

TTC標準
Standard

JT- H450.9

J T - H 3 2 3 のための呼完了付加
サービス

〔 Call Completion Supplementary Services for JT-H323 〕

第 1 版

2001 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。

目次

<参考>	5
要約	6
1. 適用範囲	7
2. 参照している標準	7
3. 用語と定義	7
4. 略語と頭字語	8
5. SS-CCBS サービスの説明	9
5.1. 一般的な手順	9
5.1.1. 開始 / 停止 / 登録 / 問い合わせ	9
5.1.2. インボケーションおよび処理	9
5.1.3. 取消	10
5.2. 例外的な手順	10
5.2.1. インボケーションおよび処理	10
5.2.2. 取消	12
5.3. その他の付加サービスとの相互作用	12
5.3.1. 無応答時の呼完了 (SS-CCNR)	12
5.3.2. コールトランスファ (SS-CT)	12
5.3.3. 無条件着信転送 (SS-CFU)	12
5.3.4. ビジー時着信転送 (SS-CFB)	12
5.3.5. 無応答時着信転送 (SS-CFNR)	13
5.3.6. 呼保留	13
5.3.7. コールパーク / コールピックアップ	13
5.3.8. コールウェイティング	13
5.3.9. メッセージウェイティング通知	13
5.3.10. 名前表示	13
5.3.11. コールリンケージ	13
6. SS-CCNR サービスの説明	13
6.1. 一般的な手順	14
6.1.1. 開始 / 停止 / 登録 / 問い合わせ	14
6.1.2. インボケーションおよび処理	14
6.1.3. 取消	15
6.2. 例外的な手順	15
6.2.1. インボケーションおよび処理	15
6.2.2. 取消	16
6.3. 他の付加サービスとの相互作用	16
6.3.1. ビジー加入者への呼完了 (SS-CCBS)	16
6.3.2. コールトランスファ (SS-CT)	16
6.3.3. 無条件着信転送 (SS-CFU)	16
6.3.4. ビジー時着信転送 (SS-CFB)	16
6.3.5. 無応答時着信転送 (SS-CFNR)	17
6.3.6. 呼保留	17

6.3.7. コールパーク/コールピックアップ	17
6.3.8. コールウェイティング	17
6.3.9. メッセージ待機通知	17
6.3.10. 名前表示	17
6.3.11. コールリンケージ	17
7. メッセージと情報要素	17
8. 手順	18
8.1. ユーザ A のエンドポイントの動作	18
8.1.1. 正常手順	18
8.1.2. 例外手順	21
8.2. ユーザ B のエンドポイントでの動作	23
8.2.1. 正常手順	23
8.2.2. 例外手順	25
9. 相互接続と相互動作	26
9.1. SCN との相互接続	26
9.1.1. ファンクショナル呼完了エンティティが SCN に存在しない	27
9.1.2. ファンクショナル呼完了エンティティが SCN に存在する	27
9.2. SS-CCBS と他の付加サービスとのプロトコル相互動作	28
9.2.1. コールトランスファ (JT-H450.2)	28
9.2.2. 着信転送 (JT-H450.3)	28
9.2.3. コールホールド (JT-H450.4)	29
9.2.4. コールパークとコールピックアップ (JT-H.450.5)	29
9.2.5. コールウェイティング (JT-H450.6)	30
9.2.6. メッセージウェイティング (JT-H450.7)	30
9.2.7. 名前表示 (JT-H450.8)	30
9.2.8. コールリンケージ (H.323 version 4)	30
9.3. SS-CCNR と他の付加サービスとのプロトコル相互動作	30
9.3.1. コールトランスファ (JT-H450.2)	30
9.3.2. 着信転送 (JT-H450.3)	30
9.3.3. コールホールド (JT-H450.4)	32
9.3.4. コールパークとコールピックアップ (JT-H450.5)	32
9.3.5. コールウェイティング (JT-H450.6)	32
9.3.6. メッセージウェイティング (JT-H450.7)	32
9.3.7. 名前表示 (JT-H450.8)	32
9.3.8. コールリンケージ (JT-H323 Version 4)	32
10. ゲートキーパー動作	33
10.1. ゲートキーパーはエンドポイントに SS-CC オペレーションを透過する。	33
10.2. ゲートキーパー/プロキシがエンドポイントの代わりに動作する	33
10.2.1. ゲートキーパー/プロキシがエンドポイント B の代わりに動作する	33
10.2.2. ゲートキーパー/プロキシがエンドポイント A の代わりに動作する	33
11. ダイナミック記述	34
11.1. オペレーションモデルおよびシグナルフロー	34
11.1.1. 成功 CCBS	34

11.1.2. 成功 CCNR.....	38
11.1.3. ユーザ A ビジー	39
11.1.4 ユーザ B 再ビジー	40
11.2. エンドポイント A シグナリングエンティティ (EASE) とそのシグナリングエンティティユーザ (情報を与える) 間の通信.....	43
11.2.1. プリミティブ表.....	44
11.2.2. プリミティブ定義.....	44
11.2.3. パラメータ定義.....	44
11.2.4. 呼状態.....	46
11.3. エンドポイント B シグナリングエンティティ (EBSE) とそのシグナリングエンティティユーザ (情報を与える) 間の通信.....	46
11.3.1. プリミティブ表.....	46
11.3.2. プリミティブ定義.....	46
11.3.3. パラメータ定義.....	47
11.3.4. 呼状態.....	48
11.4. タイマ.....	48
11.4.1. ユーザ A のエンドポイントにおけるタイマ.....	48
11.4.2. ユーザ B のエンドポイントにおけるタイマ	49
12. 呼完了付加サービスのサポートにおける操作.....	49
13. SS-CCBS の仕様記述言語 (SDL) 図.....	52
13.1. ユーザ A のエンドポイントの動作.....	53
13.2. ユーザ B のエンドポイントの動作	57

< 参考 >

1 . 国際勧告等との関連

本標準はITU において制定されたH . 3 2 3 のための呼完了サービスH . 4 5 0 . 9 第1 版(2 0 0 0 年) に準拠している。

2 . 上記勧告等に対する追加項目等

本標準では国際標準において編集上の誤りと考えられる点については修正すると共に本文中にその旨を「注記」として明記した。

3 . 改版の履歴

版 数	制 定 日	改版内容
第1 版	2001 年 11 月 27 日	制定

4 . 工業所有権

本標準に関わる「工業所有権等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、標準等

ITU - T 勧告 :

H.323 、 H.225.0 、 H.245 、 H.450.1 、 H.450.9

ISO / IEC 標準 :

ISO/IEC 13866

ISO/IEC 13870

TTC 標準 :

JT-H323 、 JT-H225.0 、 JT-H245 、 JT-H450.1 、 JT-H450.3

(2)他の国内標準との関連

他の国内標準との関連はない。

6 . 標準作成部門

第三部門委員会 第二専門委員会

要約

この新標準は JT-H323 における呼完了付加サービスの手順とシグナリングプロトコルについて記述している。

呼完了付加サービスとはビジー加入者への呼完了 (SS-CCBS) と無応答時の呼完了 (SS-CCNR) のことである。

SS-CCBS は、発ユーザ A が、着ユーザ B がビジーであった場合、ユーザ B がビジーでなくなった時に新たな呼を試みることなく呼を完了することができる。

SS-CCNR は、発ユーザ A が、着ユーザ B を呼出しているにもかかわらず返事がない場合、アクティビティ期間後に再びビジーでなければ、新たな呼を試みることなく呼を完了することができる。

この標準は、JT-H450.1 標準で規定されている「JT-H323 における付加サービス実現のための汎用機能プロトコル」を利用している。

この標準は、JT-H323 バージョン 2 (1998) またはそれ以降のバージョンを要求する。バージョン 2 製品は、**protocolIdentifier** = {itu-t (0) recommendation (0) h (8) 2250 version (0) 2} を含む JT-H225.0 メッセージと **protocolIdentifier** = {itu-t (0) recommendation (0) h (8) 245 version (0) 3} を含む JT-H245 メッセージによって識別できる。

この標準の手順とシグナリングプロトコルは、ISO/IEC 13866 と 13870 で明記されている呼完了付加サービスから得ている。

注意：この文章は、公に利用可能な ITU 発刊物ではない。しかし、内部 ITU ドキュメントとして ITU メンバー、そのセクターメンバー、それぞれのスタッフおよび ITU に関連した仕事をしている共同体によってのみ利用される。事前の ITU の書類による承諾なしに、他の人や団体に使用させてはならない。

1. 適用範囲

この標準は呼完了付加サービス (SS-CC) について記述しており、それは JT-H323 マルチメディアエンドポイントにおいてサポートされる様々な基本サービスに適用される。

SS-CC は二つの付加サービスにより構成される。それはビジイー加入者への呼完了付加サービス (SS-CCBS) と、無応答時の呼完了付加サービス (SS-CCNR) である。SS-CCBS は、ビジイー状態のため失敗した、加入者 (すなわちユーザ) への呼完了を認め、SS-CCNR は、呼出しているにもかかわらず無応答のため失敗した、ユーザへの呼完了を認める。

2. 参照している標準

以下の ITU-T 勧告、TTC 標準と他の参考文献は、本文中の参照している標準により標準の規定を構成する条項を含んでいる。発行の際、示されている版は有効であった。すべての勧告、標準および他の参考文献は改訂される。そのため、本標準の全ユーザは、以下にリストされている勧告、標準および他の参考文献の最新版を適用する可能性を調査することを奨励する。現在有効な ITU-T 勧告、TTC 標準のリストは、定期的に出版されている。

- [1] TTC 標準 JT-H323 Version 2 (1998) またはそれ以降 – パケットに基づくマルチメディア通信システム
- [2] TTC 標準 JT-H225.0 Version 2 (1998) またはそれ以降 – パケットベースマルチメディア伝送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
- [3] ITU-T 標準 H.245 Version 3 (1998) またはそれ以降 – マルチメディア通信のための制御プロトコル
- [4] ITU-T 標準 H450.1 (1998) – JT-H323 における付加サービス実現のための汎用機能プロトコル

3. 用語と定義

本標準の目的のために以下の定義が適用される。

呼 (Call) :

標準 JT-H323 参照。

呼完了 (Call Completion) :

前に失敗した着ユーザ (ユーザ B) への成功した呼を表現。それは呼が呼出しフェーズにはいったとき、または返答されたときに発生する。

CC 呼 (CC Call) :

ユーザ A にかわって、前に失敗したユーザ A のエンドポイントからユーザ B への呼について CC 要求を実行している間での再開。

CC リコール (CC Recall) :

ユーザ B はもはやビジイーでない (SS-CCBS での状況)、またはアクティビティ期間が完了した (SS-CCNR での状況) ことをユーザ A に知らせる通知。ユーザ A によるこの通知の受理によりその呼は完了される。

CC 要求 (CC Request) :

SS-CCBS、または SS-CCNR の一例。

接続解放 / 維持 (Connection release/retain) :

CC 要求のいろいろな段階で、呼と独立したシグナリング接続を解放 / 維持する能力。

エンドポイント、ゲートキーパー、ゲートウェイ、端末、ユーザ (Endpoint; gatekeeper; gateway; terminal; user) :

標準 JT-H323 参照。

空き (Free) :

呼を受け入れられるユーザの特性。(すなわち呼出、または応答状態に呼が到達できること。)

プロキシ (Proxy) :

SS-CC 手順をエンドポイントの代わりに実行するエンティティ。プロキシは、ゲートキーパーとともに設置してもよいし、設置しなくてもよい。

リコールタイマ (Recall timer) :

このタイマは、ユーザ A のエンドポイントが、ユーザ A から CC リコールへの応答を待たなければならない時間長を指定する。

サービス持続タイマ (Service duration timer) :

このタイマは、SS-CC 付加サービスが、ユーザ A のエンドポイントの内部でアクティブでなければならない時間長を指定する。

サービス維持 (Service retention) :

ユーザ B が再びビジーであったため CC 呼が失敗した後、CC 要求とともに継続するオプションの能力。

ユーザ A (User A) :

呼を発信したり、付加サービスを要求する発信ユーザ。

ユーザ B (User B) :

発信呼のセットアップではじめに呼出された遠隔ユーザ。

4. 略語と頭字語

本標準の目的のために、以下の略語を使用する。

APDU	Application Protocol Data Unit アプリケーションプロトコルデータユニット
ASN.1	Abstract Syntax Notation No. 1 抽象構文記法 1
CC	Call Completion 呼完了
CCBS	Completion of Calls to Busy Subscribers ビジー加入者への呼完了
CCNR	Completion of Calls on No Reply 無応答時の呼完了
CT	Call Transfer コールトランスファ
EASE	Endpoint A Signalling Entity エンドポイント A シグナリングエンティティ
EBSE	Endpoint B Signalling Entity エンドポイント B シグナリングエンティティ
ID	Identification, Identifier 識別、識別子
IE	Information Element 情報要素

NFE	Network Facility Extension 拡張ネットワークファシリティ
SCN	Switched Circuit Network 交換型回線ネットワーク
SDL	Specification and Description Language 仕様記述言語
SS-CC	Supplementary Service Call Completion 呼完了付加サービス（これは一般用語であり、SS-CCBS と SS-CCNR の共通する側面を表現するために用いられる）
SS-CCBS	Supplementary Service Completion of Calls to Busy Subscribers ビジー加入者への呼完了付加サービス
SS-CCNR	Supplementary Service Completion of Calls on No Reply 無応答時の呼完了付加サービス

5. SS-CCBS サービスの説明

ビジー加入者への呼完了（SS-CCBS）は、発ユーザ A に提供される付加サービスである。着ユーザ B がビジーであるとき、ユーザ A はユーザ B のエンドポイントにユーザ B を監視させることや、ユーザ B が空きになったときにユーザ A のエンドポイントに通知させることを要求することができる。ユーザ A によるこの通知の応答で、ユーザ A のエンドポイントは、ユーザ B への呼を完了することを試みなければならない。

5.1. 一般的な手順

5.1.1. 開始 / 停止 / 登録 / 問い合わせ

SS-CCBS は永久に起動されている。

5.1.2. インボケーションおよび処理

ユーザ A からの呼が、着ユーザ B がビジーであるために失敗した時、ユーザ A は SS-CCBS を要求できなくはない。

SS-CCBS の要求を受けたとき、ユーザ B のエンドポイントはそのサービスが起動できるかを調査しなくてはならず、もしできるのならばユーザ A のエンドポイントに確認を送り、ユーザ B の監視をはじめなければならない。確認を受けたとき、ユーザ A のエンドポイントは SS-CCBS サービス持続タイマを開始しなければならない。確認の意味は、SS-CCBS サービス持続タイマの時間内にユーザ B がフリーになったとき、ユーザ A が CC リコールの受け取りを求めることができるということである。

注1：その他のユーザから呼完了要求の結果としてユーザ B がすでに監視されているという事実は、ユーザ A からの要求の拒否理由には必要ない。同じユーザ B への複数の要求を処理することは、インプリメントマタ - である。それは一般的に、日付順、または優先順に整列されたキューの一種を伴う。

注2：ユーザ A がその他のユーザに対してすでに呼完了をインボークしたという事実は、ユーザ A からの更なる呼完了の拒否理由にはならない。同じユーザ A による複数の要求を処理することは、インプリメントマタ - である。

注3：SS-CCBS はユーザ B がビジーであるという通知の後インボークされる。SS-CCBS サービス持続タイマの推奨値は、1-60 分の範囲である。適当な初期値は 15 分にするとい。

注4：特定の基本サービスと連携した呼でのSS-CCBSの選択処理は、今後の検討課題である。

SS-CCBS 要求を知らされた後でも、ユーザ A は他の呼を受けたり、呼を起動できなくてはならない。

ユーザ A のエンドポイントは、ユーザ A がインボークしている未解決の CC 要求のリストを要求する能力をユーザ A に提供してもよい。そのリストは、未解決の CC 要求がない場合、空である。

ユーザ B の監視が、ユーザ B は空きになったと示したとき、ユーザ B のエンドポイントは CC リコールを提供しなければならない。ユーザ A のエンドポイントは、ユーザ A に CC リコールを通知し、リコールタイマを開始しなければならない。（タイマの値は 10-30 秒の範囲である。）

注：このような通知を受け取るとすぐに、ユーザ A は以下のことが可能となる

- l CC リコールの受け入れ
- l 通知の無視、それにより、CC リコールタイマが満了したとき、CC リコールはユーザ A のエンドポイントによって取消される。また
- l SS-CCBS 要求の取消

ユーザ A が CC リコールを受け取ったならば、ユーザ A のエンドポイントはリコールタイマを停止し、ユーザ A とユーザ B との間の呼を完了させようとしなくてはならない。ユーザ B への呼が成功し、呼出フェーズに入った場合、または応答した場合、SS-CCBS は完了とみなされなければならない。

5.1.3. 取消

ユーザ A のエンドポイントは、以下の取消のうち少なくとも一つを要求する能力をユーザ A に提供しなくてはならない。

- l CC リコールが今もなお期待されているすべての未解決の CC 要求
- l CC リコールが今もなお期待されている最も新しい CC 要求および
- l CC リコールが今もなお期待されている特定の CC 要求。

成功した取消はユーザ A に通知しなければならない。

5.2. 例外的な手順

5.2.1. インボケーションおよび処理

5.2.1.1. SS-CCBS サービス要求の拒否

ユーザ A が SS-CCBS の要求を許可されていない場合、ユーザ A のエンドポイントは、否認が短期間か長期間かの表示をして、SS-CCBS 要求を拒否しなければならない。

短期間の否認は、その後の SS-CCBS の要求が成功するという一時の状態で使用されなければならない。短期間の否認に帰着する状態の例として

- l ユーザ A による要求の限界がすでに達した
- l ユーザ B に対する要求の限界がすでに達した、また
- l 重複要求（5.2.1.4 項を参照）

長期間の否認は、その後の要求もまた拒否されてしまう場合に用いられなければならない。長期間の否認

に帰着する状態の例として、

- l ユーザ A に供給されない SS-CCBS
- l SS-CCBS をサポートしていないネットワークとの相互接続、または
- l ユーザ B に対して許可されていない SS-CCBS (注)。

注：いくつかのユーザのクラスに適用できるインプリメントオプション

5.2.1.2. CC リコールでユーザ A がビジー

ユーザ B が空きになったときユーザ A がビジーであると分かった場合、ユーザ A のエンドポイントは CC 要求をすぐに保留するか、もしくはユーザ A に CC リコールの通知がされ、CC リコールが実行される前(すなわちユーザ B への新たな呼が起動される前)にビジーでなくなるようにユーザ A のために短期間を待機した後、依然としてユーザ A がビジーである場合に保留しなければならない。

注：このような通知を受けるとすぐに、ユーザ A は以下のことが可能となる

- l 通知の無視、それにより CC 要求は保留される
- l SS-CCBS 要求の取消、また
- l リソースを空ける(たとえば既存呼を処理することによる)、それにより CC リコールを許容する

SS-CC を保留した後、ユーザ A が空きになったとき、ユーザ A のエンドポイントは SS-CC を再開しなければならない、それによりユーザ B は再び監視されるようになる。または 5.1.2 項で述べた呼の完了を試みなければならない。

5.2.1.3. CC リコールの成功後にユーザ B がビジー

ユーザ B が、ユーザ A の成功した CC リコールにより生じた呼のためビジーであるとき、ユーザ B のエンドポイントは以下のどちらかをしなければならない

- l 失敗の理由をユーザ A に表示し、CC 要求が取消されたことを表示して SS-CCBS を放棄する。エンドポイント A は、ユーザ B への呼完了が依然として必要とされている場合、インプリメントオプションとして、SS-CCBS の再インボケーションを許可できる。
- l ユーザ B の監視を再開し、失敗の理由をユーザ A に表示し、その CC 要求は保持される。この場合、ユーザ B への呼完了がもはや必要とされていないならば、ユーザ A はユーザオプションとして CC 要求の取消を要求できる。

5.2.1.4. 重複 SS-CCBS 要求

ユーザ A がユーザ B の SS-CCBS をすでに要求し、リコールを待っているならば、ユーザ A からユーザ B への SS-CCBS をインボークするためのあらゆるその後の要求は、重複要求としてユーザ B のエンドポイントが要求を拒否する原因としなければならない。

5.2.1.5. その他の失敗状況

サービスに対する異常な要求は、ユーザ A のエンドポイントにより自動的に取消されなければならない。またユーザ A は以下のことを通知されなければならない。

- l SS-CCBS サービス持続タイマが満了した後もユーザ B がビジーであるか、また
- l リコールタイマが終了する以前に、ユーザ A が CC リコールを受け取っていないか。

5.2.2. 取消

ユーザ A への CC 要求がない、またはその要求が、存在しない特定の CC 要求を取り消すものである場合、取消の要求は拒否されなければならない。

5.3. その他の付加サービスとの相互作用

5.3.1. 無応答時の呼完了 (SS-CCNR)

ユーザ A がユーザ B で起動される SS-CCNR を受け取り、ユーザ A がユーザ B の SS-CCBS を要求する場合、5.2.1.4 項と同じく、この要求は重複 SS-CCBS 要求として扱われなければならない。

