルで、1レコードに1メッセージ分 (1トランザクション分) のすべてのデータが収容されているものとする。最大4000byteの固定長か可変長のレコードで構成されるファイルとする。このファイルは、トランスレータの実行中にダイナミック・アサインのファイルであり、実ファイル名 (ボリューム名) は、TPCTLFLファイルの中の外部ファイル名パラメータで、与えられる。

尚、処理終了後は、ダイナミック・ディアサインされる。

(2) ACKFIL E (Acknowledge File)

送信すべき受信確認メッセージが収容されている最大4000byteレコード長の可変長か、251byteレコード長の固定長ファイルである。

(3) OTSFIL E (Output Standard Format File)

送信用のデータが出力されるシーケンシャル編成のファイルで、下記の属性とする。

- ① TYPE 1 2 通常モードの出力ファイル……………最大4000byteのレコード長の可変長ファイル

- ② TYPE 1 2 分割モードの出力ファイル..... 251byteのレコード長の固定長ファイル
- ③ TYPE-Eの出力ファイル..... 最大4000byteのレコード長の可変長ファイル

(4) EROUTFL (Error Data Output File)

最大4000byteの固定長か可変長レコードのシーケンシャル・ファイルで、変換過程でエラーが発生した入力レコードを、そのまま出力する。

(5) SYSIN

トランスレータに起動の指示と、オプション・パラメータを与えるファイルである。オプションを、以下に示す。

- ① TYPEオプション出力フォーマットのTYPEを決定する。省略は、『TYPE 1 2』を意味する。
 - ② 運用モード・オプションテスト電文の時、指定する。省略は、『通常電文』を意味する。
 - ③ BPID機関オプションシンタックスルール制定機関を指定する。省略は、『C I I 1 0 0』を意味する。
 - ④ 文字コード変換オプション文字コードを変換しない時、指定する。省略は、『JIS-X0201, X0208』への変換を意味する。
 - ⑤ 所属 VANCOD(発信側)
 - ⑥ 発信センターコード
 - ⑦ 発信者コード
-ヘッダーにセットされる。TYPE-Eでは、発信者コードは、無視される。省略は、blank
を意味する。

(6) TPCTFL (Translation Process Control File)

250byte固定長レコードのシーケンシャル編成のファイルで、1レコードで1メッセージグループの生成を、コントロールする。以下に示すパラメータがある。

- ① BPIDが機関 版ヘッダーにセットされる。省略不可である。
 - ② 情報区分ヘッダーにセットされる。省略不可である。
 - ③ 所属 VANCOD(受信側)
 - ④ 受信センターコード
 - ⑤ 受信者コード
-ヘッダーにセットされる。省略は、blankを意味するが、受信者コードは省略できない。TYPE-Eでは、受信者コードは、ヘッダーには、セットされない。
- ⑥ 受信者コード項目No対応するSMCVTF上で、受信者コードと同一の値が入っている項目Noを指定する。省略不可。
 - ⑦ トータル項目No. 1, 2トータル・チェック用項目Noを指定する。省略は、blank (チェック無) を意味する。TYPE-Eでは、無視される。
 - ⑧ 外部ファイル名入力ファイルをダイナミック・アサインするために必要なファイル名である。省略不可である。
 - ⑨ 処理件数オプションこのメッセージグループに含めるメッセージ件数を指定する。省略時は、ファイルのENDまでを意味する。有限の件数を指定し、指定の件数をカウントする前にファイルENDになった場合、その時点で終わる。件数として、ゼロを指定するとゼロ件メッセージの作成指定になる。

⑩ 交換コントロール参照番号.....任意の文字列（Max14文字）とする。TYPE-Eの時のみ有効である。

⑪ 受信確認要求.....要求する時は、『1』をセットする。それ以外（省略を含む）の時、要求無になる。TYPE-Eの時のみ有効である。

⑫ 作成日付時刻オプション.....省略は、処理開始時刻になる。

(7) CHCVTF L (Character Code Conversion Table File)

文字コード変換用テーブルが収容されている 260byte固定長レコードのシーケンシャル編成のファイルである。

(8) SMCVTFL (Standard Message Conversion Table File)

