

JT-T124
一般的会議制御

[Generic Conference Control]

第1版

1996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．国際勧告等との関連

本標準は、オーディオ グラフィック会議における、一般的会議制御を規定するもので、決議 1 による郵便投票により 1995 年 8 月に承認された ITU-T 勧告 T.124 に準拠したものである。

2．上記国際勧告等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事項

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、削除した項目はない。

2.4 原勧告との章立て構成比較表

本標準では、第 1 章に「概要」を追加し、原勧告第 2 章「参照」を本<参照>中に移動したため、T.124 勧告に対して、「範囲」が 1 章から 2 章に繰り下がっているが、3 章以降は原勧告と同一の章立てとなっている。

3．改版の履歴

版数	制定日	改版内容
第 1 版	1996 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。

5．その他

(1) 参照している勧告、標準等

TTC 標準 : JT-H221、JT-T122、JT-T123、JT-T125

ITU-T 勧告 : F.710、T.35、T.126、T.127、X.200、X.214、X.208、X.209、X.680、X.691

ISO/IEC 標準 : ISO/IEC 10646-1

目次

1. 概要	1
2. 範囲	1
3. 定義	3
3.1. アプリケーション プロトコル.....	3
3.2. アプリケーション プロトコル エンティティ	3
3.3. アプリケーション プロトコル キー	3
3.4. アプリケーション プロトコル セッション.....	3
3.5. アプリケーション レコード	3
3.6. アプリケーション登録.....	3
3.7. アプリケーション名簿.....	4
3.8. アプリケーション ユーザ ID.....	4
3.9. 議長制御モード	4
3.10. 議長.....	4
3.11. 会議.....	4
3.12. 会議アプリケーション名簿.....	4
3.13. 会議プロファイル.....	4
3.14. 会議名簿.....	4
3.15. コントロール GCC サービス アクセス ポイント.....	5
3.16. 召集者	5
3.17. 召集者パスワード.....	5
3.18. デフォルト セッション	5
3.19. エンティティ ID.....	5
3.20. GCC プロバイダ	5
3.21. GCC サービス アクセス ポイント	5
3.22. ハンドル.....	5
3.23. ローカル アプリケーション名簿	5
3.24. MCS ドメイン.....	6
3.25. MCS ドメイン セクレタ	6
3.26. MCS ユーザ ID	6
3.27. 多地点	6
3.28. 多地点制御装置	6
3.29. マルチポート端末.....	6
3.30. ノード	6
3.31. ノード コントローラ.....	6
3.32. ノード ID	6

3.33. 非議長制御モード.....	7
3.34. 参加者.....	7
3.35. パスワード.....	7
3.36. 同位アプリケーション プロトコル エンティティ.....	7
3.37. リソース.....	7
3.38. セッション ID.....	7
3.39. セッション キー.....	7
3.40. 端末.....	7
3.41. トップ GCC プロバイダ.....	8
3.42. ユニコード.....	8
3.43. ユニコードの区 00.....	8
3.44. ユーザ アプリケーション.....	8
4. 省略形.....	9
5. 規則.....	10
6. 一般的会議制御の概要.....	11
6.1. 会議ノードのためのシステムモデル.....	12
6.2. 会議の確立と終了.....	13
6.3. 会議名簿.....	14
6.4. アプリケーション名簿.....	14
6.5. アプリケーションレジストリ.....	14
6.6. 会議議長権.....	14
6.7. その他の機能.....	15
6.8. GCC 抽象サービスのまとめ.....	15
7. GCC サービスの定義.....	21
7.1. 会議の確立と終了.....	21
7.1.1. 会議プロファイル (Conference Profile).....	21
7.1.2. 抽象サービスの記述.....	21
7.1.2.1. GCC-CONFERENCE-CREATE.....	22
7.1.2.2. GCC-CONFERENCE-QUERY.....	26
7.1.2.3. GCC-CONFERENCE-JOIN.....	29
7.1.2.4. GCC-CONFERENCE-INVITE.....	34
7.1.2.5. GCC-CONFERENCE-ADD.....	37
7.1.2.6. GCC-CONFERENCE-LOCK.....	41
7.1.2.7. GCC-CONFERENCE-UNLOCK.....	42
7.1.2.8. GCC-CONFERENCE-LOCK-REPORT.....	43
7.1.2.9. GCC-CONFERENCE-DISCONNECT.....	44
7.1.2.10. GCC-CONFERENCE-TERMINATE.....	45