注：CC リコールが先の SS-CCNR 要求に関係して開始される以前に、ユーザ B がビジーであるとき（ユーザ A による SS-CCBS のインボケーションの前提条件）、動作中の SS-CCNR 要求は、ユーザ A をリコールするためにユーザ B が空きになるのを待ったまま、事実上、SS-CCBS 要求となっている。このとき SS-CCBS 要求をユーザ A から受け取ったならば、ユーザ B に関して、これはそれゆえに事実上、重複 SS-CCBS 要求であり、ユーザ B のエンドポイントにはそのように扱われる。

5.3.2. コールトランスファ (SS-CT)

相互作用なし。

5.3.3. 無条件着信転送 (SS-CFU)

a. ユーザ A が SS-CCBS を要求する以前に、ユーザ B により起動される SS-CFU

ユーザ B への呼が、SS-CFU によりユーザ C に転送され、かつユーザ C がビジーである場合、ユーザ A からの SS-CCBS 要求は、転送先であるユーザ C に適用されなければならない。

b. ユーザ A が SS-CCBS を要求した後に、ユーザ B により起動される SS-CFU

ユーザ A が SS-CCBS を要求したのちにユーザ B が SS-CFU を起動した場合で、CC リコールをユーザ A が受け取っていない間は、SS-CCBS 要求はユーザ B に適用され続けなければならない。

c. ユーザ A による SS-CFU の起動

SS-CFU が起動されている間に、ユーザ A が SS-CCBS をインボークする場合、またはユーザ A が SS-CCBS をインボークし、続いて SS-CFU を起動する場合、これはユーザ A への CC リコールの提供に影響を及ぼしてはならない。

5.3.4. ビジー時着信転送 (SS-CFB)

a. ユーザ A が SS-CCBS を要求する以前に、ユーザ B により起動される SS-CFB

ユーザ A からユーザ B への呼が SS-CFB によりユーザ C に転送され、かつユーザ C がビジーである場合、ユーザ A による SS-CCBS 要求はユーザ B、または転送先であるユーザ C に適用されなければならない。

b. ユーザ A が SS-CCBS を要求した後に、ユーザ B により起動される SS-CFB

ユーザ A が SS-CCBS を要求した後にユーザ B が SS-CFB を起動する場合で、CC リコールをユーザ A が受け取っていない間は、SS-CCBS はユーザ B に適用され続けなければならない。

c. ユーザ A による SS-CFB の起動

SS-CFB が起動している間に、ユーザ A が SS-CCBS をインボークする場合、またはユーザ A が SS-CCBS をインボークし、続いて SS-CFB を起動する場合、ユーザ A への CC リコールの提供に影響を及ぼしてはならない。

5.3.5. 無応答時着信転送 (SS-CFNR)

- a. ユーザ A が SS-CCBS を要求する以前に、ユーザ B により起動される SS-CFNR

ユーザ A からユーザ B への呼が SS-CFNR によりユーザ C に転送され、かつユーザ C がビジーである場合、ユーザ A による SS-CCBS 要求は、転送先であるユーザ C に適用されなければならない。

- b. ユーザ A が SS-CCBS を要求した後に、ユーザ B により起動される SS-CFNR

ユーザ A が SS-CCBS を要求した後にユーザ B が SS-CFNR を起動した場合で、CC リコールをユーザ A が受け取っていない間は、SS-CCBS 要求はユーザ B に適用され続けなければならない。

注：このような場合、SS-CCBS がうまく完了したことにより生じる呼は、応答がない場合 SS-CFNR に従ってもよい。

- c. ユーザ A による SS-CFNR の起動

SS-CFNR が起動されている間にユーザ A が SS-CCBS をインボークした場合、またはユーザ A が SS-CCBS をインボークし、続いて SS-CFNR を起動する場合、これはユーザ A への CC リコールの提供に影響を及ぼしてはならない。

5.3.6. 呼保留

相互作用なし。

5.3.7. コールパーク/コールピックアップ

同じピックアップグループであっても、その他のユーザは CC 呼をピックアップできない。

5.3.8. コールウェイトニング

待機中の呼がビジーであるとみなされないならば、SS-CCBS を起こすことができない。

5.3.9. メッセージウェイトニング通知

相互作用なし。

5.3.10. 名前表示

相互作用なし。

5.3.11. コールリネージ

可能ならば、すべての呼と CC 要求に関係したすべてのシグナリング接続は、異なるグローバルコール ID ではなくて、同じスレッド ID を用いなければならない。スレッド ID は、以下のことを結合するのに用いられる。

- l ユーザ B がビジーであるために失敗した発信呼の試み
- l SS-CCBS 要求
- l CC リコール
- l CC 呼

発信呼の試みがスレッド ID を含んでいなかった場合、その呼識別子は SS-CCBS 要求、CC リコール、そして CC 呼のスレッド ID として使用できる。

6. SS-CCNR サービスの説明

無応答時の呼完了 (SS-CCNR) は、発信ユーザ A に与えられた付加サービスである。着信ユーザ B が無

応答であるとき、ユーザ A はユーザ B のエンドポイントにユーザ B を監視させることや、次のアクティビティ期間の後にユーザ B が空きになったときにユーザ A に通知させることを許可する。その通知に対するユーザ A の応答で、ユーザ A のエンドポイントはユーザ B への呼の完了を試みなければならない。

注1：次のアクティビティ期間を構成するユーザアクティビティは、実装固有かつ本標準の範囲外ある。

注2：ユーザ B が有効でない(たとえば登録されていない)場合でも SS-CCNR は適用してもよい。このとき、ユーザ B の代わりに SS-CCNR を行うゲートキーパー、もしくはプロキシが必要となる。

6.1. 一般的な手順

6.1.1. 開始 / 停止 / 登録 / 問い合わせ

SS-CCNR は永久に起動している。

6.1.2. インボケーションおよび処理

ユーザ A からの呼が着ユーザ B の応答がないために失敗したとき、ユーザ A は SS-CCNR を要求することができなければならない。ユーザ B が呼出ししている間でも、ユーザ A は SS-CCNR を要求することができなければならない。

SS-CCNR の要求を受けるとすぐに、ユーザ B のエンドポイントはサービスを開始できるかどうかを調査しなければならず、もしできるのならばユーザ A に確認を送り、ユーザ B の監視を始めなければならない。その確認を受けるとすぐに、ユーザ A のエンドポイントは SS-CCNR サービス持続タイマを開始しなければならない。確認の意味は、SS-CCNR サービス持続タイマの時間内で、次のアクティビティ期間の後にユーザ B が空きになったとき、ユーザ A が CC リコールの受け取りを求められることができるということである。

注1：その他のユーザから呼完了要求の結果としてユーザ B がすでに監視されているという事実は、ユーザ A からの拒否理由にはならない。同じユーザ B への複数の要求を処理することは、インプリメントマタ - である。それは一般的に、日付順、または優先順に整列されたキューの一種を伴う

注2：ユーザ A がその他のユーザに対してすでに呼完了をインボークしたという事実は、ユーザ A からの更なる呼完了の拒否理由を必要としない。同じユーザ A により複数の要求を処理することは、インプリメントマタ - である。

注3：SS-CCNR はユーザ B が呼出し中であるという通知の後、インボークされうる。SS-CCNR 持続タイマの推奨値は、1-1440 分 (すなわち 24 時間) の範囲である。適当な初期値は 60 分にするとよい。

注4：特定の基本サービスと提携された呼での SS-CCNR の選択処理は、今後の検討課題である。

SS-CCNR 要求を受け取ったことを知らされた後でも、ユーザ A は他の呼を受け取り、起動できなくてはならない。

ユーザ A のエンドポイントは、ユーザ A がインボークした未解決の CC 要求のリストを要求する能力を、ユーザ A に提供してもよい。

ユーザ B の監視が、ユーザ B がアクティビティ期間の後に空きになったことを表示したとき、ユーザ B のエンドポイントは CC リコールを提供しなくてはならない。ユーザ A のエンドポイントはユーザ A に CC リコールを通知し、リコールタイマを開始しなくてはならない。

ユーザ A が CC リコールを受けとったならば、ユーザ A のエンドポイントはユーザ A とユーザ B との間
の呼の完了を試みなくてはならない。ユーザ B への呼が成功し、呼出フェーズに入る、または応答された場
合、SS-CCNR は完了とみなされなければならない。

6.1.3. 取消

5.1.3 項が適用されなくてはならない。

6.2. 例外的な手順

6.2.1. インボケーションおよび処理

6.2.1.1. SS-CCNR サービス要求の拒否

“SS-CCBS”を“SS-CCNR”に置き換えて、5.2.1.1 項を適用しなければならない。

6.2.1.2. CC リコールでユーザ A がビジー

アクティビティ期間の後にユーザ B が空きとなったときユーザ A がビジーであるとわかった場合、ユー
ザ A のエンドポイントは CC 要求をすぐに保留するか、もしくはユーザ A に CC リコールの通知がされ、CC
リコールが実行される前（すなわちユーザ B に新たな呼を起動する前）ビジーでなくなるようにユーザ A
のために短期間を待機した後、依然としてユーザ A がビジーである場合に保留する。

注：このような通知を受けるとすぐに、ユーザ A は以下のことが可能となる

- l 通知の無視、それにより CC 要求は保留される
- l SS-CCBS 要求の取消、また
- l リソースを空ける（たとえば既存呼を処理することによる）、それにより CC リコールは続行で
きる

SS-CC を保留した後にユーザ A が空きとなると、ユーザ A のエンドポイントは SS-CC を再開しなけれ
ばならず、それによりユーザ B は再び監視される。もしくは 6.1.2 項で述べたように呼の完了を試みよう
としなければならない。

6.2.1.3. CC リコールが成功した後、ユーザ B がビジーとなる

ユーザ B が、ユーザ A の成功した CC リコールにより生じる呼によりビジーである場合、ユーザ B のエ
ンドポイントは以下のどちらかをしなければならない

- l ユーザ A に失敗の理由を表示し、CC 要求が取消されていることを表示して SS-CCNR を放棄す
る。このような場合ユーザ A のエンドポイントは、ユーザ B への呼完了が今もなお必要とされ
ているならば、SS-CCBS をインボークする可能性をユーザ A に与えなければならない。
- l ユーザ A に失敗の理由を表示し、SS-CCBS がインボークされていることを表示して、ユーザ B
を監視しながら自動的に SS-CCBS をインボークする。このような場合、ユーザ B への呼完了が
もはや要求されていないのならば、ユーザ A は SS-CCBS 要求の取消を要求してもよい。

6.2.1.4. 重複 SS-CCNR 要求

“SS-CCBS”を“SS-CCNR”に置き換えて、5.2.1.4 項を適用しなければならない。

6.2.1.5. その他の失敗状況

サービスへの異常な要求は、自動的にユーザ A のエンドポイントに取消され、ユーザ A は以下のことを通知されなければならない

- I SS-CCNR サービス持続タイマが満了する前に、ユーザ B がアクティビティ期間を持っていなかったかどうか
- I (ユーザ B によるアクティビティ期間の後) SS-CCNR サービス持続タイマが満了するときに、依然としてユーザ B がビジーであるかどうか
- I リコールタイマが終了する前にユーザ A が CC リコールを受け取っていないかどうか

6.2.2. 取消

5.2.2 項を適用しなければならない。

6.3. 他の付加サービスとの相互作用

6.3.1. ビジー加入者への呼完了 (SS-CCBS)

5.3.1 項を適用しなければならない。

6.3.2. コールトランスファ (SS-CT)

相互作用なし。

6.3.3. 無条件着信転送 (SS-CFU)

a. ユーザ A が SS-CCNR を要求する前に、ユーザ B により起動される SS-CFU

ユーザ B への呼が SS-CFU によりユーザ C に転送され、かつユーザ C の応答がない場合、ユーザ A からの SS-CCNR 要求は転送先であるユーザ C に適用されなければならない。

b. ユーザ A が SS-CCNR を要求した後、ユーザ B により起動される SS-CFU

ユーザ A が SS-CCNR を要求した後ユーザ B が SS-CFU を起動した場合で、ユーザ A が CC リコールを受け取っていない間は、SS-CCNR 要求はユーザ B に適用され続けなければならない。

c. ユーザ A による SS-CFU の起動

SS-CFU が起動されているときにユーザ A が SS-CCNR をインボークする場合、またはユーザ A が SS-CCNR をインボークし、続いて SS-CFU を起動する場合、ユーザ A への CC リコールの提供に影響を及ぼしてはならない。

6.3.4. ビジー時着信転送 (SS-CFB)

a. ユーザ A が SS-CCNR を要求する前に、ユーザ B により起動される SS-CFB

ユーザ A からユーザ B への呼が SS-CFB によりユーザ C に転送され、かつユーザ C の応答がない場合、ユーザ A による SS-CCNR 要求は転送先であるユーザ C に適用されなければならない。

b. ユーザ A が SS-CCNR を要求した後に、ユーザ B により起動される SS-CFB

ユーザ A が SS-CCNR を要求した後にユーザ B が SS-CFB を起動した場合で、ユーザ A が CC リコールを受け取っていない間は、SS-CCNR 要求はユーザ B に適用され続けなければならない。

c. ユーザ A による SS-CFB の起動

SS-CFB が起動されているときにユーザ A が SS-CCNR をインボークする場合、またはユーザ A が SS-CCNR をインボークし、続いて SS-CFB を起動した場合、ユーザ A への CC リコールの提供に影響を及ぼしてはならない。

6.3.5. 無応答時着信転送 (SS-CFNR)

a. ユーザ A が SS-CCNR を要求する前に、ユーザ B により起動される SS-CFNR

ユーザ A からユーザ B への呼が SS-CFNR によりユーザ C に転送され、かつユーザ C の応答がない場合、ユーザ A による SS-CCNR 要求はユーザ B、または転送先であるユーザ C に適用されなければならない。

b. ユーザ A が SS-CCNR を要求した後に、ユーザ B により起動される SS-CFNR

ユーザ A が SS-CCNR を要求した後にユーザ B が SS-CFNR を起動した場合で、ユーザ A が CC リコールを受け取っていない間は、SS-CCNR 要求はユーザ B に適用され続けなければならない。

c. ユーザ A による SS-CFNR の起動

SS-CFNR が起動しているときにユーザ A が SS-CCNR をインボークした場合、またはユーザ A が SS-CCNR をインボークして、続いて SS-CFNR を起動した場合、ユーザ A への CC リコールの提供に影響を及ぼしてはならない。

6.3.6. 呼保留

相互作用なし。

6.3.7. コールパーク / コールピックアップ

5.3.7 項を適用しなければならない。

6.3.8. コールウェイティング

SS-CCNR のインボケーションは、相当する呼のコールウェイティングを終了しなければならない。

6.3.9. メッセージ待機通知

相互作用なし。

6.3.10. 名前表示

相互作用なし。

6.3.11. コールリンケージ

可能ならば、すべての呼と CC 要求に係るすべてのシグナリング接続は異なるグローバルコール ID ではなく、同じスレッド ID を使用しなければならない。したがってスレッド ID は以下のことを結びつけるのに使用される

- I ユーザ B の応答がない発信呼の試み
- I SS-CCNR 要求
- I CC リコール
- I CC 呼

発信呼の試みがスレッド ID を含んでいなかった場合、その呼識別子は SS-CCNR 要求、CC リコール、そして CC 呼のスレッド ID として使用できる。

7. メッセージと情報要素

12 章で定義されるオペレーション APDU は、標準 JT- H450.1 で明記されるように、ユーザ・ユーザ情報要素に含まれる H.450.1 付加サービス APDU で伝達しなければならない。

12章の抽象構文表記法1 (ASN.1) で定義されるオペレーションは適用されなければならない。

12章で定義されるオペレーションの invoke APDU を伝達するときは、NFE の *destinationEntity* データ要素は *endpoint* 値を含まなければならない。

12章で定義されるオペレーションの invoke APDU を伝達するとき、Interpretation APDU は、省略するかもしくは *rejectAnyUnrecognizedInvokePdu* 値を含まなければならない。

8. 手順

以下の手順は呼依存シグナリングと呼非依存シグナリングとの組合せである。後者は標準 JT-H450.1 の 6.2 節で定義されるように呼非依存手順を使用し、それは付加サービス制御のための呼非依存シグナリングコネクションを提供する。SS-CC のために、呼非依存シグナリングコネクションは、コネクション解放オプションを適用しないならば、SS-CC の完了または取消まで維持しなければならない。その場合には、呼非依存シグナリングコネクションは SS-CC シグナリングのアイドルフェーズの間解放し、もし後続の SS-CC 制御情報が送信されるなら再確立しなければならない。一度選択されると、その方法 (コネクション解放かコネクション維持) は、相互接続上の理由により要求される場合を除き、この CC 要求の存続期間中ずっと適用しなければならない。

全ての SS-CC 制御情報はユーザ A のエンドポイントとユーザ B のエンドポイントとの間で交換する。

ユーザ A が CC リコールに応答した後、ユーザ B が再びビジーになったとき、2つの可能な動作がある：

- a. サービス維持方式 - CC 要求はユーザ A とユーザ B のエンドポイントで有効で残り、ユーザ B のエンドポイントは再びユーザ B の監視を開始する。
- b. サービス取消方式 - CC 要求は、ユーザ A とユーザ B のエンドポイントで取消される。

8.1. ユーザ A のエンドポイントの動作

8.1.1. 正常手順

8.1.1.1. CCBS インボケーション

被呼エンドポイントから解放完了メッセージで *inConf* の *ReleaseCompleteReason* または理由表示値 #17 “user busy” の Cause IE を受信するとき “Busy” 状態は発生する。

ユーザ A が SS-CCBS を要求しユーザ A のエンドポイントに受け付けられれば、ユーザ A のエンドポイントは、8.1.1.3 項に従ってユーザ B のエンドポイントに *ccbsRequest* invoke APDU を送信しなければならない。

注：SS-CCBS は呼の試みがユーザ B でビジー状態に遭遇した後に要求できる。

オリジナル呼の解放はこの標準の範囲外である。

ccbsRequest return result APDU の受信で、ユーザ A のエンドポイントは、SS-CCBS (11.4.1 項参照) にふさわしい値でサービス持続タイマ T2 をスタートし、ユーザ B がビジーでなくなった表示を待たなければならない。

8.1.1.2. CCNR インボケーション

ユーザ A が SS-CCNR を要求しユーザ A のエンドポイントに受け付けられれば、ユーザ A のエンドポイントは、8.1.1.3 項に従ってユーザ B のエンドポイントに *ccnrRequest* invoke APDU を送信しなければならない。

注：SS-CCNR はユーザ B への呼が無応答となった後に要求できる。

オリジナル呼の解放はこの標準の範囲外である。

ccnrRequest return result APDU の受信で、ユーザ A のエンドポイントは、SS-CCNR (11.4.1 項参照) にふさわしい値でサービス持続タイマ T2 をスタートし、ユーザ B が以降の動作期間の後にビジーでなくなった表示を待たなければならない。

8.1.1.3. SS-CC インボケーション - 詳細手順

ccbsRequest または *ccnrRequest* invoke APDU は新しい呼被依存シグナリングコネクションの呼設定メッセージでユーザ B のエンドポイントに送り、標準 JT-H450.1 に明記された手順に従って確立しなければならない。

ccbsRequest または *ccnrRequest* invoke APDU のアーギュメントには以下の情報を含まなければならない：

- I オリジナル呼からの基本呼情報：
 - Ø *numberA* 要素の中のユーザ A のアドレスまたは利用できないか制限されている表示；
 - Ø *numberB* 要素の中のユーザ B のアドレス；
 - Ø *service* 要素の中の意図された基本サービスタイプ；
 - Ø *ccIdentifier* 要素(推奨；もし提供するなら、値は前の不成功であった基本呼の試みの CallIdentifier 値を反映しなければならない)；
- I *can-retain-service* 要素；その値は、エンドポイント A がサービス維持方式を使用できるなら *TRUE* に、その他は *FALSE* に設定しなければならない；
- I オプションの *retain-sig-connection* 要素は、エンドポイント A がこの CC 要求の存続期間中ずっとシグナリングコネクションを維持するよう要求するなら値を *TRUE* にし、エンドポイント A が SS-CC シグナリングのアイドルフェーズの間シグナリングコネクションを解放するよう要求するなら値を *FALSE* にする。もし含まれなければ、エンドポイント A は意志を持たない。

ccbsRequest/ccnrRequest invoke APDU を送信した後、ユーザ A のエンドポイントはタイマ T1 をスタートし CC-Wait-Ack 状態に入らなければならない。

応答または解放完了メッセージで *ccbsRequest* または *ccnrRequest* return result APDU を受信すると、ユーザ A のエンドポイントは、タイマ T1 を停止し、CC 要求に関連するデータを蓄積して、CC-Invoked-User-A-RET 状態(応答メッセージで APDU を受信したなら)または CC-Invoked-User-A-RLS 状態(解放完了メッセージで APDU を受信したなら)に入らなければならない。もし結果が *TRUE* の値である *retain-service* 要素を含み、対応する invoke APDU のアーギュメントで値が *TRUE* の *can-retain-service* 要素が送信されていれば、ユーザ A のエンドポイントはサービス維持方式が使用された事実を記録し、その他の場合は、サービス取消方式が使用された事実を記録しなければならない。

