

## JT-H225.0

# パケットに基づくマルチメディア 通信システムのためのシグナリング プロトコルとメディア信号のパケット化

Call Signaling Protocols and Media Stream  
Packetization for Packet Based Multimedia  
Communications Systems

第6版

2008年8月25日制定

社団法人  
情報通信技術委員会

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE

本書は、(社)情報通信技術委員会が著作権を保有しています。

内容の一部又は全部を(社)情報通信技術委員会の許諾を得ることなく複製、転載、改変、転用及びネットワーク上での送信、配布を行うことを禁止します。

## 目次

<参考> .....	7
要約 .....	9
1 適応範囲 .....	10
2 参照 .....	11
3 定義 .....	14
4 規定 .....	14
5 略号 .....	14
5.1 略号全般 .....	14
5.2 RASメッセージの略号 .....	16
6 パケット化と同期のメカニズム .....	17
6.1 アプローチの概要 .....	17
6.2 RTP/RTCPの使用 .....	20
6.2.1 オーディオ .....	21
6.2.2 ビデオメッセージ .....	22
6.2.3 データメッセージ .....	22
7 JT-H225.0 メッセージ定義 .....	23
7.1 JT-Q931 メッセージの使用 .....	23
7.2 共通のJT-Q931 情報要素 .....	25
7.2.1 ヘッダ情報要素 .....	25
7.2.2 メッセージ固有の情報要素 .....	26
7.3 JT-Q931 に基づいたJT-H225.0 呼制御メッセージ詳細 .....	34
7.3.1 呼出 (Alerting) .....	34
7.3.2 呼設定受付 (Call Proceeding) .....	35
7.3.3 応答 (Connect) .....	36
7.3.4 応答確認 (Connect Acknowledge) .....	38
7.3.5 切断 (Disconnect) .....	38
7.3.6 情報 (Information) .....	38
7.3.7 経過表示 (Progress) .....	39
7.3.8 解放 (Release) .....	40
7.3.9 解放完了 (Release Complete) .....	40
7.3.10 呼設定 (Setup) .....	41
7.3.11 呼設定確認 (Setup Acknowledge) .....	44
7.3.12 状態表示 (Status) .....	45
7.3.13 状態問合せ (Status Inquiry) .....	46

7.4 JT-H225.0 呼制御に基づいたJT-Q932 メッセージ詳細 .....	46
7.4.1 ファシリティ (Facility) .....	46
7.4.2 通知 (Notify) .....	48
7.4.3 その他のメッセージ .....	49
7.5 JT-H225.0 呼制御タイマ値 .....	49
7.6 JT-H225.0 共通メッセージ要素 .....	50
7.7 サポート必須のRASメッセージ .....	61
7.8 端末とゲートキーパーの検出メッセージ .....	62
7.8.1 ゲートキーパー要求 (GRQ) .....	62
7.8.2 ゲートキーパー確認 (GCF) .....	63
7.8.3 ゲートキーパー拒否 (GRJ) .....	63
7.9 端末とゲートウェイの登録メッセージ .....	64
7.9.1 登録要求 (RRQ) .....	64
7.9.2 登録確認 (RCF) .....	66
7.9.3 登録拒否 (RRJ) .....	68
7.10 端末/ゲートキーパーの非登録メッセージ .....	68
7.10.1 非登録要求 (URQ) .....	68
7.10.2 非登録確認 (UCF) .....	69
7.10.3 非登録拒否 (URJ) .....	69
7.11 端末からゲートキーパーへの許可メッセージ .....	70
7.11.1 参加要求 (ARQ) .....	70
7.11.2 許可確認 (ACF) .....	71
7.11.3 参加拒否 (ARJ) .....	73
7.12 端末からゲートキーパーへの帯域幅変更要求 .....	73
7.12.1 帯域幅変更要求 (BRQ) .....	73
7.12.2 帯域幅変更確認 (BCF) .....	74
7.12.3 帯域幅変更拒否 (BRJ) .....	75
7.13 位置情報要求メッセージ .....	75
7.13.1 位置情報要求 (LRQ) .....	75
7.13.2 位置情報確認 (LCF) .....	76
7.13.3 位置情報拒否 (LRJ) .....	77
7.14 離脱メッセージ .....	77
7.14.1 離脱要求 (DRQ) .....	77
7.14.2 離脱確認 (DCF) .....	78
7.14.3 離脱拒否 (DRJ) .....	79
7.15 状態要求メッセージ .....	79
7.15.1 情報要求 (IRQ) .....	79
7.15.2 情報要求応答 (IRR) .....	80
7.15.3 情報承認 (IACK) .....	81
7.15.4 情報否定承認 (INAK) .....	82
7.16 非標準メッセージ .....	82
7.17 認識不能なメッセージ .....	82
7.18 ゲートウェイ利用可能資源メッセージ .....	83

7.18.1 利用可能資源通知 (RAI) .....	83
7.18.2 利用可能資源確認 (RAC) .....	83
7.19 RASタイムおよび進行中 (RIP) メッセージ .....	84
7.20 サービス制御メッセージ .....	86
7.20.1 サービス制御識別子 (SCI) .....	86
7.20.2 サービス制御応答 (SCR) .....	86
7.21 ADMISSIONCONFIRMSETSEQUENCE .....	87
<b>8 サービス品質 (QOS) を維持する機構.....</b>	<b>88</b>
8.1 一般的なアプローチと想定 .....	88
8.2 サービス品質(QOS)の測定へのRTCPの使用.....	89
8.2.1 送信側レポート .....	89
8.2.2 受信側レポート.....	89
8.3 オーディオ/ビデオジッタ手順.....	89
8.4 オーディオ/ビデオスキュー手順.....	89
8.5 サービス品質 (QOS) 維持のための手順 .....	89
8.6 エコー制御 .....	90
<b>付属資料 A : RTP/RTCP .....</b>	<b>91</b>
<b>付録資料C : JT-H261 のビデオストリームのRTPペイロードフォーマット (標準) .....</b>	<b>93</b>
<b>付属資料 D : JT-H261AビデオストリームのRTPペイロードフォーマット (標準) .....</b>	<b>93</b>
D.1 はじめに .....	93
D.2 JT-H261A RTPパケット化.....	93
<b>付属資料E : ビデオパケット化 .....</b>	<b>94</b>
E.1 JT-H263 .....	94
<b>付属資料 F : オーディオと多重パケット化.....</b>	<b>95</b>
F.1 JT-G723.1 .....	95
F.2 JT-G728 .....	95
F.3 JT-G729 .....	96
F.4 無音抑圧 .....	101
F.5 GSMコーデック .....	101
F.5.1 フレームパケット化.....	101
F.5.2 関連文献.....	102
F.6 JT-G722.1 .....	102
F.7 TIA / EIA - 136 ACELP .....	103
F.7.1 TIA / EIA - 136 ACELP フレームフォーマット .....	103
F.7.2 TIA / EIA - 136 ACELP 無音抑圧モード.....	103
F.7.3 TIA / EIA - 136 ACELP パケット化.....	104
F.7.4 TIA / EIA - 136 ACELP の参照する標準.....	104
F.8 TIA/EIA-136 USI.....	105
F.8.1 TIA/EIA-136 USI フレームフォーマット .....	105

F.8.2 TIA/EIA-136 USI 無音モードフレーム (TX-DTX) .....	105
F.8.3 TIA/EIA-136 USI パケット化.....	105
F.8.4 TIA/EIA-136 USI の参照する標準.....	105
F.9 IS-127 EVRC.....	105
F.9.1 IS-127 EVRC 解説.....	105
F.9.2 IS-127 EVRC パケット化.....	107
F.9.3 IS-127 EVRCの参照する標準.....	108
F.10 JT-H223 MUX-PDUパケット化.....	108
F.10.1 はじめに.....	108
F.10.2 MUX-PDUパケット化フォーマット.....	108
<b>付属資料G： 管理ドメイン間および管理ドメイン内の通信.....</b>	<b>110</b>
G.1 適用範囲 .....	110
G.2 定義 .....	111
G.3 略号 .....	112
G.4 参照 .....	112
G.5 システムモデル .....	112
G.5.1 階層型構造.....	113
G.5.2 分散型またはフルメッシュ型構造.....	113
G.5.3 クリアリングハウス型構造.....	113
G.5.4 結合ポイント.....	114
G.5.5 管理ドメインのオーバーラップ.....	114
G.6 オペレーション .....	114
G.6.1 JT-H501 メッセージの利用.....	114
G.6.2 アドレステンプレートと記述子.....	115
G.6.3 1つの同位エレメント、または同位エレメント群の発見.....	117
G.6.4 解決手順.....	117
G.6.5 利用情報の交換.....	118
G.6.6 番号ポータビリティ情報シグナリング .....	119
G.7 シグナリングの例 .....	120
G.7.1 分散型または、フルメッシュ型構造.....	120
G.9.2 クリアリングハウス .....	123
G.8 付属資料Gプロファイル.....	128
G.8.1 イントロダクション.....	128
G.8.2 プロファイル"A": 信頼度の高いゲートキーパーへのゾーン間呼ルーティング 129	
G.8.2.2 要求フィールド.....	131
G.8.2.3 要求手順.....	131
<b>付属資料H： JT-H225.0 メッセージ シンタックス(ASN.1).....</b>	<b>133</b>
<b>付属資料 I： JT-H263 第2版ビデオのパケット化.....</b>	<b>162</b>
<b>付録 I： RTP/RTCPアルゴリズム .....</b>	<b>163</b>

付録Ⅱ： RTPプロファイル.....	164
付録Ⅲ： JT-H261 パケット化.....	165
付録Ⅳ： 異なるパケットベースネットワークプロトコルスタックにおけるJT-H225.0 の動作 .....	166
IV.1 TCP/IP, UDP/IP.....	166
IV.1.1 ゲートキーパーの発見.....	166
IV.1.2 エンドポイントとエンドポイントの通信.....	169
IV.2 SPX/IPX.....	169
IV.2.1 ゲートキーパーの発見.....	169
IV.2.2 エンドポイントとエンドポイントの通信.....	170
IV.3 SCTP.....	170
IV.3.1 ストリーム.....	170
IV.3.2 ペイロード・プロトコル識別子.....	170
付録Ⅴ： JT-H225.0 におけるASN.1 の使用.....	171
V.1 タギング.....	171
V.2 タイプ.....	171
V.3 制約と範囲.....	171
V.4 拡張性.....	171
付録Ⅵ： JT-H225.0 トンネリングシグナリングプロトコルの識別子.....	172

パケットに基づくマルチメディア通信システムのための  
シグナリングプロトコルとメディア信号のパケット化

<参考>

1. 国際勧告などとの関連

本標準は、パケットに基づくマルチメディア通信システムのためのシグナリングプロトコルとメディア信号のパケット化について規定しており、2006年5月に発行されたITU-T勧告H.225.0、2007年1月に発行されたITU-T勧告H.225.0 Amendment 1に準拠している。

2. 上記国際勧告などに対する追加項目など

2. 1 オプション選択項目

なし

2. 2 ナショナルマター決定項目

なし

2. 3 その他

なし

3. 改版の履歴

版数	発行日	改版内容
第1版	1996年11月27日	制定
第2版	1998年11月26日	改定
第2.1版	1999年4月22日	改定
第3版	2000年4月20日	改定
第3.1版	2000年11月30日	付属資料G追加
第4版	2001年11月27日	本文改定、付属資料B改定、付属資料E改定、 付属資料F改定、付属資料H改定、付録IV改定、 付録V追加、付録VI追加
第4.1版	2002年11月28日	2002年2月版インプリメンターズガイドの反 映
第5版	2004年4月20日	本文改定、付属資料A, B外部参照、付属資料 G改定
第6版	2008年8月25日	割り当てゲートキーパー手順、H.361、 H.460.21のためのASN.1と関連記述の追加、 Amendment 1およびインプリメンターズガイド の反映

4. 工業所有権

本標準に関わる「工業所有権の実施の権利に係る確認書」の提出状況は、TTCホームページでご覧にな



れます。

## 5. その他

### (1) 参照している勧告、標準など

TTC標準 : JT-G722、JT-G723. 1、JT-G728、  
JT-G729、JT-H221、JT-H230、  
JT-H233、JT-H242、JT-H243、  
JT-H245、JT-H263、JT-H261、  
JT-H320、JT-T122、JT-T123、  
JT-T125、JT-H321、JT-H322、  
JT-H324、JT-H310、JT-Q931、  
JT-Q932、JT-Q850、JT-Q950  
ITU-T勧告 : G. 711、X. 680、X. 681、X. 691、  
E. 164、H. 235、H. 248  
ISO/IEC : 10646-1、11571  
IETF標準 : RFC1738、RFC2068、RFC1766、  
RFC 2429、RFC2190

## 6. 標準作成部門

メディア符号化専門委員会

## パケットに基づくマルチメディア通信システムのための

### シグナリングプロトコルとメディア信号のパケット化

TTC は、

I シリーズ標準における狭帯域 ISDN の特性スペックに従うネットワーク上で、ビデオ電話およびビデオ会議サービスを行う JT-H320 の広範囲な適応および使用の拡大について、検討し、

JT-H320 端末との相互接続能力を保ち続けながら、全てもしくは部分的に、ローカルエリアネットワーク上で上記のサービスを実現することへの要望や利益について、評価し、

多くのタイプのローカルエリアネットワークの特性や性能の潜在的な利益について、指摘し、

これらの設備を提供するために、JT-H322 または JT-H323 の要求条件に準拠するシステムや機器が、利用されることを、推奨する。

#### 要約

本標準は、各々が非保証サービス品質 (QOS) を提供するために配置／運用されている、1 つもしくは複数のパケットベースネットワークを含む伝送パスにおいて、JT-H200/ JT-AV120 シリーズに定義されている狭帯域ビジュアル電話サービスのための技術的必要条件をカバーしている。本標準は、狭帯域 ISDN の JT-H320 端末で規定されている機能以上に、付加的な保護やリカバリー機能を必須とする。

JT-H322 は、JT-H323/JT-H225.0 では仮定されていない下位レイヤの伝送性能を持った、その他の LAN での使用について述べていることに注意すること。

本標準は、パケットベースネットワーク上で、JT-H323 装置を用いた、会議サービスを提供するための、オーディオ、ビデオ、データ、制御情報の管理方法を示す。

JT-H225.0 第 6 版に準拠することを要求する製品は、本標準の全ての必須要求に従わなくてはならない。第 6 版製品であることは、**protocolIdentifier** 値が {itu-t (0) recommendation (0) h (8) 2250 version (0) 6} であるものを含む JT-H225.0 メッセージにより識別される。

今回の改定で、次の機能が導入される。

- 1) 10 から 14 の数値符号 (digit code) をサポートするための JT-H225.0 AliasAddress の拡張
- 2) 自分自身では何も登録しないエンドポイントに対し E.164 アドレスを割り当てるというゲートキーパーの能力を追加
- 3) JT-H225.0 AdmissionRejectReason に帯域不足エラーコードを追加
- 4) 割り当てゲートキーパー手順 (Assigned Gatekeeper procedures) で必要となる ASN.1 及び記述の変更
- 5) 1 または 8 の PI 値を受信した際の T310 タイマの再スタートする要件を追加し、7.5 節を変更
- 6) H.361 の変更を考慮し、JT-H225.0 の ASN.1 を変更
- 7) 新しい勧告 H.460.21 における LRQ 構造と RRQ 構造の language フィールドの追加に対処するため ASN.1 の規定及び記述を変更
- 8) ASN.1 規定における unallocatedNumber のコメントで誤字を修正

本改定では、文章の明確化と、開発者のガイドで以前指定された誤りの修正も行った。具体的には、LocationRejectReason/ AdmissionRejectReason や AccessRejectionReason/ AdmissionRejectReason のマッピングテーブルの追加、ゲートキーパーによる additionalSourceAddresses 挿入の記述の明確化、h245Address を運ぶファシリティ (Facility) メッセージ使用時の記述の明確化、UUIE フィールドの長さに関する表記の修正を行った。

## 1 適応範囲

本標準は、パケットベースネットワークにおいて、JT-H323 機器間転送のためのオーディオ、ビデオ、データ、制御情報の関連付け、コード化、パケット化の方法を規定する。本標準は、狭帯域 ISDN 上で動作している JT-H320 端末、GSTN 上の JT-H324 端末、広帯域 ISDN 上の JT-H310/JT-H321 端末それぞれとの接続に、JT-H323 ゲートウェイの使用を含む。JT-H323 では機器および手順を記述し、本標準がプロトコルおよびメッセージのフォーマットを記述する。JT-H323 ゲートウェイから、サービス品質 (QOS) 保証型 LAN のための JT-H322 ゲートウェイへ、そして JT-H322 エンドポイントへの通信が可能となる。

本標準は、IEEE802.3、トークンリング、その他を含む異なった様々なパケットベースネットワーク上で動作することを意図している。よって本標準では、TCP/IP、UDP/IP や SPX/IPX のようなトランスポート層の上位層として定義される。本標準の付録IVはトランスポートプロトコル群の特性の詳細を含んでいる。従って JT-H225.0 通信の適応範囲は、同一トランスポートプロトコルを使用する同一パケットベースネットワークにおける JT-H323 エンティティ間である。このパケットベースネットワークは単一セグメントネットワークであったり、リングネットワーク、1つの相互接続ネットワークを生成するためのブリッジやルータで接続した複数種のパケットベースネットワークにより構成される論理的なエンタープライズ・データネットワークでもよい。すべてのインターネット上もしくはいくつかのパケットベースネットワークを接続した場合、JT-H323 端末の動作は低い性能でしか発揮できないかもしれないことは強調されるべきである。このようなパケットベースネットワークや一般的なインターネットにおいて、QOS を保証する手段を与えることは、本標準の範囲外である。しかしながら、パケットベースネットワークの訂正動作の手順や輻輳に起因する品質の問題と断定する手段を、本標準は JT-H323 端末のユーザのために提供する。また、複数の JT-H323 ゲートウェイを公衆 ISDN ネットワークにて接続することがサービス品質を保つ簡単な方法であることに留意すること。

JT-H323 と本標準は、サービス品質 (QOS) 非保証のパケットベースネットワーク環境における会議/接続の JT-H320 と JT-H221 の拡張を狙っている。第一の会議モデル<sup>1</sup>は、数人から数千人の規模で、大規模なブロードキャスト、強力な認可制御や厳密な会議制御を伴う。

本標準は、全てのパケットベースネットワーク (付属資料 A, B, C 参照) において、メディア信号のパケット化および同期のためにリアルタイム トランスポート プロトコル/リアルタイム トランスポート制御プロトコル (RTP/RTCP) を使用する。本標準においての RTP/RTCP の使用法は、TCP/IP、UDP/IP 使用のいかなる方法においても拘束されるものではない。本標準は、RTP 以外の伝送手段による呼確立および能力交換 (JT-H323, JT-H245 参照) の初期シグナル後、1つもしくは複数 RTP/RTCP コネクションを確立する呼モデルを仮定している。

JT-H221 においては、オーディオ、ビデオ、データと、制御信号は、1つもしくは複数の同期した物理SCN呼に多重化される。一方、JT-H323 呼のパケットベースネットワーク側は、このような概念は適応されない。本標準では、SCNでのJT-H221 の概念である $P \times 64\text{kbps}$ 、例えば、 $2 \times 64\text{kbps}$ 、 $3 \times 64\text{kbps}$ などを適応する必要はない。このようにパケットベースネットワーク側は、例えば、 $2 \times 64\text{kbps}$ の固定レート呼ではなく、 $128\text{kbps}$ の最大レートを上限とした唯一のコネクション呼である。他に、SCN側<sup>2</sup>の  $6 \times 64\text{kbps}$ と、 $384\text{kbps}$ の最大レートを上限とした単一コネクション・パケットベースネットワーク呼の相互接続がある。このアプローチの基本原則は、端末よりもゲートウェイに複雑さを配置することであり、JT-H320 のISDNを前提とした要素を、パケットベースネットワークにまで適応することは避けるべきである。

---

<sup>1</sup> オプションである放送型のみでの会議モデルは検討中である。必然的に放送型モデルは、厳密な認可制御や会議制御は提供しない。

<sup>2</sup> LAN 側のビデオとデータのレートは、SCN 側の JT-H320 多重におけるビデオとデータのレートに一致しなければならない。オーディオと制御のレートは一致することを要求しない。その他の一般的な方法は、JT-H245 のフロー・コントロールの使用である。LAN/SCN ゲートウェイは、ビデオとデータのレートを強制的に、JT-H221 SCN 多重に合わせる。しかしながら、オーディオは度々ゲートウェイで符号変換され、LAN と SCN のオーディオ・レートが一致しないことがある。一般的に制御のための JT-H221 ビットレート(800bps)は、LAN 側の JT-H225 ビットレートと一致することを期待してはいけない。同様に、LAN のレートは、ビデオとデータのいずれか、もしくは両方のための SCN レートを下回るかもしれない。しかし、SCN 側の多重の最大値の合計を超えることはできない。

一般的に、JT-H323 端末は、JT-H323 ゲートウェイを介して相互接続している間は、JT-H320 の転送レートを直接意識することはない。その代わりに、ゲートウェイは、メディアの速度を JT-H221 多重により許可された各々の論理チャンネルに制限するために、JT-H245 **FlowControlCommand** メッセージを使用する。ゲートウェイは、パケットベースネットワーク側のビデオ信号の速度を、SCN 側の速度に対し十分低い（またはその逆）速度を許可するかもしれない。これらのオペレーションの詳細は、JT-H323 および本標準の適応範囲を超えている。JT-H323 端末は JT-H245 メッセージ中のビデオ最大ビットレート領域より、間接的に JT-H320 の転送速度を知ることができる。よって、JT-H323 端末は、このレート以上に転送速度を上げてはならない。

本標準は、JT-H323 ゲートウェイを介して、JT-H320(1990)、JT-H320(1993)、そして JT-H320(1996)との相互接続が可能となるよう設計される。しかし、本標準の幾つかの特徴として、JT-H320 の将来のバージョンにおいて拡張オペレーションにより直接接続が許可されるかもしれない。JT-H320 側のサービス品質 (QoS) が、JT-H323 ゲートウェイの特徴や能力により異なるものとなってもよい。

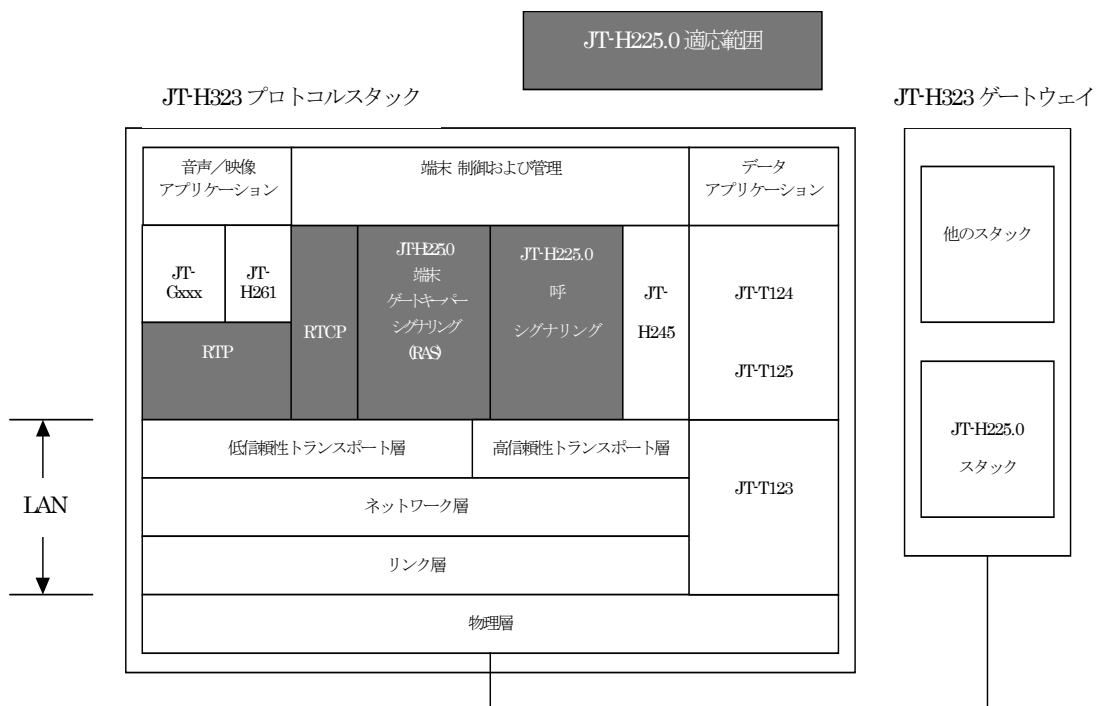


図 1/JT-H225.0 JT-H225.0 の適応範囲 (ITU-T H.225.0)

本標準の一般的なアプローチは、パケットベースネットワーク/伝送設備を利用し、パケットの同期を提供することにある。本標準においては、全てのメディア信号と制御信号を単一のストリームに多重化（単一のパケット化）するものではない。JT-H221 のフレーム構造は、以下の理由で利用しない。

JT-H221 を使用しないことで各々のメディアに対して、異なる適切なエラー処理が受けられる。

JT-H221 は、ランダムなビットのバーストエラーに敏感であり、パケット化はパケットベースネットワークにおいて、より大きな信頼性を与える。

JT-H245 および JT-H225.0 呼制御メッセージは、パケットベースネットワーク上で高信頼性リンクにより送ることができる。

JT-H245 は JT-H242 と比べて柔軟性や能力が優れている。

## 2 参照

以下に示す TTC 標準および他の参考文献は、本標準の前提および参照すべき事柄を規定している。発行時に有効であったバージョン（版）が示されている。すべての標準および他の参考文献は、バージョン（版）が更新されることがある。この標準の全てのユーザは、標準や他の参考文献の最新版を適応することの可能性を調べるのが望ましい。現在有効な TTC 標準のリストが定期的に発行される。

[1] CCITT Recommendation G.711 (1988), *Pulse Code Modulation (PCM) of voice frequencies*.

(TTC 注：G.711 に関しては、既に TTC より標準文書「TTC 標準 JT-G711: 音声周波数帯域信号の PCM

符号化方式」が発行されている。しかし JT-G711 では、日本国内事情により A 則符号化方式および A 則  $\leftrightarrow \mu$  則の相互変換に関する記述が削除されているため、本標準では敢えて国際標準である G.711 を参照している。)

- [2] TTC 標準 JT-G722:64kbit/s 以下の 7kHz オーディオ符号化方式  
CCITT Recommendation G.722 (1988), *7 kHz audiocoding within 64 kbit/s*.
- [3] TTC 標準 JT-G728:低遅延符号励振線形予測(LDCELP)を用いた 16kbit/s 音声符号化  
CCITT Recommendation G.728 (1992), *Coding of speech at 16 kbit/s using lowdelay code excited linear prediction*.
- [4] TTC 標準 JT-G723.1:マルチメディア通信伝送のための 5.3 および 6.3kbit/s 音声符号化方式  
ITU-T Recommendation G.723.1 (2006), *Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s*.
- [5] TTC 標準 JT-G729:8kbit/s CS-ACELP を用いた音声符号化方式  
ITU-T Recommendation G.729 (1996), *Coding of speech at 8 kbit/s using conjugate structure algebraiccodeexcited linearprediction (CS-ACELP)*.
- [6] TTC 標準 JT-H221:オーディオビジュアルテレサービスにおける 64Kbit/s から 1920kbit/s チャネルフレーム構成  
ITU-T Recommendation H.221(2004), *Frame Structure for a 64 to 1920 Kbit/s channel in audiovisual teleservices*.
- [7] TTC 標準 JT-H230:オーディオビジュアルシステムのためのフレーム同期の制御信号と通知信号  
ITU-T Recommendation H.230(2004), *Frame Synchronous Control and Indication Signals for Audiovisual Systems*, December 1990
- [8] TTC 標準 JT-H233: オーディオビジュアルサービスのための機密保持システム  
ITU-T Recommendation H.233(2002) *Confidentiality System for Audiovisual Services*.
- [9] TTC 標準 JT-H242:2Mbit/s までのデジタルチャネルを利用したオーディオビジュアル端末間の通信を設定する方式  
ITU-T Recommendation H.242(2004), *System for establishing communication between audiovisual terminals using digital channels up to 2 Mbit/s*.
- [10] TTC 標準 JT-H243:2Mbit/s までのデジタルチャネルを利用した 3 箇所以上のオーディオビジュアル端末の通信確立手順  
ITU-T Recommendation H.243(2005), *Procedures for establishing communication between three or more audiovisual terminals using channels up to 1920kbit/sec*.
- [11] TTC 標準 JT-H245:マルチメディア通信制御用プロトコル  
ITU-T Recommendation H.245 (2006), *Control protocol for multimedia communications*.
- [12] TTC 標準 JT-H261:p×64kbit/s オーディオビジュアルサービス用ビデオ符号化方式  
ITU-T Recommendation H.261<sup>3</sup> (1993), *Video codec for audiovisual services at p × 64 kbit/s*.
- [13] TTC 標準 JT-H263:低ビットレート通信用ビデオ符号化方式  
ITU-T Recommendation H.263 (2005), *Video coding for low bit rate communication*.

- [14] TTC 標準 JT-H320:狭帯域テレビ電話会議システムとその端末  
ITU-T Recommendation H.320 (2004), *Narrowband visual telephone systems and terminal equipment*.
- [15] TTC 標準 JT-T122:オーディオグラフィック会議のための多地点通信サービス—サービス定義  
ITU-T Recommendation T.122(1998), *Multipoint Communication Service - Service Definition*.
- [16] TTC 標準 JT-T123: マルチメディア会議のためのネットワーク特定データプロトコルスタック  
ITU-T Recommendation T.123(1999), *Network specific data protocol stacks for multimedia conferencing*.
- [17] TTC 標準 JT-T125:多地点通信サービス—プロトコル仕様  
ITU-T Recommendation T.125(1998), *Multipoint Communication Service Protocol Specification*.
- [18] TTC 標準 JT-H321:TV 電話会議システムとその端末装置の B-ISDN 環境への適用  
ITU-T Recommendation H.321 (1998), *Adaptation of H.320 visual telephone terminals to B-ISDN environments*.
- [19] TTC 標準 JT-H322:サービス品質(QoS)の保証されたローカルエリアネットワーク(LAN)で用いられる  
ビジュアル電話システムと端末  
ITU-T Recommendation H.322 (1996), *Visual telephone systems and terminal equipment for local area networks which provide a guaranteed quality of service*.
- [20] TTC 標準 JT-H324:低ビットレートマルチメディア通信用端末  
ITU-T Recommendation H.324 (2005), *Terminal for low bit rate multimedia communication*
- [21] TTC 標準 JT-H310:広帯域オーディオビジュアル通信システムと端末  
ITU-T Recommendation H.310 (1998), *Broadband and audiovisual communication systems and terminals*.
- [22] TTC 標準 JT-Q931:ISDN ユーザ網インタフェース レイヤ 3 仕様  
ITU-T Recommendation JT-Q931 (1998), *ISDN user-network interface layer 3 specification for basic call control*
- [23] TTC 標準 JT-Q932:デジタル加入者信号システム No1 —ISDN 付加サービス制御手順の共通原則  
ITU-T Recommendation Q.932 (1998), *Digital subscriber Signalling System No.1 — Generic procedures for the control of ISDN supplementary services*
- [24] ITU-T X.680 (2002) | ISO/IEC 8824-1:2002, *Information technology – Abstract Syntax Notation One (ASN.1) Specification of basic notation*.
- [25] ITU-T X.681 (2002), *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification*.
- [26] ITU-T X.691 (2002) | ISO/IEC 8852-2:2002, *Information technology – ASN.1 encoding rules – Specification of Packed Encoding Rules (PER)*.
- [27] ITU-T E.164 (2005), *The international public telecommunication numbering plan*.
- [28] ISO/IEC 10646:2003 *Information technology – Universal Multiple-Octet Coded Character Set (UCS)*.
- [29] TTC 標準 JT-Q850: デジタル加入者線信号方式 No.1(DSS1)および No.7 信号方式 ISDN ユーザ部(ISUP)における理由表示の使用法および生成源  
ITU-T Recommendation Q.850(1998), *Usage of cause and location in the Digital Subscriber signaling system No. 1 and the signaling system No. 7 ISDN user part*
- [30] TTC 標準 JT-Q950:ISDN ユーザ網インタフェース 付加サービスのプロトコル、構造及び一般原則  
ITU-T Recommendation Q.950 (1997), *Supplementary services protocols, structure and general principles*
- [31] ITU-T H.235.0 (2005), *H.323 security: Framework for security in H-series (H.323 and other H.245-based multimedia systems)*.

- [32] ISO/IEC 11571:1998, *Information technology – Telecommunications and information exchange between systems – Private Integrated Services Networks – Addressing*.
- [33] IETF RFC 1738 (1994), *Uniform Resource Locators (URL)*.
- [34] IETF RFC 2068 (1997), *Hypertext Transfer Protocol – HTTP/1.1*.
- [35] IETF RFC 1766 (1995), *Tags for the Identification of Languages*.
- [36] ITU-T H.248.1 (2005), *Gateway Control Protocol: Version 3*.
- [37] IETF RFC 3550 (2003), *RTP: A Transport Protocol for Real-Time Applications*.
- [38] IETF RFC 3551 (2003), *RTP Profile for Audio and Video Conferences with Minimal Control*.
- [39] IETF RFC 2032 (1996), *RTP Payload Format for H.261 Video Streams*.
- [40] ITU-T Recommendation X.690 (2002) | ISO/IEC 8825-1:2002, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*.

### 3 定義

JT-H323 の定義を参照のこと。JT-H323 の用語「エンドポイント」は、受信および発信の行える要素である端末、ゲートウェイ、MCU を指す。本標準では、用語「端末」は、呼設定の記述の中で一般的にしばしば使用される。「端末」は、呼設定の章においては、ゲートウェイおよび MCU を含むことを理解すべきである。

### 4 規定

本標準では、必須条件は「しなければならない」、推奨されるオプション特性や手順は「すべきである」と記述する。推奨を表示していないオプションについては「してもよい」を記述する。

用語「MCU」は、JT-H323 の MCU を指す。JT-H231 の MCU を指す場合は、特にそのことを明記する。

本標準では、kilobits/sec は、短縮された kbit/s と記述し、1000 の単位系で測定される。64kbit/s は、正確には毎秒 64,000 ビットとなる。

特に指定しない場合は、ASN.1 の可変 PER 符号化は、本標準の全ての ASN.1 で使用される。

JT-Q931 メッセージ名は、全て大文字で示す。ASN.1 はボールド体で示す。

### 5 略号

本文書では、次の略語が使用される。

#### 5.1 略号全般

BAS	ビットレート割り当て信号 (Bit rate Allocation Signal)
CIF	共通中間フォーマット (Common Intermediate Format)
CRV	呼参照値 (Call Reference Value)
ECS	暗号化制御信号 (Encryption Control Signal)
FFS	今後の検討課題 (For Further Study)
GOB	グループ オブ ブロック (Group of Blocks)
H-MLP	高速マルチレイヤプロトコル (High speed MultiLayer Protocol)

HSD	高速データ (High Speed Data)
IA5	International Alphabet No.5
IE	情報要素 (Information Element)
IETF	Internet Engineering Task Force
IP	インターネットプロトコル (Internet Protocol)
LAN	ローカル エリア ネットワーク (Local Area Network)
LD-CELP	低遅延符号励振線型予測 (Low Delay – Code Excited Linear Prediction)
LSB	最下位ビット (Least Significant Bit)
LSD	低速データ (Low Speed Data)
MB	マクロブロック (Macro Block) [JT-H261 参照]
MBE	多ビット拡張 (MultiByte Extension)
MCC	Multipoint Command Conference
MCN	Multipoint Command Negating
MCS	Multipoint Command Symmetrical Data Transmission
MCS	多地点通信サービス (Multipoint Communication Service)
MCU	多地点制御ユニット (Multipoint Control Unit)
MF	多重化フレーム (MultiFrame)
MLP	マルチレイヤ プロトコル (MultiLayer Protocol)
MPI	Minimum Picture Interval
MSB	最上位ビット (Most Significant Bit)
NA	非適用 (Not Applicable)
NS	非標準 (NonStandard)
NSAP	ネットワーク層サービスアクセスポイント (Network Service Access Point)
PCM	Pulse Code Modulation
PDU	プロトコルデータユニット (Protocol Data Unit)
QCIF	1/4 CIF (Quarter Common Intermediate Format)
QOS	サービス品質 (Quality of Service)
RAS	登録, 承認, 状態表示 (Registration, Admission, and Status)
RTCP	リアルタイム トランスポート 制御プロトコル (Realtime Transport Control Protocol)
RTP	リアルタイム トランスポート プロトコル (Realtime Transport Protocol)
SBE	単バイト拡張 (Single Byte Extension)
SC	サービスチャネル (Service Channel)
SCM	Selected Communications Mode
SCN	回線交換網 (Switched Circuit Network)
TCP	トランスポート コントロール プロトコル (Transport Control Protocol)
TSAP	トランスポート層サービスアクセスポイント (Transport Service Access Point)



URL	Uniform Resource Locator
UDP	ユーザ データグラム プロトコル (User Datagram Protocol)
VCF	Video Command “Freeze picture Request”
VCU	Video Command “Fast Update Request”

## 5.2 RASメッセージの略号

ACF	参加確認 (Admissions Confirm)
ARJ	参加拒否 (Admissions Reject)
ARQ	参加要求 (Admissions Request)
BCF	帯域幅変更確認 (Bandwidth Confirm)
BRJ	帯域幅変更拒否 (Bandwidth Reject)
BRQ	帯域幅変更要求 (Bandwidth Request)
DCF	離脱確認 (Disengage Confirm)
DRJ	離脱拒否 (Disengage Reject)
DRQ	離脱要求 (Disengage Request)
GCF	ゲートキーパー確認 (Gatekeeper Confirm)
GRJ	ゲートキーパー拒否 (Gatekeeper Reject)
GRQ	ゲートキーパー要求 (Gatekeeper Request)
IACK	情報承認 (Information Request Acknowledgement)
INAK	情報否定承認 (Information Request Negative Acknowledgement)
IRQ	情報要求 (Information Request)
IRR	情報要求応答 (Information Request Response)
LCF	位置情報確認 (Location Confirm)
LRJ	位置情報拒否 (Location Reject)
LRQ	位置情報要求 (Location Request)
RAC	資源利用可能確認 (Resource Availability Confirmation)
RAI	資源利用可能通知 (Resource Availability Indication)
RCF	登録確認 (Registration Confirm)
RIP	進行中 (Request In Progress)
RRJ	登録拒否 (Registration Reject)
RRQ	登録要求 (Registration Request)

SCI	サービス制御通知 (Service Control Indication)
SCR	サービス制御応答 (Service Control Response)
UCF	非登録確認 (Unregistration Confirm)
URJ	非登録拒否 (Unregistration Reject)
URQ	非登録要求 (Unregistration Request)

## 6 パケット化と同期のメカニズム

### 6.1 アプローチの概要

エンドポイントは、いかなる呼の設定の以前に、ゲートキーパーに発見／登録されてもよい。この場合、エンドポイントは自身が登録されているゲートキーパーの版を知ることは望ましい。また、ゲートキーパーは、登録されたエンドポイントの版を知ることが望ましい。このような理由で、発見と登録両者のシーケンスに、JT-H245 スタイルの OBJECT IDENTIFIER (JT-H323 実装バージョンの記号を型番として割り振られた) を含む。このシーケンスには、非標準なエンドポイントを割り当てるために、オプションとして非標準メッセージ領域を含んでもよい。このシーケンス後には、ゲートキーパーとエンドポイントは、お互いのバージョン (版) 番号と非標準のステータスを知る。

2 つのエンドポイントに各々の型番と非標準ステータスの情報を知らせるために、呼設定／接続のシーケンスにおいて、バージョン (版) 番号は必須であり、非標準情報はオプションである。しかしながら、全ての JT-H225.0 呼制御メッセージは、ユーザ・ユーザ情報要素にオプションの非標準メッセージ領域を持っていて、全ての RAS チャンネルメッセージ非標準情報のためにオプション領域を持っていることに注意すること。加えて、非標準 RAS メッセージは、いつでも送ることができると定義されている。

登録、承認、状態メッセージのための低信頼性チャンネルはRASチャンネルと呼ばれる。呼を開始する一般的なアプローチは、RASチャンネル<sup>4</sup>に必須の参加要求を送り、その後の初期呼設定メッセージは高信頼性トランスポートアドレス (このアドレスは、参加確認メッセージにて返されるか、あらかじめ発呼端末が知っている) のチャンネルにより転送される。この初期メッセージの結果、呼設定シーケンスは、後述の拡張 JT-H225.0 呼制御のオペレーションに基づいて開始される。シーケンスは、端末が応答メッセージにて、JT-H245 制御メッセージ<sup>5</sup>を送信するための高信頼性のトランスポートアドレスを受信した時に完了する。

高信頼性 JT-H225.0 シグナリングチャンネルにてメッセージが送られる時は、単一のメッセージはそれ全体として、高信頼性トランスポートにて規定される最大 PDU 長内で送信しなければならない。すなわち、トランスポート PDU を使用して JT-H225.0 メッセージを分割してはいけない (付録IVに IP における実装のアウトラインを示す。この PDU は TPKT により規定される)。

一度、高信頼性JT-H245 制御チャンネルが確立されると、オーディオ、ビデオ、データのチャンネルの追加は、JT-H245 論理チャンネル手順を使用して能力情報交換の結果、確立される。さらに、パケットベースネットワーク側の性質を利用したマルチメディア会議 (集中／非集中／多地点) では、コネクション毎<sup>6</sup>にネゴシエーションが行われる。このネゴシエーションは、例えば、データと制御は集中させ、オーディオ／ビデオは非集中とする様に、メディア毎に行われる。

高信頼性 JT-H245 制御チャンネルにてメッセージが送られる時は、1 つ以上のメッセージを、高信頼性トラ

<sup>4</sup> ゲートキーパーに登録されない端末は、参加要求の送信の要求をしない。

<sup>5</sup> JT-H245 アドレスは、呼のセットアップ時間を短縮するために呼出や呼設定受付メッセージにより送られてもよいことに注意すること。JT-H245 チャンネルは、呼設定メッセージにて JT-H245 アドレスの受信後、直ちにオープンされてもよいことに注意すること。

<sup>6</sup> LAN 側の会議は、会議の MC 制御により決められるように、一部集中で、一部非集中でもよい。しかし、端末はこの事実を認識していない。通常、当然だが、端末は同じ SCM (Selected Communication Mode) を参照する。 [標準 JT-H243 参照]

ンスポート PDU にて送信してよい。ただし、全てのメッセージを1つにまとめた長さは最大 PDU 長以下でなくてはならない。すなわち、トランスポート PDU を使用して JT-H245 メッセージを分割してはいけない（付録IVに IP における実装のアウトラインを示す。この PDU は TPKT により規定される）。

JT-H225.0 端末は、低信頼性チャンネルを介した、オーディオおよびビデオの送信に RTP を使用することで、遅延を最小化しなければならない。エラーの隠蔽や、他の回復動作は、パケット損失を克服するために適応されてもよい。一般的に、パケットベースネットワークの環境<sup>7</sup>では過度の遅延の結果となるので、オーディオ/ビデオパケットは、再送されない。ビットエラーが下位レイヤで検出され、エラーパケットが JT-H225.0 まで送られないことが考えられる。オーディオ/ビデオと呼制御/JT-H245 制御は、決して同じチャンネルでは送られない。また、同じメッセージ構造も共有していない。JT-H225.0 端末は、個別の RTP インスタンスをオーディオとビデオに用い、それぞれ異なるトランスポートアドレスにて送信および受信する能力を持たなくてはならない。これにより、メディア特有のシーケンス番号を持たせたり、各々のメディアに対応するサービス品質 (QoS) の処理を適応することが可能になる。しかしながら、オーディオとビデオを同一のフレームに多重化し、1つのトランスポートアドレスへ伝送するオプションモードは今後の課題である。

T.120 の能力は JT-H245 を用いてネゴシエーションされる。適切なメッセージを受信した段階で、T.120 会議は、パケットベースネットワーク上のトランスポート層を使用した適切な JT-T123 スタックにより確立される。T.120 は、パケットベースネットワークにおける、異なるトランスポートアドレスのエンドポイント間にて運ばなければならない。表 1/JT-H225.0 は、ポイント・ポイント呼の各メディアに使用される TSAP 識別子の番号を示す。JT-H323 端末は、TSAP 識別子の追加使用により、一度に1つ以上の会議に参加することが可能となることもある。全ての JT-H245 論理チャンネルは、双方向である T.120 の接続を省いては、片方向である。

表 1/JT-H225.0 JT-H225.0 ポイント・ポイント ユニキャスト呼毎 TSAP 識別子 (ITU-T H.225.0)

使用される TSAP 識別子	高信頼性 or 低信頼性	公知 (Well Known) or 動的 (Dynamic)
オーディオ/RTP	低信頼性	動的
オーディオ/RTCP	低信頼性	動的
ビデオ/RTP	低信頼性	動的
ビデオ/RTCP	低信頼性	動的
呼制御	高信頼性	公知 または 動的
JT-H245	高信頼性	動的
データ (T.120)	高信頼性	公知 または 動的
RAS	低信頼性	公知 または 動的

注：既存の TSAP 識別子を使用する場合は、ネットワークアドレス毎にただ1つのエンドポイントが可能となる。また、直接呼モデルでは、発信者が発呼のための呼制御チャンネルとして TSAP 識別子を要求する。

オーディオとビデオのトランスポートアドレスとして、パケットベースネットワークにおける同一のネットワークアドレスを共有し、TSAP識別子だけが異なることがあるかもしれないが、製造業者によっては、オーディオとビデオにパケットベースネットワークの異なるネットワークアドレスを使用することを選択するかもしれない。唯一の要求は、RTPセッション<sup>8</sup>におけるTSAP識別子の番号付けが付属資料AとBの規定に従わなければならないということである。

表 1/JT-H225.0 は、2 端末間ポイント・ポイント接続のユニキャスト通信手順の基本ケースを示す。ゲートウェイ、MCU、ゲートキーパーの構造を簡易にするために、公知 (well known) の TSAP 識別子の代わりに、動的な TSAP 識別子を使用してもよい。表 2/JT-H225.0 は、ゲートウェイ/MCU の例、表 3/JT-H225.0 はゲートキーパーの例である。

表 2/JT-H225.0 1つの MCU/ゲートウェイポート上で使用される TSAP 識別子

<sup>7</sup> フル・フレームの Fast Update、MB、GOB は、JT-H245 シグナリングにて要求されてもよい。

<sup>8</sup> RTP の初期化セッションで任意の TSAP 識別子を利用することが可能であることに注意すること。RTP の規定に従う主要な理由は、IETF RTP の相互接続性を可能とするためである。

(ユニキャストの例)  
(ITU-T H.225.0)

使用される TSAP 識別子	高信頼性 or 低信頼性	公知 (Well Known) or 動的 (Dynamic)
オーディオ/RTP	低信頼性	動的
オーディオ/RTCP	低信頼性	動的
ビデオ/RTP	低信頼性	動的
ビデオ/RTCP	低信頼性	動的
呼制御	高信頼性	動的 (注 1)
JT-H245	高信頼性	動的
データ(T.120)	高信頼性	動的
RAS	低信頼性	動的 (注 1)
注 1 : 表 3/JT-H225.0 の注 1 参照		

表 3/JT-H225.0 ポイント・ポイント通信の図 28/JT-H323 に示されたゲートキーパー経由型呼モデルをサポートする JT-H225.0 ゲートキーパーにより使用される TSAP 識別子の例  
(ITU-T H.225.0)

使用される TSAP 識別子	高信頼性 or 低信頼性	公知 (Well Known) or 動的 (Dynamic)	チャンネルの数
呼制御	高信頼性	動的 または 公知 (注 1)	呼毎に 2 (注 2)
JT-H245	高信頼性	動的	呼毎に 2 (注 2)
RAS	低信頼性	公知	1
注 1 : 公知の TSAP 識別子が使用されると、ゲートキーパーは機器毎の単一エンドポイントに制限される。それゆえ、動的 TSAP が使用されるべきである。 注 2 : 0 は直接呼モデル、2 はゲートキーパー経由型モデル。			

端末間の呼設定のケースおよびゲートキーパー経由型呼設定のケースに対して、高信頼性トランスポートアドレスが使用される。高信頼性呼制御のコネクションは、呼制御チャンネル上でシグナリングされたすべての有効状態の呼に対する解放完了メッセージが受信されるまで、有効状態に保持しなければならない。

ある時点で、1 つ以上の JT-H245 チャンネルがオープンされていることが有り得ることに留意すること。これは、エンドポイントが一度に 2 つ以上の呼/会議に同時参加する時に起こる。さらに、特定の呼において、端末は同じ種別の 2 つ以上のチャンネルをオープンするかもしれないことに留意すること。これは例えば、ステレオのために 2 オーディオチャンネルをオープンするときに起きる。唯一の制限は、ポイント・ポイント呼毎に JT-H245 制御チャンネルは、唯一存在しなければならないことである。

JT-H245 論理チャンネルシグナリングは、ビデオ、オーディオ、データプロトコルの開始および停止のために使用する。このプロセスは、開いているチャンネルを閉じるために呼ばれ、新しいモードで動作させるために再度開かれる。チャンネルを開くプロセスの一部として、論理チャンネル開設のアクノレッジの前に、エンドポイントは新しいチャンネルに対して十分な帯域幅が確保できることを確実にするために、ARQ/ACF または BRQ/BCF シーケンスを使用する (十分な帯域幅が事前の ARQ/ACF または BRQ/BCF シーケンスによって確保されない限り)。いくつかのケースでは、ゲートウェイは、SCN 側のモード変更が、パケットベースネットワーク側のモード変更よりも早く発生し、その結果オーディオ情報が失われることを見つけることがある。ゲートウェイは製造者の裁量において、いくつかのアプローチを採用してもよい。

ゲートウェイは、SCN モード変更を隠すために、オーディオの符号化変換をしてもよい。  
ゲートウェイは、単にオーディオ情報を捨ててもよい。  
ゲートウェイは、JT-H231 MCU として動作してもよく、全ての SCN 側のモード変更の制御を受ける。

JT-H245 および RTP の手順 (付属資料 A,B,C 参照) のどちらが優位性を持つかについての、一般的なルールは存在しない。本標準において、各々の不一致と解決法が明確に述べられる。

SSRC と論理チャネル間には固定的な関連はないことに留意すること。オーディオとビデオの同期のために使用される SSRC と論理チャネル間の関連は、JT-H245 が提供する。

一般的に、パケットベースネットワーク側の会議モードのオペレーションは、集中と非集中の 2 種類可能である。メディア毎に異なる選択が可能である。例えば、オーディオ/ビデオが非集中型で、データが集中型となることなどが考えられる。どのような種類の会議を確立するかの決定手順は、JT-H323 に示される。本標準のメッセージは、全ての許される組み合わせをサポートすることが意図される。なお、非集中型の制御とデータ通信については、JT-H245 能力シグナリングによってサポートされているものの、今後の検討課題となっている。

## 6.2 RTP/RTCPの使用

JT-H225.0 エンドポイントは、付属資料 A および B に記載するように、オーディオ、ビデオとそれぞれに関連付けた RTCP チャネルに対して個別の TSAP 識別子を使用する能力がなければならない。オプションとして、オーディオおよびビデオに関して異なったパケットベースネットワークアドレスの使用を選択してもよいが、TSAP 識別子の使用において、付属資料 A と B の規約に従うべきである。端末がこの機能をサポートする場合、JT-H245 シグナリングを使って追加のオーディオないしビデオチャネルを開設してもよい。

オーディオおよびビデオの双方において単一のトランスポートアドレスを使用するオプションについては、今後の検討課題とする。

本標準に例外が明確に記載されない限り、実装は、本標準の本文で修正されている場合を除き、付属資料 A に記載する RTP に従わなければならない。実装は、本標準に明確に記載している RTP プロファイル（付属資料 B を参照）のみに従わなければならない。

RTP トランスレータおよびミキサは、JT-H323 システムの要素ではない。また、これらに関する付属資料 A と B での情報は参照 (informative) として扱わなければならない。ゲートウェイと MCU の両方には幾分、ミキサとトランスレータの面があり、付属資料 A と B の情報は、ゲートウェイおよび MCU の実装に役に立つことに留意されたい。しかし、MCU はミキサではないし、ミキサは MCU ではない。ゲートウェイは例えば、ゲートウェイ経由のパケットベースネットワーク対パケットベースネットワーク呼上で、トランスレータとして動作してもよい。

**Version(V) :** RTP のバージョン 2 を使用しなければならない。

**CSRC Count(CC) :** 本標準における CSRC カウントの使用はオプションである。使用されない場合、CC 値はゼロ(0)としなければならない。MCU は、非集中型オーディオ処理が発生している時に、オーディオの発信者に関する情報を提供するために CSRC を使用してもよい。CSRC カウントを認識することができるかどうかに関する能力の表現がないので、MCU/MC は、端末が会議中に、この情報を利用してはいるのか、また、どのように利用しているのかは知るべきでない。

**CNAME :** パケットベースネットワークのポイント・ポイント接続の最も単純な場合では、SSRC は、端末からのオーディオ/ビデオの発生源を識別するために使用される。また、付属資料 A で明記するように、オーディオビデオなどのストリームは CNAME を用いることで同じエンドポイントから供給されたと関連付けされる。

RTCP を使用する時は、付属資料 A に記載するように、RR または SR パケットのどちらかが、定期的に送信されなければならない。CNAME SDDES メッセージを使用しなければならない。その他の SDDES メッセージ（付属資料 A 参照）はオプションである。しかし、JT-H245 および/または T.120 のいずれかの制御機能が使用されている時は、このメッセージを会議制御または会議情報に使用してはならない。JT-H245 および/または T.120 によって提供される情報が正しい情報と見なされなければならない。

RTP セッションを終了するのに、RTCP BYE メッセージに依存してはならない。JT-H323 端末は、JT-H323 手順により呼が何時切断されるのかを決定する。RTCP BYE パケットの使用が必須なのは、SSRC 競合解消の時のみである。JT-H323 端末は、全ての会議に携わる場合、ポイント・ポイント会議かマルチポイント会議かに関わらず、JT-H245 で定義される期間にわたって平均化された論理チャネルのビットレートを JT-H245 **FlowControlCommands** や、JT-H245 論理チャネルコマンドおよび T.120 フロー制御メカニズムによりシグナリングされたレートに制限しなければならない。

JT-H323 端末が、JT-H323 ゲートウェイに接続される場合、ゲートウェイは、JT-H323 端末に SCN 側のメディアレート以下で送信し、SCN レート以上で受信させるために、JT-H245 および T.120 の手法を使用しなければならない。ただし、以下の例外を除く。

- パケットベースネットワークの制御帯域幅は、JT-H221 における帯域幅に適合する必要はない。

- パケットベースネットワークのオーディオ帯域幅は、SCN 上で JT-H221 における帯域幅に適合してもよい。しかし、ゲートウェイでトランスコーディングを行う場合は、適合は要求されない。
- ゲートウェイが減速器を使用している場合は、パケットベースネットワーク側の JT-H323 端末は、JT-H245 のシグナルレートを超えてはならない。この速度は、おそらく SCN 上で送られる速度よりも低い。

JT-H323 エンドポイントの暗号化については、今後の検討課題とする。

## 6.2.1 オーディオ

RTP を使用して、どのようにオーディオがパケット化されるのかを考える前に、JT-H245 を経由したオーディオシグナリングの方法および、このシグナリングと RTP の関係について、注意しなければならない。一般的に、オーディオチャネルが開かれる場合、JT-H245 論理チャネルが開かれる。**AudioCapability** 構造における JT-H245 シグナリングは、パケットあたりの最大フレーム数で与えられる。本標準のフレームサイズは、使用されるオーディオ符号化に従って変化する。

オーディオ通信を提供する JT-H323 端末は、全て G.711 をサポートしなければならない。受信側は、フレーム方式のオーディオコーデック全てに対し、単一のオーディオパケットで受け付けることのできるオーディオフレームの最大数を通知しなければならない。送信側は、受信側が指示する最大限度まで、各パケットで任意の数のオーディオフレームを送信することができる。送信側は、複数のパケットにまたがってオーディオフレームを分割してはならず、各オーディオパケットで全ての数のオクテットを送信しなければならない。

G.711 や JT-G722 のようなサンプルを使用するコーデックは、フレームサイズが 8 サンプルのフレーム方式と見なされなければならない（サンプルに基づくオーディオ符号化の指針についての追加情報は、付属資料 B を参照）。複数のサイズのオーディオフレームを使用する JT-G723.1 のようなオーディオのアルゴリズムについては、各パケット内のオーディオフレーム境界はオーディオチャネルに対しインバンドの信号で通知されなければならない。

固定のフレームサイズを使用するオーディオアルゴリズムについては（各アルゴリズムによって使用されるフレームサイズについては、JT-G728 および JT-G729 参照）、オーディオフレーム境界は、オーディオフレームサイズに対するパケットサイズの比率によって暗黙のうちに示されなければならない。言い換えれば、オーディオフレームは分割されてはならず、それ全体として RTP パケットに収められなければならない。

**Payload Type(PT) :** JT-H245 中で通知される ITU コーデックについては、(0)[PCMU]、(8)[PCMA]、(9)[JT-G722]および(15)[JT-G728]のような ITU-T ペイロードタイプのみが使用されなければならない。付属資料 B に掲げられていない ITU-T ペイロードタイプについては、JT-H245 シグナリングを使用して交換されるダイナミックペイロードタイプを使用しなければならない。

シーケンス番号の中で中断が見られる場合、直前に受信したサウンドを受信側が反復しつつ、サウンドの音量をゼロに下げてもよいことを推奨する。製造業者の自由裁量により、同様のその他の手順を採用することもできる。

各 G.711 オクテットは、RTP パケットの中に並べられたオクテットでなければならない。各 G.711 オクテットのサインビットは、RTP パケット内のオクテットの最上位ビットと一致しなければならない（すなわち、ホストマシン上で G.711 サンプルがオクテットとして扱われ、サインビットはホストマシンのフォーマットによって定義されたオクテットの最上位ビットでなければならない、と推測される）。

JT-H323 ゲートウェイは、48/56kbit/sPCM のパケットベースネットワークへの送信に際し、G.711 の表 1b の注 2 の通りに各オクテットに余分の 1 ビットか 2 ビットをパッドし、PCMA または PCMU（8 または 0）の RTP 値を使用しなければならない。μ 則については、パディングは、7 番目と 8 番目のビットの双方で 1 とする。A 則については、7 番目のビットを 0、8 番目のビットを 1 としなければならない。逆の方向においては、JT-H323 ゲートウェイは、JT-H320 で使用中の G.711 速度に適合するため、パケットベースネットワーク側で 64kbit/s G.711 に切り捨てを行わなければならない。従って、パケットベースネットワーク側では 64kbit/s G.711 のみが使用されなければならない。

A 則については、7 番目のビットを 0、8 番目のビットを 1 としなければならない。逆の方向においては、JT-H323 ゲートウェイは、JT-H320 で使用中の G.711 速度に適合するため、パケットベースネットワーク側で 64kbit/s G.711 に切り捨てを行わなければならない。従って、パケットベースネットワーク側では 64kbit/s G.711 のみが使用されなければならない。

JT-H323 ゲートウェイは、パケットベースネットワークに 48/56kbit/s JT-G722 を送信する場合、各オクテ

ットに余分の 1 ビットか 2 ビットをパッドし、64kbps (PT=9 を使用する) と減速ケースとを区別するため JT-H245 によって通知されるダイナミック RTP ペイロードタイプを使用しなければならない。逆の方向においては、JT-H323 ゲートウェイは、JT-H320 で使用中の G.711 速度に適合するため、パケットベースネットワーク側で 64kbit/s JT-G722 に切り捨てを行わなければならない。従って、パケットベースネットワーク側では 64kbit/s JT-G722 のみを使用しなければならない。

JT-H323 端末は、特に多地点会議の場合には、可能であれば RTP の無音圧縮機能を使用することが望ましい。JT-H323 端末は、無音圧縮 RTP ストリームを受信することができなければならない。コーデックは、無音の単一フレームを送信した後、無音の期間、オーディオ信号の送信を省略するか、あるいは、使用中の符号化則に無音時の背景オーディオフレーム充填技術が規定されている場合、このフレームを送信することができる。

## 6.2.2 ビデオメッセージ

**Payload Type (PT) :** JT-H245 内で通知されるコーデックには、JT-H261 または JT-H263 のペイロードのよ  
うな ITU-T ペイロードタイプのみが使用されなければならない。JT-H245 を介して通知することは  
可能だが、パケット形式が定義されていないコーデックには、ダイナミックペイロードタイプを使  
用してもよい。

**Maker(M) :** マーカービットがエンド・エンドの遅延を増大させるような場合を除き、付属資料 A で述べ  
ている手順に従ってマーカービットを設定するべきである。

ビデオパケットの損失から回復するため、JT-H245 の **VideoFastUpdatePicture**、**VideoFastUpdateMB** およ  
び **VideoFastUpdateGOB** がサポートされなければならない。RTCP 制御パケットフルイントラ要求 (FIR)  
[送信フレームの送信要求]、および否定応答 (NACK) [確実なパケット送信要求] の使用はオプション  
で、JT-H245 能力で通知される。

C.3.3 節の中の誤り回復方法 3)は、NACK が 1 フレーム時間内に到着しない場合、実用的ではないことが  
ある。

JT-H261 は、付属資料 C に従ってパケットベースネットワーク側でパケット化される。十分な大きさの RTP  
パケットが利用できる限り、送信側による MB 境界での細分化は必要としない。しかし、JT-H323 端末が RTP  
レベルで JT-H261 パケットの細分化を行う場合、この細分化は、MB 境界で発生しなければならない。全ての  
JT-H323 端末は、MB 細分化されたパケットのほか、GOB 細分化されたパケットまたは MB と GOB の混  
合のパケットを受信できなければならない。送信側での MB 細分化に失敗すると、GOB 全体の喪失が起  
こり、また、パケット速度が低下することのある点に留意されたい。制御の確実性を最大限度に維持するため、  
使用される RTP パケットは、所定のパケットベースネットワーク上での最大転送ユニット (MTU) のサイ  
ズを超えないこととするが、コード化体系 (例えば、マクロブロック) の最小独立コード単位が MTU サイ  
ズよりも大きい場合、MTU 上でのパケット分解は必要とされない。MB は、パケットにわたって分割されて  
はならない。全てのパケットは、GOB または MB 境界で終了しなければならない。JT-H323 送信側は、小さ  
な GOB を含むパケットを追加の MB で充填することもできるが、これは要求されているわけではない。

JT-H323 エンドポイントの RTP パケット化器は、RTP パケットの喪失による複数画像の崩壊が生じないよ  
うにするため、1 つの RTP パケットに複数の画像からなるビデオを含めてはならない。

SBIT は最初のデータオクテットの中で、無視されなければならない最上位ビットの数である。EBIT は最  
後のデータオクテットの中で、無視されなければならない最下位ビットの数である。

RTP パケット化器は、RTP パケットの先頭で意図的にビデオのオクテット整合を行ってはならない。言い  
換えれば、RTP パケットにおいて EBIT=n の場合、次の RTP パケットにおける SBIT は、8-n、 $0 < n < 8$  に等し  
くなければならない。また、RTP パケットにおいて EBIT=0 の場合、次の RTP パケットにおける SBIT は、0  
に等しくなければならない。この要求条件は、ビットシフティングによりエンド・エンドの遅延がさらに追  
加されることを防ぐ。この要求条件は、画像境界を越えて適用されなければならない。

付属資料 D は、オプションのオクテットカウントを含むビデオパケットヘッダに対する JT-H323 の拡張を  
規定したものである。このオプションの拡張の使用法については、付属資料 D に記述されている。

パケットベースネットワークにおけるビデオのパケット化に関する具体的な助言については、付録 IV を参  
照のこと。

## 6.2.3 データメッセージ

特別なデータメッセージやフォーマットは無い。JT-T123 により、パケットベースネットワークでは、T.120

が使われる。パケットベースネットワークでの、集中型対非集中型データ会議は、JT-H323 で記述され、JT-H245 経由でネゴシエーションされる。

パケットベースネットワークでの T.120 フロー制御は、JT-H245 **FlowControlCommand** または **maxBitRate** による制限が要求された場合、パケットベースネットワークプロトコルの使用により管理される。

会議中の T.120 会議に、JT-H323 会議を接続する場合、または T.120 会議に JT-H323 呼を加える場合に使用する手順については、JT-H323 を参照のこと。

パケットベースネットワークで、JT-H224 が使用するプロトコルは、今後の検討課題である。

## 7 JT-H225.0 メッセージ定義

本章は、呼設定、呼制御および端末、ゲートウェイ、ゲートキーパー、MCU 間の通信に必要なメッセージ定義に関するものである。

全ての JT-H225.0 メッセージに必要な ASN.1 定義は、付属資料 H に記す。

### 7.1 JT-Q931 メッセージの使用

実装は、本標準に規定する JT-Q931 に準拠しなければならない。また、端末は、オプションの JT-H450 ユーザユーザ情報要素の APDU をサポートしてもよい。メッセージは、JT-H225.0 で規定した JT-Q931 で定義される全ての必須情報要素を含まなければならない。また、本標準で示す JT-Q931 で定義されるオプションの情報要素を含むことができる。JT-H225.0 エンドポイントは、JT-Q931 に従って、サポートしなくても相互接続性を損うことのない全てのオプションのメッセージを無視することができるが、未知のメッセージに対しては状態表示メッセージで応答しなければならない点に注意すること。

各 JT-H225.0 エンドポイントは、ユーザユーザ情報要素中に JT-H450 APDU を持つものを含む着信 JT-H225.0 呼制御メッセージを受信し、認識できなければならない。JT-H225.0 エンドポイントは、必須の JT-H225.0 呼制御メッセージを処理できる能力がなければならない。オプションの JT-H225.0 呼制御メッセージを処理する能力があってもよい。いずれにせよ、各 JT-H225.0 エンドポイントは動作を妨げることなく、未知のメッセージを無視できなければならない。