7.1.2.11. GCC-CONFERENCE-EJECT-USER	47
7.1.2.12. GCC-CONFERENCE-TRANSFER.....	48
7.1.3. 会議確立条件.....	50
7.1.4. 会議確立手順の例.....	50
7.1.4.1. Meet-me 会議確立.....	51
7.1.4.2. Call-out 会議確立.....	52
7.1.4.3. Call-through 会議確立	52
7.1.4.4. ポイント ポイント(Point-to-Point)会議確立	52
7.1.4.5. マルチポート端末間の会議確立	53
7.2. 会議名簿.....	54
7.2.1. サービス概要記述.....	55
7.2.1.1. GCC-CONFERENCE-ANNOUNCE-PRESENCE.....	55
7.2.1.2. GCC-CONFERENCE-ROSTER-REPORT.....	57
7.2.1.3. GCC-CONFERENCE-ROSTER-INQUIRE	59
7.3. アプリケーション名簿.....	60
7.3.1. アプリケーション名簿の内容	60
7.3.2. アプリケーション名簿交換手順.....	64
7.3.3. サービスの概要記述	66
7.3.3.1. GCC-APPLICATION-PERMISSION-TO-ENROLL	67
7.3.3.2. GCC-APPLICATION-ENROLL.....	68
7.3.3.3. GCC-APPLICATION-ROSTER-REPORT	71
7.3.3.4. GCC-APPLICATION-ROSTER-INQUIRE.....	75
7.3.3.5. GCC-APPLICATION-INVOKE.....	76
7.4. アプリケーション レジストリ	79
7.4.1. レジストリ キー.....	79
7.4.2. 所有権と持続性	80
7.4.3. 動的な割り当て	80
7.4.4. サービス概要記述.....	81
7.4.4.1. GCC-REGISTRY-REGISTER-CHANNEL	81
7.4.4.2. GCC-REGISTRY-ASSIGN-TOKEN	82
7.4.4.3. GCC-REGISTRY-SET-PARAMETER	84
7.4.4.4. GCC-REGISTRY-RETRIEVE-ENTRY	85
7.4.4.5. GCC-REGISTRY-DELETE-ENTRY.....	87
7.4.4.6. GCC-REGISTRY-MONITOR	88
7.4.4.7. GCC-REGISTRY-ALLOCATE-HANDLE.....	89
7.5. 会議議長権	90
7.5.1. サービスの概要記述	91
7.5.1.1. GCC-CONDUCTOR-ASSIGN	92
7.5.1.2. GCC-CONDUCTOR-RELEASE.....	93

7.5.1.3. GCC-CONDUCTOR-PLEASE.....	94
7.5.1.4. GCC-CONDUCTOR-GIVE.....	95
7.5.1.5. GCC-CONDUCTOR-INQUIRE.....	96
7.5.1.6. GCC-CONDUCTOR-PERMISSION-ASK.....	97
7.5.1.7. GCC-CONDUCTOR-PERMISSION-GRANT.....	99
7.6. その他の機能.....	100
7.6.1. サービスの概要記述.....	100
7.6.1.1. GCC-CONFERENCE-TIME-REMAINING.....	100
7.6.1.2. GCC-CONFERENCE-TIME-INQUIRE.....	101
7.6.1.3. GCC-CONFERENCE-EXTEND.....	102
7.6.1.4. GCC-CONFERENCE-ASSISTANCE.....	103
7.6.1.5. GCC-TEXT-MESSAGE.....	104
8. GCC プロトコル仕様.....	106
8.1. 一般操作.....	106
8.2. 会議の確立と終了.....	107
8.2.1. 会議生成 (Conference Creation).....	107
8.2.2. 会議問合せ (Querying Conference).....	112
8.2.3. 会議への加入.....	115
8.2.4. 会議へのノードの招待.....	123
8.2.5. 会議へノードを追加するための要求.....	128
8.2.6. 会議のロック.....	130
8.2.7. 会議のアンロック.....	132
8.2.8. 会議からの切断.....	133
8.2.9. 会議の終了.....	134
8.2.10. 会議からのノードの排除.....	136
8.2.11. 会議間のノードの移動.....	138
8.3. 会議名簿とアプリケーション名簿.....	141
8.3.1. 概要.....	141
8.3.2. ノードの会議への参加.....	143
8.3.3. アプリケーション プロトコル エンティティの登録.....	145
8.3.4. 会議名簿エントリの更新.....	147
8.3.5. トップGCC プロバイダへの名簿更新の伝搬.....	147
8.3.6. 会議及びアプリケーション名簿の分配.....	149
8.3.7. ノードの会議からの退場.....	150
8.3.8. 集約アプリケーション能力リスト.....	150
8.3.9. アプリケーション及び会議名簿問い合わせ.....	151
8.3.10. 名簿更新の例.....	151
8.3.11. アプリケーション プロトコル エンティティの遠隔起動.....	152
8.4. アプリケーション レジストリ.....	153