8.1.1.4. ユーザ A がビジーでないときの CC 呼

CC-Invoked-User-A-RET 状態の間に呼非依存シグナリングコネクションに於いてファシリティメッセージで *ccExecPossible* invoke APDU を受信し、ユーザ A がビジーでなければ、ユーザ A のエンドポイントは、ユーザ A に CC リコールを表示し、リコールタイマ T3 をスタートして CC-Wait-User-A-Answer 状態に入らなければならない。

呼非依存シグナリングコネクションのための呼設定メッセージで *ccExecPossible* invoke APDU を受信し、ユーザ A がビジーでなければ、ユーザ A のエンドポイントは、invoke APDU で受信した情報とローカルで蓄積した情報とを比較することによって、CC-Invoked-User-A-RLS 状態において、APDU と CC 要求とを関連付けるようにしなければならない。そして成功すれば、ユーザ A に CC リコールを表示し、リコールタイマ T3 をスタートして CC-Wait-User-A-Answer 状態に入らなければならない。その呼非依存シグナリングコネクションは解放しなければならない。

CC リコールがタイマ T3 が満了する前にユーザ A に受け入れられると、ユーザ A のエンドポイントは、ユーザ B のエンドポイントに向けて、呼設定メッセージで *ccRingout* invoke APDU を送信することにより、CC 呼を起動し、タイマ T3 を停止して、CC-Ringout 状態に入らなければならない。呼設定メッセージは、オリジナルの不成功の呼の試みと等しい情報を含まなければならないし、*ccRingout* invoke APDU は、そのアークメントの中に、以前の *ccbsRequest/ccnrRequest* invoke APDU に含まれていた値を示す *ccIdentifier* 要素を含むべきである。

CC-Ringout 状態で呼出または応答メッセージを受信すれば、ユーザ A のエンドポイントは、サービス持続タイマ T2 を停止し、CC 要求を消去して、CC-Idle 状態に戻らなければならない。SS-CC のための呼非依存シグナリングコネクションは、まだ存在すれば解放し、CC 呼は基本呼手順に従って継続しなければならない。

8.1.1.5. ユーザ A がビジーのときの CC 呼

中断手順：

CC-Invoked-User-A-RET 状態の間に呼非依存シグナリングコネクションに於いてファシリティメッセージで *ccExecPossible* invoke APDU を受信し、ユーザ A がビジーであれば、ユーザ A のエンドポイントは、呼非依存シグナリングコネクションにおいて、ファシリティメッセージでユーザ B のエンドポイントに *ccSuspend* invoke APDU を送信し、ユーザ A がビジーでなくなるための監視を開始して、CC-Suspended-User-A 状態に入らなければならない。

呼非依存シグナリングコネクションのための呼設定メッセージで *ccExecPossible* invoke APDU を受信し、ユーザ A がビジーであれば、ユーザ A のエンドポイントは、invoke APDU で受信した情報とローカルで蓄積した情報とを比較することによって、CC-Invoked-User-A-RLS 状態において、APDU と CC 要求とを関連付けるようにしなければならない。そして成功すれば、ユーザ B のエンドポイントに *ccSuspend* invoke APDU を送信し、ユーザ A がビジーでなくなるための監視を開始して、CC-Suspended-User-A 状態に入らなければならない。*ccSuspend* invoke APDU は、シグナリングコネクションが維持されることをインターワーキングが特に要求する場合は、応答メッセージで送信してもいいが、そうでない限り、呼非依存シグナリングコネクションに於いて、解放完了メッセージで送信しなければならない。

再開手順：

ユーザ A のために CC-Suspended-User-A 状態で CC 要求が存在し、ユーザ A がビジーでなくなれば、ユーザ A のエンドポイントは、現在の呼非依存シグナリングコネクションにおけるファシリティメッセージか、または、そのようなコネクションが存在しなければ、新しい呼非依存シグナリングコネクションのための呼設定メッセージで、*ccResume* invoke APDU を送信し、CC-Invoked-User-A-RET 状態に入って、ユーザ B がビジーでないというもうひとつの表示を待たなければならない。呼設定メッセージで送信されるならば、invoke APDU はこの CC 要求のために全てのローカルで蓄積した情報を含まなければならない。

あるいは、ユーザ A のエンドポイントは、8.1.1.4.項で指定される CC 呼手順を継続してもよい。

8.1.1.6. CCBS / CCNR 取消

a. ユーザ A のエンドポイントにより開始される取消

CC 要求を取消するために、ユーザ A のエンドポイントは、現在の呼非依存シグナリングコネクションを解放する解放完了メッセージか、別の方法では、新しい呼非依存シグナリングコネクションのための呼設定メッセージで、ユーザ B のエンドポイントに *ccCancel* invoke APDU を送信しなければならない。解放完了メッセージは、Cause IE #16 “*normal call clearing*” または *undefinedReason* の *ReleaseCompleteReason* を含まなければならない。呼設定メッセージで送信されるならば、invoke APDU はこの CC 要求のために全てのローカルで蓄積した情報を含まなければならない。ユーザ A のエンドポイントは、また、Cause IE #16 “*normal call clearing*” または *undefinedReason* の *ReleaseCompleteReason* を含む解放完了メッセージを送信することにより（既に開始されていれば）CC 呼を解放し、CC 要求のために蓄積された全てのデータを削除して、動作しているタイマを停止し、CC-Idle 状態に戻らなければならない。もし、ユーザの要求によるか、または、自動的に SS-CC を再インボークするユーザ A のエンドポイントにより、取消が開始されなければ、能力があれば、ユーザ A にサービスの失敗を告げなければならない。

b. ユーザ B のエンドポイントにより開始される取消

現在の呼非依存シグナリングコネクションのための解放完了メッセージで、ユーザ B のエンドポイントから *ccCancel* invoke APDU を受信すると、ユーザ A のエンドポイントは、関連付けられた CC 要求のために蓄積された全てのデータを削除し、まだ動作しているタイマを停止して、（既に開始されていれば）CC 呼を解放し、そしてもし能力があるならばユーザ A に取消を告げて、CC-Idle 状態に戻らなければならない。

新しい呼非依存シグナリングコネクションのための呼設定メッセージで、ユーザ B のエンドポイントから *ccCancel* invoke APDU を受信すると、ユーザ A のエンドポイントは、invoke APDU で受信した情報とローカルで蓄積した情報とを比較することによって、APDU とアクティブな CC 要求とを関連付けるようにしなければならない。成功すれば、関連付けられた CC 要求のために蓄積された全てのデータを削除し、まだ動作しているタイマを停止して、（既に開始されていれば）CC 呼を解放し、そしてもし能力があるならばユーザ A に取消を告げて、Cause IE #16 “*normal call clearing*” または *undefinedReason* の *ReleaseCompleteReason* を含む解放完了メッセージを送り返し、CC-Idle 状態に戻らなければならない。

8.1.2. 例外手順

8.1.2.1. CCBS / CCNR インボケーション

呼非依存シグナリングコネクションの確立が失敗したなら、または、*ccbsRequest* または *ccnrRequest* invoke APDU を送信した後に、タイマ T1 が満了する前に、ユーザ B のエンドポイントから応答を受信しなかったならば、または、応答が *return error* APDU または *reject* APDU であれば、ユーザ A に失敗表示をしなければ

ならない。また、ユーザ A のエンドポイントは CC-Idle 状態に戻らなければならない。呼非依存シグナリングコネクションがまだ存在すれば解放しなければならない。タイマ T1 がまだ動作していれば停止しなければならない。

8.1.2.2. サービス持続タイマ満了

CC-Invoked-User-A-RET、CC-Invoked-User-A-RLS、CC-Wait-User-A-Answer または CC-Suspended-User-A 状態の間にサービス持続タイマ T2 が満了すれば、ユーザ A のエンドポイントは、8.1.1.6.項に記述されている手順を使用して、CC 要求を取消さなければならない。

CC-Ringout 状態の間にサービス持続タイマ T2 が満了すれば、ユーザ A のエンドポイントは、ユーザ B のエンドポイントからの呼出または応答メッセージの受信を許すために、どの動作をすることも延ばさなければならない。このように、T2 の満了は無視され、8.1.1.4.項の手順を適用しなければならない。

8.1.2.3. リコールタイマ満了

リコールタイマ T3 が満了する前にユーザ A が CC リコールを受け付けなければ、ユーザ A のエンドポイントは、8.1.1.6.項に記述されている手順を使用して、CC 要求を取消さなければならない。

8.1.2.4. CC 呼プレゼンテーションの失敗

エラー値が *remoteUserBusyAgain* の *ccRingout* return error APDU を持つ解放完了メッセージを受信すると、ユーザ A のエンドポイントは：

- I サービス維持オプションが有効であるならば、CC-Invoked-User-A-RET 状態（コネクション維持の場合）または CC-Invoked-User-A-RLS 状態（コネクション解放の場合）に戻らなければならない。
- I サービス取消オプションが有効であるならば、タイマ T2 を停止し、Cause IE #16 “normal call clearing” または *undefinedReason* の *ReleaseCompleteReason* を含む解放完了メッセージを送信することにより、まだ存在するならば、呼非依存シグナリングコネクションを解放しなければならない。ユーザ A のエンドポイントはそれから：
 - a. ユーザ A に失敗を表示し、CC 要求を削除して、CC-Idle 状態に戻るか、または
 - b. 8.1.1.1.項に従って自動的に SS-CCBS をインボークしなければならない。

8.1.2.5. 呼非依存シグナリングコネクションの解放

CC-Wait-User-A-Answer、CC-Ringout または CC-Suspended-User-A 状態において CC 要求と関連付けられた呼非依存シグナリングコネクションが、*ccCancel* invoke APDU を送信または受信することなしに、なんらかの理由により解放されれば、ユーザ A のエンドポイントは現在の状態に留まらなければならない。

CC-Invoked-User-A-RET 状態の間に呼非依存シグナリングコネクションにおいて *ccCancel* invoke APDU を持たない解放完了メッセージを受信すると、ユーザ A のエンドポイントは CC-Invoked-User-A-RLS 状態に入らなければならない。

8.1.2.6. 予期しない APDU

CC-Invoked-User-A-RLS 状態において、どの CC 要求にも合致しない *ccExecPossible* invoke APDU を含む呼非依存シグナリングコネクションのための呼設定メッセージを受信すれば、*ccCancel* invoke APDU と *destinationReject* である *ReleaseCompleteReason* を持つ解放完了メッセージを返送しなければならない。

どの未解決の CC 要求にも合致しない *ccCancel* invoke APDU を含む呼非依存シグナリングコネクションのための呼設定メッセージを受信すれば、*destinationReject* である *ReleaseCompleteReason* を持つ解放完了メッセージを返さなければならない。

8.2. ユーザ B のエンドポイントでの動作

8.2.1. 正常手順

8.2.1.1. CCBS インボケーション

呼設定メッセージで *ccbsRequest* invoke APDU を受信すれば、ユーザ B のエンドポイントは、CC 要求を受け入れるならば、8.2.1.3.項に従って要求を処理し、ユーザ B がビジーでなくなるのを監視し始めなければならない。

ユーザ B が既にビジーでなければ、*ccbsRequest* return result APDU を送信した後、8.2.1.4.項の手順をすぐに開始する結果になるであろう。

8.2.1.2. CCNR インボケーション

呼設定メッセージで *ccnrRequest* invoke APDU を受信すれば、ユーザ B のエンドポイントは、CC 要求を受け入れるならば、8.2.1.3.項に従って要求を処理し、ユーザ B が以降の動作期間の後にビジーでなくなるのを監視し始めなければならない。

8.2.1.3. SS-CC インボケーション - 詳細手順

CC 要求が受け入れられるならば、ユーザ B のエンドポイントは、基本呼情報と（もし在れば）*ccbsRequest/ccnrRequest* invoke APDU のアークギュメントに含まれる *ccIdentifier* 値を蓄積しなければならない。

承認として、ユーザ B のエンドポイントは、呼非依存シグナリングコネクションにおいて、応答または解放完了メッセージで *ccbsRequest/ccnrRequest* return result APDU をユーザ A のエンドポイントに送信しなければならない。特別なインターワーキングの要求により禁止されずに、対応する invoke APDU が値 TRUE の *retain-sig-connection* 要素を含んでいるならば、応答メッセージを選択し、対応する invoke APDU が値 FALSE の *retain-sig-connection* 要素を含んでいるならば、解放完了メッセージを選択しなければならない。*retain-sig-connection* 要素がない場合は、ユーザ B のエンドポイントは、どちらのメッセージでも選択できる。

対応する invoke APDU に値 TRUE の *can-retain-service* 要素があり、ユーザ B のエンドポイントがサービス維持方式を選択すれば、値 TRUE の *retain-service* 要素を結果に含めなければならない。その他の全ての場合は、この要素は値 FALSE を持たなければならない。

ユーザ B のエンドポイントは、それから、CC-Invoked-User-B 状態に入り、ユーザ B がビジーでなくなるのを（CCBS の場合）または、動作期間の後にビジーでなくなるのを（CCNR の場合）監視し始めなければならない。

8.2.1.4. ユーザ B がビジーでない表示

（SS-CCNR の場合には、SS-CC インボケーションに続いて起こる動作の期間の後、）ユーザ B がビジーでなくなるとすぐに、ユーザ B のエンドポイントは、現在の呼非依存シグナリングコネクションにおいてファシリティメッセージで、または、そのようなコネクションが存在しなければ、新しい呼非依存シグナリングコネクションのための呼設定メッセージで *ccExecPossible* invoke APDU を送信し、

CC-Await-Call-Completion 状態に入って、着信の CC 呼を待たなければならない。呼設定メッセージで送信すれば、invoke APDU は、この CC 要求のためにローカルで蓄積した全ての情報を含まなければならない。

注：それは CC 呼の非受信から守るためのエンドポイント B の責任である。これをどう達成するかはこの標準の範囲外である。

8.2.1.5. CC 呼処理

ccRingout invoke APDU を持つ呼設定メッセージを受信すれば、ユーザ B のエンドポイントは、CC-Await-Call-Completion 状態で着信の CC 呼を CC 要求と関連付けし、もし成功してユーザ B がまだビジーでないならば、CC 呼をユーザ B に届けて、CC-Wait-User-B-Alert 状態に入らなければならない。

その関連付けは、ローカルで蓄積された基本呼情報と *ccIdentifier* 値、それと、呼設定メッセージで受信された基本呼情報と *ccRingout* invoke APDU に含まれている *ccIdentifier* 値を比較することにより達成する。利用できるならば、要素 *ccIdentifier* は、この比較のための最良の機構である。別の方法で、ローカルに蓄積した要素が、呼設定メッセージの対応する情報要素と対等であるならば、マッチングが存在することを考えなければならない。対応する要素が蓄積されていない呼設定メッセージの情報要素は（逆もまた同じ）、比較の間無視しなければならない。

CC-Wait-User-B-Alert 状態で、呼出または応答メッセージがユーザ A のエンドポイントに送信されれば、ユーザ B のエンドポイントは、それぞれの CC 要求を削除し、CC-Idle 状態に入らなければならない。関連付けられた呼非依存シグナリングコネクションは、まだ存在すれば解放し、CC 呼は、確立した基本呼ごとに継続しなければならない。

8.2.1.6. CCBS / CCNR 中断 / 再開

CC-Await-Call-Completion 状態の間に、現在の呼非依存シグナリングコネクションにおいて、ファシリティ、応答または解放完了メッセージで *ccSuspend* invoke APDU を受信すれば、ユーザ B のエンドポイントは、CC-Await-Call-Completion 状態に留まらなければならない。

注：その他の動作はこの標準の範囲外である。

CC-Await-Call-Completion 状態の間に、現在の呼非依存シグナリングコネクションにおいて、ファシリティメッセージで *ccResume* invoke APDU を受信すれば、ユーザ B のエンドポイントは、CC-Invoked-User-B 状態に入り、ユーザ B の監視を再開しなければならない。

呼非依存シグナリングコネクションのための呼設定メッセージで *ccResume* invoke APDU を受信すれば、ユーザ B のエンドポイントは、CC-Await-Call-Completion 状態において、invoke APDU で受信した情報とローカルに蓄積した情報とを比較することにより、CC 要求と関連付けようとし、成功すれば、CC-Invoked-User-B 状態に入って、ユーザ B の監視を再開しなければならない。

コネクション解放オプションが適用されるならば、シグナリングコネクションを解放しなければならない。

ユーザ B が既にビジーでないならば、8.2.1.4.項の手順をすぐに開始する結果になるであろう。

8.2.1.7. CCBS / CCNR 取消

a. ユーザ A のエンドポイントにより開始される取消

呼非依存シグナリングコネクションの解放完了メッセージで *ccCancel* invoke APDU を受信すれば、ユーザ B のエンドポイントは、関連付けられた CC 要求のために蓄積された情報を削除し、シグナリングコネクションを解放して、CC-Idle 状態に入らなければならない。

呼非依存シグナリングコネクションの呼設定メッセージで *ccCancel* invoke APDU を受信すれば、ユーザ B のエンドポイントは、それを CC 要求と関連付けようとし、成功すれば、CC 要求のために蓄積された情報を削除して、CC-Idle 状態に入らなければならない。その関連付けは、invoke APDU で受信した情報とローカルで蓄積した情報とを比較することにより達成しなければならない（8.2.1.5項を参照）。ユーザ B のエンドポイントは、Cause IE #16 “*normal call clearing*” または *undefinedReason* の *ReleaseCompleteReason* を持つ解放完了メッセージを返送しなければならない。

b. ユーザ B のエンドポイントにより開始される取消

CC 要求を取消すために、ユーザ B のエンドポイントは、新しい呼非依存シグナリングコネクションの呼設定メッセージ（コネクション解放の場合）、または、現在の呼非依存シグナリングコネクションにおいて、Cause IE #16 “*normal call clearing*” または *undefinedReason* の *ReleaseCompleteReason* を含む解放完了メッセージのどちらかで、ユーザ A のエンドポイントに *ccCancel* invoke APDU を送信し、CC 要求を削除して、CC-Idle 状態に入らなければならない。呼設定メッセージで送信するならば、invoke APDU は、この CC 要求のためにローカルで蓄積された全ての情報を含まなければならない。

8.2.2. 例外手順

8.2.2.1. CCBS / CCNR インボケーション

既に蓄積された CC 要求が重なったならば、CC 要求は受け入れてはならない。

注：重複した要求は、ユーザ B のエンドポイントが受信した基本呼情報を捨てることに起因することがある。

SS-CCBS または SS-CCNR のための要求が受け入れられないならば、ユーザ B のエンドポイントは、それぞれに現在のシグナリングコネクションにおいて、*ccbsRequest* return error APDU または *ccnrRequest* return error APDU を持つ解放完了メッセージを返送しなければならない。それは *destinationReject* である *ReleaseCompleteReason* を示さなければならない。

失敗状態がいつまでも続かならば（例えば、ユーザ B にサービスが提供されていない）、エラー値 *longTermRejection* を含まなければならない。

失敗状態が長続きしないならば（例えば、重複した CC 要求、動作中 CC 要求の許容数超え）、エラー値 *shortTermRejection* を含まなければならない。

8.2.2.2. 予期しない APDU

CC-Await-Call-Completion 状態において、CC 要求と関連付けできない *ccRingout* invoke APDU を持つ呼に関連した呼設定メッセージを受信すれば、ユーザ B のエンドポイントは、エラー値 *failureToMatch* を含む *ccRingout* return error APDU と *destinationReject* である *ReleaseCompleteReason* を持つ解放完了メッセージを返送しなければならない。

CC-Await-Call-Completion 状態において、CC 要求と関連付けできない *ccResume* invoke APDU を持つ呼非依存シグナリングコネクションのための呼設定メッセージを受信すれば、ユーザ B のエンドポイントは、*ccCancel* invoke APDU と *destinationReject* である *ReleaseCompleteReason* を持つ解放完了メッセージを返送しなければならない。

CC 要求と関連付けできない *ccCancel* invoke APDU を持つ呼非依存シグナリングコネクションのための呼設定メッセージを受信すれば、ユーザ B のエンドポイントは、*destinationReject* である *ReleaseCompleteReason* を持つ解放完了メッセージを返送しなければならない。

8.2.2.3. CC 呼プレゼンテーションでのユーザ B 再ビジー

CC-Await-Call-Completion 状態の間に、呼設定メッセージで *ccRingout* invoke APDU を受信したとき、ユーザ B が再度ビジーであれば、ユーザ B のエンドポイントは、エラー値 *remoteUserBusyAgain* を含む *ccRingout* return error APDU と *destinationReject* である *ReleaseCompleteReason* を持つ解放完了メッセージを返送しなければならない。

サービス維持オプションが使用されるならば、対応する CC 要求を維持し、ユーザ B の監視を再開して、CC-Invoked-User-B 状態に戻らなければならない。コネクション解放オプションが適用されるならば、シグナリングコネクションは解放しなければならない。