各 JT-H225.0 エンドポイントは、JT-H.225.0 呼制御メッセージ、ユーザユーザ情報要素中の JT-H450 APDU それぞれについて、以下の必須情報要素を解釈し、生成できなければならない。同様に以下に挙げるオプションの情報要素を解釈し、生成してもよい。また、JT-Q931 や、その他 Q シリーズまたは JT-H450 プロトコルのその他の情報要素を解釈してもよい。エンドポイントは動作を妨げることなく、JT-H225.0 呼制御メッセージあるいは JT-H450 APDU に含まれる未知の情報要素を無視できなければならない。理解不能の「理解が要求される」情報要素を受信した時の手順は、5.8.7.1 節/JT-Q931 に従って適用されなければならない。JT-H225.0 エンドポイントは、同一メッセージ内で同一タイプの複数の情報要素を送信してはならない。例えば、JT-Q951 付属資料 A に記述されているような複数の発番号情報要素を送信してはならない。

情報要素は、本標準で修正された個所を除き、JT-Q931 に従ってコード化されなくてはならない。しかし、JT-Q931 は、本標準の中でリストされた要素の順序に関係なく、メッセージの中で情報要素の適切な順序を常に指示しなくてはならない。

中間システム（ゲートウェイおよびゲートキーパー）は、JT-H225.0 呼制御オプションのメッセージおよび情報要素に関し、以下の規則に従わなければならない。

- 1) 以下は、ゲートウェイでは要望、ゲートキーパーでは必須である。中間システムは、必須 JT-H225.0 呼制御メッセージに伴う全ての情報要素（オプションまたは必須）を適当な修正の後、端末からゲートウェイ/端末方向または逆方向へ転送する。これには、ユーザ・ユーザ情報や、表示情報の様な情報要素が含まれる。
- 2) ゲートウェイは、JT-H450 APDU を持つものを含む全ての JT-H225.0 呼制御メッセージと、情報要素を双方向に転送するべきである。
- 3) ゲートキーパーは適当な修正の後に、JT-H450 APDU を持つものを含む全ての JT-H225.0 呼制御メッセージおよび情報要素を双方向に転送しなければならない。ゲートキーパーは特色（付加サービスのような）を供給できるシグナリング要素として動作してもよい。それゆえ、JT-H225.0 呼制御メッセージを修正、終端、生成してもよい。

JT-H323 ゲートウェイは、JT-H450 シリーズ付加サービスや JT-H225.0 メッセージを、対応する ISO/IEC 11582、ISUP、他の SCN シグナリング標準の付加サービスやメッセージに変換する能力があってもよい。詳細は、JT-H246 と付属資料の範囲にある。

JT-H323 ゲートウェイは、JT-H225.0 での非 JT-H323 シグナリングのトンネリングを使用して、未修正の



ISO/IEC 11582 や ISUP や他の SCN シグナリング標準のシグナリングメッセージを通す能力があってもよい。詳細は、JT-H323 の付属資料 M (M.1、M.2、その他) にある。

本標準の本版では、全ての参照は JT-Q931(1998)に対するものとする。回線モード接続設定に関する 3.1 節/JT-Q931 の手順は次に述べる。しかし、実装者は、「ベアラ」がシグナリングされているが、実際にはパケットベースネットワーク側で ISDN タイプの「B チャネル」が存在しない点に注意すること。「呼」が正常に張られると、JT-H245 メッセージ交換をサポートするエンド・エンドの高信頼性チャネルが開かれる。実際に「ベアラ」設定は JT-H245 を使用して行われる。しかし、パケットベースネットワークでの JT-Q931 の使用は、一般的なコネクション型の呼というよくテストされた枠組みだけでなく、SCN 側における JT-Q931 の相互運用を可能にする。

一般的に、JT-Q931 付属資料 D の対称手順が用いられる。JT-Q931 状態遷移が、D.3 (呼の衝突) 手順に従ってはならない点を除いて、JT-Q931 付属資料 D に従う。また、このグレア状態からの回復は、アプリケーション層に委ねられる。

JT-Q931 シフトコードセットをサポートしないエンドポイントは、このような方式を使用した JT-Q931 メッセージは全て無視しなければならない。

表 4/JT-H225.0 に、パケットベースネットワークでの JT-Q931 を使用した JT-H323 と JT-H225.0 呼設定について何が必須で、何がオプションかを示す。

表 4/JT-H225.0 JT-Q931/2 メッセージの JT-H225.0 での使用  
(ITU-T H.225.0)

	送信 (M,F,O,CM) <sup>a)</sup>	受信および動作 (M,F,O <sup>b)</sup> ,CM)
呼設定メッセージ		
呼出 (Alerting)	M	M
呼設定受付 (Call Proceeding)	O	CM <sup>e)f)</sup>
応答 (Connect)	M	M
応答確認 (Connect Acknowledge)	F	F
経過表示 (Progress)	O	CM <sup>f)</sup>
呼設定 (Setup)	M	M
呼設定確認 (Setup Acknowledge)	O	O
呼終結メッセージ		
切断 (Disconnect)	F	F
解放 (Release)	F	F
解放完了 (Release Complete)	M <sup>d)</sup>	M
呼情報フェーズメッセージ		
再開 (Resume)	F	F
再開確認 (Resume Acknowledge)	F	F
再開拒否 (Resume Reject)	F	F
中断 (Suspend)	F	F
中断確認 (Suspend Acknowledge)	F	F
中断拒否 (Suspend Reject)	F	F
ユーザ情報 (User Information)	O	O
その他メッセージ		
輻輳制御 (Congestion Control)	F	F
付加情報 (Information)	O	CM <sup>f)</sup>
通知 (Notify)	O	O
状態表示 (Status)	M <sup>e)</sup>	M
状態問合せ (Status Inquiry)	O	M
JT-Q932/JT-H450 メッセージ		
ファシリティ (Facility)	M	M
保留 (Hold)	F	F
保留確認 (Hold Acknowledge)	F	F
保留拒否 (Hold Reject)	F	F

保留解除 (Retrieve)	F	F
保留解除確認 (Retrieve Acknowledge)	F	F
保留解除拒否 (Retrieve Reject)	F	F

- a) M は必須 (Mandatory)、F は禁止 (Forbidden)、O はオプション (Optional)、CM は条件付き必須 (Conditional Mandatory) である。一度オプションがサポートされると要求されるのが CM である。
- b) 状態表示(Staus)は、ここで「O」とリストされたメッセージへの応答として送られてはならず、受信側は、これをサポートしない場合、単純にこのメッセージを無視しなければならないことに注意すること。
- c) ゲートウェイを使用するつもり端末は、呼設定受付 (Call Proceeding) メッセージを受信し、動作しなければならない。
- d) 解放完了メッセージは、JT-H225.0 高信頼性呼制御チャネルが開かれている状況ではいつでも必要とされる。このチャネルが開かれていない場合、JT-H245 セッション終了が、会議終了に使用されてもよい。
- e) エンドポイントは、未知のメッセージに対して状態表示(Staus)メッセージで応答しなければならない。状態問合せ(Status Inquiry)メッセージに対する応答も必須である。しかし、エンドポイントは状態問合せ (Status Inquiry)を送信することは必要とされない。実際問題として、受信側にとって未知のメッセージに対する応答の状態表示(Status)メッセージについては、エンドポイントが理解するべきである。
- f) これらのメッセージ (JT-H245 トンネリング、JT-H450 付加サービス、シグナリングプロトコルのトンネリング、または、**genericData** を使用する機能) を使用するオプション機能を提供するエンドポイントは、これらのメッセージを処理しなければならない。

## 7.2 共通のJT-Q931 情報要素

### 7.2.1 ヘッダ情報要素

すべての JT-H225.0 呼制御メッセージには、このセクションで記述されるメッセージ種別のほかに必須である 3 つの共通フィールドがある。

#### 7.2.1.1 プロトコル識別子

4.2 節/JT-Q931 で定義される通り。

08H に設定しなければならない。— これにより、(図 4-2/JT-Q931 に従ってコード化された) JT-Q931 /JT-I451 のユーザーネットワークメッセージとして識別される。もしゲートキーパーが付加サービスを提供するネットワークとして動作しているなら、別の値を使うことが適当であるかもしれない。これは今後の検討課題とする。

#### 7.2.1.2 呼番号

4.3 節/JT-Q931 で定義される通り。

2 オクテット長の呼番号値は、どんな JT-H323 エンポイントでもサポートされなければならない。

呼番号値は発信側で選択され、ローカルではユニークでなければならない。引き続き行われる通信では、発信側と着信側はこの特定の呼に属している全てのメッセージでこの呼番号値を使用しなければならない。

値は、2 オクテット呼番号値用の図 4-5/JT-Q931 に従ってコード化される。番号値の最上位オクテットは、常に第 2 オクテット目にコード化される。

(TTC 注: オクテット番号 1 は、オクテット長に使われるので、番号値そのものはオクテット番号 2 以降にコード化される)

呼番号は 1 オクテット (長) と 2 オクテット (長) の場合があるが、JT-H225.0 では、必ず 2 オクテット (長) を採用し、オクテット 2 を上位として扱うことを述べている。

2 バイトで表現されるため、オクテット 2、オクテット 3 のどちらが上位として扱うかを明確にするのが、この規定の意図。

#### (1)1 オクテットの呼番号の coding

```

-----
予 備          呼番号長
0 0 0 0 | 0 0 0 1   オクテット 1
-----

```

```

flag          呼番号値
0/1 0 0 0 0 0 0   オクテット 2
-----

```

## (2)2 オクテットの呼番号の coding

予 備	呼番号長	
0 0 0 0	0 0 0 1	オクテット 1
flag	呼番号値	
0/1 0 0 0	0 0 0 0	オクテット 2 <- こちらが上位
0 0 0 0	0 0 0 0	オクテット 3

T T C注 終わり)

CRV は、例えば 2 端末間、あるいは端末とゲートキーパー間といった呼の特定部分でのみユニークであることに注意すること。もし既定の端末が同じ会議で 2 つの呼を持っているなら、それぞれの呼は同じ会議 ID で異なった CRV を持たなければならない。

呼番号フラグは、JT-Q931 で記述される手順に従って設定されなければならない。

RAS メッセージで受け渡される CRV 値は、JT-Q931 で指定される構造に従わなければならないことに注意すること。特に、呼番号フラグは呼番号値の最上位ビットとして含まれなければならない。これは、実際の CRV を 0 から 32767 までの範囲に制限する。

図 4-5/JT-Q931 で示される数値 0 を持つグローバル呼番号は、呼制御チャンネルあるいは RAS チャンネルの上の全ての呼を参照するために使われる。

### 7.2.1.3 メッセージ種別

メッセージ種別は、表 4-2/JT-Q931 で指定された値を使って、図 4-6/JT-Q931 に従ってコード化される。JT-H225.0 固有の拡張は、今後の検討である。

### 7.2.2 メッセージ固有の情報要素

次の情報要素のための一般的な符号化規則は、4.5.1 節/JT-Q931 と表 4-3/JT-Q931 で定義される。これらの規則に従わなければならない。エスケープ機構 (図 4-8/JT-Q931 参照) はオプションである。

#### 7.2.2.1 伝達能力

この情報要素は、図 4-11/JT-Q931 と表 4-6/JT-Q931 に従ってコード化される。もしこの情報要素がパケットネットワークからパケットベースネットワークへの呼で受信されたら、受信側によって無視されるかもしれない。もしこの情報要素が、JT-H450.1 で定義される呼に依存しないシグナリング接続のための呼設定メッセージに現れるなら、コーディングは 7.2.2.1.2 節に従わなければならない。全ての他の場合では、コード化は 7.2.2.1.1 節に従わなければならない。オクテット番号照合は図 4-11/JT-Q931 を参照する。

##### 7.2.2.1.1 伝達能力デフォルトコード化

次節で他のことが示されないなら、JT-H323 エンティティが次のように伝達能力情報要素をコード化しなければならない。

#### オクテット No.3 の拡張ビット (ビット 8)

「1」に設定されなければならない。

#### コーディング標準 (オクテット No.3、ビット 6-7)

「ITU-T」を示している「00」に設定されなければならない。

#### 情報転送能力 (オクテット No.3、ビット 1-5)

ISDN エンドポイントから発信される呼では、ゲートウェイに示された情報は転送されなければならない。

注—これは JT-H323 エンドポイントに転送される接続の特性について、いくつかの拡張情報のためである。例えば、オーディオのみ対データ対ビデオ；これは、呼を受け入れるかの能力/意志に影響を与えるのと同様に要求される帯域幅に影響を与える。

JT-H323 エンドポイントから発信される呼は、オーディオビジュアル呼を発信する意向を示すためにこのフィールドを使用しなければならない。従って、フィールドは「非制限デジタル情報」すなわち「01000」、あるいは「制限デジタル情報」すなわち「01001」のどちらかに設定されなければならない。もしオーディオだけの呼が発信されたら、JT-H323 端末は情報転送能力を「音声」(すなわち「00000」)、あるいは「3.1KHz オーディオ」(すなわち「10000」)に設定しなければならない。

#### オクテット No.4 の拡張ビット (ビット 8)

もし情報転送速度が「マルチレート」に設定されるなら「0」に、そうでなければ「1」に設定されなければならない。

#### 転送モード (オクテット No.4、ビット 6、7)

「回線交換モード」を示す値、「00」を指定しなければならない。

#### 情報転送速度 (オクテット No.4、ビット 1-5)

ゲートウェイがパケットネットワークに接続されていない限り、値「00000」が許されない以外は、表 4-6/JT-Q931 に従ってコード化されなければならない。

#### レートマルチプライヤ (オクテット No.4.1)

もし情報転送速度が「マルチレート」に設定されるなら、設定されなければならない。

拡張ビット (ビット 8) は「1」に設定されなければならない。

ビット 1 から 7 は、以下に定義されるように、呼に必要な帯域幅を示さなければならない。(注: JT-Q931 とは対照的に「0000001」の値がここでは許される)

ISDN エンドポイントから発信される呼については、ゲートウェイは ISDN から受信する情報を単純に渡さなければならない。

JT-H324 エンドポイントから着信する呼については、ゲートウェイはレートマルチプライヤを 01H に設定しなければならない。

B-ISDN から着信する呼については、JT-Q2931 から JT-Q931 にいくつかの置き換えが行われる必要がある。これは今後の検討課題である。

JT-H323 エンドポイントから発信する呼については、これはこの呼に使われる帯域幅を示すために使用されなければならない。もし着信するシステムがもう 1 つの JT-H323 エンドポイントであるなら、この値はパケットネットワーク上で使われるため帯域幅を反映してもよいが、受信端末はこの情報に従う必要がない。もしゲートウェイが呼に含まれているなら、この値は呼設定する必要がある外部の接続数を反映しなければならない。この呼に必要な帯域幅は、SCN 側で必要とされる帯域幅であり、ACF/BCF メッセージによってパケットネットワーク上で許された帯域幅に適合してもよいし、適合しなくてもよい。

#### レイヤ 1 プロトコル (オクテット No.5)

拡張ビット (ビット 8) は「1」に設定されなければならない。

ビット 6 と 7 はレイヤ 1 識別、すなわち「01」を示さなければならない。

ビット 1 から 5 はレイヤ 1 プロトコルを示さなければならない。

許容値は、オーディオのみの呼を示す G.711 (A 則「00011」と  $\mu$  則「00010」) と、JT-H323 テレビ電話の呼を示す JT-H221 と JT-H242 (「00101」) がある。

オクテット No.5a、5b、5c、5d、6、7 は存在してはならない。

#### 7.2.2.1.2 JT-H450.1 呼独立シグナリング接続のための伝達能力コード化

JT-H450.1 で定義される呼独立シグナリング接続では、JT-H323 エンティティは伝達能力情報要素を次のようにコード化しなければならない。

#### オクテット No.3 の拡張ビット (ビット 8)

「1」に設定されなければならない。

#### コーディング標準 (オクテット No.3、ビット 6-7)

「他の国際標準」を示す「01」に設定されなければならない。このコーディング標準が示される場合、JT-Q931 で定義されたコード化がオクテット 1 から 2 に、オクテット 3 のビット 8 からオクテット 4 に適用されなければならない。情報転送能力、転送モード、情報転送速度は指示された通りにコード化されなければならない。それ以外のオクテットは含まれてはならないことに留意すること。

#### 情報転送能力 (オクテット No.3、ビット 1-5)

「非制限デジタル情報」を示す「01000」に設定されなければならない。

#### オクテット No.4 (ビット 8) のための拡張ビット

「1」に設定されなければならない。

#### 転送モード (オクテット No.4、ビット 6、7)

「呼非独立シグナリング接続」を示す「00」に設定されなければならない。

#### 情報転送速度 (オクテット No.4、ビット 1-5)

「呼独立シグナリング接続」を示す「00000」に設定されなければならない。

オクテット 4.1 以上は含んではならない。

#### 7.2.2.2 呼識別

呼識別情報要素 (IE) の利用法は今後の検討課題である。この検討は、端末-ゲートキーパー-端末と端末-ゲートウェイ-端末を含む多段階ダイヤリングと、ルースソースルーチングを考慮すべきである。

(TTC 注: ルースソースルーチング=IP ソースルーチングの一種で、経由することが必須のルータについてのみ送信元で指定する方式)

#### 7.2.2.3 呼状態

この情報要素は図 4-13/JT-Q931 に従ってコード化される。

### オクテット No.3 コーディング標準 (ビット 87)

ITU-T 標準符号化方式を示す値、「00」に設定すること。

#### 呼状態値 (オクテット No.3、ビット 1-6)

グローバルインタフェース状態値を使用しない以外は、表 4-8/JT-Q931 に従って設定すること。値は付属資料 D/JT-Q931 に従ってユーザ状態と解釈される。リストされたコードの大部分は、JT-H323 端末によって生成されたものではないであろうということに注意しなければならない。

#### 7.2.2.4 着番号

この情報要素は、図 4-14/JT-Q931 と表 4-9/JT-Q931 に従ってコード化される。

#### オクテット No.3 拡張子 (ビット 8)

「1」に設定される。

#### 番号種別 (オクテット No.3、ビット 5-7)

表 4-9/JT-Q931 の値と規則に従ってコード化される。

#### 番号計画識別 (オクテット No.3、ビット 1-4)

表 4-9/JT-Q931 の値と規則に従ってコード化される。ダイヤル番号形式の番号は「00000」(不定)としてコード化すべきである。もしパケットネットワーク発信呼で「1001」(私設網番号計画)に設定されたなら、これは以下のことを示している。

- 1) ダイヤルされたディジット列が呼設定に存在しない、そして
- 2) 呼はユーザ・ユーザ情報における別名アドレスを経由してルーチングされる。

#### 番号種別 (オクテット No.3、ビット 5-7)

表 4-9/JT-Q931 の値と規則に従ってコード化される。番号計画識別が「0000」(不定)としてコード化された番号は「000」(不定)としてコード化されなければならない。番号計画識別番号が「0001」(ISDN/電話番号計画、E.164)であり、番号種別「000」(不定)としてコード化された番号は、下位互換として使用してもよい。

#### 番号“ディジット”

適切なナンバリング/ダイヤリング計画で指定されたフォーマットによる IA5 キャラクタの任意の番号。

注：E.164 番号は IA5 文字「0」、「1」、「2」、「3」、「4」、「5」、「6」、「7」、「8」および「9」からのみ構成されなければならない。

#### 7.2.2.5 着サブアドレス

JT-Q931 に従って使用すること。

#### 7.2.2.6 発番号

この情報要素は、図 4-16/JT-Q931 と表 4-11/JT-Q931 に従ってコード化される。

#### 番号種別 (オクテット No.3、ビット 5-7)

表 4-11/JT-Q931 の値と規則に従ってコード化される。番号計画識別が「0000」(不定)としてコード化された番号は「000」(不定)としてコード化されなければならない。番号計画識別番号が「0001」(ISDN/電話番号計画、E.164)であり、番号種別「000」(不定)としてコード化された番号は、下位互換として使用してもよい。

#### 番号計画識別 (オクテット No.3、ビット 1-4)

表 4-11/JT-Q931 の値と規則に従ってコード化される。ダイヤル番号列形式の番号は「00000」(不定)としてコード化すべきである。もしパケットネットワーク発信呼で「1001」(私設網番号計画)に設定されたなら、これは以下のことを示している。

- 1) ダイヤルされたディジット列が呼設定に存在しない、そして
- 2) 呼はユーザ・ユーザ情報の別名アドレスを経由してルーチングされる。

#### オクテット No.3a

表 4-11/JT-Q931 の値と規則に従ってコード化される。

#### 番号“ディジット”

適切なナンバリング/ダイヤリング計画で指定されたフォーマットによる IA5 キャラクタの任意の番号。

注：E.164 番号は IA5 文字「0」、「1」、「2」、「3」、「4」、「5」、「6」、「7」、「8」および「9」からのみ構成されなければならない。

JT-H323 エンドポイントは、複数の発番号情報要素を同一メッセージに送信してはいけない。ゲートウェ

イは複数の発番号情報要素を含む JT-Q931 呼設定メッセージとの相互接続のサポートを提供してもよい。このようなサポートを提供するゲートウェイは、最初の JT-Q931 発番号情報要素を JT-H225.0 呼設定メッセージの発番号情報要素に割り当て、それ以降の JT-Q931 発番号情報要素を JT-H225.0 呼設定メッセージの **additionalSourceAddresses** フィールドに割り当てなければならない。JT-H323 エンドポイントにより開始された JT-H225.0 呼設定メッセージを転送するゲートキーパーは、次の受信者へ転送する前に、**additionalSourceAddresses** フィールドに番号を挿入してもよい。

#### 7.2.2.7 発サブアドレス

JT-Q931 に従って使用する。

#### 7.2.2.8 理由表示

JT-Q850 で定義された規則を受信した場合に適用される。理由表示あるいは **ReleaseCompleteReason** は解放完了のために必須であり、他の場合においては理由表示情報要素はオプションであることに注意しなければならない。理由表示情報要素と **ReleaseCompleteReason** (解放完了メッセージの一部) は相互に排他的である。パケットネットワーク側から回線交換網側に解放完了メッセージを送信する時、ゲートウェイは **ReleaseCompleteReason** から理由表示情報要素にマッピングしなければならない(表 5/JT-H225.0 参照)。(逆マッピングは、パケットネットワークエンティティが理由表示情報要素をデコード化/復号するよう要求されているため要求されない。)

ゲートウェイは、**AdmissionReject** または **LocationReject** を受信したあとに回線交換側に解放完了メッセージを送るとき、**AdmissionRejectReason** と **LocationRejectReason** を理由表示情報要素にマッピングしなくてはならない。(表 6/JT-H225.0 参照)

表 5/JT-H225.0 **ReleaseCompleteReason** から理由表示情報要素へのマッピング  
(ITU-T H225.0)

ReleaseCompleteReasonコード	対応するJT-Q931/JT-Q850の理由値
noBandwidth	34 - 利用可能な回線/チャンネルなし
gatekeeperResources	47 - リソース使用不可
unreachableDestination	3 - 着信先へのルートなし
destinationRejection	16 - 正常呼切断
invalidRevision	88 - 着信先不一致
noPermission	127 - インタワーキング、特定されていない
unreachableGatekeeper	38 - ネットワーク障害
gatewayResources	42 - 交換機輻輳
badFormatAddress	28 - 無効番号フォーマット
adaptiveBusy	41 - 一時的障害
inConf	17 - ユーザビジー
underfinedReason	31 - 正常、特定されていない
facilityCallDeflection	16 - 正常呼切断
securityDenied	31 - 正常、特定されていない
securityWrongSyncTime	31 - 正常、特定されていない
securityReplay	31 - 正常、特定されていない
securityWrongGeneralID	31 - 正常、特定されていない
securityWrongSendersID	31 - 正常、特定されていない
securityMessageIntegrityFailed	31 - 正常、特定されていない
securityWrongOID	31 - 正常、特定されていない
securityDHmismatch	31 - 正常、特定されていない
securityCertificateExpired	31 - 正常、特定されていない

securityCertificateDateInvalid	31 - 正常、特定されていない
securityCertificateRevoked	31 - 正常、特定されていない
securityCertificateNotReadable	31 - 正常、特定されていない
securityCertificateSignatureInvalid	31 - 正常、特定されていない
securityCertificateMissing	31 - 正常、特定されていない
securityCertificateIncomplete	31 - 正常、特定されていない
securityUnsupportedCertificateAlgOID	31 - 正常、特定されていない
securityUnknownCA	31 - 正常、特定されていない
calledPartyNotRegistered	20 - 加入者不在
callerNotRegistered	31 - 正常、特定されていない
newConnectionNeeded	47 - リソース使用不可
nonStandardReason	127 - インタワーキング、特定されていない
replaceWithConferenceInvite	31 - 正常、特定されていない
genericDataReason	31 - 正常、特定されていない
neededFeatureNotSupported	31 - 正常、特定されていない
tunnelledSignallingRejected	127 - インタワーキング、特定されていない
invalidCID	3 - 宛先へのルーティング情報なし
hopCountExceeded	3 - 宛先へのルーティング情報なし

表 6/JT-H225.0 AdmissionRejectReason/LocationRejectReason から理由表示情報要素へのマッピング  
(ITU-T H225.0)

AdmissionRejectReason または LocationRejectReason コード	対応する Q. 931/Q. 850の理由値
calledPartyNotRegistered	20 - 加入者不在
invalidPermission	127 - インタワーキング、特定されていない
requestDenied	31 - 正常、特定されていない
undefinedReason	31 - 正常、特定されていない
callerNotRegistered	31 - 正常、特定されていない
routeCallToGatekeeper	使用不可
invalidEndpointIdentifier	127 - インタワーキング、特定されていない
resourceUnavailable	47 - リソース使用不可
securityDenial	31 - 正常、特定されていない
qosControlNotSupported	63 - サービスまたはオプションが使用不可、特定されていない
incompleteAddress	28 - 無効番号フォーマット
aliasesInconsistent	31 - 正常、特定されていない
routeCallToSCN	3 - 着信先へのルートなし

exceedsCallCapacity	41 — 一時的障害
collectDestination	31 — 正常、特定されていない
collectPIN	31 — 正常、特定されていない
genericDataReason	31 — 正常、特定されていない
neededFeatureNotSupported	31 — 正常、特定されていない
securityWrongSyncTime	31 — 正常、特定されていない
securityReplay	31 — 正常、特定されていない
securityWrongGeneralID	31 — 正常、特定されていない
securityWrongSendersID	31 — 正常、特定されていない
securityIntegrityFailed	31 — 正常、特定されていない
securityWrongOID	31 — 正常、特定されていない
securtyDHMismatch	31 — 正常、特定されていない
noRouteToDestination	3 — 着信先へのルートなし
unallocatedNumber	1 — 欠番
noBandwidthAvailable	34 — 回線/チャネルの使用不可

#### 7.2.2.9 チャネル識別子

使用は、今後の検討課題である。複数の呼でフィードバックのために使われるかもしれない。

#### 7.2.2.10 接続番号

5.4.1 節/JT-Q951 に従ってコード化される。

#### 7.2.2.11 接続サブアドレス

5.4.1 節/JT-Q951 に従ってコード化される。

#### 7.2.2.12 輻輳制御レベル

使用してはならない。

#### 7.2.2.13 日時

図 4-21/JT-Q931 に従ってコード化される。

#### 7.2.2.14 表示

図 4-22/JT-Q931 に従ってコード化される。全情報要素の最大長は 82 オクテットである。

#### 7.2.2.15 拡張ファシリティ情報要素

JT-Q95x シリーズで定義されるように、変更されないセマンティックスを示すために使用されるあらゆる拡張されたファシリティ情報要素は、8.2.4 節/JT-Q932 に従ってコード化されなければならない。この場合、サービス ADU、JT-X229 で定義されるように、遠隔操作サービス要素 [JT-X680 (ASN.1 の規格) と JT-X690 (ASN.1 の基本的な符号化規則の規格) を使った] に従って構成されなければならない。

#### 7.2.2.16 ファシリティ

JT-H323 手順に基づいた呼のリダイレクト (呼の転送 (forwarding)、MC への呼のリダイレクト、あるいはゲートキーパーにルーチングされた呼の強制) を示すために、あるいは JT-H450 に従う付加サービスシグナリングの場合に、ファシリティメッセージのユーザ・ユーザ情報要素が使われる。この特定のケースは長さゼロのファシリティ情報要素をコード化することによって示されなければならない。すなわち、ファシリティ情報要素は 以下のようにちょうど 2 オクテットで構成されなければならない：

- オクテット No.1 (情報要素識別子) は、ファシリティ情報要素を示すために、「00011100」 (1CH) に設定されなければならない。
- オクテット No.2 (情報要素長) は、以降にはこの情報要素に属するオクテットは無いことを示すために、「0」に設定されなければならない。



呼の転送 (forwarding) を示すために、ファシリティ情報要素は空でなければならない、そして **Facility-UUIE** (ユーザ・ユーザ情報要素) は **alternativeAddress** あるいは **alternativeAliasAddress** で、呼がリダイレクトされる端末を示さなければならない。この場合、**facilityReason** は **callForwarded** に設定されなければならない。

発信エンドポイントが会議に参加しようとし、着信エンドポイントが **MC** を持っていないことから、エンドポイントに異なるエンドポイントへの呼び出しを指示するために、ファシリティ情報要素は同様に空のままにしておかれる。**conferenceID** は、参加する会議を示さなければならない。そして **Facility-UUIE** (ユーザ・ユーザ情報要素) での理由は、**routeCallToMC** でなければならない。

同じく、被呼エンドポイントのゲートキーパーを経由して被呼エンドポイントにシグナリングするよう発信エンドポイントに指示するために、ファシリティ情報要素は空のままにしておかれる。**Facility-UUIE** (ユーザ・ユーザ情報要素) での **conferenceID** は、参加する会議を示さなければならない。そして **Facility-UUIE** (ユーザ・ユーザ情報要素) での理由は、**routeCallToGatekeeper** でなければならない。

JT-Q95x シリーズで定義されるように、変更されないセマンティックスを示すために使用される、あらゆるどんなファシリティ情報要素でも、8.2.3 節/JT-Q932 に従ってコード化されなければならない。この場合、サービス ADU は、JT-X229 で定義されるように、遠隔操作サービス要素 [JT-X680 (ASN.1 の規格) と JT-X690 (ASN.1 の基本的な符号化規則の規格) を使った] に従って構成されなければならない。

#### 7.2.2.17 高位レイヤ整合性

将来の検討課題。

#### 7.2.2.18 キーパッドファシリティ

図 4-24/JT-Q931 に従ってコード化される。感嘆符「！」を使用してフックフラッシュ表示を意味しなければならない。フックフラッシュ表示の受信をサポートしないエンドポイントは、「！」を受信した場合、それを無視しなければならない。

#### 7.2.2.19 低位レイヤ整合性

将来の検討課題。

#### 7.2.2.20 モアデータ

使用してはならない。

#### 7.2.2.21 網特有ファシリティ

使用してはならない。

#### 7.2.2.22 通知識別子

4.5.22 節/JT-Q931 に従ってコード化される。

#### 7.2.2.23 経過識別子

図 4-29/JT-Q931 と表 4-20/JT-Q931 に従ってコード化される。

この情報要素は、詳細な呼処理情報が利用可能な ISDN または ATM ベースの端末と JT-H323 端末を接続する事に対してのみ必要とする。この場合、ゲートウェイは JT-H323 端末にこの情報を通知するべきである。JT-H323 エンドシステムではこの情報要素を理解する必要はない。

この情報要素が JT-H323 端末によって生成されるのならば、以下の制限がある。

#### コーディング標準 (オクテット No.3, ビット 6,7)

「ITU-T」を示す ('00')。

#### 生成源

表 4-20/JT-Q931 に従う。

「ユーザ」 ('0000'), 「ローカルユーザ収容私設網」 ('0001'), 「リモートユーザ収容私設網」 ('0101') が設定される。

#### 経過内容

表 4-20/JT-Q931 に従う。

#### 7.2.2.24 リダイレクト番号

JT-Q931 の 4.6.7 節に従ってコード化される。この情報要素は、回線交換網 (SCN) との相互接続を容易にするためのみに、用意されており、JT-H323 ベースの着信転送サービスに対する機構は提供しない事に注意

すること。JT-H323 の着信転送サービスは、JT-H450.3 により定義される。

#### 7.2.2.25 繰り返し識別子

使用してはならない。

#### 7.2.2.26 初期設定表示

使用してはならない。

#### 7.2.2.27 分割メッセージ

使用してはならない。

JT-H323 および本標準では、メッセージサイズの明確な上限がないことに注意すること。

#### 7.2.2.28 送信完了

図 4-33/JT-Q931 に従ってコード化される。

対応していない。

#### 7.2.2.29 シグナル

図 4-34/JT-Q931 と表 4-24/JT-Q931 に従ってコード化される。

制約はない。

#### 7.2.2.30 中継網選択

使用してはならない。

#### 7.2.2.31 ユーザ・ユーザ

図 4-36/JT-Q931 と表 4-26/JT-Q931 に従い、ここで定義された内容でコード化される。

ユーザ・ユーザ情報要素は、全 JT-H323 エンティティを通知するのに使用しなければならない。

関係する端末間のみで交換された実際のユーザ・ユーザ情報は **H323-UserInformation** PDU の **user-data** フィールド上でネストされる (PDU に制限はない)。

制約事項を以下に示す：

#### ユーザ・ユーザ内容長

(図 4-36/JT-Q931 で示している) 1 オクテットから変更して 2 オクテットにしなければならない。

#### プロトコル識別子

X.680 および X.690 (ASN.1) 符号化ユーザ情報 ('00000101') にしなければならない。

注：これは ASN.1 の古いバージョンを参照している 1998 年版 JT-Q931 から取得した。ASN.1 の正しいリファレンスは X.680 (シンタックス) と X.691 (PER) である。

#### ユーザ情報

以下に示す現状のユーザデータを含んだ ASN.1 構造 (**H323-UserInformation**) を含まなければならない。これは JT-H323 関連の情報も含む。ASN.1 は X.691 で指定されたパック符号化則の **aligned variant** を用いてコード化される。

**H323-UserInformation** は、**h323-uu-pdu** と **user-data** フィールドを含む。

**H323-UserInformation** 構成の **h323-uu-pdu** フィールドは、以下のフィールドを含む。**h323-uu-pdu** 内の全てのフィールドが、全てのメッセージ中で使えるわけではない事に注意。個々のメッセージの制限事項に関する記述を参照のこと。

**h323-message-body** — このフィールドは、7.3 節や 7.4 節で述べられている様な、特定の JT-H225.0 シグナリングメッセージに特有な情報を含む。ファシリティメッセージが呼に関連しないメッセージの送信に使われる時のように、特定のメッセージ内で **UUIE** フィールド (**Facility-UUIE** 等) を送る必要が無い場合は、送信者は **empty** を選択しても良い。本標準勧告の第 4 版の開始に伴い、メッセージが特定の呼に関連付けられる場合、発信者は **UUIE** フィールドを含めねばならないことに注意すること。これは **callIdentifier** フィールドを準備するために必要である。

**nonStandardData** — このフィールドは、本標準の規定範囲外の情報を送る (例えば、独自データ)。

**h4501SupplementaryService** — このフィールドは、表 3/JT-H430 で定義される、**H4501SupplementaryServiceAPDU** のシーケンスを伝達する。

**h245Tunneling** — JT-H245 メッセージのトンネリングが可能な場合、この要素は **TRUE** に設定される。ファーストコネクト手順が呼の確立に使われる場合、JT-H225 の第 4 版以降に準拠するシステムは、この要素を **TRUE** に設定しなければならない。

- h245Control** — このフィールドは、トンネリングされた JT-H245 PDU のシーケンスを伝達する。それぞれのオクテットストリングは、ただ一つの JT-H245 PDU を含まねばならない。
- nonStandardControl** — このフィールドは、本標準の規定範囲外の制御情報を含む（例えば、独自の制御情報）。
- callLinkage** — このフィールドの内容は、一般的には呼結合サービスにより制御される。このフィールドの手順と意味は、標準 JT-H323 を参照のこと。
- tunnelledSignallingMessage** — ネイティブフォーマットのトンネリングされた全てのシグナリングメッセージ。付加的なエンド・エンド制御シグナリングをサポートする。 **tunnelledProtocolID** フィールドは、トンネリングされたプロトコルを識別する。 **messageContent** フィールドは、トンネリングされた実際のメッセージ全体のネイティブなバイナリーフォーマットにおけるシーケンスである。これは、トンネリングされたメッセージを、一つの JT-H225 メッセージにまとめることを許す。
- tunnellingRequired** フィールドが存在する場合は、トンネリングがサポートされている場合のみ、呼は処理されねばならない。
- provisionalRespToH245Tunneling** — このフラグは、着呼側エンティティが JT-H245 トンネリングをこの呼に適用可能か判断していないことを知らせるために用いられる。このフラグがある場合、受信側エンティティで、 **h245Tunneling** フラグは無視されねばならない。
- stimulusControl** — このフィールドは、ステイミュラスペースのプロトコル用に ITU-T が将来使用するために予約されている。
- genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、これらのパラメータは、JT-H225.0 で透過的に情報をトンネリングするために利用されてもよい。

**H323-UserInfo** 構造の **user-data** フィールドは、以下のフィールドを含む：

- protocol-discriminator** — このフィールドは、表 4-26/JT-Q931 に従いコード化される。
- user-information** — このフィールドは、4.5.30 節/JT-Q931 に従いコード化される。

### 7.3 JT-Q931 に基づいた JT-H225.0 呼制御メッセージ詳細

以下、表中の情報要素長は JT-H323 端末のみにより生成されたメッセージについて示す。ユーザ・ユーザ情報要素のサイズは、明示はしないが、PER 符号化された **H323-Userinformation** シーケンスのサイズとして理解されている。 **H323-Userinformation** の全長は 65,536 オクテットに制限されている。指定された情報長にかかわらず、SCN サイドから転送されたメッセージは 異なったサイズになることがある（JT-H323 ユーザ情報長上限を超える事がある）。

以下の情報要素の必須、オプション、禁止の定義は JT-H323 端末が送信するかどうかについての規定であることに注意。

#### 7.3.1 呼出 (Alerting)

本メッセージは着信ユーザの呼出が開始された事を示すために、着信ユーザが送信する事がある。日常用語では「電話が鳴っている」である。

表 3-2/JT-Q931 を、表 7/JT-H225.0 のように変更する。

表 7/JT-H225.0 呼出 (Alerting)  
(ITU-T H.225.0)

情報要素	JT-H225.0状態 表示(M/F/O)	JT-H225.0情 報長
プロトコル識別子	M	1
呼番号	M	3
メッセージ種別	M	1
伝達能力	O	5-6
拡張ファシリティ	O	8-*
チャンネル識別子	FFS	NA
ファシリティ	O	8-*
経過識別子	O	2-4

通知識別子	O	2-*
表示	O	2-82
シグナル	O	2-3
高位レイヤ整合性	FFS	NA
ユーザ・ユーザ	M	*

ユーザ・ユーザ情報要素は JT-H225.0 メッセージシンタックスで定義された **Alerting-UUIE** 情報を含む。  
**Alerting-UUIE** 情報は、以下の内容を含む：

- protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。
- destinationInfo** — 発呼者側に、呼がゲートウェイを必要とするか否かを決定可能とさせるために **EndpointType** を含む。
- h245Address** — 着呼側エンドポイントまたはゲートキーパーが、JT-H245 シグナリングに用いることを望む特定のトランスポートアドレスである。このアドレスは同様に呼設定受付、経過表示、応答、ファシリティで通知される可能性がある。
- callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな呼識別子。
- h245SecurityMode** — **h245SecurityCapability** セットと共に呼設定メッセージを受信する JT-H323 エンティティは、許容可能で対応している呼設定受付、呼出、経過表示、応答メッセージに **h245SecurityMode** を入れて応答しなければならない。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- fastStart** — ファーストコネク手順のみ使用する。**fastStart** は論理チャネルを開くのに必要なシグナリングをサポートする。**fastStart** は JT-H245 で定義された **OpenLogicalChannel** を使用する。しかし、**fastStart** の送信側は送受信したいモードと、メディアストリームが受信することを期待するトランスポートアドレスを示す。
- multipleCalls** — TRUE である場合、メッセージの送信者は、単一呼制御コネクションを超えて、複数呼のシグナリングが可能であることを示す。
- maintainConnection** — TRUE である場合、メッセージの送信者は、コネクションを超えて現在シグナリングされている呼が存在しない場合に、コネクションのシグナリングをサポートすることが可能であることを示す。
- alertingAddress** — 呼び出し中のパーティに対する別名アドレスを含む。
- presentationIndicator** — **alertingAddress** の表示が許可されるべきか、禁止されるべきかを示す。
- screeningIndicator** — **alertingAddress** がエンドポイントまたは網（ゲートキーパー）により提供されているか否か、および **alertingAddress** がゲートキーパーにより遮蔽されているか否かを示す。
- fastConnectRefused** — 着信エンドポイントはファーストコネク手順を拒否することを示しつつ呼を確立する場合には、呼を確立する時の応答メッセージまたは、それ以前の何らかのメッセージ中でこの要素を返すべきである。
- serviceControl** — 例えば、JT-H323 付属資料 K に述べられているように、発呼側エンドポイントの呼設定手順の一部に使われるかもしれないサービス固有のデータやそれに関連するものを含む（例：呼の着信転送用のオプションメニュー）。
- capacity** — この呼出メッセージが通信中の呼を表している時には、このフィールドは、送信側エンドポイントのこの時点で利用可能な呼キャパシティを示す。このフィールドの送信時にはエンドポイントは **currentCallCapacity** 要素を含まなくてはならない。
- featureSet** — このフィールドは、この呼に関係する一般的な機能セットを記述する。

### 7.3.2 呼設定受付 (Call Proceeding)

本メッセージは要求した呼設定が開始され、これ以上呼設定情報が受け付けられない事を示すために着信側ユーザによって送信されることがある。表 8/JT-H225.0 参照。

表 8/JT-H225.0 呼設定受付 (Call Proceeding)  
(ITU-T H.225.0)

情報要素	JT-H225.0状態 表示(M/F/O)	JT-H225.0 情報長

プロトコル識別子	M	1
呼番号	M	3
メッセージ種別	M	1
伝達能力	O	5-6
拡張ファシリティ	O	8-*
チャネル識別子	FFS	NA
ファシリティ	O	8-*
経過識別子	O	2-4
通知識別子	O	2-*
表示	O	2-82
高位レイヤ整合性	FFS	NA
ユーザ・ユーザ	M	*

ユーザ・ユーザ情報要素は、JT-H225.0 メッセージシンタックスで定義された **CallProceeding-UUIE** 情報を含む。**CallProceeding-UUIE** 情報は、以下の内容を含む：

- protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。
- destinationInfo** — 発呼者側に、呼がゲートウェイを必要とするか否かを決定可能とさせるために **EndpointType** を含む。
- h245Address** — 着呼側エンドポイントまたはゲートキーパーが、JT-H245 シグナリングに用いることを望む特定のトランスポートアドレスである。
- callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな呼識別子。
- h245SecurityMode** — **h245SecurityCapability** セットと共に呼設定メッセージを受信する JT-H323 エンティティは、許容可能で対応している呼設定受付、呼出、経過表示、応答メッセージに **h245SecurityMode** を入れて応答するべきである。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** 暗号化された **tokens**。
- fastStart** — ファーストコネクト手順でのみ使用する。**fastStart** は論理チャネルを開くのに必要なシグナリングをサポートする。**fastStart** は JT-H245 で定義された **OpenLogicalChannel** を使用する。しかし、**fastStart** の送信側はむしろ送受信するモードと、メディアストリームが受信することを期待するトランスポートアドレスを示す。
- multipleCalls** — TRUE である場合、メッセージの送信者は、単一呼制御コネクションを超えて、複数呼のシグナリングが可能であることを示す。
- maintainConnection** — TRUE である場合、メッセージの送信者は、コネクションを超えて現在シグナリングされている呼が存在しない場合に、コネクションのシグナリングをサポートすることが可能であることを示す。
- fastConnectRefused** — 着信側エンドポイントはファーストコネクト手順を拒否することを示しつつ呼を確立する場合には、呼を確立する時の応答メッセージまたは、それ以前の何らかのメッセージ中でこの要素を返すべきである。
- featureSet** — このフィールドは、この呼に関係する一般的な機能セットを記述する。

### 7.3.3 応答 (Connect)

本メッセージは、着信側エンティティが呼を受け付けた事を通知するために、着信側エンティティ (ゲートキーパー、ゲートウェイ、発信端末) が発信側エンティティに通知しなければならない。表 3-4/JT-Q931 に従い、下記の表 9/JT-H225.0 を変更する。

**表 9/JT-H225.0 応答 (Connect)**  
(ITU-T H.225.0)

情報要素	JT-H225.0状態表示(M/F/O)	JT-H225.0情報長
プロトコル識別子	M	1
呼番号	M	3
メッセージ種別	M	1
伝達能力	O	5-6
拡張ファシリティ	O	8-*
チャンネル識別子	FFS	NA
ファシリティ	O	8-*
経過識別子	O	2-4
通知識別子	O	2-*
表示	O	2-82
日時	O	8
接続先番号	O	2-*
接続先サブアドレス	O	2-23
低位レイヤ整合性	FFS	NA
高位レイヤ整合性	FFS	NA
ユーザ・ユーザ	M	*

ユーザ・ユーザ情報要素は、JT-H225.0 メッセージシンタックスで定義された **Connect-UIIE** 情報を含む。**Connect-UIIE** 情報は、以下の内容を含む：

**protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。

**h245Address** — 着呼側エンドポイントまたはゲートキーパーが、JT-H245 シグナリングに用いることを望む特定のトランスポートアドレスである。このアドレスは、呼出、経過表示、呼設定受付、ファシリティのいずれかで先に送信されていた場合には、送信されなければならない。

**destinationInfo** — 発呼者側に、呼がゲートウェイを必要とするか否かを決定可能とさせるために **EndpointType** を含む。

**conferenceID** — 呼設定を受信した会議と、他の会議とをユニークに識別できるユニークな番号が含まれる。

**callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。

**h245SecurityMode** — **h245SecurityCapability** セットと共に呼設定メッセージを受信する JT-H323 エンティティは、許容可能で対応している呼設定受付、呼出、経過表示、応答メッセージに **h245SecurityMode** を入れて応答するべきである。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

**fastStart** — ファーストコネクト手順でのみ使用する。**fastStart** は論理チャンネルを開くのに必要なシグナリングをサポートする。**fastStart** は JT-H245 で定義された **OpenLogicalChannel** を使用する。しかし、**fastStart** の送信側はむしろ送受信するモードと、メディアストリームが受信することを期待するトランスポートアドレスを示す

**multipleCalls** — TRUE である場合、メッセージの送信者は、単一呼制御コネクションを超えて、複数呼のシグナリングが可能であることを示す。

**maintainConnection** — TRUE である場合、メッセージの送信者は、コネクションを超えて現在シグナリングされている呼が存在しない場合に、コネクションのシグナリングをサポートすることが可能であることを示す。

**language** — ユーザが希望するアナウンスあるいはプロンプトの言語を示す。このフィールドは RFC1766 に準拠した一つあるいは複数の言語タグを含む。

**connectedAddress** — 接続された（応答中の）パーティーに対する別名アドレスを含む。接続されたパーティーのダイヤルされたディジット列は、接続番号 IE 中に存在する。

- presentationIndicator** — **connectedAddress** の表示が許可されるべきか、禁止されるべきかを示す。  
**presentationIndicator** および接続番号 IE の表示インジケータの双方が表示可となっており矛盾がある場合、接続番号 IE の表示インジケータが使用されるべきである。
- screeningIndicator** — **connectedAddress** がエンドポイントまたは網（ゲートキーパー）により提供されているか否か、および **connectedAddress** がゲートキーパーにより遮蔽されているか否かを示す。  
**screeningIndicator** および接続番号 IE の遮蔽インジケータの双方が表示可となっており矛盾がある場合、接続番号 IE の遮蔽インジケータが使用されるべきである。
- fastConnectRefused** — 着信エンドポイントはファーストコネクト手順を拒否することを示しつつ、呼を確立する場合には、呼を確立するときの応答メッセージまたは、それ以前の何らかのメッセージ中でこの要素を返すべきである。
- serviceControl** — 例えば、JT-H323 付属資料 K に記述されているように、発呼側エンドポイントの呼設定手順の一部に使われるかもしれないサービス固有のデータやそれに関連するものを含む（例：呼の着信転送用のオプションメニュー）。
- capacity** — この呼出メッセージが通信中の呼を表している時には、このフィールドは、送信側エンドポイントのこの時点で利用可能な呼キャパシティを示す。このフィールドの送信時にはエンドポイントは **currentCallCapacity** 要素を含まなくてはならない。
- featureSet** — このフィールドは、この呼に関係する一般的な機能セットを記述する。

### 7.3.4 応答確認 (Connect Acknowledge)

このメッセージは送られてはならない。

### 7.3.5 切断 (Disconnect)

このメッセージは、JT-H323 エンティティでは送られてはならない。

IP ネットワークから受け取る切断メッセージの内容と意味は 表 3-6/JT-Q931 の様に示され、ISO/IEC 11582 の 10.5 節で定義される。

### 7.3.6 情報 (Information)

このメッセージは付加情報を提供する為に送ってもよい。呼確立（例えばオーバーラップ送信）のための情報、または多種の呼に関連した情報を提供するのに使用してもよい。これは独自の特性の配信のために使われるだろう。

このメッセージは JT-H323 エンティティで送る事が許されている。

このメッセージは以下に示す変更を含み、表 3-7/JT-Q931 に従っている（表 10/JT-H225.0 を参照）。

表 10/JT-H225.0 情報メッセージ内容  
(ITU-T H.225.0)

情報要素	JT-H225.0 状態表示 (M/F/O)	JT-H225.0 情報長
プロトコル識別子	M	1
呼番号	M	3
メッセージ種別	M	1
送信完了	O	1
表示	O	2-82
キーパッド ファシリティ	O	2-34
シグナル	O	2-3
着番号	O (注)	2-35
ユーザ・ユーザ	M	*

注：着番号UUIEはJT-H323の8.1.12節のオーバーラップ送信を用いる時、私設番号計画からの番号を運ぶために使われる。

ユーザ・ユーザ情報要素は、JT-H225.0 メッセージ シンタックスで定義される **Information-UUIE** を含む。  
**Information -UUIE** は、以下の内容を含む：

- protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。
- callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入されなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- fastStart** — このフィールドは含まれてはならず、受信しても無視しなくてはならない。
- fastConnectRefused** — このフィールドは含まれてはならず、受信しても無視しなくてはならない。
- circuitInfo** — このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。

### 7.3.7 経過表示 (Progress)

SCN との相互接続のイベントにおける、呼の経過表示をするために、このメッセージは、JT-H323 ゲートウェイにより送出される。さらに、付加サービスとの相互作用次第で、応答メッセージの前に、このメッセージは JT-H323 エンドポイントにより送出される。

以下の表 11/JT-H225.0 に示す通り、表 3-9/JT-Q931 と ISO/IEC 11582 の 10.10 節に従う事。

**表 11/JT-H225.0 経過表示 (Progress)**  
**(ITU-T H.225.0)**

情報要素	JT-H225.0 状態表示(M/F/O)	JT-H225.0情報長
プロトコル識別子	M	1
呼番号	M	3
メッセージ種別	M	1
伝達能力	O	5-6
理由表示	O	2-32
拡張ファシリティ	O	8-*
チャネル識別子	FFS	NA
ファシリティ	O	8-*
経過識別子	M	2-4
通知識別子	O	2-*
表示	O	2-82
高位レイヤ整合性	FFS	NA
ユーザ・ユーザ	M	*

ユーザ・ユーザ情報要素は、JT-H225.0 メッセージシンタックスで定義される **Progress-UUIE** を含む。  
**Progress-UUIE** は、以下の内容を含む：

- protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。
- destinationInfo** — 発呼者側に、呼がゲートウェイを必要とするか否かを決定可能とさせるために **EndpointType** を含む。
- h245Address** — 着呼側エンドポイントまたはゲートキーパーが、JT-H245 シグナリングに用いることを望む特定のトランスポートアドレスである。このアドレスは、呼設定受付、呼出、応答、ファシリティのいずれかで既に送信されていた場合は、送信されなければならない。
- callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。
- h245SecurityMode** — **h245SecurityCapability** セットと共に呼設定メッセージを受信する JT-H323 エンティ



ティは、許容可能で対応している呼設定受付、呼出、経過表示、応答メッセージに **h245SecurityMode** を入れて応答するべきである。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

**fastStart** — ファーストコネクト手順でのみ使用する。**fastStart** は論理チャネルを開くのに必要なシグナリングをサポートする。**fastStart** は JT-H245 で定義された **OpenLogicalChannel** を使用する。しかし、**fastStart** の送信側は送受信したいモードと、メディアストリームが受信することを期待するトランスポートアドレスを示す

**multipleCalls** — TRUE である場合、メッセージの送信者は、単一呼制御コネクションを超えて、複数呼のシグナリングが可能であることを示す。

**maintainConnection** — TRUE である場合、メッセージの送信者は、コネクションを超えて現在シグナリングされている呼が存在しない場合に、コネクションのシグナリングをサポートすることが可能であることを示す。

**fastConnectRefused** — 着信エンドポイントはファーストコネクト手順を拒否することを示しつつ、呼を確立する場合には、呼を確立する時の応答メッセージまたは、それ以前の何らかのメッセージ中での要素を返すべきである。

### 7.3.8 解放 (Release)

このメッセージは JT-H323 エンティティでは送られてはならない。

受信した解放メッセージの内容と意味は 表 3-10/JT-Q931 の様になり、ISO/IEC 11582 の 10.5 節で定義される。

### 7.3.9 解放完了 (Release Complete)

このメッセージは、呼の解放を示す為に、端末から送られねばならない。その後、呼番号 (CRV) が再利用可能となる。

切断/解放/解放完了シーケンスは、主な用途が回線交換リソースの解放の完了を示すことであるため使用されない。この事は、パケットベースのネットワーク環境には当てはまらないため、解放完了のみを送る、単一ステップの方法が使われる。

表 12/JT-H225.0 に示す修正を含み、表 3-11/JT-Q931 に従っている。

表 12/JT-H225.0 解放完了 (Release Complete)  
(ITU-T H.225.0)

情報要素	JT-H225.0状態表示 (M/F/O)	JT-H225.0情報 長
プロトコル識別子	M	1
呼番号	M	3
メッセージ種別	M	1
理由表示	CM (注1)	2-32
ファシリティ	O	8-*
通知識別子	O	2-*
表示	O	2-82
シグナル	O	2-3
ユーザ・ユーザ	M	*
注1: 理由表示情報要素または <b>ReleaseCompleteReason</b> のいずれかが表示されなくてはならない。		

空のファシリティ情報要素によるファシリティメッセージの応答として、このメッセージが送られる場合は、解放完了理由は **facilityCallDeflection** に設定されなければならない。

このメッセージがゲートウェイにより SCN から転送される場合、理由表示値は JT-Q931 に規定される値に設定されねばならない。

ユーザ・ユーザ情報要素は、JT-H225.0 メッセージシンタックスで定義される **ReleaseComplete-UUIE** を含む。**ReleaseComplete-UUIE** は、以下の内容を含む：

- protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。
- reason** — 呼が解放された理由に関する補足情報。**genericDataReason** の理由は、一般的な要素および機能の結果のように呼がクリアされたことを示す。この場合、付加情報がこのメッセージの **h323-uu-pdu** の **genericData** フィールド内に明記されてもよい。**neededFeatureNotSupported** の理由は、あるエンティティによって要求される機能が他のエンティティでサポートされていないことを示す。**tunnelledSignallingRejected** の理由は、もし送信側が非 JT-H323 シグナルのトンネリングと発呼を成功させるためにトンネリングを要求することを許可していないために呼がクリアされる場合に、送信される。**hopCountExceeded** の理由は **hopCount** の値が 0 に到達したため呼をさらに処理することができないという理由で呼が拒否されたことを示す。
- callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。
- busyAddress** — 話中のパーティに対する別名アドレスを含む。
- presentationIndicator** — **busyAddress** の表示が許可されているか不許可であるかを示す。
- screeningIndicator** — **busyAddress** がエンドポイントまたは網（ゲートキーパー）により提供されているか否か、および **busyAddress** がゲートキーパーにより遮蔽されているか否かを示す。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入されなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- capacity** — この呼出メッセージが通信中の呼を表している時には、このフィールドは、送信側エンドポイントのこの時点で利用可能な呼キャパシティを示す。このフィールドの送信時にはエンドポイントは **currentCallCapacity** 要素を含まなくてはならない。
- serviceControl** — 例えば、JT-H323 付属資料 K に述べられているように、ポストコールサービスのためのサービス固有のデータやそれに関連するものを含む（例：エラーメッセージまたはアナウンス）。
- featureSet** — このフィールドは、この呼に関係する一般的な機能セットを記述する。

### 7.3.10 呼設定 (Setup)

このメッセージは着信側エンティティとの接続設定希望を示すために、発呼側 JT-H323 エンティティから送られねばならない。

表 13/JT-H225.0 に示す通り、表 3-15/JT-Q931 に従う事。

表 13/JT-H225.0 呼設定 (Setup)  
(ITU-T H.225.0)

情報要素	JT-H225.0状態表示 (M/F/O/CM)	JT-H225.0情報 長
プロトコル識別子	M	1
呼番号	M (注2)	3
メッセージ種別	M	1
送信完了	O	1
繰り返し識別子	F	NA
伝達能力	M	5-6
拡張ファシリティ	O	8-*
チャンネル識別子	F/S	NA
ファシリティ	O	8-*
経過識別子	O	2-4

ネットワーク特有ファシリティ	F	NA
通知識別子	O	2-*
表示	O	2-82
キーパッドファシリティ	O	2-34
シグナル	O	2-3
発 番号	O	2-131
発 サブアドレス	CM (注1)	NA
着 番号	O	2-131
着 サブアドレス	CM (注2)	NA
リダイレクト番号	O	2-*
中継網選択	F	NA
低位レイヤ整合性	FFS	NA
高位レイヤ整合性	FFS	NA
ユーザ・ユーザ	M	*
<p>注 1:いくつかの SCN 呼シナリオではサブアドレスが必要である。パケットベースネットワーク側に閉じた呼に使用すべきでない。</p> <p>注 2: ARQが前もって送られたのなら、ここで用いられるCRVは同じでなくてはならない。</p>		

ユーザ・ユーザ情報要素は、JT-H225.0 メッセージ シンタックスで定義される **Setup-UUIE** を含む。**Setup-UUIE** は、以下の内容を含む：

- protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。
- h245Address** — 着呼側エンドポイントまたはゲートキーパーが、JT-H245 シグナリングに用いることを望む特定のトランスポートアドレスである。呼制御チャネルで応答メッセージを受ける前に、JT-H245 手順を操作する事が可能である場合のみ、送信側で用意されるべき物である。
- sourceAddress** — 発信元の別名アドレスを含む。E.164 発信元番号は JT-Q931 発番号情報要素内にある。もし発番号に、1 つでも含まれるならば、プライマリーアドレスが最初でなくてはならない。
- sourceInfo** — 着信側ユーザがゲートウェイを必要とするかしないかの判断を可能にする **EndpointType** を含む。
- destinationAddress** — エンドポイントが接続を希望するアドレス。プライマリーアドレスが最初でなければならぬ。あるエンドポイントにダイヤルされたディジット列のみで発呼した時、このアドレスは JT-H225.0 呼制御メッセージ着番号情報要素に置き換えられねばならない。**destinationAddress** が得られる場合は、第 2 版またはそれ以降の版に準拠した端末は、これを呼設定メッセージの中に入れねばならない。
- destCallSignalAddress** — ゲートキーパーに着信側端末の呼制御 トランスポート アドレスを通知するために必要。端末間直接の場合は冗長である。呼設定メッセージの送信側がこの情報を得られる場合は全て、このフィールドにデータを満たさねばならない。
- destExtraCallInfo** — 追加チャネルの呼、すなわち、SCN 側での 2×64kbps 発呼を可能にするために必要である。ダイヤルされたディジット列、E.164 番号、または私設網番号のみを含まねばならず、インシヤルチャネルの番号を含んではならない。(注を参照)
- destExtraCRV** — **destExtraCallInfo** で規定される追加の SCN 呼のための呼番号 (CRV)。使い方は今後の検討による。RAS シグナリングと JT-H225.0 で使われる変更された JT-Q931 シグナリングとの関連付けに利用可能。
- activeMC** — 発呼側エンドポイントがアクティブ MC の影響下にある事を示す。
- conferenceID** — ユニークな会議識別子。
- conferenceGoal:**
  - create** — 新しい会議を始める。
  - invite** — すでにある会議に招集する。
  - join** — すでにある会議に参加する。

**capability-negotiation** — 疎結合 (Loosely-Coupled) 会議のために、能力ネゴシエーションを行う。

**callIndependentSupplementaryService** — 呼に関連しない方法で PDU 付加サービスを伝送する。

**callServices** — ゲートキーパーや着呼側端末のために、Q シリーズプロトコルでのオプションサポートに関する情報を準備する。

**callType** — この値を使い、着信側ユーザのゲートキーパーは実際の帯域使用量の決定を試みる事が可能である。全ての呼に対するデフォルト値は、**pointToPoint** である。通信中において、**callType** は動的に変更でき、呼設定メッセージが送られた時点では、最終の **callType** は不明であることが認識されるべきである。

**sourceCallSignalAddress** — 送信元のトランスポートアドレスを含む。この値は呼設定の受信者により、ARQ メッセージ内で使われねばならない。情報が呼設定メッセージの送信側で得られる場合はすべて、このフィールドが満たされねばならない。呼設定の送信側が ARQ で使う値と、**sourceCallSignalAddress** は一致し、ARQ 内で呼設定を受け取ったエンドポイントにより返されねばならない。

**remoteExtensionAddress** — 複数のゲートウェイを通過するのにこの情報が必要な場合は、着呼側エンドポイントの別名アドレスを含む。情報が呼設定メッセージの送信側で得られる場合はすべて、このフィールドが満たされるべきである。

**callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。

**h245SecurityCapability** — JT-H245 チャンネルを安全にするために送信側が使用できる能力セット。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

**fastStart** — ファーストコネクト手順でのみ使用する。**fastStart** は論理チャンネルを開くのに必要なシグナリングをサポートする。**fastStart** は JT-H245 で定義された **OpenLogicalChannel** を使用する。しかし、**fastStart** の送信側は送受信したいモードと、メディアストリームが受信することを期待するトランスポートアドレスを示す。

**mediaWaitForConnect** — TRUE の場合、呼設定メッセージの受信者は、応答メッセージを送るまではメディアを送信してはならないことを示す。

**canOverlapSend** — TRUE の場合、呼設定の送信側はオーバーラップ送出をサポートしなければいけないことを示す。

**endpointIdentifier** — これは RCF メッセージで端末に割り当てられるエンドポイントの識別子である。このフィールドは、端末が自らが登録されているゲートキーパーに呼設定を送る場合に現われなければならない。他のエンティティに送る場合には現われてはならない。

**multipleCalls** — TRUE の場合、メッセージの送信元は単一の呼制御接続に対して複数のシグナリングに対応できることを示す。

**maintainConnection** — TRUE の場合、メッセージの送信元は、接続に対して現在呼がひとつも送信されていない場合に、接続を通知することができることを示す。

**ConnectionParameters** — 複数の接続タイプおよび／あるいはアグレーション (例として、JT-H323/H320 ゲートウェイ) を提供するゲートウェイに必要なパラメータを規定することを認める。

**scnConnectionType** — 全体の SCN 呼を生成するため使用される個々の接続のタイプにおいてゲートウェイに情報を提供する。ゲートウェイに対する情報があればエンドポイントまたはゲートキーパーはこのフィールドを埋めるべきである。オプションの 'マルチレート' を表示する場合は、ベアラ能力の中の転送レートオクテット情報は 'マルチレート' を表示しなければならず、かつレートマルチプライアオクテットで接続の数を示さなければならない。他のすべての場合で、**scnConnectionType** フィールドが現われる場合には、ベアラ能力情報要素の転送レート (オクテット 4) とレートマルチプライア (オクテット 4.1) に含まれる個々の接続タイプに関する通知は無効になる。

**numberOfSCNConnections** — SCN 呼を生成するために、共にアグレートされる **scnConnectionType** のタイプの接続数を示す。**scnConnectionType** で指定された個々の接続のバンド幅で掛算される場合には、このフィールドは SCN の中の全体の呼に対するバンド幅であることを意味する。エンドポイントまたはゲートキーパーは情報があればこのフィールドを埋めるべきである。  
**scnConnectionType** が不定 (unknown) に設定された場合、64kbit/s のバンド幅の単位であると見なすことに注意すること。この両方のフィールドと **scnConnectionType** が現われた場合には、全体のバンド幅は、ベアラ能力情報要素の転送レート (オクテット 4) とレートマルチプライア (オクテット 4.1) で示された SCN 全体のバンド幅に一致していなければならない。

**scnConnectionAggregation** — 個々の接続が、どのように完全な SCN 呼を生成するために共にアグレートされるかを示す。エンドポイントまたはゲートキーパーは情報があればこのフィールドを埋めるべきである。実際のアグレーションのメカニズムが不定 (unknown) の場合に使用されるデフォルトのオプションは 'auto' である。ボンディングが使用されることを知っているが詳細のボンディングモードが不定 (unknown) の場合には、オプションの 'ボンディングモード 1 (bonded-mode1)' を使用すべきである。

- language** — ユーザが希望するアナウンスあるいはプロンプトの言語を示す。このフィールドは RFC1766 に準拠した一つあるいは複数の言語タグを含む。
- presentationIndicator** — 発アドレスの表示が許可されているか、あるいは制限されているかを示す。  
**PresentationIndicator** と発番号情報要素の中の両方にこれが含まれ不一致である場合には、発番号情報要素のそれを使用しなければならない。
- ScreenIndicator** — 発アドレスが、エンドポイントと網（ゲートキーパー）のいずれにより提供されたかを示す。また、発アドレスがゲートキーパーにより隠蔽されているかを示す。**ScreenIndicator** と発番号情報要素の中の両方にこれが含まれ不一致である場合には、発番号情報要素のそれを使用しなければならない。
- serviceControl** — サービスに特有のデータまたはそれへの参照を含み、着信先エンドポイント（例えば呼出中に表示されるべきイメージやアイコン）で呼設定手順の一部として使ってもよい。これは JT-H323 付属資料 K に例として記述がある。
- symmetricOperationRequired** — もし存在すれば、着信先エンドポイントは同一の送信と受信のオーディオ能力を選択しなければならないことを示す。**fastStart** 要素を含めずに、この要素を含めてはならない。
- capacity** — このフィールドは、この時刻における送信側エンドポイントの利用可能な呼キャパシティを示すが、呼設定は通信中の呼を示すものとする。このフィールドを送る場合には、エンドポイントは **currentCallCapacity** 要素を含めなければならない。
- circuitInfo** — このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。
- desiredProtocols** — これはプロトコルのタイプの選択の順序であり、この呼（例えばオーディオ、FAX）に対し発呼側エンドポイントが望むものであることを示す。解決するエンティティは、このフィールドを、選択の順を考慮しながら、そのプロトコルをサポートするエンドポイントのアドレスを特定するために使ってもよい。
- neededFeatures** — このフィールドは呼を成立させるために必要な一般的な機能のリストを示す。
- desiredFeatures** — このフィールドは呼で選択された一般的な機能のリストを示すが、呼を成立させるためには必要とされない。
- supportedFeatures** — このフィールドは送信側がサポートし選択することを表した一般的な機能のリストを示す。
- parallelH245Control** — このフィールドは、トンネリングされた JT-H245 端末能力セットの PDU とマスター・スレーブ決定を行う PDU の列を送る。それぞれのオクテットの文字列は、正確にひとつの JT-H245 PDU を含まなければならない。
- additionalSourceAddresses** — このフィールドは、非 JT-H323 ネットワークにおいて、2 番目またはそれに続く発アドレス情報要素に対応する別名アドレスの列を送る。例えば ISDN では、JT-Q951 付属資料 A で規定された「2つの発アドレス情報要素の配送オプション」をサポートするために、複数の発アドレスが存在してもよい。
- hopCount** — 呼制御がたどることが可能なホップ数を示す整数値を記す。