8.4.1. チャンネルの登録.....	153
8.4.2. トークンの割当て.....	155
8.4.3. パラメータのセット.....	157
8.4.4. エントリの検索.....	159
8.4.5. エントリの削除.....	159
8.4.6. エントリの監視.....	161
8.4.7. ユニークなハンドルの割当て.....	163
8.4.8. 所有権の変更とレジストリの整理.....	164
8.5. 会議議長権.....	164
8.5.1. 議長権の獲得.....	164
8.5.2. 議長権の解放.....	165
8.5.3. 議長割り当てと解放の指示.....	165
8.5.4. 議長権の委譲依頼.....	166
8.5.5. 議長権の委譲.....	166
8.5.6. 議長権の状態の取得.....	167
8.5.7. 新ノードの会議参加時の議長権アナウンス.....	168
8.5.8. 予期せぬ議長の切断.....	168
8.5.9. 議長制御モード許可の問い合わせ.....	168
8.5.10. 議長制御モード許可の授与.....	169
8.6. その他の機能.....	170
8.6.1. 時間制約のある会議.....	170
8.6.2. 会議アシスタンスの要求.....	171
8.6.3. テキストメッセージの放送.....	172
8.7. GCCPDU 定義.....	172
9. 多地点通信サービスの利用.....	193
9.1. MCS サービス.....	193
9.2. チャンネル割り当て.....	194
9.3. トークン割り当て.....	195
9.4. MCS データ伝送サービスの使用.....	195
9.5. MCS プリミティブ中の PDU のコード化.....	199
9.6. MCS-CONNECT-PROVIDER のユーザデータパラメータのフォーマット.....	199
9.7. MCS ドメインセレクタの割り込み.....	199

1.概要

この標準は、マルチメディア端末と多地点制御装置(MCU)の会議管理と制御に対する高水準フレームワークを提供する。この標準は、会議の開始/終了、会議参加端末登録管理、アプリケーションとアプリケーション能力管理、会議議長の調整、その他多くの機能を持つ一般的会議制御(GCC)機能を含んでいる。T.120 シリーズの下位構造は関連標準 JT-T122/JT-T125(MCS)と JT-T123 に依存する。

2.範囲

この標準は、オーディオ グラフィックおよびオーディオ ビジュアル端末の会議管理と制御および多地点制御装置(MCU)に対して高水準のフレームワークを提供する。それは、会議の開始/終了、会議参加端末登録管理、アプリケーションとアプリケーション能力管理、会議議長の調整、その他多数の機能を持つ一般的会議制御(GCC)機能を含んでいる。

この標準は、フレームワーク勧告 T.120 の中で定義される。このフレームワークに含まれる標準である関連 TTC 標準 JT-T122 と JT-T125 は、この標準において使用される多地点通信メカニズムを規定し、TTC 標準 JT-T123 はオーディオビジュアルプロトコルスタックを規定する。