標準メッセージとローカル・フォーマットとの関係を示すテーブルが収容されているファイルで、Max 4000byteの可変長レコードのインデックス・シーケンシャル編成かランダム編成のファイルである。アクセス・キーは、12桁の文字列である（通常、『SM管理機関+サブ機関+情報区分コード+版』）。

(9) SYSOUT

トランスレーターの実行結果を出力する。

(10) CHCVTSL (Character Code Conversion Table Source)

文字コード変換用テーブル（CHCVTF L）を作るためのソース・ファイルで、パソコンのスクリーン・エディターから出力されるMSDOSファイルである。CHCVTCL (Character Code Conversion Table Compiler)で処理され、CHCVTF Lが生成される。

(11) SMCVTSL (Standard Message Conversion Table Source)

SMCVTF Lを作るためのソース・ファイルで、パソコンのスクリーン・エディターから出力されるMSDOSファイルである。SMCVTCL (Standard Message Conversion Table Compiler)で処理され、SMCVTF Lが生成される。

2-2 送信用トランスレーター (STRSEMA) の動作概要

(1) 初期処理

SYSINから実行パラメータを受け取って起動したSTRSEMAは、最初に、『文字コード変換オプション』の指示に従って、CHCVTF Lから文字コード変換テーブルを読み込む。

(2) メッセージグループ作成準備

PCCTLFLから、第一レコードを読み込み、このレコードに指定されているオプションに従って、以下のことを行う。

① INPFILEのダイナミック・アサインとオープン

② SMCVTFLからのSMテーブルの読み込み（アクセス・キーは、通常、『SM管理機関+サブ機関+情報区分コード+版』）。

③ SMテーブルの検査

もし、拡張モードが必要であれば、ヘッダーの拡張モードを『E』とし、必要なければ、『S』とする。

- ④ TYPEに応じたメッセージグループ・ヘッダー・レコード (TYPE-Eの時は、UNAとUNBセグメント) の作成
- ⑤ 文字コードの変換と、OTSFILEへの書き込み。(TYPE-Eの時、UNAとUNBセグメントは、同一レコードになる。)

(3) メッセージの作成

INPFILEから、1レコードを読み込み、以下を行う。

① 情報区分と受信者のチェック

PCCTLFLのオプションで与えられた値と、INPFILEから、読み込んだレコード上の値が、一致するかどうかチェックする。

② メッセージのカウント

③ 縮小モードを、初期セットする。

④ SMテーブルに従って、メッセージを組み立てる。この時、以下を検出したら、拡張モードをセットする。

- ・ 240以上の項目Noの検出
- ・ 明示的マルチ明細 (明細番号付きマルチ明細) の検出

⑤ トータル・チェック値の計算

⑥ 文字コードの変換

(4) メッセージの書き込み

正常にメッセージが作成できれば、OTSFILEに書き込む。TYPE12の分割モードの時は、レコード分割して書き込む。エラーがある時は、入力レコードを、そのまま、EROUTFLに書き込む。

(5) END判定とメッセージグループ・トレーラ・レコードの作成

PCCTLFLのオプションに従って、END判定を行い、メッセージグループ・トレーラ・レコードを作成し、文字コードの変換を行って、OTSFILEに書き込む。(TYPE3では、UNZセグメントとなる。)

(6) INPFILEのクローズとダイナミック・ディアサイン

(7) PCCTLFLのEND判定

PCCTLFLのEND判定を行う (ファイルENDの検出)。処理続行であれば、(2)~(6)を繰り返す。

(8) 受信確認メッセージの付け込み

PCCTLFLのEND (メッセージグループの作成完了) であれば、ACKFILE (受信確認メッセージ) の内容をOTSFILEへコピーする。この時、ACKFILE内の受信確認メッセージのTYPEとSYSINのパラメータで指定されたTYPEとが合致する場合のみ、コピー処理を行う。

