

TTC標準
Standard

JT-H460.1

**機能拡張のための汎用フレームワーク
使用のためのガイドライン**

Guidelines for the Use of the Generic Extensible
Framework

第1版

2003年11月27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。

< 参考 >

1 . 国際勧告などとの関連

本標準はITUにおいて制定されたH . 4 6 0 . 1 第1版(2 0 0 2 年) に準拠している。

2 . 上記国際勧告に対する追加項目等

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

(1) 本標準は上記ITU - T 勧告に対し、先行している項目はない。

(2) 本標準は上記ITU - T 勧告に対し、追加した項目はない。

(3) 本標準は上記ITU - T 勧告に対し、削減した項目はない。

3 . 改版の履歴

版数	制定日	改版内容
第1版	2003年11月27日制定	

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。

目次

1	適用範囲.....	5
2	参照している標準.....	5
3	略語.....	5
4	GEF とは.....	5
5	GEF を使う場面.....	5
6	GEF モジュールの識別.....	7
7	GEF モジュールの定義.....	7
7.1	表に基づく方法.....	7
7.1.1	表に基づく方法を用いた、架空フィーチャの定義例.....	8
7.2	未加工のまま符号化する方法.....	9
	補遺 I (参考資料).....	10
I.1	GEF ネゴシエーションの例.....	10
I.1.1	RAS におけるシナリオ.....	10
I.1.2	呼シグナリングにおけるシナリオ.....	12
	参考文献.....	20

要約

本標準では、H.323 (2000) (第4版) で述べられた“機能拡張のための汎用フレームワーク”の使い方についてのガイドラインを与える。ガイドラインではフレームワークをいつ用いるか、そしてフレームワークを用いてフィーチャをいかに記述するかについて述べる。またフレームワークのネゴシエーションスキームの動作についても例を与えている。

1 適用範囲

本標準は、JT-H323 で定義された機能拡張のための汎用フレームワーク（GEF）を用いる場面と方法についての情報を提供する。本標準は JT-H323 の記述を繰り返すことはしない。本標準はむしろ JT-H323 を拡張し洗練させるものであり、JT-H323 の置き換えとしてではなく、共に参照されるべきものである。

本標準が扱う主題は、GEF を使う場面、GEF モジュールの記述方法、および GEF ネゴシエーションメカニズムの例である。

2 参照している標準

以下に示す TTC 標準 / ITU-T 勧告およびその他の参照文献は、本標準を構成する規定が含まれており、本標準の本文中から参照されている。本標準出版時には、以下に示す版が有効であった。すべての標準 / 勧告や参照文献は改定されることがある。そのため、本標準を使用する場合は、以下に挙げた標準 / 勧告およびその他の参照文献について、最新版が適用できるかどうかを調べることが望ましい。最新版の TTC 標準 / ITU-T 勧告リストは定期的に出版されている。

- ITU-T Recommendation H.323 (2000), Packet based Multimedia Communications Systems.
- ITU-T Recommendation H.225.0 (2000), Media Stream Packetization and Synchronization on Non-Guaranteed Quality of Service LANs

3 略語

本標準の目的のために、次の略語が使われる。

ABNF	拡張バックスナウア記法
ASN.1	抽象構文記法 1
GEF	機能拡張のための汎用フレームワーク
XML	拡張可能なマーク付け言語

4 GEF とは

機能拡張のための汎用フレームワーク（GEF）は、H.225.0 の ASN.1 に基づく仕様を新たに追加することなく H.323 に機能を追加するためのオーバーヘッドの小さな方法を提供する。GEF は、共通のフィーチャネゴシエーションメカニズムとすべての JT-H225.0 および JT-H225.0 付属資料 G のメッセージの中で任意の未定義データを運ぶ能力を提供する。即ち GEF により JT-H323 標準に対するアプリケーション固有の拡張を、他の全ての JT-H323 実装がアプリケーション固有の拡張が要求する全ての拡張機能を、実装する負担を負わせることなく行うことが可能になる。

GEF は JT-H323 の本文中で形式的に説明される。

5 GEF を使う場面

新しい機能は、JT-H225.0 の ASN.1 に基づく定義への新しい文法の追加、GEF モジュールの定義、または JT-H450 付加サービスの定義によって JT-H323 に追加することができる。いず

れの方法を用いるかを決定するとき、フィーチャネゴシエーションの観点とフィーチャのデータ転送の観点を別々に考慮する方が適切なことがある。つまり、主として JT-H450 付加サービスとしてフィーチャを実装し、一方 GEF のフィーチャネゴシエーション能力を用いてフィーチャをネゴシエーションするのが、ある場合においては適当である。