図 2-1/JT-T124 は、TTC 標準 JT-T124 とその関連標準の構成を示す。

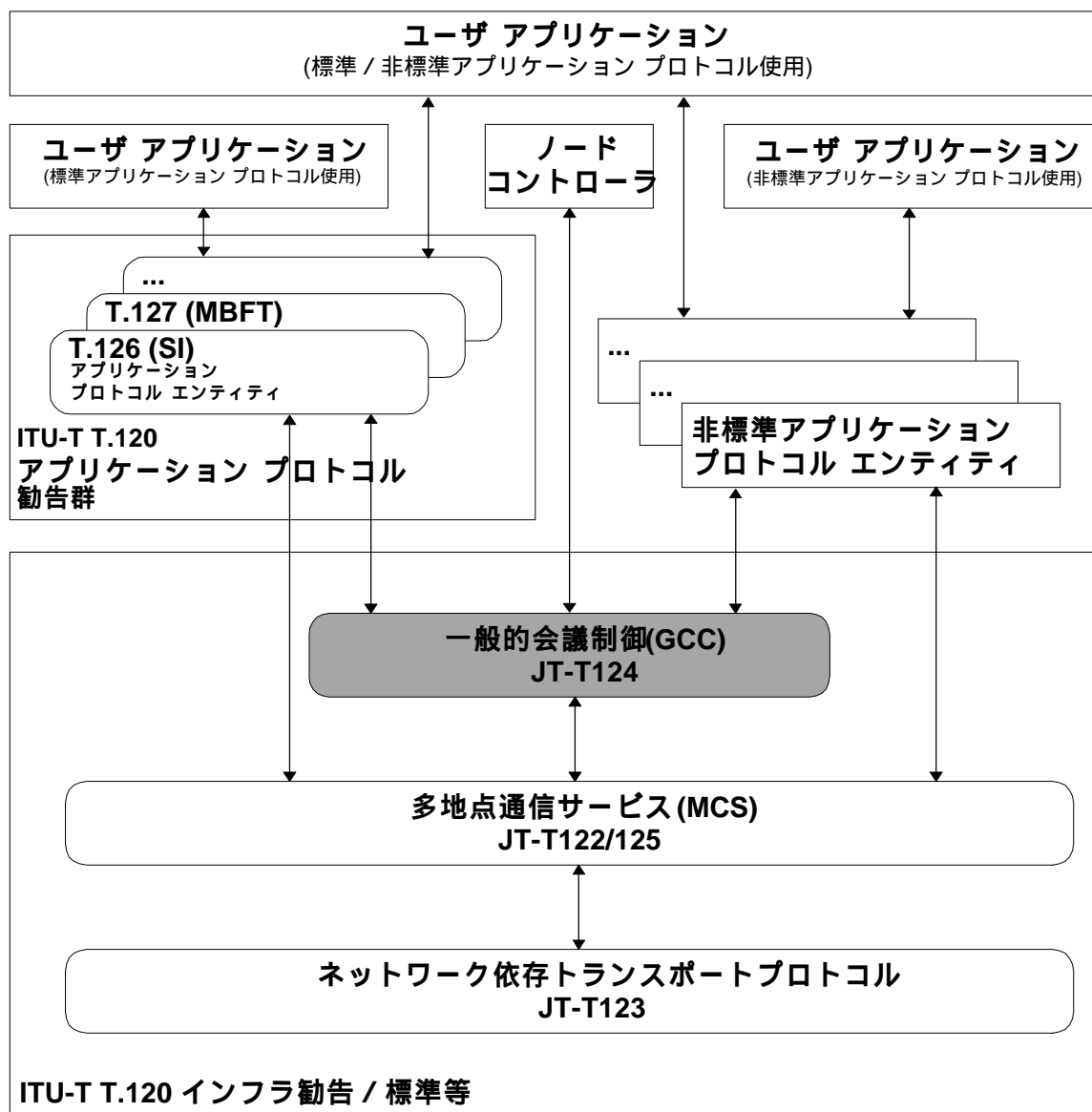


図 2-1/JT-T124 JT-T124 の範囲
(ITU-T T.124)

3.定義

この標準のために、以下の定義が当てはまる。

3.1.アプリケーション プロトコル

T.120 サービスを使用するために設計された標準または非標準プロトコル規定。

3.2.アプリケーション プロトコル エンティティ

端末またはMCUのアプリケーション プロトコルの実体。アプリケーション プロトコル エンティティは、ユーザ アプリケーションによって使用されているが、それ自身はユーザ アプリケーションでない。アプリケーション プロトコル エンティティは、GCC プロバイダがローカルな端末やMCUで提供するGCCSAPを経由してGCCと通信する。アプリケーション プロトコル エンティティだけが、GCC プロバイダと通信し、ユーザ アプリケーションは通信しない。同じアプリケーション プロトコルに基づくアプリケーション プロトコル エンティティは、一つのノードで複数登録可能である。これらは同一または別々のアプリケーション プロトコル セッションであるかもしれない。一つのアプリケーション プロトコル エンティティは、一つのGCCSAPを経由するローカルなGCC プロバイダと通信すると仮定され、また、活性状態で登録するならば、一つのMCS ユーザ ID を持つと仮定される。活性状態で登録したアプリケーション プロトコル エンティティは、そのセッション ID によって示される一つのアプリケーション プロトコル セッションの一部である（あるいは、セッション ID なしである場合デフォルト セッションの一部）。非活性状態で登録したアプリケーション プロトコル エンティティはアプリケーション プロトコル セッションの一部として考慮されない。しかし、非活性なアプリケーション プロトコル エンティティはGCC サービスを使用するかもしれない。そして、そのようなアプリケーション プロトコル エンティティと関連する情報はアプリケーション名簿に含められる。

3.3.アプリケーション プロトコル キー

アプリケーション プロトコル キーは、アプリケーション プロトコル セッションのためにアプリケーション プロトコルの種類を識別する。複数の同じ型のアプリケーション プロトコル セッションは、同じアプリケーション プロトコル キー（しかし、異なるセッション ID）を使って識別される。アプリケーション プロトコル キーは、勧告または標準のASN.1 オブジェクト識別子、または非標準プロトコル、あるいはTTC 標準 JT-H221 の符号化規則を利用する非標準識別子である。

3.4.アプリケーション プロトコル セッション

一組の同位アプリケーション プロトコル エンティティ。

3.5.アプリケーション レコード

特定のノードにおける特定のアプリケーション プロトコル エンティティのための一組の情報。このセットは、アプリケーション ユーザ ID、活性/非活性のフラグおよび他のパラメータを含む。