(9) 最終処理

処理実行レポートをSYSOUTへ出力し、処理を完了する。

3-4 CHCVTCLとSMCVTCLの動作概要

(1) CHCVTCLの動作概要

CHCVTCLは、'0' ~ 'F' までの文字で作成された文字コード変換テーブル（CHCVTSL）を、バイナリー形式に変換するツールである。

(2) SMCVTCLの動作概要

SMCVTCLは、数字と記号で作成された標準メッセージ・テーブル（SMCVTSL）をバイナリー形式に変換するツールである。この時、以下のような処理が行われる。

マルチ明細内のデータエレメントを除いたデータエレメント（単独項目）を項目No.で昇順にソートする。そして、マルチ明細は、標準メッセージの最後へ移動する。この結果、SMCVTFL内では、先ず、単独項目が項目No.順に並び、その後にマルチ明細が続く形になる。項目No.239以下の単独項目は、常に縮小モードでメッセージが作成されることになり、メッセージ長の短縮に役立つ。

2-4 受信用トランスレーター（STRSDMA）の構成

図3-2に、受信用トランスレーターの構成を示す。以下に、各ファイルの概要について、説明する。

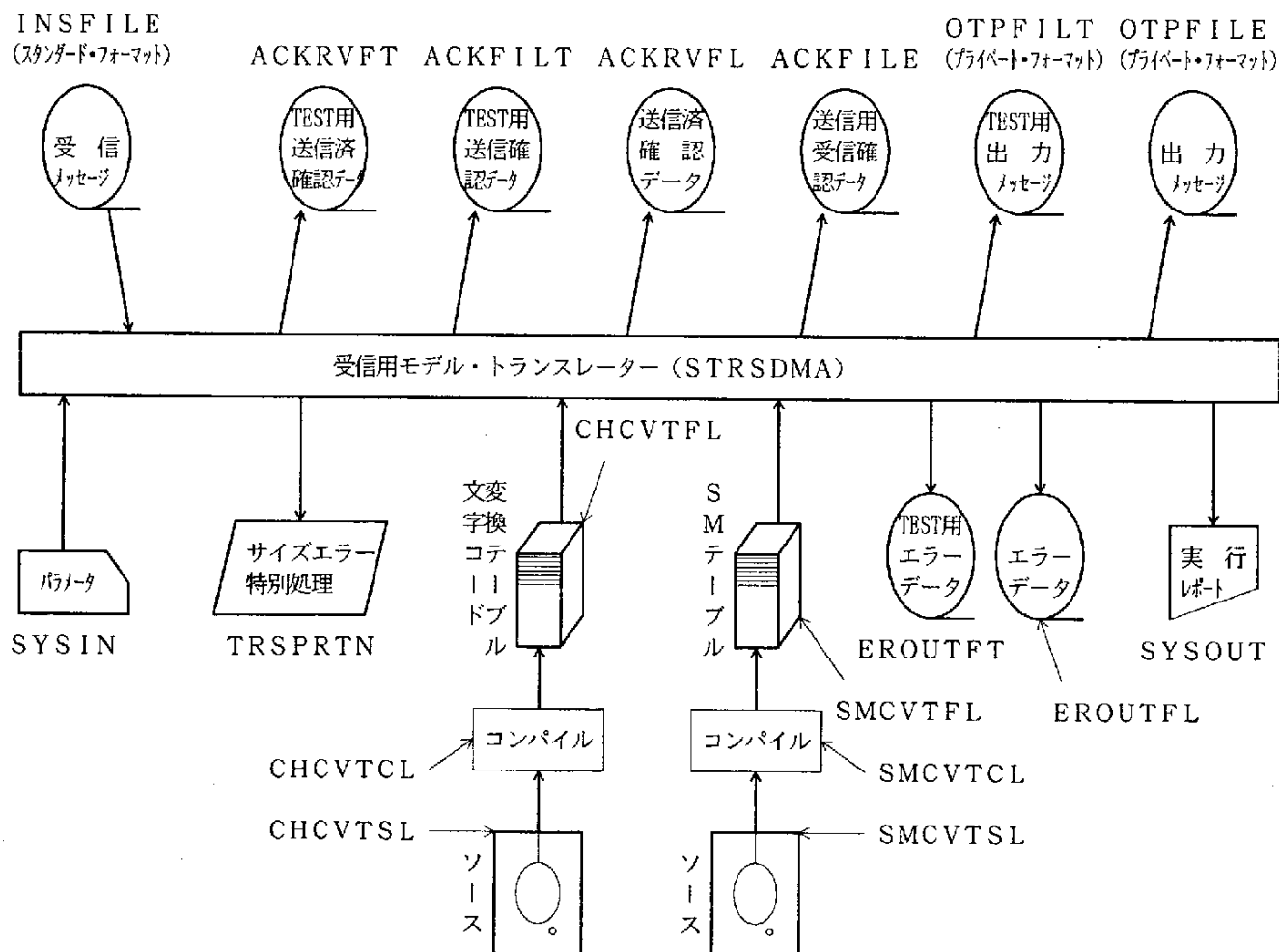


図3-2 受信用トランスレーターの構成

(1) IN S F I L E (Input Standard Format File)

受信したデータが格納されているシーケンシャル編成のファイルで、最大4000byteのレコード長の可変長ファイルとする。

(2) A C K R V F L (Previous Sending Acknowledge File)

送信済メッセージの受信確認メッセージを格納する 251byteレコード長の固定長ファイルである。

(3) A C K R V F T (Previous Sending Acknowledge File For Test)

テスト用送信済メッセージの受信確認メッセージを格納する A C K R V F L と同一形式のファイルである。

(4) A C K F I L E (Acknowledge File)

受信メッセージから作成した受信確認メッセージを格納する 251byteレコード長の固定長ファイルである。

(5) A C K F I L T (Acknowledge File For Test)

テスト用の受信確認メッセージを格納する A C K F I L E と同一形式のファイルである。

(6) O T P F I L E (Output Private Format File)