注：**destExtraCallInfo** が存在する場合、それぞれの呼の呼番号 (CRV) は **destExtraCRV** の中で供給される。これらの呼番号 (CRV) はそれぞれの呼に対するレスポンスを識別するのに用いられる。これらの手順は今後の検討による。**destExtraCRV** フィールドが無い場合は、ゲートウェイは全ての呼情報を集めて一つの応答としなければいけない。その結果、1つの呼が SCN 側で失敗した場合、全ての呼は失敗として扱われる。

### 7.3.11 呼設定確認 (Setup Acknowledge)

このメッセージは JT-H323 エンティティによって送出される。しかし、ゲートウェイ経由でネットワークからフォワードされるかもしれない。受信における処理はオプションであるが、呼設定の中で **canOverlapSend** を示すエンティティは呼設定確認をサポートしなければならない。

ネットワークから受信した呼設定確認メッセージの内容と意味は表 14/JT-H225.0 に示す修正を含み、表 3-16/JT-Q931 に定義される。

表 14/JT-H225.0 呼設定確認 (Setup Acknowledge)  
(ITU-T H.225.0)

情報要素	JT-H225.0状態表示(M/F/O)	JT-H225.0情報長
------	----------------------	--------------

プロトコル識別子	M	1
呼番号	M	3
メッセージ種別	M	1
チャンネル識別子	FFS	NA
表示	O	2-82
ユーザ・ユーザ	M	*

JT-H225.0 第 4 版より前のシステムとの下位互換性のために、このメッセージの送信側は、ユーザ・ユーザ情報要素の **h323-message-body** フィールドに **h4501SupplementaryService** や **h245Control** フィールドを含めてはならない。

ユーザ・ユーザ情報要素は、JT-H.225.0 Message Syntax で定義されている **SetupAcknowledge-UUIE** を含む。**SetupAcknowledge -UUIE** は、以下の内容を含んでいる：

**protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。

**callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

### 7.3.12 状態表示 (Status)

状態表示は、未知の呼制御メッセージ、または、状態表示問合せメッセージへの応答に使用されねばならない。

表 15/JT-H225.0 に示す修正を含み、表 3-17/JT-Q931 に従う。

表 15/JT-H225.0 状態表示 (Status)  
(ITU-T H.225.0)

情報要素	JT-H225.0状態表示(M/F/O)	JT-H225.0情報長
プロトコル識別子	M	1
呼番号 (注1)	M	3
メッセージ種別	M	1
理由表示	M	4-32
呼状態	M	3
表示	O	2-82
ユーザ・ユーザ	M	*
注1: 複数の呼を含む接続にて、このメッセージが全ての呼に適用されるならば、メッセージに全体的な呼番号を含めてよい		

JT-H225.0 第 4 版より前のシステムとの下位互換性のために、このメッセージの送信側は、ユーザ・ユーザ情報要素の **h323-message-body** フィールドに、**h4501SupplementaryService** や **h245Control** フィールドを含めてはならない。

ユーザ・ユーザ情報要素は、JT-H.225.0 メッセージシンタックスで定義されている **Status-UUIE** を含む。**Status-UUIE** は、以下の内容を含む：

**protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。

**callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

### 7.3.13 状態問合せ (Status Inquiry)

状態問合せは、JT-H323 の 8.4.2 節で記述されるように呼状態表示を要求するのに使用される。表 16/JT-H225.0 に示す修正を含み、表 3-18/JT-Q931 に従う。

表 16/JT-H225.0 状態問合せ (Status Inquiry)  
(ITU-T H.225.0)

情報要素	JT-H225.0状態表示(M/F/O)	JT-H225.0情報長
プロトコル識別子	M	1
呼番号 (注1)	M	3
メッセージ種別	M	1
表示	O	2-82
ユーザ・ユーザ	M	*

注1: 複数の呼を含む接続にて、このメッセージが全ての呼に適用されるならば、メッセージに全体的な呼番号を含めてよい

JT-H225.0 第 4 版より前のシステムとの下位互換性のために、このメッセージの送信側は、ユーザ・ユーザ情報要素の **h323-message-body** フィールドに **h4501SupplementaryService** や **h245Control** フィールドを含めてはならない。

ユーザ・ユーザ情報要素は、JT-H.225.0 メッセージシンタックスで定義されている **StatusInquiry-UUIE** を含む。**StatusInquiry-UUIE** は、以下の内容を含む：

**protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。

**callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

### 7.4 JT-H225.0 呼制御に基づいたJT-Q932 メッセージ詳細

以下で定義されるメッセージは、JT-Q932 と JT-H450 に由来している。詳細は、JT-Q932 と JT-H450 を参照のこと。

#### 7.4.1 ファシリティ (Facility)

ファシリティメッセージは、呼の宛先 (**FacilityReason = routeCallToMC**)、あるいはエンドポイントに着呼がゲートキーパーを通過しなければならないことを示す (**FacilityReason = routeCallToGatekeeper**) 情報を与えるために用いられなければならない。

JT-H323 手順に特有の呼のリダイレクト用シグナリングのために、ファシリティメッセージのユーザ・ユーザ情報要素が用いられる。この特別なケースの場合は、ファシリティ情報要素を内容長 0 でコード化することでその旨を明示しなければならない。この場合、ファシリティ情報要素は 2 オクテットで構成される。JT-H323 エンティティは (JT-H323 特有の) 空のファシリティ情報要素を適切に処理しなければならない、他の認識できないファシリティ情報要素をスキップ可能でなければならない。

ファシリティメッセージは、JT-H450 シリーズによる付加サービスの要求や確認のために使用してもよい。その場合には、ファシリティメッセージのユーザ・ユーザ情報要素に 1 つ以上の JT-H450 付加サービス APDU を含んで伝送しなければならない。JT-H450 付加サービス APDU は、JT-H450.1 の 8 章に従ってコード化されなければならない。その中には、内容長が 0 のファシリティ情報要素も含まなければならない。JT-H450 付加サービス APDU だけを含む JT-H225.0 第 2 版もしくは第 3 版のファシリティメッセージは、**h323-message-body** 選択型の「empty」を使う代わりに、**Facility-UUIE** を含まないことを選択してもよいことに注意すること。この場合、ファシリティメッセージは **callIdentifier** フィールドをその中に有さなくても良い。JT-H225.0 第 4 版以降においては、送信側は全ての呼対応ファシリティメッセージ中に **callIdentifier** フィールドを伝送する **FacilityUUIE** を含む必要があり、**reason** フィールド値を **transportedInformation** に設定

する必要がある。

JT-Q932 や JT-Q95x で定義に従ってコード化され、JT-Q932 のセマンティクスを伝達するファシリティ情報要素が存在する場合、そのメッセージの長さは JT-Q932 の表 7-2 で要求されているように少なくとも 8 オクテットなければならない。このタイプのファシリティ情報要素の使用は今後の検討課題である。

ファシリティメッセージは、エンドポイントまたはゲートキーパーが 2 つのエンティティ間の JT-H245 チャネル確立のための受信を要求するために使われてもよい ( **FacilityReason=startH245** ) 。

ファシリティメッセージの **tokens** および/または **cryptoTokens** フィールド中の新たなトークンセットを送信するために、ファシリティメッセージはエンドポイントやゲートキーパーによって利用されてもよい ( **FacilityReason = newTokens** ) 。これは例えば、トークンが限定された時間内に行われる処理を許可する様なアプリケーションに対して有用である。

下の表 17/JT-H225.0 は、JT-Q932 の 7.1.1 節と ISO/IEC11582 の 10.8 節の表を一部変更したものである。

表 17/JT-H225.0 ファシリティ (Facility)  
(ITU-T H.225.0)

情報要素	JT-H225.0 状態表示 (M/F/O)	JT-H225.0 情報長
プロトコル識別子	M	1
呼番号 (注 1)	M	3
メッセージ種別	M	1
拡張ファシリティ	O (注 2)	8-*
ファシリティ	O (注 2)	2or8-*
通知識別子	O	2-*
表示	O	2-82
発番号	F	NA
着番号	F	NA
ユーザ・ユーザ	M	*
<p>注 1 : もしこのメッセージが複数の呼を伝送する接続において全ての呼に対し適用されるならば、そのメッセージはグローバル呼番号を伝送することが可能である。            注 2 : ファシリティメッセージが JT-Q95x 付加サービスシグナリングを伝達する場合、ファシリティか拡張ファシリティのいずれかが必要である。ファシリティメッセージが、JT-H450.x シリーズによる付加サービス制御に使用される場合、あるいは MC/GK 機能への再ルーティングに使用される場合は、内容長 0 のファシリティ情報要素が必要である。</p>		

メッセージタイプ情報要素の符号化 :

ファシリティメッセージのメッセージタイプ情報要素は「0110 0010」で表現される。

ユーザ・ユーザ情報要素は、JT-H225.0 メッセージシンタックスで定義されている **Facility-UUIE** を含む。

**Facility-UUIE** は、以下の内容を含んでいる :

**protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。

**alternativeAddress** — 発呼側が呼を配送するべき特定のトランスポートアドレス。これが存在する場合、**alternativeAliasAddress** は不要。

**alternativeAliasAddress** — 呼をリダイレクトするためのいくつかの別名アドレスを含む。別名アドレスが与えられていれば、**alternativeAddress** は不要。

**conferenceID** — ユニークな会議識別子。**conferences** フィールドを使用する場合は不要。

**reason** — ファシリティメッセージについての追加情報。

**FeatureSetUpdate** の **reason** は、メッセージの目的が以前に送信されている **featureSet** 情報を更新することであることを示している。**ForwardedElements** の **reason** は、そのメッセージを送信することができなかった場合に、メッセージの目的が他のメッセージの要素を転送することであることを示している。例えば、ルーティングを行うゲートキーパーがすでに呼設定受付を送信した後に呼設定受付メッセージを受信した場合、**transportedInformation** の **reason** は、メッセージの目的が高レイヤの情報を伝



送することであることを示している。例えば **h4501SupplementaryService** フィールド中；すなわちこの場合 **Facility-UIE** は **callIdentifier** を提供するためだけに含まれている。

- callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。
- destExtraCallInfo** — 追加チャネルの呼、すなわち SCN 側の 2×64Kbit/s 呼を可能にするために必要である。ダイヤルされたディジット列、E.164 番号や、私設網番号だけを含まなければならない、イニシャルチャネルの番号を含んではならない。
- remoteExtensionAddress** — 複数のゲートウェイを経由するために必要な場合に、着呼側エンドポイントの別名アドレスが含まれる。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- conferences** — 参加してよい 1 つ以上の会議。
- h245Address** — このファシリティを送信するエンドポイントあるいはゲートキーパーが JT-H245 シグナリングを設定する相手先のトランスポートアドレスを示す。このフィールドは、中継シグナリングエンティティが呼設定受付メッセージから **h245Address** フィールドを伝送している時に現れる可能性があることに注意すること。受信するエンティティは **reason** が **startH245** である時のみ JT-H245 手順を初期化することを指示される。
- fastStart** — ファーストコネクト手順のみで使用する。**fastStart** は論理チャネルを開くために必要なシグナリングをサポートする。これは JT-H245 標準内に定義されている **OpenLogicalChannel** 構造を使用するが、この送信側は受信と送信を実施するモードやメディアストリームの送信先アドレスを示す。ルーチングするゲートキーパーが着呼利用者からの呼設定受付メッセージ内でファシリティメッセージを受信し、発呼利用者に情報を転送する時、このフィールドはファシリティメッセージ内に存在する。このフィールドはエンドポイントが内包してはいけない。
- multipleCalls** — TRUE の場合、メッセージの送信側が単一呼制御接続上で複数呼をシグナリングする能力を有していることを示す。
- maintainConnection** — TRUE の場合、接続上に呼がシグナリングされていない時にメッセージの送信側がシグナリング接続をサポートする能力を有していることを示す。
- fastConnectRefused** — 着信エンドポイントはファーストコネクト手順を拒否することを示しつつ、呼を確立する場合には、呼を確立する時の応答メッセージまたは、それ以前の何らかのメッセージ中でこの要素を返すべきである。ルーチングするゲートキーパーが着呼利用者からの呼設定受付メッセージ内でファシリティメッセージを受信し、発呼利用者に転送する時、このフィールドはファシリティメッセージ内に存在する。
- serviceControl** — サービス仕様データを含んだり、それに対する参照を行う。例えば JT-H323 付属資料 K に書かれている様にエンドポイントやゲートキーパーにより使用される（例えば、呼の参加者にオプションメニューを見せる）。
- circuitInfo** — このフィールドはこの呼のために使われる SCN 回線に関する情報を提供する。
- featureSet** — このフィールドは、この呼に関連する一般的な機能セットを記述する。
- destinationInfo** — 発呼者側に、呼がゲートウェイを必要とするか否かを決定可能とさせるために **EndpointType** を含む。ルーチングするゲートキーパーが着呼利用者からの呼設定受付メッセージ内でファシリティメッセージを受信し、発呼利用者に転送する時、このフィールドはファシリティメッセージ内に存在する。このフィールドは JT-H225.0 第 4 版より以前にはファシリティメッセージに存在しなかった。
- h245SecurityMode** — **h245SecurityCapability** セットと共に呼設定を受信する JT-H323 エンティティは、許容可能で対応している呼設定受付、呼出、経過表示、応答メッセージに **h245SecurityMode** を入れて応答すべきである。ルーチングするゲートキーパーが着呼利用者からの呼設定受付メッセージ内でファシリティメッセージを受信し、発呼利用者に転送する時、このフィールドはファシリティメッセージ内に存在する。このフィールドは JT-H225 第 4 版より以前にはファシリティメッセージに存在しなかった。

#### 7.4.2 通知 (Notify)

JT-H323 エンティティは、このメッセージを送信してもよい。受信処理はオプションである。次表 18/JT-H225.0 は表 3-8/Q931 を修正したものである。

表 18/JT-H.225.0 通知 (Notify)  
(ITU-T H.225.0)

情報要素	JT-H225.0 状態表示 (M/F/O)	JT-H225.0 情報長
------	---------------------------	---------------

プロトコル識別子	M	1
呼番号	M	3
メッセージ種別	M	1
伝達能力	O (注 1)	5-6
通知識別子	M	3
表示	O	2-82
ユーザ・ユーザ	M	*
注 1：伝達能力の変更通知を含む		

JT-H225.0 第 4 版より前のシステムとの下位互換性のために、このメッセージの送信側は、ユーザ・ユーザ情報要素の **h323-message-body** フィールド内に **h4501SupplementaryService** もしくは **h245Control** フィールドを含んではいけない。

ユーザ・ユーザ情報要素は、JT-H225.0 のメッセージシンタックスに定義されている **Notify-UUIE** を含んでいる。**Notify-UUIE** は、以下の内容を含む：

**protocolIdentifier** — サポートしている JT-H225.0 の版に設定する。

**callIdentifier** — RAS シグナリングと本標準で使用するために変更された JT-931 シグナリングを関連付けるために使われる、発呼側エンドポイントによって設定されたグローバルでユニークな識別子。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

#### 7.4.3 その他のメッセージ

オプションファシリティ、拡張ファシリティ、または通知識別子情報要素を運ぶことが可能な呼制御メッセージは、8.3 節で詳述する。

#### 7.5 JT-H225.0 呼制御タイマ値

次の JT-Q931 タイマがサポートされなければならない。

- 「呼設定タイマ」（T303/JT-Q931 表 9-1、表 9-2 参照）は、発呼側エンドポイントが呼設定メッセージを送信した後、着呼側エンドポイントからの呼出、呼設定受付、応答、解放完了、その他のメッセージを待たなければならない時間を定義する。このタイムアウト値は 4 秒以上でなければならない。本質的に長い遅延を持つアプリケーションがネットワーク上に現れることがあることに注意（例えば、インターネットをローカルな企業内ネットワークやイントラネットと比較してみよ）。
- 「確立タイマ」T301（JT-Q931 表 9-1、表 9-2 参照）は、発呼側エンドポイントが着呼側エンドポイントからの返答を待つのを止めなければならない時間を定義する。このタイマは呼出を受信したときにスタートし、応答を受信した場合、あるいは発呼側が呼設定を中断し、解放完了を送信した場合に正常終了する。このタイムアウト値は 180 秒（3 分）以上でなければならない。
- 「オーバーラップ送信タイマ」T302（JT-Q931 表 9-1、表 9-2 参照）は、オーバーラップ送信中の着呼側エンドポイントからのダイヤル番号を待つのを止めなければならない時間を定義する。このタイマは呼設定確認が送信された場合、あるいは、付加情報を受信したときにスタートし、送信完了通知を受信した場合に正常終了する。このタイムアウト値は 10～15 秒でなければならない。
- 「オーバーラップ受信タイマ」T304（JT-Q931 表 9-1、表 9-2 参照）は、発呼側エンドポイントがオーバーラップ受信中の着呼側エンドポイントユーザからのダイヤル番号を待つのを止めなければならない時間を定義する。このタイマは呼設定確認を受信したときにスタートし、付加情報が送信されたときにリスタートし、呼設定受付、呼出、応答を受信した場合に正常終了する。このタイムアウト値は 20 秒以上でなければならない。
- 「入り側呼設定受付タイマ」T310（JT-Q931 表 9-1、表 9-2 参照）は、着呼側エンドポイントがオーバーラップ送信中の着呼側エンドポイントからのダイヤル番号を待つのを止めなければならない時間を定義する。このタイマは呼設定確認を受信したときにスタートし、呼出、応答の場合、あるいは発呼側が呼設定を中断し、解放完了を送信した場合に正常終了する。1 または 8 の値を持つ経過識別子 IE を受信すると、T310 タイマをストップし、T301 タイマをスタートしなければならない。このタイムアウト

ト値は 10 秒以上でなければならない。

- 「状態タイマ」 T322 (JT-Q931 表 9-1、表 9-2 参照) は、着呼側エンドポイントが送信した状態問合せへの状態メッセージ応答を待つのを止めなければならない時間を定義する。このタイマは状態問合せが送信された場合にスタートし、状態メッセージを受信したときに正常終了する。このタイムアウト値は 4 秒以上でなければならない。

パケットネットワーク側でのこれらのタイマ値は、SCN で用いられる値と同じであることに注意。

JT-H450.x 付加サービスのオプションの一部としてその他のタイマをサポートしてもよい。

## 7.6 JT-H225.0 共通メッセージ要素

これは 1 つ以上の RAS (登録、参加承認、状態) メッセージに用いられる ASN.1 構造を記述する。呼制御メッセージのユーザ・ユーザ部で用いてよいものもある。

メッセージ内の **requestSeqNum** は、処理中の複数の要求の進捗を管理するために用いられる。関連するあらゆる応答メッセージ (成功または失敗) を返送する際には、対応する **requestSeqNum** を付加しなければならない。再送されたメッセージは同じ **requestSeqNum** を持っている。**requestSeqNum** は 1 ずつインクリメントされ、モジュロ 65,536 である。

**protocolIdentifier** は、呼に関わっているユーザによる関連する実装タイプの決定を可能にするために、発見、登録、呼設定/応答の一部として含まれる。

**nonStandardParameter** : このパラメータは、呼に関わっているパーティに、関連するエンドポイントの非標準状態の決定を可能にするための発見、登録、設定/応答シーケンス内のオプションである。ゲートキーパーあるいはゲートウェイは、動作に支障を来たすかもしれない、非サポートあるいは認識されない **nonStandardData** を通過させる義務を負わない。

**TransportAddress** 構造は、様々なトランスポートフォーマットに対応しており、TSAP 識別子へのローカル参照に加えて、あらゆるトランスポート仕様体系も含んでいる。

IPv4 や IPv6 アドレスは、アドレスの最上位オクテットが各々の OCTET STRING の最初のオクテットになるようにコード化されなければならない。すなわち、クラス B IPv4 アドレス 130.1.2.97 は、「130」を OCTET STRING の第 1 オクテットにコード化し、続いて「1」と第 4 オクテットまでコード化する。

IPv6 アドレスの a148:2:3:4:a:b:c:d は、最初のオクテットに「a」、次に「48」、3 番目に「00」、4 番目に「02」とコード化する。

**route SEQUENCE** がエンティティを持たないタイプ **ipSourceRoute** の **TransportAddress** は、同じ **ip** と **port** を含むタイプ **ipAddress** と同じアドレスを示すと解釈しなければならない。

IPX アドレスの **node**、**netnum**、**port** は、個々のフィールドの最上位オクテットが各々の OCTET STRING の第 1 オクテットとしてコード化されなければならない。

**TransportAddress** 構造は、トランスポートアドレス = 「パケットネットワークアドレス + TSAP 識別子」という JT-H323 の言語仕様を用いないことに注意すること。代わりに、それぞれのトランスポートドメイン内の共通言語が用いられる。

**EndpointType** 構造は、シグナリングリンクを終端する JT-H323 エンティティについての情報を伝送する。JT-H323 エンティティは 1 つ以上の **gatekeeper**、**gateway**、**mcu**、または **terminal** メッセージ要素を完結される。JT-H323 エンティティが MC を持つ場合、**mc Boolean** は **TRUE** になる。JT-H323 の 6.3 節は、ゲートウェイと共に配置された MCU の記述を示す。この場合、JT-H323 装置は、その **EndpointType** 定義の中に **gateway** と **mcu** の要素の両方を含んでよい。**set** 要素があらわれることは、エンティティが、例えば JT-H323 付属資料 F で定義されたシンプル・エンドポイント・タイプ (SET) 装置であることを示す。**set** 要素のビット位置は SET 装置のタイプを示すが、その意味は JT-H323 付属資料 F と SET 装置タイプを規定する別の標準で定義される。**supportedTunnelledProtocols** フィールドは、サポートしているトンネリングされたプロトコルの優先順位リスト (優先順位が最も高いものを最初に配置) を提供する。

**TunnelledProtocol** 構造は、例えば JT-H323 付属資料 M.1 および M.2 で示すように、トンネリングされたシグナリングプロトコルを識別する。**tunnelledProtocolObjectID** フィールドは、トンネリングされているプロトコルを識別する **OBJECT IDENTIFIER** である。**tunnelledProtocolAlternateID** は、代替の識別子のフォ

フォーマットを提供する。**subIdentifier** フィールドは、標準プロトコルの特定の版の指定を可能にする。

**TunnelledProtocolAlternateIdentifier** 構造は、トンネリングされたプロトコルに対し、文字列を基本とした識別子のフォーマットを提供する。**protocolType** は、**ISUP** のような一般的なプロトコルのタイプを提供する。**protocolVariant** フィールドは、**ANSI** のような標準の特定のバリエーションを提供する。

本標準で定義された、トンネリングされたプロトコルは、本標準付録VIの表VI-1 および表VI-2 で示される。ただし、トンネリングは、これらの表にリストされたプロトコルに限定されないことに注意。

**GatewayInfo** 構造は、**protocol** 要素を含む。ゲートウェイがそのサポートするプロトコルを示すためにこの要素を使用する。

**SupportedProtocols** 構造は、JT-H323 エンティティが、そのプロトコルとの相互接続が可能であるところのプロトコルを選択したことを示す。例えば、**h310** を選択した場合、エンティティが JT-H310 における相互接続を提供することを示す。

それぞれのサポートされるプロトコル能力に関する構造 (**H310Caps**、**H320Caps**、その他) において、**dataRatesSupported** 要素は、デバイスがサポートする個々のプロトコルのためにサポートされているデータレートを示す。**supportedPrefixes** 要素は、サポートされるプロトコルに関連するプレフィクスを示し、場合によってはデータレートも示す。

**McuInfo** 構造は、MCU がサポートするプロトコルを示すことを可能にする、**protocol** 要素を含む。

**CapacityReportingCapability** 構造は、エンドポイントの呼キャパシティ情報をレポートする能力を示す。

**CapacityReportingSpecification** 構造は、エンドポイントに対しレポートが要求される呼キャパシティ情報を示す。**callStart** は、呼の開始時 (即ち、ARQ または呼設定の中で) における、呼キャパシティ情報の要求を示す。**callEnd** は、呼の終了時 (即ち、DRQ または解放完了の中で) における、呼キャパシティ情報の要求を示す。空の **when** シーケンスは、エンドポイントがキャパシティ情報をレポートしないことを要求することを示す。

**CallCapacityInfo** 構造は、そのエンドポイントがサポートする呼の各タイプごとに受信可能な呼のキャパシティを示すことを可能とする。それゆえ、**CallCapacityInfo** 構造はエンドポイントの現在のアイドル状態を示す。例えば、オーディオゲートウェイの **CallCapacityInfo** は、アイドル中の回線数を示すであろう。

**CallCapacity** 構造は、そのエンドポイントがサポートする呼の各タイプごとの最大のキャパシティと、呼の各タイプごとの現在利用可能なキャパシティを示すことを可能とする。

**CallsAvailable** 構造は、エンドポイントの呼キャパシティの合計のサブセットを表す。**group** フィールドは、サブセットがグループ (group) をラベルとして識別されることを可能にする。**group** は、**CircuitIdentifier** にてレポートされるものと同一でよい。

**DataRate** 構造は、ゲートウェイプロトコルレート情報を提供する。**channelRate** は、100 ビット単位の基本チャンネルレートである。**channelMultiplier** は、**channelRate** のチャンネル数を示す。例えば、ゲートウェイが 3B 呼をサポートする場合、**channelMultiplier** = 3 でチャンネル数を表し、**channelRate** = 640 で 64kbps のチャンネルであることを示す。

**VendorIdentifier** 構造は、ベンダーに製品の識別を可能にする。**vendor** 要素は国番号 (country code)、国番号の拡張部分 (extension)、および製造者コードによる識別を可能にする。**productId** や **versionId** は製品情報を与える文字列である。**EnterpriseNumber** フィールドは製造者を識別し、Internet Assigned Numbers Authority (IANA) にて割り当てられる。

**H221NonStandard** 構造は、非標準 (nonstandard) フィールドの定義を可能にする。**t35CountryCode** 要素は、T.35 AnnexA で記述されるように、国を識別しなければならない。**t35Extension** 要素は、**t35CountryCode** がバイナリ "1111 1111" である場合を除いて、国別に割り当てられている国番号の拡張部分 (country code extension) を含まなければならない。この場合 (**t35CountryCode** がバイナリ "1111 1111" である場合) には、このフィールドは T.35 AnnexB で示す国番号を含まなければならない。**manufacturerCode** は、国別に割り当てられ、製造業者を識別しなければならない。

**AliasAddress** 構造は、パケットネットワーク上の特定のトランスポートロケーションを参照する、様々な拡張アドレスに対応している。ゲートキーパーにダイヤルされたディジットを含むアドレスを登録する場合、エンドポイントは **dialedDigits** フィールドを使用し、0～9 のディジット以外使用してはならない。ゲートキーパーに E.164 アドレスを登録する場合、エンドポイントは **e164Number** フィールドにおいて 0～9 のディジット以外使用してはならない。登録またはその他の場合において、プレフィクスを記述する場合、エンドポイントは **dialedDigits** フィールドを使用し、0～9、「#」および「\*」のディジット以外使用してはならない。**mobileUIM** フィールドは、第 2 世代および第 3 世代の無線システム互換性に対するモジュール識別子であり、JT-H246 付属資料 E にて例が示されるように、公衆陸上移動網 (PLMN) における相互接続を可能にする。エンドポイントは、ゲートキーパーに ISUP アドレスシグナルコードを登録する際、**isupNumber** フィールドを使用しなければならない。

**AddressPattern** 構造は、**AliasAddress** のワイルドカード記述、または **PartyNumber** の範囲の指定を可能にする。**wildcard** フィールドは、**AliasAddress** 構造にて可能なワイルドカード拡張を記述する。ダイヤルされたディジットまたは E.164 番号については、この拡張は番号の最後に対して可能である。電子メールアドレスについては、その拡張は番号の最初に対して可能である。例えば、ワイルドカードが "+1 303" である場合、そのパターンはデンバーのエリアコード内におけるあらゆる番号を記述することが可能である。**AddressPattern** 構造の **range** フィールドは、最初と最後が指定されたアドレスの範囲を記述する。

エンドポイントがアドレスタイプを決定するために使用するメカニズムは実装の問題として残されている。メッセージ上での種々のアドレスタイプ表記を表 19/JT-H225.0 に示す。エンドポイントがアドレスタイプやその適用範囲を知らない場合、アドレスタイプは JT-H225.0 呼制御メッセージでは私設網 (Private)、不定 (Unknown)、RAS メッセージでは **dialedDigits AliasAddress** と表記すべきであることに注意。

表 19/JT-H255.0 番号体系と表記の対応  
(ITU-T H.225.0)

番号体系	JT-Q931 表記	JT-H225.0 情報要素表記	JT-H225.0 UUIE 表記
不定 (デフォルトおよび第 1 版相互運用モード)	私設網番号計画、番号種別 = 不定 ('000') (注 1)	私設網番号計画、番号種別 = 不定 ('000') (注 1)	dialedDigits AliasAddress (注 2)
私設網、不定	私設網番号計画、番号種別 = 不定 ('000') (注 1)	私設網番号計画、番号種別 = 不定 ('000') (注 1)	dialedDigits AliasAddress (注 2)
私設網、レベル 2 地域番号	私設網番号計画、番号種別 = レベル 2 地域番号 ('001')	私設網番号計画、番号種別 = 不定 ('000') (注 1)	PartyNumber AliasAddress の privateNumber、TypeOfNumber = level2RegionalNumber
私設網、レベル 1 地域番号	私設網番号計画、番号種別 = レベル 1 地域番号 ('010')	私設網番号計画、番号種別 = 不定 ('000') (注 1)	PartyNumber AliasAddress の privateNumber、TypeOfNumber = level1RegionalNumber
私設網、PISN 特有番号	私設網番号計画、番号種別 = PISN 特有番号 ('011')	私設網番号計画、番号種別 = 不定 ('000') (注 1)	PartyNumber AliasAddress の privateNumber、TypeOfNumber = pISNSpecificNumber
私設網、レベル 0 地域番号 (ローカル)	私設網番号計画、番号種別 = レベル 0 地域番号 ('100')	私設網番号計画、番号種別 = 不定 ('000') (注 1)	PartyNumber AliasAddress の privateNumber、TypeOfNumber = localNumber
E.164 公衆網番号、不定	ISDN/電話番号計画、番号種別 = 不定 ('000')	ISDN/電話番号計画、番号種別 = 不定 ('000')	PartyNumber AliasAddress の e164Number、TypeOfNumber = Unknown
E.164 公衆網番号、国際番号	ISDN/電話番号計画、番号種別 = 国際番号 ('001')	ISDN/電話番号計画、番号種別 = 国際番号 ('001')	PartyNumber AliasAddress の e164Number、TypeOfNumber = internationalNumber
E.164 公衆網番号、国内番号	ISDN/電話番号計画、番号種別 = 国内番号 ('010')	ISDN/電話番号計画、番号種別 = 国内番号 ('010')	PartyNumber AliasAddress の e164Number、TypeOfNumber = nationalNumber
E.164 公衆網番号、網特有番号	ISDN/電話番号計画、番号種別 = 網特有番号 ('011')	ISDN/電話番号計画、番号種別 = 網特有番号 ('011')	PartyNumber AliasAddress の e164Number、TypeOfNumber

	(‘011’)		=networkSpecificNumber
E.164 公衆網番号、市内番号 (subscriber number)	ISDN/電話番号計画、番号種別 = 市内番号 (‘100’)	ISDN/電話番号計画、番号種別 = 市内番号 (‘100’)	PartyNumber AliasAddress の e164Number、TypeOfNumber = subscriberNumber
E.164 公衆網番号、短縮番号	ISDN/電話番号計画、番号種別 = 短縮番号 (‘110’)	ISDN/電話番号計画、番号種別 = 短縮番号 (‘110’)	PartyNumber AliasAddress の e164Number、TypeOfNumber = abbreviatedNumber
注 1 : 番号計画識別子 = 私設網の場合、私設網番号ディジットは番号種別を含む <b>PartyNumber</b> の <b>privateNumber</b> でコード化される。情報要素内の番号種別フィールドは、受信時に無視され、送信時にこの表に従ってコード化されなければならない。 注 2 : privateTypeOfNumber = UnknownPartyNumber AliasAddress の場合は、dialedDigits AliasAddress と同一であるとみなされなければならない。			

**MobileUIM** 構造は、第 2 世代および第 3 世代無線網におけるシステム互換性に関する識別モジュールを表す。可能な選択肢は以下の通りである：

- ansi-41-uim** – アメリカ標準により規定される無線網を示す。
- gsm-uim** – ヨーロッパ標準により規定される無線網を示す。

**ANSI-41-UIM** 構造は、アメリカ標準の無線網とのシステム互換性の識別モジュールを識別する。可能な選択肢は以下の通りである：

- imsi** – 国際移動機識別番号 (International Mobile Station Identification numbers)
- min** – 移動通信識別番号 (Mobile Identification Numbers)
- mdn** – 移動通信ディレクトリ番号 (Mobile Directory Numbers)
- msisdn** – 移動機 ISDN 番号 (Mobile Station ISDN numbers)
- esn** – エレクトロニックシリアル番号 (Electronic Serial Numbers)
- mscid** – 移動通信交換局番号 (Mobile Switching Center numbers) + マーケット識別番号 (Market Identification (numbers)) またはシステム識別番号 (System Identification numbers)
- sid** – システム識別番号 (System Identification numbers)
- mid** – マーケット識別番号 (Market Identification numbers)
- systemMyTypeCode** – ベンダ識別番号 (vendor identification numbers)
- systemAccessType** – システムアクセスタイプ (system access type)
- qualificationInformationCode** – 権限情報コード (qualification information code)
- sesn** – SIM エレクトロニックシリアル番号 (SIM Electronic Serial Numbers)  
(TTC 補足 : SIM = 加入者情報モジュール (Subscriber Identity Module) )
- soc** – システムオペレータコード (System Operator Codes)

**GSM-UIM** 構造は、ヨーロッパ標準の無線網とのシステム互換性の識別モジュールを識別する。可能な選択肢は以下の通りである：

- imsi** – 国際移動機識別番号 (International Mobile Station Identification (numbers))
- tmsi** – 一時的な移動機識別番号 (Temporary Mobile Station Identification (numbers))
- msisdn** – 移動機 ISDN 番号 (Mobile Station ISDN numbers)
- imei** – 国際移動通信装置識別番号 (International Mobile Equipment Identification numbers)
- hplmn** – ホーム公衆陸上移動網番号 (Home Public Land Mobile Network Numbers)

**vplmn** – 在圏公衆陸上移動網番号（Visiting Public Land Mobile Network Numbers）

**IsupNumber** 構造は、TTC 標準 JT-Q763 で規定されている ISUP アドレスシグナルコードを表現する。可能な選択肢は以下の通りである：

- e164Number** – ITU-T 勧告 E.163 や E.164 に従う番号計画を使用するアドレスシグナルコード
- dataPartyNumber** – 現在未使用
- telexPartyNumber** – 現在未使用
- privateNumber** – ISO/IEC 11571 に従う番号計画を使用するアドレスシグナルコード
- nationalStandardPartyNumber** – 現在未使用

**IsupPublicPartyNumber** 構造は、公衆網番号計画に従う ISUP アドレスシグナルコードを表現する：

- natureOfAddress** – この番号で使用する番号体系
- address** – これは番号の数値を運ぶ

**IsupPrivatePartyNumber** 構造は、私設網番号計画に従う ISUP アドレスシグナルコードを表現する：

- privateTypeOfAddress** – この番号で使用する番号体系
- address** – このフィールドは番号の数値を運ぶ

**NatureOfAddress** 構造は、ISUP 番号で使用する番号体系を表現する。番号体系の選択は、TTC 標準 JT-Q762 で規定されている Nature Of Address (NOA) 識別子に従う。可能な選択肢は以下の通りである：

- unknown** – 規定の番号体系でないか、番号体系が不明
- subscriberNumber** – TTC 標準 JT-Q763 で規定されている加入者番号
- nationalNumber** – TTC 標準 JT-Q763 で規定されている国内番号
- internationalNumber** – TTC 標準 JT-Q763 で規定されている国際番号
- networkSpecificNumber** – TTC 標準 JT-Q763 で規定されている網特有番号
- routingNumberNationalFormat** – TTC 標準 JT-Q769.1 で規定されている国内番号フォーマットによるネットワークルーティング番号
- routingNumberNetworkSpecificFormat** – TTC 標準 JT-Q769.1 で規定されている網特有番号フォーマットによるネットワークルーティング番号
- routingNumberWithCalledDirectoryNumber** – TTC 標準 JT-Q769.1 で規定されている着ディレクトリ番号に連結されたネットワークルーティング番号

ISUP アドレスコードと IsupNumber の JT-H225.0 アドレスディジットの対応は表 20/JT-H225.0 の通りである：

表20/JT-H225.0 – 番号表記の対応関係

ISUPコード	ISUPアドレスシグナル	JT-H225.0 IsupNumber digits
0 0 0 0	digit 0	0
0 0 0 1	digit 1	1
0 0 1 0	digit 2	2
0 0 1 1	digit 3	3
0 1 0 0	digit 4	4
0 1 0 1	digit 5	5
0 1 1 0	digit 6	6
0 1 1 1	digit 6	7

表20/JT-H225.0 – 番号表記の対応関係

1 0 0 0	digit 8	8
1 0 0 1	digit 9	9
1 0 1 0	Spare	A
1 0 1 1	code 11	B
1 1 0 0	code 12	C
1 1 0 1	Spare	D
1 1 1 0	Spare	E
注 – 値`1111`は、H.246 Annex CのQ.931 Sending Complete IEに対応する。		

**ExtendedAliasAddress** 構造は、別名アドレスと一般的な情報を関連付ける方法を提供する。**presentationIndicator** は、**address** が公開されるべきか、或は制限されるべきかを示す。**screeningIndicator** は、**address** がエンドポイントとネットワークのどちらにより提供されるか、およびネットワークにより隠蔽されているかどうかを示す。

**Endpoint** 構造は、エンドポイントのバックアップ、冗長性、または選択情報を示すために使用される：

- nonStandardData** – 本標準の規定範囲外の情報を送る。(例えば、独自データ)
- aliasAddress** – 別名アドレスのリスト。他のエンドポイントはこの情報を参照してこのエンドポイントを識別する。
- callSignalAddress** – このエンドポイントの呼制御トランスポートアドレス。
- rasAddress** – このエンドポイントの RAS トランスポートアドレス。
- endpointType** – エンドポイントのタイプを規定する。
- tokens** – このエンドポイント (すなわち、この **Endpoint** 構造で記述されるエンドポイント) と関係するトークン。
- cryptoTokens** – このエンドポイント (すなわち、この **Endpoint** 構造で記述されるエンドポイント) と関係する暗号化されたトークン (**cryptoTokens**) 。
- priority** – エンドポイントの SEQUENCE が示されている場合に使用される。小さい **priority** 番号を持つエンドポイントは、大きな **priority** 番号を持つエンドポイントよりも優先される。**priority** 番号を持たないエンドポイントは 0 (最優先) の **priority** を持つとみなされる。
- remoteExtensionAddress** – 複数のゲートウェイを経由するために必要な場合に、エンドポイントの別名アドレスが含まれる。
- destExtraCallInfo** – 複数呼 (multiple calls) のための外部アドレスが含まれる。
- alternateTransportAddresses** – TCP 以外のトランスポート方式のサポートを示す。

**AlternateTransportAddresses** 構造は、TCP 以外のトランスポート方式に対し、呼制御アドレスを伝達する。

**UseSpecifiedTransport** 構造は、シグナリングトランスポートプロトコルの選択を定義する。**tcp** の値は、TCP プロトコルを示し、**annexE** の値は JT-H323 付属資料 E により規定されるプロトコルを示し、**setp** の値はストリーム制御伝送プロトコル (SCTP) の使用を示す。

**AlternateGK** 構造は、代替の、あるいはバックアップのためのゲートキーパーまたは割り当てゲートキーパーのリストを示すために使用される：

- rasAddress** – RAS シグナリングに使用されるトランスポートアドレス。
- gatekeeperIdentifier** – バックアップあるいは代替ゲートキーパーを識別するためにオプションとして含まれる。これが提供される場合は、それ以降にバックアップゲートキーパーへ送信されるメッセージはこれを含まなければならない。
- needToRegister** – エンドポイントが、他の RAS メッセージを送出する前に、代替ゲートキーパーに登録をしなければならない場合、それを示すために TRUE にセットされる。
- priority** – 代替ゲートキーパーのバックアップや選択の優先度を示す。小さな番号ほど高い優先度であることを意味する。

**AltGKInfo** 構造は、代替ゲートキーパーについての情報を提供するために使用される：

- alternateGatekeeper** – 優先度をつけられた代替ゲートキーパーのシーケンス。
- altGKisPermanent** – それ以降の全ての RAS 信号が **alternateGatekeeper** フィールド内でリストされるゲートキーパーへリダイレクトされなければならないことを示すために TRUE に、拒



否を引き起こしたメッセージのみダイレクトされなければならないことを示すために FALSE にセットされる。このフラグは、**alternateGeetkeeper** フィールドの **needToRegister** フラグが TRUE にセットされている場合、TRUE にセットされなければならない。

**QseriesOptions** 構造は、ゲートキーパーまたは端末が提供するサポートに関連する他のエンドポイントにオプションの Q シリーズプロトコルのための情報を提供する。これは、ARQ、呼設定、および GRQ メッセージで使用される。Q シリーズオプションの使用は今後の検討として保留され、まだ定義されていない。

**GloballyUniqueID** と **ConferenceIdentifier** は、グローバルでユニークな識別子 (**GloballyUniqueID**) を意味しており、その取扱いは JT-H323 で述べられている。**GloballyUniqueID** はオクテット 0 が最初に符号化される。**GloballyUniqueID** は表 21/JT-H225.0 にしたがって構成される。

表 21/JT-H225.0 **GloballyUniqueID** フォーマット  
(ITU-T H.225.0)

フィールド	データタイプ	オクテット番号	内容
time_low	符号なし 32 ビット整数	0-3	タイムスタンプの下位フィールド
time_mid	符号なし 16 ビット整数	4-5	タイムスタンプの中位フィールド
time_hi_and_version	符号なし 16 ビット整数	6-7	タイムスタンプの上位フィールドとバージョン番号
clock_seq_hi_and_reserved	符号なし 8 ビット整数	8	クロックシーケンスの上位フィールドと種別情報
clock_seq_low	符号なし 8 ビット整数	9	クロックシーケンスの下位フィールド
node	符号なし 8 ビット整数	10-15	全体で一意的なノード識別子

**GloballyUniqueID** は、1 つの 16 オクテットのレコードにより構成され、フィールド間にパディングを含んではならない。全体のサイズは 128 ビットである。

オクテット単位のビット割り付けの際に混乱を最小限にするために、**GloballyUniqueID** レコードは整数個のオクテットから成るフィールドのみで定義されている。バージョン番号はタイムスタンプのフィールド (*time\_high*) に多重化され、種別情報はクロックシーケンスのフィールド (*clock\_seq\_high*) に多重化される。

タイムスタンプは、協定世界時 (UTC) で表される 60 ビットの値である。協定世界時は、1582 年 10 月 15 日 00:00:00:00 (キリスト教がグレゴリオ暦を採用した日) から 100 ナノ秒間隔でカウントした数である。

バージョン番号は、*time\_hi\_and\_version* フィールドの上位 4 ビットに割り当てられており、1 にセットされる (2 進数で '0001')

種別フィールドは、**GloballyUniqueID** のレイアウトを決定する。DCE **GloballyUniqueID** の構成は異種バージョン間で固定される。他の種別の **GloballyUniqueID** は DCE **GloballyUniqueID** と相互接続できなくともよい。**GloballyUniqueID** の相互接続性は、異種システム間の文字列変換、比較、および辞書順並べ替えのような操作の適用可能性として定義される。種別フィールドは *clock\_seq\_hi\_and\_reserved* フィールドの最上位ビットからの可変長ビット列として割り当てられる。

表 22/JT-H.225.0 に DCE 種別フィールドの構成を示す。

表 22/JT-H225.0 DCE 種別フィールドの構成  
(ITU-T H.225.0)

msb1	msb2	msb3	説明
0	-	-	予約済み, NCS 下位互換
1	0	-	DCE 種別
1	1	0	予約済み, Microsoft Corporation GUID.
1	1	1	拡張定義のため予約済み

クロックシーケンスには、クロックに狂いが生じたことを検出する能力が要求される。クロックシーケンスは、*clock\_seq\_hi\_and\_reserved* フィールドの下位 6 ビットと *clock\_seq\_low* フィールドに符号化される。

*node* フィールドは、IEEE アドレス（通常はホストアドレス）から成る。複数の IEEE 802 ノードを含むシステムのために、利用可能なあらゆるノードアドレスが利用可能になっている。最下位に位置するオクテット（第 10 オクテット）は、*global/local* ビットと *unicast/multicast* ビットを含み、802.3 パケットネットワークのアドレスの最初のオクテットである。

クロックシーケンス値は、以下の場合すぐに更新されなければならない。

- **GloballyUniqueID** 生成者が、UTC のローカル値の遅れを検出した場合。これは DCE タイムサービスの通常の機能によってもよい。
- **GloballyUniqueID** 生成者が、最後に参照した UTC 値の状態を消失した場合。これは時間に遅れを生じた可能性があることを示す。リポートの際の典型的な現象である。

ノードが操作可能である間は、**GloballyUniqueID** 生成者は、**GloballyUniqueID** を生成するために、常に最新の UTC を保存している。新しい **GloballyUniqueID** が生成されるたびに、現在の UTC が保存している値と比較される。比較の結果、現在値の方が小さい（クロックに狂いを生じた場合）、あるいは保存していた値が失われた場合、クロックシーケンスはモジュロ 16,384 でインクリメントされ、重複 **GloballyUniqueID** の発生を防止する。

クロックシーケンスは、システムを通じての相関を最小にするために、乱数値で初期化されなければならない。

**GloballyUniqueID** は、以下のアルゴリズムによって生成される：

- 1) **GloballyUniqueID** に使用するクロックシーケンスと UTC ベースのタイムスタンプの値を決定する。2) *time\_low* フィールドに、タイムスタンプの下位 32 ビット（ビット番号 0~31）をビット順を保持して設定する。
- 3) *time\_mid* に、タイムスタンプの第 32~47 ビットをビット順を保持して設定する。
- 4) *time\_hi\_and\_version* フィールドの最下位 12 ビット（第 0~11 ビット）に、タイムスタンプの第 48~59 ビットをビット順を保持して設定する。
- 5) 表 22/JT-H225.0 に示すように、*time\_hi\_and\_version* フィールドの最上位 4 ビット（第 12~15 ビット）に、生成される **GloballyUniqueID** のバージョンと対応する 4 ビットのバージョン番号を設定する。
- 6) *clock\_seq\_low* フィールドに、クロックシーケンスの最下位 8 ビット（第 0~7 ビット）をビット順を保持して設定する。
- 7) *clock\_seq\_hi\_and\_reserved* フィールドの最下位 6 ビット（第 0~5 ビット）に、クロックシーケンスの最上位 6 ビット（第 8~13 ビット）をビット順を保持して設定する。
- 8) *clock\_seq\_hi\_and\_reserved* フィールドの最上位 2 ビット（第 6,7 ビット）をそれぞれ 0,1 に設定する。
- 9) *node* フィールドに 48 ビットの IEEE アドレスをアドレスとしてのビット順を保持して設定する。

システムが **GloballyUniqueID** を生成したいが IEEE802 準拠のネットワークカードなどの IEEE802 アドレスの生成源を持たない場合、アドレスに使用する代替値を生成するために別の方法を使うべきである。理想的な解決法は、47 ビットの暗号作成品質の乱数値を得てそれをノード ID の最上位 47 ビットとして使用し、更にノード ID の最初のオクテットの最下位ビットを 1 にセットすることである。このビットは *unicast/multicast* ビットであり、ネットワークカードから得られる IEEE 802 アドレスではセットされない。したがって、ネットワークカードを持つマシンが生成した **GloballyUniqueID** と、持たないマシンが生成した **GloballyUniqueID** とが衝突することは起こり得ない。

システムが暗号作成品質の乱数値を生成するための原始関数を持たない場合でも、ほとんどのシステムでは、通常、そのような乱数値を得るために、適度に大きな数の乱数生成源が利用可能である。そのような乱数生成源はシステムに依存するが、しばしば使用中のメモリのパーセンテージ、メインメモリのサイズのバイト表示、メインメモリの空き容量のバイト表示、ページングまたはスワップファイルのサイズのバイト表示、ページングまたはスワップファイルの空きバイト数、ユーザ仮想アドレス空間の合計サイズのバイト表示、利用可能な全ユーザ仮想アドレス空間のバイト数、ブートディスクドライブのサイズのバイト表示、ブートディスクドライブの空き領域のバイト数、現在時刻、システムブート後の経過時間、種々のシステムディレクトリのファイルの個々のサイズ、などを含む。

人が読めるテキストに使用するために、**GloballyUniqueID** 文字列表現はシングルダッシュで分けられたフィールドのシーケンスとして規定される。

個々のフィールドは、整数として取り扱われ、最上位ディジットを先頭にしてゼロの桁を'0'で埋めた 16 進数列としてプリントされた値を持つ。16 進値に含まれる a~f は小文字で出力され、入力時には大文字/小文字は区別されない。シーケンスは **GloballyUniqueID constructed type** と同じである。

**GloballyUniqueID** 文字列表現の正式な定義は以下の拡張 BNF で与えられる：

```

UUID                = <time_low> <hyphen> <time_mid> <hyphen>
                    <time_high_and_version> <hyphen>
                    <clock_seq_and_reserved>
                    <clock_seq_low> <hyphen> <node>
time_low            = <hexOctet> <hexOctet> <hexOctet> <hexOctet>
time_mid           = <hexOctet> <hexOctet>
time_high_and_version = <hexOctet> <hexOctet>
clock_seq_and_reserved = <hexOctet>
clock_seq_low      = <hexOctet>
node               = <hexOctet><hexOctet><hexOctet>
                    <hexOctet><hexOctet><hexOctet>
hexOctet           = <hexDigit> <hexDigit>
hexDigit           = <digit> | <a> | <b> | <c> | <d> | <e> | <f>
digit              = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |
                    "8" | "9"
hyphen             = "-"
a                  = "a" | "A"
b                  = "b" | "B"
c                  = "c" | "C"
d                  = "d" | "D"
e                  = "e" | "E"
f                  = "f" | "F"

```

以下に GloballyUniqueID 文字列表現の例を示す。

f81d4fac-7dec-11d0-a765-00a0c91e6bf6

**TimeToLive** は、登録が有効として扱われる秒数である。

**H248PackagesDescriptor** 構造は、ASN.1 PER 符号化された H.248 の **PackagesDescriptor** を含むオクテット列である。

**H248SignalsDescriptor** 構造は、ASN.1 PER 符号化された H.248 の **SignalsDescriptor** を含むオクテット列である。

**FeatureDescriptor** 構造は、一般的に機能を識別する **GenericData** 要素である。

**CircuitInfo** – この構造は SCN 網またはこの呼で使用される網についての情報を提供する。**sourceCircuitID** フィールドは、SCN 上で発呼する時の網の発信元網についての情報を提供する。また入側ゲートウェイによって、ゲートキーパーに発信元網の識別子をレポートするために使用されてもよい。

**destinationCircuitID** は、SCN 上で着呼する時の着信先の網についての情報を提供する。また出側ゲートウェイ上の着信先の網を選択するためにゲートキーパーに使用されてもよい。

**CircuitIdentifier** 構造は、ゲートウェイによってレポートされるため、またはゲートキーパーによる選択のためのファシリティを示す。**CircuitIdentifier** 構造は、様々なインタフェースをサポートする。

**CicInfo** 構造は、SS7 ベアラチャネルを示す。**cic** フィールドは、標準 JT-Q763 に定義されるような網識別コードである。それは最初のオクテットでの最下位ビットと最後のオクテットの最上位ビットでコード化される。**pointCode** フィールドは、標準 JT-Q763 に定義されるようなポイントコードを含む。**pointCode** の最初のオクテットはネットワーク（ネットワーク表示コード）を識別し、残りのオクテットは SS7 ポイントコード値を識別する。**cic** と **pointCode** フィールドは、国内での変更を認めるため可変長である。

**GroupID** 構造は、そのグループ内の物理的なまたは論理的な **group** と、**member**（または **member** のセッ

ト)を識別する。例えば、**member**がそのインタフェース上の特定のDS0を識別する間、**group**は物理的なインタフェースを識別する。**member**フィールドが省かれている場合は、ゲートウェイは特定の**group**内の使用できるファシリティを選択することを要求する。

**CarrierInfo**構造は、キャリア選択に関する情報を含む。**CarrierIdentificationCode**は、(ISUP UAMメッセージにおけるキャリア識別コードのように)加入者によって選択され、もしくは、ルーチングアプリケーションによって決定されたキャリアをディジットのバイナリ列として識別する。**CarrierName**フィールドは、アスキー文字列にてキャリアを識別する別の手段である。

**Carrier** – ルーチングアプリケーションによって決定され、もしくは、加入者によって参照される呼ルーチングのためのキャリア識別子/選択コード

**ServiceControlDescriptor**構造は、サービス仕様データ、またはそれへの参照を含む。それは例えばJT-H323付属資料Kに記述されているように、ユーザプレゼンテーションまたは他のサービス制御通信を目的とする。以下に実行できるオプションを示す：

- url** – この選択はURL基準プロトコルまたはリソースを含む。
- signal** – この選択はJT-H248.1に定義されるようなバイナリ形式の**SignalsDescriptor**を含む。オプションの**streamID**と**notifyCompletion**要素は、**SignalsDescriptor**内の**Signal**シーケンスから省かれなければならない。
- nonStandard** – この選択は本標準の規定範囲外の情報を含む。(例えば独自データ)
- callCreditServiceControl** – この選択は、呼の持続時間を制御し、ユーザに勘定残高を通知することに関わる情報を含む。

**ServiceControlSession**構造は、例えば付属資料K/JT-H323に記述されるようなサービス制御セッションの記述を含む。それは以下に示すフィールドを含む：

- sessionId** – このセッションを識別する整数。クライアントに対してユニークである。違うシグナリングパスを通して受信した識別子(例えば、RASと呼制御)は相互に識別可能であり、重複してもよいことに注意すること。
- contents** – 適切なコンテンツを持つ**ServiceControl**構造、または通信手段。
- reason** – 新しいセッション(**open**)なのか、存在するセッションへの変更(**refresh**)なのか、または提供者および存在するリソース(例えば、終了されるべきGUI等のような)によって終了させられるセッション(**close**)なのかどうかを表す。

**RasUsageInfoTypes**構造は、エンドポイントによってゲートキーパーにレポートされてもよい使用量情報のタイプをリストアップする。エンドポイントは集金と使用量情報の報告に関してその能力を示すためのこの構造を使用し、そしてゲートキーパーは特定のタイプの使用量情報を要求するためこの構造を使用する。**nonStandardUsageTypes**フィールドは、ベンダーが独自の使用量情報のタイプを参照することを可能にする。**startTime**と**endTime**フィールドは、呼が開始された時点、終了した時点を個々に参照する。**terminationCause**パラメータは、呼が終了した理由を参照する。

**RasUsageSpecification**構造は、ゲートキーパーが呼における特定ポイントで使用量情報の特定のタイプを要求することを可能にするテンプレートである。**when**フィールドは、エンドポイントが使用量情報を報告することを要求する時のポイントまたはその呼におけるポイントを示す。その情報とは、呼の開始を参照する**start**、呼の終了を参照する**end**、要求していないIRRメッセージを参照する**inIrr**である。**callStartingPoint**フィールドは、使用量情報を報告する目的として呼の開始を認められなければならないポイントまたはその呼におけるポイントを示す。その情報とは、呼の開始を参照する**start**、呼の終了を参照する**end**、自発型IRRメッセージを参照する**inIrr**である。接続メッセージの送信または受信を参照する**connect**の値、そして警報メッセージの送信または受信を参照する**alerting**である。**required**フィールドは、エンドポイントが報告することを要求された使用量情報のタイプを示す。**when**または**required**のいずれかでどちらも選択されていない**RasUsageSpecification**構造は、使用量情報を報告することを無効にする要求を示す。

**RasUsageInformation**構造は、特定の呼に関する使用データの収集である。**nonStandardUsageFields**フィールドは、ベンダーが特定のタイプの使用量情報をリストアップすることを可能にする。**alertingTime**フィールドは、警報メッセージが送信または受信された時の時間を示す。**connectTime**フィールドは、接続メッセージが送信または受信された時の時間を示す。**endTime**フィールドは、解放完了メッセージが送信または受信された時の時間を示す。

**CallTerminationCause**構造は、呼の終了の理由を示す。**releaseCompleteReason**フィールドは、解放完了メッセージ中に規定された**reason**を示す。**releaseCompleteCauseIE**フィールドは、解放完了メッセージからの理由表示情報要素(Cause IE)を提供する。

**BandwidthDetails** 構造は、**BandWidth** 構造において利用可能ではない追加の帯域幅使用量情報を定義する。そのメッセージがストリームの送信側で送られた場合、**sender** フィールドは TRUE に設定され、受信側で送られた場合、FALSE に設定される。ストリームがマルチキャストである場合、**multicast** フィールドは TRUE に設定され、その他の場合、FALSE に設定される。**bandwidth** フィールドは、100bit/s を単位とするストリームのために使用される帯域幅を示す。**rtcpAddresses** フィールドは、メディアストリームのために使われる RTCP アドレスを示す。

**CallCreditCapability** 構造は、呼の料金請求に関連する、確定したエンドポイントの能力を示す。デフォルトでは、エンドポイントはこれらのオプション能力を持たないとみなされる。この構造にフィールドが含まれない場合、このことはそのフィールドで表された能力の状態が最後に報告された時以降、変わらなかったということを示す。**canDisplayAmountString** フィールドは、そのエンドポイントが、ユーザーのアカウントに金額を含むテキストストリングを表示し得るかどうかを示す。**canEnforceDurationLimit** フィールドは、エンドポイントがゲートキーパーによって示された呼時間制限が経過した時、呼を解放する能力を持っているかどうかを示す。

**CallCreditServiceControl** 構造は、ゲートキーパーが、確定した料金請求関連の制御、および情報をエンドポイントに提供することを可能にする。この構造は、以下に示すフィールドを提供する：

- amountString** — このフィールドは、ユーザーアカウントにおける総金額を示す。例「10.00 ドル」。そのストリングは、適切な通貨記号を含むことになっている。U.S.ドルの「USD」のような通貨の標準の省略形は ISO 4217 によって定義されることに注意すること。**amountString** フィールドは、Basic ISO/IEC 10646 (ユニコード) においてコード化されなければならない。
- billingMode** — このフィールドは、この呼のために料金請求モードを示す。**debit** のモードは、その呼がユーザーのアカウントにおける利用可能な総金額に対して課金されることを示している。**credit** のモードは、その呼が後での支払いのために課金されることを示している。エンドポイントは、例えばプレーしたり、もしくは表示するためのアナウンスメントのタイプを決定するためにこの情報を使用することができる。
- callDurationLimit** — このフィールドは、特別な呼に許された時間残量を示す。
- enforceCallDurationLimit** — このフィールドは、そのエンドポイントが **callDurationLimit** によって示された時間が経過した後で、呼を解放するように要求されるかどうかを示す。このフィールドが提供されない場合、そのエンドポイントは、その指令が以前の状態から変わらなかったことを示すと解釈しなければならない。
- callStartingPoint** — このフィールドは、呼時間強制がエンドポイントによって提供される場合、開始を要求されるタイミングを呼の中のポイントに示す。

**GenericData** 構造は、実際のパラメータを伝達するためのデータ、及び **parameters** フィールドを識別するために、**id** で構成される。

**GenericIdentifier** 構造は、オブジェクトを確認する様々な方法を提供する。

**EnumeratedParameter** 構造は、一般的パラメータを提供する。それは、パラメータとあらゆる関連するデータを伝達するための **content** フィールドを認識するための **id** で構成される。

**Content** 構造は、いくつかの異なるデータタイプをサポートする。それは **raw**、**text**、**unicode**、**bool**、**number8**、**number16**、**number32**、**id**、**alias**、**transport**、**compound**、**nested** を含む。これは、一般的パラメータの柔軟な定義を許している。**raw** の選択は、実際のデータ構造が他で定義されるパラメータ、またはパラメータのセットを許している。例えば、PER コード化された ASN.1、または **type-length-value** 形式のデータで構成されてもよく、他のシグナリングプロトコルのメッセージでカプセル化されてもよい。

**FeatureSet** 構造は、エンティティが一般的な機能情報を規定することを可能にする。そのエンティティは、**neededFeatures** フィールドを使用して呼の成立のために必要とする機能のセットを、**desiredFeature** フィールドを使用して、望んでいるが必要とされない機能のセットを、**supportedFeatures** フィールドでサポートする機能のセットを規定している。**replacementFeatureSet** ブールは、この機能セットが以前に送られた機能セットを置き換えることを示す場合に TRUE に設定され、その他の場合には FALSE に設定される。

**TransportChannelInfo** 構造は、メディアトランスポートチャンネルに関する情報を提供する。**sendAddress** フィールドは、送信側のトランスポートアドレスであり、**recvAddress** は、受信側のトランスポートアドレスである。

**RTPSession** 構造は、RTP セッションの記述を提供する。それには、以下に示すフィールドがある：

- rtpAddress** - このフィールドは、RTP ストリームの送信アドレス、受信アドレスを提供する。
- rtcpAddress** - このフィールドは、RTCP メッセージの送信アドレス、受信アドレスを提供する。
- cname** - このフィールドは、6 章、及び本標準の付属資料 A で規定される CNAME を提供する。
- ssrc** - このフィールドは、6 章、及び本標準の付属資料 A で記述される RTP ストリームのソースを識別するために使用される。
- sessionId** - このフィールドは、JT-H245 で記述される RTP セッションの識別子を提供する。
- associatedSessionIds** - このフィールドは、JT-H245 で記述される関連する RTP セッションの識別子を提供する。
- multicast** - このフィールドは、これがマルチキャストセッションであるかどうかを示す。
- bandwidth** - このフィールドは、100bit/s を単位でストリームに使用される帯域幅を示す。

**RehomngModel** 構造は、エンドポイントが割り当てゲートキーパーに対し識別や再登録するのに使用するモデルを指定するのに使用する。以下に実行できるオプションを示す：

- gatekeeperBased** - エンドポイントは、ゲートキーパーに再登録を指示されれば、割り当てゲートキーパーに再登録する。
- endpointBased** - エンドポイントは、割り当てゲートキーパーをポーリングし、そのゲートキーパーが応答すれば再登録する。

**TransportQoS** 構造は、エンドポイントがサポートする資源予約能力を指定するのに使用する。以下に実行できるオプションを示す：

- endpointControlled** - エンドポイントは、自身の予約メカニズムを適用する。
- gatekeeperControlled** - ゲートキーパーは、エンドポイントに代わり資源予約を行う。
- noControl** - 資源予約を必要としない。
- qOSCapabilities** - エンドポイントの QoS 能力は QoS Capability 構造内のフィールドに記載される。

#### 7.7 サポート必須のRASメッセージ

各エンドポイントタイプがサポートする RAS メッセージを表 23/JT-H225.0 に示す。

**表 23/JT-H225.0 RAS メッセージのステータス  
(ITU-T H.225.0)**

RAS メッセージ	エンドポイント (送信側)	エンドポイント (受信側)	ゲートキーパー (送信側)	ゲートキーパー (受信側)
GRQ	O			M
GCF		O	M	
GRJ		O	M	
RRQ	M			M
RCF		M	M	
RRJ		M	M	
URQ	O	M	O	M
UCF	M	O	M	O
URJ	O	O	M	O
ARQ	M			M
ACF		M	M	
ARJ		M	M	
BRQ	M	M	O	M
BCF	M (注 1)	M	M	O
BRJ	M	M	M	O
IRQ		M	M	
IRR	M			M
IACK		O	CM	
INAK		O	CM	

DRQ	M	M	O	M
DCF	M	M	M	M
DRJ	M (注2)	M	M	M
LRQ	O		O	M
LCF		O	M	O
LRJ		O	M	O
NSM	O	O	O	O
XRS	M	M	M	M
RIP	CM	M	CM	M
RAI	O			M
RAC		O	M	
SCI	O	O	O	O
SCR	O	O	O	O

M=必須, O=オプション, F=禁止, CM=条件により必須, 空白は「適用不可」を示す。

注 1: ゲートキーパーが低レートを要求する BRQ を送信する場合は、エンドポイントは低レートをサポートしているなら、BCF で返信しなければならない。そうでなければ BRJ を返信しなければならない。ゲートキーパーが高レートを要求する BRQ を送信する場合は、エンドポイントは BCF または BRJ で返信しなければならない。

注 2: 端末は、ゲートキーパーからの有効な DRQ に対する応答として DRJ を送信してはならない。

## 7.8 端末とゲートキーパーの検出メッセージ

GRQ メッセージは、それを受信したすべてのゲートキーパーが GCF を返すこと要求するものである。GCF は登録要求の許可を示すメッセージである。GRJ は、この要求の拒否であり、エンドポイントは別のゲートキーパーを探すべきであることを示す。