新しい機能の追加に用いるべき方法を決定する、以下のガイドラインを提案する。

- そのフィーチャが多くのアプリケーションに適用できるならば、JT-H225.0 の ASN.1 に基づく仕様にそれを直接加えることは適当であろう。ただし JT-H323 は十分成熟しているので、そのようなフィーチャに遭遇する可能性は低いこと、そしてこの方針をとるためには有力な事例を作る必要があるであろうことに注意せよ。この方法が起こるもっとも可能性の高い状況は、AliasAddress や TransportAddress のような既に存在するパラメータを拡張する場合である。
- または、フィーチャが少数のパラメータのみを要求するならば、それを GEF モジュールとして定義する。
- または、フィーチャが JT-H225.0 RAS または JT-H225.0 付属資料 G の中でデータを送ることを要求するならば、それを GEF モジュールとして定義する。
- または、フィーチャが、データが JT-H225.0 RAS または JT-H225.0 付属資料 G で運ばれることを必要とするならば、それを GEF モジュールとして定義する。
- または、フィーチャが多数のパラメータを含むならば、それを GEF content 構造体の未加工の要素に符号化される ASN.1 (または、ABNF や XML のような同様の方法) として定義するか、JT-H450 付加サービスを定義する。

これらの考慮点は下の図 1 に示すフローチャートに表される。

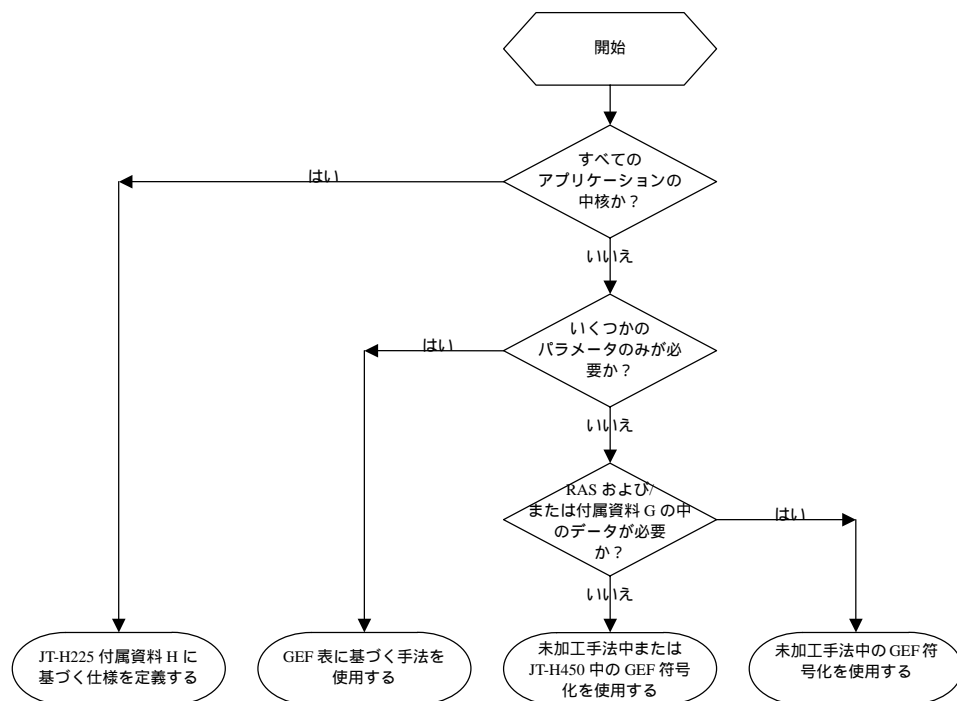


図 1/JT-H460.1

新しいフィーチャを記述する方法のための決定木

6 GEF モジュールの識別

GEF は、TTC/ITU-T 標準のモジュール定義と非 TTC/ITU-T 標準の両方を許容する。すべての TTC/ITU-T 標準 GEF モジュール定義は、JT-H460/H.460 シリーズの一部として定義されなくてはならない。そして、それらを識別するために GenericData 中の id の standard バリエーションを用いなくてはならない。すべての非 TTC/ITU-T 標準モジュールはそれらを識別するために、GenericData 中の id の nonStandard または oid バリエーションのいずれかを使わなくてはならない。

JT-H460 シリーズ標準の識別と GenericData 中の standard に符号化された値との間には直接の関係がある。すなわちモジュールの識別のために、標準 JT-H460.n(n は整数)は、その n を GenericData 内の id の standard 変数に使用しなくてはならない。たとえば、JT-H460.2 は GenericData の id が 2 にセットされた standard バリエーションを持たなくてはならない。

7 GEF モジュールの定義

GEF モジュールは様々な方法で定義することができるが、定義の明確さが最も重要である。GEF モジュール定義についてある程度の一貫性を提供するために、本セクションでは GEF モジュールを定義する 2 つの方法に付いて述べる。テーブルに基づく方法と、未加工のまま符号化する方法である。

フィーチャがネゴシエーションと任意データの転送の双方を必要とする場合、仕様書には要求事項を、それぞれについて別個の節として記述しなくてはならないことがあることに注意せよ。

7.1 表に基づく方法

表に基づく方法では、フィーチャを構成する個々のパラメータは表形式で表される。データ交換のため、それぞれのパラメータ定義は別個の EnumeratedParameter にマッピングされる。なお、通常この表はフィーチャの全ての使用目的や操作を表しているものではなく、一般に追加の説明文書が必要とされることに注意せよ。