3.6.アプリケーション登録

中央の格納装置はトップGCC プロバイダに位置しており、そこでは、アプリケーション プロトコル エンティティはトークン、チャンネルと他のパラメータの使用を登録することができる。同位アプリケーション プロトコル エンティティは、この登録した情報を見つけるためにアクセスすることができる。

3.7.アプリケーション名簿

会議中の全ノード上の全てのアプリケーション プロトコル エンティティが登録された、全てのアプリケーション レコードのセットで、各アプリケーション プロトコル セッションのためのアプリケーション能力リストを含む。

3.8.アプリケーション ユーザ ID

MCS によってアプリケーション プロトコル エンティティに割り当てられた MCS ユーザ ID。

3.9.議長制御モード

議長制御モードでは、議長ノードは、会議における全てのノードのアプリケーション プロトコル エンティティを制御し、他のノードによるアプリケーション プロトコル エンティティの動作を制限することができる。議長が会議に割り当てられたとき、議長制御モードで会議が確立される。ノードがうまく議長トークンをつかんだ時、議長制御モードが確立する。

3.10.議長

議長は、もし存在すれば、会議のいくつかの側面を制御するノードである(例えば、アプリケーション プロトコル エンティティ間の通信制御、会議参加者に対する制御、会議の終了)。会議には0または一つの議長がいる。会議では、議長がいる場合といない場合がある。ノードは議長トークンを獲得することにより、または現在の議長から議長権を要求または受諾することによって議長になる。

3.11.会議

いっしょに結合したり、いろいろな通信網を経由するオーディオ グラフィックとオーディオ ビジュアル情報を交換する能力を持つ、多くのノード。

3.12.会議アプリケーション名簿

一組のアプリケーション レコードから成っている各 GCC プロバイダにより維持されるデータベースで、会議中の全てのノードでの全てのアプリケーション プロトコル エンティティのためのデータベース、および各アプリケーション プロトコル セッションのためのアプリケーション能力リストのような他の情報である。

3.13.会議プロファイル

会議名、パスワード(もしあれば)のような会議に関係がある情報から成っている各 GCC プロバイダにより維持されるデータベース。

3.14.会議名簿

会議におけるノードリストから成っている各 GCC プロバイダにより維持されるデータベース。各ノードに対して、このリストは、そのノードのノード ID、ノードの型と名前を含み、また、ノードにて参加者のリストおよび他のオプション情報を含むかもしれない。

3.15.コントロール GCC サービス アクセス ポイント

一つのノード内での GCC プロバイダとノード コントローラ間の通信インタフェース。

3.16.召集者

GCC-CONFERENCE-CREATE-要求プリミティブを発行して会議を生成したノード。

3.17.召集者パスワード

会議の召集者が会議から退席しその後再参加することが許される会議において、召集者特権を回復するために使用されるパスワードで、数字列およびオプションの文字列。召集者パスワードは会議が生成されている間（二つの可能な形式の内の一つで）は、それが可能であるように含まれる。正しい召集者パスワードの使用は、ロックされた会議に接続することを許可する。しかし、パスワードで保護された会議に参加する時、正しいパスワードを指定する必要がある。

3.18.デフォルト セッション

セッション キーにセッション ID のない活性なアプリケーション プロトコル エンティティは、別々のユニークなセッションの部分でデフォルト セッションとみなされる。

3.19.エンティティ ID

ノードに登録された各アプリケーション プロトコル エンティティを確認するために使用される 16 ビットの識別子。エンティティ ID の値は、一つのノードの内ですべての型のアプリケーション プロトコル エンティティの中でユニークである。ノード間ではユニークである必要はない。会議中の特別なアプリケーション プロトコル エンティティは、アプリケーション プロトコル エンティティが登録されたノードに対応した、エンティティ ID とノード ID の組み合わせによってユニークに識別される。

3.20.GCC プロバイダ

端末および MCU におけるローカルなノード コントローラとアプリケーション プロトコル エンティティに GCC サービスを提供しているエージェント。

3.21.GCC サービス アクセス ポイント

一つのノードの内の GCC プロバイダとアプリケーション プロトコル エンティティ間の通信インタフェース。

3.22.ハンドル

GCC-REGISTRY-ALLOCATE-HANDLE-要求プリミティブを利用して、トップ GCC プロバイダにより割り当てられた 32 ビット整数。この数は会議内でユニークであると保証されている。

3.23.ローカル アプリケーション名簿

GCC プロバイダでローカルに登録された各アプリケーション プロトコル エンティティのために、アプリケーション能力リストのような、一つのアプリケーション レコードと他の情報から成っている個々の GCC プロバイダにより維持されるデータベース。これは、会議アプリケーション名簿を確定するために、他のノードと交換された情報を記録するために使用される。