### 7.8.1 ゲートキーパー要求 (GRQ)

GRQ は論理エンドポイントごとに送られる。従って、MCU またはゲートウェイは多くの GRQ を送信する可能性があることに注意すること。

GRQ メッセージは、以下を含む：

**requestSeqNum** – 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全てのメッセージにおいて、この番号を返さなくてはならない。

**protocolIdentifier** – 送信側エンドポイントの JT-H225.0 の版を識別する。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**endpointType** – 登録するエンドポイントのタイプを記述する。(MC ビットは単独でセットしてはならない。)

**rasAddress** – このエンドポイントが、登録および状態表示メッセージのために使うトランスポートアドレス。ゲートキーパーは **rasAddress** がデコードできない場合を除いて、RAS メッセージを(エンドポイントの)メッセージの送り出し元ではなく、この **rasAddress** に送らなければならない。

**gatekeeperIdentifier** – 端末が登録の許可を受けることを望むゲートキーパーを識別するための文字列。**gatekeeperIdentifier** がない、または null 文字列の場合は、端末は、利用可能なゲートキーパーであれば(登録先は) どれでもよいことを示す。

**callServices** – ゲートキーパーや着信端末に対して、オプションの Q シリーズプロトコルをサポートするための情報を提供する。

**endpointAlias** – 別名アドレスのリストで、他の端末はこれによりこの端末を識別してよい。

**alternateEndpoints** – **rasAddress**、**endpointType** または **endpointAlias** のための、優先順位付けされた代替エンドポイントの系列。

**tokens** – これは、動作を許可するために必要になりうる、いくつかのこのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**authenticationCapability** – エンドポイントによりサポートされる認証メカニズムを示す。

**algorithmOID** – エンドポイントによりサポートされる暗号化アルゴリズムの完全なセットを示す。

**integrity** – 受信側に対し、RAS メッセージにおいて、どの完全性のメカニズムが適応されるかを示す。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**supportsAltGK** – そのエンドポイントが、代替ゲートキーパーのメカニズムをサポートしているかどうかを示す。

**featureSet** – このフィールドは、一般的な機能セットを記述する。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**supportsAssignedGK** – エンドポイントが割り当てゲートキーパーメカニズムをサポートしているかどうかを示す。

**assignedGatekeeper** – このフィールドは、エンドポイントに対し、現在の割り当てゲートキーパーを示す。

### 7.8.2 ゲートキーパー確認 (GCF)

GCF メッセージは、以下を含む：

**requestSeqNum** – この値は、GRQ で送信された値と同じ値でなければならない。

**protocolIdentifier** – 受け入れるゲートキーパーの版を識別する。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**gatekeeperIdentifier** – GCF を送信しているゲートキーパーを識別するための文字列。

**rasAddress** – ゲートキーパーが、登録および状態表示メッセージのために使うトランスポートアドレス。

**alternateGatekeeper** – **gatekeeperIdentifier** と **rasAddress** のための優先付けされた代替 (ゲートキーパー) の系列。

**authenticationMode** – これは認証メカニズムが使われることを示す。ゲートキーパーは GRQ に含まれるエンドポイントにより提供される **authenticationCapability** から **authenticationMode** を選択しなければならない。

**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**algorithmOID** – ゲートキーパーが必要とする暗号化アルゴリズムを示す。

**integrity** – 受信側に対し、RAS メッセージにおいて、どの完全性のメカニズムが適応されるかを示す。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**featureSet** – このフィールドは、一般的な機能セットを記述する。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**assignedGatekeeper** – エンドポイントに対する割り当てゲートキーパー。

**rehomeingModel** – 割り当てゲートキーパーに対しエンドポイントが再登録する際に利用されるメカニズムを示す。

### 7.8.3 ゲートキーパー拒否 (GRJ)

GRJ メッセージは、以下を含む：

**RequestSeqNum** – この値は、GRQ で送信された値と同じ値でなければならない。

**ProtocolIdentifier** – 拒否するゲートキーパーの版を識別する。

**NonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**gatekeeperIdentifier** – GRJ を送信しているゲートキーパーを識別するための文字列。

**rejectReason** – このゲートキーパーが GRQ を拒否した理由を示すコード。**genericDataReason** の理由は、要求が一般的要素または機能の結果として拒否されたことを示す。この場合、追加情報が **genericData** フィールドに記されてもよい。

**altGKInfo** – 代替ゲートキーパーに関するオプションの情報。

**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく



完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を

**integrityCheckValue** フィールドに入れて、メッセージを送信する。

**featureSet** – このフィールドは、一般的な機能セットを記述する。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

## 7.9 端末とゲートウェイの登録メッセージ

RRQ は、端末からゲートキーパーへの登録要求である。ゲートキーパーが RCF で応答した場合、端末は、それ以降に呼を設定する際には応答してきたゲートキーパーを使用しなければならない。ゲートキーパーが RRJ で応答した場合は、端末は別の登録可能なゲートキーパーを探さなければならない。

### 7.9.1 登録要求 (RRQ)

RRQ メッセージは、以下を含む：

**requestSeqNum** – 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全ての応答において、この番号を返さなくてはならない。

**protocolIdentifier** – 送信側エンドポイントの JT-H225.0 の版を識別する。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**discoveryComplete** – 要求を行っているエンドポイントがゲートキーパー検出手順をこのメッセージに先行使せた場合、TRUE をセットし、登録のみであれば FALSE をセットする。登録は古くなる可能性があり、このときエンドポイントは、RRQ、ARQ に対してそれぞれ **discoveryRequired** または **notRegistered** の理由コードで失敗することになることに注意すること。このことは、エンドポイントは **discoveryComplete** を TRUE にセットした RRQ を発行する前に(動的または静的いずれかの)検出手順を行うべきであることを示す。

**callSignalAddress** – このエンドポイントの呼制御トランスポートアドレス。複数トランスポートをサポートする場合は、これらは一度に登録されなければならない。

**rasAddress** – このエンドポイントの登録と状態表示用のトランスポートアドレス。ゲートキーパーは **rasAddress** がデコードできない場合を除いて、RAS メッセージを(エンドポイントの)メッセージの送り出し元ではなく、この **rasAddress** に送らなければならない。

**terminalType** – 登録を行うエンドポイントタイプを示す。**mc** ビットは単独でセットしてはならず、**terminal**、**mcu**、**gateway**、**gatekeeper** のいずれかのビットもセットしなければならないことに注意すること。**vendor** 情報が提供される場合には、この情報は **endpointVendor** に含まれるものと一致しなければならない。もし、**terminalType** が **gateway** または **mcu** であるなら、オプションの **supportedPrefixes** 値は、この値により他のエンドポイントがこのエンティティでサポートする SCN プロトコルとデータ速度を識別するかもしれない、プレフィックスアドレスのリストである。このフィールドは **terminalAlias** および **terminalAliasPattern** フィールドに加えて、またはこの代わりに使用してもよい。エンドポイントのサポートするすべてのプレフィックスは **additiveRegistration** オプションが明示されない限り各々の RRQ に含まれなければならない。**additiveRegistration** オプションが明示された場合、ある RRQ 内のサポートされるプレフィックスはそのエンドポイントについて現在登録されているプレフィックスに追加しなければならない。追加的 RRQ では、このエンドポイントに既に登録されているサポートされるプレフィックスは依然として登録されていると考えなければならない。プレフィックスは(E.164 または他の)**PartyNumber** の一部ではないことに注意すること。

**PartyNumber**(またはそれらの範囲やパターン)を登録するには、エンドポイントは以下に示すように **terminalAlias** および **terminalAliasPattern** フィールドを使わなければならない。

**terminalAlias** – このオプション値は、他の端末が、これを使い、この端末を識別するかもしれない別名アドレスのリストである。このフィールドは、**terminalAliasPattern** および **supportedPrefixes** フィールドに加えて、または代わりとして使用してもよい。**terminalAlias** が null の場合には、**terminalAlias** アドレスがゲートキーパーにより割り当てられ、RCF の中に含まれてもよい。もし **terminalAlias** に **dialedDigits**、**partyNumber**、**isupNumber** のいずれのアドレスも含まれていなければ、**dialedDigits**、**partyNumber** または **isupNumber** アドレスがゲートキーパーによって割り当てられ、RCF の中に含まれてもよい。エンドポイントの **email-ID** が利用可能な場合には、それは登録されるべきである。複数の別名アドレスが同じトランスポートアドレスを参照してもよいことに注意すること。登録を希望する全てのエンドポイントの別名は、**additiveRegistration** オプションが明示されない限り、このリストに含まなければならない。**additiveRegistration** オプションが示された場合、ある RRQ 内のエンドポイントの別名は、そのエンドポイントについて現在登録されている別名リストに追加されなくてはならない。

**gatekeeperIdentifier** – 端末が登録を希望するゲートキーパーを識別する文字列。

**endpointVendor** – エンドポイントの提供者に関する情報。

**alternateEndpoints** – **callSignalAddress**、**rasAddress**、**terminalType**、**terminalAlias** のための優先順位付けされた代替エンドポイントの系列。

**timeToLive** – 登録が有効であることを示す、秒単位の時間。この時間が経過した場合、ゲートキーパーは登録が無効であるとみなしてもよい。

**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない、空でなくてはならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**keepAlive** – TRUE にセットされた場合、エンドポイントがこの RRQ を「keepAlive」として送信したことを示す。エンドポイントは **rasAddress**、**keepAlive**、**endpointIdentifier**、**gatekeeperIdentifier**、**tokens**、**timeToLive** のみを含む標準以下の小さな RRQ を送ってもよい。**keepAlive** フィールドが TRUE の RRQ を受信したゲートキーパーは **endpointIdentifier**、**gatekeeperIdentifier**、**tokens**、**timeToLive** 以外のフィールドを無視すべきである。標準以下の小さな RRQ の中の **rasAddress** は、エンドポイントが登録されていない時は、RRJ の宛先としてゲートキーパーによってのみ使用されなくてはならない。

**endpointIdentifier** – 最初の RCF の期間ゲートキーパーにより提供される **endpointIdentifier**

**willSupplyUIEs** – これが TRUE にセットされた場合、エンドポイントは、ゲートキーパーから要求があれば、IRR メッセージの中で JT-Q931 メッセージ情報を提供することを示す。

**maintainConnection** – TRUE の場合、このメッセージの送信側が、現時点で接続上にシグナリングされている呼がない時に、シグナリング接続をサポートできることを示す。

**alternativeTransportAddresses** – このフィールドは TCP 以外の伝送用の呼制御アドレスを伝達する。アドレスを含むことは、対応する伝送方式をサポートすることを示す。

**additiveRegistration** – 存在するとき、このフィールドはメッセージが“追加的な”RRQ であり、エンドポイントが、既存の登録に対して追加的な情報としてこの RRQ を送信したことを意味する。エンドポイントは、**callSignalAddress**、**rasAddress**、**terminalType**、**terminalAlias**、**terminalAliasPattern**、**alternateEndpoints**、**endpointIdentifier**、**gatekeeperIdentifier**、**tokens** のみから構成される追加的な RRQ を送信してもよい。**additiveRegistration** フィールドの存在する RRQ を受信したゲートキーパーは、上記以外のフィールドを無視しなくてはならない。エンドポイントが登録されていない場合、または **terminalAlias** および/または **terminalAliasPattern** がゲートキーパーの登録ポリシーに反するとき、追加的な RRQ 中の **rasAddress** はゲートキーパーにより、それに続く RRJ の宛先として使われなければならない。

**terminalAliasPattern** – このオプション値は、他のエンドポイントが、これを使い、このエンドポイントを識別するかもしれない別名やアドレスを記述するアドレスパターンのリストである。このフィールドは **terminalAlias** および **supportedPrefixes** フィールドに加えて、または代わりに使用してもよい。エンドポイントのすべての別名とアドレスは、**additiveRegistration** オプションが TRUE でない限り、各 RRQ に含まなくてはならない。**additiveRegistration** オプションが TRUE の場合は、RRQ 中のエンドポイントの別名とアドレスは、そのエンドポイントについて現在登録されている別名リストに追加されなくてはならない。

**supportsAltGK** – そのエンドポイントが、代替ゲートキーパーのメカニズムをサポートしているかどうかを示す。

**usageReportingCapability** – このフィールドは、そのエンドポイントがさまざまなタイプの使用量情報を収集・レポートする能力を示すためにエンドポイントにより含まれることがある。

**multipleCalls** – TRUE の場合、このフィールドは、このメッセージの送信側が、単一の呼制御接続上で複数の呼制御ができることを示す。

**supportedH248Packages** – このフィールドはこのエンドポイントがサポートする H.248 パッケージのリストを示す。

**callCreditCapability** – このフィールドはこのエンドポイントについての特定の課金関連能力について記述する。

**capacityReportingCapability** – このフィールドは呼キャパシティ情報のレポートについてのエンドポイントの能力について記述する。

**capacity** – このフィールドはエンドポイントの最大および現在の呼キャパシティを示す。このフィールドを送信する場合、エンドポイントは **maximumCallCapacity** および **currentCallCapacity** 要素を含めなければならない。

**featureSet** – このフィールドは、一般的な機能セットを記述する。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リスト

である。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**restart** – セットされた場合、このフィールドはリブートした後、もしくは、呼喪失の結果、異常イベントの後にエンドポイントによって最初に送信された RRQ であることを示す。これは、ゲートキーパーがクリーンアップ、もしくは、他の必要な機能を実行することを許可する。

**supportsACFSequences** – セットされた場合、このフィールドはエンドポイントがひとつの ARQ メッセージに対する ACF メッセージのシーケンスを受信し、処理することが可能であることを示す。

**supportsAssignedGatekeeper** – エンドポイントが割り当てゲートキーパーメカニズムをサポートしているかどうかを表す。

**assignedGatekeeper** – エンドポイントの現在の割り当てゲートキーパーを表す。

**transportQoS** – エンドポイントは、このフィールドをトランスポート資源を予約するための能力指示に用いることができる。

**language** – ユーザがアナウンスとプロンプトを受信する際に利用したい言語を表す。

## 7.9.2 登録確認 (RCF)

RCF メッセージは下記を含む。

**requestSeqNum** – この値は、RRQ で送信された値と同じ値でなくてはならない。

**protocolIdentifier** – 受け入れるゲートキーパーの版を識別する。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**callSignalAddress** – JT-H225.0 の呼制御メッセージに対するトランスポートアドレスの配列であり、ゲートキーパーが応答するそれぞれのトランスポート毎に 1 つ対応する。このアドレスは TSAP 識別子を含む。

**terminalAlias** – このオプション値は、他の端末が、これを使い、この端末を識別するかもしれない別名アドレスのリストである。このフィールドは、**terminalAliasPattern** および **supportedPrefixes** フィールドに加えて、または代わりとして使用してもよい。ここでは、関連する RRQ メッセージで提示されたものの中から、受け入れられた別名アドレスを記述する。RRQ で提示されたものが無い場合、このリストはゲートキーパーにより割り当てられた別名である。このフィールドが含まれず別名アドレスが RRQ で提示された場合、ゲートキーパーは提示された別名アドレス全てを受け入れる。このフィールドが含まれ RRQ で提示された別名アドレスのサブセットを指定するならば、ゲートキーパーはそのアドレスだけを受け入れる。

**gatekeeperIdentifier** – 端末登録を受け入れたゲートキーパーを識別する文字列。

**endpointIdentifier** – ゲートキーパーによって割り当てられた端末識別子文字列であり、その後の RAS メッセージにエコーされなければならない。

**alternateGatekeeper** – **gatekeeperIdentifier** と **rasAddress** のための優先付けされた代替 (ゲートキーパー) の系列。

**timeToLive** – 登録が有効であることを示す、秒単位の時間。この時間が経過した場合、ゲートキーパーは登録が無効であるとみなしてもよい。

**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済み の完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**willRespondToIRR** – その (IRR の) **needsResponse** フィールドを TRUE にセットして、ゲートキーパーが自発型 IRR メッセージに対して、IACK または INAK メッセージで応答する場合に TRUE となる。

**preGrantedARQ** – ゲートキーパーが許可をあらかじめ承認したイベントを示す。ARQ/ACF 交換以外の手段で許可が保証された環境において、より高速な呼設定時間を可能にする。これらのフィールドが TRUE にセットされていたとしても、エンドポイントはアドレス解釈またはエンドポイントがこの変更されたシグナリングモードをサポートしていないという理由でゲートキーパーに対してなお ARQ を送ることができるという点に注意。**preGrantedARQ** シーケンスが存在しなければ すべてのケースで ARQ シグナリングが使用されなければならない。フィールドは次のとおり。

**makeCall** – **makeCall** フラグが TRUE なら、ゲートキーパーは、エンドポイントが最初に送る ARQ なしに呼を初期化することについて許可をあらかじめ承認する。**makeCall** フラグが FALSE なら、エンドポイントは発呼の許可を得るために常に ARQ を送らなければならない。

**useGKCallSignalAddressToMakeCall** – もし **makeCall** と **useGKCallSignalAddressToMakeCall** フラ

グが両方とも TRUE であり、エンドポイントが発呼するためのゲートキーパーに対する ARQ を送信しないならば、エンドポイントはゲートキーパーの呼制御チャンネルにすべての JT-H225.0 呼制御を送らなければならない。

**answerCall – answerCall** フラグが TRUE の場合、ゲートキーパーは、エンドポイントが最初に ARQ を送ることなく呼に応答する許可をあらかじめ承認する。**answerCall** フラグが FALSE の場合は、エンドポイントは、呼に応答するための許可を得るために、常に ARQ を送信しなければならない。

**useGKCallSignalAddressToAnswer – answerCall** と **useGKCallSignalAddressToAnswer** のフラグが両方とも TRUE であり、エンドポイントが呼に応答するためのゲートキーパーに対する ARQ を送信しないとき、エンドポイントはゲートキーパーから来るすべての JT-H225.0 呼制御を保証しなければならない。もしエンドポイントが応答時にゲートキーパーを使うように指示されており、しかし着呼がゲートキーパーから来るか否か不明な場合（トランスポートアドレスを見ているような場合を含む）エンドポイントは **useGKCallSignalAddressToAnswer** フラグの状態とは関係なく ARQ を発行しなければならない。

**irrFrequencyInCall** – エンドポイントが一つ以上の呼の中にある時、ゲートキーパーに送信される IRR メッセージの頻度を秒単位で示す。エンドポイントがない場合は、ゲートキーパーは自発的な IRR メッセージを必要としない。エンドポイントがこれらの IRR メッセージを送信している時、呼参照値は端末にとって参加要求にて生成されたのと同じでなければならない。しかしこれは”通常”の CRV ではなく、また更なる通信（DRQ、IRQ または BRQ）に再利用されることはできない。呼識別子は、関連する呼にて呼制御チャンネルメッセージで使用されるものと同じでなければならない。

**totalBandwidthRestriction** – このフィールドは、通信中、エンドポイントの帯域の総使用量を制限する。これがない場合、固定帯域制限はなくなる。

**alternateTransportAddresses** – このフィールドは、TCP 以外の伝送用の呼制御アドレスを伝達する。アドレスを含むことは、対応する伝送方式をサポートすることを示す。

**useSpecifiedTransport** – このフィールドは、ゲートキーパーが、エンドポイントに対し呼を生成するために、どのシグナリングトランスポートアドレスを使用するかを指定することを可能とする。このフィールドが存在し、指定されたトランスポートが TCP でなかった場合、**alternateTransportAddresses** も、このメッセージ中に含まなければならない。

**maintainConnection** – TRUE の場合、（ゲートキーパー経由型の場合に）ゲートキーパーが、現時点でコネクション上にシグナリングされている呼がない時に、シグナリング接続をサポートできることを示す。

**serviceControl** – 例えば、JT-H323 付属資料 K に記述されているように、エンドポイントが、ネットワークで呼と関連しないサービス制御通信のために使ってもよいサービス固有のデータまたはアドレスング情報を含む。

**supportsAdditiveRegistration** – 存在する時、このフィールドはゲートキーパーが追加的な登録能力をサポートすることを示す。存在しない時、ゲートキーパーは追加的な登録をサポートしない。

**terminalAliasPattern** – このオプション値は、他のエンドポイントが、これを使い、このエンドポイントを識別するかもしれない別名およびアドレスを記述するアドレスパターンのリストである。このフィールドは **terminalAlias** および **supportedPrefixes** フィールドに加えて、または代わりとして使用してもよい。ここでは、関連する RRQ メッセージで提示されたものの中から、受け入れられた別名やアドレスを記述する。RRQ で提示されたものが無い場合、このリストはゲートキーパーにより割り当てられた別名とアドレスである。このフィールドが含まれずアドレスパターンが RRQ で提示された場合、ゲートキーパーは提示されたアドレスパターン全てを受け入れる。このフィールドが含まれ RRQ で提示されたアドレスパターンのサブセットを指定するならば、ゲートキーパーはそのパターンだけを受け入れる。

**supportedPrefixes** – このオプション値は、他のエンドポイントが、これを使い、このエンドポイントを識別するかもしれないプレフィックスのリストである。このフィールドは、**terminalAlias** および **terminalAliasPattern** フィールドに加えて、またはこの代わりとして使用してもよい。ここでは、関連する RRQ メッセージで提示されたものの中から、受け入れられたアドレスプレフィックスを記述する。RRQ で提示されたものが無い場合、このリストはゲートキーパーにより割り当てられたプレフィックスである。このフィールドが含まれずアドレスプレフィックスが RRQ で提示された場合、ゲートキーパーは提示されたアドレスプレフィックス全てを受け入れる。このフィールドが含まれ RRQ で提示されたアドレスプレフィックスのサブセットを指定するならば、ゲートキーパーはそのプレフィックスだけを受け入れる。

**usageSpec** – このフィールドは、ゲートキーパーが、エンドポイントに対し、指示された時点の、指定された呼の利用情報を収集・レポートすることを要求するために使ってもよい。

**featureServerAlias** – このフィールドはスティミュラスベースプロトコルにおいて ITU-T により将来使われるために予約されている。

**capacityReportingSpec** – このフィールドは、エンドポイントが報告するように求められた呼キャパシティ情

報の種別を示す。

**featureSet** – このフィールドは、一般的な機能セットを記述する。

**genericData** – このフィールドは JT-H225.0 標準外で定義された機能に関する一般要素のリストである。これらのパラメータは、例えば、RAS において透過的にトンネリングされる情報として使われてもよい。

**assignedGatekeeper** – エンドポイントに対する割り当てゲートキーパー

**rehomingModel** – 割り当てゲートキーパーに対し再登録するためにエンドポイントによって利用されるメカニズムを表す。

**transportQoS** – ゲートキーパーはエンドポイントによって利用されるリソース予約メカニズムを示すためにこのフィールドを利用してもよい。

### 7.9.3 登録拒否 (RRJ)

RRJ メッセージは下記を含む。

**requestSeqNum** – この値は、RRQ で送信された値と同じ値でなくてはならない。

**protocolIdentifier** – 拒否するゲートキーパーの版を識別する。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**rejectReason** – 登録拒否の理由。このフィールドは **invalidTerminalAliases** 値を含んでもよい。この場合、このフィールドは関連する RRQ メッセージ中で無効であると決定された別名、アドレス、サポートされるプレフィックスのリストを含む。あらゆる場合において、RRQ に関連付けられた全ての別名、アドレス、サポートされるプレフィックスは、**invalidTerminalAliases** フィールドの指定に従って拒否される。**genericDataReason** という理由は、要求が一般的要素または機能の結果、拒否されたことを示す。この場合、追加情報が **genericData** フィールドに記されてもよい。**registerWithAssignedGK** という理由は、割り当てゲートキーパーが利用可能になったためにリクエストが拒否されたことを示す。エンドポイントは、この理由を受信したときに割り当てゲートキーパーに登録する。

**gatekeeperIdentifier** – 端末の登録を拒否したゲートキーパーを識別するための文字列。

**altGKInfo** – 代替ゲートキーパーに関するオプションの情報。**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**featureSet** – このフィールドは、一般的な機能セットを記述する。

**genericData** – このフィールドは JT-H225.0 標準外で定義された機能に関連する一般要素のリストである。これらのパラメータは、例えば、RAS において透過的にトンネリングされる情報として使われてもよい。

**assignedGatekeeper** – エンドポイントに対する割り当てゲートキーパー

## 7.10 端末/ゲートキーパーの非登録メッセージ

### 7.10.1 非登録要求 (URQ)

URQ は、端末とゲートキーパーの関連を破棄することを要求する。非登録は双方向であることを注意すること。つまり、ゲートキーパーは端末自身が非登録状態であると認識させることができる。そして、端末はゲートキーパーに対して前回の登録が無効であることを通知することができる。

URQ メッセージは、以下を含む：

**requestSeqNum** – 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全ての応答において、この番号を返さなくてはならない。

**callSignalAddress** – 非登録にされようとしているエンドポイントの 1 つ以上のトランスポート呼制御アドレス。

**endpointAlias** – このオプションの値は、これにより他の端末がこの端末を識別してもよい別名アドレスのリストである。このフィールドは **endpointAliasPattern** および **supportedPrefixes** フィールドに加えて、またはこの代わりに使用してもよい。もし、**endpointAliasPattern** フィールドと **supportedPrefixes** フィールドが存在しない場合、全ての別名は単一のメッセージによって非登録とされる。

**dialedDigits** 値は、割り当てられていた場合、必要である。ここにリストされる値だけが非登録にされる。これにより、例えば、**dialedDigits** 値を登録したままにしておく一方、**h323-ID** を非登録にすることが可能となる。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**endpointIdentifier** – 身元の確認で、ゲートキーパーにより送られることはない。

**alternateEndpoints** – **callSignalAddress** または **endpointAlias** についての優先付けされた代替エンドポイント

の系列。

- gatekeeperIdentifier** – 登録時のゲートキーパーからの RCF、または以前の URJ メッセージで受信した **alternateGatekeeper** リストで、エンドポイントが受信した **gatekeeperIdentifier**。
- tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** – 暗号化された **tokens**。
- integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない、空でなくてはならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。
- reason** – ゲートキーパーがエンドポイントが非登録であると判断した理由を示すために URQ を送信するときに使われる。**Maintenance** という理由は、ゲートキーパーまたはエンドポイントがメンテナンスのためにダウンしていることを示す。**registerWithAssignedGK** という理由は、割り当てゲートキーパーが利用可能になったためにエンドポイントが非登録になることを示す。エンドポイントはこの理由を受信した際に割り当てられたゲートキーパーに登録する。
- endpointAliasPattern** – このオプション値は、これにより他のエンドポイントがこのエンドポイントを識別してもよい別名やアドレスを記述するアドレスパターンのリストである。このフィールドは **endpointAlias** や **supportedPrefixes** フィールドに加えて、またはこの代わりに使用してもよい。このフィールド、**endpointAlias** フィールドおよび **supportedPrefixes** フィールドが存在しないとき、すべての別名とアドレスは単一のメッセージによって非登録とされる。それ以外の場合、ここにリストされる値だけが非登録にされる。
- supportedPrefixes** – このオプション値は、他のエンドポイントが、これを使い、このエンドポイントを識別するかもしれないプレフィックスのリストである。このフィールドは、**terminalAlias** および **terminalAliasPattern** フィールドに加えて、または代わりとして使用してもよい。このフィールド、**endpointAlias** フィールドおよび **endpointAliasPattern** フィールドが存在しない場合、全ての別名およびアドレスがただ 1 つのメッセージで非登録となる。それ以外の場合には、ここにリストされた値のみが非登録となる。
- alternateGatekeeper** – **gatekeeperIdentifier** と **rasAddress** のための優先付けされた代替 (ゲートキーパー) の系列。
- genericData** – このフィールドは JT-H225.0 標準外で定義される機能に関する一般要素のリストである。これらのパラメータは、例えば、RAS において透過的にトンネリングされる情報として使われてもよい。
- assignedGatekeeper** – エンドポイントに対する割り当てゲートキーパー

#### 7.10.2 非登録確認 (UCF)

UCF メッセージは、以下を含む：

- requestSeqNum** – この値は、URQ で送信された値と同じ値でなくてはならない。
- nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。
- tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** – 暗号化された **tokens**。
- integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない、空でなくてはならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。
- genericData** – このフィールドは JT-H225.0 標準外で定義される機能に関する一般要素のリストである。これらのパラメータは、例えば、RAS において透過的にトンネリングされる情報として使われてもよい。
- assignedGatekeeper** – エンドポイントに対する割り当てゲートキーパー

#### 7.10.3 非登録拒否 (URJ)

URJ メッセージは、以下を含む：

- requestSeqNum** – この値は、URQ で送信された値と同じ値でなくてはならない。この値は、URQ で送信された値と同じ値でなくてはならない。
- rejectReason** – 非登録の拒否の理由。
- nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。
- altGKInfo** – 代替ゲートキーパーに関するオプションの情報。**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を

**integrityCheckValue** フィールドに入れて、メッセージを送信する。

**genericData** – このフィールドは JT-H225.0 標準外で定義される機能に関する一般要素のリストである。これらのパラメータは、例えば、RAS において透過的にトンネリングされる情報として使われてもよい。

## 7.11 端末からゲートキーパーへの許可メッセージ

ARQ メッセージはエンドポイントがゲートキーパーに対して、パケットネットワークへのアクセスの許可を要求する。ACF は要求に対する許可、ARJ は拒否を示す。

### 7.11.1 参加要求 (ARQ)

ARQ メッセージは、以下を含む：

**requestSeqNum** – 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全ての応答において、この番号を返さなくてはならない。

**callType** – この値を使用することにより、ゲートキーパーは「実際に」使用されている帯域幅の決定を試みることができる。デフォルト値は、全ての呼について **pointToPoint** である。呼のタイプは、呼の最中に動的に変化する可能性があり、また最終的な呼のタイプは ARQ が送信された時点では、不明であると認識されるべきである。

**callModel** – **direct** の場合は、エンドポイントは「端末-端末、直接」の呼モデルを要求している。**gatekeeperRouted** の場合には、エンドポイントは「ゲートキーパー間接」の呼モデルを要求している。ゲートキーパーはこの要求に従う必要はない。

**endpointIdentifier** – RCF によって端末に割り当てられたエンドポイント識別子。

**destinationInfo** – **dialedDigits** や、**PartyNumber** (**e164Number** または **privateNumber**)、もしくは **h323-ID** で表される着信先の別名アドレスの系列である。呼に応じるため ARQ を送信する場合、**destinationInfo** はその呼の着信先（応答するエンドポイント）を示している。少なくとも 1 つの別名がゲートキーパーに登録されており、ARQ 内に複数含まれている別名が同一人物として登録されている場合、ゲートキーパーは ARQ がその人物を指していると認識しなければならない。別名が矛盾している場合、ARQ は **AliasesInconsistent** の理由で拒否されなければならない。もし、ゲートキーパーがこの認識機能を持たない場合には、ゲートキーパーは最初に登録されたアドレスが着信先のアドレスであると考えなくてはならない。

**destCallSignalAddress** – 呼制御のために着信先で使用されるトランスポートアドレス。

**destExtraCallInfo** – 複数の呼のための外部アドレスを含む。

**srcInfo** – 送信元に対する別名アドレスのシーケンスであり、**dialedDigits** や、**PartyNumber** (**e164Number** 又は **privateNumber**) もしくは **h323-ID** のようなもの。呼に応じるため ARQ を送信したとき、**srcInfo** はその呼応答エンドポイントの送信元であることを示している。

**srcCallSignalAddress** – 呼制御のために送信元で使用されるトランスポートアドレス。

**bandWidth** – 100bit/s 単位で表される、呼に対して要求された双方向の帯域幅。例えば、128Kbit/s の呼は、256Kbit/s の要求としてシグナリングされる。この値は、ヘッダとオーバーヘッドを除いたオーディオとビデオのみに関連している。

**callReferenceValue** – この呼に対する JT-H225.0 呼制御メッセージからの CRV。ローカルでのみ有効。特定の呼と ARQ を関連付けるためにゲートキーパーによって使用される。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**callServices** – ゲートキーパーや着呼端末に対して、オプションの Q シリーズプロトコルをサポートするための情報を提供する。

**conferenceID** – ユニークな会議 ID。

**activeMC** – TRUE ならば、発信側はアクティブ MC を持つ。それ以外は FALSE。

**answerCall** – ゲートキーパーに対し、着呼中であることを示すために使用される。**canMapAlias** – TRUE の場合、もし返される ACF が **destinationInfo**、**destExtraCallInfo** および/または **remoteExtensionAddress** フィールドを含んでいるならば、エンドポイントが、この情報を呼設定メッセージの **destinationAddress**、**destExtraCallInfo** および **remoteExtensionAddress** フィールドそれぞれにコピーし、または適切ならば着信側パーティ番号 (Called Party Number) 情報要素にコピーしなければならないことを示している。エンドポイントが、JT-H323 ネットワークから出るのに用いられるゲートウェイである場合、ゲートウェイは、着信先情報を、JT-H323 ネットワーク外で使う適切な番号フォーマット(例えば、DTMF)に変換するであろう。もし、ゲートキーパーが ARQ からのアドレス情報を書き替え、かつ **canMapAlias** が FALSE である場合は、そのゲートキーパーは ARQ を拒否するべきである。JT-H225.0 第 4 版以降の版に準拠するシステムは、このフィールドを

TRUE に設定しなければならない。

**callIdentifier** – RAS シグナリングと、本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、送信側エンドポイントによってセットされたグローバルでユニークな呼識別子。

**srcAlternatives** – **srcInfo**, **srcCallSignalAddress** あるいは **rasAddress** に対する発信元エンドポイントの代替の優先付けされた系列。

**destAlternatives** – **destinationInfo** か **destCallSignalAddress** に対する着信先エンドポイントの代替の優先付けされた系列。

**gatekeeperIdentifier** – 登録時のゲートキーパーからの RCF、または以前の ARJ メッセージで受信した **alternateGatekeeper** リストで、エンドポイントが受信した **gatekeeperIdentifier**。 **tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。 **cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。 **integrityCheckValue** の計算の前は、このフィールドは無視されなければならない、空でなくてはならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**transportQOS** – エンドポイントは、このフィールドをトランスポート資源を予約するための能力指示に用いることができる。

**willSupplyUIIEs** – これが TRUE にセットされた場合、エンドポイントは、ゲートキーパーから要求があれば、IRR メッセージの中で JT-H225.0 呼制御メッセージ情報を提供することを示す。

**callLinkage** – このフィールドの内容は、一般的には呼結合サービスにより制御される。このフィールドの手順と意味は、標準 JT-H323 の 10 章を参照のこと。

**gatewayDataRate** – ゲートウェイを通る SCN 回線側の呼の要求データレート。このデータレートがある場合は、呼設定メッセージの伝達能力情報要素に示されるデータレートと等しくなければならない。ゲートキーパーがこのフィールドを呼の処理でゲートウェイを選択するのに使用するかもしれない。

**capacity** – このフィールドは、ゲートキーパーが ACF を送信することによって ARQ を確認したと想定して、送信側エンドポイントのこの時点で利用可能な呼キャパシティを示す。このフィールドを送信する場合、エンドポイントは **currentCallCapacity** 要素を含まなくてはならない。

**circuitInfo** – このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。

**desiredProtocols** – これは発呼側エンドポイントが、その呼（例えばオーディオ、fax）に望ましいと思うプロトコルタイプの優先順序を示す。解読側のエンティティは、優先順序を考慮し、そのプロトコルをサポートするエンドポイントの位置を示すために、このフィールドを使ってもよい。

**desiredTunnelledProtocol** – このフィールドは、トンネリングが必要であるプロトコルを示す。

**featureSet** – このフィールドは、この呼に関係する一般的な機能セットを記述する。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**canMapSrcAlias** – これが TRUE にセットされた場合、生じる ACF が **modifiedSrcInfo** を含んでいる場合、適切な場合、エンドポイントは呼設定メッセージの **sourceInfo** フィールド、および(または)、Calling Party Number IE にこの情報をコピーすべきであることを示す。ゲートキーパーが ARQ からのアドレッシング情報を交換し、**canMapSrcAlias** が FALSE である場合、ゲートキーパーは ARQ を拒絶すべきである。

注： **destinationInfo** と **destCallSignalAddress** の両方はオプションであるが、エンドポイントが呼に回答しないならば、少なくともどちらかひとつはある必要がある。これはサイト依存であるため、優先すべき絶対的な規則はない、しかし利用可能なならアドレスが使用されるべきである。最善の結果は使用トランスポートプロトコルの特性を考慮することにより得られることに注意すること。

### 7.11.2 許可確認 (ACF)

ACF メッセージは、以下を含む：

**requestSeqNum** – この値は、ARQ で送信された値と同じ値でなくてはならない。

**bandWidth** – 呼に許可された最大の帯域幅。要求した値以下になるかもしれない。

**callModel** – 端末に対し、**destCallSignalAddress** において送られる呼制御が、ゲートキーパーに行くのか端末に行くのかを通知する。 **gatekeeperRouted** は、呼制御がゲートキーパーを通過したことを示し、**direct** は「エンドポイント→エンドポイント」の呼モードが使用されたことを示す。

**destCallSignalAddress** – JT-H225.0 の呼制御を送信するためのトランスポートアドレス。ただし、使用される呼モデルにより、エンドポイントもしくはゲートキーパーアドレスとなりうる。

**irrFrequency** – その呼（保留中も含む）においてエンドポイントがゲートキーパーに IRRs を送る頻度。こ



れない場合、エンドポイントは通信中に IRRs を送らず、ゲートキーパーはエンドポイントを順次呼び出しすることが期待される。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**destinationInfo** – イニシャルチャネルのアドレス。ゲートウェイを介した通信で使用される。

**destExtraCallInfo** – 追加チャネルの呼、すなわち SCN 網側における 2×64kbit/s の呼を可能とするために必要である。 **dialedDigits** または **PartyNumber** アドレスのみを含まねばならず、イニシャルチャネルの番号は含んではならない。

**destinationType** – 着信先エンドポイントのタイプを示す。

**remoteExtensionAddress** – 複数のゲートウェイを越えるために、この情報が必要である場合、着呼側エンドポイントの別名アドレスを含む。

**alternateEndpoints** – **destCallSignalAddress** または **destinationInfo** のための、優先順位付けされた代替エンドポイントの系列。

**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない、空でなくてはならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**transportQOS** – ゲートキーパーはエンドポイントに、どこがリソース予約に関して責任を持つかを知らせてもよい。ゲートキーパーが ARQ において **TransportQOS** を受信した場合は、ACF 中にも **transportQOS** を含まなくてはならない(ゲートキーパーの実装によって修正される可能性がある)。

**willRespondToIRR** – IRR の **needsResponse** フィールドが true にセットされた時、ゲートキーパーが自発型 IRR メッセージに対して、IACK または INAK メッセージで応答する場合に TRUE となる。**uuiisRequested** – エンドポイントが ARQ メッセージにて **willSupplyUUIE** を TRUE にすることで、この能力があることを示しているならば、ゲートキーパーはエンドポイントに対して、エンドポイントが送信するか受信するかした JT-H225.0 呼制御メッセージをゲートキーパーに通知するよう要求してもよい。**uuiisRequested** は、エンドポイントがゲートキーパーに通知しなければならない JT-H225.0 呼制御メッセージのセットを示す。

**language** – ユーザが希望するアナウンスあるいはプロンプトの言語を示す。このフィールドは RFC1766 に準拠した一つあるいは複数の言語タグを含む。

**alternateTransportAddresses** – このフィールドは、TCP 以外の伝送用の呼制御アドレスを伝達する。アドレスを含むことは、対応する伝送方式をサポートすることを示す。

**useSpecifiedTransport** – このフィールドは、ゲートキーパーが、エンドポイントに対し呼を生成するために、どのシグナリングトランスポートアドレスを使用するかを指定することを可能とする。このフィールドが存在し、指定されたトランスポートが TCP でなかった場合、**alternateTransportAddresses** も、このメッセージ中に含まれなければならない。

**circuitInfo** – このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。例えば、これによりゲートキーパーが、その呼のために使われる特定の SCN 網機能を選択するために、外部網発信用ゲートウェイを指定することが可能となる。

**usageSpec** – このフィールドは、ゲートキーパーが、エンドポイントに対し、この呼の指示された時点の、指定された呼の利用情報を収集・レポートすることを要求するために使ってもよい。

**supportedProtocols** – このフィールドは、着信先エンドポイントがサポートするプロトコルを示す。

**serviceControl** – このフィールドは、例えば、JT-H323 付属資料 K に記述されているように、エンドポイントによって使われることが可能なサービス固有のデータやそれに関連するものを含む(例:呼が失敗した時の理由表示のため)。

**multipleCalls** – TRUE の場合、このフィールドは、着信先エンドポイントが単一の呼制御コネクション上で、複数の呼制御ができることを示す。FALSE の場合は、着信先エンドポイントはこの能力を持たない。このフィールドが提供されていない場合、ゲートキーパーは、遠隔のエンドポイントがこの能力を持っているかどうか分からない。

**featureSet** – このフィールドは、この呼に関係する一般的な機能セットを記述する。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**modifiedSrcInfo** – **dialedDigits**、**PartyNumber** (**e164Number** または **privateNumber**)、あるいは **h323-ID** のようなソースとなるエンドポイントのために使用されるべき別名アドレス。このフィールドは発信側エンドポイントの別名アドレスが変換、もしくは変更される場合、もしくは、一次着信先、代替エンドポイントへの呼び出しのルーチングを試みる場合に利用されるべきである。これらのアドレスはこの呼び出しのみのためにエンドポイントによって利用されるべきである。

**assignedGatekeeper** – エンドポイントに対する割り当てゲートキーパー

### 7.11.3 参加拒否 (ARJ)

ARJ メッセージは、以下を含む：

**requestSeqNum** – この値は、ARQ で送信された値と同じ値でなくてはならない。

**rejectReason** – 参加要求が拒否された理由。ARJ が入側ゲートウェイへ送られた場合にのみ、

**routeCallToSCN** の **rejectReason** が適切な選択であることに注意すること (ARQ がゲートウェイから送信され、ARQ の **answerCall** 値が FALSE である)。**rejectReason** が **routeCallToSCN** の場合には、この呼に対する **rejectReason** は、電話番号または電話番号のリストを含む。これにより、ゲートウェイがそのような手順をサポートしていれば、呼を SCN へリダイレクトすることができる。

**rejectReason** が **exceedsCallCapacity** の場合は、ゲートキーパーは着信先が、その時点で呼を受け入れる能力を持たないと判断する。**collectDestination** の **rejectReason** は、ゲートウェイが最終の着信先アドレスを集めることを、ゲートキーパーが要求していることを示し、ARJ の **serviceControl** フィールドはユーザに示されるプロンプトを表す。**collectPIN** の **rejectReason** は、ゲートウェイが個人識別番号もしくは、公認コードを集めることを、ゲートキーパーが要求していることを示し、ARJ の **serviceControl** フィールドは、ユーザに示されるプロンプトを表す。**genericDataReason** の理由は、要求が一般的要素または機能の結果として、拒否されたことを示す。この場合、追加情報が **genericData** フィールドに記されてもよい。エンドポイントは **invalidEndpointIdentifier** を受信した場合、ゲートキーパーで再登録すべきである。**registerWithAssignedGK** という理由は、割り当てゲートキーパーが利用可能になったためにリクエストが拒否されたことを示す。エンドポイントは、この理由を受信したときに割り当てゲートキーパーに登録する。

**nonStandardData** – 本標準の規定範囲外の情報を送る (例えば、独自データ)。

**altGKInfo** – 代替ゲートキーパーに関するオプションの情報。

**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**callSignalAddress** – 拒否理由が **routeCallToGatekeeper** の場合に返信される、ゲートキーパーの呼制御アドレス。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**serviceControl** – このフィールドは、例えば、JT-H323 付属資料 K に記述されているように、エンドポイントによって使われることが可能なサービス固有のデータやそれに関連するものを含む (例: 呼が失敗した時の理由表示のため)。

**featureSet** – このフィールドは、この呼に関係する一般的な機能セットを記述する。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**assignedGatekeeper** – エンドポイントに対する割り当てゲートキーパー

## 7.12 端末からゲートキーパーへの帯域幅変更要求

BRQ メッセージは、パケットネットワークにおける帯域幅割り当ての変更が許可されるよう、エンドポイントがゲートキーパーに要求するメッセージである。このとき、ゲートキーパーは、その要求を BCF によって許可するか、あるいは BRJ によって拒絶する。

ゲートキーパーは、BRQ の使用によってエンドポイントの帯域幅を増大させるか、縮小させることを要求することがある。もし、要求が帯域幅の増大であれば、エンドポイントは BRJ または BCF で応答する。もし、要求が帯域幅の縮小であれば、エンドポイントは指示された低いレートをサポートしているときには必ず BCF で応答しなければならないが、サポートしていなければ BRJ で応答する。

### 7.12.1 帯域幅変更要求 (BRQ)

BRQ メッセージは、以下を含む：

**requestSeqNum** – 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全ての応答において、この番号を返さなくてはならない。

**endpointIdentifier** – RCF によって端末に割り当てられたエンドポイント識別子。

**conferenceID** – 帯域幅が変更されようとしている呼の ID。

**callReferenceValue** – この呼に対する JT-H225.0 呼制御メッセージからの CRV。ローカルでのみ有効。特

定の呼と BRQ を関連付けるために、ゲートキーパーによって使用される。

**callType** – この値を使用することにより、ゲートキーパーは「実際に」使用されている帯域幅の決定を試みることができる。

**bandWidth** – 100bit/s 単位で表される、呼に対して要求された新しい双方向の帯域幅。これは、オーディオとビデオのビットストリームだけの厳密な帯域幅であり、ヘッダやオーバーヘッドは含まれない。ユニークなマルチキャストストリームは、メディアストリームの受信者が複数の場合でも、全帯域使用量への追加は1度だけにされなくてはならない。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**callIdentifier** – RAS シグナリングと、本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、送信側エンドポイントによってセットされたグローバルでユニークな呼識別子。

**gatekeeperIdentifier** – 登録時のゲートキーパーからの RCF、または以前の BRJ メッセージで受信した **alternateGatekeeper** リストで、エンドポイントが受信した **gatekeeperIdentifier**。 **tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**answeredCall** – このパーティ（このパーティが呼に応答した）が本来の着信先であったことを示すために TRUE に設定される。

**callLinkage** – このフィールドの内容は、一般的には呼結合サービスによりコントロールされる。このフィールドの手順と意味は、標準 JT-H323 の 10 章を参照のこと。

**capacity** – このフィールドは、ゲートキーパーが BCF を送信することによって BRQ を確認したと想定して、送信側エンドポイントのこの時点で利用可能な呼キャパシティを示す。このフィールドを送信する場合、エンドポイントは **currentCallCapacity** 要素を含まなくてはならない。

**usageInformation** – このフィールドは、エンドポイントが、この呼の利用情報をレポートすることを可能とする。ゲートキーパーは、BRQ 送信時に、このフィールドを含めてはならない。

**bandwidthDetails** – それぞれのメディアストリームに、エンドポイントが **bandWidth** フィールドと同じユニットを送信または受信している帯域情報を用意する。メディアストリームの複数回の受信があった場合でも、それぞれのマルチキャストストリームは1回だけレポートされなければならない。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**transportQOS** – エンドポイントは、このフィールドをトランスポート資源を予約するための能力指示に用いることができる。

## 7.12.2 帯域幅変更確認 (BCF)

BCF メッセージは、以下を含む：

**requestSeqNum** – この値は、BRQ で送信された値と同じ値でなくてはならない。

**bandWidth** – 現時点で許可された最大（の帯域幅）であり、100 ビット単位で表される。

**nonStandardData** – 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**capacity** – このフィールドは、送信側エンドポイントのこの時点で利用可能な呼キャパシティを示す。このフィールドを送信する場合、エンドポイントは **currentCallCapacity** 要素を含まなくてはならない。ゲートキーパーによって BCF が送信される場合、このフィールドは含まれない。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**transportQOS** – ゲートキーパーは、このフィールドをエンドポイントによって利用されるリソース予約メ

カニズムを示すために用いることができる。

### 7.12.3 帯域幅変更拒否 (BRJ)

BRJ メッセージは、以下を含む：

- requestSeqNum** — この値は、BRQ で送信された値と同じ値でなくてはならない。
- rejectReason** — 帯域幅変更要求がゲートキーパーによって拒否された理由。**allowedBandWidth** — 現在の割り当て値も含めて、現時点での許容最大値であり、100 bit/s 刻みで表される。
- nonStandardData** — 本標準の規定範囲外の情報を送る(例えば、独自データ)。
- altGKInfo** — 代替ゲートキーパーに関するオプションの情報。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。
- genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

### 7.13 位置情報要求メッセージ

LRQ はゲートキーパーがアドレス変換を提供するように要求する。ゲートキーパーは着信先アドレスのトランスポートアドレスを含む LCF にて応答するか、あるいは要求を LRJ にて拒絶する。

#### 7.13.1 位置情報要求 (LRQ)

LRQ メッセージは、以下を含む：

- requestSeqNum** — 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全てのメッセージにおいて、この番号を返さなくてはならない。
- endpointIdentifier** — RCF によって端末に割り当てられたエンドポイント識別子。
- destinationInfo** — **dialedDigits** や、**PartyNumber (e164Number または privateNumber)**、もしくは **h323-ID** で表される着信先の別名アドレスの系列である。呼に応じるため ARQ を送信する場合、**destinationInfo** はその呼の着信先 (応答するエンドポイント) を示している。少なくとも 1 つの別名がゲートキーパーに登録されており、ARQ 内に複数含まれている別名が同一人物として登録されている場合、ゲートキーパーは ARQ がその人物を指していると認識しなければならない。別名が矛盾している場合、ARQ は **AliasesInconsistent** の理由で拒否されなければならない。もし、ゲートキーパーがこの認識機能を持たない場合には、ゲートキーパーは最初に登録されたアドレスが着信先のアドレスであると考えなくてはならない。
- nonStandardData** — 本標準の規定範囲外の情報を送る(例えば、独自データ)。
- replyAddress** — LCF/LRJ を送る先のトランスポートアドレス。
- sourceInfo** — LRQ の送信者を識別する。ゲートキーパーはこの情報に基づいて LRQ に対する応答を決めることができる。
- canMapAlias** — TRUE の場合、もし返される ACF が **destinationInfo**、**destExtraCallInfo** および/または **remoteExtensionAddress** フィールドを含んでいるならば、エンドポイントが、この情報を呼設定メッセージの **destinationAddress**、**destExtraCallInfo** および **remoteExtentionAddress** フィールドそれぞれにコピーし、または適切ならば着信側パーティ番号 (Called Party Number) 情報要素にコピーしなければならないことを示している。エンドポイントが、JT-H323 ネットワークから出るのに用いられるゲートウェイである場合、ゲートウェイは、着信先情報を、JT-H323 ネットワーク外で使う適切な信号フォーマット(例えば、DTMF)に変換するであろう。もし、ゲートキーパーが ARQ からのアドレス情報を書き替え、かつ **canMapAlias** が FALSE である場合は、そのゲートキーパーは ARQ を拒否するべきである。JT-H255.0 第 4 版以降の版に準拠するシステムは、このフィールドを TRUE に設定しなければならない。**gatekeeperIdentifier** — 登録時のゲートキーパーからの RCF、または以前の LRJ メッセージで受信した **alternateGatekeeper** リストで、エンドポイントが受信した **gatekeeperIdentifier**。**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用

して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**desiredProtocols** – これは発呼側エンドポイントが、その呼（例えばオーディオ、fax）に望ましいと思うプロトコルタイプの優先順序を示す。解読側のエンティティは、優先順序を考慮し、そのプロトコルをサポートするエンドポイントの位置を示すために、このフィールドを使ってもよい。

**desiredTunnelledProtocol** – このフィールドは、トンネリングが必要であるプロトコルを示す。

**featureSet** – このフィールドは、この呼に関係する一般的な機能セットを記述する。

**genericData** – このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

**hopCount** – このフィールドはこのメッセージが通過するゲートキーパーの数を規定する。ゲートキーパーが LRQ を受け、ゲートキーパーが、このメッセージを他のゲートキーパーに転送すべきと判断した場合、**hopCount** を 1 減ずる。**hopCount** が 0 より大きい場合、ゲートキーパーは新しいカウント値を転送するメッセージに挿入する。**hopCount** が 0 に到達した場合、ゲートキーパーはメッセージを転送してはならない。

**circuitInfo** – このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。

**callIdentifier** – グローバルユニーク呼識別子は、本標準で使用される呼制御シグナリングに対応する RAS シグナリングの使用が可能となる発呼エンドポイントによってセットされる。ARQ、もしくは SETUP をサポートした LRQ を送信する場合、ゲートキーパーは ARQ、もしくは SETUP から LRQ に呼識別子をコピーすべきである。発呼準備にて LRQ を送信するエンドポイントは呼のための呼識別子にてこのフィールドを埋めるべきである。呼のコンテキストの外部に送信された LRQ は呼識別子フィールドを含んでいないであろう。

**bandWidth** – 呼のために要求される双方向の帯域幅で 100 bit/s で表される。例えば 128kbit/s の呼は 256kbit/s の要求としてシグナリングされる。値はヘッダ、オーバーヘッドを除くオーディオ、ビデオのビットレートとしてのみ参照する。

**sourceEndpointInfo** – **dialedDigits**、**PartyNumber** (**e164Number** または **privateNumber**)、あるいは **h323-ID** のようなソースとなるエンドポイントのための別名アドレスのシーケンス。ゲートキーパーはこの LRQ を送信する事を支持するエンドポイントのために情報をコピーすべきであり、もしくは受信した LRQ を転送する場合にはゲートキーパーは受信した LRQ から **sourceEndpointInfo** をコピーすべきである。

**canMapSrcAlias** – これが TRUE にセットされた場合、生じる LCF が **modifiedSrcInfo** を含んでいる場合、適切な場合、エンドポイントは呼設定メッセージの **sourceInfo** フィールド、および(または)、**Calling Party Number IE** にこの情報をコピーしても可能であることを示す。ゲートキーパーが LRQ からのアドレス情報を変換し、**canMapSrcAlias** が FALSE である場合、ゲートキーパーは LRQ を拒絶すべきである。

**language** – ユーザがアナウンスとプロンプトを受信する際に使用する言語を表す。

### 7.13.2 位置情報確認 (LCF)

LCF メッセージは、以下を含む：

**requestSeqNum** – この値は、LRQ で送信された値と同じ値でなければならない。

**callSignalAddress** – JT-H225.0 呼制御送信用トランスポートアドレス。高信頼性で公知の、あるいは動的なポートを用いるが、使用される呼モデルによってはエンドポイントまたはゲートキーパーのアドレスの可能性もある。

**rasAddress** – ロケーション済みのエンドポイントの登録、許可、状態表示用アドレス (RAS アドレス)。

**nonStandardData** – 本標準の規定範囲外の情報を送る。(例えば、独自データ)

**destinationInfo** – **dialedDigits**、**partyNumber** (**e164Number** または **privateNumber**) または **h323-ID** で表される着信先の別名アドレスの系列である。

**destExtraCallInfo** – 複数の呼のための外部アドレスを含む。

**destinationType** – 着信先エンドポイントのタイプを示す。

**remoteExtensionAddress** – 複数のゲートウェイを越えるために、この情報が必要である場合、着呼側エンドポイントの別名アドレスを含む。

**alternateEndpoints** – **callSignalAddress**、**rasAddress**、または **destinationInfo** のための優先順位付けされた代替エンドポイントの系列

**tokens** – これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** – 暗号化された **tokens**。

**integrityCheckValue** – RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を

**integrityCheckValue** フィールドに入れて、メッセージを送信する。

**alternateTransportAddresses** — このフィールドは、TCP 以外の伝送用の呼制御アドレスを伝達する。アドレスを含むことは、対応する伝送方式をサポートすることを示す。

**supportedProtocols** — このフィールドはエンドポイントがサポートするプロトコルを示す。

**multipleCalls** — TRUE の場合、このフィールドは、配置されたエンドポイントが単一の呼制御コネクション上で複数の呼制御ができることを示す。FALSE の場合は、着信先エンドポイントはこの能力を持たない。このフィールドが提供されていない場合、ゲートキーパーは、そのエンドポイントがこの能力を持っているかどうか分からない。

**featureSet** — このフィールドは、この呼に関係する一般的な機能セットを記述する。

**genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されてもよい。

**circuitInfo** — このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。

**serviceControl** — このフィールドは、例えば、JT-H323 付属資料 K に記述されているように、エンドポイントが、ネットワークで呼に関連したサービス制御通信に使用してもよいアドレス情報を含む。

**modifiedSrcInfo** — **dialedDigits**、**PartyNumber (e164Number)** または **privateNumber**、あるいは **h323-ID** のようなソースとなるエンドポイントのために使用されるべき別名アドレス。このフィールドは発信側エンドポイントの別名アドレスが変換、もしくは変更される場合、もしくは、一次着信先、代替エンドポイントへの呼び出しのルーチングを試みる場合に利用されるべきである。LCF メッセージがエンドポイントへの ACF 応答に帰着する場合、このフィールドは ACF メッセージにコピーされるべきである。

**bandWidth** — 要求未満となり得る呼のために許可された最大帯域幅。

### 7.13.3 位置情報拒否 (LRJ)

LRJ メッセージは、以下を含む：

**requestSeqNum** — この値は、LRQ で送信された値と同じ値でなくてはならない。

**rejectReason** — 位置情報要求が拒否された理由。**rejectReason** が **routeCallToSCN** で、ゲートウェイがこのような手順をサポートする場合は、この選択のために **rejectReason** はゲートウェイが SCN で呼をリダイレクトすることができる電話番号、あるいは電話番号のリストが同様に含まれる。

**resourceUnavailable** の理由は、帯域幅を過度に使用しているか、現時刻において要求されたロケーションへの呼を取り扱える能力をもったゲートキーパーに登録されているエンティティがないことを示す。**genericDataReason** の理由は、要求が一般的な要素または機能の結果として拒否されたことを示す。この場合、追加情報が **genericData** フィールドに記されてもよい。

**nonStandardData** — 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**altGKInfo** — 代替ゲートキーパーに関するオプションの情報。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

**integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない、空でなくてはならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**featureSet** — このフィールドは、この呼に関係する一般的な機能セットを記述する。

**genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されてもよい。

**serviceControl** — このフィールドは、例えば、JT-H323 付属資料 K に記述されているように、エンドポイントが、ネットワークで呼に関連したサービス制御通信に使用してもよいアドレス情報を含む。

## 7.14 離脱メッセージ

### 7.14.1 離脱要求 (DRQ)

エンドポイントからゲートキーパーへ送られた場合、DRQ はゲートキーパーにエンドポイントが終了されようとしている (being dropped) ことを知らせる。ゲートキーパーからエンドポイントに送られる場合は、DRQ は呼を強制的に廃棄する。この要求は拒絶してはならない。DRQ はエンドポイント間で直接送られることはない。

DRQ は (JT-Q931 の) 解放完了と同じではないことに留意すること。解放完了はゲートキーパーに呼の終了を通知するためのものだからである。ゲートキーパーは呼制御チャネルを終結しようとしているのでなければ、解放完了を受信することはないかもしれない。(TTC 注：呼制御チャネルを終結しようとしている場

合にだけ、解放完了が送られてくるという実装がありえる。)

DRQ メッセージは、以下を含む：

- requestSeqNum** — 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全てのメッセージにおいて、この番号を返さなくてはならない。
- endpointIdentifier** — RCF によって端末に割り当てられたエンドポイント識別子。
- conference ID** — 帯域が解放される呼の ID。
- callReferenceValue** — この呼に対応する JT-H225.0 呼制御メッセージからの CRV。ローカルでのみ有効。特定の呼とそのメッセージ(DRQ)を関連付けるためにゲートキーパーによって使用される。
- disengageReason** — ゲートキーパーや端末から変更が要求された理由。
- nonStandardData** — 本標準の規定範囲外の情報を送る(例えば、独自データ)。
- callIdentifier** — RAS シグナリングと、本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、送信側エンドポイントによって設定されたグローバルでユニークな呼識別子。
- gatekeeperIdentifier** — 登録時のゲートキーパーからの RCF、または以前の DRJ メッセージで受信した **alternateGatekeeper** リストで、エンドポイントが受信した **gatekeeperIdentifier**。(本来のゲートキーパーが応答しない場合や要求を拒絶された場合のバックアップとして使用される。)
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。
- answeredCall** — このパーティ(このパーティが呼に応答した)が本来の着信先であったことを示すために TRUE に設定される。
- remainingCallCapacity** — 残りの全能力のパーセントを示す。
- callLinkage** — このフィールドの内容は、一般的には呼結合サービスによりコントロールされる。このフィールドの手順と意味は、標準 JT-H323 の 10 章を参照のこと。
- capacity** — このフィールドは、ゲートキーパーが DCF を送信することによって DRQ を確認したと想定して、送信側エンドポイントのこの時点で利用可能な呼キャパシティを示す。このフィールドを送信する場合、エンドポイントは **currentCallCapacity** 要素を含まなくてはならない。ゲートキーパーによって DRQ が送信される場合、このフィールドは含まれない。
- circuitInfo** — このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。
  - usageInformation** — このフィールドは、エンドポイントが、この呼の利用情報をレポートすることを可能とする。ゲートキーパーは、DRQ 送信時に、このフィールドを含めてはならない。
- terminationCause** — このフィールドは、呼が終了した理由を記述する。この情報は **disengageReason** フィールド内で提供される理由より詳細な情報である。ゲートキーパーは DRQ の送信時、このフィールドを含んではならない。
- serviceControl** — このフィールドは、例えば、JT-H323 付属資料 K に記述されているように、エンドポイントによって使われることが可能なサービス固有のデータやそれに関連するものを含む。ゲートキーパーは、あるアカウントの有効期限が切れたか、または呼のための支払い総額を使い果たしたため、呼が終了することを示すために、このフィールドを使用できる。
- genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されてもよい。

#### 7.14.2 離脱確認 (DCF)

DCF メッセージは、以下を含む：

- requestSeqNum** — この値は、DRQ で送信された値と同じ値でなくてはならない。
- nonStandardData** — 本標準の規定範囲外の情報を送る(例えば、独自データ)。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。
- capacity** — このフィールドは、DCF 内に示された呼が解放された後の送信側エンドポイントの利用可能な呼キャパシティを示す。このフィールドを送信する場合、エンドポイントは **currentCallCapacity**

要素を含まなくてはならない。ゲートキーパーによって DCF が送信される場合、このフィールドは含まれない。

**circuitInfo** — このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。

**usageInformation** — このフィールドは、エンドポイントが、この呼の利用情報をレポートすることを可能とする。ゲートキーパーは、DCF 送信時に、このフィールドを含めてはならない。

**genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されてもよい。

**assignedGatekeeper** — エンドポイントに対する割り当てゲートキーパー

### 7.14.3 離脱拒否 (DRJ)

DRJ はエンドポイントが登録されていないときにゲートキーパーから送られる。

DRJ メッセージは、以下を含む：

**requestSeqNum** — この値は、DRQ で送信された値と同じ値でなくてはならない。

**rejectReason** — 要求が拒否された理由。

**nonStandardData** — 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**altGKInfo** — 代替ゲートキーパーに関するオプションの情報。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

**integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を

**integrityCheckValue** フィールドに入れて、メッセージを送信する。

**genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されてもよい。

## 7.15 状態要求メッセージ

IRQ は IRR の形式での状態情報を要求するためにゲートキーパーから端末へ送られる。ゲートキーパーから IRQ を受信しなくても、端末は ACF メッセージにて規定された時間間隔で IRR を送ってもよい。このメッセージは JT-H225.0 呼制御状態表示メッセージと混同すべきでない。

第 2 版以降のゲートキーパーに対して自発的な IRR が送られてくるとき、その中の **needResponse** フィールドを用いて、ゲートキーパーが IRR を受信したことを受領確認することを希望することがある。この場合、**requestSeqNum** フィールドに 1 以外の数値を埋める。ゲートキーパーは IACK (肯定の受領確認) または INAK (否定の受領確認) メッセージのいずれかを返す。また、ゲートキーパーは送られてきたのと同じ数値を **requestSeqNum** フィールドに入れて返さなくてはならない。

### 7.15.1 情報要求 (IRQ)

IRQ メッセージは、以下を含む：

**requestSeqNum** — 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全てのメッセージにおいて、この番号を返さなくてはならない。

**callReferenceValue** — 問い合わせ対象の呼の CRV。ゼロの場合、端末が通信中である各呼に対する IRR の要求であると解釈される。端末が通信中の呼がない場合は、**callReferenceValue** を 0 とし、他の適切なフィールドを含んだ IRR を応答として返さなくてはならない。**callReferenceValue** が 0 の場合は、エンドポイントは **callIdentifier** を無視しなくてはならない。この場合、ゲートキーパーは **callIdentifier** を 0 で埋めなくてはならない。

**nonStandardData** — 本標準の規定範囲外の情報を送る(例えば、独自データ)。

**replyAddress** — IRR を送る先のトランスポートアドレス。もしかすると、ゲートキーパーのトランスポートアドレスではないかもしれない。

**callIdentifier** — RAS シグナリングと、本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、送信側エンドポイントによって設定されたグローバルでユニークな呼識別子。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

**integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。



ならず、空でなくてはならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**uuiesRequested** — エンドポイントが ARQ メッセージにて **willSupplyUIIE** を TRUE にすることで、この能力があることを示しているならば、ゲートキーパーはエンドポイントに対して、エンドポイントが送信するか受信するかした JT-H225.0 呼制御メッセージをゲートキーパーに通知するよう要求してもよい。**uuiesRequested** は、エンドポイントがゲートキーパーに通知しなければならない JT-H225.0 呼制御メッセージのセットを示す。

**callLinkage** — このフィールドの内容は、一般的には呼結合サービスにより制御される。このフィールドの手順と意味は、標準 JT-H323 の 10 章を参照のこと。

**usageInfoRequested** — このフィールドは、IRR メッセージ内に示された呼利用情報をエンドポイントがレポートすることを、要求するゲートキーパーによって含まれてもよい。

**segmentedResponseSupported** — このフィールドは、ゲートキーパーがエンドポイントに多重 IRR メッセージ内のすべての呼のための呼情報またはセグメントを、返すことを許可するかどうかを示す。このフィールドが現われればセグメント化は許可され、そうでなければ許可されない。このフィールドは、ゲートキーパーが **callReferenceValue** が 0 である IRQ を送る場合にのみ意味をなし、それ以外の場合には送ってはならない。

**nextSegmentRequested** — ゲートキーパーが 0 の **callReferenceValue** と **segmentedResponseSupported** フィールドを含んで IRQ メッセージを送信したら、エンドポイントは呼情報のうち、IRR 内のセグメントフィールドを含むことによって示される部分だけ IRR と共に返してもよい。ゲートキーパーは受信を希望する次のセグメントの値を **nextSegmentRequested** フィールドに設定し、IRQ メッセージを再送することによって、次のセグメントを要求してよい。