サービス維持オプションが使用されないならば、CC 要求を削除し、CC-Idle 状態に入らなければならない。シグナリングコネクションがまだ存在すれば、解放しなければならない。

8.2.2.4. CC 呼の中止

CC 呼の過程において、どの SS-CC invoke APDU も持たない解放完了メッセージを受信すれば、または、ユーザ B がビジー以外の理由で呼の確立が失敗すれば、関連付けられた CC 要求を削除し、CC-Idle 状態に入って、まだ存在するならば、呼非依存シグナリングコネクションを解放しなければならない。

8.2.2.5. 呼非依存シグナリングコネクションの解放

CC-Await-Call-Completion または CC-Wait-User-B-Alert 状態で、CC 要求と関連付けられた呼非依存シグナリングコネクションが、*ccCancel* invoke APDU を送信または受信することなく、何らかの理由により解放されれば、ユーザ B のエンドポイントは、現在の状態に留まらなければならない。

9. 相互接続と相互動作

9.1. SCN との相互接続

SS-CC はゲートウェイ相互接続機能によって、他の標準によって定義される呼完了に関係する付加サービスと相互接続をしてもよい。

この標準の中で定義されるように、SS-CC は 2 つの機能エンティティ EASE と EBSE の有効性を必要とする。これら 2 つの機能エンティティは可能な限りユーザエンドポイントの近くに置くべきである。SCN で呼完了に対応する機能エンティティの有効性を受けて、SS-CC 相互接続のためにゲートウェイが完全な EASE か EBSE 機能 (9.1.1 項参照) を提供するか、限定された EASE / EBSE 機能 (9.1.2 項参照) のみを提供しなければならない。

TTC 注:上記EASE/EBSEの記述は、ITU-T 原文ではEASE/EASEとなっているが、明らかに誤りであると思われるため修正を行った。

9.1.1. ファンクショナル呼完了エンティティが SCN に存在しない

ゲートウェイは SCN が呼完了の機能エンティティを提供していないときに、SCN に SS-CC の相互接続機能を提供しなければならない場合には、SCN ユーザに対して必要な情報を伝える適切な SCN シグナリングを含めて、8.1 節、8.2 節で定義したようにゲートウェイが手順を提供してもよい。この機能エンティティは同様にリモート SCN ユーザの代わりにゲートウェイによって提供された SS-CC プロキシ機能として見なされる。

9.1.2. ファンクショナル呼完了エンティティが SCN に存在する

ゲートウェイは SCN に存在する呼完了 (EASE もしくは EBSE に対応) の機能エンティティを含む SCN に SS-CC の相互接続機能を提供しなければならない場合には、ゲートウェイは以下のように限定された SS-CC 機能を提供しなければならない。

11.2.4 項、11.3.4 項と 11.4 節で定義された SS-CC 関連呼状態とタイマは、このタイプのゲートウェイに適用しない。ゲートウェイ機能はその代わりに、SCN サイドにおいて呼完了シグナリングに対応する SS-CC 関連の APDU の相互接続を含んだメッセージの相互接続に限定されている。そして逆もまた同様である。

これは関連するシグナリング相互接続を含む

- CC 要求 (ccbsRequest と ccnrRequest APDUs)
- SS-CC の経過 (ccExecPossible, ccSuspend と ccResume APDUs)
- CC 呼 (ccRingout APDUs)、および
- SS-CC の取消 (ccCancel APDUs)

SCN A のゲートウェイが SCN A がサービス維持方式をサポートする通知を受けた場合には、SCN A のゲートウェイはエンドポイント B に対してサービス維持方式に対するサポートを通知しなければならない。

SCN A が接続維持オプションを通知した場合には、SCN A のゲートウェイはエンドポイント B に接続維持オプションを通知しなければならない。SCN A が接続解除オプションを指示した場合には、エンドポイント B に接続解除オプションを通知しなければならない。SCN A がユーザ B モニタリングフェーズ中にシグナリング接続を維持するのか解放するのかの選択を通知しない場合には、SCN A のゲートウェイはどちらか選択してもよい。

特定の相互接続要求により禁止されていないければ、SCN B ゲートウェイはエンドポイント A によって通知された接続維持もしくは接続解除オプションを尊重しなければならない。

呼設定メッセージの *ccExecPossible* invoke APDU を受信したゲートウェイは、必ずしも SS-CC インボケーションを処理したゲートウェイと同じではない。同じく、*ccRingout* invoke APDU を受信したゲートウェイも同じであり、SS-CC インボケーションもしくはユーザ B がビジーでなくなったとの通知を処理した、同じゲートウェイである必要はない。しかしながら、エンドポイントは、1 つの特定の (すなわち同じ) ゲートウェイによって 1 つの SS-CC 手順に属している全ての呼設定メッセージの経路を決める能力を用いるために選択してもよい。

注:これは、要素 *destinationAddress* (=ゲートウェイアドレス)と *Setup-UUIE* の *remoteExtensionAddress* (=SCN ユーザアドレス) あるいは要素 *destinationAddress* (=ゲートウェイアドレス)と呼設定メッセージに含まれた JT-H450.9 APDU の *EndpointAddress* の *remoteExtensionAddress* (=SCN ユーザアドレス)を用いているゲートウェイと SCN ユーザの両方のアドレッシングにより、たいていの状況で達成される。同様の方法 - SCN で可能であるなら - 同じ SS-CC 手順に属している対応する呼確立メッセージを用いてもよい。

SS-CC の詳細なゲートウェイ相互接続手順の記述はこの標準の範囲外で他の標準により SCN を定義している。

9.2. SS-CCBS と他の付加サービスとのプロトコル相互動作

この節は本標準の発行時にすでに発行されている他の JT-H450 標準の付加サービスと SS-CCBS とのプロトコル相互動作を記述する。本標準の後で発行された JT-H450 標準の付加サービスとの相互接続については、それらの JT-H450 標準を参照すること。

9.2.1. コールトランスファ (JT-H450.2)

プロトコル相互動作なし。

9.2.2. 着信転送 (JT-H450.3)

9.2.2.1. 無条件着信転送 (SS-CFU)

SS-CFU が JT-H450.3 のとおりにサポートした場合には、以下の相互動作に適用しなければならない。

9.2.2.1.1. SS-CFU 転送先ユーザに SS-CCBS をインボークするためのユーザ A のエンドポイント手順

注:この場合、SS-CCBS に関してのユーザ A のエンドポイントは、同様に SS-CFU に関しての発信エンドポイントである。

SS-CCBS がビジー転送先ユーザをインボークした場合には、ユーザ A のエンドポイントは *nominatedNr* の内容を蓄積しなければならない。オペレーション *divertingLegInformation1* の変数を受信したなら、そしてこのように用いる:

- I この要素を必要とする SS-CCBS オペレーションの変数での要素 *numberB*
- I 呼非依存シグナリングコネクションの呼設定メッセージでの着番号情報要素あるいは *destinationAddress* 要素
- I CC 呼の呼設定メッセージでの着番号情報要素あるいは *destinationAddress* 要素

オリジナル着ユーザのアドレスは SS-CCBS のために用いてはいけない。

要素 *nominatedNr* が利用出来ない場合には、ユーザ A から受信した SS-CCBS 要求は拒否しなければならない。

9.2.2.1.2. SS-CFU がユーザ A によって起動された場合のユーザ A エンドポイント手順

プロトコル相互動作なし。

9.2.2.1.3. SS-CCBS がインボークされた後ユーザ B によって SS-CFU が起動された場合のユーザ B のエン

ドポイント手順

プロトコル相互動作なし。

注：着信の CC 呼は無条件 転送を受けられない。

9.2.2.2. ビジー時着信転送 (SS-CFB)

SS-CFB が JT-H450.3 のとおりにサポートされた場合には、以下の相互動作に適用しなければならない。

9.2.2.2.1. SS-CFB 転送先ユーザに SS-CCBS をインボークするためのユーザ A のエンドポイント手順

注：この場合、SS-CCBS に関してのユーザ A のエンドポイントは、同様に SS-CFB に関しての発信エンドポイントである。

SS-CCBS はビジー転送元ユーザ B に対してよりもビジー転送先ユーザ (ユーザ C) に要求された場合には、9.2.2.1.1 項の手順を適用しなければならない。他の点ではプロトコル相互動作なし。

9.2.2.2.2. SS-CFB がユーザ A によって起動された場合のユーザ A のエンドポイント手順

プロトコル相互動作なし。

9.2.2.2.3. SS-CCBS がインボークした後 SS-CFB がユーザ B によって起動された場合のユーザ B のエンドポイント手順

プロトコル相互動作なし。

注：着信の CC 呼はビジー時転送を受けられない。

9.2.2.3. 無応答時着信転送 (SS-CFNR)

9.2.2.1 項の “SS-CFU” を “SS-CFNR” に置き換えて適用しなければならない。

注：ユーザ B から応答が無いときに、着信の CC 呼は無応答で転送を受けられる。

9.2.2.4. 呼毎着信転送 (SS-CD)

9.2.2.1 項の “SS-CFU” を “SS-CD” に置き換えて適用しなければならない。

注：着信の CC 呼はユーザ B によって転送を受けられる。

9.2.3. コールホールド (JT-H450.4)

プロトコル相互動作なし。

9.2.4. コールパークとコールピックアップ (JT-H.450.5)

プロトコル相互動作なし。

注：ユーザ B への着信の CC 呼は他のユーザによってピックアップできない。

9.2.5. コールウェイティング (JT-H450.6)

プロトコル相互動作なし。

9.2.6. メッセージウェイティング (JT-H450.7)

プロトコル相互動作なし。

9.2.7. 名前表示 (JT-H450.8)

プロトコル相互動作なし。

9.2.8. コールリンケージ (H.323 version 4)

callLinkage 要素は以下の規則に従ってコード化され、どのメッセージに含まれてもよい。

- I 利用可能である場合には、オリジナル呼の試みのスレッド ID、すなわちユーザ B がビジー状態の呼、を同様に用いなければならない (要素 *callLinkage.threadId*)
 - ⇒ *ccRingout* を除いた全ての SS-CC オペレーションを運ぶ呼非依存シグナリングコネクションのために
 - ⇒ CC 呼のために
- I オリジナル呼の試みがスレッド ID を含んでいなかった場合には、その呼識別子値 (要素 *callIdentifier*) は以降のメッセージでスレッド ID (要素 *callLinkage.threadId*) として用いてもよい。
- I すべての呼 / シグナリングコネクションは新しいグローバルコール ID を用なければならない。
(要素 *callLinkage.globalCallId*)

9.3. SS-CCNR と他の付加サービスとのプロトコル相互動作

この節は本標準の発行時にすでに発行されている他の JT-H450 標準の付加サービスと SS-CCNR とのプロトコル相互動作を記述する。本標準の後で発行された JT-H450 標準の付加サービスとの相互接続については、それらの JT-H450 標準を参照すること。

9.3.1. コールトランスファ (JT-H450.2)

プロトコル相互動作なし。

9.3.2. 着信転送 (JT-H450.3)

9.3.2.1. 無条件着信転送 (SS-CFU)

SS-CFU は JT-H450.3 のとおりにサポートされた場合には、以下の相互動作に適用しなければならない。

9.3.2.1.1. SS-CFU 転送先ユーザに SS-CCNR をインボークするためのユーザ A のエンドポイント手順

注：この場合、SS-CCNR に関してのユーザ A のエンドポイントは、同様に SS-CFU に関しての発信エンドポイントである。

SS-CCNR が応答がない転送先ユーザをインボークした場合には、ユーザ A のエンドポイントは要素 *nominatedNr* の内容を蓄積しなければならない。オペレーション *divertingLegInformation1* の変数を受信した場合には、以下のように用いる。

- I この要素を必要とするどのような SS-CCNR オペレーションの変数での要素 *numberB*

- I 呼非依存シグナリングコネクションの呼設定メッセージでの着番号情報要素あるいは *destinationAddress* 要素
- I CC呼の呼設定メッセージでの着番号情報要素あるいは *destinationAddress* 要素

オリジナル着ユーザのアドレスは SS-CCNR に用いてはいけない。

要素 *nominatedNr* が利用出来ない場合には、ユーザ A から受信した SS-CCNR 要求は拒否しなければならない。

9.3.2.1.2. SS-CFU がユーザ A によって起動した場合のユーザ A のエンドポイント手順

プロトコル相互動作なし。

9.3.2.1.3. SS-CCNR がインボークされた後 SS-CFU がユーザ B によって起動した場合のユーザ B のエンドポイント手順

プロトコル相互動作なし。

注：着信の CC 呼は無条件 転送を受けられない。

9.3.2.2. ビジー時着信転送 (SS-CFB)

SS-CFB は JT-H450.3 のとおりにサポートされた場合には、以下の相互動作を適用しなければならない。

9.3.2.2.1. SS-CFB 転送先ユーザに SS-CCNR をインボークするためのユーザ A のエンドポイント手順

注：この場合、SS-CCNR に関してのユーザ A のエンドポイントは、同様に SS-CFB に関しての発信エンドポイントである。

9.3.2.1.1.項の手順を適用しなければならない。

9.3.2.2.2. SS-CFB がユーザ A によって起動した場合のユーザ A のエンドポイント手順

プロトコル相互動作なし。

9.3.2.2.3. SS-CCNR がインボークされた後 SS-CFB がユーザ B によって起動した場合のユーザ B のエンドポイント手順

プロトコル相互動作なし。

注：着信の CC 呼はビジー時転送の適用を受けていない。

9.3.2.3. 無応答時 着信転送 (SS-CFNR)

SS-CFNR が JT-H450.3 のとおりにサポートされた場合には、以下の相互動作を適用しなければならない。

9.3.2.3.1. SS-CFNR 転送先ユーザに SS-CCNR をインボークするためのユーザ A のエンドポイント手順

SS-CCNR が転送元ユーザ B (両ユーザが応答なし) に対してよりむしろ転送先ユーザ (ユーザ C) に要求された場合には、9.3.2.1.1 項の手順を適用しなければならない。他の点ではプロトコル相互動作なし。

9.3.2.3.2. SS-CFNR がユーザ A によって起動された場合のユーザ A のエンドポイント手順。

プロトコル相互動作なし。

9.3.2.3.3. SS-CCNR がインボークした後 SS-CFNR がユーザ B によって起動された場合のユーザ B のエンドポイント手順

プロトコル相互動作なし。

9.3.2.4. 呼毎着信転送 (SS-CD)

呼毎着信転送のプロトコル相互動作は SS-CFNR の相互動作による 9.3.2.3 項を適用しなければならない。

9.3.3. コールホールド (JT-H450.4)

プロトコル相互動作なし。

9.3.4. コールパークとコールピックアップ (JT-H450.5)

プロトコル相互動作なし。

注：ユーザ B への着信の CC 呼は他のユーザによりピックアップ出来ない

9.3.5. コールウェイティング (JT-H450.6)

プロトコル相互動作なし。

注：SS-CCNR のインボケーションはコールウェイティングを終了する。

9.3.6. メッセージウェイティング (JT-H450.7)

プロトコル相互動作なし。

9.3.7. 名前表示 (JT-H450.8)

プロトコル相互動作なし。

9.3.8. コールリンケージ (JT-H323 Version 4)

callLinkage 要素は以下の規則に従ってコード化され、どのメッセージに含まれてもよい。

- Ⅰ 利用可能である場合には、オリジナル呼の試みのスレッド ID、すなわちユーザ B によって応答がない呼、は同様に用いなければならない (要素 *callLinkage.threadId*)
 - ⇒ ccRingout を除いた全ての SS-CC オペレーションを運ぶ呼非依存シグナリングコネクションのために
 - ⇒ CC 呼のために。
- Ⅰ オリジナルコールの試みがスレッド ID を含んでいなかった場合には、その呼識別子値 (*callIdentifier* 要素) は以降のメッセージでスレッド ID (*callLinkage.threadId* 要素) として用いてもよい。
- Ⅰ すべての呼 / シグナリングコネクションは新しいグローバルコール ID を用いなければならない。 (要素 *callLinkage.globalCallId*)

10. ゲートキーパー動作

ゲートキーパールーティッドモデルの場合には、2つのモードが可能である。:

- I ゲートキーパーはエンドポイント(10.1節)が処理をした全ての受信された SS-CC オペレーションを透過する。もしくは
- I ゲートキーパーは、SS-CC(10.2節)によりエンドポイント A および / もしくはエンドポイント B の代わりに動作する。

注: ゲートキーパーのほか、他の“transit”エンティティは SS-CC によりエンドポイントの代わりに動作するかもしれない。この意味での“transit”エンティティは以下の節で“プロキシ”として述べられている。

10.1. ゲートキーパーはエンドポイントに SS-CC オペレーションを透過する。

このモードは、ゲートキーパーは適切なエンドポイント処理のために、エンドポイントに SS-CC オペレーションを透過しなければならない。

注: 必もし要求されれば、ゲートキーパーは SS-CC オペレーションの内容を変更するかもしれない。(例えば、表示制限としてマークされたアドレス情報の削除、また着側エンドポイントが表示制限指示の尊重に対して信頼されない場合)。

10.2. ゲートキーパー / プロキシがエンドポイントの代わりに動作する

10.2.1. ゲートキーパー / プロキシがエンドポイント B の代わりに動作する

エンドポイント B に代替動作をしているゲートキーパー / プロキシ(ゲートキーパールーティッドモデルのための、あるいは、それぞれ、プロキシを通して経路を定められた呼の場合)は、全ての SS-CC オペレーションがエンドポイント B を予定した着信元になることを決めてもよい、そして全ての SS-CC オペレーションの発信元はエンドポイント A に送信する。ゲートキーパー / プロキシは 8.2 節の“ユーザ B のエンドポイントでの動作”で定義したように動作しなければならない。

この目的のためにゲートキーパー / プロキシはエンドポイント B のビジー / フリー状態を監視しなければならない。これが達成される方法は本標準の範囲外である。

スティミュラス基本プロトコルは、この場合ゲートキーパー / プロキシとゲートキーパー / プロキシが SS-CC 制御エンティティとして動作をしているユーザ B のエンドポイントとの間で用いられる。

10.2.2. ゲートキーパー / プロキシがエンドポイント A の代わりに動作する

ゲートキーパー / プロキシ(ゲートキーパールーティッドモデルのためにあるいは、それぞれ、プロキシを通して経路を決められた呼の場合)がエンドポイント A の代わりに SS-CC 制御エンティティとして動作する。そして、全ての SS-CC オペレーションはエンドポイント B に送信された発信元になり、全ての SS-CC オペレーションの着信元はエンドポイント A を予定する。ゲートキーパー / プロキシは、8.1 節“ユーザ A のエンドポイントの動作”で定義された動作を提供しなければならない。

スティミュラス基本プロトコルはゲートキーパー / プロキシで SS-CC を開始および制御するためにユーザ A のエンドポイントにより用いられる。CC リコールの時にエンドポイント A は通知しなければならない。

そして、CC呼はエンドポイントAとエンドポイントBの間で確立されなくてはならない。これが達成される方法は本標準の範囲外である。

11. ダイナミック記述

11.1. オペレーションモデルおよびシグナルフロー

本章では、いくつかの典型的な SS-CCBS および SS-CCNR のメッセージフローについて記述する。以下の決まりが、本章の図中で使用される。

1. 以下の記号が使用される。

- ▶ 呼依存プロトコルメッセージ
- ▶ ユーザエンドポイント間での呼非依存プロトコルメッセージ
-▶ プリミティブ (JT-H450.9 の規定範囲外)

呼設定	メッセージ名
Cx	接続番号 x
xxx.inv	操作 xxx の invoke APDU
xxx.rr	操作 xxx の return result APDU
xxx.re	操作 xxx の return error APDU

注：

以下の手順は個々のシグナルフロー図を示す。

- *ダイレクターヘッドコールモデル (例 図 2/JT-H450.9)*
- *ゲートキーパーを通る H.450.9 APDU のゲートキーパーヘッドコールモデル (例 図 3/JT-H450.9)*
- *エンドポイントAとしてゲートキーパー/プロキシが動作するゲートキーパーヘッドコールモデル (例 図 4/JT-H450.9)*
- *エンドポイントBとしてゲートキーパー/プロキシが動作するゲートキーパーヘッドコールモデル (例 図 5/JT-H450.9)*

エンドポイント処理における全ての SS-CC APDU がゲートキーパーを通過する場合、エンドポイントは SS-CC の処理をサポートする (10.1 節参照)

10.2 節で述べたモデルを適用する場合、発信元のゲートキーパー/プロキシAは8.1節に定義したユーザAのエンドポイントとしてSS-CC処理を行ってもよいし、宛先のゲートキーパー/プロキシBは8.2節に定義したユーザBのエンドポイントとしてSS-CC処理を行ってもよい。この場合、エンドポイントとゲートキーパー/プロキシ間のインタフェースは、情報提供の目的のみに示し、本勧告の範囲外であるステイミュラスフィーチャープロトコルとみてもよい。