3.24.MCS ドメイン

ノードの間の MCS 接続の階層構造。ノードは、一つのドメイン内での通信には MCS サービスを使用できるが、別々のドメイン間では使用できない。一つの GCC 会議は、一つの MCS ドメインに 1 対 1 で対応する。

3.25.MCS ドメイン セクレタ

MCS ドメインのローカルでユニークな識別子。

3.26.MCS ユーザ ID

MCS により MCS ユーザに割り当てられたユニークな識別番号。アプリケーション プロトコル エンティティと同様、GCC プロバイダは MCS ユーザである。GCC プロバイダに割り当てられた MCS ユーザ ID は、ノード ID として参照される。アプリケーション プロトコル エンティティに割り当てられた MCS ユーザ ID は、アプリケーション ユーザ ID として参照される。MCS ユーザ ID は一つの MCS ドメイン内でのみ有効である。

3.27.多地点

直接接続された二つのノード間でデータが交換されるポイント ポイントと比較して、同時に複数のノード間でデータを交換する能力。

3.28.多地点制御装置

一般的に MCU またはブリッジと呼ばれるマルチポート装置で、多地点機能において端末と他の MCU を接続するサービスを提供する。GCC 能力のある MCU は GCC と MCS を実行する。MCU は、元々、ユーザ通信のための終端ではない。

3.29.マルチポート端末

勧告 T.120 情報をブリッジする能力を持った終端のオーディオ グラフィックまたはオーディオ ビジュアル装置。マルチポート端末の動作は、一つの会議の自動的な確立によって典型づけられる。しかし、MCU のようなマルチポート端末の中で与えられた会議においては、二つ以上の MCS コネクションがあるかもしれない。

3.30.ノード

端末、マルチポート端末または MCU。一つのノードは、一つの GCC プロバイダを含む。一つのノードは、一つ以上の物理的な装置から成る。同様に、一つの物理的な装置は、いくつかの論理的なノードをホストする。

3.31.ノード コントローラ

ノードのコントローラとして動作する、それぞれの端末または MCU に対して存在する機能上のエンティティ。

3.32.ノード ID

ノードにおいて MCS によって GCC プロバイダに割り当てられた MCS ユーザ ID。

3.33.非議長制御モード

議長を持たない会議モード。

3.34.参加者

ノードにおいて会議に参加している人。

3.35.パスワード

会議が生成されたときに特定される数字列とオプションの文字列。会議が生成され会議に加わる場合、パスワードを用いた会議では、ノードが会議への参加を許可されるために、ノードはこのパスワード（二つの可能な形式の内の一つで）を GCC-CONFERENCE-JOIN プリミティブ中に含めなければならない。

3.36.同位アプリケーション プロトコル エンティティ

セッション キーのセッション ID 部分を含み、同一のセッション キーを使用して活性状態で登録するアプリケーション プロトコル エンティティ。同位アプリケーション プロトコルは、会議の間にお互いに通信する。

3.37.リソース

会議中にノードによって使用され、また共有されるもの。リソースはチャンネルとトークンの両方から成る。

3.38.セッション ID

セッション キーに含まれているオプション パラメータで、同じアプリケーション プロトコルに基づく同位アプリケーション プロトコル エンティティの複数の組を区別するために使われる。共通のアプリケーション プロトコルと、同一のセッション ID の使用によって定義される同位アプリケーション プロトコル エンティティのそれぞれの組は、別々に通信を行う。セッション キー中に含まれた、セッション ID のない活性なアプリケーション プロトコル エンティティは、別々にユニークなセッションをデフォルト セッションとみなす。一組の同位アプリケーション プロトコル エンティティはアプリケーション プロトコル セッションと呼ばれる。セッション ID は、MCS チャンネル ID の形になっている。

3.39.セッション キー

同位アプリケーション プロトコル エンティティに共通の識別子。一番目の構成要素であるアプリケーション プロトコル キーは、アプリケーション プロトコルの種類を識別する。二番目の構成要素は、セッション キーのオプションであるセッション ID であり、それは、アプリケーション プロトコルのために特定のセッションを識別する。（セッション ID がない場合は、デフォルト セッションである。）全セッション キーが共通であり、またセッション ID とアプリケーション プロトコル キーを含んだアプリケーション プロトコル エンティティは、同位アプリケーション プロトコル エンティティとみなされる。

3.40.端末

終端のオーディオ グラフィックまたはオーディオ ビジュアルの装置。GCC 能力のある端末は GCC と MCS を実行する。会議内では、端末は一つの MCS コネクションに限られている。