表 1 がフィーチャの全容を表すために使用される。この表はフィーチャの名称、識別子、それと簡単な説明を持っている。この表 1 つでネゴシエーションとデータ転送の双方を持つフィーチャについて十分に表すことができる。

フィーチャ名:	フィーチャの名称
フィーチャ概説:	このフィーチャの簡潔な説明。本項目に対してはフィーチャの仕様文書において詳説されている場合がある。
フィーチャ識別子種別:	識別子が standard,oid,nonStandard のいずれであることを示す
フィーチャ識別子値:	識別子の値

表 1/JT-H460.1

フィーチャの表形式定義

フィーチャが使用するそれぞれのパラメータは、表 2 の例によって説明される。ここでまた、パラメータ名と識別子と概説が他の情報と共に表に含まれる。構造体に含まれて送られるパ

ラメータや入れ子になっているパラメータは、仕様書の別個の節において説明されるべきである。パラメータがフィーチャのネゴシエーションとデータ転送の双方の局面において必要とされる場合、それぞれフィーチャ仕様書の別個の節に置くほうが適切であろう。

パラメータ名	パラメータの名称
パラメータ概説	このパラメータの簡潔な説明
パラメータ識別子種別	識別子が standard,oid,nonStandard のいずれであるかを示す
パラメータ識別子値	識別子の値
パラメータ種別	未加工、text、Unicode text、ブール値、, number8, number16, number32, id, alias, transport, compound, nested. のいずれか
パラメータ基数	このパラメータが何回現れるか。

表 2/JT-H460.1

パラメータの表形式定義

7.1.1 表に基づく方法を用いた、架空フィーチャの定義例

本節では表に基づく方法を用いた架空フィーチャの定義例を示す。

架空サンプルフィーチャの定義

フィーチャ名:	架空サンプル
フィーチャ概説	このフィーチャは、テーブルに基づく方法による GEF モジュールの定義を解説するための架空フィーチャである。
フィーチャ識別子種別	Standard
フィーチャ識別子値:	1

架空サンプルフィーチャのパラメータ

パラメータ名	priority
パラメータ概説	他の着信呼に対する優先度記述
パラメータ識別子種別	Standard
パラメータ識別子値	0
パラメータ種別	number8
パラメータ基数	必ず 1 回、また唯一 1 回のみ

パラメータ名	destination
パラメータ概説	接続しようとしている相手の識別子
パラメータ識別子種別	Standard
パラメータ識別子値	1
パラメータ種別	alias
パラメータ基数:	1 つ以上

パラメータ名	message-display-lines
パラメータ概説	呼の処理中に表示すべき 1 つ以上のメッセージ
パラメータ識別子種別	Standard
パラメータ識別子値	2
パラメータ種別	text
パラメータ基数	0 個以上

7.2 未加工のまま符号化する方法

GEF モジュールを定義する 2 番目の方法では、フィーチャを既存のメッセージ符号化法、例えば ASN.1,XML や ABNF を用いて定義する。本手法を用いる場合、メッセージ片はそれぞれの符号化法の標準ルールに基づいて符号化され、GEF パラメータの未加工変数として交換される。本方法は特にパラメータを多く含み、テーブルに基づく方法では煩雑であるフィーチャに対して有効である。

GEF パラメータの内容を定義するのに加えて、GEF パラメータの識別子を定義する必要がある。識別子の定義はテーブルに基づく方法を用いるか、説明文によって行う。

ASN.1 を用いてフィーチャを定義する場合、PER 符号化の基本配置化を用いるのが望ましい。しかしながら、本勧告が上記符号化を推奨するかどうかに関わらず、使用する符号化ルールはフィーチャの仕様書で明言されていなくてはならない。

ABNF を用いてフィーチャを定義する場合、特に変換等を行うことなく、H.323 と類似した機能を持つ他のプロトコル (SIP 等) とフィーチャ間相互接続を行うことができる可能性がある。このような表現を用いる場合においても、文字について UTF-8 符号化を使用できるように未加工レメント内で転送されるべきであることに注意せよ。

補遺 I (参考資料)

(この補遺は本標準の必須要素ではない)

I.1 GEF ネゴシエーションの例

本補遺では GEF ネゴシエーション機構の様々な局面について解説する。本補遺は解説のみを目的としたものである。

I.1.1 RAS におけるシナリオ

下図はゲートキーパとネゴシエーションを行っているエンドポイントを表している。

解説の為にこのネゴシエーションは RRQ/RCF/RRJ メッセージセットを用いて示されているが、GEF のネゴシエーションはこのメッセージセットに限られるわけではない。

この例ではエンドポイントが課金通知機能(架空の機能、以下 CAF と表記)の利用を希望しているとする。CAF 機能とは、詳細はともかく、基本的にはゲートキーパがエンドポイントに対し、呼の終了時に DCF メッセージを用いて何らかの課金情報を通知する、というようなものである。

同様にゲートキーパが帯域管理機能(架空の機能、BWF と表記)の利用を希望しているとする。BWF 機能とは、詳細はともかく、基本的にはモジュール内のパラメータにエンドポイントが守らなくてはならない何らかの帯域管理テンプレートが含まれる、というようなものである。