11.1.1. 成功 CCBS

図 1/JT-H450.9 から 6/JT-H450.9 は、ダイレクターヘッドコールモデル、ゲートキーパーヘッドコールモデル (エンドポイント処理の SS-CCBS APDU を通す)、SS-CCBS に関してそれぞれのエンドポイント A または B の代わりに動作するゲートキーパーあるいはプロキシの、成功 CCBS 要求およびリコール手順を示す。

図 3/JT-H450.9 は、ユーザ B の監視中にシグナル接続が解放される接続解放オプションのダイレクトル-ティッドコールを示す。このオプションは、ゲートキーパーレティッドコールの場合も適用する（ここで例は示さない）。

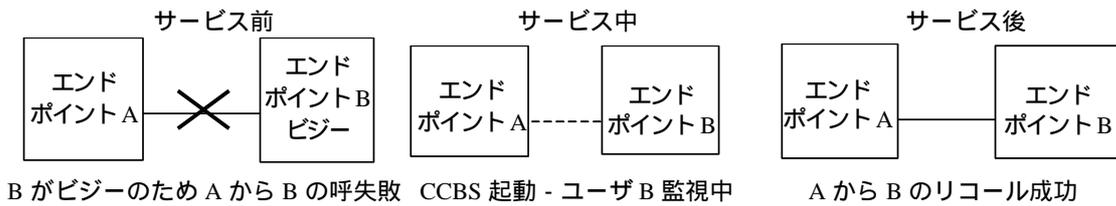


図 1/JT-H450.9

SS-CCBS のオペレーションモデル

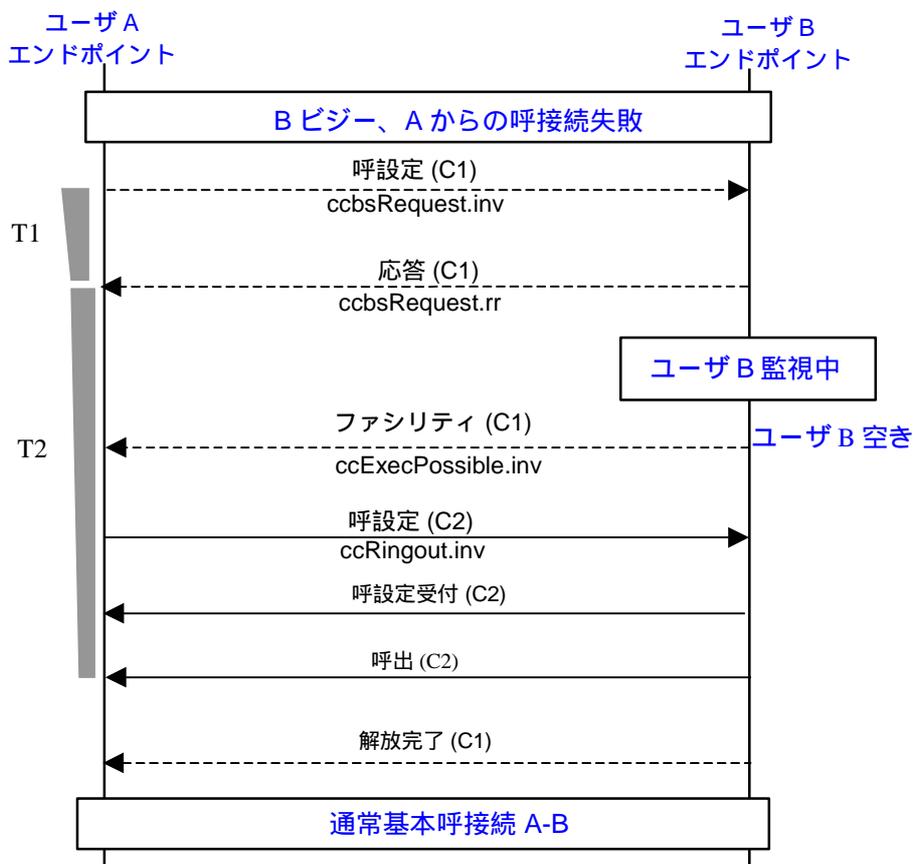


図 2/JT-H450.9

成功 CCBS のメッセージフロー例

ダイレクトル-ティッドコールシグナリング - シグナリング接続維持

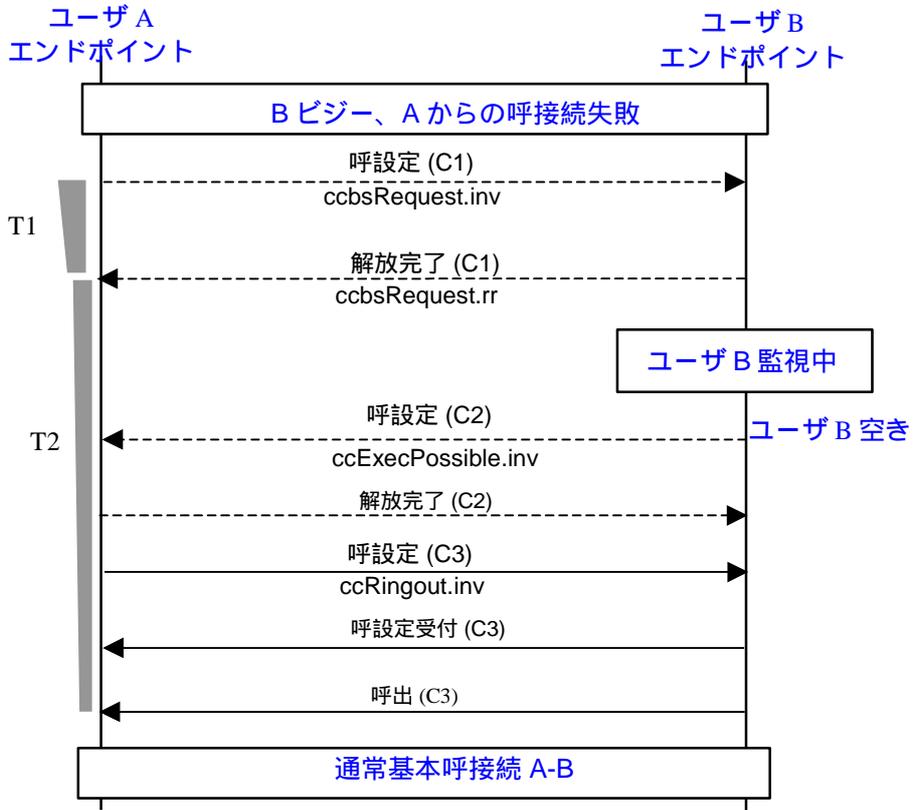


図 3/JT-H450.9

成功 CCBS メッセージフロー例

ダイレクトルーティッドコールシグナリング - 接続解放オプション

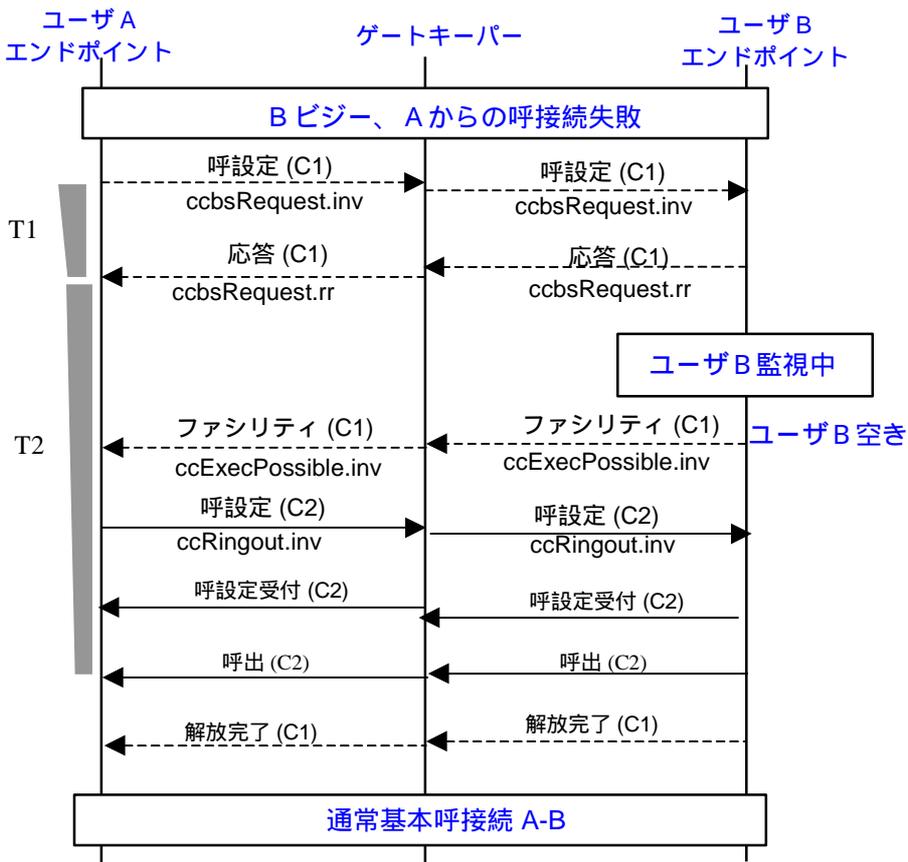
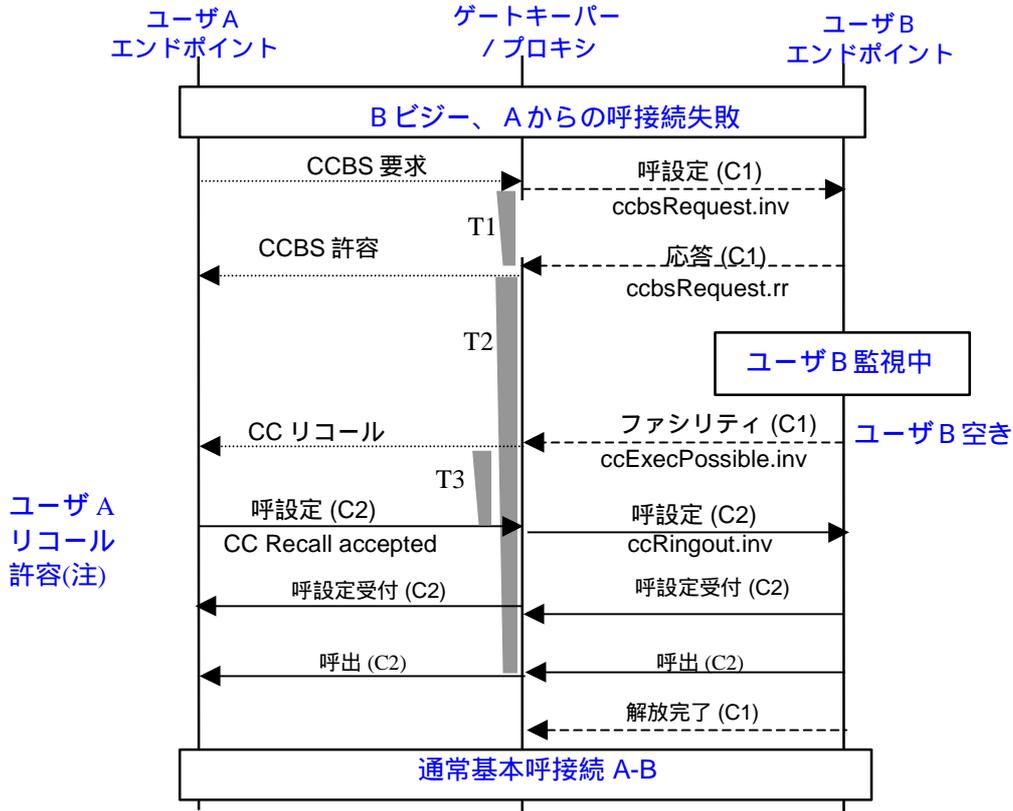


図 4/JT-H450.9

成功 CCBS メッセージフロー例
 GK ルーティッドコールシグナリング、GK は SS-CCBS に関係ない



注：ユーザAのエンドポイントから送信される呼設定は、CC呼を確立するための1つ可能な方法である。他の方法も可能である。これは、本勧告の範囲外である。

図 5/JT-H450.9
 成功 CCBS メッセージフロー例
 GK ルーティッドコールシグナリング、GK によるユーザAの SS-CCBS 制御

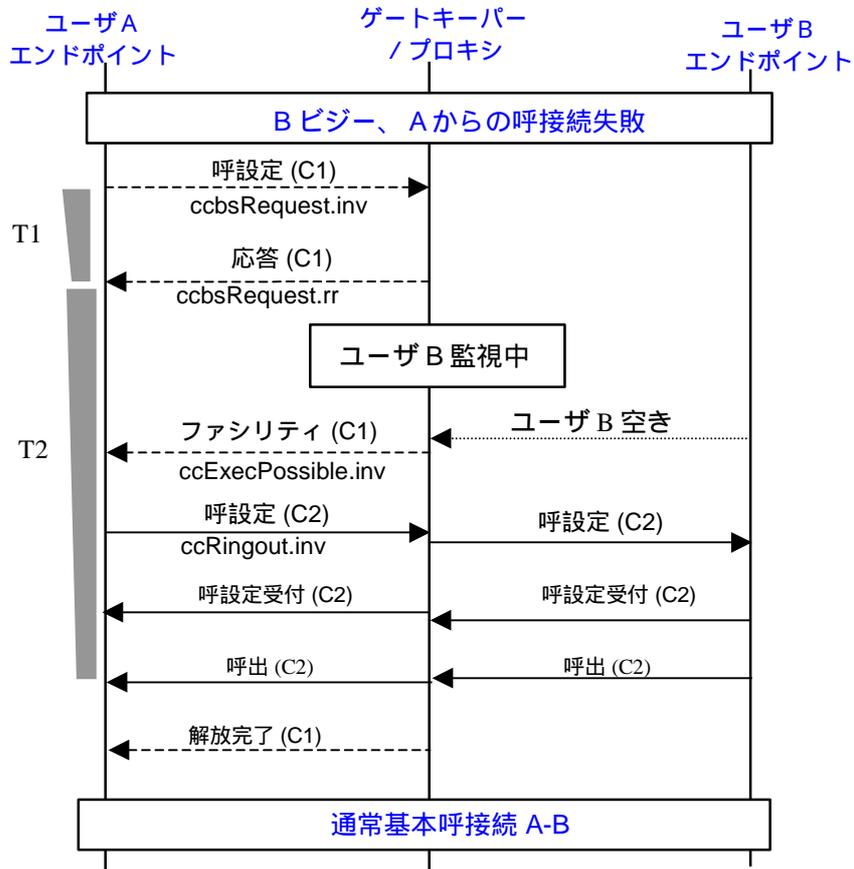


図 6/JT-H450.9

成功 CCBS メッセージフロー例

GK ルーティッドコールシグナリング、GK によるユーザBのSS-CCBS制御

11.1.2. 成功 CCNR

図 7/JT-H450.9 および 8/JT-H450.9 は、成功 CCNR 要求とリコールの例を示す。原則的には、SS-CCBS と同様の手順が可能である (11.1.1 項参照)。

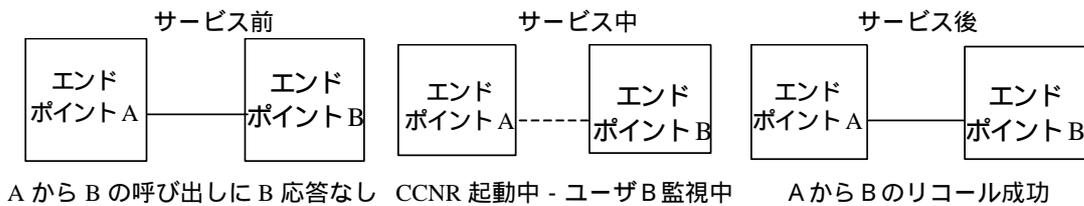


図 7/JT-H450.9

SS-CCNR のオペレーションモデル

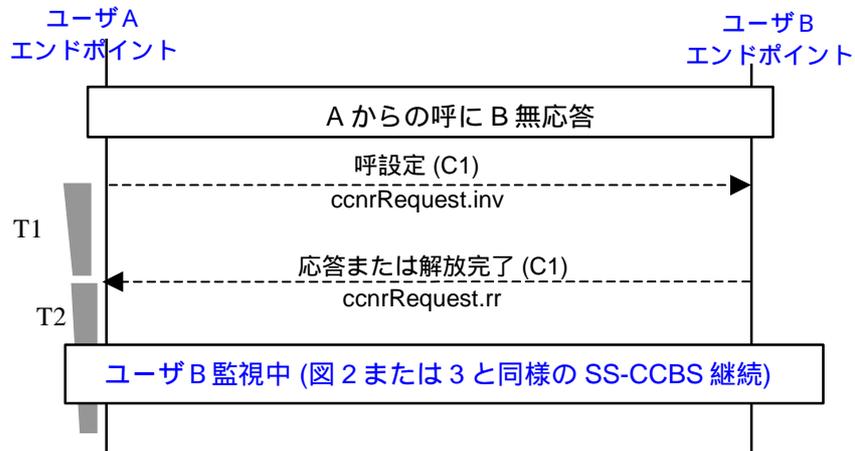


図 8/JT-H450.9

成功 CCNR のメッセージフロー例
ダイレクターティッドコールシグナリング

11.1.3. ユーザ A ビジー

図 9/JT-H450.9 から 11/JT-H450.9 は、ユーザ B がビジーでなくなったときにユーザ A がビジーになった場合の CC 要求を示す。ダイレクターティッドコールモデルにおいて、シグナリング接続の維持および接続解放ケースを示す。他の可能な手順については、11.1.1 項参照。