3.41. トップ GCC プロバイダ

会議において、トップ MCS プロバイダと共存する GCC プロバイダ。トップ GCC プロバイダは、会議において他の GCC プロバイダには必要とされない責任を持つ。トップ GCC プロバイダの位置は、会議の間、変更されない。

3.42. ユニコード

ISO/IEC 10646-1(JISX0221)において定義されるような多言語を用いた文字列フォーマット。

3.43. ユニコードの区 00

コントロール キャラクタと予約コードに加えて、ベーシック ラテンおよびラテン - 1 補足キャラクタのセットを含む、256 のコード位置から成るユニコードのサブセット。

3.44. ユーザ アプリケーション

一つ以上のアプリケーション プロトコル エンティティを利用するエンティティ。

ユーザ アプリケーションは、エンドユーザへ渡すような同位アプリケーション プロトコル エンティティ間での情報の解釈に影響を及ぼさない範囲に動作が限られている。したがって、ユーザ アプリケーションは、それらが使用するアプリケーション プロトコル エンティティと同位アプリケーション プロトコル エンティティとの間で通信するために、標準勧告のような規格書を必要としない。

4.省略形

CSDN	Circuit Switched Digital Network (回線交換デジタル網)
GCC	Generic Conference Control (一般的会議制御)
GCCSAP	Generic Conference Control Service Access Point (GCC サービス アクセス ポイント)
ISDN	Integrated Services Digital Network (サービス統合デジタル網)
MCS	Multipoint Communication Service (多地点通信サービス)
MCU	Multipoint Control Unit (多地点制御装置)
PDU	Protocol Data Unit (プロトコル データ ユニット)
PSDN	Packet Switched Data Network (パケット交換データ網)
PSTN	Public Switched Telephone Network (公衆電話網)

以下の省略形が、この標準の中で使われる。これは TTC 標準 JT-T122 の中で定義される。

MCS	Multipoint Communication Service (多地点通信サービス)
MCSAP	Multipoint Communication Service Access Point (MCS サービス アクセス ポイント)

5.規則

この標準の中で定義された抽象的なサービスのプリミティブ パラメータは、以下のキーを使用する。

- M: パラメータは必須
- C: パラメータは条件つき
- U: パラメータはオプション
- 空白: パラメータは無
- (=): パラメータの値は前述のプリミティブの対応するパラメータと同一である ;
要求、指示、応答、確認の順序

- (=RQ): パラメータの値は前述のプリミティブの対応するパラメータと同一である ;
ここで、RQ =要求、IN =指示、RS =応答、CF =確認

プリミティブは、4種類の型に分類される：要求、指示、応答、確認。いくつかのプリミティブは全ての型を持つ。これらの4種類の型は以下に定義される。

要求プリミティブ：ある手順を起動するために、ノード コントローラまたはアプリケーション プロトコル エンティティから発行されるものを表現したもの。

指示プリミティブ：要求プリミティブの結果として、または GCC 自身である手順を起動した結果として、GCC プロバイダから発行されるものを表現したもの。

応答プリミティブ：応答を必要とする指示プリミティブに対する応答で、ノード コントローラまたはアプリケーション プロトコル エンティティから発行されるものを表現したもの。

確認プリミティブ：要求プリミティブによって以前に起動された手順が完了したことを表示するために、GCC プロバイダから発行されるものを表現したもの。

PDU は、3種類の型に分類される。PDU の名前は、PDU の使用を表す「要求」「指示」「応答」という語を含んでいる。これらは、以下のように定義される。

要求 PDU: 応答として、応答 PDU を要求するもの。もし、要求が受信ノードによってサポートされない機能に対するものであるならば（例えばオプションや非標準 PDU）、一般的な応答 PDU である `FunctionNotSupportedResponse` が、要求ノード（要求ノードの `NODE-ID-CHANNEL` へ送信される）へ応答を返すために使用される。

指示 PDU: 応答を必要としないもの（例えば情報伝達目的のため）。

応答 PDU: 要求ノードの `NODE-ID-CHANNEL` へ送信されるために、要求 PDU に対して応答するもの。

6. 一般的会議制御の概要

オーディオビジュアル会議サービスのコンテキスト内で、会議は、「参加し、様々な通信網を介してオーディオ グラフィックとオーディオ ビジュアルの情報を交換できる一つのグループの地理的に分散されたノード」と参照される。会議に参加している参加者は、音声（電話技術）だけ、音声とデータ（オーディオ グラフィック）、音声とビデオ（オーディオ ビジュアル）、さらに、音声、ビデオ、およびデータ（マルチメディア）などの様々なタイプのメディア処理機能にアクセスできる。