**capacityInfoRequested** — もし提供されれば、このフィールドはエンドポイントが IRR 内に呼キャパシティ情報を含むことをゲートキーパーが要求していることを示す。

**genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されてもよい。

**assignedGatekeeper** — エンドポイントに対する割り当てゲートキーパー

#### 7.15.2 情報要求応答 (IRR)

IRR メッセージは、以下を含む：

**nonStandardData** — 本標準の規定範囲外の情報を送る (例えば、独自データ)。

**requestSeqNum** — 勧誘型 IRR の場合、このフィールドは IRQ からのシーケンス番号を含まなければならない。バージョン 1 ゲートキーパーへの自発的なレポートの場合、このフィールドは 1 を含まなければならない。他の全ての自発型 IRR の場合、単調に増加していく番号 (**needResponse** が TRUE ならゲートキーパーのレスポンスによって返される) を含まなければならない。

**endpointType** — エンドポイントについての情報を提供する。

**endpointIdentifier** — RCF でゲートキーパーによって割り当てられた数値。

**rasAddress** — 登録、許可などに使用するアドレス。

**callSignalAddress** — JT-H225.0 呼制御のアドレス。

**endpointAlias** — エンドポイントの別名。

**perCallInfo** — 特定の呼についての情報：

**nonStandardData** — 本標準の規定範囲外の情報を送る (例えば、独自データ)。

**callReferenceValue** — 応答対象となる呼の JT-H225.0 呼制御 CRV。

**conferenceID** — ユニークな会議 ID。

**originator** — もし TRUE であれば、問い合わせを受けているエンドポイントは呼の発信元であり、FALSE であれば呼の着信先である。

**audio** — オーディオチャンネル (単数または複数) についての情報。そのセッションがマルチキャストの場合、**multicast** 要素が含まれていなければならない。

**video** — ビデオチャンネル (単数または複数) についての情報。そのセッションがマルチキャストの場合、**multicast** 要素が含まれていなければならない。

**data** — データチャンネル (単数または複数) についての情報。

**h245** — JT-H245 制御チャンネルのトランスポートアドレス。

**callSignaling** — JT-H225.0 呼制御チャンネルのトランスポートアドレス。

**callType** — 呼トポロジーの情報を提供する。

**bandwidth** — 100bit/s 単位で表される現在の帯域使用量。ヘッダとオーバヘッダを除いたオーディオとビデオのみを含む。

**callModel** — 使用されるエンドポイントの呼モデルの概念を示す。

**callIdentifier** — RAS シグナリングと、本標準で使用するために変更された JT-Q931 シグナリングを関連付けるために使われる、送信側エンドポイントによって設定されたグローバルでユニークな呼識別子。

- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- substituteCnfIDs** — 元の **perCallInfo conferenceID** に属している JT-H245 SubstituteCID メッセージによって受信された全ての ConferenceID のリスト。
- pdu** :
- h323pdu** — ゲートキーパーが、ACF または IRQ の **uuiesRequested** で要求した JT-H225.0 と JT-Q931 PDU の複写。
  - sent** — TRUE の場合、エンドポイントが **h323pdu** を送信したことを示す。FALSE の場合、エンドポイントが **h323pdu** を受信したことを示す。
  - callLinkage** — このフィールドの内容は、一般的には呼結合サービスにより制御される。このフィールドの手順と意味は、標準 JT-H323 の 10 章を参照のこと。
  - usageInformation** — このフィールドは、エンドポイントが、この呼の利用情報をレポートすることを可能とする。
  - circuitInfo** — このフィールドは、SCN 網またはこの呼で使われる網に関する情報を提供する。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。
- needResponse** — TRUE にセットされ、かつ、ゲートキーパーが RCF または ACF において自発的な IRR へ応答することを (**willRespondToIRR** を TRUE にセットすることによって) 示しているならば、ゲートキーパーは IACK または INAK によって返答しなければならない。もし、ゲートキーパーが RCF または ACF において自発的な IRR へ応答しないことを (**willRespondToIRR** を FALSE にセットすることによって) 示しているならば、ゲートキーパーはブール型要素 **needResponse** を無視してもよい。
- capacity** — この時点での送信側エンドポイントの呼キャパシティを示す。このフィールドを送信する場合、エンドポイントは **currentCallCapacity** 要素を含まなければならない。また **capacityInfoRequested** 要素を含む IRQ に応答する場合、**maximumCallCapacity** だけを含まなければならない。
- irrStatus** — この要素はゲートキーパーによって送信された IRQ への応答である IRR メッセージ内で返されなければならない。この要素がないということは IRR メッセージが完全な呼の詳細情報を含むことを示す。以下の値が可能である :
- complete** — 全ての呼の詳細を要求する IRQ のための最後の呼情報セグメントを IRR が含むことを示す。分割されないとき、このフィールドは単一の IRR メッセージ内の呼の詳細すべてを IRR が含むことを示す。
  - incomplete** — 0 の **callReferenceValue** を含む IRQ メッセージに回答するとき単一の IRR メッセージ内の呼情報の要求すべてに合わせられないことを示す。
  - segment** — このフィールドは分割された IRR が 0 の **callReferenceValue** を含んだ IRQ への応答として送信されるとき、この IRR メッセージの単調増加する値モジュール 65536 であるシーケンス番号を示す。
  - invalidCall** — このフィールドは IRQ メッセージ内の参照される呼が存在しないことを示す。
- unsolicited** — JT-H323 第 4 版以降の版のエンドポイントは JT-H323 の 8.4.2 節に記載されているように自発型 IRR メッセージ内のこのフィールドに TRUE をセットしなければならない。そして勧誘型 IRR メッセージ内では FALSE をセットしなければならない。
- genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されてもよい。

### 7.15.3 情報承認 (IACK)

IACK メッセージは、以下を含む :

- requestSeqNum** — このフィールドは、IRR 中の **requestSeqNum** を含まなければならない。
- nonStandardData** — 本標準の規定範囲外の情報を送る (例えば、独自データ)。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づ

く完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

#### 7.15.4 情報否定承認 (INAK)

INAK メッセージは、以下を含む：

- requestSeqNum** — このフィールドは、IRR 中の **requestSeqNum** を含まなければならない。
- nonStandardData** — 本標準の規定範囲外の情報を送る (例えば、独自データ)。
- nakReason** — IRR に NAK が返される理由。
- altGKInfo** — 代替ゲートキーパーに関するオプションの情報。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

#### 7.16 非標準メッセージ

NonStandardMessage 構造は、以下のようになる：

- requestSeqNum** — 送信側に対してユニークな単調に増加する番号。
- nonStandardData** — 本標準の規定範囲外の情報を送る (例えば、独自データ)。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。
- featureSet** — このフィールドは、一般的な機能セットを記述する。
- genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

#### 7.17 認識不能なメッセージ

JT-H323 エンドポイントが認識できない、もしくはデコードできない RAS メッセージを受信した場合は常に、このメッセージが送信される。XRS メッセージの着信先トランスポートアドレスを利用できない (すなわち、受信した RAS メッセージをデコードできない) 場合には、XRS は認識不能な RAS メッセージを受信した所からトランスポートアドレスへ送信されることが可能である。このトランスポートアドレスは、下層のトランスポートレイヤから入手してもよい。受け取った XRS メッセージに答えて XRS メッセージを送信してはならない。JT-H323 エンドポイントは、誤ったメッセージが受信される状況において、ネットワークの輻輳を避けるために、同じトランスポートアドレスへ 1 秒に 1 つしか XRS メッセージを送ってはならない。

- requestSeqNum** — デコードできる場合は認識不能メッセージの **requestSeqNum** としなければならない。もし、認識不能メッセージをデコードできなければ、このフィールドは送信側に対してユニークな、単調に増加する番号である。**requestSeqNum** は、JT-H323 第 3 版以前のエンドポイントとの下位互換性のために使われることが望ましい。JT-H323 第 4 版以降のエンドポイントは、以前に送られたメッセージと XRS を関連づけるために、**messageNotUnderstood** パラメータに着目することが望ましい。
- tokens** — これは、動作を許可するために必要となりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージ挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づ

く完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**messageNotUnderstood** — 受信して解釈できなかったメッセージの複写。

## 7.18 ゲートウェイ利用可能資源メッセージ

利用可能資源通知 (RAI) は、各 H シリーズプロトコルのための現在の呼容量およびデータ速度をゲートウェイからゲートキーパーへ通知するのに用いられる。ゲートキーパーは RAI を受信すると、その通知受領を行うために利用可能資源応答 (RAC) で応答する。

### 7.18.1 利用可能資源通知 (RAI)

RAI メッセージは、以下を含む：

**requestSeqNum** — 送信側に対してユニークな単調に増加する番号。受信側は、この特定のメッセージに関連する全ての応答において、この番号を返さなくてはならない。

**protocolIdentifier** — 送信側エンドポイントの版を示す。

**nonStandardData** — 本標準の規定範囲外の情報を送る (例えば、独自データ)。

**endpointIdentifier** — ゲートキーパーによって割り当てられたエンドポイント識別子文字列。

**protocols** — 与えられたデバイスの現状でサポートできるプロトコル別の現在のデータ速度を示す。

**almostOutOfResources** — TRUE にセットされているときは、デバイスは最大容量か、もしくはそれに近い状態である。このフィールド上で定義するいくつかの動作は製造者の自由である。もしデバイスが最大容量でないか、またはそれに近くないなら、フィールドを FALSE にセットすべきである。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

**integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**capacity** — この時点での送信側エンドポイントの呼キャパシティを示す。もし **capacity** が提供されるならば、**capacity** フィールドがより詳細な情報を提供するため、**almostOutOfResources** BOOLEAN が受信側によって無視されることが望ましいが、**almostOutOfResources** BOOLEAN は下位互換性を維持するために正しく設定する必要がある点に注意すること。**capacity** フィールドを送る場合、エンドポイントは **currentCallCapacity** 要素を含めなければならない。

**genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

### 7.18.2 利用可能資源確認 (RAC)

RAC メッセージは、以下を含む：

**requestSeqNum** — この値は、RAI で通知された値と同じ値でなくてはならない

**protocolIdentifier** — 受け入れるゲートキーパーの版を識別する。

**nonStandardData** — 本標準の規定範囲外の情報を送る (例えば、独自データ)。

**tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。

**cryptoTokens** — 暗号化された **tokens**。

**integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない。計算後は、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。

**genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関係する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

### 7.19 RASタイマおよび進行中 (RIP) メッセージ

表 24/JT-H225.0 は、RAS メッセージに対して応答がない場合のタイムアウト値と再試行回数の推奨デフォルト値である。(これらの値は今後の実装経験に基づいて変更される。)

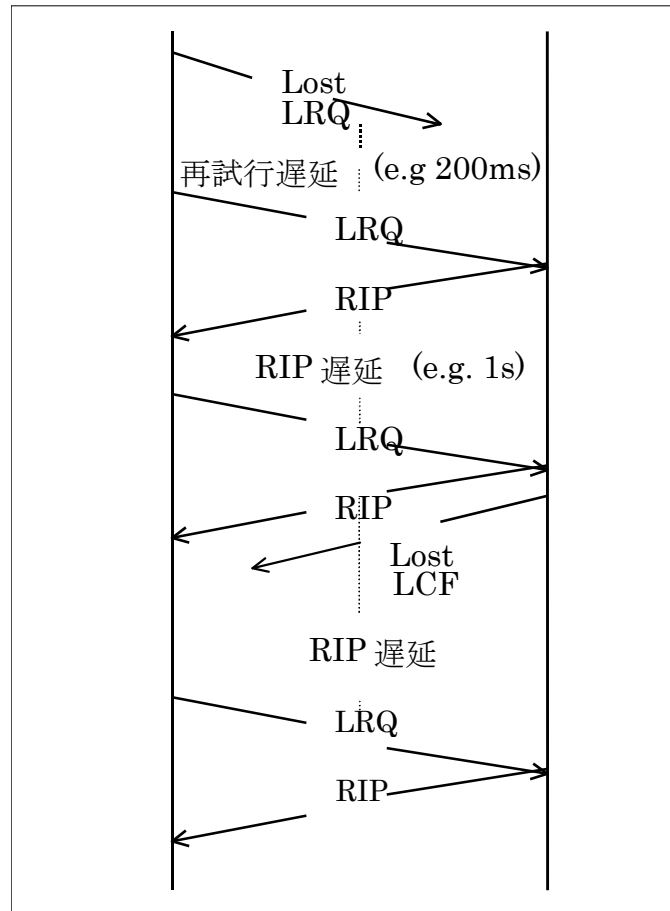
**表 24/JT-H255.0 推奨デフォルトタイムアウト値  
(ITU-T H.225.0)**

RASメッセージ	タイムアウト値(sec)	再試行回数
GRQ	5	2
RRQ (注1)	3	2
URQ	3	1
ARQ	3	2
BRQ	3	2
IRQ	3	1
IRQ (注2)	5	2
DRQ	3	2
LRQ	5	2
RAI	3	2
SCI	3	2

注1 - タイムアウト値は現在の時間 (RCF メッセージ内にゲートキーパーによって示されるかもしれない) と望ましいリトライ回数両方の上に  
基づいて計算し直さなければならない。  
注2 - ゲートキーパーが自発的IRR に対してIACKまたはINAKで応答  
することが予期されている場合には、もしIRR に対する応答が受信され  
ない場合、タイムアウトが発生するかもしれない。

もしエンティティが代表的な再試行タイムアウト時間内に応答を生成できなかった第2版 (またはそれ以降の版) のエンティティから要求を受けた場合、エンティティは応答を生成するのに要する時間を (**delay** フィールドを用いて) 特定する RIP メッセージを送出できる。応答しようとしているエンティティは、応答内容が得られるとすぐに RIP 遅延時間を待つことなく応答を送出すべきである。要求を送ったエンティティが許容時間内に RIP 応答を受信できなかった場合、エンティティは要求を再度送し出さなければならない。応答しようとしているエンティティは複写応答または別の RIP メッセージのどちらかを送し出できる。図 2/JT-H225.0 は再試行手順のメッセージ交換例を与える。

製造者はいくつかの再試行が呼設定時間に与える影響を認識しなければならず、最小にしなければならない。従って短い再試行時間が望ましい。それゆえ遠く離れたエンティティが RIP メッセージを送出すべき時間を決定する目的により代表的な再試行時間を予想できるよう、100ms より短い再試行時間は避けるべきである。指数関数的な再試行時間の短縮(exponential backoff)および測定した往復遅延時間への適応が推奨される。エンティティは RRQ/RCF での往復遅延時間の測定結果を用いて、安全を見越した (2~3 秒の) タイムアウト初期値を変更することが可能である。エンティティはバージョン 1 エンティティがシ



グナリングに含まれるとき、RIP に基づく再試行機構を使用しないようにするため、バージョン番号を交換するために登録プロセスを使用してもよい。

図 2/JT-H225.0 RIP メッセージ使用例  
(ITU-T H.225.0)

RIP メッセージは、以下を含む：

- requestSeqNum** — 現在処理中の要求の requestSeqNum。
- nonStandardData** — 本標準の規定範囲外の情報を送る (例えば、独自データ)。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された tokens。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。integrityCheckValue の計算の前は、このフィールドは無視されなければならない。計算後、送信側は計算された完全性チェック値を integrityCheckValue フィールドに入れて、メッセージを送信する。
- delay** — エンドポイントが再試行を試みる前に待たなければならない msec 単位の時間を記述する。応答しているエンドポイントは時間いっぱい待機せず応答してもよい。

## 7.20 サービス制御メッセージ

### 7.20.1 サービス制御識別子 (SCI)

SCI メッセージは、サービス・プロバイダから、与えられたアドレスに対して分離されたサービス制御セッションが初期化可能であることをサービスクライアントに示すために送られる。それは、ゲートキーパーからエンドポイント（例えば、サービス機能のユーザー表示のため）へ送られてもよいし、エンドポイントからゲートキーパー（例えば、呼処理スクリプトをアップロードするため）へ送られてもよい。第3版以前の JT-H323 エンティティはこのメッセージをデコードできなく、それゆえ返答しない点に留意すること。

SCI メッセージは、以下を含む：

- requestSeqNum** — 送信側に対してユニークな、単調に増加する番号。受信側は、この特定のメッセージに関連する全ての応答においてこの番号を返さなくてはならない。
- nonStandardData** — 本標準の規定範囲外の情報を送る（例えば、独自データ）。
- serviceControl** — サービス制御セッション情報のセットを送る。
- endpointIdentifier** — このメッセージがエンドポイントからゲートキーパーへ送られる場合、RCF メッセージ中でゲートキーパーから受信した値を設定する。
- callSpecific** — 与えられたセッションが1つの特定の呼と関係がある場合に提供される。**callIdentifier**、**conferenceID**、及び **answeredCall** は ARQ メッセージ中の関係のあるサービス・セッションと同じ値にセットしなければならない。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージのメッセージ完全性/メッセージ認証を改善する。暗号技術に基づく完全性チェック値は、交渉済みの完全性チェックアルゴリズムと秘密鍵をメッセージ全体に適用して、送信側で計算する。**integrityCheckValue** の計算の前は、このフィールドは無視されなければならない、空でなくてはならない。計算後、送信側は計算された完全性チェック値を **integrityCheckValue** フィールドに入れて、メッセージを送信する。
- featureSet** — このフィールドは一般的な機能セットを記述する。
- genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

### 7.20.2 サービス制御応答 (SCR)

SCR メッセージは SCI メッセージの受信確認のために送信される。しかし SCR メッセージは、必ずしもサービスクライアントが SCI メッセージの様にそのセッションを初期化するというを意味しない。

SCR メッセージは、以下を含む：

- RequestSeqNum** — この値は、SCI で送信された値と同じ値でなくてはならない。
- result** — このフィールドは SCI メッセージに含まれる情報の処理結果を示す。次の値が定義されている。
  - started** — 要求されたサービス制御が開始された。
  - failed** — 要求にエラーが生じたため、要求が失敗した。
  - stopped** — サービス制御が停止した。
  - notAvailable** — 要求があった時点では、要求されたサービス制御は利用できない。
- nonStandardData** — 本標準の規定範囲外の情報を送る（例えば、独自データ）。
- tokens** — これは、動作を許可するために必要になりうる、いくつかのデータである。利用可能な場合は、このデータはメッセージに挿入しなければならない。
- cryptoTokens** — 暗号化された **tokens**。
- integrityCheckValue** — RAS メッセージの改善された完全性、認証を提供する。暗号が基本となる完全性のチェック値は、交渉された完全性アルゴリズムとメッセージ全体に対する秘密鍵を利用する送信側により計算される。**integrityCheckValue** が計算される前では、このフィールドは無視されるか、空である必要がある。計算後は、送信側が計算された完全性チェック値を **integrityCheckValue** フィールドに記入し、メッセージを送出する。
- featureSet** — このフィールドは、一般的な機能セットを記述する。
- genericData** — このフィールドは、JT-H225.0 基本仕様以外で定義される機能に関する一般的な要素リストである。例えば、このパラメータは RAS を通じて透過的に情報をトンネリングするために利用されることが可能である。

### 7.21 AdmissionConfirmSetSequence

AdmissionConfirmSequence は一つ以上の RAS ACF メッセージのシーケンスである。一つの ACF メッセージでは表現し難いかもしれない異なるセキュリティトークン、異なる変換ソース情報などを持つ場合、一つの ACF メッセージに替わって一つの ARQ メッセージに回答するためにゲートキーパーによって利用されることが可能である。エンドポイントは RRQ の **supportsACFSequences** フラグをセットすることで AdmissionConfirmSequence の受信をサポートすることを示す。

### 7.22 エラーコードマッピング

**LocationRequest** または **AccessRequest** メッセージの送信に対する応答において **LocationReject** または H.501 の **AccessRejection** を受信した結果として、エンドポイントから出された **AdmissionRequest** に対する応答において **AdmissionReject** を返送する必要があるゲートキーパーは、**AdmissionReject** メッセージにおいて、返送するエラーコードをマッピングするために、表 25/JT-H225.0 と表 26/JT-H225.0 を利用しなければならない。

表 25/JT-H225.0 – LocationRejectReason と AdmissionRejectReason の対応 (ITU-T H.225.0)

LocationRejectReason	対応する AdmissionRejectReason
notRegistered	calledPartyNotRegistered
invalidPermission	invalidPermission
requestDenied	requestDenied
undefinedReason	undefinedReason
securityDenial	securityDenial
aliasInconsistent	aliasesInconsistent
routeCallToSCN	routeCallToSCN
resourceUnavailable	resourceUnavailable
genericDataReason	genericDataReason
neededFeatureNotSupported	neededFeatureNotSupported
hopCountExceeded	noRouteToDestination
incompleteAddress	incompleteAddress
securityWrongSyncTime	securityWrongSyncTime
securityReplay	securityReplay
securityWrongGeneralID	securityWrongGeneralID
securityWrongSendersID	securityWrongSendersID
securityMessageIntegrityFailed	securityMessageIntegrityFailed
securityWrongOID	securityWrongOID
securityDHmismatch	securityDHmismatch
noRouteToDestination	noRouteToDestination
unallocatedNumber	unallocatedNumber

表26/JT-H225.0 – AccessRejectionReason と AdmissionRejectReason の対応 (ITU-T H.225.0)

AccessRejectionReason	対応する AdmissionRejectReason
noMatch	noRouteToDestination



packetSizeExceeded	undefinedReason
Security	securityDenial
hopCountExceeded	noRouteToDestination
needCallInformation	undefinedReason
noServiceRelationship	noRouteToDestination
undefined	undefinedReason
neededFeature	neededFeatureNotSupported
genericDataReason	genericDataReason
destinationUnavailable	resourceUnavailable
aliasesInconsistent	aliasesInconsistent
resourceUnavailable	resourceUnavailable
incompleteAddress	incompleteAddress
unknownServiceID	noRouteToDestination
usageUnavailable	undefinedReason
cannotSupportUsageSpec	undefinedReason
unknownUsageSendTo	undefinedReason

## 8 サービス品質 (QOS) を維持する機構

### 8.1 一般的なアプローチと想定

パケットネットワークでのトランスポートサービス品質(QOS)は以下の特性を持つ。

- 符号誤り率
- パケット損失率
- 遅延

どのトランスポートサービス品質 (QOS) 関連シグナリング (例えばルータに対する予約要求等) も、端末ができる限り速やかに行なうか、またそれに代わってゲートキーパーが行う。ゲートキーパーが論理的に端末の近くにないか、あるいは端末に代わって実施する QOS 関連の要求が不可能となることがあるため、端末は何らかの予約を行うようにすることができる。端末またはゲートキーパーのどちらかがサービス品質 (QOS) または帯域幅の予約を行なう方法はこの標準の範囲外である。

RTCP の送信側レポートおよび受信側レポートはサービス品質 (QOS) を査定できる方法でなければならない。

測定される輻輳に関する遅延には 2 種類ある。

- 短期間に知覚できる結果の遅延が増加するが、フレーム速度の低下はいらだたしいものではない。
- パケットネットワークの輻輳遅延が時間とともに全体的に増加。フィードバックベース機構が役立つ。

本質的に、短期間のバーストはエラーの隠蔽により対処され、長期間のバーストはマルチメディアロードを減らすことにより対処される。すべてのパケットネットワークマルチメディア端末は JT-H323 端末であり、輻輳が増加すれば互いの帯域幅を「steal」するより、むしろパケットネットワークの使用を抑制しようと試みると想定する。

パケットネットワークのビットエラーは、下位レイヤで訂正されるか、パケット損失のどちらかとなるので、この節ではこれ以上考慮しない。

パケット損失については、受信側が可能な範囲でエラーを隠蔽する方法で損失パケットを補う必要がある。データと制御に関しては、トランスポートレイヤでの再送が使用される。オーディオとビデオに関しては、再送は今後の課題である。

トランスポートサービス品質 (QOS) の所定レベルは、結果としてユーザーが知覚するオーディオ/ビデ

オのサービス品質 (QOS) レベルであり、トランスポートサービス品質 (QOS) 問題の解消に用いられる方法の有効な一部の機能である。

## 8.2 サービス品質(QOS)の測定へのRTCPの使用

### 8.2.1 送信側レポート

送信側レポートは3つの主な目的のために使用される：

- 1) オーディオやビデオのような、複数の RTP ストリームの同期が取れるようにする。
- 2) 期待されるデータレートとパケットレートを受信側が知ることができるようにする。
- 3) 受信側が送信側までの時間的隔たりを測定できるようにする。

これら3つの目的のうち、1)はもともと本標準に関係がある。製造者は自由に送信側レポートを他の用途に用いることができる。

ストリーム同期のための適切なフィールドは、RTCP の送信側レポートにおける RTP タイムスタンプと NTP タイムスタンプである。NTP タイムスタンプは (もし利用可能であれば) 「ウォールクロック」タイムを提供し、メディアパケットにおける RTP キャプチャタイムスタンプであるところのランダムオフセットと同じユニットを持っている RTP タイムスタンプに対応する。

### 8.2.2 受信側レポート

本標準における受信側レポートの4つの部分がサービス品質 (QOS) を測定するために用いられる：

- 1) パケット損失率
- 2) 累積パケット損失
- 3) 受信した拡張最上位シーケンス番号
- 4) 到着間隔ジッタ

項目 2)と 3)は、以前の受信側レポート以後に失われたパケットの数を計算するのに用いられる。これはパケットネットワーク輻輳の長期測定に利用することができる。RFC3550 の 6.3.4 節に計算例がある。もしこの損失レートを製造者によってセットされた値を超えた場合、JT-H225.0 端末は以下の 8.4 節の手順に従って、パケットネットワーク側のメディアレートを減少させるべきである。もし項目 1)が製造者によってセットされた値を超えた場合も、同様に正常化する動作をとってもよい。

もし受信側レポート間の間隔が製造者によって設定された値を超えた場合、JT-H323 端末は項目 1)をパケットネットワーク側のメディアレート低下を必要としている深刻な輻輳の指示子として用いるべきである。

項目 4)は切迫した輻輳の指示として用いられるべきである。もし到着間隔ジッタが3つの連続した受信側レポート間で増加しているなら、JT-H323 送信側端末は正常化する動作をとるべきである。

## 8.3 オーディオ/ビデオジッタ手順

JT-H245 は **RoundTripDelayRequest** および **RoundTripDelayResponse** を用いて往復表示のためのコマンドと手順を提供する。多地点呼の MC はエンドポイントからの要求に対して応答する。RTCP は送信側レポートと受信側レポートメッセージをもとに、往復遅延を計算する方法を含んでいる。それぞれのケースで測定される量は同じでないことに注意し、したがってそれぞれの方法を用いてジッタを測定する際に矛盾は起こらない。

どのように JT-H245 レベルシグナリングが選択的にジッタ関連の遅延を減少させることができるかの議論は JT-H323 の 6.2.5 節を参照。

## 8.4 オーディオ/ビデオスキュー手順

どのように JT-H245 レベルシグナリングが異なる論理チャネル間でのスキュー限定に使用されるかの議論は JT-H323 の 6.2.6 節を参照。

## 8.5 サービス品質 (QOS) 維持のための手順

JT-H323 ゲートウェイ/端末が、遠くにある受信側中のパケット損失またはインターアライバルジッタの増加に対応するためのいくつかの方法が存在する。これらの方法は、パケットの損失や遅延のような短期的

な問題にすばやく応答するのに適しているものと、パケットネットワーク上の拡大する輻輳のような長期的な問題に反応するのに適しているものに分類することができる。これらの方法が現在のサービス品質を維持しようとはせず、かわりにサービス品質の規則的な低下を与えることに注意する。以下の優先順位を遵守しなければならない、もし以下のメディアが存在するならば、メディアは以下の順番で品質が低下していかななければならない：ビデオ、データ、オーディオ、制御。

短期的応答：

- 短期間、フレームレートを減少させる。この結果、パケットネットワークから SCN 方向のパケットアンダーフローを補償するために、JT-H323 ゲートウェイが JT-H261 フィルフレームを追加して送信することがある。
- オーディオ/ビデオが一つのパケットにミックスされるオプションモードに切り替わることによってパケットレートを減少させる。（将来検討）
- ビデオストリームの MB フラグメンテーションを用いることによっても、パケットレートを減少させることができる。

長期的応答：

- メディアのビットレートを減少させる（例えば、384kbit/s から 256kbit/s へ切り替える）。これは端末のエンコーダに対して簡単な指示をする必要があるか、JT-H323 ゲートウェイ中のレート減少機能を用いる必要がある。この変更は JT-H245 FlowControl コマンドによって行われるか、または適切な論理チャンネルシグナリングによって行われる。
- 重要度の低いメディアを止める（例えば大量の T.120 トラフィックを通すためにビデオを止める）。
- パケットネットワークの輻輳をあらわすものとして、ビジーシグナル（アダプティブビジー）を受信側に返す。これは一つのメディア、または制御トランスポートを除くすべてのメディアを止めることと組み合わせることができる。アダプティブビジーは、解放完了中の JT-Q931 の理由種別によって表示される。

パケットの大部分の順序が乱れて到着するようなマルチルータパスの中でパケット到着間隔のジッタに対応することは困難であることに注意すべきである。このジッタの原因をほかの原因と区別すること、または測定したジッタを元にエラー回復する手順を行うことは、共に不可能かもしれない。しかしながらパケット損失は定量化でき、あいまいではない。

## 8.6 エコー制御

音響エコーの制御は、JT-H シリーズ端末の責任である。一般的に、ビデオ/オーディオの圧縮に含まれる遅延があるとき、全ての JT-H320/ JT-H323 そして JT-H324 端末はエコー制御（キャンセレーションまたはスイッチング）のいくつかの形式を持つと仮定されている。

しかしながら、JT-H323 端末が GSTN 電話を呼び出す時、GSTN 電話がエコー制御をサポートしていないのが典型的なケースである。このように、JT-H323 端末のユーザは GSTN 側からの音響エコーが戻ってくるのを聞くであろう。この音響エコーの返信は、エコー制御付きのスピーカホンまたはハンドセットまたはイヤホンを用いることで最小化できる。製造者は JT-H323 端末が GSTN POTS 電話に接続する時、オーディオパスに減衰用の損失を追加できる。

ハイブリッド（2線・4線変換）エコーの制御。ハイブリッド回線は4線伝送システムと2線端末間のインタフェースを提供する。64kbps で GSTN により伝送される ISDN スピーチ呼に対しては、エコーキャンセレーションは要求されない。64kbps データ呼に対してはエコーキャンセレーションは許可されない。

SS7 ネットワークに対するインタフェースとなる分割ゲートウェイの場合、エコーキャンセレーションの準備の提示は ITU-T 勧告 Q115 に指定されている通りに、ISUP シグナリングメッセージ中で伝送される。JT-H323 メディアゲートウェイコントローラ（MGC）はシグナリング情報を解釈することが可能であり、メディアゲートウェイ（MG）におけるエコーキャンセレーションを有効化、無効化させることが可能である。GSTN が GSTN 内でエコーキャンセレーションを提供していたとしても、スピーチ呼に対しては音声品質に対する有害なエフェクトを与えることなく MGC がエコーキャンセレーションを有効化することが可能である。

JT-H323 ネットワーク上で通過もしくは終端する音声帯域データ呼（モデム呼）に対しては、エコーキャンセレーションの制御はインチャンネルトーンによるモデムが提供する。アウトバンドシグナリングは GSTN ネットワーク要素や MGC により要求されることはない。

## 付属資料 A： RTP/RTCP

RTP と RTCP は IETF RFC 3550[37]で定義される。また、IETF RFC3550 は付録 I で参照される。この付属資料と付録 I の両方がこの勧告の以前のバージョンとの等価性を維持するためのこのドキュメントで保有されている。

読者は、参照の全ての参考文献は、文献目録への参照か、標準でないことに注意すべきであるが、ただし、本標準の参考文献の章中に現れる ISO/IEC 10646[29]への参照は例外である。

また読者は、付表 A.1/JT-H225.0 に示すように、参照[37]で用いている用語と JT-H323 および本標準で用いている用語が異なることにも注意すべきである。

付表 A.1/JT-H225.0 用語の対応  
(ITU-T H.225.0)

JT-H323/JT-H225.0 の用語	参照[37] (RTP/RTCP) の用語
メディアストリーム	データ
トランスポートアドレス	トランスポートアドレス
パケットネットワークアドレス	ネットワークアドレス
TSAP 識別子	ポート
付属資料 A	仕様またはドキュメント
shall (なければならない)	must (なければならない)
should (すべきである)	should (すべきである)

さらに注意すべきことは「トランスレータ」と「ミキサー」が JT-H323 システムには含まれないということである。ゲートウェイや MCU 等の JT-H323 のエンドポイントはトランスレータやミキサーの特性をいくつか持っており、したがってこのテキストは実装者へのガイドとして使用される。しかしながら、トランスレータとミキサーのサポートは JT-H323 には含まれず、この章は情報を提供するだけと考えてほしい。

付属資料 B： RTP プロファイル

RTP Profile は IETF RFC 3551[38]で定義される。また、IETF RFC3551 は付録 II で参照される。この付属資料と付録 II の両方がこの勧告の以前のバージョンとの等価性を維持するためにこのドキュメントで保有されている。

付属資料 A の序文を参照すること。そこで述べられている全ての警告は本付属資料にも適用する。

付属資料 C : JT-H261 のビデオストリームのRTPペイロードフォーマット (標準)

H.261 ビデオ・ストリームのための RTP ペイロードフォーマットは IETF RFC 2032[39]で定義される。また、IETF RFC2032 は付録 III で参照される。この付属資料と付録 III の両方がこの勧告の以前のバージョンとの等価性を維持するためにこのドキュメントで保有されている。

付属資料 A の序文を参照の事。そこに記された注意事項は本付属資料にも適用される。

付属資料 D : JT-H261AビデオストリームのRTPペイロードフォーマット (標準)

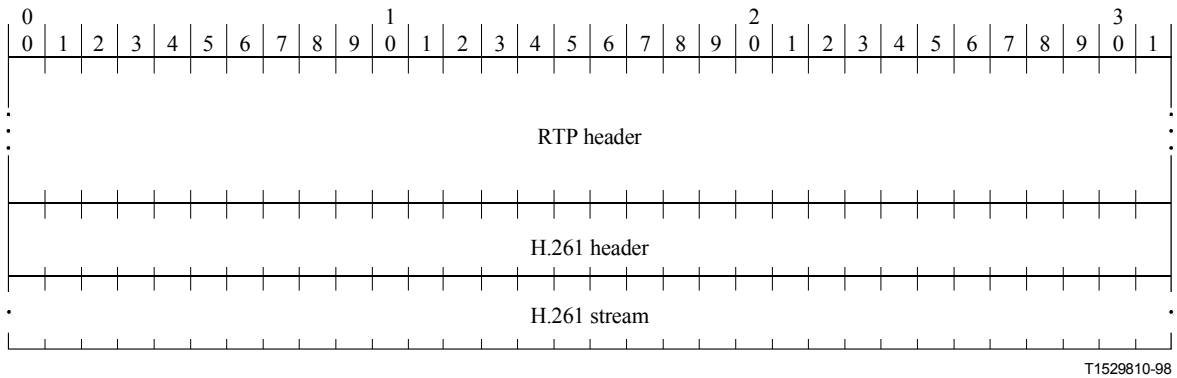
D.1 はじめに

JT-H323 ビデオストリームはゲートウェイを通して SCN に接続するのを容易にする為、JT-H323 は RTP JT-H261 ビデオペイロードの改変したものを定義している。これはリモート SCN コーデックの際のバッファ管理と外部とのやり取りを簡易化する。JT-H261A ペイロードタイプのサポートは、JT-H245 機能群と **openLogicalChannel** メッセージでの RTP 可変ペイロードタイプの使用を意味している。

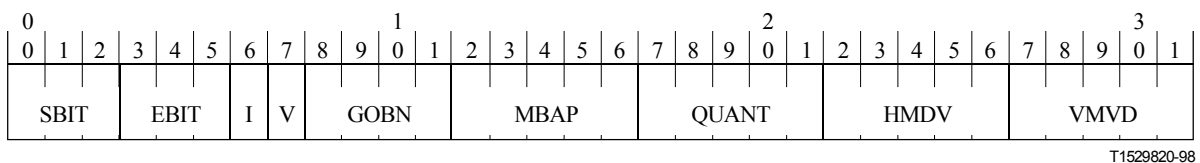
D.2 JT-H261A RTPパケット化

このバージョンは、JT-H261 ヘッダに 32 ビットワードが付加されている点を除いて、付属資料 C で記したものの拡張である。付属資料 C で述べた事項もまたこの付属資料に適用される。

JT-H261A のデータには以下の RTP ヘッダが付く。



JT-H261A ヘッダは次のように定義される。



JT-H261A の各フィールドは、以下の意味を持つ。

- JT-H261 header** : 32 ビット。付属資料 C で記述している。
- Last GOB Number (LGOBN)** : 4 ビット RTP パケット中で最後の GOB 番号。(JT-H261 では最大の GOB 番号は 12 である。)
- Reserved (RES)** : 予約領域
- Byte Count** : 24 ビット RTP パケットの JT-H261 ストリームの部分で送信されたオクテットの累積数。パケットの最後のバイトが部分的にしか使用されていない場合は (これは EBIT で表示されるが)、これは累積バイト数にはカウントされない。このモジュロ  $2^{24}$  のバイトカウントはランダムな値で始まり、決してリセットされる事はない。

この各追加フィールドは、パケットが消失したり、順番どおりに送られなかった場合に使用される。**Byte Count** は SCN ストリーム中に必要なスタッフィング (stuffing) の量を決めるのに使用でき、バッファの管理を容易にする。**Last GOB Number** はパケット消失により失われた GOB の検索を容易にする。

## 付属資料E：ビデオパケット化

本付属資料では、ビデオ符号化方式に関する RTP パケット化の詳細について記述する。付表 E.1/JT-H225.0 は、本標準の規定外のビデオパケット化フォーマットの規定に関する参照資料を示す。本付属資料の残りの部分は、追加のビデオパケット化フォーマットを規定する。

付表 E.1/JT-H225.0 本標準外で規定されているビデオパケット化フォーマット  
(ITU-T H.225.0)

符号化方式名	パケット化定義
ISO/IEC 14496-2 (MPEG-4 Video)	ISOC/IETF RFC 3016, <i>RTP Payload Format for MPEG-4 Streams.</i>

### E.1 JT-H263

JT-H263 ビデオに対する RTP ペイロードフォーマットは、JT-H263 ビデオビットストリームに対する IETF RFC 2190 で規定されている。ただし、現時点（1998年2月）では、JT-H263 第2版（1998年版）で採用された新しい機能（JT-H263 の付属資料H以降の節、シンタックス的には PLUSPTYPE 以降のもの）には対応していない。JT-H263 第2版ビットストリームの新しい機能を追加したペイロードフォーマットは、後日、規定される予定である。産業界で広く用いられている独自パケット化フォーマット（IETF RFC2190 の規定とは異なる）は、同位（通信相手）が能力交換においてこのフォーマットをサポートしていると表明した場合に限って使用しても良い。

B.5 節で JT-H263 ビデオストリームのシグナリングを利用するための手順が記述されている。

## 付属資料 F： オーディオと多重パケット化

本付属資料では、オーディオコーデックに関する RTP パケット化の詳細について記述する。付表 F.1/JT-H225.0 では、本標準に定義されないオーディオパケット化フォーマットの定義について言及する。付表 F.2/JT-H225.0 では、多重パケット化フォーマットの定義について言及する。この付属資料の残りの章では追加オーディオパケット化フォーマットを定義する。

付表 F.1/JT-H225.0 外部定義されたオーディオパケット化フォーマット

(ITU-T H.225.0)符号化方式名	パケット化定義
ISO/IEC 14496-3 (MPEG-4 Audio)	ISOC/IETF RFC 3016, MPEG-4 ストリームのための RTP ペイロードフォーマット

付表 F.2/JT-H225.0 外部定義された多重ストリームパケット化フォーマット  
(ITU-T H.225.0)

符号化方式名	パケット化定義
H.222 multiplexed streams (MPEG-2 transport streams)	ISOC/IETF RFC 2250, 1998 年 1 月版 MPEG1/MPEG2 ビデオのための RTP ペイロードフォーマット

### F.1 JT-G723.1

この標準は、非常に低いビットレートでマルチメディアサービスの音声信号部分を圧縮する事に対して使用することができるコード化表現を記述する。JT-G723.1 フレームは次の3つのサイズの中から、1つのサイズを取りうる：24バイト(6.3 kbit/s フレーム)、20バイト (5.3 kbit/s フレーム)、4バイト。これら4バイトフレームはSID フレーム(無音挿入記述子)と呼ばれ、擬似背景雑音パラメータの明記に使用される。4、20、24バイトフレームの混在について制限はない。フレームの先頭オクテットの最下位2ビットでフレームサイズとコーデックタイプを決定する。(ビットオーダー上のより多くの情報のために表 5/JT-G.723.1 と表 6/JT-G.723.1 を参照) 30mc 間隔において2つのレートを切り替える事は可能である。(5.3 kbit/s と 6.4 kbit/s) のレート共に符号化、復号の部品は必須である。この符号器は、上記レートで、電話に近い品質の音声を表現するために、制限された複雑さの中で最適化される。

符号化されたビットストリームの全ビットは、常に最下位ビットから最上位ビットに向かって送信される。この参照はトランスポートレイヤのビット送信順序を示すもので、伝送路でのビット送信順序を示すものではないことに注意すること。

JT-G723.1 パケットは付属資料 B に従うが、パケット間隔 (30mc 対 20mc のデフォルト) は除外する。

- 1) トークスパートの最初のパケット (無音周期の後の最初のパケット) は RTP データヘッダ内のマーカビットをセットする事により識別される。
- 2) サンプリング周波数 (RTP クロック周波数) は 8000Hz である。
- 3) パケット間隔は 20ms のデフォルトパケットに対して 30ms (1 フレーム) の継続時間をもたなければならない。
- 4) コーデックは、1つのパケットの中でいくつかの連続したフレームを符号化、復号可能であるべきである。
- 5) デフォルトのパケットの長さは 0 から 200ms であるが、受信側は 0 から 180ms の長さの音声データパケットを受け入れるべきである。

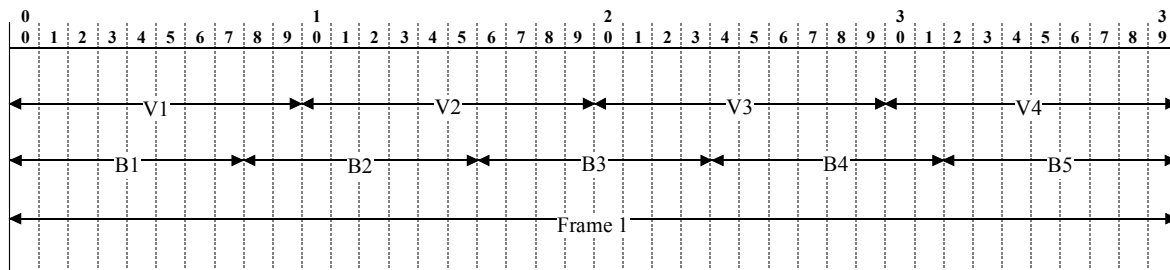
### F.2 JT-G728

- 1) フレームのパケット化:

1つの JT-G728 フレーム (4 ベクトル: V1~V4、それぞれ 10 ビット。V1 が時間的に古く、最初に処理さ



れる)は B1 から B5 の 5 バイトで構成される。次の図に示すように、ビットの並べ方の規則は「ビット順位を保つ事」である。時間的に古いベクトルからのビットは新しいベクトルからの物よりも上位である。フレームの最上位ビット (MSB) は B1 の MSB に入り、フレームの最下位ビット (LSB) は B5 の LSB に入る。補足説明:それぞれのベクトルの上位側のビットは B1 から B5 の上位側に入れられる。(B の番号が小さい程、上位である。)



T1529850-98

例 :

B1 は V1 の上位 8 ビットを含み、B1 の MSB が V1 の MSB である。

B2 は V1 の下位 2 ビットを含み、その 2 ビットの上位ビットが B2 の MSB である。また、V2 の上位 6 ビットの中の最上位ビットが、B2 の上位側に入る。

B1 は最初の packets (RTP の最上位バイト) に存在しなければならず、B5 は最後でなければならない。

## 2) マルチフレームパケット化 :

1 つの RTP パケット内で 1 つのフレームを送るとかなりのネットワークのオーバーヘッドが発生するかもしれない。従って、以下の方法でマルチフレームパケットを送る事が許されている。

1 つの RTP JT-G728 パケットは全てのフレーム番号を含まねばならない。

最初に処理される時間的に古いフレームは最初に RTP パケットに入れられねばならない。

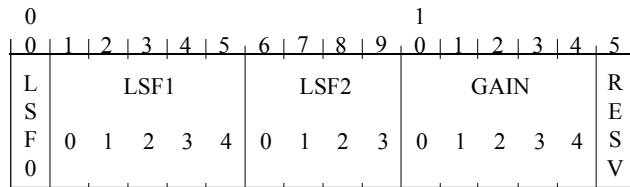
パケット内の一番古い情報データである最初のフレームの最初のベクトル(V1)の中で、タイムスタンプは、最初のサンプルの取り込み時間を示すだろう。

## 3) マーカービットは JT-H225.0 の中でアサインされた物と同じ意味を持たねばならない。

## F.3 JT-G729

この標準は、8kbit/s のビットレートでマルチメディアサービスの音声信号部分の圧縮に使われることのできる符号化表現を記述している。この符号器は、8kbit/s での長距離電話または無線通話の品質の音声を表現するように最適化された。この符号器は、ランダムで突発的な消去フレームに対しても同様にランダムなビットエラーに対しても本来的に耐性をもっている。これは、雑音の多い環境で動作させる時、高品質で音声を表現する。JT-G729 アルゴリズムの複雑さを削減したバージョンは、標準 JT-G729 付属資料 A で明記される。これら 2 つのアルゴリズムの浮動小数点版は JT-G729 の付属資料 C に明記される。JT-G729 のメインボディと JT-G729 付属資料 A と C における音声符号化アルゴリズムは、お互いに十分互換性がある。そのため、それらの間で厳密な認識をする必要はない。

JT-G729 付属資料 B の有音検出 (VAD) と擬似背景雑音発生器 (CNF) アルゴリズムが推奨されている。このアルゴリズムは、JT-G729 付属資料 F (6.4kbit/s with VAD/CNG) 、JT-G729 付属資料 G (11.8kbit/s with VAD/CNG) 、JT-G729 付属資料 B (JT-G729 と JT-G729 付属資料 A with VAD/CNG) と JT-G729 付属資料 I に適用される。JT-G729 または JT-G729 付属資料 A フレームは 10 オクテット、JT-G729 付属資料 D フレームは 8 オクテット、JT-G729 付属資料 E フレームは 15 オクテット含み、そして JT-G729 付属資料 B、F、G の擬似背景雑音フレームは、付図 F.3.1/JT-H225.0 に示されるように 2 オクテット占める。

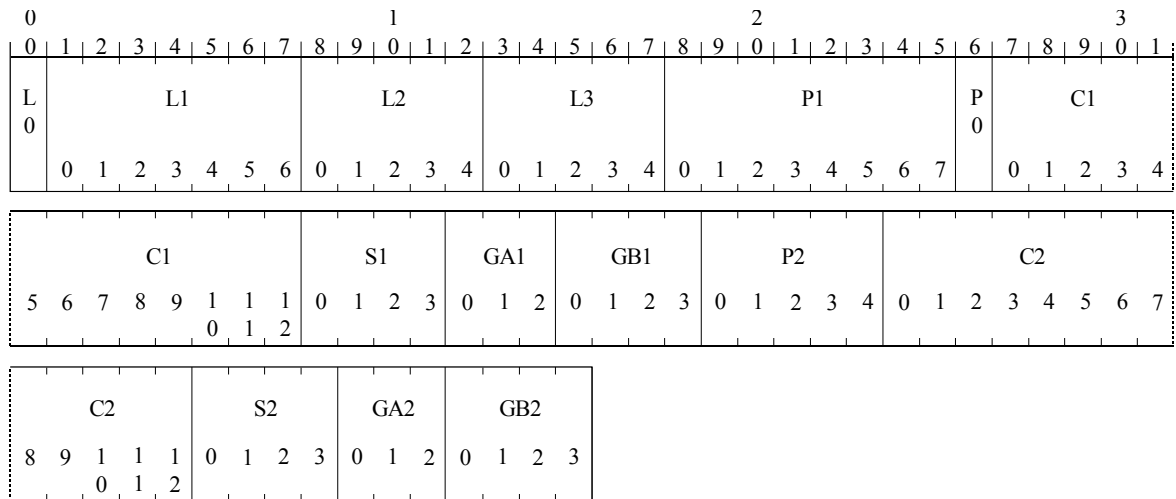


RESV = Reserved (zero)

T1529860-98

付図 F.3.1/JT-H225.0 JT-G729 付属資料 B、F、G CNG パケット化フォーマット (ITU-T H.225.0)

JT-G729、JT-G729 付属資料 A または JT-G729 付属資料 C の 80 ビットからなる 10mc フレームの転送パラメータは、標準 JT-G729 の表 8/JT-G729 に定義されている。これらのパラメータのマッピングは、付図 F.3.2/JT-H225.0 のように与えられる。ビットは、最上位ビットが 0 でインターネットオーダーで番号付けされている。

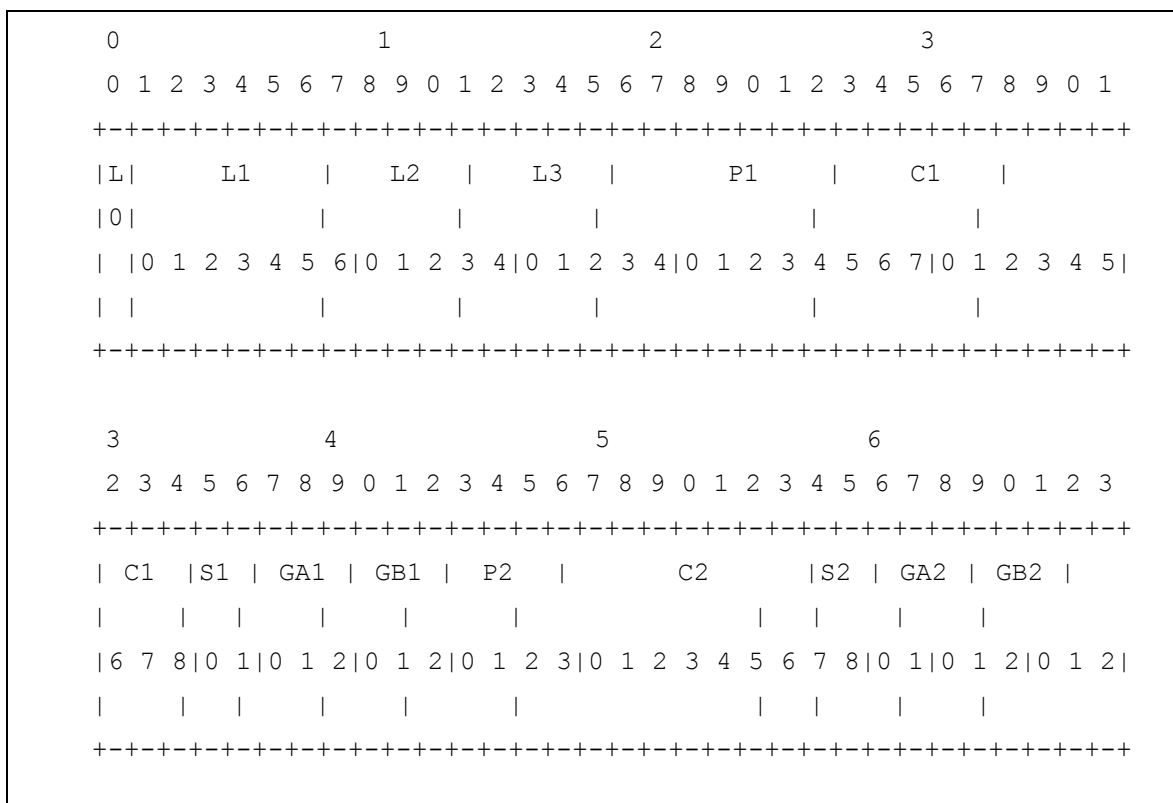


T1529870-98

付図 F.3.2/JT-H225.0 JT-G729、JT-G729 付属資料 A、C パケット化フォーマット (ITU-T H.225.0)

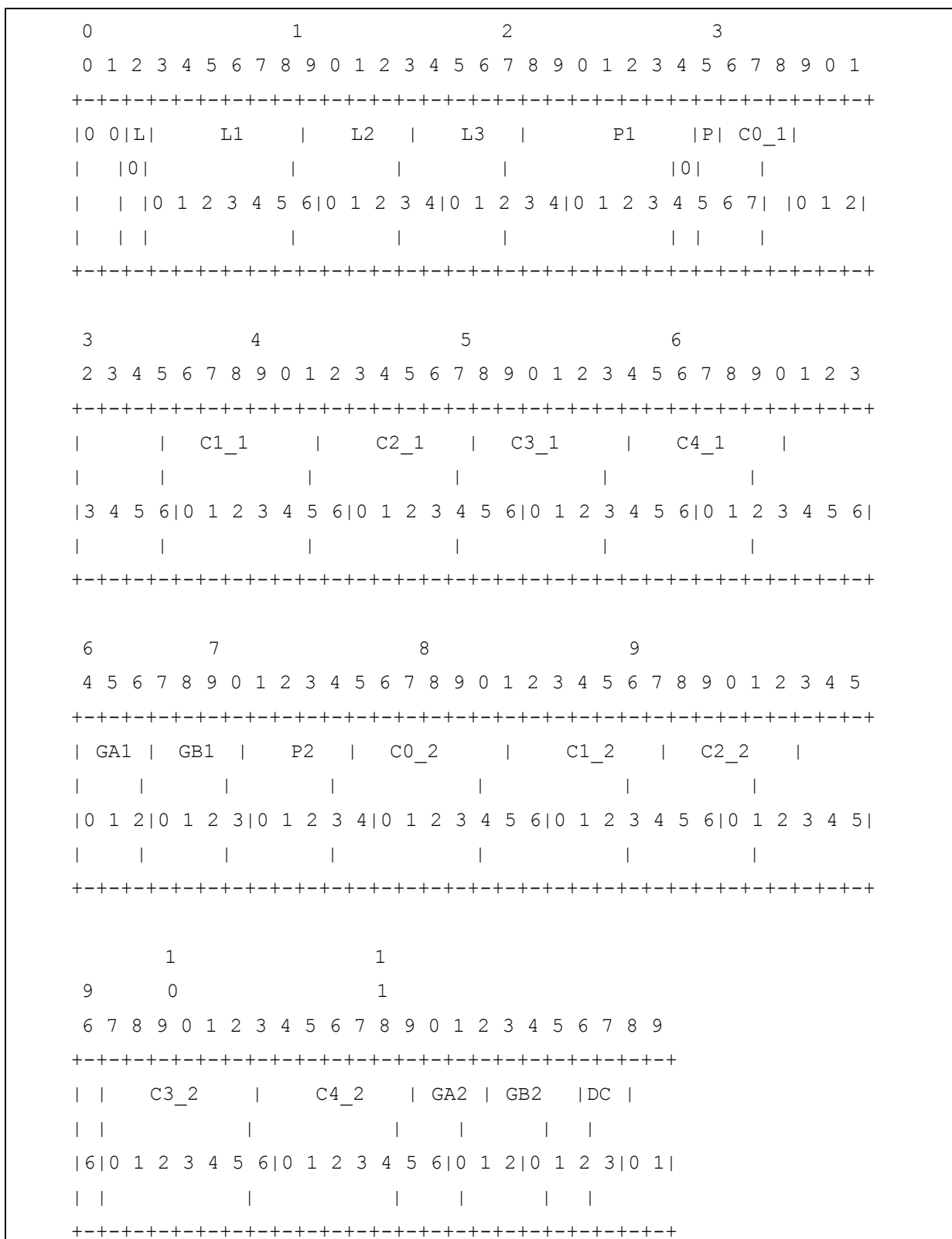
JT-G729 付属資料 D は、チャンネル容量における瞬間的な減少に関する JT-G729 の 6.4kbit/s レート拡張を定義する（例えば、過負荷条件の扱い）。JT-G729 付属資料 E は、より良い性能のために JT-G729 の 11.8kbit/s レート拡張により、背景雑音を伴う音声や音楽のような広帯域の入力信号を提供する。さらに JT-G729 付属資料 E は、逆方向適応と順方向適応の 2 つの動作モードがある。それはパケットヘッダの最初の 2 ビットで示される。

JT-G729-6.4kbit/s フレームのビットは、付図 F.3.3/JT-H225.0 に示されるような形式である（JT-G729 付表 D.1 参照）。ビットは、インターネット順で番号付けされる。つまり最上位ビットはビット 0 である。トータルで 64 ビットが使用される。

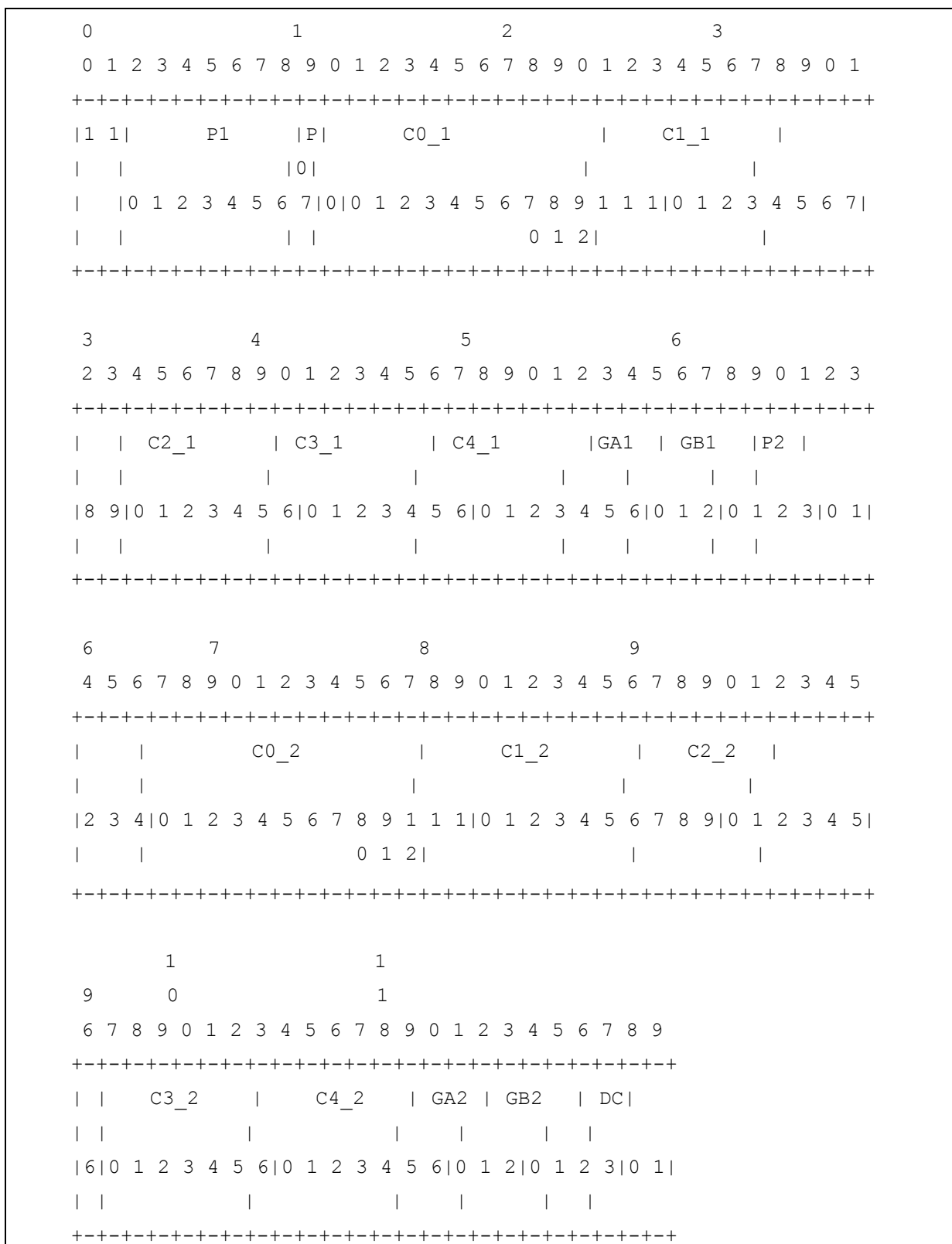


付図 F.3.3/JT-H225.0 JT-G729-6.4kbit/s パケット化フォーマット

(ITU-T H.225.0) JT-G729 付属資料 E アルゴリズムに関するネットワークのビットレートは 11.8kbit/s であり、トータルで 118 ビット使用される。JT-G729-11.8kbit/s フレームのビットは、付図 F.3.4/JT-H225.0、付図 F.3.5/JT-H225.0 に示されるような形式である (JT-G729 付表 E.1 参照)。付図 F.3.4/JT-H225.0、付図 F.3.5/JT-H225.0 は、JT-G729 付属資料 E アルゴリズムに対して個々に、順方向適応モードと逆方向適応モードに関するフィールドについて記述している。2つの最下位ビットは“don't care”ビットとして含まれ、フレームのオクテット単位で完全な整数とするために使用される。(TTC 注：フレームの境界がオクテット単位になることを保証するために使用される)



付図 F.3.4/JT-H225.0 JT-G729-11.8kb/s 順方向適応モードのパケット化フォーマット (ITU-T H.225.0)



付図 F.3.5/JT-H225.0 JT-G729-11.8kbit/s 逆方向適応モードの packets 化フォーマット (ITU-T H.225.0)

RTP パケットは、0 または 1 つの JT-G729 付属資料 B ペイロードに続く、0 または 1 以上の JT-G729 または JT-G729 付属資料 A、C、D、E フレームで構成することができる。擬似背景雑音フレームの存在は、RTP ペイロードの長さから推論されることできる。

- 1) トークスパート（無音期間の後の最初の packets）の最初の packets は、RTP ヘッダーにマーカービット置くことで識別される。
- 2) サンプリング周波数（RTP クロック周波数）は 8000Hz である。

- 3) デフォルトパケット化間隔は、20ms の持続時間を持つべきである。20ms が強く推奨された値であるが、いくつかの状況において 10ms パケットを送るよう要求してもよい。例えば、パケットの最初の 10ms の音声から非音声への遷移を考えなさい。もし、20ms パケット化間隔が必須であるなら、送信側では音声が再びアクティブになるまで待つ必要があるだろう。
- 4) コーデックは、単一パケット内にいくつかの連続フレームを符号化し復号することができるべきである。
- 5) 受信部は、音声データの 0ms から 200ms 間で表現するパケットを受け取るべきである。

#### F.4 無音抑圧

本標準では、符号器が無音期間で伝送を停止する前に無音フレームを送るかもしれないと言う事を述べている。全ての音声符号器が無音状態の為にインバンドシグナリングを持つとは限らないので、RTP レベルでの一般的なメカニズムが決められるべきである。一例は、空の RTP の送出かもしれない。この件は今後の検討課題である。

#### F.5 GSMコーデック

GSM 音声コーデックは、GSM フルレート (FR) [F-1]、GSM ハーフレート (HR) [F-2]、GSM エンハンスドフルレート (EFR) [F-3]を含む。それぞれのコーデックは、3 つの異なる音声トラフィックフレームを生成する。

- ・ 音声フレーム—実際の音声データを含む
- ・ アイドルフレーム—活動的な音がないことを示す。全てのビットは、1 にセットされる
- ・ 無音記述子 (SID) フレーム—無音の期間を示し、データは背景雑音を表す。SID フレームは、固定のビットパターンでインバンドにマークされる。

##### F.5.1 フレームパケット化

3 つの GSM 音声コーデックのトラフィックフレームは、有効なビットは先頭 (MSB) で RTP フレームにパケット化される。1 つの RTP パケットは、1 つまたはそれ以上の GSM 音声トラフィックフレームを含んでもよい。全てのエンドポイントは、アイドルフレームを受信し識別できなければならない。アイドル GSM 音声フレームは、2 進数の 1 で埋められている。

エンドポイントが `comfortNoise` パラメータを TRUE に設定した場合、特別な GSM コーデックの仕様に指定されたように擬似背景雑音と途切れた伝送 (DTX) として SID を送信しなければならない。無音期間の間、新しい SID フレーム (おそらく) 雑音情報が更新されたものは、24 フレーム毎に送られる。無音期間の後、RTP ヘッダのマーカービットは、1 に設定されなければならない。

##### フルレートコーデック :

GSM フルレートコーデックは、20ms 毎に 260 ビット (32.5 オクテット) 送信される。この情報は、シグネチャと呼ばれる 4 ビットの接頭辞 (0xD または 2 進数の 1101) で RTP フレームにパケット化されなければならない。従って、RTP 中の GSM FR ペイロードは、33 オクテットでなければならない。SID (無音記述子) フレームは、下の参照 [F-4] において示されたコーデックパラメータに格納される SID コードワードによってインバンドでマークをつけられる。SID フレームのサイズは、33 オクテットである。フルレートの SID のシグネチャは、フルレート音声フレームと同じでなければならない (0xD)。RTP コード化された FR 音声は、パケット化のオーバーヘッドを含まないで 13200 bit/s のビットレートでなければならない。

##### ハーフレートコーデック :

GSM ハーフレートコーデックは、20ms 毎に 112 ビット (14 オクテット) 送信される。この情報は、接頭辞/シグネチャをまったく持たない RTP フレームにパケット化されなければならない。SID フレームは、下の参照 [F-4] において示されたコーデックパラメータに格納される SID コードワードによってインバンドでマークをつけられる。SID フレームのサイズは、14 オクテットである。RTP コード化された音声は、パケット化のオーバーヘッドを含まないで 5600 bit/s のビットレートでなければならない。

##### エンハンスドフルレート :

GSM EFR コーデックは、20ms 毎に 244 ビット (30.5 オクテット) 送信される。この情報は、シグネチャと呼ばれる 4 ビットの接頭辞 (0xC または 2 進数の 1100) で RTP フレームにパケット化されなければならない。従って、RTP 中の GSM EFR ペイロードは、31 オクテットでなければならない。SID (無音記述子) フレームは、下の参照 [F-4] において示されたコーデックパラメータに格納される SID コードワードによってインバンドでマークをつけられる。SID フレームのサイズは、31 オクテットである。RTP コード化された EFR 音声は、パケット化のオーバーヘッドを含まないで 12400 bit/s のビットレートでなければならない。

## F.5.2 関連文献

- [F-1] GSM 06.10 (ETS 300 961): "Full rate speech; transcoding"
- [F-2] GSM 06.60 (ETS 300 726): "Enhanced Full Rate (EFR) speech transcoding"
- [F-3] GSM 06.20 (ETS 300 969): "Half rate speech; Half rate speech transcoding"
- [F-4] ETSI, TIPHON 03 001 (TS 101 318): "Using GSM speech codecs within H.323"
- [F-5] GSM 06.31 (ETS 300 964): "Full rate speech; Discontinuous Transmission (DTX) for full rate speech traffic channels"
- [F-6] GSM 06.81 (ETS 300 729): "Discontinuous Transmission (DTX) for Enhanced Full Rate (EFR) speech traffic channels"
- [F-7] GSM 06.41 (ETS 300 972): "Half rate speech; Discontinuous Transmission (DTX) for half rate speech traffic channels"
- [F-8] GSM 06.12 (ETS 300 963): "Full rate speech; Comfort noise aspect for full rate speech traffic channels"
- [F-9] GSM 06.62 (ETS 300 728): "Comfort noise aspects for Enhanced Full rate (EFR) speech traffic channels"
- [F-10] GSM 06.22 (ETS 300 971): "Half rate speech; Comfort noise aspect for half rate speech traffic channels"
- [F-11] GSM 08.60 (ETS 300 737): "Inband control of remote transcoders and rate adaptors for Enhanced Full Rate (EFR) and full rate channels"

TTC 注：参照規格に"Digital cellular telecommunications system;"が入っているが上記一覧では省略した。

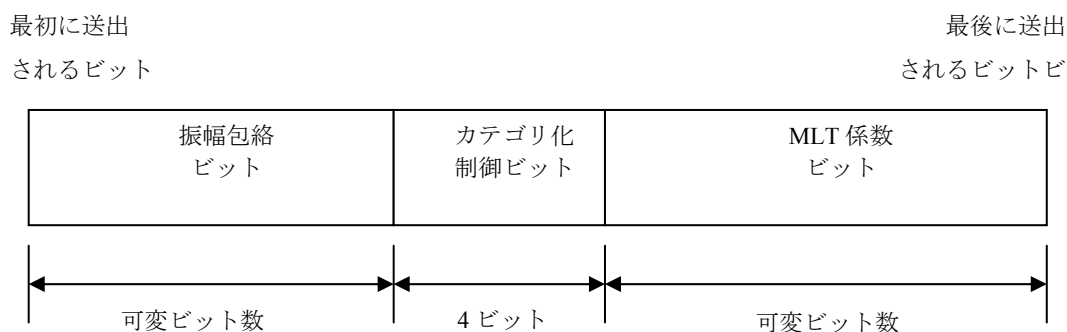
## F.6 JT-G722.1

TTC 標準 JT-G722.1 で定義されたオーディオ符号化アルゴリズムは、50Hz から 7kHz までの帯域幅をもつ広帯域オーディオ信号を、20ms フレーム長と 16kHz のサンプリング周波数で、24kbit/s あるいは 32kbit/s のいずれかのビットレートで符号化する。ビットレートは任意の 20ms フレーム単位で変更可能である。ただし、ビットレートの変更はビットストリーム中のインバンドにおいて通知されない。24kbit/s で動作する場合、480 ビット（60 オクテット）がフレーム毎に生成され、32kbit/s で動作する場合、640 ビット（80 オクテット）がフレーム毎に生成される。このように、両方のビットレートがパディングビットを必要とすることなくオクテット単位で配列を可能にしている。

フレーム単位のビット数は固定である。しかしながら、この固定フレームの中で JT-G722.1 では、符号化されたパラメータの大部分を表現するために可変長符号化（例えばハフマン符号化）が使用される。カテゴリ化制御ビットパラメータを除いて、すべての他のビットストリームパラメータは可変長符号、すなわち可変ビット数によって表現される。付図 F.6.1/JT-H225.0 は、この点を示すと共に送信パラメータフィールドの順序を示している。すべての可変長符号とカテゴリ化制御ビットは、最も左部分の（最上位 - MSB）ビットから最も右部分の（最下位 - LSB）ビットへの順で送信される。ハフマン符号化を使用するという事は、最初にフレームすべての復号を完結しなければ、ビットストリームに含まれる各種の符号化パラメータ/フィールドを特定することができないことを意味する。

付図 F.6.2/JT-H225.0 は、JT-G722.1 ビットストリームがどのようにオクテット配列のされた RTP ペイロードに割り当てられているかを示している。符号化ビットストリームは、（ビットレートに依存して 60 あるいは 80）オクテットのシーケンスに分割され、それぞれのオクテットが RTP オクテットに順番に配置される。

一つの RTP パケットは同一ビットレートの JT-G722.1 フレームのみを含むものでなければならない。RTP タイムスタンプは 1/16000 秒単位でなければならない。



付図 F.6.1/JT-H225.0 JT-G722.1 主要なビットストリームフィールドとその送信順序

(ITU-T H.225.0)



付図 F.6.2/JT-H225.0 RTP へマッピングされた JT-G722.1 符号化ビットストリーム (ITU-T H.225.0)

F.7 TIA / EIA - 136 ACELP

このボコーダは、TIA / EIA - 136 TDMA デジタルセルラーと PCS システムのために最適化されている。このボコーダは、有音検出 (VAD) や損失フレーム補完、擬似背景雑音発生器 (CNG) 能力を持つ。サンプリングレートは 8000Hz であり、圧縮後の音声フレーム長は 20ms である。このボコーダは、148 ビット音声ベクトル、s0 から s147 を 20ms の音声フレーム毎に生成する。s0 は最上位ビット (MSB) である。詳細は、参考文献 [F.7.1] 4 章を参照のこと。

F.7.1 TIA / EIA - 136 ACELP フレームフォーマット

音声インジケータフラグビット SP はボコーダによって生成されなければならない。音声フレームを示すには「1」をセットし、無音 (擬似背景雑音) フレームを示すには「0」をセットしなければならない。この SP フラグビットは 148 番目のビットに挿入されなければならない。149 番目のビットは BFI\_CN (誤りのあるフレームあるいは擬似背景雑音表示) で、150 番目のビットは CNU (擬似背景雑音更新) フラグである。151 番目のビットは常に 0 にセットされなければならない。

これらの 3 つのフラグによる論理的な組み合わせは次に示される通りである。

152 ビット (19 オクテット) の送信フレームは次の付図 F.7.1/JT-H225.0 で示される。オクテットは LSB から始まり、MSB の方へ構成される。LSB は先に送信される。

ビット 0 (MSB)	1 ... 146	147	148	149	150	ビット 151 (LSB)
S0	s1... s146	S147	SP	BFI_CN	CNU	常に 0
音声ベクトル/擬似背景雑音			フラグ	フラグ	フラグ	パディングビット

付図 F.7.1/JT-H225.0 ACELP ボコーダ音声フレーム (ITU-T H.225.0)

F.7.2 TIA / EIA - 136 ACELP 無音抑圧モード

無音モードでは、ボコーダは周囲の雑音フレームの表現を生成する。このフレームは、送信端の周辺雑音を再表現するために受信端のボコーダによって使用される。CN (擬似背景雑音) パラメータベクトルは 38 ビットで構成され、3 つのフラグビットと (すべて 0 から成り立つ) 7 つのパディングビットが付加され、6 オクテットフレームを形成する。

48 ビット (6 オクテット) CN フレームは次の付図 F.7.2/JT-H225.0 で示される。オクテットは LSB から始まり、MSB の方へ構成される。LSB は最初に送信される。



ビット 0 (MSB)	1 ... 37	38	39	40	41	41-47 (LSB)
Cn0	cn1... cn37	S147	SP	BFI_CN	CNU	常に 0
音声ベクトル／擬似背景雑音			フラグ	フラグ	フラグ	パディングビット

Key:

SP - 音声識別子

BFI\_CN - 誤りのあるフレーム識別子／擬似背景雑音識別子

CNU - 擬似背景雑音更新

これらのフラグの論理値とそれらを意味するものは次のように定義される：

SP: 1 = 音声フレーム; 0 = 非音声 (擬似背景雑音フレーム)

BFI\_CN:

If SP = 1

And BFI\_CN = 1

すると、これは誤りのある音声フレーム。

他ならば (BFI\_CN = 0), これは誤りのない音声フレーム。

If SP = 0

And BFI\_CN = 1

すると、これは誤りのある擬似背景雑音フレーム。

他ならば (BFI\_CN = 0), これは誤りのない擬似背景雑音フレーム。

CN:

If SP = 0

And BFI\_CN = 0

And CN = 1

すると、これは擬似背景雑音更新フレーム。

他ならば、無効 CN フレーム。

注：

移動体無線ボコーダでは BFI\_CN を 0 にセットしなければならない。もし受信基地局が無線チャンネルによってもたらせたエラーを修正できない場合、このフラグを 1 にセットしてもよい。

### 付図 F.7.2/JT-H225.0 ACELP ボコーダ無音抑圧フレーム (ITU-T H.225.0)

#### F.7.3 TIA / EIA - 136 ACELP パケット化

IS - ACELP のパケット化は、付属資料 B の形式でなければならない。

1. パケット化継続時間は、全て 20ms の倍数でなければならない。
2. パケットは、それぞれが 1 以上のフレームから構成してよい。
3. コーデックは、単一パケット内の連続した数フレームを符号化、復号化することが可能であるべきである。
4. 符号化されたビットストリームの全てのビットは、最下位ビットから最上位ビットへ方向で送られる。

#### F.7.4 TIA / EIA - 136 ACELP の参照する標準

[F.7.1] TIA/EIA-136-part 410 - TDMA Cellular/PCS - Radio Interface, Enhanced Full Rate Voice Codec (ACELP). Formerly IS-641.