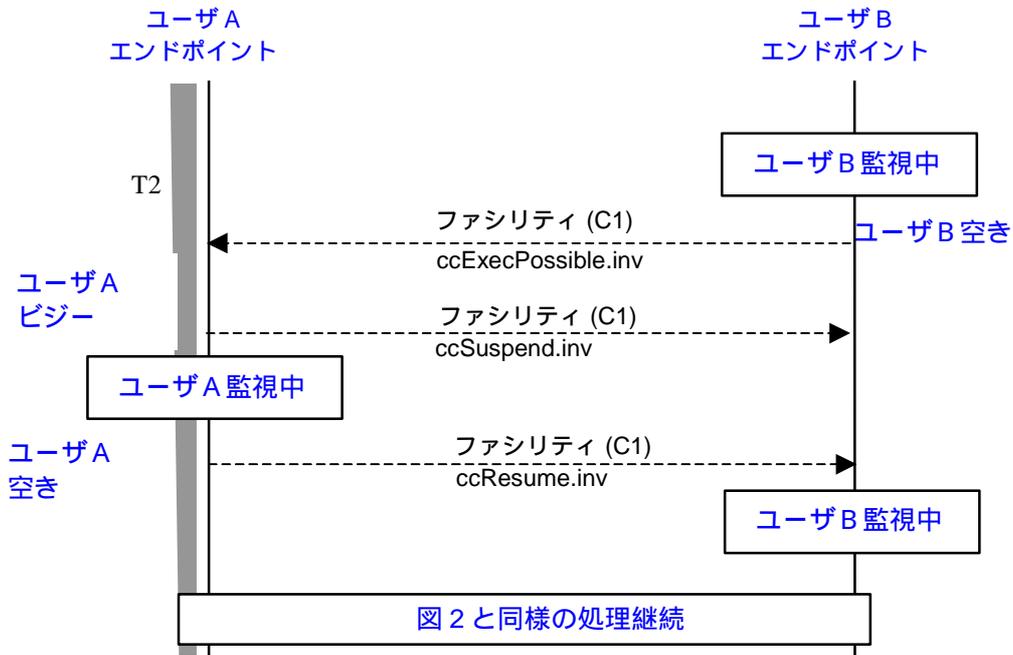


図 9/JT-H450.9

ユーザ A ビジーのメッセージフロー例
ダイレクターティッドコールシグナリング - シグナリング接続維持

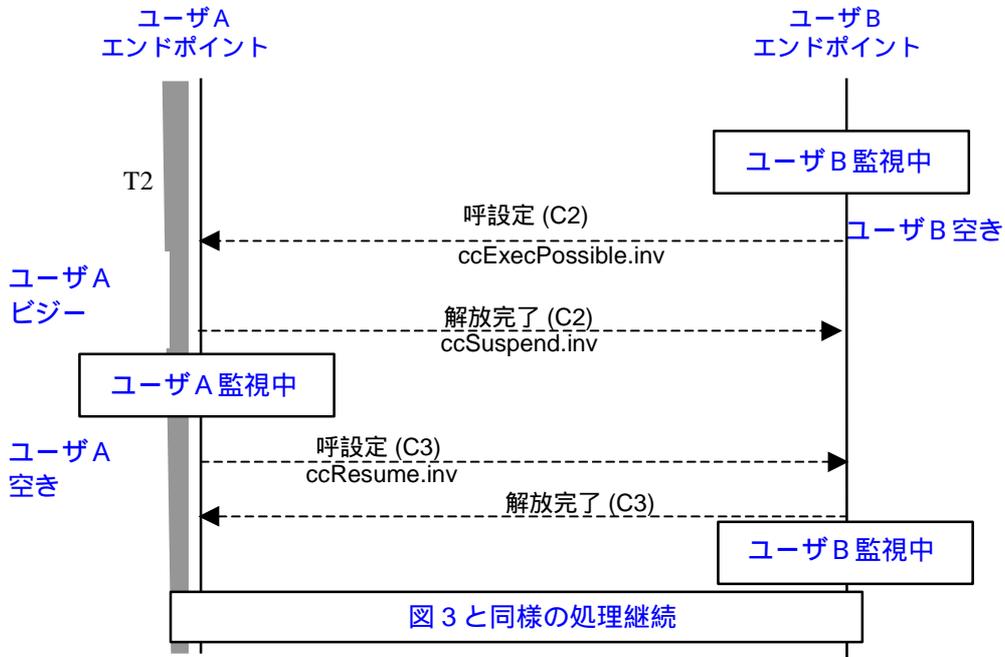


図 10/JT-H450.9

ユーザAビジーのメッセージフロー例

ダイレクトルーティッドコールシグナリング - 接続解放オプション

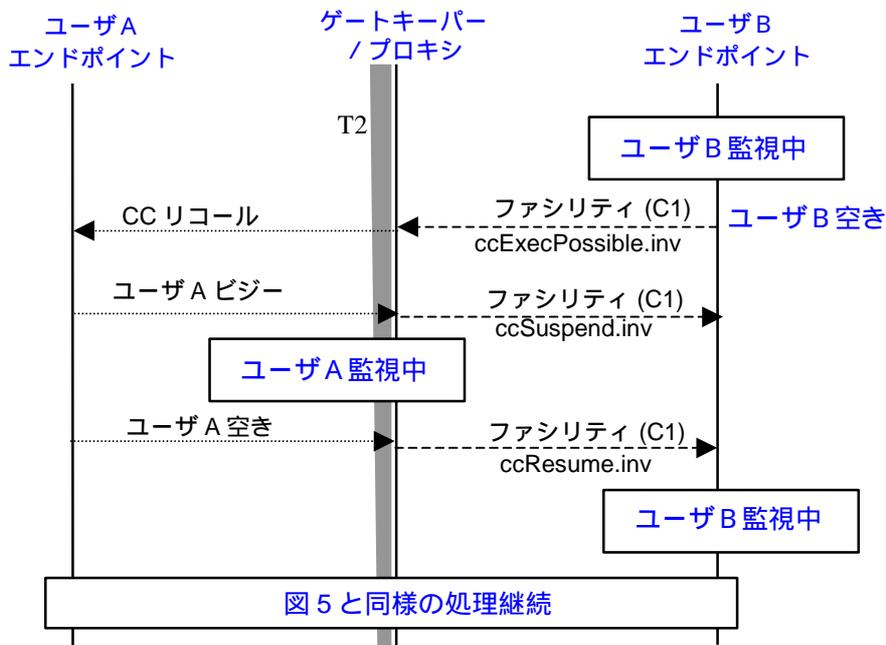


図 11/JT-H450.9

ユーザAビジーのメッセージフロー例

GK ルーティッドコールシグナリング、GK によるユーザ A の SS-CCBS 制御

11.1.4 ユーザB 再ビジー

11.1.4.1. CC 呼提供 - SS-CC サービス維持なし

図 12/JT-H450.9 から 14/JT-H450.9 は、CC 呼がユーザ B 再ビジーに遭遇し、SS-CC サービスが維持されていない場合について示す。他の可能な手順については、11.1.1 項参照。

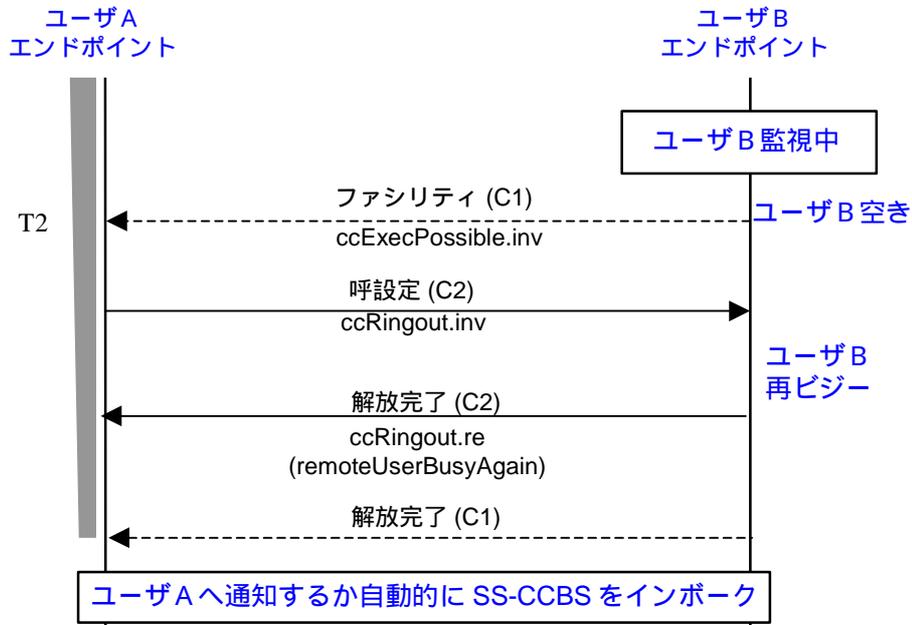


図 12/JT-H450.9

不成功 CCBS メッセージフロー例：ユーザB 再ビジー
ダイレクトルーティッドコールシグナリング

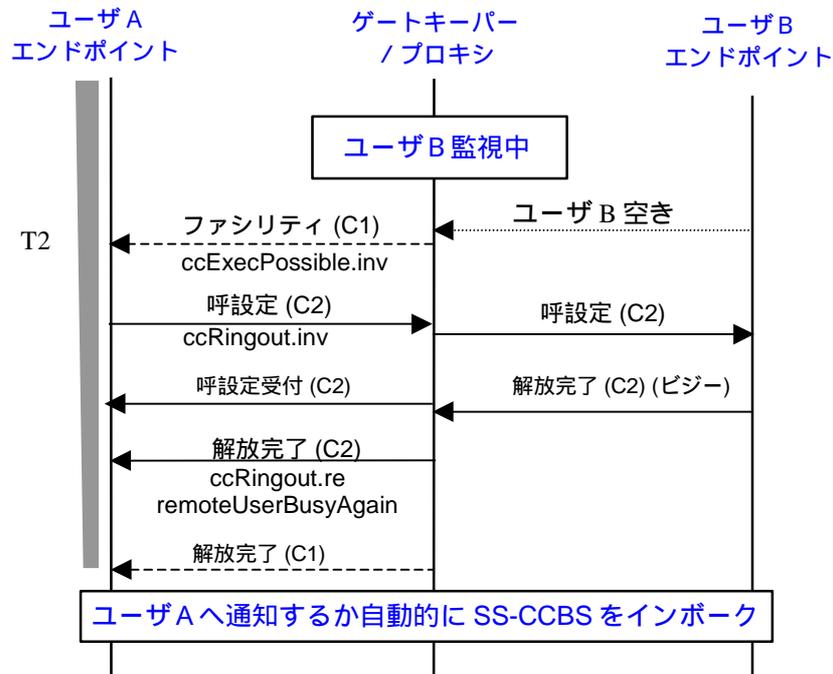


図 13/JT-H450.9

不成功 CCBS メッセージフロー例：ユーザB 再ビジー
GK ルーティッドコールシグナリング、GK によるユーザBの SS-CCBS 制御
シグナリング接続維持

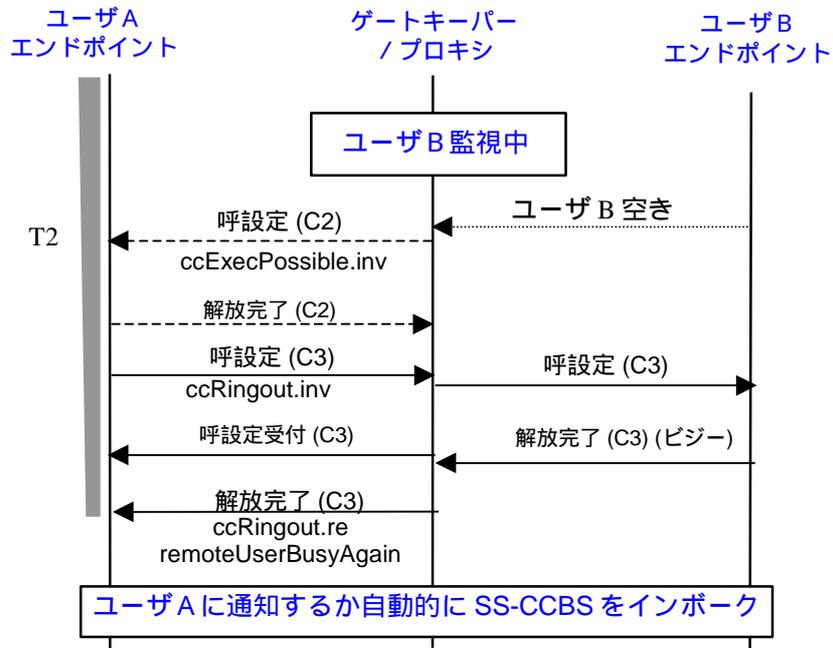


図 14/JT-H450.9

不成功 CCBS メッセージフロー例：ユーザB 再ビジー

GK ルーティッドコールシグナリング、GK によるユーザB の SS-CCBS 制御
接続解放オプション

11.1.4.2. CC 呼提供 - SS-CC サービス維持を伴う

図 15/JT-H450.9 から 17/JT-H450.9 は、CC 呼がユーザB 再ビジーに遭遇し、SS-CC サービスが維持されている場合を示す。他の可能な手順は 11.1.1 項参照。

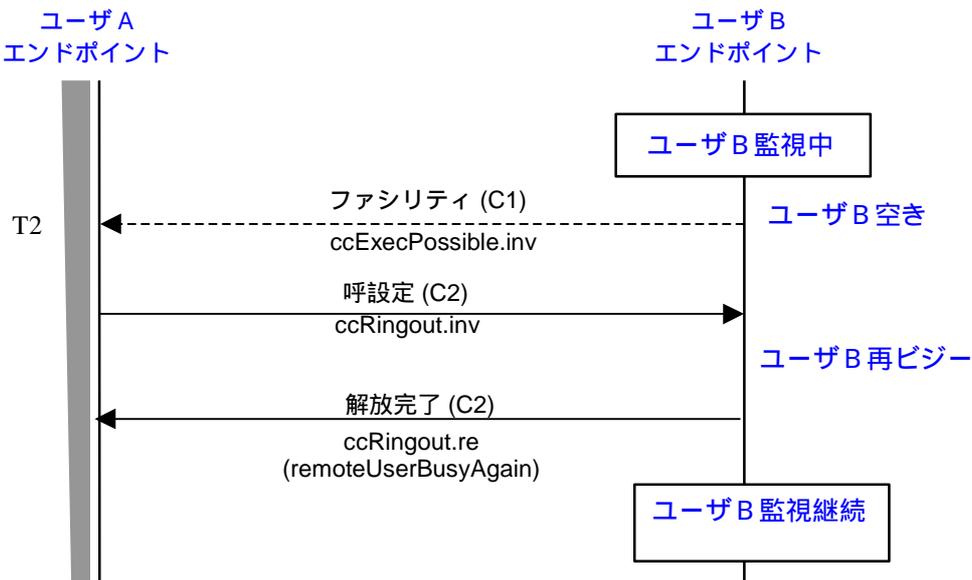


図 15/JT-H450.9

不成功 CCBS メッセージフロー例：ユーザB 再ビジー

ダイレクトコールシグナリング、シグナリング接続維持

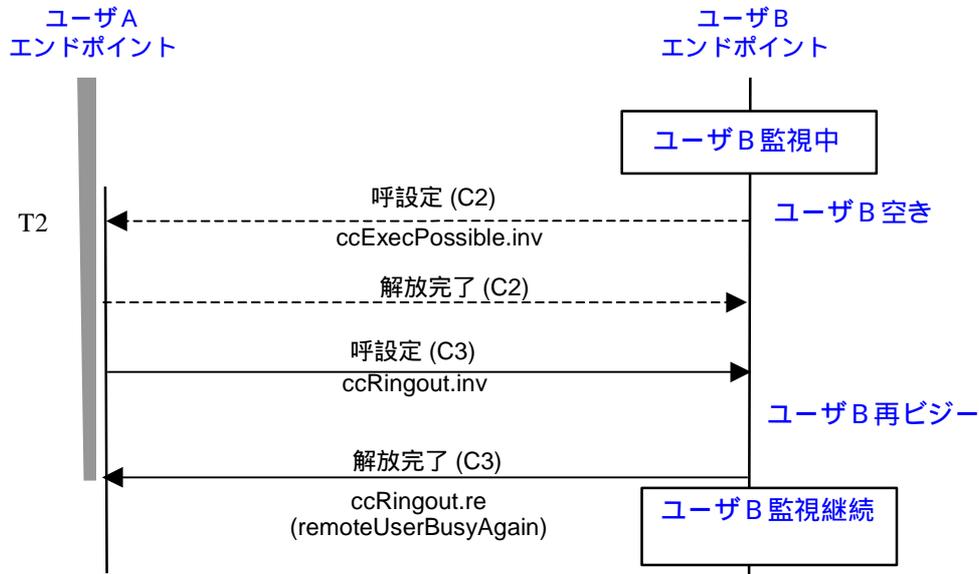


図 16/JT-H450.9

不成功 CCBS メッセージフロー例：ユーザB再ビジー
ダイレクターティッドコールシグナリング、接続解放オプション

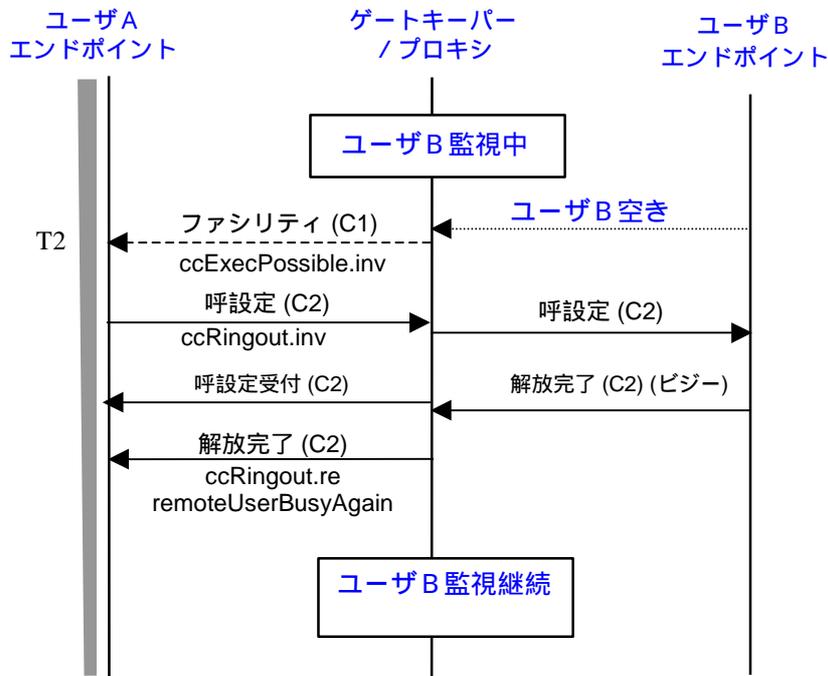


図 17/JT-H450.9

不成功 CCBS メッセージフロー例：ユーザB再ビジー
GK ルーティッドコールシグナリング、GK によるユーザBの SS-CCBS 制御

11.2. エンドポイント A シグナリングエンティティ (EASE) とそのシグナリングエンティティユーザ (情報を与える) 間の通信

もしゲートキーパー/プロキシがエンドポイントとして動作する場合、ゲートキーパー/プロキシはシグナリングエンティティと考えられ、エンドポイントであるゲートキーパー/プロキシは、シグナリングエンティティユーザとしてみられるよう対応する。この場合、ローカルなプリミティブ手順は代えられる。例えば、適切なスティミュラスフィーチャーシグナリング手順によって。

11.2.1. プリミティブ表

表 1/JT-H450.9
ユーザ A のエンドポイントにおけるプリミティブ

属称	型			
	要求 (req)	通知 (ind)	応答 (resp)	確認 (conf)
CcbsRequest	パラメータ	未定義 ¹⁾	未定義	パラメータ
CcnrRequest	パラメータ	未定義	未定義	パラメータ
CcExecPossible	未定義	パラメータ	未定義	未定義
CcRingout	パラメータ	未定義	未定義	パラメータ
CcCancel	パラメータ	パラメータ	未定義	未定義
CcSuspend	パラメータ	未定義	未定義	未定義
CcResume	パラメータ	未定義	未定義	未定義

1) は、このプリミティブが未定義であることを意味する。

11.2.2. プリミティブ定義

CcbsRequest.Request プリミティブは SS-CCBS をインボークするために使用する。CcbsRequest.Confirm プリミティブは、インボケーション試行の結果を通知するために使用する。

CcnrRequest.Request プリミティブは、SS-CCNR をインボークするために使用する。 CcnrRequest.Confirm プリミティブは、インボケーション試行の結果を通知するために使用する。

CcExecPossible.Indication プリミティブは、リコールユーザ B の可能性を通知するために使用する。

CcRingout.Request プリミティブは、CC 呼をユーザ B へ通知するために使用する。 CcRingout.Confirm プリミティブは、間違った場合に CC 呼の失敗を通知するために使用する。

CcCancel.Request プリミティブは、SS-CC などを取消するために使用する。 CcCancel.Indication プリミティブ SS-CC などの不成功終了を通知するために使用する。

CcSuspend.Request プリミティブは、ユーザ A がビジーのときに SS-CC を一時停止するために使用され、CC リコールを実施することはできない。

CcResume.Request プリミティブは、ユーザ A がビジーのため一時停止中の後、SS-CC を再開するために使用する。

11.2.3. パラメータ定義

CcbsRequest.Request パラメータ

sourceAddress: ユーザ A のアドレス
 destinationAddress: ユーザ B のアドレス
 ccIdentifier: ビジーユーザ B へ試みた前回の (失敗した) 呼の ID (オプション)
 service: 呼の試みに失敗したことを意図するサービスタイプ
 canRetainService: サービス維持オプションの表示

retainSigConn: シグナリング接続の制御を表示
extension: 非標準 (例 製造者独自)、情報 (オプション)

CcbsRequest.Confirm パラメータ

11.3.3 項参照 (CcbsRequest.Response プリミティブのためのパラメータ)

CcnrRequest.Request パラメータ

sourceAddress: ユーザ A のアドレス
destinationAddress: ユーザ B のアドレス
ccIdentifier: ビジーユーザ B へ試みた前回の無応答呼の ID (オプション)
service: 呼の試みに無応答であったことを意図するサービスタイプ
canRetainService: サービス維持オプションの表示
retainSigConn: シグナリング接続の制御を表示
extension: 非標準 (例 製造者独自)、情報 (オプション)

CcnrRequest.Confirm パラメータ

11.3.3 項参照 (CcnrRequest.Response プリミティブのためのパラメータ)

CcExecPossible.Indication パラメータ

11.3.3 項参照 (CcExecPossible.Request プリミティブのためのパラメータ)

CcRingout.Request パラメータ

ccIdentifier: EASE における SS-CC など CC 呼との関連 (オプション)
extension: 非標準 (例 製造者独自)、情報 (オプション)

CcRingout.Confirm パラメータ

11.3.3 項参照 (CcRingout.Response プリミティブのためのパラメータ)

CcCancel.Request および CcCancel.Indication パラメータ

ccIdentifier: SS-CC などが取消されたことを示す (オプション)
sourceAddress: 記憶したユーザ A のアドレス (オプション)
destinationAddress: 記憶したユーザ B のアドレス (オプション)
service: 記憶したサービスタイプ (オプション)
extension: 非標準 (例 製造者独自)、情報 (オプション)

CcSuspend.Request パラメータ

ccIdentifier: SS-CC などが一時停止されたことを示す (オプション)
extension: 非標準 (例 製造者独自)、情報 (オプション)

CcResume.Request parameters

ccIdentifier: SS-CC などが再開されたことを示す (オプション)
sourceAddress: 記憶したユーザ A のアドレス (オプション)
destinationAddress: 記憶したユーザ B のアドレス (オプション)

service: 記憶したサービスタイプ (オプション)
 extension: 非標準 (例 製造者独自)、情報 (オプション)

11.2.4. 呼状態

CC 状態

概要

CC-Idle この状態は、もし SS-CC が動作していなければ存在する。
 CC-Invoked-User-A-RET この状態は、シグナリング接続維持された状態でユーザ B がビジーでないという通知を待っている間、アクティブ CC 要求で存在する。
 CC-Invoked-User-A-RLS この状態は、シグナリング接続が解放された状態でユーザ B がビジーでないという通知を待っている間、アクティブ CC 要求で存在する。
 CC-Ringout この状態は、ユーザ A が CC リコールを許容しているが、ユーザ B への呼完了が、まだ保留中であるとき存在する。
 CC-Suspended-User-A この状態は、ユーザ A がビジーのため CC 呼が延長されているとき存在する。
 CC-Wait-Ack この状態は、SS-CC インボケーション中のとき存在する。
 CC-Wait-User-A-Answer この状態は、ユーザ A からの CC リコール許容を待っている間存在する。