以下のダイアグラムでは、ネゴシエーション処理の様々な局面について解説する。

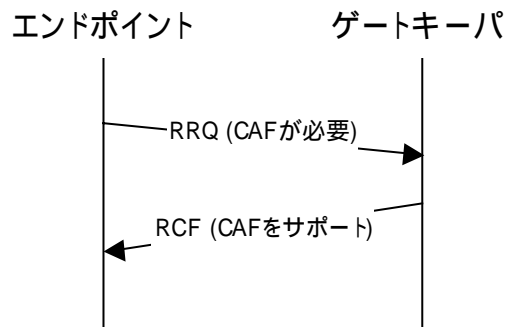


図 I.1/JT-H460.1: 条件:

エンドポイントは CAF 機能を必要/要求している

ゲートキーパは CAF 機能をサポートしており、

リクエストは成功した

解説: 図 I.1 では、エンドポイントが CAF 機能を要求し、ゲートキーパは CAF 機能をサポートしていることを通知する。ネゴシエーションは成功し、エンドポイントは CAF 機能を利用することができる。

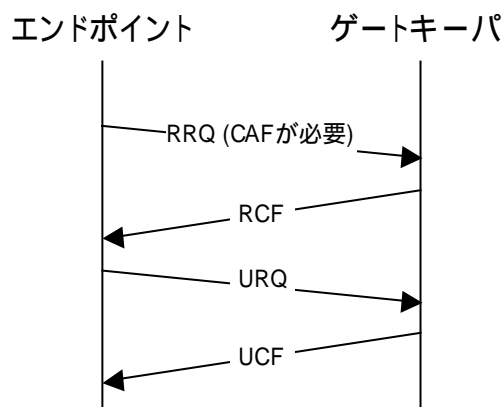


図 I.2/JT-H460.1: 条件:

エンドポイントは CAF 機能を必要/要求している
 ゲートキーパは GEF をサポートしていない
 GEF 手順は失敗する

(エンドポイントは応答を受けた際、後方互換性を処理しなくてはならない)

解説: 図 I.2 では、エンドポイントは CAF 機能を要求したが、ゲートキーパは GEF ネゴシエーション機構をサポートしておらず、CAF 機能要求の存在を認識しない場合を示している。ゲートキーパは CAF 機能の要求を認識しないため、RRQ を受け付けることができると考え、RCF を送信する。エンドポイントはゲートキーパが CAF のサポートを通知しなかったことを受け、エンドポイントが必要とする機能を提供しないゲートキーパから離脱する。

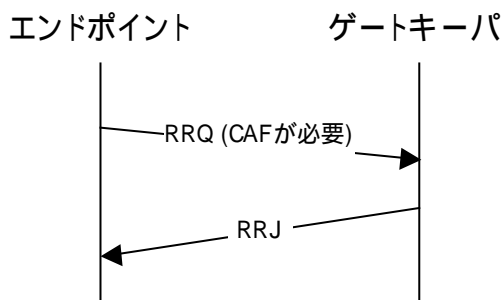


図 I.3/JT-H460.1: 条件:

エンドポイントは CAF 機能を必要としている
 ゲートキーパは CAF 機能をサポートしていない
 リクエストは失敗する

解説: 図 I.3 は図 I.2 に類似しているが、ゲートキーパが GEF ネゴシエーション機構をサポートしている点が異なる。ゲートキーパは自身が CAF 機能をサポートしていないことを認識しており、RRJ にてリクエストを拒否する。

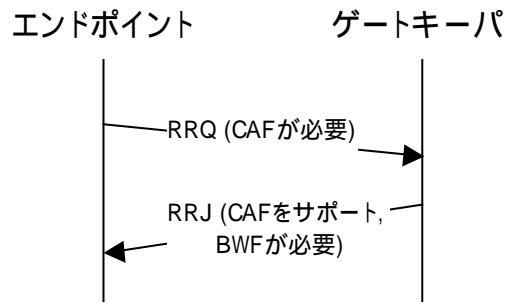


図 I.4/JT-H460.1: 条件:

エンドポイントは CAF 機能を必要としている
ゲートキーパは CAF をサポートするが、同時に BWF 機能も必要としている
リクエストは失敗する

解説: 図 I.4 では、エンドポイントは CAF 機能を必要としており、またゲートキーパは BWF 機能を必要としている。エンドポイントは RRQ にて CAF 機能が必要だと示している。ゲートキーパは CAF が必要とされていることを認識し、またゲートキーパは CAF をサポートしている。しかしながらゲートキーパはエンドポイントが BWF をサポートしていることを確認できない。そのためゲートキーパは、CAF 機能はサポートしているが BWF 機能を必要としていることを示し、リクエストを拒否する。