## F.8 TIA/EIA-136 USI

このボコーダは、TIA/EIA-136 TDMA デジタルセルラー及び PCS システムに最適化されている。参考文献 [F.8.1]は、ボコーダの詳細について記述している。

### F.8.1 TIA/EIA-136 USIフレームフォーマット

サンプリング周波数は 8000Hz で、圧縮された音声フレーム長は 20ms である。ボコーダは音声フレームあたり、244 個の一連のビット列を生成する。BFI、SID、TAF の 3つのフラグビットは、音声ベクトルに付加される。247 のビット位置に 1つのパディングビットが整数オクテット (31) を形作るために付け加えられる。最終ビットが、最下位ビット (LSB) として参照される。このボコーダは、DTX (不連続伝送) 無音モードもサポートする。

伝送音声フレーム構造は、以下の様に示される。

MSB – bit 0	1 ... 243	244	245	246	247 (LSB)
s0	s1 ... s243	BFI	SID	TAF	常に 0
音声ベクトル		フラグ	フラグ	フラグ	パディングビット

付図 F.8.1/JT-H225.0 USI ボコーダ音声フレーム  
(ITU-T H.225.0)

### F.8.2 TIA/EIA-136 USI 無音モードフレーム (TX-DTX)

無音モードでは、SID (無音記述子) フレームと呼ばれる特別なフレームが、参考文献[F.8.1]の 1.3 節に記載されているスケジュールで伝送される。

SID フレームのビット数と通常の音声フレームのビット数は同じであるが、ビットマップは違っている。詳細は、参考文献[F.8.1]参照のこと。SID フレームは擬似背景雑音 (CN) パラメータと、95 ビット SID コードワードを含んでいる。SID コードワードは全て 0 である。244 ビットベクトルペイロード中の他の未使用ビットもまた、0 にセットされる。

MSB – bit 0	1 ... 243	244	245	246	247 (LSB)
cn0	cn1 ... cn243	BFI	SID	TAF	常に 0
擬似背景雑音ベクトル		フラグ	フラグ	フラグ	パディングビット

付図 F.8.2/JT-H225.0 基地局から有線側(Landline)への擬似背景雑音伝送フレーム(USI)  
(ITU-T H.225.0)

BFI、SID、TAF フラグのロジックは、F.7 節に記載されている TIA/EIA-136 ACELP ボコーダの相当するフラグと同様である。

### F.8.3 TIA/EIA-136 USI パケット化

パケット化は、付属資料 B の全てと一致していなければならない。

パケット化継続時間は、全て 20ms の倍数でなければならない。

伝送パケットは 0、1 またはそれ以上のフレームで構成してよい。

コーデックは単一伝送パケット内の連続した数フレームを符号化、復号化することが可能であるべきである。

符号化されたビットストリームの全てのビットは、最下位ビットから最上位ビットへの方向で送らなければならない。

### F.8.4 TIA/EIA-136 USI の参照する標準

[F.8.1] TIA/EIA-136-part 430-TDMA Cellular/PCS-Radio Interface,US1 Full Rate Voice Codec.

## F.9 IS-127 EVRC

### F.9.1 IS-127 EVRC 解説

#### F.9.1.1 概説

TIA/EIA IS-127 エンハンスト可変レートコーデック (EVRC) は、TIA/EIA IS-95 CDMA デジタルセルラー及び PCS システムに最適化されている。サンプリング周波数は、8000 サンプル/s で音声フレーム長は 20ms

である（フレームあたり、160 サンプルになる）。EVRC は、有音を、フルレート又はハーフレートで符号化し、背景雑音（音声のない部分）を 1/8 のレートで符号化する。これは、非常に低い平均レートで、電話品質音声を伝達する。EVRC コーデックの詳細記述は、公開されている TIA/EIA IS-127 標準[F.9.1]の中で参照することが出来る。

#### F.9.1.2 圧縮率

EVRC エンコーダは、フルレート（レート 1）、ハーフレート（レート 1/2）、1/8 レート（レート 1/8）の 3 つのレートの 1 つを使って入力信号を圧縮する。1/8 が背景雑音（無音モード）の符号化に用いられるのに対し、フル及びハーフレートは、主に有音の符号化に用いられる。全てのフレーム長は符号化レートに関わらず、20ms である。

#### F.9.1.3 空のケット

インバントシグナリングやセカンダリトラヒック（文献[F.9.1]の 1.4.1 節を参照のこと）を許すために、音声フレームを空にする。生成された音声ケットは使用されず、デコーダはそれを消去されたケットとして扱う。詳細は文献[F.9.1]を参照のこと。

#### F.9.1.4 ハーフレート

シグナリングメッセージをトラヒックチャンネルに付加しなければならない時は、通常のフルレートの代わりにハーフレート符号化が用いられる。

#### F.9.1.5 1/8 レート、マルチトラヒックチャンネルデータ

全てのビットが 1 に設定された、1/8 レートケットは、マルチトラヒックチャンネルデータと見なされる。その様なケットは、「消去されたケット（erased packets）」と申告され、文献[F.9.1]の 5 章に記述されている様に処理される。

TIA/EIA IS-95 に従い、レート情報とチャンネル符号化ビットは、無線伝送の為にボコーダ出力ビットに付加される。

ケットタイプ、ケットあたりのビット数、ボコーダ自体のビットレート及び総合されたレート（ボコーダのビットレートと付加ビットの和）は、付表 F.9.1/JT-H225.0 の様に示される。

**付表 F.9.1/JT-H225.0 EVRC パケットとビットレート  
(ITU-T H.225.0)**

パケットタイプ (3 bits)	レート	パケットあたり のビット数	ボコーダ ビットレート	総合レート
1	フル	171	8.55 Kbs	9.6 Kbs
2	ハーフ	80	4.0 Kbs	4.8 Kbs
3*	1/4 (サービスオプション1互換)	40		
4	1/8	16	0.8 Kbs	1.2 Kbs
5	ブランク	0	-	-
6	エラーを伴ったフルレート	171	-	-
7	不正フレーム (消去)	0	-	-

\* タイプ3のパケットは旧 IS-96 エンコーダでのみ、生成される。IS-127 デコーダは、これらのパケットを消去されたパケット (erased packets) として扱わねばならない。

### F.9.2 IS-127 EVRC パケット化

#### F.9.2.1 一般要求事項

伝送のパケット化は JT-H225 付属資料 B の形式でなければならない。

1. パケット化継続時間は、全て 20msec の倍数でなければならない。
2. 伝送パケットは 0、1 またはそれ以上のフレームで構成してよい。
3. コーデックは単一伝送パケット内の連続した数フレームを符号化、復号化することが可能であるべきである。
4. 符号化されたビットストリームの全てのビットは、最下位ビットから最上位ビットへの方向で送られなければならない。

#### F.9.2.2 フレームフォーマット

##### F.9.2.2.1 フルレート-F1

フルレート、176 ビット (22 オクテット) である EVRC は、付図 F.9.1/JT-H225.0 で示されるフレーム (F1) を送信する。オクテットは LSB から始まり、MSB の方へ構成される。その LSB (175 ビット) は最初に送信される。

ビット 0 (MSB)	ビット 1~ビット 170	ビット 171~ビット 175 (LSB)
s0	s1... s170	常に 0
音声ベクトル		パディング ビット

**付図 F.9.1/JT-H225.0 F1、フルレート EVRC フレーム  
(ITU-T H.225.0)**

##### F.9.2.2.2 ハーフレート-F2

ハーフレート、80 ビット (10 オクテット) である EVRC は付図 F.9.2/JT-H225.0 で示されるフレーム (F2) を送信する。オクテットは LSB から始まり、MSB の方へ構成される。その LSB (ビット 79) は最初に送信される。

ビット 0 (MSB)	ビット 1~ビット 79 (LSB)
s0	s1... s79
音声ベクトル	

**付図 F.9.2/JT-H225.0 F2、ハーフレート EVRC フレーム**

(ITU-T H.225.0)

F.9.2.2.3 1/8 レート-F3

1/8 レート、16 ビット (2 オクテット) である EVRC は下記の付図 F.9.3/JT-H225.0 で示されるフレーム (F3) を送信する。オクテットは LSB から始まり、MSB の方へ構成される。その LSB (15 ビット) は最初に送信される。

ビット 0 (MSB)	ビット 1~ビット 15 (LSB)
s0	s1... s15
音声ベクトル	

付図 F.9.3/JT-H225.0 F3, 1/8 レート EVRC フレーム (ITU-T H.225.0)

F.9.3 IS-127 EVRCの参照する標準

[F.9.1] TIA/EIA IS-127—Enhanced Variable Rate Codec, Speech Service Option 3 for Wideband Spread Spectrum Digital Systems, January 1997.

[F.9.2] TIA/EIA IS-95-B—Mobile Station-Base Station Compatibility Standard for Wideband Spread Spectrum Cellular Systems, March 1999.

F.10 JT-H223 MUX-PDUパケット化

F.10.1はじめに

TTC 標準 JT-H223 で定義されているように、JT-H223MUX-PDU は、データプロトコル、制御プロトコル、オーディオコーデック、ビデオコーデックのような高位レイヤエンティティ間で、1 つ以上の情報ストリームを交換するために設計されたパケット指向多重プロトコルによって使用される。それぞれの情報ストリームは 0 から 65,535 までの整数であるユニークな論理チャンネル番号によって識別される JT-H245 片方向論理チャンネルによって表される。

LCN0 は、JT-H245 の制御チャンネルに恒久的に割り当てられた論理チャンネルである。

他の全ての論理チャンネルは、JT-H245 の OpenLogicalChannel メッセージと CloseLogicalChannel メッセージを使用する送信側によって動的に開かれたり、閉じられたりする。

論理チャンネルの全ての必要な属性は OpenLogicalChannel メッセージ内に指定される。

折り返しチャンネルを要求するアプリケーションのために、双方向チャンネルを開く手順も TTC 標準 JT-H245 に定義される。

多重化部の一般的な構造は図 2/JT-H223 に示される。

多重化部は明確に分離された 2 つのレイヤ (多重化 (MUX) レイヤとアダプテーションレイヤ (AL) ) から構成される。

JT-H223 ペイロードタイプのサポートは、JT-H245 能力セットと JT-H245 の OpenLogicalChannel メッセージでの RTP 動的ペイロードタイプの使用を意味している。

F.10.2 MUX-PDUパケット化フォーマット

図 3/JT-H223 に示される JT-H223 の MUX-PDU は RTP 内のペイロードデータとして運ばれる。送信ビットの順序は JT-H223 の 3.2.2 節に示され、フィールドのマッピング規定は JT-H223 の 3.2.3 節に示される。

MUX-PDU は 2 つ以上の RTP パケットを使用することができるが、MUX-PDU は RTP パケットペイロードの最初のオクテットから開始しなければならない。

それぞれの RTP パケットは送信側の基準クロックから得たタイムスタンプを含む。

タイムスタンプは、JT-H223 の MUX-PDU の最初のバイトの目標送信時間を表さなければならない。このタイムスタンプの主要な目的は、受信側がネットワーク上で引き起こされるジッタを推定し、軽減できることと、一定のビットレートの JT-H223 ビットストリームを再生成できることである。

RTP ヘッダのフィールドの使用方法は、以下のようであればならない。

1. RTP 動的ペイロードタイプが使用される。
2. RTP タイムスタンプは、一定のビットレートチャンネル上でパケット内の MUX-PDU の最初のバイトの目標送信時間を表す。このタイムスタンプはデフォルト値 90kHz であるクロック周波数から得られる。送

信側は、この周波数を変更でき、選択された値は JT-H245 メッセージ内の **H223Capability** 構造中の **BitRate** パラメータによって示される。MUX-PDU が 1 つを越える RTP パケットを占める場合、RTP タイムスタンプは連続するパケット上で同じであるものとする。タイムスタンプは、送信された MUX-PDU に含まれるバイトの数に基づいて、算出されるべきである。

3. RTP ヘッダのマーカビットは、MUX-PDU の最終パケットにおいて 1 にセットされる。そうでなければ、0 でなくてはならない。したがって、以降のパケットが MUX-PDU の境界を検知することを待つ必要はない。

JT-H223MUX-PDU は RTP ヘッダに続き、次のようになる

RTP Header	MUX-PDU data
------------	--------------

## 付属資料G： 管理ドメイン間および管理ドメイン内の通信

### G.1 適用範囲

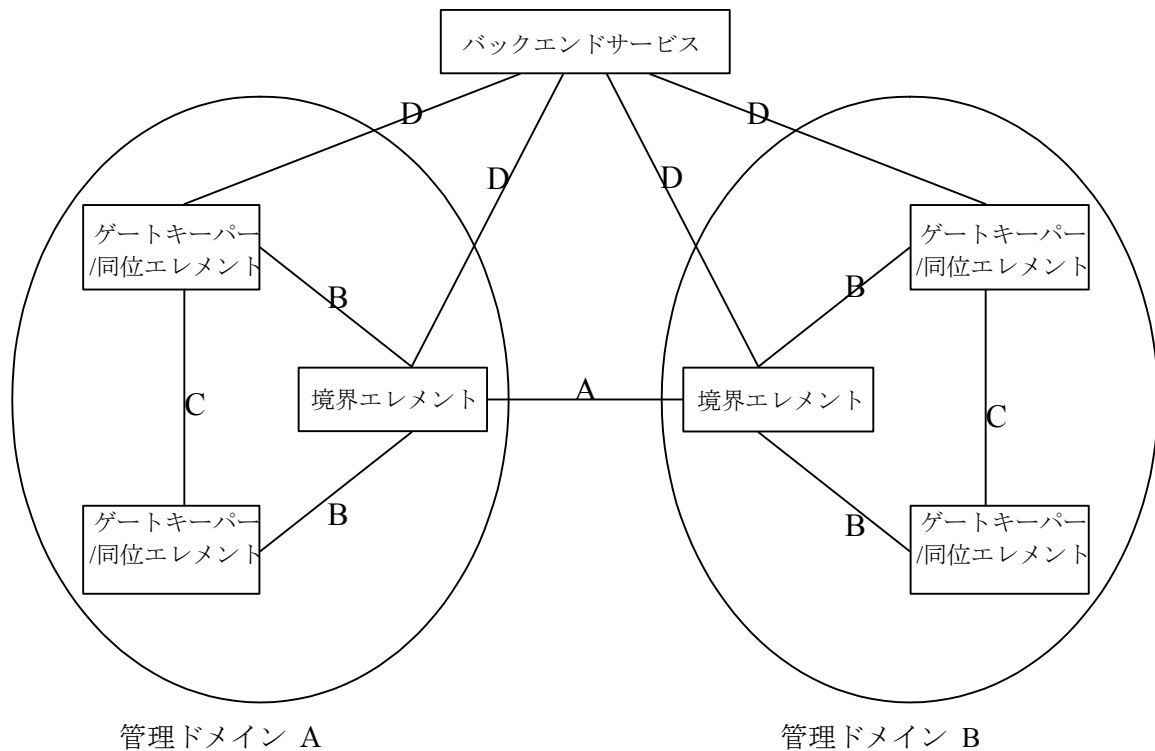
JT-H323 ネットワーク全体は、管理ドメインのように装置がなんらかの方法で組織化された、より小さなサブセットから構成されることが考えられる。JT-H323 ネットワーク中に存在する JT-H323 装置の台数が膨大なものとなる可能性があるため、管理ドメイン間の通話を接続させる効率的なプロトコルが必要とされている。最も基本的な例は、ある管理ドメイン内の利用者（エンドポイント）から、別の管理ドメインによってサービスを受けている利用者（エンドポイント）へ到達することである。JT-H225.0 の RAS プロトコルが管理ドメイン間の多くの通信要求を提供することができるとしても、それは本目的のためには完全でも効果的でも無い。

同じ理由により、同一の管理ドメイン内における JT-H323 要素間の通信のためにも、効率的なプロトコルが必要とされている。

この付属資料では、管理ドメイン間で通話を接続させることを目的とした、JT-H323 システムにおける管理ドメイン間および管理ドメイン内でのアドレス解決やアクセス認証、利用状況報告を可能にするための方法について述べる。この付属資料に記述されている手順に従って通信する JT-H323 要素のことを同位エレメントと呼ぶ。ある管理ドメインは、境界エレメントとして知られる論理要素を経由して、他の管理ドメインへそれ自身を公開する。境界エレメントは、少なくとも一つの同位が別の管理ドメインに属しているという意味で、同位エレメントの特殊なケースである。同位エレメントは、なんらかの他のエンティティ（例えば、ゲートキーパー）とともに配置されても良い。本付属資料では、管理ドメインに対しその組織構成やアーキテクチャについて詳細を明らかにすることを要求してはいない。本付属資料では、管理ドメイン内の特定のシステムアーキテクチャを要求するものではない。さらに本付属資料では、様々な呼モデル（ゲートキーパー経由型呼モデルや直接エンドポイント呼モデル）の利用をサポートしている。

基本的な手順は、それぞれが解決できるアドレスに関して、同位エレメントが情報交換することである。境界エレメントは、それぞれの管理ドメインにおいて解決できるアドレスに関する情報を交換する。アドレスは、一般的な方法またはより具体的な方法によって明示されることができる。追加情報は、通話における宛先として動作するために最適な管理ドメインを決定することを、管理ドメイン内のエレメントに可能にする。境界エレメントは、自身の公開アドレスへのアクセス制御を行ってもよく、また、これらのアドレスを介した通話の継続中に利用状況に関する報告を要求しても良い。

付図 G.1/JT-H225.0 は、JT-H323 ネットワーク内の様々なエレメント間でのシグナリングを表す、いくつかの参照点を示している。この図では、管理ドメインはシームレスなグローバルパケットネットワークの一部となっている。この図は JT-H323 システム機構の明確な定義ではないが、シグナリング参照点を図解するためのものであることに注意すること。



付図 G.1/JT-H225.0 システム参照点  
(ITU-T H.225.0)

付図 G.1 は、以下の参照点を示している。

- A — 異なる管理ドメインに属する境界エレメント間
- B — 同一管理ドメイン内の境界エレメントと同位エレメント間
- C — 同一管理ドメイン内の同位エレメント間
- D — JT-H323 エレメントとバックエンドサービス間 (本付属資料の範囲外)

参照点 A, B, C が、本付属資料における注目点である。前に述べたように同位エレメントがいくつかの他の JT-H323 エレメントとともに配置される可能性がある。

G.7 節では、理解補助のためのいくつかのシグナリングの例を示す。

## G.2 定義

本付属資料では次の用語を定義する。

管理ドメイン	管理ドメインは、1つの管理エンティティによって管理された JT-H323 エンティティの集合である。管理ドメインは、1つまたは複数のゲートキーパー (つまり、1つまたは複数のゾーン) から構成できる。
バックエンドサービス	バックエンドサービスは、ユーザ証明や認証、料金計算、課金、料金/使用料設定などの機能である。バックエンドサービスやバックエンドサービスにおける情報交換のためのプロトコルについては、(もし本付属資料におけるそれらと異なるならば) 本付属資料の範囲外である。
同位エレメント	JT-H501 で定義されるシグナリングメッセージを発信/着信する論理エレメント。このエレメントは他の JT-H323 エレメント、たとえばゲートキーパーやゲートウェイとともに配置されることがある。一つの管理ドメイン内に複数の同位エレメントが存在してもよい。
境界エレメント	境界エレメントは、同位エレメントの特殊なケースであり、他の管理ドメイン内に少なくとも一つの同位が存在するものをいう。境界エレメントは他のエレメントとのマルチメディア通信に伴う呼の成立や各種サービスの目的のために、管理ドメイン内への外部からのアクセスをサポートするための機能エレメントである。境界エレメントは、管理ドメインの (ネットワーク的な) 外部からの見え方を制御する。



クリアリングハウス 全てのアドレス解決を提供することが可能な（実際には境界エレメントの形態をとった）サービス。（すなわち、結合ポイントの一種）

（TTC注 一般的にクリアリングハウスとは、複数のインターネット電話サービス事業者間の通信を中継させることが可能な情報センターを意味する。各インターネット電話サービス事業者は、それぞれ固有のシステムでサービスを実現してきたため、ユーザ認証や課金などの運用管理方法が異なる場合がある。クリアリングハウスでは、これらの仕様の違いを吸収し、各事業者はクリアリングハウス・サービス上の共用の認証／課金／運用管理システムを利用可能となる。これにより、管理ドメイン間での精算等が可能になり、相互接続も可能となる。）

### G.3 略号

本付属資料では次の略号を使用する。

AD	管理ドメイン (Administrative domain)
CH	クリアリングハウス (Clearing house)
DST	夏時間 (Daylight Saving Time)
EP	エンドポイント (Endpoint)
GK	ゲートキーパー (Gatekeeper)
GW	ゲートウェイ (Gateway)
PE	同位エレメント (Peer element)
SCN	回線交換網 (Switched Circuit Network)
T	端末 (Terminal)

### G.4 参照

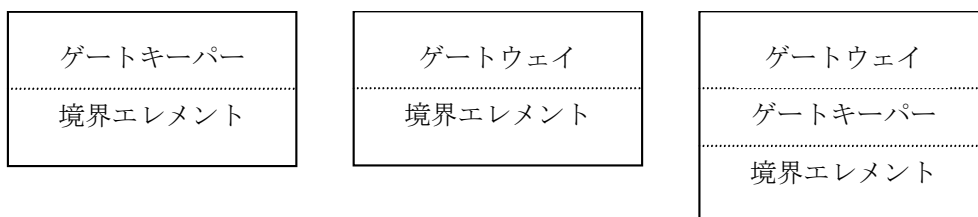
次の標準、および他の参照は本文書から参照される規定やこの標準の構成する規定を含んでいる。出版時に示された版は有効であった。全ての標準および他の参照は改訂を免れず、従って本標準のユーザーには下でリストされた標準、および他の参照の最も最近の版を適用することの可能性を調査することを奨励する。現在の有効な JT 標準のリストは規則正しく出版される。本標準中における文書の参照は、独立した文書であるため、標準の状態は示さない。

- TTC 標準 JT-H323, “パケットに基づくマルチメディア通信システム”
- TTC 標準 JT-H501, “マルチメディアシステムにおけるドメイン内／ドメイン間通信及びモビリティ管理プロトコル”
- TTC 標準 JT-H460.2, “H.323 ネットワークと SCN ネットワーク間の番号ポータビリティインタワーキング”

### G.5 システムモデル

本付属資料は、管理ドメイン間または管理ドメイン内に、特定のシステムアーキテクチャを要求することはない。以下の副節は、いくつかのサンプルアーキテクチャを提供するが、これらは厳密なものではなく説明のためのものである。

同位エレメントは、他の JT-H323 エレメントと共に存在できる機能エレメントであることに注意すること。付図 G.2/JT-H225.0 は、同位エレメントと他のエレメントを組み合わせた実装の例を示している。



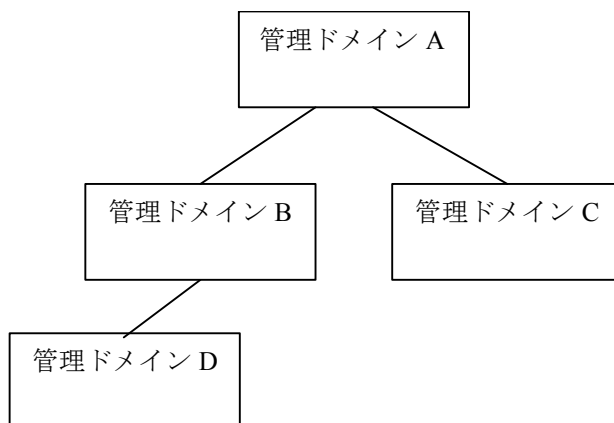
付図 G.2/JT-H225.0 同位エレメントの配置例  
(ITU-T H.225.0)

一般に、管理ドメインは任意の数のゾーンと任意の数の同位エレメントからなるとみなされる。管理ドメイン間および管理ドメイン内の同位エレメント間の関係は、多様な構成をとることができる。以下の副節は、関係と構成の例を示している。これらは管理ドメイン間のものとして記述されるが、階層的、分散型/フルメッシュおよび統合型の例は管理ドメイン内の同位エレメントを構成するのにも用いられる。

繰り返しになるが、下記の例はあくまで説明用であって、これ以外の構成を排除するものではないことに留意すること。

#### G.5.1 階層型構造

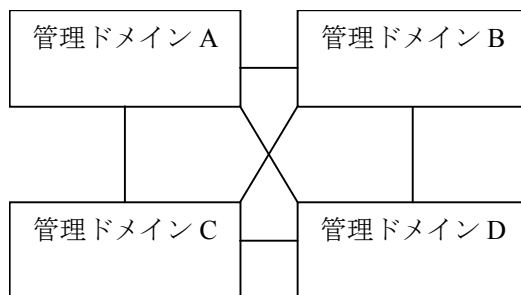
付図 G.3/JT-H225.0 は、管理ドメイン間のシンプルな階層型構造の配置を示している。このような配置では、ある管理ドメイン内の境界エレメントは、アドレス解決のため、より高い階層の管理ドメイン内の境界エレメントに問合せを行う。



付図 G.3/JT-H225.0 階層型構造の例  
(ITU-T H.225.0)

#### G.5.2 分散型またはフルメッシュ型構造

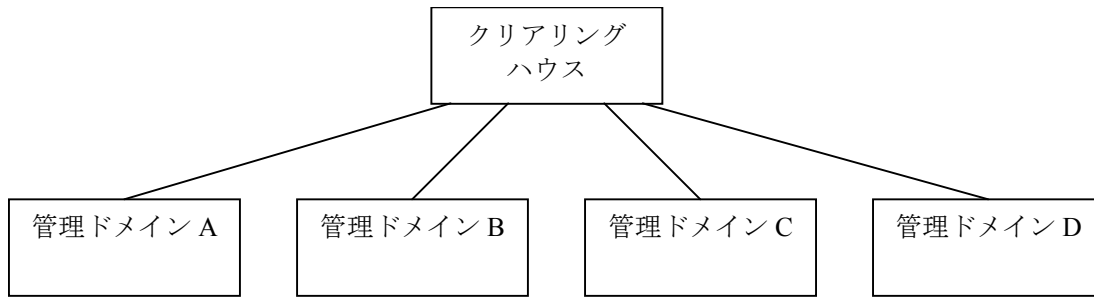
付図 G.4/JT-H225.0 のように、完全な分散型構造またはフルメッシュ型構造モデルが可能である。この例では、各管理ドメイン内の境界エレメントは、他の既知の管理ドメイン内の境界エレメントと通信する。



付図 G.4/JT-H225.0 分散型構造の例  
(ITU-T H.225.0)

#### G.5.3 クリアリングハウス型構造

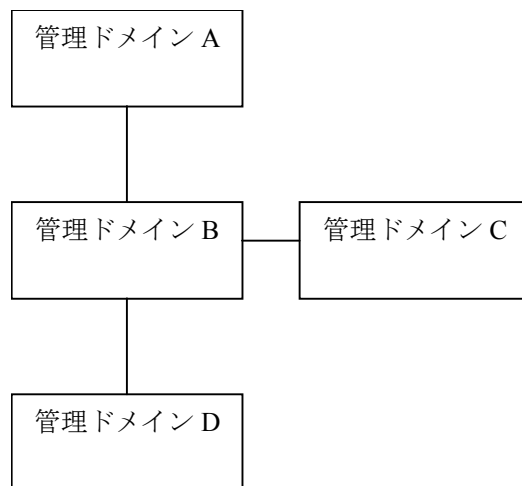
クリアリングハウス配置の例が、付図 G.5/JT-H225.0 に示される。この配置では、各管理ドメインは、アドレス解決のため、クリアリングハウスに問合せを行う。クリアリングハウスは管理ドメイン外のエンティティであるため、それと通信を行う同位エレメントは、定義により、境界エレメントとなることに留意すること。



付図 G.5/JT-H225.0 クリアリングハウス型構造  
(ITU-T H.225.0)

#### G.5.4 結合ポイント

付図 G.6/JT-H225.0 は、結合ポイントの例を示す。この例では、管理ドメイン B は、B 自身と管理ドメイン C と D の両方のアドレス解決に使用できる結合ポイントである。この例のように、管理ドメイン B は、管理ドメイン A から管理ドメイン C へ（アドレス）解決要求を転送してもよいし、管理ドメイン A に対し管理ドメイン C と直接コンタクトするための確かな着信先情報を知らせてもよい。もし、管理ドメイン B が、管理ドメイン A から管理ドメイン C への（アドレス解決）要求を転送するのであれば、管理ドメイン B は、管理ドメイン C からの応答をキャッシュしてもよい。



付図 G.6/JT-H225.0 結合ポイントの例  
(ITU-T H.225.0)

#### G.5.5 管理ドメインのオーバーラップ

二つ以上の管理ドメインが、与えられたアドレスの解決をしてもよい。例えば、複数の管理ドメインが、一般回線交換網（GSTN）用端末への呼を完結するゲートウェイを含むことができる。適切な着信先管理ドメインの選択は、発信元管理ドメインの責任である。着信先管理ドメイン選択に使うアルゴリズムは、製造業者に任されている。

### G.6 オペレーション

#### G.6.1 JT-H501 メッセージの利用

JT-H225.0 付属資料 G の実装には、JT-H501 で規定されるメッセージを用いなければならない。標準 JT-H501 では、JT-H501 メッセージをやり取りするエンティティのことを同位エンティティと呼んでいる。以下は本付属資料で用いられる JT-H501 のメッセージのリストである。

ServiceRequest

ServiceConfirmation

ServiceRejection  
ServiceRelease  
DescriptorRequest  
DescriptorConfirmation  
DescriptorRejection  
DescriptorIDRequest  
DescriptorIDConfirmation  
DescriptorIDRejection  
DescriptorUpdate  
DescriptorUpdateAck  
AccessRequest  
AccessConfirmation  
AccessRejection  
RequestInProgress  
NonStandardRequest  
NonStandardConfirmation  
NonStandardRejection  
UnknownMessageResponse  
UsageRequest  
UsageConfirmation  
UsageRejection  
UsageIndication  
UsageIndicationConfirmation  
UsageIndicationRejection  
ValidationRequest  
ValidationConfirmation  
ValidationRejection

上記のリストにない JT-H501 メッセージを受信した JT-H225.0 付属資料 G 同位エレメントは、UnknownMessageResponse メッセージで応答しなければならない。

やりとりされるメッセージは、JT-H501 で必須とされたフィールドは全て含まなくてはならない。オプションフィールドは必要に応じて追加してよい。

#### G.6.2 アドレステンプレートと記述子

同位エレメントは、以下の方法でテンプレートを得る：

- ・ スタティックな構成
- ・ 一般的な要求に応じて他の境界エレメントから記述子を受信する

- ・ 特定の照会に対する応答を受信する

#### G.6.2.1 スタティックな構成

同位エレメントは、責任を持つすべてのゾーンのためにテンプレートを保守するであろう。テンプレートは、同位エレメント自身が提供してもよい。あるいは、同位エレメントがゲートキーパーと同じ位置に存在する場合には、テンプレートはドメインの中で同位エレメントが通信しているゲートキーパーから得られた情報を要約することによって構成してもよい。境界エレメントは、要求に対する応答によって、他の境界エレメントが情報を入手できるようにしてもよい。

例：

- ・ 内部の構造を隠すことを望む境界エレメントは、そのゾーン全体を記述して、すべての着呼を処理するゲートキーパーに関係する記述子（AccessRequest メッセージを送るための表示）を提供する。
- ・ 内部の構造が明らかになってしまうことに注意を払っていない境界エレメントは、ドメイン内のゾーンにそれぞれゲートキーパーを記述して、テンプレートのセットを提供する。
- ・ ファイアウォール（あるいはゲートキーパーによって経路を決められたモデルを使っているもの）にある境界エレメントは、ゾーン全体に対して呼設定メッセージを送信するための指標をテンプレートに提供する。
- ・ ドメインに欠番がある境界エレメント（番号がもう1つの管理ドメインに変更されたということ）は、他の管理ドメインと連絡するために使われる境界エレメントを示す「送信 AccessRequest」を記したテンプレートを提供する。
- ・ クリアリングハウス境界エレメント（44の完全コピーを持っているような）は、44の中でそれぞれの管理ドメインのために「送信 AccessRequest」を記したテンプレートを維持する。

同位エレメントは、データベース全体のコピーを保管する必要はない。もし同位エレメントがデータベース全体のコピーを維持しないのであれば、他の問い合わせを解決するために使われるクリアリングハウス境界エレメントを示す静的に構成を設定された「送信 AccessRequest」テンプレートを含むべきである。

#### G.6.2.2 記述子の受信

同位エレメントは、もう1つの同位エレメントに対して静的に構成を設定されたテンプレートを要求してもよい。要求に対する応答は、テンプレートが要求された同位エレメントによって定められる。転送を要求するために、同位エレメントは受信を望む記述子を指定して DescriptorRequest メッセージを送信する。もし所有する同位エレメントが記述子を転送することが可能であれば、DescriptorConfirmation メッセージはすべてのテンプレートを指定して応答する。

要求する同位エレメントは、テンプレートが寿命になるまで、この方法で受信したテンプレートのコピーをキャッシュしてもよい、そして寿命がきた時点で同位エレメントはテンプレートのコピーを削除すべきである。もし所有している同位エレメントが寿命となる前に、静的に構成を設定されたテンプレートを変えるなら、関係する同位エレメントに DescriptorUpdate メッセージを送信しなければならない。DescriptorUpdate メッセージを受信した同位エレメントは、キャッシュ内のテンプレートのすべてを削除、追加、あるいは変更すべきである、あるいは所有者から示された記述子のコピーを要求すべきである。

中間同位エレメント（発側と着側の同位エレメントの間、あるいは管理ドメイン間の同位エレメント、たとえばクリアリングハウスあるいは結合ポイントなど）は、受信する記述子に基づいて記述子を発行してもよい。例えば、AccessRequest メッセージの宛先に関して、たとえクリアリングハウスが他の境界エレメントから受信した記述子が別の境界エレメントを宛先として示していたとしても、クリアリングハウスは自身をメッセージの宛先として指定しても良い。

同位エレメントは、発信者が管理ドメインの中に発信する許可を受けるために、テンプレートで要求条件を示してもよい。callSpecific フラグをテンプレートにセットし、メッセージタイプを AccessRequest メッセージで送信しなければならない時、発信者は AccessRequest メッセージで呼毎情報を提供しなければならない。もし同位エレメントが呼毎情報無しで AccessRequest メッセージを受信し、処理で呼毎情報を必要とするのであれば、同位エレメントは needCallInformation の理由で AccessRejection メッセージを返信しなければなら

ない。

同位エレメントが、他の既知の同位エレメントに、DescriptorUpdate メッセージを送信してもよい、あるいは同位エレメントは DescriptorUpdate メッセージをマルチキャストしてもよい。もし DescriptorUpdate メッセージがマルチキャストであるなら、同位エレメントはマルチキャストの範囲を考慮すべきである。DescriptorUpdate メッセージは変更のあった記述子を含むことができる。代わりに、DescriptorUpdate メッセージは、新しい情報について問い合わせできるように受信者を割り当てて、変更のあった記述子の識別だけを示してもよい。もし多数の記述子に変更があるのであれば、情報は、特定の DescriptorUpdate メッセージが最大の送信パケットサイズを越えないように、複数の DescriptorUpdate メッセージで送信すべきである。

### G.6.2.3 特定の問い合わせに対する応答の受信

同位エレメントは、完全に制限された、あるいは部分的に制限されたアドレスの解決を求めて、もう一つの同位エレメントに AccessRequest メッセージを送っても良い。AccessRequest メッセージは通常、低信頼性トランスポート（例：UDP）で送られるが、高信頼性トランスポート（例：TCP）で送っても良い。

AccessRequest メッセージを受け取った同位エレメントは自身のデータベースを検索し、相手側に対して最も有効なテンプレートで応える。もし複数のテンプレートが要求を満足する場合、条件に合う全てのテンプレートを同位エレメントは返さなければならない。相手先同位エレメントが特定の AliasAddress に対して実際の責任を持つとき、通常その同位エレメントは AccessRequest メッセージ、または呼設定メッセージのどちらか一方を送信すべきことを示すテンプレートで応えるであろう。相手先境界エレメントがクリアリングハウスの場合、通常 AccessRequest メッセージを送信すべきことを示すテンプレートで応えるであろう。

相手先同位エレメントは応答するテンプレートに将来的に有効であろうと思われるものを追加しても良い。これらのテンプレートの追加は、応答がトランスポート層でフラグメントする必要がでる程、大きくすべきではない（例：IPv4 の場合 576 オクテット、IPv6 の場合 1200 オクテット）。

例えば、ファイアウォールでしっかりと連結された同位エレメントは AccessRequest メッセージに対して 2 つのテンプレートを提供するかもしれない：一つは呼設定メッセージが送られるべき場所を指定する(数分、あるいは数秒の)短い生存時間のテンプレート、そして AccessRequest メッセージが管理ドメイン内の別の AliasAddress を持つ同位エレメントに送られるべきことを示す追加のテンプレート。

境界エレメントは生存時間が終わるまで AccessConfirmation メッセージで受信したテンプレートをキャッシュしてもよい。

## G.6.3 1つの同位エレメント、または同位エレメント群の発見

### G.6.3.1 静的

同位エレメントはアドレス解決のための問い合わせ先として管理された他の同位エレメント群を有してもよい。境界エレメントに対しては、この管理された境界エレメント群は管理ドメイン間の同意のもとに定義されるかもしれない。管理ドメインはクリアリングハウスのサービスをオプションとして利用しても良い。

### G.6.3.2 動的

IP ネットワーク上では Email-ID 形式のアドレスは DNS システムで定義される。このように、いくらか情報が足りないときに、同位エレメントは、@ 記号(例えば、person@example.org に対して \_h2250-annex-g\_udp.example.org の DNS SRV 探索)の右側に Email-ID による DNS SRV レコード探索をするかもしれない。この探索への応答は、解決処理中に使うことができる sendAccessRequest テンプレートを作成するために、使われるべきである。DNS の要求から作成されたテンプレートは DNS の応答で供給される生存時間よりも長くキャッシュされるべきではない。

### G.6.3.3 他の方法

他の同位エレメントを見つけるための他の方法の使用は将来的な課題である。

## G.6.4 解決手順

### G.6.4.1 管理ドメイン領域内の解決手順

同位エレメントが AliasAddress (例：共同配置されたゲートウェイ、ゲートキーパーによる)の解決を求められたときにキャッシュの中から条件にあったテンプレートを見つける。

もし条件にあったテンプレートが二つ以上の場合、ローカルポリシーに従いソートし適正なテンプレートを選ぶ。例えば、テンプレートの最初のソートはワイルドカード長(特殊なテンプレート程良い、特定のプロトコルタイプ(sendAccessRequest より sendSetup の方が良い)によって行っても良い。

もし複数のテンプレートがリクエストに合致するならば、同位エレメントは、条件にあった全てのテンプレートを返さなければならない。

もしテンプレートの選択処理で `sendSetup` としてマークされたテンプレートがなければ同位エレメントは、特定の宛先アドレスの `AccessRequest` メッセージをテンプレートで指定したアドレスに送ります。同位エレメントから答えを受け取るときに、キャッシュに保存して要求者に呼設定メッセージを送るべきアドレスを返すかもしれない。

#### G.6.4.2 管理ドメイン間の解決処理

境界エレメントが別の管理ドメインの境界エレメントから `AccessRequest` メッセージを受け取るとき、それはキャッシュ内のテンプレートを通じて検索され、クエリーの中に入っているアドレスに一致するテンプレートを見つける。

もし二つ以上一致するテンプレートがあれば、一致テンプレートは最初にワイルドカード長によりソートされる（より特殊なテンプレートがより適している）。次にそれらは特定のメッセージタイプによりソートされる（`sendAccessRequest` より `sendSetup` の方が良い）。それぞれの場合で、最も詳細に動作を特定できるもの以外、全てのテンプレートは破棄される。

もし一致するテンプレートが `sendAccessRequest` でマークされていれば、境界エレメントはテンプレート内で指定された同位エレメントへの `AccessRequest` メッセージを転送してもよい。またはそれらのテンプレートをそのまま返してもよい。

もし受け取られた `AccessRequest` メッセージのホップ・カウンターが 0 になったら、境界エレメントは他の同位エレメントへ `AccessRequest` メッセージを送ることが出来ない。しかし代わりに一致するテンプレートを返さなければならない。もしホップ・カウンターが 0 になり、境界エレメントが `AccessConfirmation` メッセージで提供する情報がない場合、境界エレメントはホップカウントが超過したことを示す `AccessRejection` メッセージで応答しなければならない。

その点で、境界エレメントはアクセス要求の認証をするために別の境界エレメント（例：クリアリングハウス）を利用しても良い。`AccessRequest` メッセージ内で要求している境界エレメントにより供給されるアクセストークンを運ぶ `ValidationRequest` メッセージを送る。

受理された境界エレメントはトークンを検証し、`ValidationConfirmation` メッセージを返す。境界エレメントは有効だと考えた他のテンプレートや検索で見つけた（同じアドレスでメッセージタイプフィールドを持つ）テンプレートを含む `AccessConfirmation` メッセージを返す。

もし複数のテンプレートが要求を満足するとき境界エレメントは一致したテンプレートを全て返さなければならない。もしアクセス要求が特定の発呼情報を含むとき、返されるテンプレートは特定の発呼要求に対してのみ正当なものである。これは管理ドメインが基本発呼単位でアクセスを許可したいときに使用される。管理ドメインは送信したそれぞれの `AccessRequest` メッセージ当たりの発呼情報を含むように要求しても良い。それは問い合わせるテンプレート内でフラグをセットすることで行われる。

#### G.6.5 利用情報の交換

同位エレメントは他の同位エレメントに対して特定の呼におけるリソースの用法についての情報を供給するよう要求しても良い。`UsageIndication` メッセージは呼の任意の段階で供給されてもよい。また、複数の `UsageIndication` メッセージはおそらく、より多くの最新の情報、連続した呼セグメント、または異なるメディアタイプの用法のレポートを同じ呼に対して送ってもよい。詳細は G.6.5.1 節を参照のこと。`UsageIndication` メッセージは 2 つの同位エレメントがサービスしあっている関係にあるかどうか関係無く交換されてもよい。しかし同位エレメントのポリシーは、サービスしあっている関係がないときの交換を許可しなくともよい。その場合、同位エレメントは `noServiceRelationship` エラーコードを送って `UsageIndication` メッセージを拒否してもよい。

`UsageRequest` メッセージ、`AccessRequest` メッセージ、`ValidationRequest` メッセージ、および `ValidationConfirmation` メッセージのうちいずれかで `UsageIndication` メッセージを要求する呼のコンテキストが送られることによって、または同位エレメントが相互にやり取りを行うためのテンプレートのうちどちらかを要求したときに `UsageIndication` メッセージの要求は送られなければならない。

##### G.6.5.1 同じ呼に対する複数の `UsageIndication` メッセージ

同じ呼に対する複数の `UsageIndication` メッセージは同じメディアタイプ上でより多くの最新情報、または同じ呼で作られた新しいメディアタイプについての利用情報を提供する。また同位エレメントが処理中に呼を引き継いでも良いので、必ずしも同じ同位エレメントからすべての `UsageIndication` メッセージが送信されるわけではない。以下のルールは意味を定義する。

1. usageCallStatus フィールドが callInProgress になっている状態で受信した UsageIndication メッセージは、その後の UsageIndication メッセージが同じ callIdentifier で senderRole フィールドを受け取られなければならないことを暗示している。もし、受信側が障害回復構成をとっているならば、設定された期間中に UsageIndication が受信されない場合、障害が発生していると判断し、それまでに受信した UsageIndication メッセージからデータを回復するかもしれない。
2. その後の usageField が同じ id の UsageIndication メッセージは前回のメッセージ(これが代替りの同位エレメントにとっては不可能であるとしても)の endTime と一致した startTime をレポートしなければならない。受信側はそれぞれのレポートは期間内でだけ有効であると仮定しなければならない。同じ usageField id を持つ前回受信したメッセージを新しい usageField 中の情報で上書きする。
3. 同位エレメントは新しい UsageIndication メッセージを呼の間でメディアタイプのそれぞれの変化に対して送信しなければならない。例えばオーディオが停止した、ファックスが開始した、またはコーデックが変わったなど。もし複数のメディアタイプ (例えば、オーディオとビデオ) が同時に使用されていれば、それらは同じ UsageIndication メッセージ内でレポートされるべきである。

#### 6.6.5.2 サービス関係確立時における UsageInformation の要求と交渉

同位エレメント PE<sub>A</sub> は、次の同位エレメント PE<sub>B</sub> に対する ServiceRequest メッセージ中に UsageSpecification 要素を含んでも良い。UsageInformation 要素を送ることにより、その二つの同位エレメント間のサービス関係がある限りに置いて、すべての呼に対する既定の usage information を報告するようにすることができる。この UsageSpecification は PE<sub>B</sub> から PE<sub>A</sub> に UsageIndication を送る全ての呼に対して利用されなくてはならない。

UsageSpecification が PE<sub>A</sub> から PE<sub>B</sub> への別のメッセージ中で到着した場合(例えば AccessConfirmation に含まれて)、新しい UsageSpecification で既定の UsageSpecification を上書きし、新しいメッセージに関連する全ての呼に適用する。

UsageSpecification 要素を含む ServiceRequest を受信した同位エレメントは、伊かのように振舞うべきである。

- (i) 受信した同位エレメントがその ServiceRequest を受理しようとしており、かつ UsageSpecification がその中に含まれている場合、その同位エレメントは受信した ServiceRequest に含まれているのと同じ UsageSpecification を ServiceConfirmation に入れて送信しなければならない。UsageSpecification は送受両方向の呼に対して適用されなければならない。すなわち、要求した同位エレメントから受理した同位エレメントへ入ってくる呼と、受理した同位エレメントから要求した同位エレメントへ向けて出て行く呼の両方である。
- (ii) 受信した同位エレメントがその ServiceRequest を受理しようとしているが、その中に含まれている UsageSpecification は受理しない場合、次のいずれかの対応をしなければならない。一つは、要求している同位エレメントに対して提供可能な usage information を規定する異なった UsageSpecification を ServiceConfirmation に入れて送る。もう一つの対応は、理由を cannotSupportUsageSpec とした ServiceRejection メッセージを送ることである。
- (iii) 受信した同位エレメントが usage 報告をまったくサポートしていない場合、理由を usageUnavailable とした ServiceRejection メッセージを送らなくてはならない。

ServiceConfirmation を受信した同位エレメントは以下のように振舞わなくてはならない。

- (i) ServiceConfirmation 中の UsageSpecification が SendRequest で送ったものと同じである場合、発側の同位エレメントと着側の同意エレメントの間にはサービス関係が成立する。
- (ii) ServiceConfirmation 中の UsageSpecification が SendRequest で送ったものと異なる場合、発側の同位エレメントが新しい UsageSpecification でよい場合は、サービス関係が成立する。発側の同位エレメントが新しい UsageSpecification の利用を望まない場合は、理由を terminated にした ServiceRelease メッセージを送らなければならない。そして、発側の同位エレメントは受信した UsageSpecification を解析して、次回は両者にとって受理可能な UsageSpecification を作る参考にすればよい。
- (iii) ServiceRequest には UsageSpecification を入れたにもかかわらず受信した ServiceConfirmation に UsageSpecification が含まれていない場合には、その ServiceConfirmation を送ってきた同位エレメントはサービス関係のレベルでの usage 報告をサポートできないか、するつもりがないことを示している。これは、例えば相手の同位エレメントがこの付属資料の第 1 版の実装である場合に起こる。このとき、発側の同位エレメントはサービス関係を終結しても、終結しなくても良い。終結する場合は、理由コードを terminated にした ServiceRelease メッセージを送る。どちらの場合でも、発側の同位エレメントが呼の usage information を受信することに興味があるなら、本付属資料の第 1 版の仕組みを用いて要求すべきである。(UsageSpecification を AccessRequest、AccessConfirmation(返されたアドレステンプレートに含める)、UsageRequest、ValidationRequest、または ValidationConfirmation に入れて送ればよい。)

#### G.6.6 番号ポータビリティ情報シグナリング

H.460.2 は JT-H323 システムにおける番号ポータビリティの仕組みを規定している。H.460.2 をサポートす



るには、本付属資料についてもアドレス解決メッセージ交換において番号ポータビリティ情報の伝送の能力を持つ必要がある。付属資料 G の境界エレメントと他の JTJ-H323 ネットワーク要素とのインタフェースはこの付属資料ではカバーされていないが、そのインタフェースは H.460.2 番号ポータビリティ情報を付属資料 G の境界エレメントとの間でやり取りできる能力があることが想定されている。

AccessRequest が送られるとき、もしあれば、メッセージの共通情報部分の genericData フィールドを用いて H.460.2 番号ポータビリティ情報を伝送する。

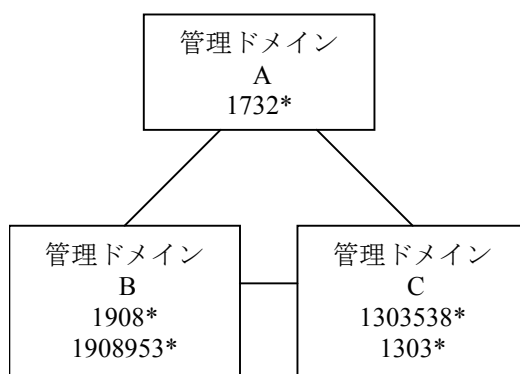
AccessConfirmation や AccessRejection メッセージも同様に genericData フィールドに番号ポータビリティ応答情報を含めて伝送する。AccessRejection の場合、拒絶理由は genericDataReason でなくてはならない。

## G.7 シグナリングの例

これらのシグナリングの例は、基本的な動作を説明するために用意されている。これらの例では、管理ドメインはお互いに合意していることを前提としており、したがって境界エレメントはお互いについての情報(例. TCP ポート)を与えられている。以下に示す、多くの例の中で、RAS LRQ/LCF メッセージが、同じ管理ドメインの中のゲートキーパーと境界エレメントの間で交換されることが示される。これは、単なる説明目的の物である。なお、同様の付属資料 G メッセージが境界エレメントと同位エレメントの間でも交換される。

### G.7.1 分散型または、フルメッシュ型構造

分散型ネットワークの例を付図 G.7/JT-H225.0 に示す。



付図 G.7/JT-H225.0 シグナリング説明のための分散型ネットワークの例 (ITU-T H.225.0)

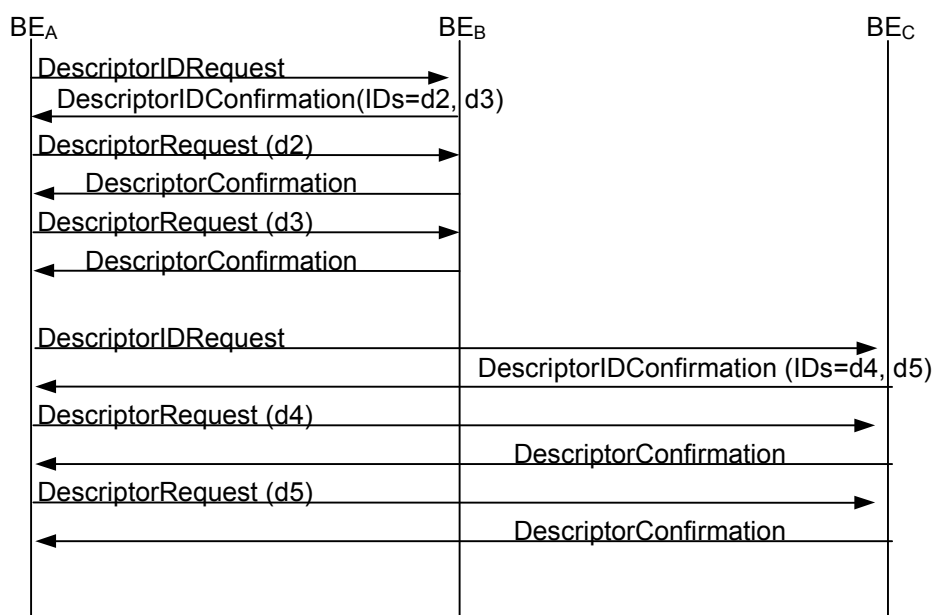
この例では、それぞれの管理ドメインは1つの境界エレメントを持ち、境界エレメントは、以下に示すようなアドレス解決をするように構成されている事を前提とする。

管理ドメイン	テンプレートの定義	コメント
A	Descriptor "d1": Pattern = 1732* Transport address = BE <sub>A</sub> call signal address Message type = <b>sendSetup</b>	管理ドメインAに向けた呼へのシグナリングは、管理ドメインAの境界エレメントを通る。
B	Descriptor "d2": Pattern = 1908* Transport address = BE <sub>B</sub> annex g address Message type = <b>sendAccessRequest</b>  Descriptor "d3": Pattern = 1908953* Transport address = GW <sub>B1</sub> CALL SIGNALLINGaddress Message type = <b>sendSetup</b>	1908* 向けの呼のために、AccessRequest メッセージが着信先(すなわち、ゲートウェイ)の呼シグナリングアドレスを得るために必要とされる。  1908953*向けの呼のために、このゲートウェイに直接 Setup が送られる事ができる。
C	Descriptor "d4": Pattern = 1303538* Transport address = GK <sub>C1</sub> call signal address	1303538*向けの呼は、このゲートキーパーを通してルーチングされる。

	Message type = <b>sendSetup</b>  Descriptor “d5”: Pattern = 1303* Transport address = BE <sub>C</sub> annex g address Message type = <b>sendAccessRequest</b>	1303*向けの呼は、目的のゲートウェイへ直接シグナリングされる事ができる。しかし、AccessRequest はゲートウェイの呼シグナリングアドレスを得るために送らねばならない。
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------

### G.7.1.1 ゾーン情報の交換

分散型または、フルメッシュ構造の場合には、おそらく多数の双方の契約上の同意を通じて、各管理ドメインはその他の管理ドメインを知っている。いかなる時でも管理ドメイン中の境界エレメントは、アドレス情報を得るために他の管理ドメインに問い合わせることができる。このシグナリングの例は付図 G.8/JT-H225.0 に現れる。

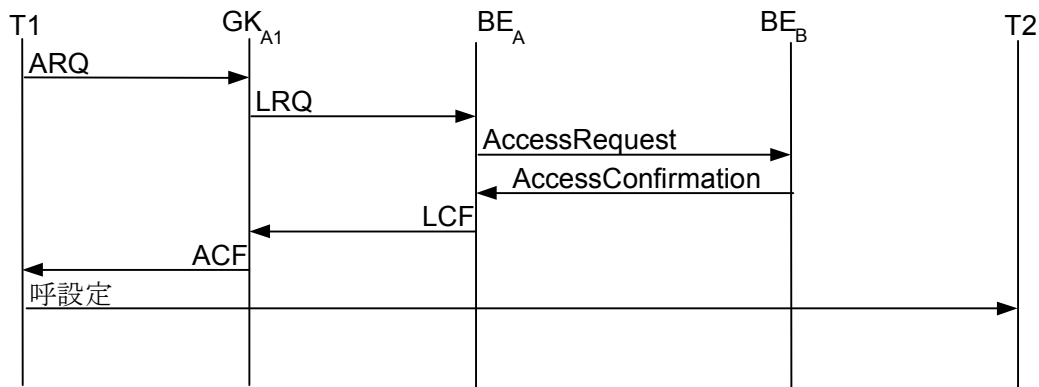


付図 G.8/JT-H225.0 記述子交換の例  
(ITU-T H.225.0)

同様に、BE<sub>B</sub> は、BE<sub>A</sub> と BE<sub>C</sub> に問い合わせる。そして、BE<sub>C</sub> は BE<sub>A</sub> と BE<sub>B</sub> に問い合わせる。

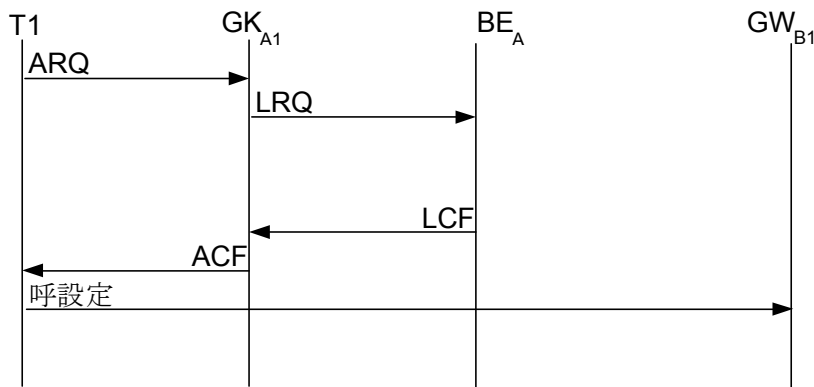
### G.7.1.2 呼の開始

管理ドメイン A 中の T1 が 19085551515(T2)に発呼すると仮定する。T1 からの ARQ を受信すると、T1 のゲートキーパーは LRQ を送信する。管理ドメイン A の境界エレメントである BE<sub>A</sub> は、既にゾーン記述子を受信しており、そのリクエストをどのように処理したらよいかを知っている。付図 G.9/JT-H225.0 に示すように、BE<sub>A</sub> は、BE<sub>B</sub> から受信した記述子の規定に従って、BE<sub>B</sub> に AccessRequest メッセージを送信する。BE<sub>B</sub> は、T2 の呼シグナリングアドレスを返す（この例では、T2 は、いかなる種類のエンドポイントでも良い）。すると、T1 は JT-H323 または、付属資料に示されている標準的な手順に従って JT-H225.0 の呼設定メッセージを T2 の呼シグナリングアドレスに送信する。