11.3. エンドポイント B シグナリングエンティティ (EBSE) とそのシグナリングエンティティユーザ (情報を与える) 間の通信

もしゲートキーパー/プロキシがエンドポイントとして動作する場合、ゲートキーパー/プロキシはシグナリングエンティティと考えられる。エンドポイントであるゲートキーパー/プロキシは、シグナリングエンティティユーザとしてみられるように対応する。この場合、ローカルなプリミティブ手順は代えられる。例えば、適切なステイミュラスフィーチャーシグナリング手順によって。

11.3.1. プリミティブ表

表 2/JT-H450.9
 ユーザ B のエンドポイントにおけるプリミティブ

属称	型			
	要求 (req)	表示 (ind)	応答 (resp)	確認 (conf)
CcbsRequest	未定義 ¹⁾	パラメータ	パラメータ	未定義
CcnrRequest	未定義	パラメータ	パラメータ	未定義
CcExecPossible	パラメータ	未定義	未定義	未定義
CcRingout	未定義	パラメータ	パラメータ	未定義
CcCancel	パラメータ	パラメータ	未定義	未定義
CcSuspend	未定義	パラメータ	未定義	未定義
CcResume	未定義	パラメータ	未定義	未定義

1) は、このプリミティブが未定義であることを意味する。

11.3.2. プリミティブ定義

CcbsRequest.Indication プリミティブは SS-CCBS のインボケーションを試みることを通知するために使用する。CcbsRequest.Response プリミティブは、インボケーション試行の結果を通知するために使用する。

CcnrRequest.Indication プリミティブは、SS-CCNR のインボケーションを試みることを通知するために使

用する。 CcnrRequest.Indication プリミティブは、インボケーション試行の結果を通知するために使用する。

CcExecPossible.Request プリミティブは、ユーザ B が空きになったことを通知するために使用する。

CcRingout.Indication プリミティブは、CC 呼として着信している呼を識別するために使用する。

CcRingout.Response プリミティブは、CC 呼試みの失敗を通知するために使用する。

CcCancel.Request プリミティブは、SS-CC などを取り消すために使用する。 CcCancel.Indication プリミティブ SS-CC などの取り消しを通知するために使用する。

CcSuspend.Indication プリミティブは、SS-CC などの一時停止を通知するために使用する。

CcResume.Indication プリミティブは、一時停止された SS-CC などの再開を通知するために使用する。

11.3.3. パラメータ定義

CcbsRequest.Indication パラメータ

11.2.3 項参照 (CcbsRequest.Request プリミティブのためのパラメータ)

CcbsRequest.Response パラメータ (ack および rej)

(ack) retainService: インボケーションが許容されている場合、サービス維持オプションについて表示する

(ack) extension: 非標準 (例 製造者独自)、情報 (オプション)

(rej) rejectPerm: インボケーション拒否 (長期間)

(rej) rejectTemp: インボケーション拒否 (短期間)

CcnrRequest.Indication パラメータ

11.2.3 項参照 (CcnrRequest.Request プリミティブのためのパラメータ)

CcnrRequest.Response パラメータ (ack および rej)

(ack) retainService: インボケーションが許容されている場合、サービス維持オプションについて表示する。

(ack) extension: 非標準 (例 製造者独自)、情報 (オプション)

(rej) rejectPerm: インボケーション拒否 (長期間)

(rej) rejectTemp: インボケーション拒否 (短期間)

CcExecPossible.Request パラメータ

ccIdentifier: EBSE における SS-CC などの表示 (オプション)

sourceAddress: 記憶したユーザ A のアドレス (オプション)

destinationAddress: 記憶したユーザ B のアドレス (オプション)

service: 記憶したサービスタイプ (オプション)

extension: 非標準 (例 製造者独自)、情報 (オプション)

CcRingout.Indication パラメータ

11.2.3 項参照 (CcRingout.Request プリミティブのためのパラメータ)

CcRingout.Response パラメータ (rej)

error: CC 呼の失敗理由

CcCancel.Request および CcCancel.Indication パラメータ

11.2.3 項参照 (CcCancel.Request および CcCancel.indication プリミティブのためのパラメータ)

CcSuspend.Indication パラメータ

11.2.3 項参照 (CcSuspend.Request プリミティブのためのパラメータ)

CcResume.Indication パラメータ

11.2.3 項参照 (CcResume.Request プリミティブのためのパラメータ)

11.3.4. 呼状態

<u>CC 状態</u>	<u>概要</u>
CC-Idle	この状態は、もし SS-CC が動作していなければ存在する。
CC-Await-Call-Completion	この状態は、ユーザ B がビジーでないことを通知した後、着信 CC 呼を待っている間存在する。
CC-Invoked-User-B	この状態は、CC 要求を受信した結果としてユーザ B が監視されている間存在する。
CC-Wait-User-B-Alert	この状態は、許可 (呼出あるいは応答) を待っている間に、CC 呼がユーザ B へ拡張された後存在する。

11.4. タイマ

11.4.1. ユーザ A のエンドポイントにおけるタイマ

ユーザ A のエンドポイントは、以下のタイマを実装しなければならない。

タイマ T1 : SS-CC 要求保護

このタイマは、*ccbsRequest* または *ccnrRequest* invoke APDU を送信するとき開始し、応答を受信したときに停止する。

タイマ T1 が満了した場合、失敗表示がユーザ A へ送信される。

タイマ T1 は、10-30 秒の範囲で期間を持たなければならない。

タイマ T2 : SS-CC サービス期間

このタイマは、*ccbsRequest* または *ccnrRequest* return result APDU を受信したとき開始し、CC 呼が完了したときか CC 要求が取消されたときに停止する。

タイマ T2 が満了となった場合、CC 要求は取消される。

タイマ T2 の期間は、1 分から 60 分の範囲で、*ccbsRequest* return result APDU の受信により開始したときは初期値 15 分とする。

タイマ T2 の期間は、1 分から 1440 分 (すなわち 24 時間) で、*ccnrRequest* return result APDU の受信により開始したときは、初期値 60 分とする。

タイマ T3 : SS-CC リコール

このタイマは、CC リコールがユーザ A に通知されるときに開始し、ユーザ A が応答したとき停止する。

タイマ T3 が満了となったとき、CC 要求は取消される。

タイマ T3 は、10-30 秒の範囲で期間を持たなければならない。

11.4.2. ユーザ B のエンドポイントにおけるタイマ

なし。

12. 呼完了付加サービスのサポートにおける操作

以下の抽象構文記法 1 で定義される操作を適用すべきである。

Call-Completion-Operations

```
{ itu-t recommendation h 450 9 version1(0) call-completion-operations(0) }
```

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

```
IMPORTS      OPERATION, ERROR FROM Remote-Operations-Information-Objects
              { joint-iso-itu-t remote-operations(4)
                informationObjects(5) version1(0) }
EXTENSION, Extension{ } FROM
              Manufacturer-specific-service-extension-definition
              { itu-t recommendation h 450 1 version1(0) msi-definition(18) }
CallIdentifier FROM H323-MESSAGES -- JT-H225.0 参照
EndpointAddress FROM
              Addressing-Data-Elements
              { itu-t recommendation h 450 1 version1(0)
addressing-data-elements (9) }
              MixedExtension, undefined FROM Call-Hold-Operations
              { itu-t recommendation h 450 4 version1(0)
call-hold-operations(0) }
              supplementaryServiceInteractionNotAllowed FROM
H4501-General-Error-List
              { itu-t recommendation h 450 1 version1(0)
              general-error-list (1) }
              BasicService FROM Message-Waiting-Indication-Operations
              { itu-t recommendation h 450 7 version1(0)
              message-waiting-operations(0) };
```

```
H323CallCompletionOperations      OPERATION ::=
{ ccbsRequest | ccnrRequest | ccCancel | ccExecPossible | ccRingout | ccSuspend | ccResume }
```

```
ccbsRequest      OPERATION      ::=
{
  ARGUMENT      CcRequestArg
  RESULT        CcRequestRes
  ERRORS        { shortTermRejection | longTermRejection | undefined |
supplementaryServiceInteractionNotAllowed
}
  CODE          local: 40
}
```

```

    }

ccnrRequest    OPERATION    ::=
{
    ARGUMENT      CcRequestArg
    RESULT        CcRequestRes
    ERRORS        { shortTermRejection | longTermRejection | undefined |
                  supplementaryServiceInteractionNotAllowed
                }
    CODE          local: 27
}

ccCancel       OPERATION    ::=
{
    ARGUMENT      CcArg OPTIONAL TRUE
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE          local: 28
}

ccExecPossible OPERATION    ::=
{
    ARGUMENT      CcArg OPTIONAL TRUE
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE          local: 29
}

ccRingout     OPERATION    ::=
{
    ARGUMENT      CcShortArg OPTIONAL TRUE
    RETURN RESULT FALSE
    ERRORS        {
                  remoteUserBusyAgain | failureToMatch |
                  undefined
                }
    CODE          local: 31
}

ccSuspend     OPERATION    ::=
{
    ARGUMENT      CcShortArg OPTIONAL TRUE
    RETURN RESULT FALSE
    ALWAYS RESPONDS FALSE
    CODE          local: 32
}

ccResume      OPERATION    ::=
{
    ARGUMENT      CcArg OPTIONAL TRUE
    RETURN RESULT FALSE

```

```

        ALWAYS RESPONDS FALSE
        CODE local: 33
    }

CcRequestArg ::= SEQUENCE
{
    numberA      EndpointAddress,
    numberB      EndpointAddress,
    ccIdentifier  CallIdentifier OPTIONAL,
    -- もしあるなら、不成功呼、CC 要求および CC 呼に関連する主な
    -- 構造として使用しなければならない。
    service      BasicService,

    can-retain-service BOOLEAN,
    retain-sig-connection BOOLEAN OPTIONAL,
    extension     SEQUENCE SIZE (0..255) OF MixedExtension
OPTIONAL,
    ...
}

CcRequestRes ::= SEQUENCE
{
    retain-service BOOLEAN,
    extension     SEQUENCE SIZE (0..255) OF MixedExtension
OPTIONAL,
    ...
}

CcArg ::= CHOICE
{
    shortArg CcShortArg, --シグナリング接続が維持された場合
    longArg  CcLongArg,  --接続解放オプションが適用される場合
    ...
}

CcShortArg ::= SEQUENCE
{
    ccIdentifier CallIdentifier OPTIONAL,
    extension     SEQUENCE SIZE (0..255) OF MixedExtension OPTIONAL,
    ...
}

CcLongArg ::= SEQUENCE
{
    numberA      EndpointAddress OPTIONAL,
    numberB      EndpointAddress OPTIONAL,
    ccIdentifier CallIdentifier OPTIONAL,
    service      BasicService OPTIONAL,
    --これらの要素は適切な CC 要求を識別するために使用する。
    --もしあるなら、ccIdentifier は、この目的の主な構造とならなければ
    --ならない。
}

```

```

        extension      SEQUENCE SIZE (0..255) OF MixedExtension OPTIONAL,
        ...
    }

```

```

shortTermRejection    ERROR      ::=
    { CODE             local: 1010}

longTermRejection     ERROR      ::=
    { CODE             local: 1011}

remoteUserBusyAgain  ERROR      ::=
    { CODE             local: 1012}

failureToMatch        ERROR      ::=
    { CODE             local: 1013}

```

END -- of Call-Completion-Operations
 END -- 呼完了操作

13. SS-CCBS の仕様記述言語 (SDL) 図

呼完了シグナリングエンティティの手順は図 19/JT-H450.9 から図 29/JT-H450.9 までの SDL で記述する。SDL は JT-H225.0 コネクション上で伝送される SS-CC の特定情報を示すのみである。JT-H245 手順 (例えば、端末能力交換、マスター/スレーブ決定、論理チャネルのオープンとクローズ等) は示さない。次の略語を使用する。

```

ack   確認応答
BC    基本呼
conn  コネクション
conf  確認
err   Return error APDU
ind   表示
inv   Invoke APDU
rej   Reject APDU or Rejection
res   Return result APDU
sc    シグナリングコネクション
sig   シグナリング

```

SDL と以前の章の本文とが食い違った場合は、本文を優先する。

エンドポイントに代わって SS-CC でゲートキーパ/プロキシが動作するモデルのための特別なゲートキーパ/プロキシ SDL は提供しない。

以下の SDL で使用されるシンボルを図 18/JT-H450.9 に示す。

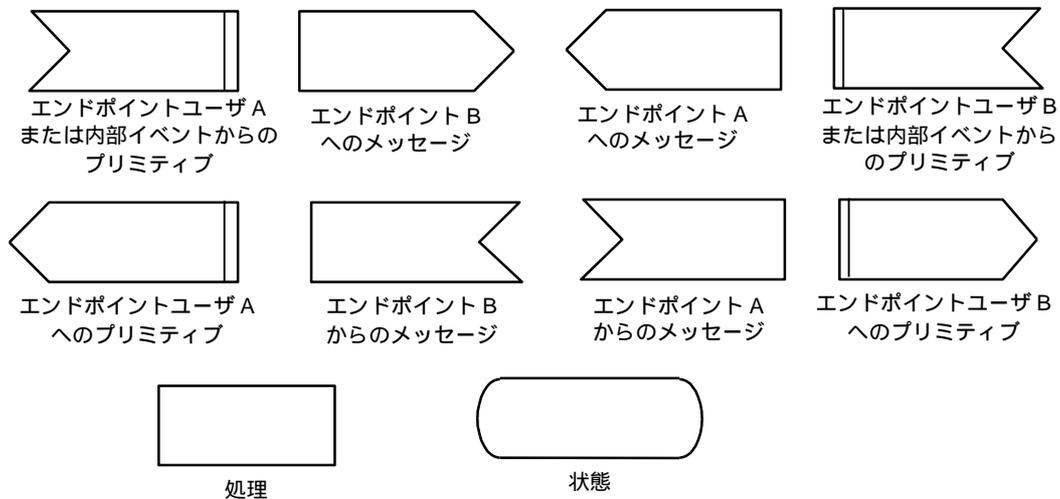


図 18/JT-H450.9
SDL シンボル

13.1. ユーザ A のエンドポイントの動作

図 19/JT-H450.9 から図 25/JT-H450.9 はユーザ A のエンドポイントの動作を示す。

左からの入力シグナルと左への出力シグナルは以下のプリミティブを表す。

- l ユーザ A からまたはユーザ A へ
- l 基本呼制御からまたは基本呼制御へ。これらのプリミティブはプレフィックス BC-で示される
- l 内部シグナル、例えばタイマ満了

右からの入力シグナルと右への出力シグナルは、SS-CC 制御情報を運ぶ同等の SS 制御エンティティ（すなわちユーザ B のエンドポイント）へのメッセージを表す。呼非依存シグナリングコネクションメッセージは 'sc' で示す。'sc' の付いていないメッセージは呼依存である。

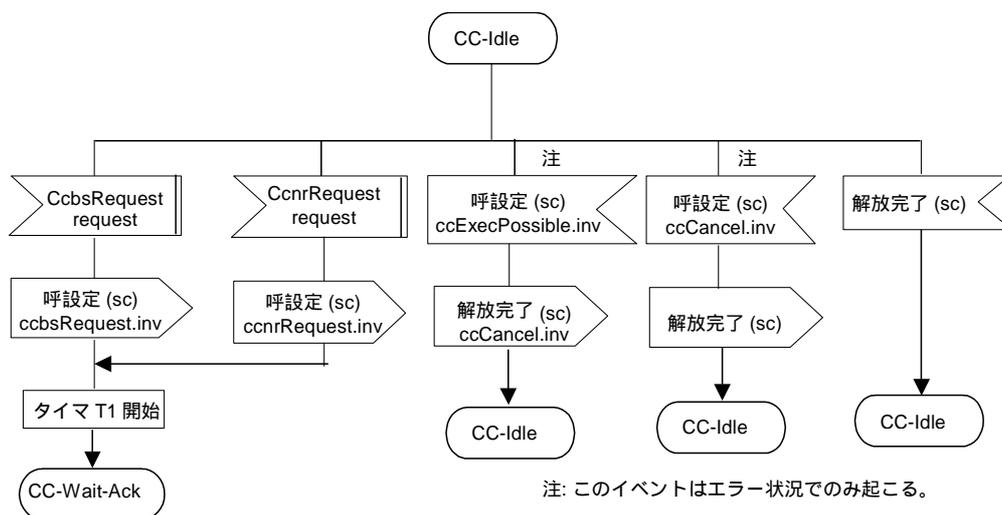


図 19/JT-H450.9 - ユーザ A のエンドポイントの動作

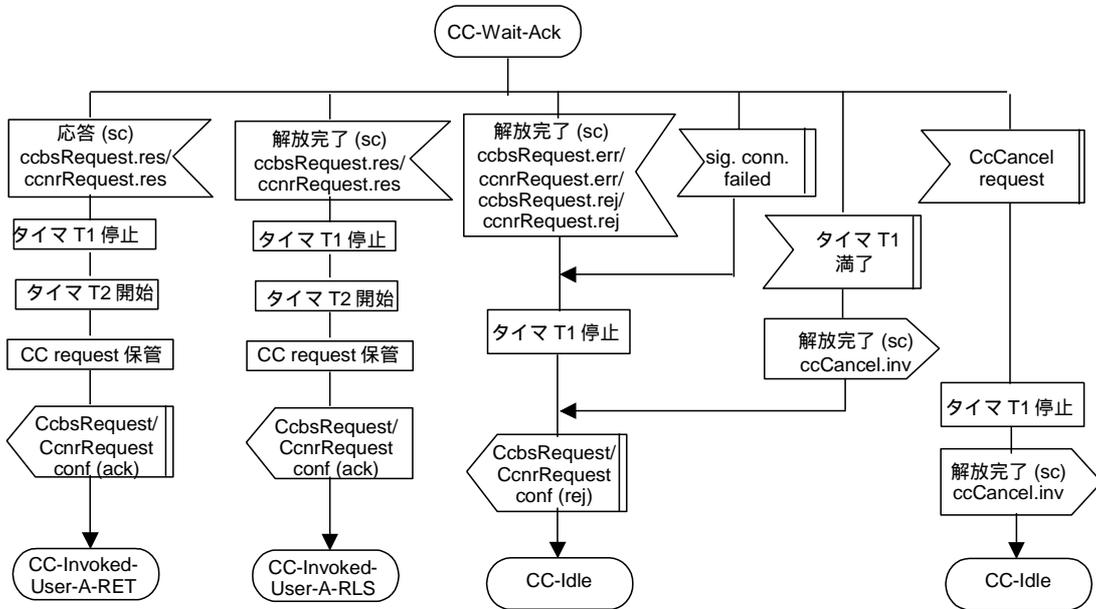


図 20/JT-H450.9 - ユーザ A のエンドポイントの動作 (続き)

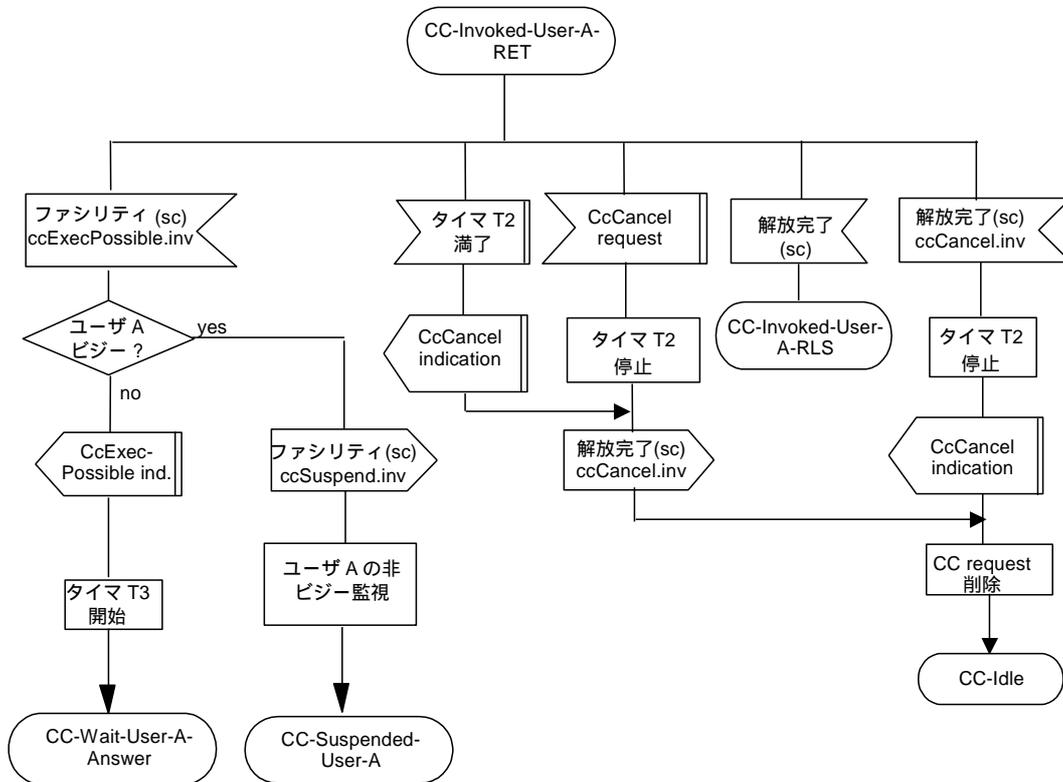


図 21/JT-H450.9 - ユーザ A のエンドポイントの動作 (続き)