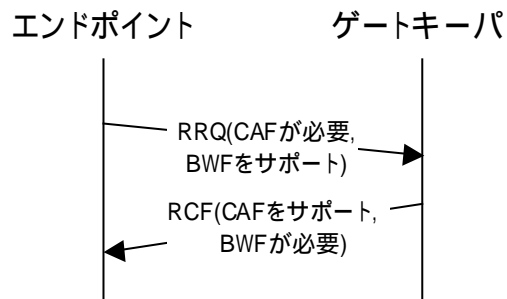


図 I.5/JT-H460.1: 条件:

エンドポイントは CAF を必要とし、また BWF をサポートする
ゲートキーパは CAF をサポートし、また BWF を必要とする
リクエストは成功する

解説: 図 I.5 では、エンドポイントは CAF を必要としていることと BWF をサポートしていることを示している。ゲートキーパは CAF 機能をサポートしているため、CAF サポートをその応答に示している。またゲートキーパはエンドポイントが BWF 機能を使用することを必要としているため、BWF が必要だと応答に示している。

I.1.2 呼シグナリングにおけるシナリオ

以下の図に GEF の呼シグナリングネゴシエーションの例を示す。ダイレクトモデルはゲートキーパルーテッドモデルの特別な(単純な)例であるということを踏まえ、ゲートキーパルー

テッドモデルの例を示す。

これらの例において、仮想の機能 *DPNSS*, *450Proxy* および *アカウント* がを使用する。

これらの機能の内部詳細はここでの議論では重要ではないので記述しない。

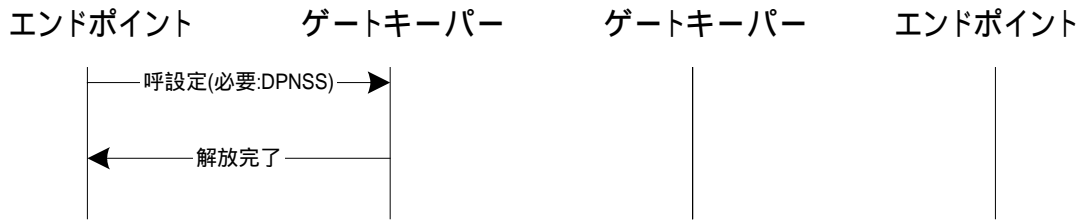


図 I.6/JT-H460.1. 条件:

発側エンドポイントは DPNSS 機能が必要
最初のゲートキーパ は機能をサポートしない
ネゴシエーション失敗

解説: 図 I.6 は、仮想の DPNSS 機能を要求する呼を開始しているエンドポイントを示す。最初のゲートキーパは、GEF ネゴシエーションをサポートするが、DPNSS 機能をサポートしないため呼を拒絶する

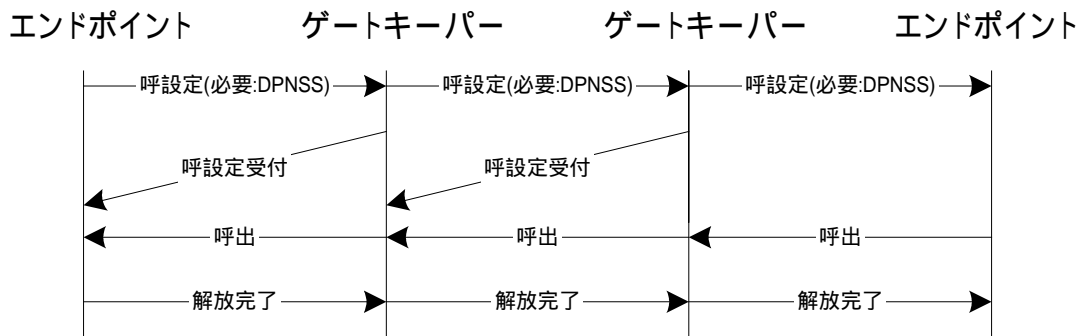


図 I.7/JT-H460.1. 条件:

発側エンドポイントは DPNSS 機能が必要
ゲートキーパは GEF をサポートする
着側エンドポイントは GEF をサポートしない
ネゴシエーション失敗

解説: 図 I.7 にエンドポイントが DPNSS 機能を要求する呼を開始している例を示す。仲介するゲートキーパは GEF をサポートするが、受け側のエンドポイントは (GEF を) サポートしない。したがって DPNSS 機能が必要であることを認識できず、従来手順で呼出メッセージを送信する。呼出メッセージは DPNSS 機能のサポート表示を含んでいないため、発信エンドポイントは呼を終了させる。

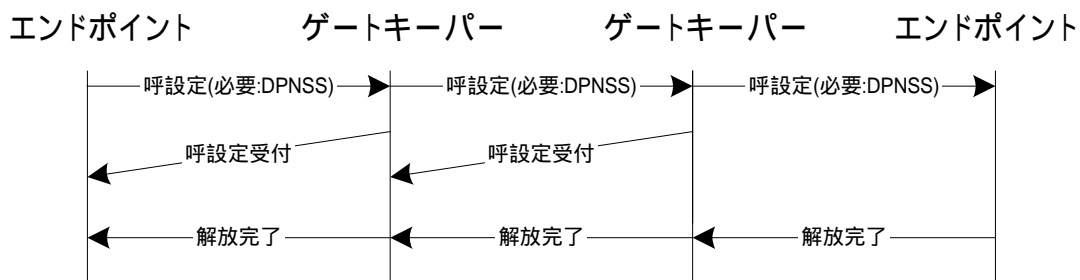


図 I.8/JT-H460.1. 条件:

発側エンドポイントは DPNSS 機能が必要

ゲートキーパーは GEF をサポートする

着側エンドポイントは DPNSS 機能をサポートしない

ネゴシエーション失敗

解説: 図 I.8 ではエンドポイントは DPNSS 機能を使って呼を開設することを希望する。着側エンドポイントは GEF ネゴシエーションをサポートするが、DPNSS 機能をサポートしないため呼を拒絶する。

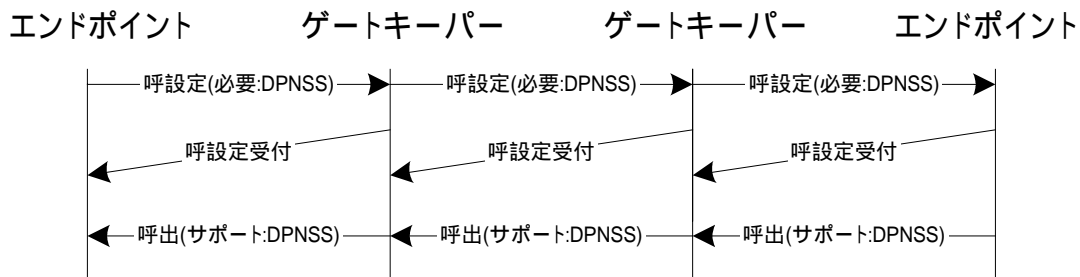


図 I.9/JT-H460.1. 条件:

発側エンドポイントは DPNSS 機能が必要

着側エンドポイントは DPNSS 機能をサポートする

ネゴシエーション成功

解説: 図 I.9 で示されるケースでは、着側エンドポイントは DPNSS 機能をサポートする。したがって呼を受け入れることが可能である。それは呼出メッセージ内の DPNSS のサポートで示され、そしてそれは最初のエンド - エンドのメッセージとなる。

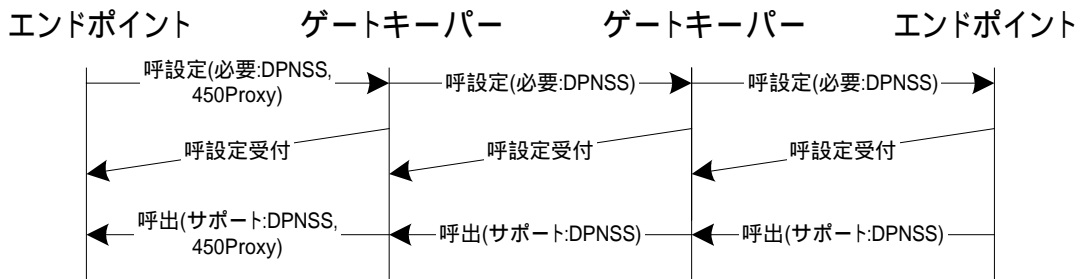


図 I.10/JT-H460.1. 条件:

発側エンドポイントは DPNSS 機能と 450Proxy 機能が必要
 ローカルゲートキーパは 450Proxy 機能をサポートする
 着側エンドポイントは DPNSS 機能をサポートする
 ネゴシエーション成功

解説: 図 I.10 はエンドポイントが (450Proxy のように認識される) H.450 proxying 機能のサポートも要求していることを除くと、図 I.9 に示されたケースに類似している。最初のゲートキーパはこれ (H.450 proxying 機能) に対する責務を受け入れて、そして前方ルーティングする前に、必要とされる機能をメッセージから取り除く。着側エンドポイントは DPNSS 機能を実装し、そしてそれを示した呼出メッセージを返送する。呼出メッセージが最初のゲートキーパを通過するとき、ゲートキーパは H.450 プロキシ機能をサポートするメッセージに修正する。

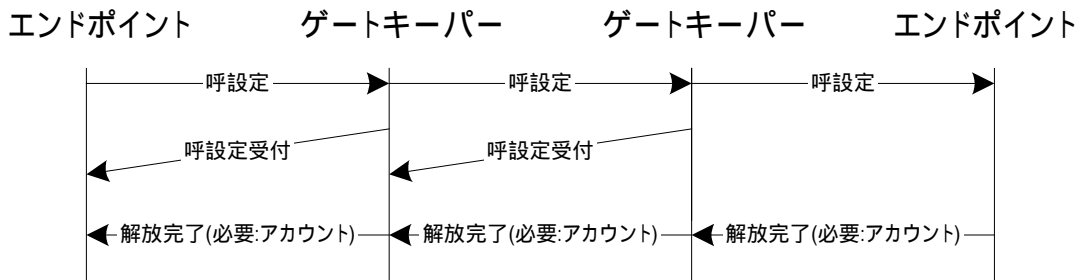


図 I.11/JT-H460.1. 条件:

発側エンドポイントはフレームワークをサポートしないまたは機能を持っていない
 着側エンドポイントはアカウント機能を必要
 ネゴシエーション失敗

記述: 図 I.11 でエンドポイントはアカウント機能の指定が必要な着側エンドポイントに対して呼を開始している。発側エンドポイントがアカウント機能をサポートすることを示さなかった場合、着側エンドポイントは呼をクリアする。

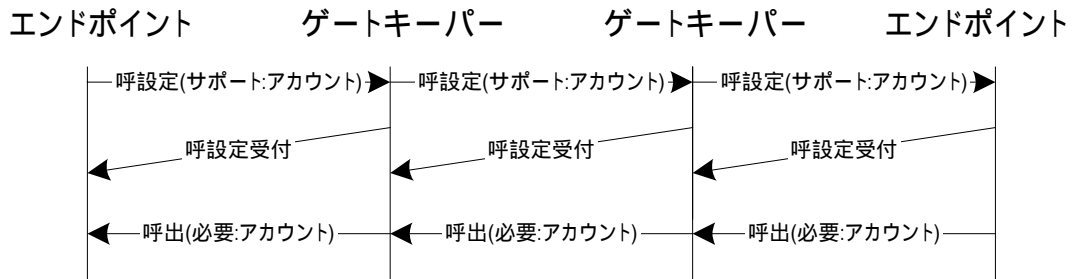


図 I.12/JT-H460.1. 条件:

発側エンドポイントはアカウント機能をサポートする
 着側エンドポイントはアカウント機能が必要
 ネゴシエーション成功

解説：図 I.12 は発側エンドポイントがアカウント機能をサポートすることを示す場合を示す。着側エンドポイントはこの機能を必要とし、そしてその機能に対するサポートが呼設定メッセージで示される場合、呼を受け入れることが可能となる。

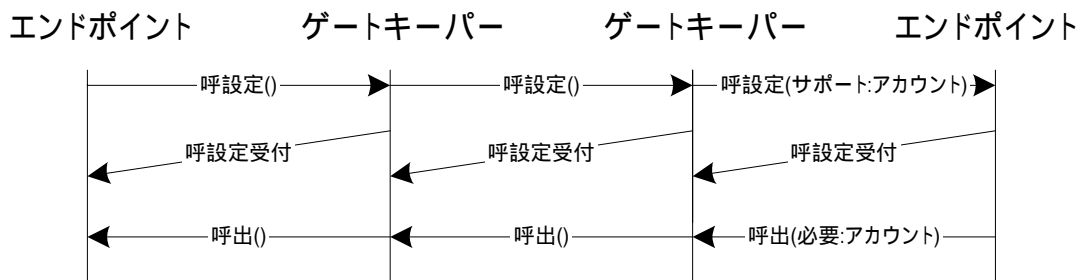


図 I.13/JT-H460.1: 条件:

発側エンドポイントはアカウント機能をサポートしない
 2番目のゲートキーパーはアカウント機能をサポートする
 着側エンドポイントはアカウント機能が必要
 ネゴシエーション成功

記述：図 I.13 ではゲートキーパーが前の例と同様にそれ自身のサポートする機能を追加している場合を示す。本例においても着側エンドポイントはアカウント機能を必要とするがアカウント機能は発側エンドポイントによって示されていない。本例の場合、最後のゲートキーパーは着側エンドポイントが呼を受け入れるためにアカウント機能のサポートが必要なことを知っているため、この機能のサポートを示す。アカウント機能がサポートされるので、着側エンドポイントは呼を受け入れることが可能となる。

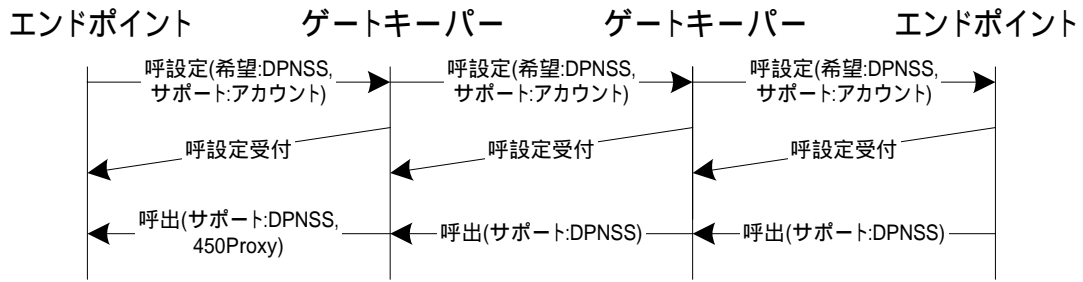


図 I.14/JT-H460.1. 条件:

発側エンドポイントは DPNSS 機能を希望するが必須ではない。

アカウント機能をサポートする。

着側エンドポイントは DPNSS 機能をサポート

最初のゲートキーパーは 450Proxy 機能をサポートし、求められていなくても、

それを明示する。

ネゴシエーション成功

解説：図 I.14 では、発側エンドポイントは DPNSS 機能を希望するが必須ではない。着側エンドポイントは DPNSS 機能をサポートしているため、呼出メッセージでこれを示す。（注釈：この場合、最初のゲートキーパーは要求されていない 450Proxy 機能のサポートを付加している。これはサポートしている機能を示す最も容易な妥当で一般的な実装手段である。）