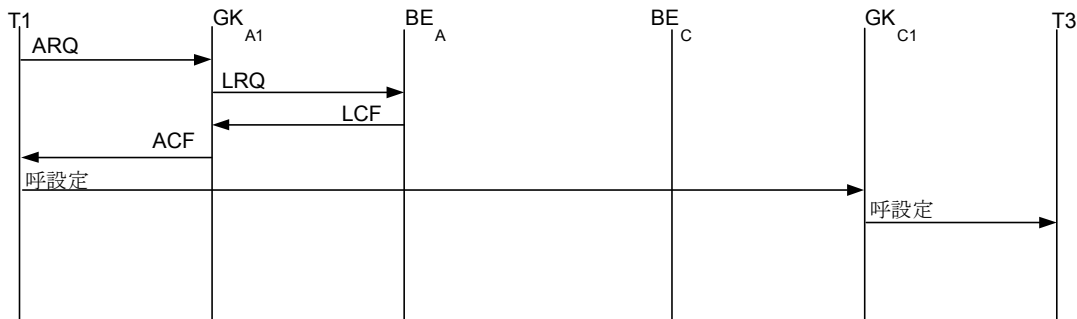
付図 G.9/JT-H225.0  
(ITU-T H.225.0)

ここで、T1 が 19089532000 へ発呼すると仮定する。この例では、BE<sub>A</sub> はすでに管理ドメインの中で、呼を受理するゲートウェイの呼シグナリングアドレスを取得している。付図 G.10/JT-H225.0 に示すように、BE<sub>A</sub> は管理ドメイン B との間でいかなるメッセージの交換も行わず、LRQ に応答する事が可能であり、T1 がゲートウェイに呼設定メッセージを直接送信することを許可する。



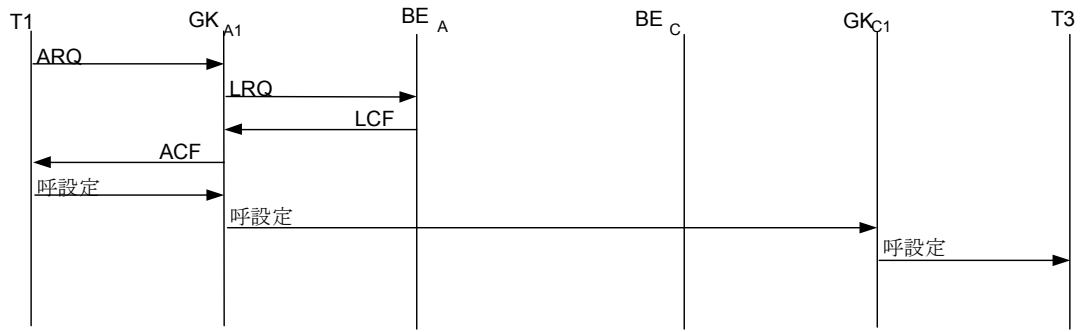
付図 G.10/JT-H225.0  
(ITU-T H.225.0)

他の例では、T1 は 13035382899 に発呼すると仮定している。管理ドメイン C はこの番号への呼を受理する能力を既に公示しているので、ゲートキーパー経由型モデルに従い、ゲートキーパーを通じて呼シグナリングを受理する。付図 G.11/JT-H225.0 に示すように、BE<sub>A</sub> は管理ドメイン C との間でいかなるメッセージの交換も行わず、LRQ に対して管理ドメイン C のゲートキーパーの呼シグナリングアドレスを含む LCF で応答することが可能である。



付図 G.11/JT-H225.0  
(ITU-T H.225.0)

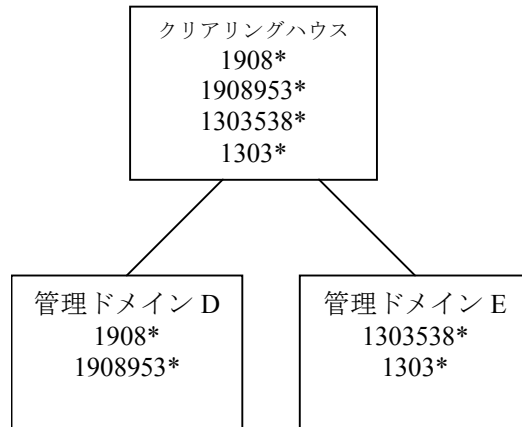
あるいは、付図 G.12/JT-H225.0 に示すように、T1 のゲートキーパーがゲートキーパー経由型モデルを実行することもできる。



付図 G.12/JT-H225.0  
(ITU-T H.225.0)

### G.9.2 クリアリングハウス

クリアリングハウスを利用した構成の例を付図 G.13/JT-H225.0 に示す。以下に続く例では、この図を参照されたい。この例では、クリアリングハウスは、クリアリングハウスがサービスを提供するすべての管理ドメインのアドレス情報を保持している。



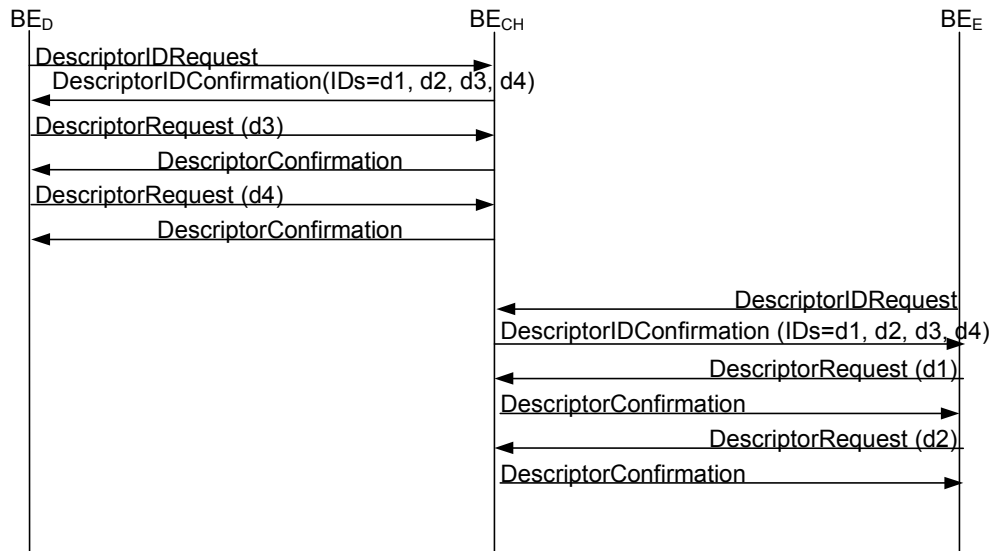
付図 G.13/JT-H225.0 クリアリングハウスの構成の例  
(ITU-T H.225.0)

この例では、管理ドメインDおよびEの境界エレメント、およびクリアリングハウスは、以下の情報を含んでいる。

管理ドメイン	テンプレートの 定義	コメント
D	<p>Descriptor “d1”:  Pattern = 1908*  Transport address = BE<sub>D</sub> annex g address  Message type = <b>sendAccessRequest</b></p> <p>Descriptor “d2”:  Pattern = 1908953*  Transport address = GW<sub>D1</sub> call signaling address  Message type = <b>sendSetup</b></p>	<p>1908*への呼に対しては、宛先（即ち：ゲートウェイ）の呼シグナリングアドレスを取得するために、AccessRequest メッセージが必要である。</p> <p>1908953*への呼に対しては、呼接定はこの特定のゲートウェイに直接送信することができる</p>
E	<p>Descriptor “d3”:  Pattern = 1303538*  Transport address = GK<sub>E1</sub> call signal address  Message type = <b>sendSetup</b></p> <p>Descriptor “d4”:  Pattern = 1303*  Transport address = BE<sub>E</sub> annex g address  Message type = <b>sendAccessRequest</b></p>	<p>1303538*への呼は、この特定のゲートキーパーによりルーチングされる。</p> <p>1303*への呼は、宛先のゲートウェイへ直接シグナルされる。しかし、ゲートウェイの呼シグナリングアドレスを取得するために AccessRequest が送信されなければならない。</p>
CH	<p>Descriptor “d1”:  Pattern = 1908*  Transport address = BE<sub>D</sub> annex g address  Message type = <b>sendAccessRequest</b></p> <p>Descriptor “d2”:  Pattern = 1908953*  Transport address = GW<sub>D1</sub> call signaling address  Message type = <b>sendSetup</b></p> <p>Descriptor “d3”:  Pattern = 1303538*  Transport address = GK<sub>E1</sub> call signal address  Message type = <b>sendSetup</b></p> <p>Descriptor “d4”:  Pattern = 1303*  Transport address = BE<sub>E</sub> annex g address  Message type = <b>sendAccessRequest</b></p>	<p>クリアリングハウスは、他の管理ドメインから記述子を取得し、記述子交換の間、分配のためにこの情報を保持する。</p>

#### G.7.2.1 ゾーン情報の交換

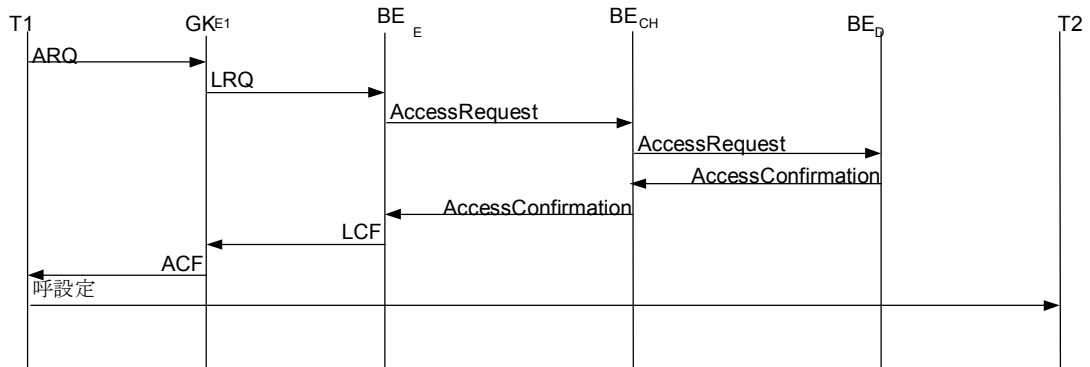
この例では、クリアリングハウスはクリアリングハウスサービスに登録している管理ドメインとの間で情報を交換する。クリアリングハウスはそれぞれの管理ドメインから受信した情報を保持し、他の管理ドメインに情報を渡す。この例では、クリアリングハウスは管理ドメイン E と管理ドメイン D から見える。管理ドメイン D と E 間でお互いに認識する必要はない。



付図 G.14/JT-H225.0 クリアリングハウスを使用して記述子交換する例 (ITU-T H.225.0)

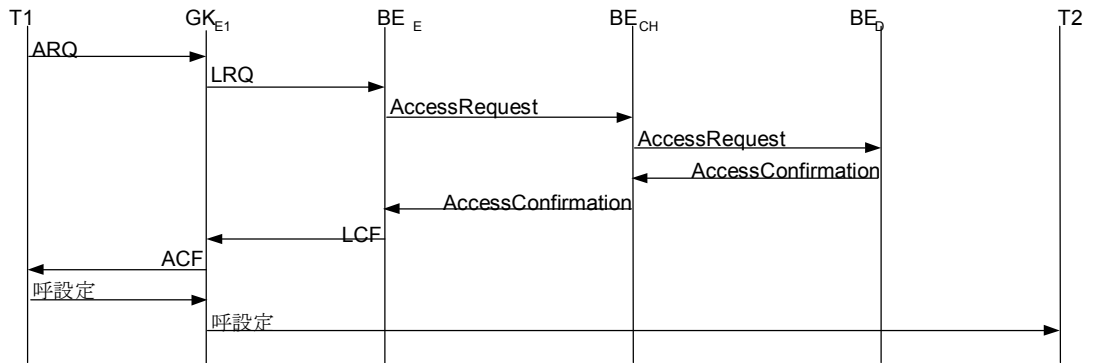
### G.7.2.2 呼の開始

管理ドメイン E 内の T1 が 19085551515 に発呼することを想定する。管理ドメイン E 内の境界エレメントは、このような呼の場合にクリアリングハウスに問い合わせるべきであることを示したクリアリングハウスからの記述子をあらかじめ受信している。境界エレメントはクリアリングハウス境界エレメントに AccessRequest を送信する。管理ドメイン D 内の境界エレメントから受信したクリアリングハウス境界エレメントの記述子に基づいて、クリアリングハウス境界エレメントは管理ドメイン D 内の境界エレメントに AccessRequest を送信する。クリアリングハウス境界エレメントから管理ドメイン E 内のクリアリングハウスに確認を返す時、確認に管理ドメイン D 内の境界エレメントから送信された情報を含む。T1 のゲートキーパーは T1 が T2 に呼設定メッセージを送信することを許可した T2 の destCallSignalAddress で ACF を返す。



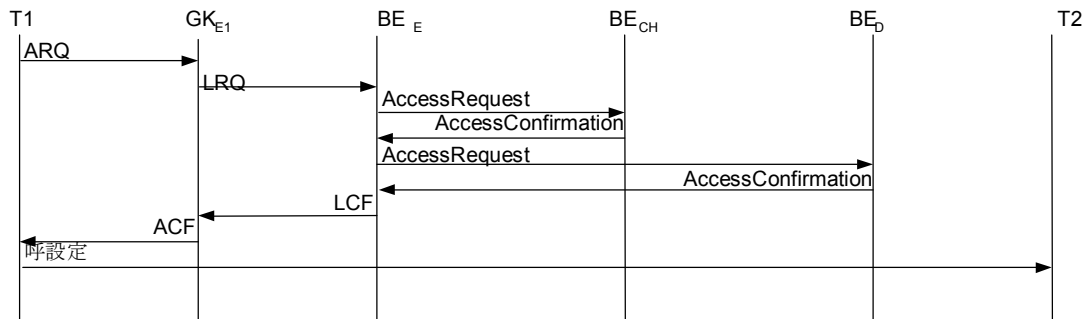
付図 G.15/JT-H225.0 (ITU-T H.225.0)

あるいは、T1 のゲートキーパーは付図 G.16/JT-H225.0 に示すように、呼シグナリングをルーティングすることができる。



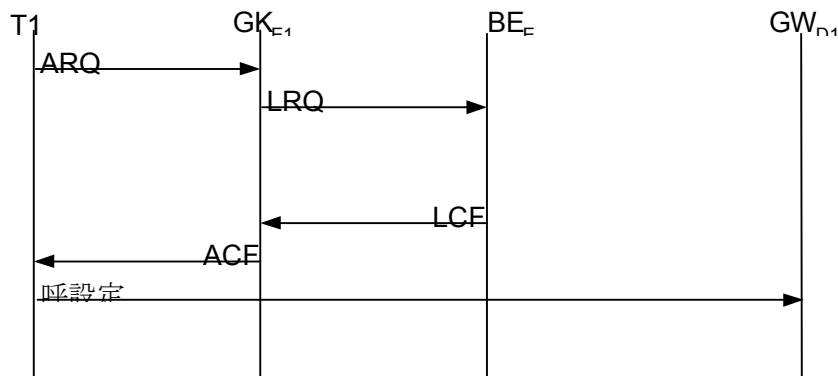
付図 G.16/JT-H225.0  
(ITU-T H.225.0)

もう1つの可能性は付図 G.17/JT-H225.0 に示されるように、クリアリングハウスが管理ドメイン D 内の境界エレメントへの接続情報を含めて管理ドメイン E 内の境界エレメントに応答する。



付図 G.17/JT-H225.0  
(ITU-T H.225.0)

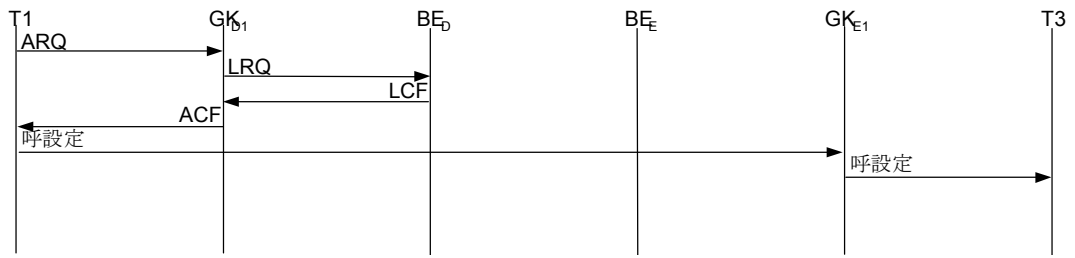
今 T1 が 19089532000 に発呼することを想定する。付図 G.18/JT-H225.0 に示すように、前もって交換された記述子は境界エレメントがクリアリングハウスに問い合わせないで T1 に呼シグナリングアドレスを返すことを許可する。



付図 G.18/JT-H225.0  
(ITU-T H.225.0)

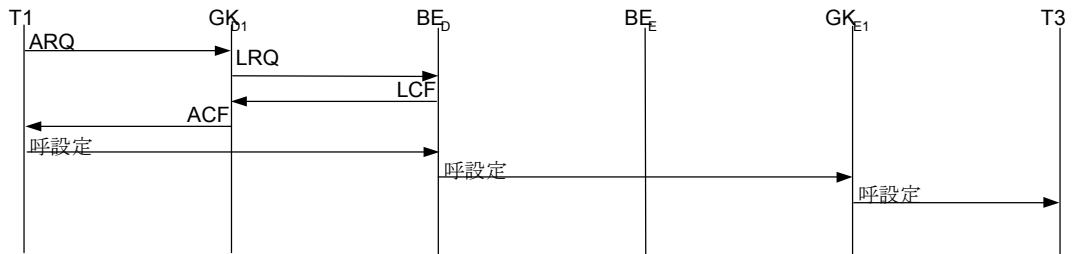
次に T1 が 13035382899 に呼を開始するシナリオを考える。付図 G.19/JT-H225.0 に示されるように、管理ドメイン E 内の境界エレメントは以前に 1303538\* の呼が Access Request メッセージの必要は無く、管理ドメイン E 内のゲートキーパーに直接ルーティングすることができる事を通告されていた。(この通告はエンティティがゲートキーパーであることを示さない。ただ呼設定メッセージが指定したアドレスに送信されることを示す。) 管理ドメイン D 内の境界エレメントはクリアリングハウスからこの情報を受け取っている。この例でクリアリングハウスを想定すると、これらの呼に対してアドレス解決を提供することは要求されていな

い。



付図 G.19/JT-H225.0  
(ITU-T H.225.0)

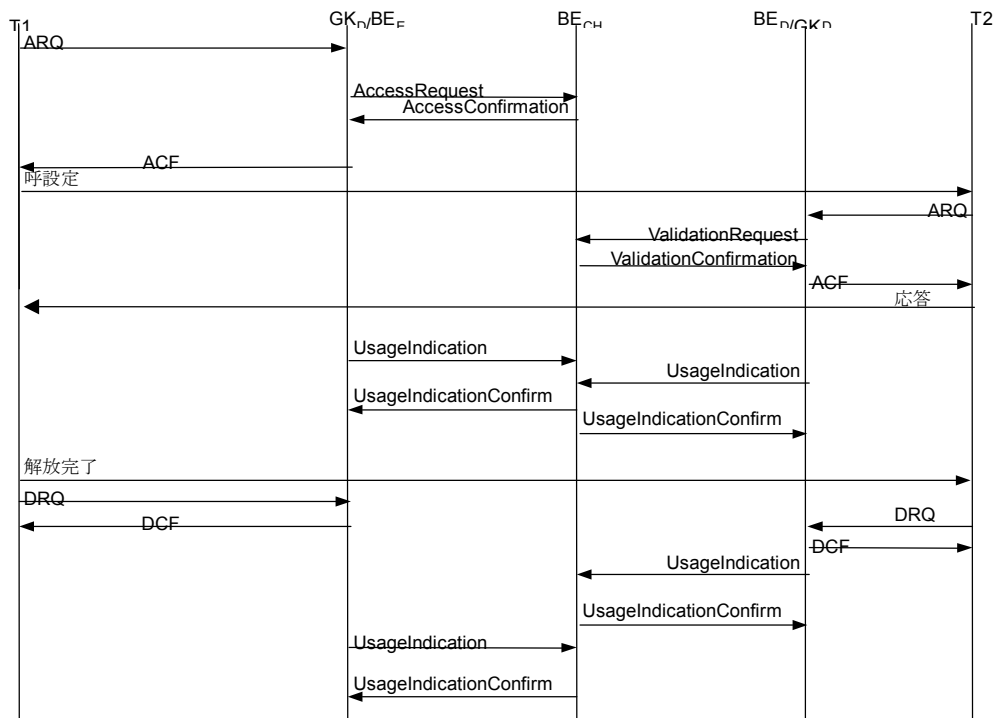
境界エレメントがゲートキーパーと結合し、そして同様にゲートキーパー経由型モデル内で呼がルーチングされることに注意すること。もう1つのシグナリング例を付図 G.20/JT-H225.0 に示す。もし記述子がそのように設定されているのなら、同様に管理ドメインの中にルーチング指定するゲートキーパーとして境界エレメントを使用することが可能である。



付図 G.20/JT-H225.0  
(ITU-T H.225.0)

付図 G.21/JT-H225.0 の例では、クリアリングハウスは管理ドメインに終端する呼に対し認証を行う。クリアリングハウスは同様に呼に対して生成する側と終端する側の両方で境界エレメントが usage indications (利用状況) を送信することを要求する。





付図 G.21/JT-H225.0  
(ITU-T H.225.0)

## G.8 付属資料Gプロファイル

### G.8.1 イントロダクション

TTC 標準 JT-H501 は本付属資料が管理ドメイン間ならびに単一管理ドメイン内の同位エレメント間の相互通信に用いるためのメッセージおよびフィールドを規定する。メッセージとフィールドの多くはオプションであり、異なるサービスおよびサービスオプションを実現するために様々な方法で利用することができる。本節ではプロファイルを規定し、そのプロファイル準拠を謳うために必要とされるメッセージ、フィールド、手順を定義する。

#### G.8.1.1 プロファイルのシグナリングと交渉

同位エレメントは他の同位エレメントに対して、JT-H323 の汎用拡張フレームワークを用いて、トランザクションを成功させるために必要なプロファイルのセット、利用することを希望するプロファイルのセット、およびサポートしているプロファイルのセットをシグナリングする。このプロファイルの交渉シグナリングは、個別のメッセージ交換(例えば AccessRequest/AccessConfirmation 交換による)あるいはサービス関係を確立する際に行われる。二つの同位エレメント間でのサービス関係の確立はプロファイルによって要求されなにかもしれないことに留意すること。

##### G.8.1.1.1 要求側エンティティによる処理

要求側のエンティティ(同位エレメント)は、それが要求するプロファイルを示すために **FeatureSet** 構造体中の要素を用いる。同位エレメントは利用することが必要なプロファイルは **neededFeatures** フィールドで、利用することを望んでいるプロファイルは **desiredFeatures** フィールドで、サポートしているプロファイルは **supportedFeatures** フィールドで示す。これらすべてのフィールドは **FeatureSet** 構造体に含まれている。

要求への応答としては、要求したエンティティは確認か拒絶かのいずれかを受信すべきである。

要求が拒絶された場合は、応答したエンティティはその要求が成功するために要求エンティティがサポー

トしなければならない **neededFeatures** セットを含めるかもしれない。このとき、要求したエンティティが必要な **feature** (例えば、ある特定のプロファイル)をサポートするならば、要求したエンティティは応答エンティティによって要求されたプロファイルのサポートを含んだ要求を再度発行してもよい。

要求が受理された場合は、広報互換を担保する方法で交渉が行われることを確実にするために特別な手順が適用される必要がある。これは、要求したエンティティが必要とされたプロファイルが応答中の **supportedFeatures** としてリストされているかをチェックすることにより行われる。要求しているエンティティが応答メッセージの **supportedFeatures** フィールド中に必要なプロファイルを発見できない場合は、応答したエンティティが必要なプロファイルをサポートしていないと仮定しなければならない。要求したエンティティがこの環境下では進められないと判断した場合には、応答したエンティティの状態が戻るように、実行しようとしている動作を差し戻さなくてはならない (**ServiceRequest** メッセージを送ってある場合には **ServiceRelease** メッセージを送る)。

#### G.8.1.1.2 応答したエンティティによる処理

応答したエンティティは、要求を受理できるか否かを判断するために要求中の **neededFeatures** フィールドに規定されているプロファイルを見る。また、要求しているエンティティが必要なプロファイルをサポートしているか否かを判断するために、**neededFeatures**, **desiredFeatures**, **supportedFeatures** フィールドを見る。

応答するエンティティが両端のエンティティによって必要なプロファイルがサポートされていると判断された場合には、応答エンティティは要求に肯定的に答えてよい。応答エンティティは応答中の **supportedFeatures** にサポートすることを選択したプロファイルをリストする。要求を受理する場合は、応答中の **supportedFeatures** フィールド中に要求に含まれている **neededFeatures** のすべてを含めなければならない。応答するエンティティは **desiredFeatures** を含めてもよい。

応答するエンティティが要求したエンティティに対して追加的にプロファイルのサポートを求める場合、要求を拒絶しなければならない。応答エンティティが要求を成功させるためにサポートしなければならないプロファイルを宣言する意思がある場合、拒絶メッセージ中の **neededFeatures** を用いて特定されるべきである。さらに、応答するエンティティは拒絶メッセージ中に **desiredFeatures** や **supportedFeatures** を含めてもよい。

#### G.8.1.1.3 識別子

以下の識別子は **FeatureDescriptor** の中で用いられ、付属資料 G のプロファイルに適用される **FeatureDescriptor** を指示する。

値	記述
idAnnexGProfiles	この識別子は、 <b>FeatureDescriptor</b> が必要な/希望の/サポートされる付属資料 G プロファイルを示すために、 <b>FeatureDescriptor</b> の”id”フィールド内で用いられる。

以下の表は、本付属資料と整合する汎用拡張フレームワークで用いられる識別子のリストを含んでいる。

標準整数値	記述
0	<b>FeatureDescriptor</b> が付属資料 G プロファイルを記述していることを示す、 <b>FeatureDescriptor</b> 中の識別子
1	付属資料 G のプロファイル”A”を示す、 <b>EnumeratedParameter</b> 中の識別子

#### G.8.2 プロファイル”A”: 信頼度の高いゲートキーパーへのゾーン間呼ルーティング

このプロファイルは、信頼度の高いゾーンの付属資料 G シグナリングアドレスが静的に与えられている

環境での呼ごとの問い合わせの単純なドメイン間サービスを規定する。これは、本付属資料のもっとも単純な利用のひとつであり、別のゾーンにエンドポイントを問い合わせる RAS LRQ の使用と同様である。同じプロファイルを用いて、ドメイン全域の知識あるいは更なる付属資料 G の問い合わせによりルーティング情報を返すことのできる、信頼度の高い同位エレメントを問い合わせることもできる。

### G.8.2.1 要求されたメッセージ

このプロファイルに準拠したエンティティは以下の表において必須と示されているメッセージをサポートしなければならない。

メッセージ	送信 (M:必須, O:オプション, R:推奨)	受信とそれに基づく動作 (M:必須, O:オプション, R:推奨)
ServiceRequest	O	M (Note 1)
ServiceConfirmation	O	O
ServiceRejection	M	O
ServiceRelease	O	O
DescriptorRequest	O	M (Note 1)
DescriptorConfirmation	R (Note 2)	O
DescriptorRejection	M	O
DescriptorIdRequest	O	M (Note 1)
DescriptorIdConfirmation	R (Note 2)	O
DescriptorIdRejection	M	O
DescriptorUpdate	O	M (Note 3)
DescriptorUpdateAck	M	O
AccessRequest	M	M
AccessConfirmation	M	M
AccessRejection	M	M
RequestInProgress	M	M
NonStandardRequest	O	M
NonStandardConfirmation	O	O
NonStandardRejection	M	O
UnknownMessageResponse	M	M
UsageRequest	O	M (Note 1)
UsageConfirmation	O	O
UsageRejection	M	O
UsageIndication	O	M (Note 1)
UsageIndicationConfirmation	O	O
UsageIndicationRejection	M	O
ValidationRequest	O	M (Note 1)
ValidationConfirmation	O	O
ValidationRejection	M	O

Note 1: 受信し、少なくとも拒絶しなくてはならない

Note 2: エンティティは自分に向けられた SendAccessRequest に付随したテンプレートに対して、少なくとも単一の記述子を返すことを推奨する

Note 3: 受信し ACK を返さなくてはならないが、処理する必要はない

### G.8.2.2 要求フィールド

標準 JT-H501 において必須と定義されたすべてのフィールドはこのプロファイルでも必須である。

このプロファイルに準拠したエンティティは以下の表に規定するフィールドもまたサポートしなければならない。

標準 JT-H501 においてオプションと定義されたその他のフィールドはオプションとして現れてもよい。

メッセージまたは構造体	要求フィールド	備考
AccessRequest message	<b>destinationInfo</b>	宛先の完全に修飾された E.164 アドレスを含む一つのアドレス
	<b>sourceInfo</b>	domainInfo と endpointType を含む
	<b>callInfo</b>	
AccessConfirmation message	<b>templates</b>	テンプレートが存在するとき、終端ゲートウェイ/ゲートキーパーごとに一つのテンプレートがある
	<b>partialResponse</b>	FALSE にセットする
AddressTemplate structure	<b>pattern</b>	E.164 アドレスを含む一つの特定期間パターンが存在する
	<b>routeInfo</b>	一つのインスタンスが存在する
	<b>timeToLive</b>	
RouteInformation structure	<b>messageType</b>	存在する
	<b>callSpecific</b>	FALSE にセットする
	<b>contacts</b>	一つのインスタンスが存在する
	<b>type</b>	messageType = sendSetup のとき、存在しなくてはならない
ContactInformation structure	<b>transportAddress</b>	ゲートウェイ/ゲートキーパーの IP アドレス
	<b>priority</b>	

### G.8.2.3 要求手順

このプロファイルでは、エンティティは本付属資料 (G.6.3.1 節) にの静的発見手順を利用してもよい。そして、エンティティは要求を送信可能な同位エレメントあるいはゲートキーパーの構成リストを得るであろう。そのリストは、第一のエレメントに到達できないときのための代替エレメントを含むかもしれない。

エンティティは、本付属資料 (G.6.3.2 節) の動的発見手順を用いることもできる。

エンティティは、呼ごとに選択された同位エレメントあるいはゲートキーパーに AccessRequest メッセージを送らなくてはならない。二つ以上の同位エレメントあるいはゲートキーパーが与えられた呼に対して利用可能であるとき、それらを縦列的に応答するか、並列的に応答するかは定められていない。この選択は、

要求エンティティに委ねられている。

応答には、ゼロあるいは一つ以上のテンプレートが含まれる。**timeToLive** は他の呼では利用できないことを示すために、60 秒あるいはそれ以下に設定されるかもしれない。

より一般的な同位との相互接続性を改善するために、同位エレメントが記述子のサポートを実装していない場合には、以下の手順に従うべきである。

- DescriptorIDRequest メッセージを受信したときは、同位エレメントは一つの DescriptorInfo を含む DescriptorIDConfirmation を返すべきである。この DescriptorInfo は同位エレメントそれ自身を指す sendAccessRequest を特定するような一つのテンプレートを含む記述子を記述する。
- DescriptorRequest メッセージを受信したときは、同位エレメントは一つの記述子を含む DescriptorConfirmation メッセージを返すべきである。この記述子は同位エレメントそれ自身を指す sendAccessRequest を特定する一つのテンプレートを含まなくてはならない。

#### G.8.2.4 プロファイルAのための識別子

以下の識別子は、EnumeratedParameter が付属資料 G プロファイル A を指定することを示すために、EnumeratedParameter の中で用いられる。

値	記述
idAnnexGProfileA	この識別子は付属資料 G プロファイル A が必要である／望まれる／サポートされるを示すために用いられる  EnumeratedParameter 中の content フィールドは存在しないことに留意のこと

## 付属资料H： JT-H225.0 メッセージ シンタックス(ASN.1)

JT-H225.0 は（ゲートキーパーでは必須となる）RAS の為の protocols と（ユーザ・ユーザ情報要素に依存する protocol データユニットでは必須となる）呼制御を定義する。これらの protocols は共に次に示す ASN.1 ツリー構造内で定義される。重要性は H.235 バージョン 1（H235-SECURITY-MESSAGES）の中に見出せる。メッセージや色々な要素の用語の定義は前の節にある。

```

H323-MESSAGES DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

IMPORTS
    SIGNED{ },
    ENCRYPTED{ },
    HASHED{ },
    ChallengeString,
    TimeStamp,
    RandomVal,
    Password,
    EncodedPwdCertToken,
    ClearToken,
    CryptoToken,
    AuthenticationMechanism
FROM H235-SECURITY-MESSAGES
    DataProtocolCapability,
    T38FaxProfile,
    QOSCapability
FROM MULTIMEDIA-SYSTEM-CONTROL;

H323-UserInformation ::= SEQUENCE          -- root for all H.225.0 call signalling messages
{
    h323-uu-pdu      H323-UU-PDU,
    user-data        SEQUENCE
    {
        protocol-discriminator  INTEGER (0..255),
        user-information         OCTET STRING (SIZE(1..131)),
        ...
    } OPTIONAL,
    ...
}

H323-UU-PDU ::= SEQUENCE
{
    h323-message-body  CHOICE
    {
        setup                Setup-UUIE,
        callProceeding       CallProceeding-UUIE,
        connect               Connect-UUIE,
        alerting              Alerting-UUIE,
        information           Information-UUIE,
        releaseComplete       ReleaseComplete-UUIE,
        facility              Facility-UUIE,
        ...,
        progress              Progress-UUIE,
        empty                 NULL, -- used when a Facility message is sent,
                                -- but the Facility-UUIE is not to be invoked
                                -- (possible when transporting supplementary
                                -- services messages in versions prior to
                                -- H.225.0 version 4)
        status                Status-UUIE,
        statusInquiry         StatusInquiry-UUIE,
        setupAcknowledge       SetupAcknowledge-UUIE,
        notify                 Notify-UUIE
    },
    nonStandardData          NonStandardParameter OPTIONAL,
    ...,
    h4501SupplementaryService SEQUENCE OF OCTET STRING OPTIONAL,
                                -- each sequence of octet string is defined as
                                -- H4501SupplementaryService APDU as defined in
                                -- Table 3/H.450.1

one
    h245Tunneling            BOOLEAN,
                                -- if TRUE, tunneling of H.245 messages is enabled
    h245Control               SEQUENCE OF OCTET STRING OPTIONAL,

```

```

nonStandardControl      SEQUENCE OF NonStandardParameter OPTIONAL,
callLinkage             CallLinkage OPTIONAL,
tunnelledSignallingMessage SEQUENCE
{
    tunnelledProtocolID  TunnelledProtocol,      -- tunnelled signalling
                                     -- protocol ID
    messageContent       SEQUENCE OF OCTET STRING, -- sequence of entire
                                     -- message(s)
    tunnellingRequired   NULL OPTIONAL,
    nonStandardData      NonStandardParameter OPTIONAL,
    ...
} OPTIONAL,
provisionalRespToH245Tunneling NULL OPTIONAL,
stimulusControl         StimulusControl OPTIONAL,
genericData             SEQUENCE OF GenericData OPTIONAL
}

StimulusControl ::= SEQUENCE
{
    nonStandard          NonStandardParameter OPTIONAL,
    isText               NULL OPTIONAL,
    h248Message          OCTET STRING OPTIONAL,
    ...
}

Alerting-UUIE ::= SEQUENCE
{
    protocolIdentifier   ProtocolIdentifier,
    destinationInfo     EndpointType,
    h245Address          TransportAddress OPTIONAL,
    ...
    callIdentifier       CallIdentifier,
    h245SecurityMode     H245Security OPTIONAL,
    tokens               SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens         SEQUENCE OF CryptoH323Token OPTIONAL,
    fastStart            SEQUENCE OF OCTET STRING OPTIONAL,
    multipleCalls        BOOLEAN,
    maintainConnection   BOOLEAN,
    alertingAddress      SEQUENCE OF AliasAddress OPTIONAL,
    presentationIndicator PresentationIndicator OPTIONAL,
    screeningIndicator   ScreeningIndicator OPTIONAL,
    fastConnectRefused   NULL OPTIONAL,
    serviceControl       SEQUENCE OF ServiceControlSession OPTIONAL,
    capacity             CallCapacity OPTIONAL,
    featureSet           FeatureSet OPTIONAL
}

CallProceeding-UUIE ::= SEQUENCE
{
    protocolIdentifier   ProtocolIdentifier,
    destinationInfo     EndpointType,
    h245Address          TransportAddress OPTIONAL,
    ...
    callIdentifier       CallIdentifier,
    h245SecurityMode     H245Security OPTIONAL,
    tokens               SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens         SEQUENCE OF CryptoH323Token OPTIONAL,
    fastStart            SEQUENCE OF OCTET STRING OPTIONAL,
    multipleCalls        BOOLEAN,
    maintainConnection   BOOLEAN,
    fastConnectRefused   NULL OPTIONAL,
    featureSet           FeatureSet OPTIONAL
}

Connect-UUIE ::= SEQUENCE
{
    protocolIdentifier   ProtocolIdentifier,
    h245Address          TransportAddress OPTIONAL,
    destinationInfo     EndpointType,
    conferenceID         ConferenceIdentifier,
    ...
    callIdentifier       CallIdentifier,
    h245SecurityMode     H245Security OPTIONAL,
    tokens               SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens         SEQUENCE OF CryptoH323Token OPTIONAL,
    fastStart            SEQUENCE OF OCTET STRING OPTIONAL,
    multipleCalls        BOOLEAN,
    maintainConnection   BOOLEAN,
    language             SEQUENCE OF IA5String (SIZE (1..32)) OPTIONAL,
                                     -- RFC1766 language tag
    connectedAddress     SEQUENCE OF AliasAddress OPTIONAL,
    presentationIndicator PresentationIndicator OPTIONAL,
    screeningIndicator   ScreeningIndicator OPTIONAL,
    fastConnectRefused   NULL OPTIONAL,
}

```

```

        serviceControl          SEQUENCE OF ServiceControlSession OPTIONAL,
        capacity                CallCapacity OPTIONAL,
        featureSet              FeatureSet OPTIONAL
    }

Information-UUIE ::=SEQUENCE
{
    protocolIdentifier          ProtocolIdentifier,
    ...,
    callIdentifier             CallIdentifier,
    tokens                     SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens               SEQUENCE OF CryptoH323Token OPTIONAL,
    fastStart                  SEQUENCE OF OCTET STRING OPTIONAL,
    fastConnectRefused         NULL OPTIONAL,
    circuitInfo                CircuitInfo OPTIONAL
}

ReleaseComplete-UUIE ::= SEQUENCE
{
    protocolIdentifier          ProtocolIdentifier,
    reason                      ReleaseCompleteReason OPTIONAL,
    ...,
    callIdentifier             CallIdentifier,
    tokens                     SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens               SEQUENCE OF CryptoH323Token OPTIONAL,
    busyAddress                SEQUENCE OF AliasAddress OPTIONAL,
    presentationIndicator      PresentationIndicator OPTIONAL,
    screeningIndicator         ScreeningIndicator OPTIONAL,
    capacity                   CallCapacity OPTIONAL,
    serviceControl             SEQUENCE OF ServiceControlSession OPTIONAL,
    featureSet                 FeatureSet OPTIONAL
}

ReleaseCompleteReason ::= CHOICE
{
    noBandwidth                NULL, -- bandwidth taken away or ARQ denied
    gatekeeperResources        NULL, -- exhausted
    unreachableDestination     NULL, -- no transport path to the destination
    destinationRejection       NULL, -- rejected at destination
    invalidRevision            NULL,
    noPermission                NULL, -- called party's gatekeeper rejects
    unreachableGatekeeper      NULL, -- terminal cannot reach gatekeeper
                                -- for ARQ

    gatewayResources           NULL,
    badFormatAddress           NULL,
    adaptiveBusy                NULL, -- call is dropping due to LAN crowding
    inConf                      NULL, -- called party busy
    undefinedReason            NULL,

    ...,
    facilityCallDeflection     NULL, -- call was deflected using a Facility
                                -- message

    securityDenied              NULL, -- incompatible security settings
    calledPartyNotRegistered    NULL, -- used by gatekeeper when endpoint has
                                -- preGrantedARQ to bypass ARQ/ACF

    callerNotRegistered         NULL, -- used by gatekeeper when endpoint has
                                -- preGrantedARQ to bypass ARQ/ACF

    newConnectionNeeded        NULL, -- indicates that the Setup was not
                                -- accepted on this connection, but that
                                -- the Setup may be accepted on
                                -- a new connection

    nonStandardReason           NonStandardParameter,
    replaceWithConferenceInvite ConferenceIdentifier, -- call dropped due to
                                -- subsequent invitation
                                -- to a conference
                                -- (see H.323 8.4.3.8)

    genericDataReason           NULL,
    neededFeatureNotSupported   NULL,
    tunnelledSignallingRejected NULL,
    invalidCID                  NULL,
    securityError                SecurityErrors,
    hopCountExceeded            NULL
}

Setup-UUIE ::= SEQUENCE
{
    protocolIdentifier          ProtocolIdentifier,
    h245Address                 TransportAddress OPTIONAL,
    sourceAddress                SEQUENCE OF AliasAddress OPTIONAL,
    sourceInfo                   EndpointType,
    destinationAddress           SEQUENCE OF AliasAddress OPTIONAL,
    destCallSignalAddress        TransportAddress OPTIONAL,
    destExtraCallInfo            SEQUENCE OF AliasAddress OPTIONAL,
    destExtraCRV                 SEQUENCE OF CallReferenceValue OPTIONAL,

```



```

activeMC                               BOOLEAN,
conferenceID                           ConferenceIdentifier,
conferenceGoal                           CHOICE
{
    create                               NULL,
    join                                 NULL,
    invite                               NULL,
    ...,
    capability-negotiation               NULL,
    callIndependentSupplementaryService  NULL
},
callServices                           QseriesOptions OPTIONAL,
callType                               CallType,
...,
sourceCallSignalAddress                 TransportAddress OPTIONAL,
remoteExtensionAddress                  AliasAddress OPTIONAL,
callIdentifier                           CallIdentifier,
h245SecurityCapability                  SEQUENCE OF H245Security OPTIONAL,
tokens                                  SEQUENCE OF ClearToken OPTIONAL,
cryptoTokens                            SEQUENCE OF CryptoH323Token OPTIONAL,
fastStart                               SEQUENCE OF OCTET STRING OPTIONAL,
mediaWaitForConnect                     BOOLEAN,
canOverlapSend                           BOOLEAN,
endpointIdentifier                       EndpointIdentifier OPTIONAL,
multipleCalls                            BOOLEAN,
maintainConnection                       BOOLEAN,
connectionParameters                     SEQUENCE
{
    connectionType                       ScnConnectionType,
    numberOfScnConnections                 INTEGER (0..65535),
    connectionAggregation                 ScnConnectionAggregation,
    ...
} OPTIONAL,
language                                 SEQUENCE OF IA5String (SIZE (1..32)) OPTIONAL,
                                          -- RFC1766 language tag
presentationIndicator                   PresentationIndicator OPTIONAL,
screeningIndicator                       ScreeningIndicator OPTIONAL,
serviceControl                           SEQUENCE OF ServiceControlSession OPTIONAL,
symmetricOperationRequired              NULL OPTIONAL,
capacity                                 CallCapacity OPTIONAL,
circuitInfo                              CircuitInfo OPTIONAL,
desiredProtocols                         SEQUENCE OF SupportedProtocols OPTIONAL,
neededFeatures                           SEQUENCE OF FeatureDescriptor OPTIONAL,
desiredFeatures                           SEQUENCE OF FeatureDescriptor OPTIONAL,
supportedFeatures                         SEQUENCE OF FeatureDescriptor OPTIONAL,
parallelH245Control                       SEQUENCE OF OCTET STRING OPTIONAL,
additionalSourceAddresses                 SEQUENCE OF ExtendedAliasAddress OPTIONAL,
hopCount                                 INTEGER (1..31) OPTIONAL
}

ScnConnectionType ::= CHOICE
{
    unknown                               NULL, -- should be selected when connection type is unknown
    bChannel                               NULL, -- each individual connection on the SCN is 64kbps.
                                          -- Note that where SCN delivers 56kbps usable data, the
                                          -- actual bandwidth allocated on SCN is still
                                          -- 64kbps.
    hybrid2x64                             NULL, -- each connection is a 128kbps hybrid call
    hybrid384                               NULL, -- each connection is an H0 (384kbps) hybrid call
    hybrid1536                              NULL, -- each connection is an H11 (1536kbps) hybrid call
    hybrid1920                              NULL, -- each connection is an H12 (1920kbps) hybrid call
    multirate                               NULL, -- bandwidth supplied by SCN using multirate.
                                          -- In this case, the information transfer rate octet
                                          -- in the bearer capability shall be set to multirate
                                          -- and the rate multiplier octet shall denote the
                                          -- number of B channels.
    ...
}

ScnConnectionAggregation ::= CHOICE
{
    auto                                   NULL, -- aggregation mechanism is unknown
    none                                   NULL, -- call produced using a single SCN connection
    h221                                   NULL, -- use H.221 framing to aggregate the connections
    bonded-mode1                           NULL, -- use ISO/IEC 13871 bonding mode 1.
                                          -- Use bonded-mode1 to signal a bonded call if the
                                          -- precise bonding mode to be used is unknown.
    bonded-mode2                           NULL, -- use ISO/IEC 13871 bonding mode 2
    bonded-mode3                           NULL, -- use ISO/IEC 13871 bonding mode 3
    ...
}

PresentationIndicator ::= CHOICE

```

```

{
    presentationAllowed          NULL,
    presentationRestricted       NULL,
    addressNotAvailable         NULL,
    ...
}

ScreeningIndicator ::= ENUMERATED
{
    userProvidedNotScreened (0),
        -- number was provided by a remote user
        -- and has not been screened by a gatekeeper
    userProvidedVerifiedAndPassed (1),
        -- number was provided by user
        -- equipment (or by a remote network), and has
        -- been screened by a gatekeeper
    userProvidedVerifiedAndFailed (2),
        -- number was provided by user
        -- equipment (or by a remote network), and the
        -- gatekeeper has determined that the
        -- information is incorrect
    networkProvided (3),
        -- number was provided by a gatekeeper
    ...
}

Facility-UUIE ::= SEQUENCE
{
    protocolIdentifier           ProtocolIdentifier,
    alternativeAddress           TransportAddress OPTIONAL,
    alternativeAliasAddress      SEQUENCE OF AliasAddress OPTIONAL,
    conferenceID                ConferenceIdentifier OPTIONAL,
    reason                       FacilityReason,
    ...,
    callIdentifier              CallIdentifier,
    destExtraCallInfo           SEQUENCE OF AliasAddress OPTIONAL,
    remoteExtensionAddress      AliasAddress OPTIONAL,
    tokens                      SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens                SEQUENCE OF CryptoH323Token OPTIONAL,
    conferences                 SEQUENCE OF ConferenceList OPTIONAL,
    h245Address                 TransportAddress OPTIONAL,
    fastStart                   SEQUENCE OF OCTET STRING OPTIONAL,
    multipleCalls               BOOLEAN,
    maintainConnection          BOOLEAN,
    fastConnectRefused          NULL OPTIONAL,
    serviceControl              SEQUENCE OF ServiceControlSession OPTIONAL,
    circuitInfo                 CircuitInfo OPTIONAL,
    featureSet                   FeatureSet OPTIONAL,
    destinationInfo             EndpointType OPTIONAL,
    h245SecurityMode            H245Security OPTIONAL
}

ConferenceList ::= SEQUENCE
{
    conferenceID                ConferenceIdentifier OPTIONAL,
    conferenceAlias              AliasAddress OPTIONAL,
    nonStandardData             NonStandardParameter OPTIONAL,
    ...
}

FacilityReason ::= CHOICE
{
    routeCallToGatekeeper      NULL,    -- call must use gatekeeper model
                                     -- gatekeeper is alternativeAddress
    callForwarded              NULL,
    routeCallToMC              NULL,
    undefinedReason            NULL,
    ...,
    conferenceListChoice       NULL,
    startH245                  NULL,    -- recipient should connect to h245Address
    noH245                     NULL,    -- endpoint does not support H.245
    newTokens                  NULL,
    featureSetUpdate            NULL,
    forwardedElements           NULL,
    transportedInformation      NULL
}

Progress-UUIE ::= SEQUENCE
{
    protocolIdentifier           ProtocolIdentifier,
    destinationInfo             EndpointType,
    h245Address                 TransportAddress OPTIONAL,
    callIdentifier              CallIdentifier,

```

```

    h245SecurityMode          H245Security OPTIONAL,
    tokens                    SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens              SEQUENCE OF CryptoH323Token OPTIONAL,
    fastStart                 SEQUENCE OF OCTET STRING OPTIONAL,
    ...,
    multipleCalls             BOOLEAN,
    maintainConnection        BOOLEAN,
    fastConnectRefused        NULL OPTIONAL
}

TransportAddress ::= CHOICE
{
    ipAddress                SEQUENCE
    {
        ip                    OCTET STRING (SIZE(4)),
        port                  INTEGER(0..65535)
    },
    ipSourceRoute            SEQUENCE
    {
        ip                    OCTET STRING (SIZE(4)),
        port                  INTEGER(0..65535),
        route                 SEQUENCE OF OCTET STRING (SIZE(4)),
        routing               CHOICE
        {
            strict NULL,
            loose  NULL,
            ...
        },
        ...
    },
    ipxBAddress              SEQUENCE
    {
        node                  OCTET STRING (SIZE(6)),
        netnum                OCTET STRING (SIZE(4)),
        port                  OCTET STRING (SIZE(2))
    },
    ip6Address               SEQUENCE
    {
        ip                    OCTET STRING (SIZE(16)),
        port                  INTEGER(0..65535),
        ...
    },
    netBios                  OCTET STRING (SIZE(16)),
    nsap                     OCTET STRING (SIZE(1..20)),
    nonStandardAddress        NonStandardParameter,
    ...
}

Status-UUIE ::= SEQUENCE
{
    protocolIdentifier        ProtocolIdentifier,
    callIdentifier            CallIdentifier,
    tokens                    SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens              SEQUENCE OF CryptoH323Token OPTIONAL,
    ...
}

StatusInquiry-UUIE ::= SEQUENCE
{
    protocolIdentifier        ProtocolIdentifier,
    callIdentifier            CallIdentifier,
    tokens                    SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens              SEQUENCE OF CryptoH323Token OPTIONAL,
    ...
}

SetupAcknowledge-UUIE ::= SEQUENCE
{
    protocolIdentifier        ProtocolIdentifier,
    callIdentifier            CallIdentifier,
    tokens                    SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens              SEQUENCE OF CryptoH323Token OPTIONAL,
    ...
}

Notify-UUIE ::= SEQUENCE
{
    protocolIdentifier        ProtocolIdentifier,
    callIdentifier            CallIdentifier,
    tokens                    SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens              SEQUENCE OF CryptoH323Token OPTIONAL,
    .....,
    connectedAddress          SEQUENCE OF AliasAddress OPTIONAL,
    presentationIndicator     PresentationIndicator OPTIONAL,
}

```

```

        screeningIndicator      ScreeningIndicator OPTIONAL
    }

-- Beginning of common message elements section

EndpointType ::= SEQUENCE
{
    nonStandardData      NonStandardParameter OPTIONAL,
    vendor                VendorIdentifier OPTIONAL,
    gatekeeper            GatekeeperInfo OPTIONAL,
    gateway               GatewayInfo OPTIONAL,
    mcu                   McuInfo OPTIONAL,          -- mc must be set as well
    terminal              TerminalInfo OPTIONAL,
    mc                    BOOLEAN,                  -- shall not be set by itself
    undefinedNode        BOOLEAN,
    ...
    set                   BIT STRING (SIZE(32)) OPTIONAL,
                                -- shall not be used with mc, gatekeeper
                                -- code points for the various SET devices
                                -- are defined in the respective SET Annexes

    supportedTunnelledProtocols SEQUENCE OF TunnelledProtocol OPTIONAL
                                -- list of supported tunnelled protocols
}

GatewayInfo ::= SEQUENCE
{
    protocol              SEQUENCE OF SupportedProtocols OPTIONAL,
    nonStandardData      NonStandardParameter OPTIONAL,
    ...
}

SupportedProtocols ::= CHOICE
{
    nonStandardData      NonStandardParameter,
    h310                  H310Caps,
    h320                  H320Caps,
    h321                  H321Caps,
    h322                  H322Caps,
    h323                  H323Caps,
    h324                  H324Caps,
    voice                 VoiceCaps,
    t120-only            T120OnlyCaps,
    ...
    nonStandardProtocol  NonStandardProtocol,
    t38FaxAnnexbOnly    T38FaxAnnexbOnlyCaps,
    sip                  SIPCaps
}

H310Caps ::= SEQUENCE
{
    nonStandardData      NonStandardParameter OPTIONAL,
    ...
    dataRatesSupported   SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes    SEQUENCE OF SupportedPrefix
}

H320Caps ::= SEQUENCE
{
    nonStandardData      NonStandardParameter OPTIONAL,
    ...
    dataRatesSupported   SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes    SEQUENCE OF SupportedPrefix
}

H321Caps ::= SEQUENCE
{
    nonStandardData      NonStandardParameter OPTIONAL,
    ...
    dataRatesSupported   SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes    SEQUENCE OF SupportedPrefix
}

H322Caps ::= SEQUENCE
{
    nonStandardData      NonStandardParameter OPTIONAL,
    ...
    dataRatesSupported   SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes    SEQUENCE OF SupportedPrefix
}

H323Caps ::= SEQUENCE
{
    nonStandardData      NonStandardParameter OPTIONAL,
    ...

```

```

        dataRatesSupported      SEQUENCE OF DataRate OPTIONAL,
        supportedPrefixes      SEQUENCE OF SupportedPrefix
    }
H324Caps ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    ...,
    dataRatesSupported        SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes        SEQUENCE OF SupportedPrefix
}
VoiceCaps ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    ...,
    dataRatesSupported        SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes        SEQUENCE OF SupportedPrefix
}
T120OnlyCaps ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    ...,
    dataRatesSupported        SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes        SEQUENCE OF SupportedPrefix
}
NonStandardProtocol ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    dataRatesSupported        SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes        SEQUENCE OF SupportedPrefix,
    ...
}
T38FaxAnnexbOnlyCaps ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    dataRatesSupported        SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes        SEQUENCE OF SupportedPrefix,
    t38FaxProtocol            DataProtocolCapability,
    t38FaxProfile              T38FaxProfile,
    ...
}
SIPCaps ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    dataRatesSupported        SEQUENCE OF DataRate OPTIONAL,
    supportedPrefixes        SEQUENCE OF SupportedPrefix OPTIONAL,
    ...
}
McuInfo ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    ...,
    protocol                  SEQUENCE OF SupportedProtocols OPTIONAL
}
TerminalInfo ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    ...
}
GatekeeperInfo ::= SEQUENCE
{
    nonStandardData            NonStandardParameter OPTIONAL,
    ...
}
VendorIdentifier ::= SEQUENCE
{
    vendor                    H221NonStandard,
    productId                  OCTET STRING (SIZE(1..256)) OPTIONAL, -- per vendor
    versionId                  OCTET STRING (SIZE(1..256)) OPTIONAL, -- per product
    ...,
    enterpriseNumber          OBJECT IDENTIFIER OPTIONAL
}
H221NonStandard ::= SEQUENCE
{
    t35CountryCode            INTEGER(0..255),

```



```

PrivatePartyNumber ::= SEQUENCE
{
    privateTypeOfNumber      PrivateTypeOfNumber,
    privateNumberDigits      NumberDigits
}

NumberDigits ::= IA5String (SIZE (1..128)) (FROM ("0123456789#*,"))

PublicTypeOfNumber ::= CHOICE
{
    unknown                  NULL,
                                -- if used number digits carry prefix
                                -- indicating type
                                -- of number according to national
                                -- recommendations.

    internationalNumber      NULL,
    nationalNumber           NULL,
    networkSpecificNumber    NULL,
                                -- not used, value reserved

    subscriberNumber        NULL,
    abbreviatedNumber        NULL,
                                -- valid only for called party number at
                                -- the outgoing access, network
                                -- substitutes
                                -- appropriate number.

    ...
}

PrivateTypeOfNumber ::= CHOICE
{
    unknown                  NULL,
    level2RegionalNumber     NULL,
    level1RegionalNumber     NULL,
    pISNSpecificNumber       NULL,
    localNumber              NULL,
    abbreviatedNumber        NULL,
    ...
}

MobileUIM ::= CHOICE
{
    ansi-41-uim ANSI-41-UIM,    -- Americas standards Wireless Networks
    gsm-uim GSM-UIM,           -- European standards Wireless Networks
    ...
}

TBCD-STRING ::= IA5String (FROM ("0123456789#*abc"))

ANSI-41-UIM ::= SEQUENCE
{
    imsi                    TBCD-STRING (SIZE (3..16)) OPTIONAL,
    min                     TBCD-STRING (SIZE (3..16)) OPTIONAL,
    mdn                     TBCD-STRING (SIZE (3..16)) OPTIONAL,
    msisdn                  TBCD-STRING (SIZE (3..16)) OPTIONAL,
    esn                     TBCD-STRING (SIZE (16)) OPTIONAL,
    mscid                   TBCD-STRING (SIZE (3..16)) OPTIONAL,
    system-id CHOICE
    {
        sid                 TBCD-STRING (SIZE (1..4)),
        mid                 TBCD-STRING (SIZE (1..4)),
        ...
    },
    systemMyTypeCode        OCTET STRING (SIZE (1)) OPTIONAL,
    systemAccessType        OCTET STRING (SIZE (1)) OPTIONAL,
    qualificationInformationCode OCTET STRING (SIZE (1)) OPTIONAL,
    sesn                   TBCD-STRING (SIZE (16)) OPTIONAL,
    soc                    TBCD-STRING (SIZE (3..16)) OPTIONAL,
    ...
    -- IMSI refers to International Mobile Station Identification
    -- MIN refers to Mobile Identification Number
    -- MDN refers to Mobile Directory Number
    -- MSISDN refers to Mobile Station ISDN number
    -- ESN Refers to Electronic Serial Number
    -- MSCID refers to Mobile Switching Center number + Market ID or System ID
    -- SID refers to System Identification and MID refers to Market
    -- Identification
    -- SystemMyTypeCode refers to vendor identification number
    -- SystemAccessType refers to the system access type like power down
    -- registration or call
    -- origination or Short Message response etc.
    -- Qualification Information Code refers to the validity
    -- SESN Refers to SIM Electronic Serial Number for Security purposes of

```

```

    -- User Identification
    -- SOC refers to System Operator Code
}

GSM-UIR ::= SEQUENCE
{
    imsi                TBCD-STRING (SIZE (3..16)) OPTIONAL,
    tmsi                OCTET STRING (SIZE (1..4)) OPTIONAL,
    msisdn              TBCD-STRING (SIZE (3..16)) OPTIONAL,
    imei                TBCD-STRING (SIZE (15..16)) OPTIONAL,
    hplmn               TBCD-STRING (SIZE (1..4)) OPTIONAL,
    vplmn               TBCD-STRING (SIZE (1..4)) OPTIONAL,
    -- IMSI refers to International Mobile Station Identification
    -- MSISDN refers to Mobile Station ISDN number
    -- IMEI Refers to International Mobile Equipment Identification
    -- VPLMN or HPLMN refers to Visiting or Home Public Land Mobile Network
    -- number
    ...
}

IsupNumber ::= CHOICE
{
    e164Number          IsupPublicPartyNumber,
                        -- the numbering plan is according to
                        -- ITU-T Recs E.163 and E.164.
    dataPartyNumber    IsupDigits,
                        -- not used, value reserved.
    telexPartyNumber   IsupDigits,
                        -- not used, value reserved.
    privateNumber       IsupPrivatePartyNumber,
                        -- the numbering plan is according to
                        -- ISO/IEC 11571.
    nationalStandardPartyNumber IsupDigits,
                        -- not used, value reserved.
    ...
}

IsupPublicPartyNumber ::= SEQUENCE
{
    natureOfAddress     NatureOfAddress,
    address             IsupDigits,
    ...
}

IsupPrivatePartyNumber ::= SEQUENCE
{
    privateTypeOfNumber PrivateTypeOfNumber,
    address             IsupDigits,
    ...
}

NatureOfAddress ::= CHOICE
{
    unknown            NULL,
    subscriberNumber   NULL,
    nationalNumber     NULL,
    internationalNumber NULL,
    networkSpecificNumber NULL,
    routingNumberNationalFormat NULL,
    routingNumberNetworkSpecificFormat NULL,
    routingNumberWithCalledDirectoryNumber NULL,
    ...
}

IsupDigits ::= IA5String (SIZE (1..128)) (FROM ("0123456789ABCDE"))

ExtendedAliasAddress ::= SEQUENCE
{
    address            AliasAddress,
    presentationIndicator PresentationIndicator OPTIONAL,
    screeningIndicator ScreeningIndicator OPTIONAL,
    ...
}

Endpoint ::= SEQUENCE
{
    nonStandardData    NonStandardParameter OPTIONAL,
    aliasAddress        SEQUENCE OF AliasAddress OPTIONAL,
    callSignalAddress  SEQUENCE OF TransportAddress OPTIONAL,
    rasAddress          SEQUENCE OF TransportAddress OPTIONAL,
    endpointType        EndpointType OPTIONAL,
    tokens              SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens        SEQUENCE OF CryptoH323Token OPTIONAL,
    priority             INTEGER(0..127) OPTIONAL,
    remoteExtensionAddress SEQUENCE OF AliasAddress OPTIONAL,
    destExtraCallInfo  SEQUENCE OF AliasAddress OPTIONAL,
}

```



```

    ...,
    alternateTransportAddresses      AlternateTransportAddresses OPTIONAL,
    circuitInfo                     CircuitInfo      OPTIONAL,
    featureSet                       FeatureSet      OPTIONAL
}

AlternateTransportAddresses ::= SEQUENCE
{
    annexE          SEQUENCE OF TransportAddress OPTIONAL,
    ...,
    sctp             SEQUENCE OF TransportAddress OPTIONAL
}

UseSpecifiedTransport ::= CHOICE
{
    tcp              NULL,
    annexE          NULL,
    ...,
    sctp            NULL
}

AlternateGK ::= SEQUENCE
{
    rasAddress      TransportAddress,
    gatekeeperIdentifier  GatekeeperIdentifier OPTIONAL,
    needToRegister  BOOLEAN,
    priority        INTEGER (0..127),
    ...
}

AltGKInfo ::=SEQUENCE
{
    alternateGatekeeper  SEQUENCE OF AlternateGK,
    altGKisPermanent    BOOLEAN,
    ...
}

SecurityServiceMode ::= CHOICE
{
    nonStandard      NonStandardParameter,
    none             NULL,
    default          NULL,
    ...              -- can be extended with other specific modes
}

SecurityCapabilities ::= SEQUENCE
{
    nonStandard      NonStandardParameter OPTIONAL,
    encryption       SecurityServiceMode,
    authenticaton    SecurityServiceMode,
    integrity        SecurityServiceMode,
    ...
}

SecurityErrors ::= CHOICE
{
    securityWrongSyncTime      NULL,  -- either time server
                                   -- problem or network delay
    securityReplay             NULL,  -- replay attack encountered
    securityWrongGeneralID     NULL,  -- wrong general ID
    securityWrongSendersID     NULL,  -- wrong senders ID
    securityIntegrityFailed    NULL,  -- integrity check failed
    securityWrongOID           NULL,  -- wrong token OIDs or crypto alg
                                   -- OIDs
    securityDHmismatch         NULL,  -- mismatch of DH parameters
    securityCertificateExpired  NULL,  -- certificate has expired
    securityCertificateDateInvalid  NULL,  -- certificate is not yet valid
    securityCertificateRevoked     NULL,  -- certificate was found revoked
    securityCertificateNotReadable  NULL,  -- decoding error
    securityCertificateSignatureInvalid  NULL,  -- wrong signature in the
                                   -- certificate
    securityCertificateMissing     NULL,  -- no certificate available
    securityCertificateIncomplete  NULL,  -- missing expected certificate
                                   -- extensions
    securityUnsupportedCertificateAlgOID  NULL,  -- crypto algs not understood
    securityUnknownCA             NULL,  -- CA/root certificate could not
                                   -- be found
    ...
}

SecurityErrors2 ::= CHOICE
{
    securityWrongSyncTime      NULL,  -- either time server problem or network