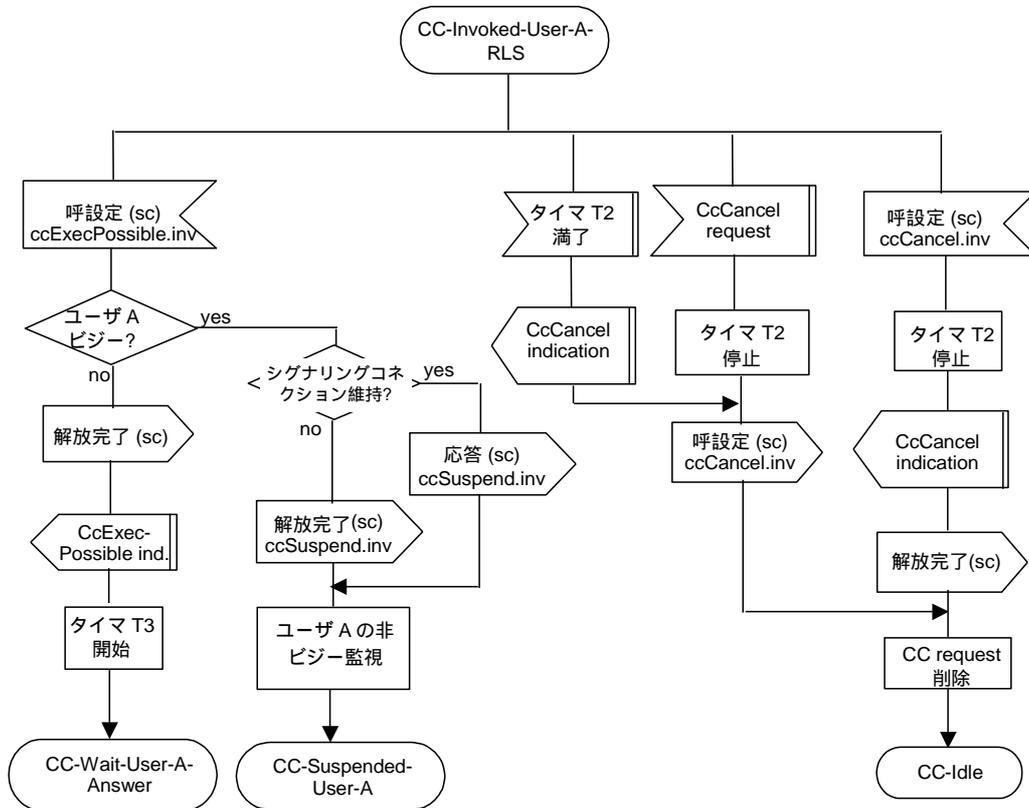


図 22/JT-H450.9 - ユーザ A のエンドポイントの動作 (続き)

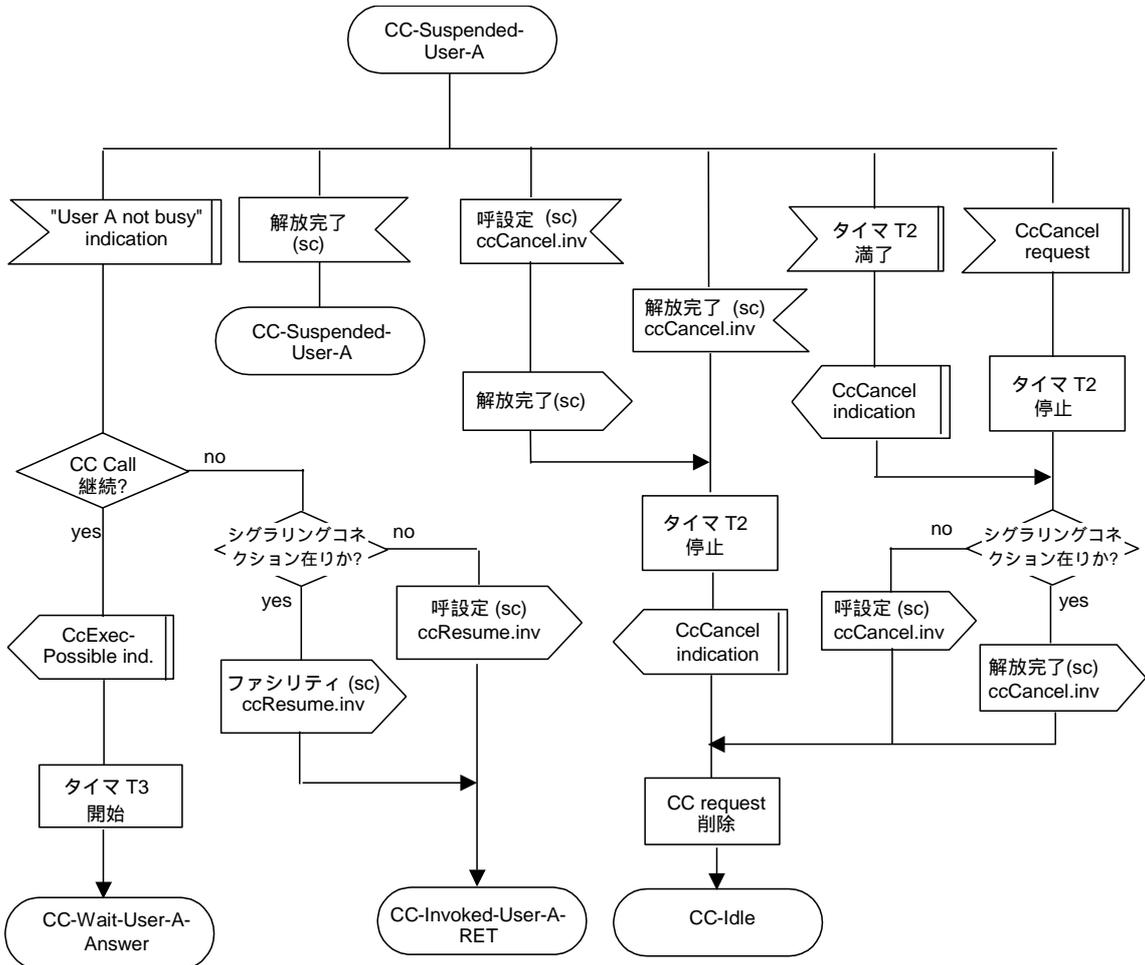


図 23/JT-H450.9 - ユーザ A のエンドポイントの動作 (続き)

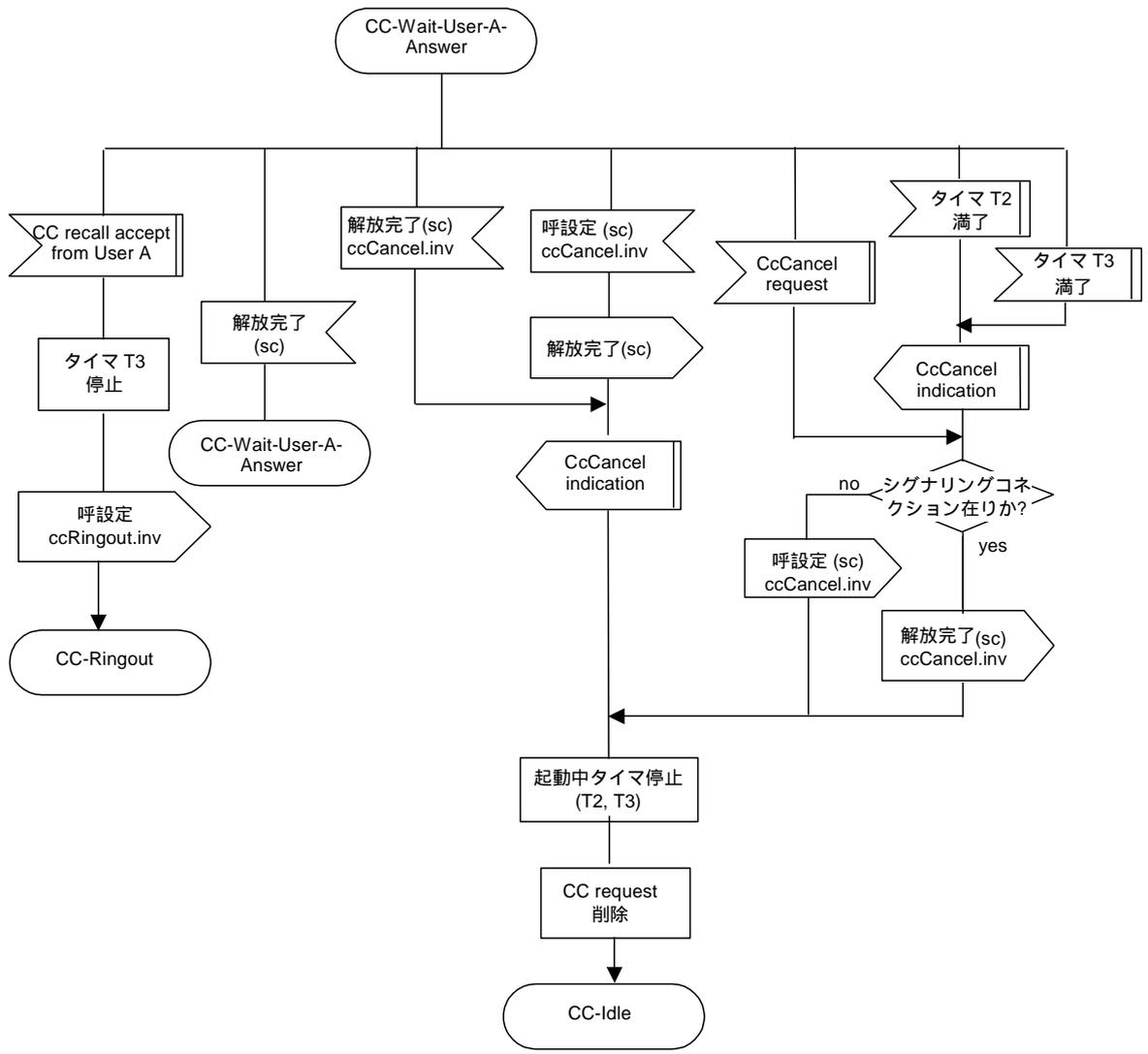


図 24/JT-H450.9 - ユーザ A のエンドポイントの動作 (続き)

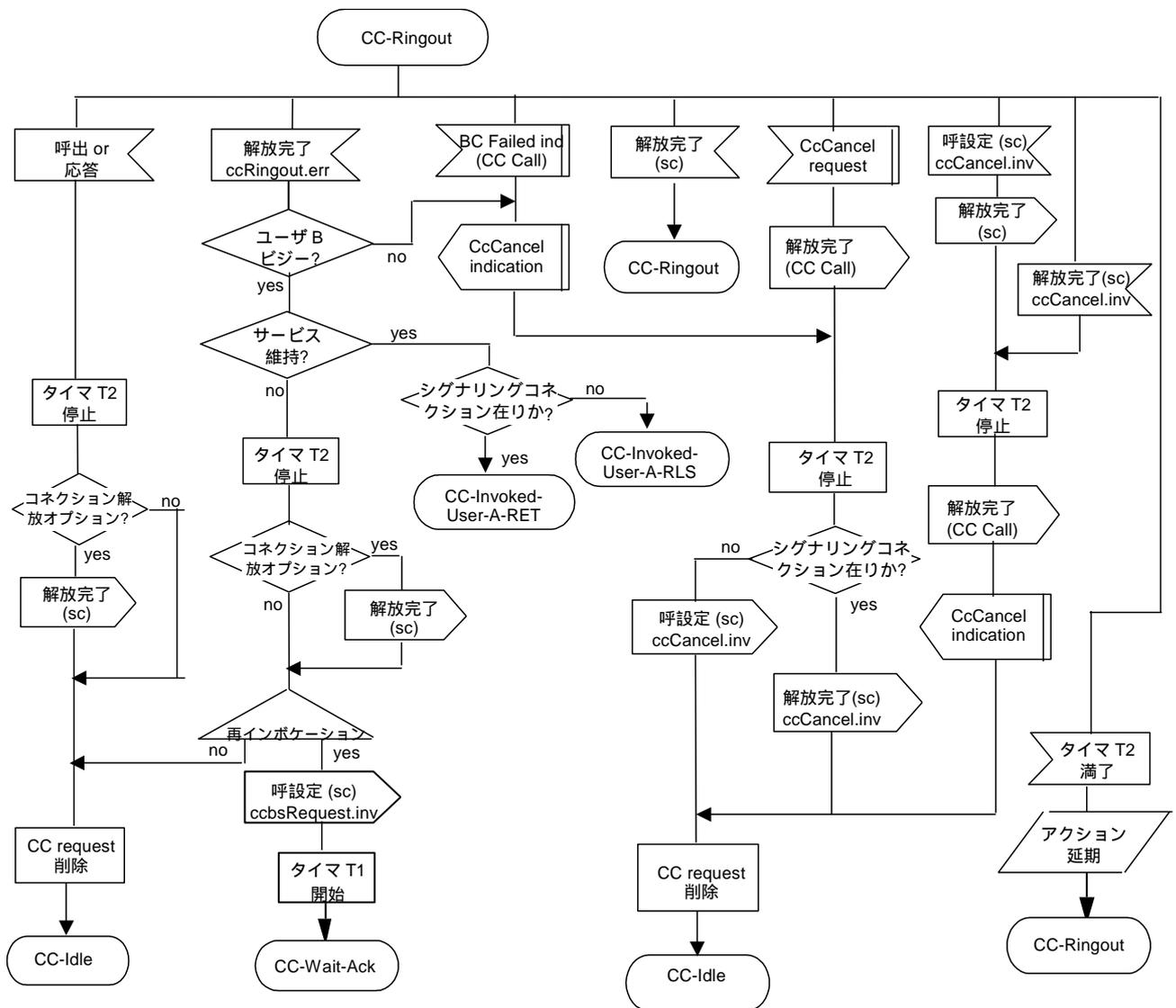


図 25/JT-H450.9 - ユーザ A のエンドポイントの動作 (続き)

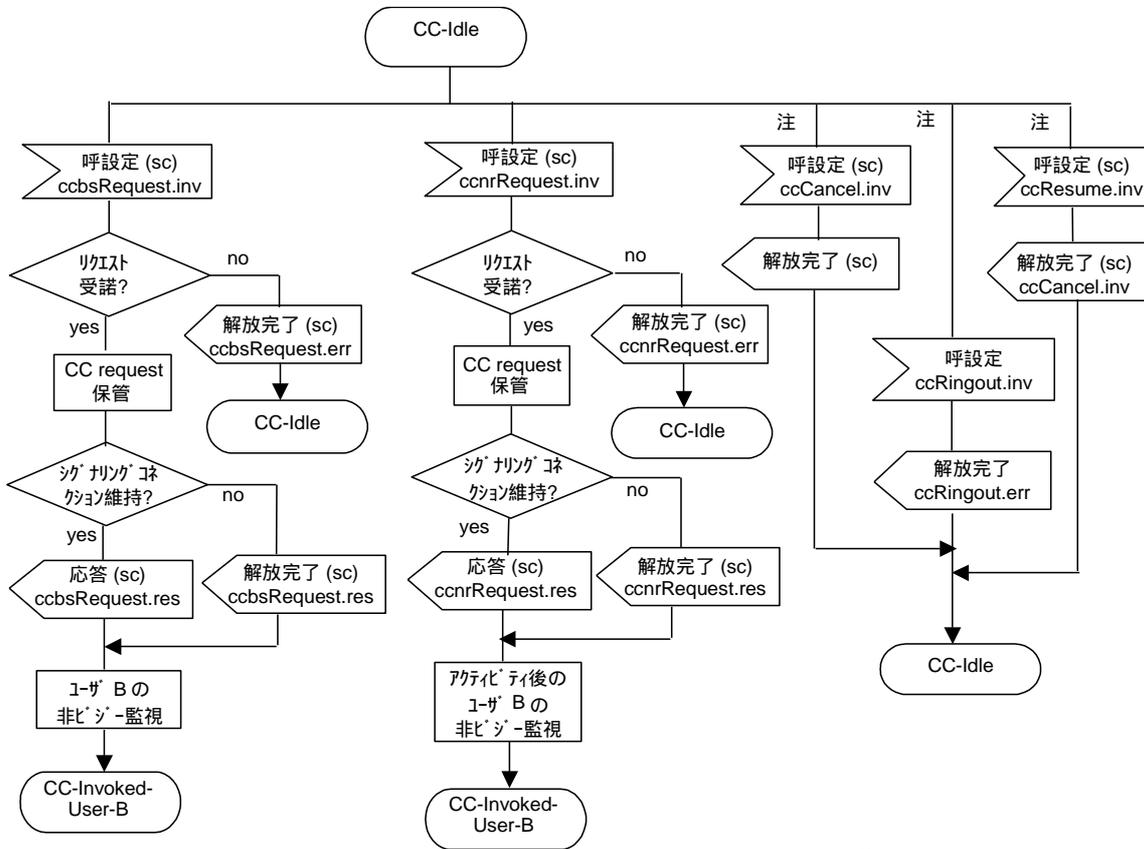
13.2. ユーザ B のエンドポイントの動作

図 26/JT-H450.9 から図 29/JT-H450.9 は、エンドポイント B の動作を示す。

右からの入力シグナルと右への出力シグナルは、SS-CC 制御情報を運ぶ同等の SS 制御エンティティ (すなわちユーザ B のエンドポイント) からと SS-CC 制御情報を運ぶ同等の SS 制御エンティティ (すなわちユーザ B のエンドポイント) へのメッセージを表す。呼非依存シグナリングコネクションメッセージは 'sc' で示す。'sc' の付いていないメッセージは呼依存である。

左からの入力シグナルと左への出力シグナルは以下のプリミティブを表す。

- l 着呼ユーザ(ユーザ B)からまたは着呼ユーザ(ユーザ B)へ
- l 基本呼制御からまたは基本呼制御へ、これらのプリミティブはプレフィックス BC-で示される
- l 内部シグナル



注: このイベントはエラー状況でのみおこる。

図 26/JT-H450.9 - ユーザ B エンドポイントの動作

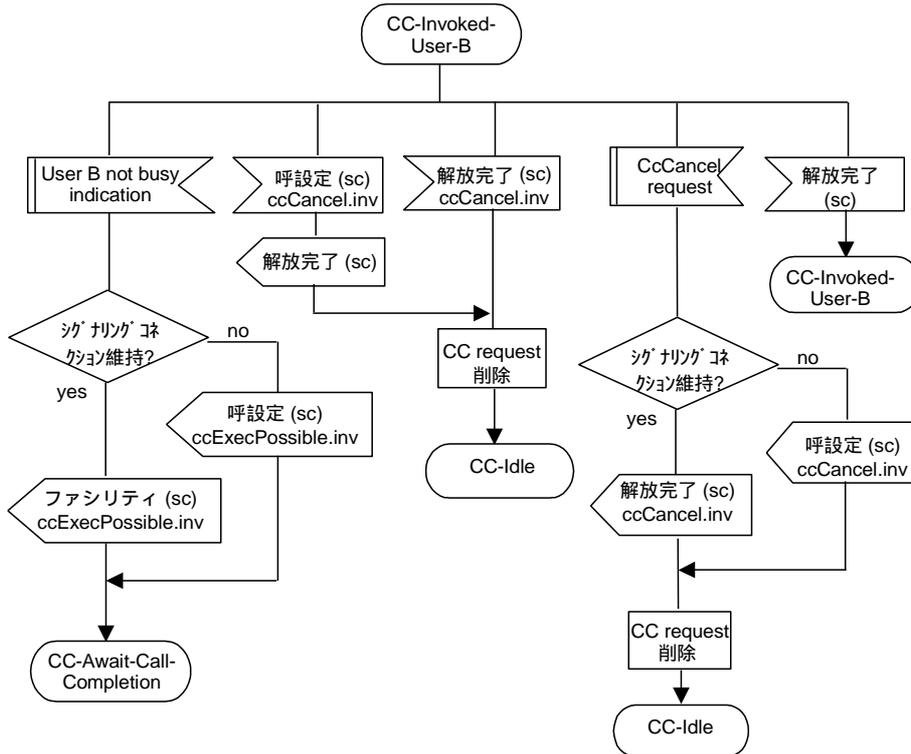


図 27/JT-H450.9 - ユーザ B エンドポイントの動作 (続き)