ユーザー・フォーマット (固定) の受信データを格納するシーケンシャル編成のファイルで、1レコードに1メッセージ分 (1トランザクション分) のすべてのデータが収容できるものとする。最大4000byteの固定長か可変長のレコードで構成されるファイルとする。

(7) O T P F I L T (Output Private Format File For Test)

テスト用のユーザー・フォーマット (固定) の受信データを格納する O T P F I L E と同一形式のファイルである。

(8) E R O U T F L (Error Data Output File)

最大4000byteの固定長か可変長レコードのシーケンシャル・ファイルで、変換過程でエラーが発生した受信メッセージを、TYPE 1かTYPE 3の形で出力する。

(9) E R O U T F T (Error Data Output File For Test)

テスト用のエラー入力データ出力ファイルで、E R O U T F L と同一形式のファイルである。

(10) S Y S I N

トランスレーターに起動の指示と、オプション・パラメータを与えるファイルである。オプションを、以下に示す。

① シンタックスIDオプションシンタックスルール制定機関を指定する。ヘッダーとのチェック用に用いる。省略は、『C I I . 1 0 0』を意味する。

② 受信センターコードヘッダーとのチェック用に用いる。省略は、無チェックを意味する。

③ 文字コード変換オプション文字コード変換をしない時、指定する。省略は、『JIS-X0201, X0208』からの変換を意味する。

④ 受信確認作成オプション受信確認メッセージを作成する時指定する。省略は、作成無しとする。

- ⑤ サイズエラーオプション ……サイズエラーの時、サイズエラー (TRSPRTN) 特別処理をコールするか、無視するかを指定する。省略すると、サイズエラーをエラー処理とし、EROUTFLへメッセージを書き出す。

- (11) CHCVTFL, CHCVTSL, SMCVTFL, SMCVTSL ……文字コード変換テーブルのファイル名を指定する。指定しない場合は、送信用トランスレータ (STRSEMA) のファイルと同一である (10-2-1 参照)。
- (12) SYSOUT ……STRSDMAの実行結果を出力する。

2-5 受信用トランスレータ (STRSDMA) の動作概要

(1) 初期処理

SYSINから実行パラメータを受け取って起動したSTRSDMAは、最初に、『文字コード変換オプション』の指示に従って、CHCVTFLから文字コード変換テーブルを読み込む。