```

```

        securityReplay                NULL, -- delay
        securityWrongGeneralID        NULL, -- replay attack encountered
        securityWrongSendersID        NULL, -- wrong general ID
        securityIntegrityFailed        NULL, -- wrong senders ID
        securityWrongOID               NULL, -- integrity check failed
        ...
    }

H245Security ::= CHOICE
{
    nonStandard                NonStandardParameter,
    noSecurity                 NULL,
    tls                        SecurityCapabilities,
    ipsec                      SecurityCapabilities,
    ...
}

QseriesOptions ::= SEQUENCE
{
    q932Full                   BOOLEAN, -- if true, indicates full support for Q.932
    q951Full                   BOOLEAN, -- if true, indicates full support for Q.951
    q952Full                   BOOLEAN, -- if true, indicates full support for Q.952
    q953Full                   BOOLEAN, -- if true, indicates full support for Q.953
    q955Full                   BOOLEAN, -- if true, indicates full support for Q.955
    q956Full                   BOOLEAN, -- if true, indicates full support for Q.956
    q957Full                   BOOLEAN, -- if true, indicates full support for Q.957
    q954Info                   Q954Details,
    ...
}

Q954Details ::= SEQUENCE
{
    conferenceCalling          BOOLEAN,
    threePartyService          BOOLEAN,
    ...
}

GloballyUniqueID              ::= OCTET STRING (SIZE(16))
ConferenceIdentifier ::= GloballyUniqueID
RequestSeqNum                  ::= INTEGER (1..65535)
GatekeeperIdentifier ::= BMPString (SIZE(1..128))
Bandwidth                      ::= INTEGER (0..4294967295) -- in 100s of bits
CallReferenceValue             ::= INTEGER (0..65535)
EndpointIdentifier             ::= BMPString (SIZE(1..128))
ProtocolIdentifier             ::= OBJECT IDENTIFIER
TimeToLive                     ::= INTEGER (1..4294967295) -- in seconds
FeatureDescriptor              ::= GenericData

CallIdentifier ::= SEQUENCE
{
    guid                        GloballyUniqueID,
    ...
}

EncryptIntAlg ::= CHOICE
{
    -- core encryption algorithms for RAS message integrity
    nonStandard                NonStandardParameter,
    isoAlgorithm                OBJECT IDENTIFIER, -- defined in ISO/IEC 9979
    ...
}

NonIsoIntegrityMechanism ::= CHOICE
{
    -- HMAC mechanism used, no truncation, tagging may be necessary!
    hMAC-MD5                   NULL,
    hMAC-iso10118-2-s          EncryptIntAlg, -- according to ISO/IEC 10118-2 using
    -- EncryptIntAlg as core block
    -- encryption algorithm (short MAC)
    hMAC-iso10118-2-1         EncryptIntAlg, -- according to ISO/IEC 10118-2 using
    -- EncryptIntAlg as core block
    -- encryption algorithm (long MAC)
    hMAC-iso10118-3           OBJECT IDENTIFIER, -- according to ISO/IEC 10118-3 using
    -- OID as hash function (OID is
    -- SHA-1,
    -- RIPE-MD160,
    -- RIPE-MD128)
    ...
}

IntegrityMechanism ::= CHOICE
{
    -- for RAS message integrity
    nonStandard                NonStandardParameter,

```

```

    digSig          NULL,                -- indicates to apply a digital signature
    iso9797         OBJECT IDENTIFIER,   -- according to ISO/IEC 9797 using OID as
                                         -- core encryption algorithm (X-CBC MAC)
    nonIsoIM       NonIsoIntegrityMechanism,
    ...
}

ICV ::= SEQUENCE
{
    algorithmOID   OBJECT IDENTIFIER,   -- the algorithm used to compute the
                                         -- signature
    icv            BIT STRING          -- the computed cryptographic
                                         -- integrity check value or signature
}

FastStartToken ::= ClearToken (WITH COMPONENTS {..., timeStamp PRESENT, dhkey PRESENT,
generalID PRESENT
                                         -- set to "alias" -- })

EncodedFastStartToken ::= TYPE-IDENTIFIER.&Type (FastStartToken)
CryptoH323Token ::= CHOICE
{
    cryptoEPPwdHash SEQUENCE
    {
        alias      AliasAddress,        -- alias of entity generating hash
        timeStamp  TimeStamp,           -- timestamp used in hash
                                         token
                                         HASHED
                                         { EncodedPwCertToken
                                         -- generalID set to
                                         -- "alias" -- }
    },
    cryptoGKPwdHash SEQUENCE
    {
        gatekeeperId GatekeeperIdentifier, -- GatekeeperID of GK generating
                                         -- hash
        timeStamp    TimeStamp,           -- timestamp used in hash
        token        HASHED { EncodedPwCertToken -- generalID set to
                                         -- Gatekeeperid -- }
    },
    cryptoEPPwdEncr ENCRYPTED { EncodedPwCertToken -- generalID set to
                                -- Gatekeeperid -- },
    cryptoGKPwdEncr ENCRYPTED { EncodedPwCertToken -- generalID set to
                                -- Gatekeeperid -- },
    cryptoEPCert   SIGNED { EncodedPwCertToken -- generalID set to
                                -- Gatekeeperid -- },
    cryptoGKCert   SIGNED { EncodedPwCertToken -- generalID set to alias -- },
    cryptoFastStart SIGNED { EncodedFastStartToken },
    nestedcryptoToken CryptoToken,
    ...
}

DataRate ::= SEQUENCE
{
    nonStandardData NonStandardParameter OPTIONAL,
    channelRate     BandWidth,
    channelMultiplier INTEGER (1..256) OPTIONAL,
    ...
}

CallLinkage ::= SEQUENCE
{
    globalCallId GloballyUniqueID OPTIONAL,
    threadId     GloballyUniqueID OPTIONAL,
    ...
}

SupportedPrefix ::= SEQUENCE
{
    nonStandardData NonStandardParameter OPTIONAL,
    prefix          AliasAddress,
    ...
}

CapacityReportingCapability ::= SEQUENCE
{
    canReportCallCapacity BOOLEAN,
    ...
}

CapacityReportingSpecification ::= SEQUENCE
{
    when SEQUENCE

```

```

        {
            callStart          NULL OPTIONAL,
            callEnd            NULL OPTIONAL,
            ...
        },
        ...
    }

CallCapacity ::= SEQUENCE
{
    maximumCallCapacity      CallCapacityInfo OPTIONAL,
    currentCallCapacity      CallCapacityInfo OPTIONAL,
    ...
}

CallCapacityInfo ::= SEQUENCE
{
    voiceGwCallsAvailable    SEQUENCE OF CallsAvailable OPTIONAL,
    h310GwCallsAvailable     SEQUENCE OF CallsAvailable OPTIONAL,
    h320GwCallsAvailable     SEQUENCE OF CallsAvailable OPTIONAL,
    h321GwCallsAvailable     SEQUENCE OF CallsAvailable OPTIONAL,
    h322GwCallsAvailable     SEQUENCE OF CallsAvailable OPTIONAL,
    h323GwCallsAvailable     SEQUENCE OF CallsAvailable OPTIONAL,
    h324GwCallsAvailable     SEQUENCE OF CallsAvailable OPTIONAL,
    t120OnlyGwCallsAvailable SEQUENCE OF CallsAvailable OPTIONAL,
    t38FaxAnnexbOnlyGwCallsAvailable SEQUENCE OF CallsAvailable OPTIONAL,
    terminalCallsAvailable   SEQUENCE OF CallsAvailable OPTIONAL,
    mcuCallsAvailable        SEQUENCE OF CallsAvailable OPTIONAL,
    ...,
    sipGwCallsAvailable      SEQUENCE OF CallsAvailable OPTIONAL
}

CallsAvailable ::= SEQUENCE
{
    calls                    INTEGER (0..4294967295),
    group                    IA5String (SIZE (1..128)) OPTIONAL,
    ...,
    carrier CarrierInfo     OPTIONAL
}

CircuitInfo ::= SEQUENCE
{
    sourceCircuitID          CircuitIdentifier OPTIONAL,
    destinationCircuitID    CircuitIdentifier OPTIONAL,
    genericData              SEQUENCE OF GenericData OPTIONAL,
    ...
}

CircuitIdentifier ::= SEQUENCE
{
    cic                      CicInfo OPTIONAL,
    group                    GroupID OPTIONAL,
    ...,
    carrier                  CarrierInfo OPTIONAL
}

CicInfo ::= SEQUENCE
{
    cic                      SEQUENCE OF OCTET STRING (SIZE (2..4)),
    pointCode                OCTET STRING (SIZE (2..5)),
    ...
}

GroupID ::= SEQUENCE
{
    member                   SEQUENCE OF INTEGER (0..65535) OPTIONAL,
    group                    IA5String (SIZE (1..128)),
    ...
}

CarrierInfo ::= SEQUENCE
{
    carrierIdentificationCode OCTET STRING (SIZE (3..4)) OPTIONAL,
    carrierName               IA5String (SIZE (1..128)) OPTIONAL,
    ...
}

ServiceControlDescriptor ::= CHOICE
{
    url                      IA5String (SIZE(0..512)), -- indicates a URL-
                                                                -- referenced
                                                                -- protocol/resource
    signal                   H248SignalsDescriptor,
    nonStandard              NonStandardParameter,
}

```

```

        callCreditServiceControl    CallCreditServiceControl,
        ...
    }

ServiceControlSession ::= SEQUENCE
{
    sessionId                INTEGER (0..255),
    contents                  ServiceControlDescriptor OPTIONAL,
    reason CHOICE
    {
        open                  NULL,
        refresh               NULL,
        close                  NULL,
        ...
    },
    ...
}

RasUsageInfoTypes ::= SEQUENCE
{
    nonStandardUsageTypes    SEQUENCE OF NonStandardParameter,
    startTime                NULL OPTIONAL,
    endTime                  NULL OPTIONAL,
    terminationCause         NULL OPTIONAL,
    ...
}

RasUsageSpecification ::= SEQUENCE
{
    when SEQUENCE
    {
        start                 NULL OPTIONAL,
        end                   NULL OPTIONAL,
        inIrr                 NULL OPTIONAL,
        ...
    },
    callStartingPoint SEQUENCE
    {
        alerting              NULL OPTIONAL,
        connect               NULL OPTIONAL,
        ...
    } OPTIONAL,
    required                  RasUsageInfoTypes,
    ...
}

RasUsageInformation ::= SEQUENCE
{
    nonStandardUsageFields   SEQUENCE OF NonStandardParameter,
    alertingTime              TimeStamp OPTIONAL,
    connectTime               TimeStamp OPTIONAL,
    endTime                   TimeStamp OPTIONAL,
    ...
}

CallTerminationCause ::= CHOICE
{
    releaseCompleteReason    ReleaseCompleteReason,
    releaseCompleteCauseIE   OCTET STRING (SIZE(2..32)),
    ...
}

BandwidthDetails ::= SEQUENCE
{
    sender                    BOOLEAN,                -- TRUE=sender, FALSE=receiver
    multicast                  BOOLEAN,                -- TRUE if stream is multicast
    bandwidth                  BandWidth,              -- Bandwidth used for stream
    rtcpAddresses              TransportChannelInfo,    -- RTCP addresses for media
    ...
}
stream

CallCreditCapability ::= SEQUENCE
{
    canDisplayAmountString    BOOLEAN OPTIONAL,
    canEnforceDurationLimit   BOOLEAN OPTIONAL,
    ...
}

CallCreditServiceControl ::= SEQUENCE
{
    amountString               BMPString (SIZE (1..512)) OPTIONAL, -- (Unicode)
    billingMode CHOICE
    {

```

```

        credit          NULL,
        debit           NULL,
        ...
    } OPTIONAL,
    callDurationLimit  INTEGER
(1..4294967295) OPTIONAL, -- in seconds
    enforceCallDurationLimit  BOOLEAN OPTIONAL,
    callStartingPoint CHOICE
    {
        alerting        NULL,
        connect         NULL,
        ...
    } OPTIONAL,
    ...
}

GenericData ::= SEQUENCE
{
    id                GenericIdentifier,
    parameters        SEQUENCE (SIZE (1..512)) OF EnumeratedParameter
OPTIONAL,
    ...
}

GenericIdentifier ::= CHOICE
{
    standard          INTEGER (0..16383,...),
    oid               OBJECT IDENTIFIER,
    nonStandard       GloballyUniqueID,
    ...
}

EnumeratedParameter ::= SEQUENCE
{
    id                GenericIdentifier,
    content           Content OPTIONAL,
    ...
}

Content ::= CHOICE
{
    raw               OCTET STRING,
    text              IA5String,
    unicode           BMPString,
    bool              BOOLEAN,
    number8           INTEGER (0..255),
    number16          INTEGER (0..65535),
    number32          INTEGER (0..4294967295),
    id                GenericIdentifier,
    alias             AliasAddress,
    transport         TransportAddress,
    compound           SEQUENCE (SIZE (1..512)) OF EnumeratedParameter,
    nested            SEQUENCE (SIZE (1..16)) OF GenericData,
    ...
}

FeatureSet ::= SEQUENCE
{
    replacementFeatureSet  BOOLEAN,
    neededFeatures         SEQUENCE OF FeatureDescriptor OPTIONAL,
    desiredFeatures        SEQUENCE OF FeatureDescriptor OPTIONAL,
    supportedFeatures      SEQUENCE OF FeatureDescriptor OPTIONAL,
    ...
}

TransportChannelInfo ::= SEQUENCE
{
    sendAddress           TransportAddress OPTIONAL,
    recvAddress           TransportAddress OPTIONAL,
    ...
}

RTPSession ::= SEQUENCE
{
    rtpAddress            TransportChannelInfo,
    rtcpAddress           TransportChannelInfo,
    cname                 PrintableString,
    ssrc                  INTEGER (1..4294967295),
    sessionId             INTEGER (1..255),
    associatedSessionIds  SEQUENCE OF INTEGER (1..255),
    ...
    multicast             NULL OPTIONAL,
    bandwidth             BandWidth OPTIONAL
}

```

```

RehomingModel ::= CHOICE
{
    gatekeeperBased          NULL,
    endpointBased           NULL
}

RasMessage ::= CHOICE
{
    gatekeeperRequest        GatekeeperRequest,
    gatekeeperConfirm        GatekeeperConfirm,
    gatekeeperReject         GatekeeperReject,
    registrationRequest      RegistrationRequest,
    registrationConfirm      RegistrationConfirm,
    registrationReject       RegistrationReject,
    unregistrationRequest    UnregistrationRequest,
    unregistrationConfirm    UnregistrationConfirm,
    unregistrationReject     UnregistrationReject,
    admissionRequest         AdmissionRequest,
    admissionConfirm         AdmissionConfirm,
    admissionReject          AdmissionReject,
    bandwidthRequest         BandwidthRequest,
    bandwidthConfirm         BandwidthConfirm,
    bandwidthReject          BandwidthReject,
    disengageRequest         DisengageRequest,
    disengageConfirm         DisengageConfirm,
    disengageReject          DisengageReject,
    locationRequest          LocationRequest,
    locationConfirm          LocationConfirm,
    locationReject           LocationReject,
    infoRequest              InfoRequest,
    infoRequestResponse      InfoRequestResponse,
    nonStandardMessage       NonStandardMessage,
    unknownMessageResponse   UnknownMessageResponse,
    ...,
    requestInProgress        RequestInProgress,
    resourcesAvailableIndicate ResourcesAvailableIndicate,
    resourcesAvailableConfirm ResourcesAvailableConfirm,
    infoRequestAck           InfoRequestAck,
    infoRequestNak           InfoRequestNak,
    serviceControlIndication ServiceControlIndication,
    serviceControlResponse   ServiceControlResponse,
    admissionConfirmSequence SEQUENCE OF AdmissionConfirm
}

GatekeeperRequest ::= SEQUENCE -- (GRQ)
{
    requestSeqNum            RequestSeqNum,
    protocolIdentifier        ProtocolIdentifier,
    nonStandardData          NonStandardParameter OPTIONAL,
    rasAddress                TransportAddress,
    endpointType              EndpointType,
    gatekeeperIdentifier      GatekeeperIdentifier OPTIONAL,
    callServices              QseriesOptions OPTIONAL,
    endpointAlias             SEQUENCE OF AliasAddress OPTIONAL,
    ...,
    alternateEndpoints        SEQUENCE OF Endpoint OPTIONAL,
    tokens                    SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens              SEQUENCE OF CryptoH323Token OPTIONAL,
    authenticationCapability  SEQUENCE OF AuthenticationMechanism OPTIONAL,
    algorithmOIDs             SEQUENCE OF OBJECT IDENTIFIER OPTIONAL,
    integrity                  SEQUENCE OF IntegrityMechanism OPTIONAL,
    integrityCheckValue       ICV OPTIONAL,
    supportsAltGK             NULL OPTIONAL,
    featureSet                FeatureSet OPTIONAL,
    genericData               SEQUENCE OF GenericData OPTIONAL,
    supportsAssignedGK        BOOLEAN,
    assignedGatekeeper        AlternateGK OPTIONAL
}

GatekeeperConfirm ::= SEQUENCE -- (GCF)
{
    requestSeqNum            RequestSeqNum,
    protocolIdentifier        ProtocolIdentifier,
    nonStandardData          NonStandardParameter OPTIONAL,
    gatekeeperIdentifier      GatekeeperIdentifier OPTIONAL,
    rasAddress                TransportAddress,
    ...,
    alternateGatekeeper       SEQUENCE OF AlternateGK OPTIONAL,
    authenticationMode        AuthenticationMechanism OPTIONAL,
    tokens                    SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens              SEQUENCE OF CryptoH323Token OPTIONAL,
    algorithmOID              OBJECT IDENTIFIER OPTIONAL,

```

```

    integrity                SEQUENCE OF IntegrityMechanism OPTIONAL,
    integrityCheckValue      ICV OPTIONAL,
    featureSet               FeatureSet OPTIONAL,
    genericData              SEQUENCE OF GenericData OPTIONAL,
    assignedGatekeeper       AlternateGK OPTIONAL,
    rehommingModel           RehommingModel OPTIONAL
}

GatekeeperReject ::= SEQUENCE -- (GRJ)
{
    requestSeqNum            RequestSeqNum,
    protocolIdentifier       ProtocolIdentifier,
    nonStandardData         NonStandardParameter OPTIONAL,
    gatekeeperIdentifier     GatekeeperIdentifier OPTIONAL,
    rejectReason            GatekeeperRejectReason,
    ...,
    altGKInfo              AltGKInfo OPTIONAL,
    tokens                  SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue      ICV OPTIONAL,
    featureSet              FeatureSet OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL
}

GatekeeperRejectReason ::= CHOICE
{
    resourceUnavailable      NULL,
    terminalExcluded         NULL, -- permission failure, not a resource
                                -- failure
    invalidRevision         NULL,
    undefinedReason         NULL,
    ...,
    securityDenial          NULL,
    genericDataReason       NULL,
    neededFeatureNotSupported NULL,
    securityError           SecurityErrors
}

RegistrationRequest ::= SEQUENCE -- (RRQ)
{
    requestSeqNum            RequestSeqNum,
    protocolIdentifier       ProtocolIdentifier,
    nonStandardData         NonStandardParameter OPTIONAL,
    discoveryComplete        BOOLEAN,
    callSignalAddress        SEQUENCE OF TransportAddress,
    rasAddress               SEQUENCE OF TransportAddress,
    terminalType             EndpointType,
    terminalAlias            SEQUENCE OF AliasAddress OPTIONAL,
    gatekeeperIdentifier     GatekeeperIdentifier OPTIONAL,
    endpointVendor          VendorIdentifier,
    ...,
    alternateEndpoints       SEQUENCE OF Endpoint OPTIONAL,
    timeToLive              TimeToLive OPTIONAL,
    tokens                  SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue      ICV OPTIONAL,
    keepAlive               BOOLEAN,
    endpointIdentifier       EndpointIdentifier OPTIONAL,
    willSupplyUUIEs         BOOLEAN,
    maintainConnection       BOOLEAN,
    alternateTransportAddresses OPTIONAL,
    additiveRegistration     NULL OPTIONAL,
    terminalAliasPattern     SEQUENCE OF AddressPattern OPTIONAL,
    supportsAltGK           NULL OPTIONAL,
    usageReportingCapability RasUsageInfoTypes OPTIONAL,
    multipleCalls           BOOLEAN OPTIONAL,
    supportedH248Packages    SEQUENCE OF H248PackagesDescriptor OPTIONAL,
    callCreditCapability     CallCreditCapability OPTIONAL,
    capacityReportingCapability OPTIONAL,
    capacity                 CallCapacity OPTIONAL,
    featureSet              FeatureSet OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL,
    restart                 NULL OPTIONAL,
    supportsACFSequences     NULL OPTIONAL,
    supportsAssignedGK       BOOLEAN,
    assignedGatekeeper       AlternateGK OPTIONAL,
    transportQOS            TransportQOS OPTIONAL,
    language                SEQUENCE OF IA5String(SIZE (1..32)) OPTIONAL
}

RegistrationConfirm ::= SEQUENCE -- (RCF)
{
    requestSeqNum            RequestSeqNum,
    protocolIdentifier       ProtocolIdentifier,

```



```

nonStandardData                NonStandardParameter OPTIONAL,
callSignalAddress              SEQUENCE OF TransportAddress,
terminalAlias                  SEQUENCE OF AliasAddress OPTIONAL,
gatekeeperIdentifier           GatekeeperIdentifier OPTIONAL,
endpointIdentifier             EndpointIdentifier,
...
alternateGatekeeper            SEQUENCE OF AlternateGK OPTIONAL,
timeToLive                     TimeToLive OPTIONAL,
tokens                         SEQUENCE OF ClearToken OPTIONAL,
cryptoTokens                   SEQUENCE OF CryptoH323Token OPTIONAL,
integrityCheckValue           ICV OPTIONAL,
willRespondToIRR              BOOLEAN,
preGrantedARQ                  SEQUENCE
{
    makeCall                    BOOLEAN,
    useGKCallSignalAddressToMakeCall BOOLEAN,
    answerCall                   BOOLEAN,
    useGKCallSignalAddressToAnswer BOOLEAN,
    ...
    irrFrequencyInCall          INTEGER (1..65535) OPTIONAL, -- in seconds;
                                                                -- not present
                                                                -- if GK does
                                                                -- not want IRRs
    totalBandwidthRestriction   BandWidth OPTIONAL,
                                                                -- total limit
                                                                -- for all
                                                                -- concurrent
                                                                -- calls
    alternateTransportAddresses AlternateTransportAddresses OPTIONAL,
    useSpecifiedTransport        UseSpecifiedTransport OPTIONAL
} OPTIONAL,
maintainConnection             BOOLEAN,
serviceControl                 SEQUENCE OF ServiceControlSession OPTIONAL,
supportsAdditiveRegistration   NULL OPTIONAL,
terminalAliasPattern           SEQUENCE OF AddressPattern OPTIONAL,
supportedPrefixes              SEQUENCE OF SupportedPrefix OPTIONAL,
usageSpec                      SEQUENCE OF RasUsageSpecification OPTIONAL,
featureServerAlias             AliasAddress OPTIONAL,
capacityReportingSpec          CapacityReportingSpecification OPTIONAL,
featureSet                     FeatureSet OPTIONAL,
genericData                    SEQUENCE OF GenericData OPTIONAL,
assignedGatekeeper             AlternateGK OPTIONAL,
rehomeingModel                 RehomeingModel OPTIONAL,
transportQOS                   TransportQOS OPTIONAL
}

RegistrationReject ::= SEQUENCE -- (RRJ)
{
    requestSeqNum               RequestSeqNum,
    protocolIdentifier           ProtocolIdentifier,
    nonStandardData              NonStandardParameter OPTIONAL,
    rejectReason                 RegistrationRejectReason,
    gatekeeperIdentifier         GatekeeperIdentifier OPTIONAL,
    ...
    altGKInfo                   AltGKInfo OPTIONAL,
    tokens                       SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens                 SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue          ICV OPTIONAL,
    featureSet                   FeatureSet OPTIONAL,
    genericData                  SEQUENCE OF GenericData OPTIONAL,
    assignedGatekeeper           AlternateGK OPTIONAL
}

RegistrationRejectReason ::= CHOICE
{
    discoveryRequired            NULL,
    invalidRevision              NULL,
    invalidCallSignalAddress     NULL,
    invalidRASAddress            NULL, -- supplied address is invalid
    duplicateAlias               SEQUENCE OF AliasAddress,
                                                                -- alias registered to another
                                                                -- endpoint
    invalidTerminalType          NULL,
    undefinedReason              NULL,
    transportNotSupported        NULL, -- one or more of the transports
    ...
    transportQOSNotSupported     NULL, -- endpoint QOS not supported
    resourceUnavailable          NULL, -- gatekeeper resources exhausted
    invalidAlias                 NULL, -- alias not consistent with
                                                                -- gatekeeper rules
    securityDenial               NULL,
    fullRegistrationRequired     NULL, -- registration permission has
                                                                -- expired
    additiveRegistrationNotSupported NULL,

```

```

invalidTerminalAliases      SEQUENCE
{
    terminalAlias            SEQUENCE OF AliasAddress OPTIONAL,
    terminalAliasPattern     SEQUENCE OF AddressPattern OPTIONAL,
    supportedPrefixes       SEQUENCE OF SupportedPrefix OPTIONAL,
    ...
},
genericDataReason          NULL,
neededFeatureNotSupported  NULL,
securityError              SecurityErrors,
registerWithAssignedGK     NULL
}

UnregistrationRequest ::= SEQUENCE -- (URQ)
{
    requestSeqNum           RequestSeqNum,
    callSignalAddress       SEQUENCE OF TransportAddress,
    endpointAlias           SEQUENCE OF AliasAddress OPTIONAL,
    nonStandardData         NonStandardParameter OPTIONAL,
    endpointIdentifier      EndpointIdentifier OPTIONAL,
    ...
    alternateEndpoints      SEQUENCE OF Endpoint OPTIONAL,
    gatekeeperIdentifier    GatekeeperIdentifier OPTIONAL,
    tokens                  SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue     ICV OPTIONAL,
    reason                  UnregRequestReason OPTIONAL,
    endpointAliasPattern    SEQUENCE OF AddressPattern OPTIONAL,
    supportedPrefixes       SEQUENCE OF SupportedPrefix OPTIONAL,
    alternateGatekeeper     SEQUENCE OF AlternateGK OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL,
    assignedGatekeeper      AlternateGK OPTIONAL
}

UnregRequestReason ::= CHOICE
{
    reregistrationRequired  NULL,
    ttlExpired              NULL,
    securityDenial          NULL,
    undefinedReason        NULL,
    ...
    maintenance            NULL,
    securityError           SecurityErrors2,
    registerWithAssignedGK  NULL
}

UnregistrationConfirm ::= SEQUENCE -- (UCF)
{
    requestSeqNum           RequestSeqNum,
    nonStandardData         NonStandardParameter OPTIONAL,
    ...
    tokens                  SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue     ICV OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL,
    assignedGatekeeper      AlternateGK OPTIONAL
}

UnregistrationReject ::= SEQUENCE -- (URJ)
{
    requestSeqNum           RequestSeqNum,
    rejectReason            UnregRejectReason,
    nonStandardData         NonStandardParameter OPTIONAL,
    ...
    altGKInfo              AltGKInfo OPTIONAL,
    tokens                  SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue     ICV OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL
}

UnregRejectReason ::= CHOICE
{
    notCurrentlyRegistered  NULL,
    callInProgress          NULL,
    undefinedReason        NULL,
    ...
    permissionDenied       NULL, -- requesting user not allowed to
                                -- unregister specified user
    securityDenial         NULL,
    securityError           SecurityErrors2
}

```

```

AdmissionRequest ::= SEQUENCE -- (ARQ)
{
    requestSeqNum          RequestSeqNum,
    callType               CallType,
    callModel              CallModel OPTIONAL,
    endpointIdentifier     EndpointIdentifier,
    destinationInfo       SEQUENCE OF AliasAddress OPTIONAL,
    destCallSignalAddress TransportAddress OPTIONAL,
    destExtraCallInfo     SEQUENCE OF AliasAddress OPTIONAL,
    srcInfo                SEQUENCE OF AliasAddress,
    srcCallSignalAddress  TransportAddress OPTIONAL,
    bandwidth              BandWidth,
    callReferenceValue    CallReferenceValue,
    nonStandardData       NonStandardParameter OPTIONAL,
    callServices           QseriesOptions OPTIONAL,
    conferenceID          ConferenceIdentifier,
    activeMC              BOOLEAN,
    answerCall            BOOLEAN,          -- answering a call
    ...,
    canMapAlias           BOOLEAN,        -- can handle alias address
    callIdentifier        CallIdentifier,
    srcAlternatives       SEQUENCE OF Endpoint OPTIONAL,
    destAlternatives      SEQUENCE OF Endpoint OPTIONAL,
    gatekeeperIdentifier  GatekeeperIdentifier OPTIONAL,
    tokens                SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens          SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue   ICV OPTIONAL,
    transportQOS          TransportQOS OPTIONAL,
    willSupplyUUIEs      BOOLEAN,
    callLinkage           CallLinkage OPTIONAL,
    gatewayDataRate      DataRate OPTIONAL,
    capacity              CallCapacity OPTIONAL,
    circuitInfo           CircuitInfo OPTIONAL,
    desiredProtocols     SEQUENCE OF SupportedProtocols OPTIONAL,
    desiredTunnelledProtocol TunnelledProtocol OPTIONAL,
    featureSet            FeatureSet OPTIONAL,
    genericData           SEQUENCE OF GenericData OPTIONAL,
    canMapSrcAlias       BOOLEAN
}

CallType ::= CHOICE
{
    pointToPoint          NULL,          -- Point-to-point
    oneToN                NULL,          -- no interaction (FFS)
    nToOne               NULL,          -- no interaction (FFS)
    nToN                 NULL,          -- interactive (multipoint)
    ...
}

CallModel ::= CHOICE
{
    direct                NULL,
    gatekeeperRouted     NULL,
    ...
}

TransportQOS ::= CHOICE
{
    endpointControlled   NULL,
    gatekeeperControlled NULL,
    noControl            NULL,
    ...,
    qosCapabilities     SEQUENCE SIZE(1..256) OF QOSCapability
}

AdmissionConfirm ::= SEQUENCE -- (ACF)
{
    requestSeqNum        RequestSeqNum,
    bandwidth            BandWidth,
    callModel            CallModel,
    destCallSignalAddress TransportAddress,
    irrFrequency         INTEGER (1..65535) OPTIONAL,
    nonStandardData      NonStandardParameter OPTIONAL,
    ...,
    destinationInfo     SEQUENCE OF AliasAddress OPTIONAL,
    destExtraCallInfo   SEQUENCE OF AliasAddress OPTIONAL,
    destinationType     EndpointType OPTIONAL,
    remoteExtensionAddress SEQUENCE OF AliasAddress OPTIONAL,
    alternateEndpoints  SEQUENCE OF Endpoint OPTIONAL,
    tokens              SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens        SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue ICV OPTIONAL,
    transportQOS        TransportQOS OPTIONAL,
    willRespondToIRR    BOOLEAN,
}

```

```

    uuiEsRequested          UUIEsRequested,
    language                SEQUENCE OF IA5String (SIZE (1..32)) OPTIONAL,
    alternateTransportAddresses AlternateTransportAddresses OPTIONAL,
    useSpecifiedTransport    UseSpecifiedTransport OPTIONAL,
    circuitInfo              CircuitInfo OPTIONAL,
    usageSpec                SEQUENCE OF RasUsageSpecification OPTIONAL,
    supportedProtocols        SEQUENCE OF SupportedProtocols OPTIONAL,
    serviceControl            SEQUENCE OF ServiceControlSession OPTIONAL,
    multipleCalls             BOOLEAN OPTIONAL,
    featureSet                FeatureSet OPTIONAL,
    genericData                SEQUENCE OF GenericData OPTIONAL,
    modifiedSrcInfo           SEQUENCE OF AliasAddress OPTIONAL,
    assignedGatekeeper        AlternateGK OPTIONAL
}

UUIEsRequested ::= SEQUENCE
{
    setup                    BOOLEAN,
    callProceeding           BOOLEAN,
    connect                  BOOLEAN,
    alerting                 BOOLEAN,
    information              BOOLEAN,
    releaseComplete         BOOLEAN,
    facility                 BOOLEAN,
    progress                 BOOLEAN,
    empty                    BOOLEAN,
    ...,
    status                   BOOLEAN,
    statusInquiry           BOOLEAN,
    setupAcknowledge         BOOLEAN,
    notify                   BOOLEAN
}

AdmissionReject ::= SEQUENCE --(ARJ)
{
    requestSeqNum            RequestSeqNum,
    rejectReason             AdmissionRejectReason,
    nonStandardData          NonStandardParameter OPTIONAL,
    ...,
    altGKInfo                AltGKInfo OPTIONAL,
    tokens                   SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens             SEQUENCE OF CryptoH323Token OPTIONAL,
    callSignalAddress         SEQUENCE OF TransportAddress OPTIONAL,
    integrityCheckValue      ICV OPTIONAL,
    serviceControl            SEQUENCE OF ServiceControlSession OPTIONAL,
    featureSet                FeatureSet OPTIONAL,
    genericData                SEQUENCE OF GenericData OPTIONAL,
    assignedGatekeeper        AlternateGK OPTIONAL
}

AdmissionRejectReason ::= CHOICE
{
    calledPartyNotRegistered NULL,    -- cannot translate address
    invalidPermission        NULL,    -- permission has expired
    requestDenied            NULL,
    undefinedReason          NULL,
    callerNotRegistered      NULL,
    routeCallToGatekeeper    NULL,
    invalidEndpointIdentifier NULL,
    resourceUnavailable       NULL,
    ...,
    securityDenial           NULL,
    qosControlNotSupported   NULL,
    incompleteAddress         NULL,
    aliasesInconsistent      NULL,    -- multiple aliases in request
                                         -- identify distinct people
    routeCallToSCN           SEQUENCE OF PartyNumber,
    exceedsCallCapacity       NULL,    -- destination does not have the
                                         -- capacity for this call
    collectDestination        NULL,
    collectPIN                NULL,
    genericDataReason         NULL,
    neededFeatureNotSupported NULL,
    securityErrors            SecurityErrors2,
    securityDHmismatch        NULL,    -- mismatch of DH parameters
    noRouteToDestination      NULL,    -- destination unreachable
    unallocatedNumber         NULL,    -- detination number unassigned
    registerWithAssignedGK    NULL
}

BandwidthRequest ::= SEQUENCE --(BRQ)
{
    requestSeqNum            RequestSeqNum,

```

```

    endpointIdentifier      EndpointIdentifier,
    conferenceID           ConferenceIdentifier,
    callReferenceValue     CallReferenceValue,
    callType               CallType OPTIONAL,
    bandwidth              BandWidth,
    nonStandardData        NonStandardParameter OPTIONAL,
    ...,
    callIdentifier         CallIdentifier,
    gatekeeperIdentifier   GatekeeperIdentifier OPTIONAL,
    tokens                 SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue    ICV OPTIONAL,
    answeredCall           BOOLEAN,
    callLinkage            CallLinkage OPTIONAL,
    capacity               CallCapacity OPTIONAL,
    usageInformation       RasUsageInformation OPTIONAL,
    bandwidthDetails       SEQUENCE OF BandwidthDetails OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL,
    transportQOS           TransportQOS OPTIONAL
}

BandwidthConfirm ::= SEQUENCE -- (BCF)
{
    requestSeqNum          RequestSeqNum,
    bandwidth              BandWidth,
    nonStandardData        NonStandardParameter OPTIONAL,
    ...,
    tokens                 SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue    ICV OPTIONAL,
    capacity               CallCapacity OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL,
    transportQOS           TransportQOS OPTIONAL
}

BandwidthReject ::= SEQUENCE -- (BRJ)
{
    requestSeqNum          RequestSeqNum,
    rejectReason           BandRejectReason,
    allowedBandWidth       BandWidth,
    nonStandardData        NonStandardParameter OPTIONAL,
    ...,
    altGKInfo              AltGKInfo OPTIONAL,
    tokens                 SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue    ICV OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL
}

BandRejectReason ::= CHOICE
{
    notBound                NULL,      -- discovery permission has aged
    invalidConferenceID     NULL,      -- possible revision
    invalidPermission       NULL,      -- true permission violation
    insufficientResources    NULL,
    invalidRevision         NULL,
    undefinedReason         NULL,
    ...,
    securityDenial          NULL,
    securityError           SecurityErrors2
}

LocationRequest ::= SEQUENCE -- (LRQ)
{
    requestSeqNum          RequestSeqNum,
    endpointIdentifier      EndpointIdentifier OPTIONAL,
    destinationInfo        SEQUENCE OF AliasAddress,
    nonStandardData        NonStandardParameter OPTIONAL,
    replyAddress           TransportAddress,
    ...,
    sourceInfo              SEQUENCE OF AliasAddress OPTIONAL,
    canMapAlias             BOOLEAN,      -- can handle alias address
    gatekeeperIdentifier   GatekeeperIdentifier OPTIONAL,
    tokens                 SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue    ICV OPTIONAL,
    desiredProtocols        SEQUENCE OF SupportedProtocols OPTIONAL,
    desiredTunnelledProtocol TunnelledProtocol OPTIONAL,
    featureSet              FeatureSet OPTIONAL,
    genericData             SEQUENCE OF GenericData OPTIONAL,
    hopCount                INTEGER (1..255) OPTIONAL,
    circuitInfo             CircuitInfo OPTIONAL,
    callIdentifier         CallIdentifier OPTIONAL,
    bandwidth              BandWidth OPTIONAL,
}

```

```

        sourceEndpointInfo      SEQUENCE OF AliasAddress OPTIONAL,
        canMapSrcAlias          BOOLEAN,
        language                 SEQUENCE OF IA5String(SIZE (1..32)) OPTIONAL
    }

LocationConfirm ::= SEQUENCE --(LCF)
{
    requestSeqNum               RequestSeqNum,
    callSignalAddress           TransportAddress,
    trasAddress                 TransportAddress,
    nonStandardData            NonStandardParameter OPTIONAL,
    ...,
    destinationInfo            SEQUENCE OF AliasAddress OPTIONAL,
    destExtraCallInfo          SEQUENCE OF AliasAddress OPTIONAL,
    destinationType            EndpointType OPTIONAL,
    remoteExtensionAddress      SEQUENCE OF AliasAddress OPTIONAL,
    alternateEndpoints          SEQUENCE OF Endpoint OPTIONAL,
    tokens                      SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens               SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue        ICV OPTIONAL,
    alternateTransportAddresses AlternateTransportAddresses OPTIONAL,
    supportedProtocols          SEQUENCE OF SupportedProtocols OPTIONAL,
    multipleCalls              BOOLEAN OPTIONAL,
    featureSet                  FeatureSet OPTIONAL,
    genericData                 SEQUENCE OF GenericData OPTIONAL,
    circuitInfo                CircuitInfo OPTIONAL,
    serviceControl              SEQUENCE OF ServiceControlSession OPTIONAL,
    modifiedSrcInfo            SEQUENCE OF AliasAddress OPTIONAL,
    bandWidth                   BandWidth OPTIONAL
}

LocationReject ::= SEQUENCE --(LRJ)
{
    requestSeqNum               RequestSeqNum,
    rejectReason                LocationRejectReason,
    nonStandardData            NonStandardParameter OPTIONAL,
    ...,
    altGKInfo                   AltGKInfo OPTIONAL,
    tokens                      SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens               SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue        ICV OPTIONAL,
    featureSet                  FeatureSet OPTIONAL,
    genericData                 SEQUENCE OF GenericData OPTIONAL,
    serviceControl              SEQUENCE OF ServiceControlSession OPTIONAL
}

LocationRejectReason ::= CHOICE
{
    notRegistered              NULL,
    invalidPermission          NULL, -- exclusion by administrator or feature
    requestDenied              NULL,
    undefinedReason            NULL,
    ...,
    securityDenial             NULL,
    aliasesInconsistent        NULL, -- multiple aliases in request
                                -- identify distinct people
    routeCalltoSCN            SEQUENCE OF PartyNumber,
    resourceUnavailable        NULL,
    genericDataReason          NULL,
    neededFeatureNotSupported  NULL,
    hopCountExceeded          NULL,
    incompleteAddress          NULL,
    securityError              SecurityErrors2,
    securityDHmismatch         NULL, -- mismatch of DH parameters
    noRouteToDestination       NULL, -- destination unreachable
    unallocatedNumber          NULL -- detination number unassigned
}

DisengageRequest ::= SEQUENCE --(DRQ)
{
    requestSeqNum               RequestSeqNum,
    endpointIdentifier          EndpointIdentifier,
    conferenceID                ConferenceIdentifier,
    callReferenceValue          CallReferenceValue,
    disengageReason            DisengageReason,
    nonStandardData            NonStandardParameter OPTIONAL,
    ...,
    callIdentifier              CallIdentifier,
    gatekeeperIdentifier        GatekeeperIdentifier OPTIONAL,
    tokens                      SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens               SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue        ICV OPTIONAL,
    answeredCall                BOOLEAN,
    callLinkage                 CallLinkage OPTIONAL,
}

```

```

        capacity                CallCapacity OPTIONAL,
        circuitInfo             CircuitInfo OPTIONAL,
        usageInformation        RasUsageInformation OPTIONAL,
        terminationCause        CallTerminationCause OPTIONAL,
        serviceControl          SEQUENCE OF ServiceControlSession OPTIONAL,
        genericData             SEQUENCE OF GenericData OPTIONAL
    }

DisengageReason ::= CHOICE
{
    forcedDrop                NULL,    -- gatekeeper is forcing the drop
    normalDrop                NULL,    -- associated with normal drop
    undefinedReason          NULL,
    ...
}

DisengageConfirm ::= SEQUENCE -- (DCF)
{
    requestSeqNum            RequestSeqNum,
    nonStandardData          NonStandardParameter OPTIONAL,
    ...,
    tokens                   SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens             SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue      ICV OPTIONAL,
    capacity                 CallCapacity OPTIONAL,
    circuitInfo              CircuitInfo OPTIONAL,
    usageInformation         RasUsageInformation OPTIONAL,
    genericData              SEQUENCE OF GenericData OPTIONAL,
    assignedGatekeeper       AlternateGK OPTIONAL
}

DisengageReject ::= SEQUENCE -- (DRJ)
{
    requestSeqNum            RequestSeqNum,
    rejectReason             DisengageRejectReason,
    nonStandardData          NonStandardParameter OPTIONAL,
    ...,
    altGKInfo                AltGKInfo OPTIONAL,
    tokens                   SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens             SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue      ICV OPTIONAL,
    genericData              SEQUENCE OF GenericData OPTIONAL
}

DisengageRejectReason ::= CHOICE
{
    notRegistered            NULL,    -- not registered with gatekeeper
    requestToDropOther       NULL,    -- cannot request drop for others
    ...,
    securityDenial           NULL,
    securityError            SecurityErrors2
}

InfoRequest ::= SEQUENCE -- (IRQ)
{
    requestSeqNum            RequestSeqNum,
    callReferenceValue       CallReferenceValue,
    nonStandardData          NonStandardParameter OPTIONAL,
    replyAddress             TransportAddress OPTIONAL,
    ...,
    callIdentifier           CallIdentifier,
    tokens                   SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens             SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue      ICV OPTIONAL,
    uuiesRequested          UUIEsRequested OPTIONAL,
    callLinkage              CallLinkage OPTIONAL,
    usageInfoRequested       RasUsageInfoTypes OPTIONAL,
    segmentedResponseSupported NULL OPTIONAL,
    nextSegmentRequested     INTEGER (0..65535) OPTIONAL,
    capacityInfoRequested    NULL OPTIONAL,
    genericData              SEQUENCE OF GenericData OPTIONAL,
    assignedGatekeeper       AlternateGK OPTIONAL
}

InfoRequestResponse ::= SEQUENCE -- (IRR)
{
    nonStandardData          NonStandardParameter OPTIONAL,
    requestSeqNum            RequestSeqNum,
    endpointType             EndpointType,
    endpointIdentifier        EndpointIdentifier,
    rasAddress               TransportAddress,
    callSignalAddress        SEQUENCE OF TransportAddress,
    endpointAlias            SEQUENCE OF AliasAddress OPTIONAL,
}

```

```

perCallInfo          SEQUENCE OF SEQUENCE
{
    nonStandardData      NonStandardParameter OPTIONAL,
    callReferenceValue   CallReferenceValue,
    conferenceID         ConferenceIdentifier,
    originator           BOOLEAN OPTIONAL,
    audio                SEQUENCE OF RTPSession OPTIONAL,
    video                SEQUENCE OF RTPSession OPTIONAL,
    data                 SEQUENCE OF TransportChannelInfo OPTIONAL,
    h245                 TransportChannelInfo,
    callSignaling        TransportChannelInfo,
    callType             CallType,
    bandwidth            BandWidth,
    callModel            CallModel,
    ...,
    callIdentifier       CallIdentifier,
    tokens               SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens         SEQUENCE OF CryptoH323Token OPTIONAL,
    substituteConfIDs    SEQUENCE OF ConferenceIdentifier,
    pdu                  SEQUENCE OF SEQUENCE
    {
        h323pdu          H323-UU-PDU,
        sent              BOOLEAN -- TRUE is sent, FALSE is received
    } OPTIONAL,
    callLinkage          CallLinkage OPTIONAL,
    usageInformation     RasUsageInformation OPTIONAL,
    circuitInfo          CircuitInfo OPTIONAL
} OPTIONAL,
...,
tokens                 SEQUENCE OF ClearToken OPTIONAL,
cryptoTokens           SEQUENCE OF CryptoH323Token OPTIONAL,
integrityCheckValue    ICV OPTIONAL,
needResponse           BOOLEAN,
capacity               CallCapacity OPTIONAL,
irrStatus              InfoRequestResponseStatus OPTIONAL,
unsolicited            BOOLEAN,
genericData            SEQUENCE OF GenericData OPTIONAL
}

InfoRequestResponseStatus ::= CHOICE
{
    complete             NULL,
    incomplete           NULL,
    segment              INTEGER (0..65535),
    invalidCall          NULL,
    ...
}

InfoRequestAck ::= SEQUENCE -- (IACK)
{
    requestSeqNum       RequestSeqNum,
    nonStandardData     NonStandardParameter OPTIONAL,
    tokens               SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens         SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue ICV OPTIONAL,
    ...
}

InfoRequestNak ::= SEQUENCE -- (INAK)
{
    requestSeqNum       RequestSeqNum,
    nonStandardData     NonStandardParameter OPTIONAL,
    nakReason           InfoRequestNakReason,
    altGKInfo           AltGKInfo OPTIONAL,
    tokens               SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens         SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue ICV OPTIONAL,
    ...
}

InfoRequestNakReason ::= CHOICE
{
    notRegistered       NULL, -- not registered with gatekeeper
    securityDenial      NULL,
    undefinedReason     NULL,
    ...,
    securityError       SecurityErrors2
}

NonStandardMessage ::= SEQUENCE
{
    requestSeqNum       RequestSeqNum,
    nonStandardData     NonStandardParameter,

```



```

    ...,
    tokens                SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens          SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue   ICV OPTIONAL,
    featureSet            FeatureSet OPTIONAL,
    genericData           SEQUENCE OF GenericData OPTIONAL
}

UnknownMessageResponse ::= SEQUENCE -- (XRS)
{
    requestSeqNum         RequestSeqNum,
    ...,
    tokens                SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens          SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue   ICV OPTIONAL,
    messageNotUnderstood OCTET STRING
}

RequestInProgress ::= SEQUENCE -- (RIP)
{
    requestSeqNum         RequestSeqNum,
    nonStandardData       NonStandardParameter OPTIONAL,
    tokens                SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens          SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue   ICV OPTIONAL,
    delay                 INTEGER (1..65535),
    ...
}

ResourcesAvailableIndicate ::= SEQUENCE -- (RAI)
{
    requestSeqNum         RequestSeqNum,
    protocolIdentifier    ProtocolIdentifier,
    nonStandardData       NonStandardParameter OPTIONAL,
    endpointIdentifier    EndpointIdentifier,
    protocols             SEQUENCE OF SupportedProtocols,
    almostOutOfResources BOOLEAN,
    tokens                SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens          SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue   ICV OPTIONAL,
    ...,
    capacity              CallCapacity OPTIONAL,
    genericData           SEQUENCE OF GenericData OPTIONAL
}

ResourcesAvailableConfirm ::= SEQUENCE -- (RAC)
{
    requestSeqNum         RequestSeqNum,
    protocolIdentifier    ProtocolIdentifier,
    nonStandardData       NonStandardParameter OPTIONAL,
    tokens                SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens          SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue   ICV OPTIONAL,
    ...,
    genericData           SEQUENCE OF GenericData OPTIONAL
}

ServiceControlIndication ::= SEQUENCE -- (SCI)
{
    requestSeqNum         RequestSeqNum,
    nonStandardData       NonStandardParameter OPTIONAL,
    serviceControl        SEQUENCE OF ServiceControlSession,
    endpointIdentifier    EndpointIdentifier OPTIONAL,
    callSpecific          SEQUENCE
    {
        callIdentifier    CallIdentifier,
        conferenceID       ConferenceIdentifier,
        answeredCall       BOOLEAN,
        ...
    } OPTIONAL,
    tokens                SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens          SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue   ICV OPTIONAL,
    featureSet            FeatureSet OPTIONAL,
    genericData           SEQUENCE OF GenericData OPTIONAL,
    ...
}

ServiceControlResponse ::= SEQUENCE -- (SCR)
{
    requestSeqNum         RequestSeqNum,
    result                CHOICE
    {
        started           NULL,

```

```

        failed                NULL,
        stopped               NULL,
        notAvailable          NULL,
        neededFeatureNotSupported  NULL,
        ...
    } OPTIONAL,
    nonStandardData          NonStandardParameter OPTIONAL,
    tokens                   SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens             SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue      ICV OPTIONAL,
    featureSet               FeatureSet OPTIONAL,
    genericData              SEQUENCE OF GenericData OPTIONAL,
    ...
}

END          -- of ASN.1

```

## 付属資料 I : JT-H263 第 2 版ビデオのパケット化

IETF RFC2429 は、JT-H263 第 2 版(1998 年) (PLUSPTYPE を使用する機能と JT-H263 の付属資料 I~T を含む)で採用された新しい機能を含む JT-H263 ビットストリームのための RTP ペイロードフォーマットを規定している。

この標準の付属資料 E に規定された RFC2190 の JT-H263 ペイロードフォーマットをサポートするための能力は、以前の実装と相互接続性を保つため、JT-H263 第 2 版の新しい特徴を使用しない JT-H263 ビットストリームのために必要である。しかしながら、RFC2429 に規定された新しいペイロードフォーマットは、受信端末の能力の範囲内であるという条件で、新しい第 2 版の機能を含まないビットストリームであっても使われるべきである。

## 付録 I : RTP/RTCPアルゴリズム

有益な参考資料は、次に示す提案されたインターネット標準の中に見い出すことができる。

SCHULZRINNE (H.), CASNER (S.), FREDERICK (R.) and JACOBSON (V.): RFC 3550, RTP: A Transport Protocol for Real-Time Applications, *Internet Engineering Task Force*, 2003.

## 付録Ⅱ： RTPプロファイル

有益な参考資料は、次に示す提案されたインターネット標準の中に見い出すことができる。

SCHULZRINNE (H.), CASNER (S.): RFC 3551, RTP Profile for Audio and Video Conferences with Minimal Control, *Internet Engineering Task Force*, 2003.

### 付録Ⅲ： JT-H261 パケット化

有益な参考資料は、次に示す提案されたインターネット標準の中に見い出すことができる。

TURLETTI (T.), HUITEMA (C.): RFC 2032, RTP Payload Format for H.261 Video Streams, *Internet Engineering Task Force*, 1996.

## 付録Ⅳ： 異なるパケットベースネットワークプロトコルスタックにおけるJT-H225.0の動作

本付録は、様々な実際のパケットベースのネットワークプロトコルスタックによる、本標準の動作に関連している付加記述である。この付録は、非標準である。JT-H225.0 で用いられるパケットベースのネットワークは、パケット境界を区別するための手段を含む、高信頼性動作モードと低信頼性動作モードの両方を与えなければならない。

### Ⅳ.1 TCP/IP, UDP/IP

注意：UDP は、大きなビデオパケットのフラグメントおよびリアセンブルができるが、MB パケット化を失敗することは全体の GOB のロスに至るかもしれない。

IP マルチキャストでは、メディアアクセスレイヤブロードキャストではなく、GRQ 配信が用いられるべきである。

低信頼性 アプリケーション配信	呼シグナリングと JT-H245 チャンネル
UDP	TPKT ----- TCP
IP	
リンクレイヤ	
物理レイヤ	

TPKT は IETF RFC1006 で定義されたパケットフォーマットである。これは、明確な境界のない連続したオクテットストリームを提供する TCP ストリームにおいて、独立したメッセージ (PDU) の境界を明確にするために用いられる。TPKT パケットは 1 オクテットのバージョン番号フィールド、続いて 1 オクテットの予約フィールド、続いて 2 オクテットのパケット長フィールド、続いて実際のデータから成る。バージョン番号フィールドの値は「3」でなければならない。予約フィールドの値は「0」でなければならない。パケット長フィールドは、番号フィールド、予約フィールド、パケット長フィールドを含むパケット全体の長さを 16 ビットワードのビッグエンディアンで表現された値でなければならない。

#### Ⅳ.1.1 ゲートキーパーの発見

##### Ⅳ.1.1.1 マルチキャストアドレスあるいは公知ポートによる発見

ゲートキーパーの発見や登録方法は、JT-H323 の 7 章で次のように述べられている。エンドポイントは、これらのネットワーク関連位置を割り当てるようにゲートキーパーの発見を試みる場合、以下のようにマルチキャストアドレスあるいは公知ポートを使用すべきである。

ゲートキーパーとのマルチキャスト通信のための UDP アドレス            224.0.1.41  
ゲートキーパーとのマルチキャスト通信のための UDP ポート            1718  
ゲートキーパーとの「他の合意」が存在しない時のユニキャスト RAS 通信のための UDP ポート    1719

注：「他の合意」とは、ゲートキーパーとのエンドポイント登録を含んでもよい。

注：インプリメンテーションの際、発見メッセージをインターネット内に氾濫させないようにマルチキャストの範囲を注意すべきである。

例えば、ゲートキーパーの IP アドレスが 134.134.12.1 と仮定して、次のシグナリングが起きるかもしれない

い。

LRQ もしくは GRQ が 134.134.12.1:ポート 1719 に到着

LRQ もしくは GRQ が 134.134.12.1:ポート 1718 に到着 (注意: これは第 1 版ゲートキーパーに対して起きるかもしれない)

LRQ もしくは GRQ が 224.0.1.41:ポート 1718 に到着

ゲートキーパーは LRQ を次のアドレスへ伝送してもよい。

224.0.1.41:ポート 1719 (すべてのゲートキーパーへマルチキャスト)

X.X.X.X:ポート 1719 (特定のゲートキーパーへ)

ポート 1719 はリクエストがユニキャストで送信されたときのみ用いられるべきである。これにより受信側は送信側に拒否 (xRJ) を送信すべきかどうかを知ることができる (全てのケースにおいてそうすべきである)。

ポート 1718 はリクエストがマルチキャストで送信されたときのみ用いられるべきである。受信側はメッセージに基づいた適切な応答を返すべきである。受信側はマルチキャストのリクエストに対しては応答を返さないため、LRQ に関しては、拒絶は要求されない。GRQ に関しては、直接 GRJ が GRQ のソースに送られるべきである。

#### IV.1.1.2 DNSを使用する発見 (情報)

##### IV.1.1.2.1 ゲートキーパーのためのURL

最初のステップとして、ゲートキーパーは、トランスポートアドレスやゲートキーパー識別によって識別されることに注意する。ゲートキーパーはインターネット上で独特のリソースなので、Uniform Resource Locator (URL) にそれを記載することが正当である。ゲートキーパーによって示されるプロトコルは、RAS なので、ゲートキーパーのための URL は次のように与えられている。

ras://gkID@domainname

gkID は、ゲートキーパー識別で、ドメインネームは、ゲートキーパーのドメインを識別する DNS ドメインネームである。これは、Fully Qualified Domain Name (FQDN) が必要ではないことに注意する。このドメインネームが DNS に記録された IP 番号に物理的なトランスポートインタフェースを持つことは必要とされていない。しかしながら、FQDN なら、その IP 番号が URL が問い合わせたゲートキーパーの番号であるというのが道理である。この場合、URL へオプションのポート番号を加えることを許可している。

ras://gkID@domainname:port\_no.

もし、ポート番号が与えられないなら、公知の値である 1719 がデフォルトとして採用される。

より興味深い場合として、FQDN がない場合、ドメインネームは DNS に記録されたトランスポートアドレスによらない。その時、ドメインネームは単なる「ゲートキーパーゾーンの裁定」による。次の章ではこの場合のゲートキーパーの見つけ方を説明する。

##### IV.1.1.2.2 URLの見つけ方

URL はゲートキーパーのロケーションの問題を解決しない。見つけるために情報に関するスタンダードフォーマットを与える。問題は、ゲートキーパーのドメインネームを与えられた RAS 信号にトランスポートアドレスやゲートキーパー識別を生成する方法である。

ゲートキーパーが IETF RFC822 基準の識別を持つなら、ゲートキーパーの RFC822 基準の識別からドメインネームを引き出すことは簡単である。実のところ、エンドポイントに RFC822 基準の識別を与え、ドメインネームの識別部分がゲートキーパードメインを問い合わせることを明記することは便利であるかもしれない。

##### IV.1.1.2.2.1 SRVリソースレコードクエリー

第一の解決策は、ゲートキーパーが基本的なシステムサービスである事実を利用し、名付けられたシステムサービスのトランスポートアドレスが SRV (service location record) と呼ばれる DNS リソースレコードの新しいタイプにクエリーを使用することによって DNS から引き出される。ドメインネームが与えられ、SRV 記録クエリーは、そのドメインに ras サービスのトランスポートアドレスを寄与することになる。ドメインネーム自身あるいは、SRV 応答から戻ってきたドメインネームがゲートキーパー識別として用いられる。SRV レコードとその使用法は IETF RFC2782 にて定義される。



#### IV.1.1.2.2.2 TXTレコードクエリー

現在の DNS 機器はすべて TXT リソースレコードをサポートしている。基本的にどのドメインネームも応答できる自由なテキストである。シングルドメインで多くの TXT リソースを蓄えることが可能である。標準にはクエリーがそれらに作られた場合すべての TXT レコードが応答されるであろうと明記されている。SRV クエリーが失敗したら TXT クエリーを使用することが可能である。前述で示したドメインネームを引き出すために同様な集会を開く。RFC822 にしたがった文字列 (Email 「-like」 names) か IETF RFC1768 にしたがった文字列 (URL) かどちらかがゲートキーパー識別に用いられることができる。どちらの場合も、ドメインネームがドメインネームに DNS TXT クエリーを作らせる。応答されたリソースレコードは自由なテキストのラインで、その時端末は、次の形式の応答を探さるう。

```
ras[<gk id>@]<domain name>[<portno>] [<priority>]
```

<gk id>フィールドはドメインネームから分離したオプションのゲートキーパーID である。この部分を間違えば、ドメイン自身がゲートキーパーID とみなされる。

<domain name>フィールドは、ゲートキーパーの IP アドレスを含む A-レコードの名前か、ドットで形成された生の IP アドレスのどちらかである。ドメインネームはすべての資格を与える必要はない。そうでなければ、TXT レコードが見つけれられたサブドメインがすべての資格を与えられた A-レコードネームを形成するためにそれを付加されるべきである。

オプションである [<portno>] は、標準 RAS ポート以外にポート番号を明記するのに用いられる。オプションである [<priority>] フィールドは、ひとつ以上の ras TXT レコードがある場合に、検出あるいは LRQ のクエリーのアクセスを行うべき、リスト上のゲートキーパーの順序を示す。低い番号がより高い優先度を持つ。

このフォーマットは、<gk id>フィールドが無い場合には、ゲートキーパーID は、事実上、正式なドメイン名であるとみなしていることに注意すること。しかし、単一のホストがそれぞれの独立した ID を持つ複数の論理的なゲートキーパーをサポートすることが必要な場合、このフォーマットは、これをサポートする。なぜなら、独立した A レコードが同一の IP アドレスを持つことが可能であるからである。

ras と、もし存在するなら gk id または domain name、さらに portno と priority のデリミタとして空白が使用される。空白は、いかなる数のスペースまたはタブから成る。

有効なゲートキーパーTXT レコード例を示す。

```
ras      gk1
ras      gk1.company.com
ras      gk1:1500 3
ras      172.11.22.33:1500 2
```

クライアントは返された行を解釈し、それによって、RAS メッセージを送ることができる、そのドメインの範囲において、ゲートキーパーのトランスポートアドレスを得る。

DNS はサーバに対し、ドメイン名から成るすべての TXT を返すことを要求するため、クライアントはそれらを取り除き、自分にとって有効なレコードのみを処理することができる。DNS は、JT-H323 で定義された代理またはバックアップとしてサービスするゲートキーパーの処理されたリストを返すことも許される。

このようなクエリーで返されるサーバはドット付きの 10 進表記の実際のトランスポートアドレスでありうること、または、トランスポートアドレスを決定するために DNS における A レコードのクエリーを必要とする FQDN でありうることに注意すること。FQDN の使用は実際の IP アドレスを隠すために有効である。IP 番号の使用は第 2 の DNS クエリーを防ぐのに有効であり、これにより事前の発信の期間を高速化する。

#### IV.1.1.2.3 ARQとLRQの期間のemail-IDのゲートキーパーの処理

ARQ または LRQ メッセージの destinationInfo フィールドが email-ID の別名アドレスを含む場合、ゲートキーパーは第一に別名に関する登録データベースを照合すべきである。もし解決できなければゲートキーパーは、ドメインの分配を復元するために別名をパズルしなければならない。もし、ドメインが与えられなければ、ゲートキーパーはデフォルトのドメインを生成してもよい。IV.1.1.2.2 節に示された手順を使い、ドメインは、ひとつまたはそれ以上のゲートキーパーを配置するために使われる。このように、ゲートキーパーは LRQ/LCF/LRJ メッセージの交換で見つかるすべてのゲートキーパーへのクエリーを行ってもよい。

一つ以上のゲートキーパーが、単一の DNS ドメインの中に対応する TXT レコードを持ってもよいことに



注：サービスタイプは、IPX のドキュメントでは SAP ソケットと表現されている。

## IV.2.2 エンドポイントとエンドポイントの通信

ゲートキーパーのゾーンの外側のエンドポイントからの呼を受信することを望むエンドポイントは、呼制御のために以下のソケットを使用しなければならない。

Endpoint IPX Call Signaling Port

将来の検討課題

ひとつのデバイスに複数のエンドポイントを許容するために、これらのソケットに対し動的な値を使うことは許されるが、ゾーンの中のゲートウェイを経由することを除けば、ゲートキーパーのゾーンの外のエンドポイントとの相互接続ができなくなることになると理解しなければならない。

## IV.3 SCTP

SCTP 上の JT-H323 のプロトコル・スタックは以下の様になっている。:

低信頼性 アプリケーションの配信	トンネリングされた呼制御 での呼シグナリング
UDP	SCTP
IP	
リンクレイヤ	
物理レイヤ	

それぞれの JT-H225.0 呼制御・メッセージは別々の SCTP DATA ブロックで転送されなければならない。どんなヘッダも加えられるべきではない(すなわち、TPKT がない)。順序だった配信が指定されるべきである。

### IV.3.1 ストリーム

同じ呼に関するすべてのメッセージが同じ SCTP ストリームを使用しなければならない。実装は異なった呼に異なったストリームを使用する。

### IV.3.2 ペイロード・プロトコル識別子

SCTP は未定義のペイロード・プロトコル識別子(0)もしくは(13)で使用される。(13)は IANA によって H.323 に割当てられた番号である)。

## 付録 V : JT-H225.0 における ASN.1 の使用

(この付録は本標準の必須部分ではない)

この付録は JT-H225.0 で使用されている ASN.1 規約を列挙する。JT-H225.0 の将来的な改訂版はこれらの構成要素のみを使用することが望ましい。追加の ASN.1 構成は例外的な状況においてのみ考慮される。

### V.1 タギング

JT-H225.0 内の全てのタグは AUTOMATIC TAGS である。

### V.2 タイプ

下記の型は JT-H225.0 の ASN.1 定義中にある。

BIT STRING	IA5String	OCTET STRING
BMPString	INTEGER	SEQUENCE
BOOLEAN	NULL	SEQUENCE OF
CHOICE	NumericString	SET
GeneralString	OBJECT IDENTIFIER	SET OF

### V.3 制約と範囲

JT-H225.0 は文字列に対するサイズ制約 (「SIZE」)、SET OF と SEQUENCE OF、整数の値域制約、容認されたアルファベット (「FROM」) を利用する。

### V.4 拡張性

JT-H225.0 は拡張マーカ―を利用する。(省略:「...」)

## 付録Ⅵ： JT-H225.0 トンネリングシグナリングプロトコルの識別子

(この付録は本標準の必須部分ではない)

標準 JT-H225.0 は、JT-H323 の 10.4 節において記述される非 JT-H323 呼制御プロトコルのトンネリングをサポートする。標準の JT-H323 シリーズ付属資料 M (JT-H323 付属資料 M.1、H.323 Annex M.2、その他) は、特定のプロトコルに対するトンネリングを定義する。JT-H225.0 のトンネリングプロトコルは、JT-H225.0 の 7.6 節と JT-H225.0 付属資料 H において定義される TunnelledProtocol ASN.1 構造体内の情報によって識別される。本付録は、特定のトンネリングプロトコルに割り当てられた TunnelledProtocol 識別子の一覧を提供する。

本刊行にて定義されるトンネリングプロトコルは、付表 VI-1/JT-H225.0 と付表 VI-2/JT-H225.0 で示される。トンネリングはこれらの表で列挙されるプロトコルに制限されない点に注意すること。

**付表 VI-1/JT-H225.0 tunnelledProtocolObjectID で識別されるトンネリングプロトコル  
(ITU-T H.225.0)**

トンネリング仕様	プロトコル仕様	tunnelledProtocolObjectID	subIdentifier
JT-H.323 付属資料 M.1	ISO/IEC 11572 及び -11582	{iso (1) identified-organization (3) icd-ecma (0012) private-isdn-signalling-domain (9)}	(None)
H.323 Annex M.2	Q.763 (1988)	{itu-t (0) recommendation (0) q (17) 763}	“1988”
H.323 Annex M.2	Q.763(1992)	{itu-t (0) recommendation (0) q (17) 763}	“1992”

**付表 VI-2/JT-H225.0 TunnelledProtocolAlternateIdentifier で識別されるトンネリングプロトコル  
(ITU-T H.225.0)**

トンネリング仕様	プロトコル仕様	protocolType	protocolVariant	subIdentifier
H.323 Annex M.2	ANSI T1.113-1988	“isup”	“ANSI T1.113-1988”	“1988”
H.323 Annex M.2	ETS 300 121	“isup”	“ETS 300 121”	“121”
H.323 Annex M.2	ETS 300 356	“isup”	“ETS 300 356”	“356”
H.323 Annex M.2	BELLCORE GR-317	“isup”	“BELLCORE GR-317”	“317”
H.323 Annex M.2	JT-Q761-4(1987-1992)	“isup”	“JT-Q761-4(1987-1992 )”	“87”
H.323 Annex M.2	JT-Q761-4(1993)	“isup”	“JT-Q761-4(1993)”	“93”

以上