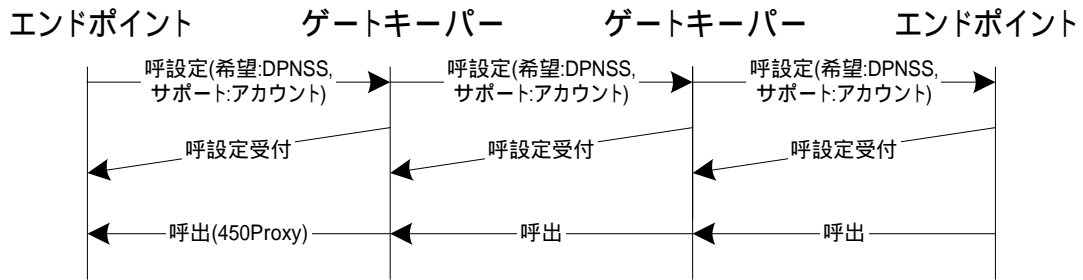


図 I.15/JT-H460.1. 条件:

発側エンドポイントは DPNSS 機能を希望するが必須ではない。

アカウント機能をサポートする。

着側エンドポイントは DPNSS 機能をサポートしない

最初のゲートキーパーは 450Proxy 機能をサポートし、求められていなくても、

それを明示する。

ネゴシエーション成功

解説: 図 I.15 に示されたケースは着側エンドポイントが DPNSS 機能をサポートしないことを除き、図 I.14 に類似のシナリオである。

しかし、DPSS 機能はただ希望されている機能に過ぎないので、呼を続けることが可能である。

発側エンドポイントはそれが呼出しメッセージで示されないことから、DPNSS 機能がサポートされないことを知る。

(注釈: この場合、最初のゲートキーパーは要求されていない 450Proxy 機能のサポートを付加している。これはサポートしている機能を示す最も容易な妥当で一般的な実装手段である。)

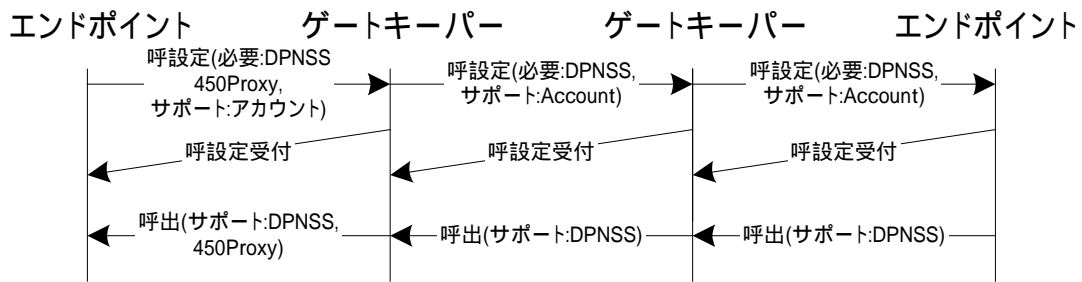


図 I.16/JT-H460.1.

能力交換要素の組み合わせ

解説:最後に、図 I.16 は呼シグナリングにおけるネゴシエーション機構について、複数の局面が同時に進行する例を示す。

発側エンドポイントは着側エンドポイントでの DPNSS 機能とローカルゲートキーパでの 450Proxy 機能を必要とする。また、アカウント機能をサポートしていることを示す。

ローカルゲートキーパは 450Proxy 機能を実行することができ、そしてメッセージを転送する前に、450Proxy 機能が必要だという要求をメッセージから取り除く。

着信エンドポイントは DPNSS 機能が必要とされることを認識し、それをサポートしている。着側エンドポイントは DPNSS 機能をサポートしていることを示す呼出メッセージを呼に返答する。

メッセージが発側エンドポイントのローカルなゲートキーパに到着すると、ゲートキーパは要求された 450Proxy 機能をサポートしていること呼出メッセージに付加することで示す。発側エンドポイントは必要な 2 つの機能がサポートされていることを呼出メッセージで知る。

参考文献

次の引用は参考のために記載しており、この標準の条項を構成しない。

- ITU-T Recommendation X.680 (1997), Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.
- ITU-T Recommendation X.681 (1997), Information technology – Abstract Syntax Notation One (ASN.1): Information object specification.
- ITU-T Recommendation X.691 (1997), Information technology – ASN.1 encoding rules – Specification of Packed Encoding Rules (PER).
- IETF RFC-2234, November 1997, Augmented BNF for Syntax Specifications: ABNF.
- IETF RFC-2543, March 1999, SIP: Session Initiation Protocol.
- IETF RFC-2279, January 1998, UTF-8, a transformation format of ISO 10646.
- W3C REC-xml, October 2000, Extensible Markup Language (XML) 1.0 (Second Edition).