(2) ヘッダーの読み込みと各種チェック

INSFILEから1レコード (ヘッダー) を読み込み、下記の処理とチェックを行う。

- ① 処理完了のチェック
- ② 文字コードの変換
- ③ TYPEの判定 (TYPE12, TYPE-Eのどれか)
- ④ シンタックス制限機関と受信センターコードのチェック
- ⑤ 受信確認メッセージの判定 (受信確認メッセージは、文字コードの変換を行って、ACKRVFLかACKRVFT (テスト・モードの時) へ書き込み。
- ⑥ SMCVTFLからのSMテーブルの読み込み (アクセス・キーは、『SM管理機関+サブ機関+情報区分コード+版』)。

(3) メッセージグループ変換準備

- ① SMテーブルの指示に従った、出力エリアのクリア
- ② INSFILEから1レコードを読み込み、END判定とゼロ件メッセージの判定を行う。
- ③ トレーラー (END) の時、トータル・チェック (トータル・チェック有の時)。さらに、受信確認メッセージの作成と、ACKFILEかACKFILT (テスト・モードの時) への書き込み。この書き込みにおける文字コードは、ローカル・コードである。

(4) メッセージの変換

- ① SMテーブルの指示に従った、フォーマットの変換 (文字コードの変換を含む)。
- ② サイズエラー発生時のサイズエラー・オプション (SYSINで指定) に従った処理。

③ OUPFILE (テストモードの時、OUPFILT) への書き込み (変換成功の時)。または、EROUTFL (テストモードの時OUPFILT) への書き込み (エラー発生時)。

(5) 繰り返し処理

メッセージグループが続く限り、(3)、(4)あるいは(2)からの繰り返し処理。

(6) 最終処理

処理実行レポートをSYSOUTへ出力し、処理を完了する。

2-6 CHCVTCLとSMCVTCLの動作概要

2-3を参照

2-7 TRSPRTN (Translator Special Routine)

SYSINのパラメータの指定により、サイズエラーが発生した時コールされるユーザー・コーディングの特殊プロセスで、これを用いることで、例外的に発生する長大データを自動処理できる。

3. 文字コードについて

バイナリー・データがある時は、文字コードの変換が行われると値が失われる可能性がある。このような時は、SMテーブル上の該当項目に、文字コード変換のバイパスを指定すればよい。トランスレーターは、当該項目についてだけ、文字コード変換を実施しない。

また、送信者と受信者のローカル協定により、文字コード変換を全面的に行わないことも可能であるが、推奨できない。

TYPE-Eで用いるデリミター (X'FC' ~ X'FF') については、ローカル・コードについても同一の値でなければならない。文字コード変換テーブルを作る時、注意しなければならない。

さらに、漢字コードについては、送信者と受信者のローカル協定により、外字を設定することも可能であるが、文字コード変換テーブルの作成に十分注意する必要があり、パソコン対応などを考慮すると、推奨できない。

4. モデル・トランスレーターの機能のまとめ

(1) 送信用トランスレーター

- ① 各種パラメータによる多様な処理
- ② オプション設定により、TYPE 1 2, TYPE-Eの、すべての形式のメッセージを作成できる。
- ③ 縮小モードと拡張モードの自動設定（SMテーブルをEIAJ用に設定（240以上の項目Noがなく、明細番号付きマルチ明細がない）することで、EIAJ標準と同一のメッセージが作成できる）。
- ④ 文字コード変換機能
- ⑤ 処理可能な最大メッセージ長は4000byteで、処理可能な1項目当たりの最大データ長は1000byteである。
- ⑥ 処理可能な1メッセージ当たりの最大項目数は、200項目である。
- ⑦ 処理可能な1メッセージ当たりのマルチ明細数は、最大9明細である。

(2) 受信用トランスレーター

- ① TYPE 1 2, TYPE-Eの、すべての形式のメッセージを解析できる。
- ② 文字コード変換機能
- ③ 処理可能な最大メッセージ長は4000byteで、処理可能な1項目当たりの最大データ長は1000byteである。
- ④ 処理可能な1メッセージ当たりの最大項目数は、200項目である。
- ⑤ 処理可能な1メッセージ当たりのマルチ明細数は、最大9明細である。
- ⑥ オプション設定により、受信確認電文を自動作成できる。
- ⑦ テスト電文を、分離できる。
- ⑧ サイズエラー特別処理機能