F、G、H、およびTシリーズ勧告/標準は、既存の通信ネットワークを介した、ポイント ポイントにおける音声、ビデオとグラフィックの相互作用のフレーム ワークを提供する。それらは、また MCU によって相互接続される同じ会議においてその機能を3つ以上の端末に提供する。

この標準は会議管理、オーディオ グラフィックおよびオーディオ ビジュアルの端末またはMCUのコントロールのための高水準のフレーム ワークを提供する。これは、会議の確立とコントロールのメカニズムを提供するために、関連 TTC 標準 JT-T122、JT-T125（MCS）および JT-T123（AVPS）と共存する。TTC 標準 JT-T122、JT-T123、JT-T124 および JT-T125 は、十分に機能する端末またはMCUを開発するための、標準の最小セットを成形する。

この標準は、以下の一般的会議制御（GCC）の機能コンポーネントを含んでいる：

- 会議の確立と終了、
- 会議名簿の管理、
- アプリケーション名簿の管理、
- アプリケーションの登録サービス、
- 会議議長権

これらの機能上のコンポーネントと関連したプリミティブのためのサービス定義は、第7章に示す。対応するプロトコル定義は、第8章に示す。

図6-1/JT-T124 は、MCS ドメインを介して分散する GCC コンポーネントの例を示す。GCC コンポーネントは図中白で示される。個々の端末または MCU は、ローカルなノード コントローラとアプリケーション プロトコル エンティティへの GCC サービスを提供する GCC プロバイダを含んでいる。

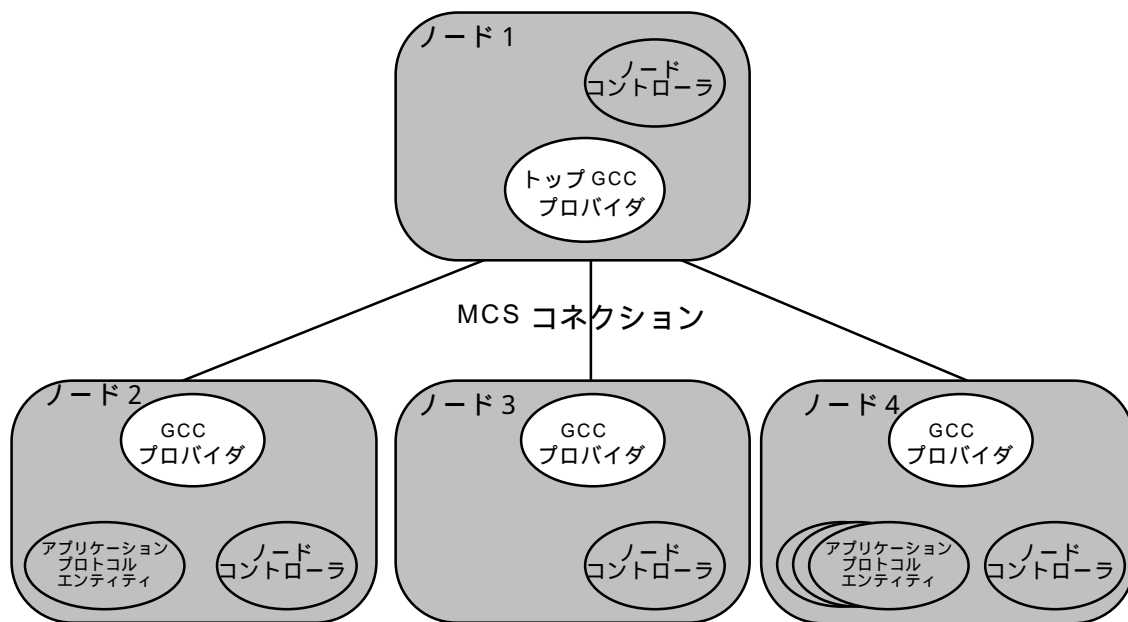


図 6-1/JT-T124 MCS ドメインを介して分散する GCC コンポーネントの例
(ITU-T T.124)

6.1.会議ノードのためのシステムモデル

GCC 会議に参加している個々のノードは、MCS 層、GCC 層、ノード コントローラから成り、さらに、一個以上のアプリケーション プロトコル エンティティも含んでいる。一つのノード内におけるこれらのコンポーネントの関係は、図 6-2/JT-T124 において説明される。ノードコントローラはノードにおけるコントロール エンティティであり、すべてのノードに適用される会議の側面を扱う。ノードコントローラは GCC と対話するが、MCS とは直接対話しない。アプリケーション プロトコル エンティティはまた GCC と対話し同様に MCS と直接対話する。アプリケーション プロトコル エンティティに GCC により提供されたサービスは、第一に、同位アプリケーション プロトコル エンティティが MCS を介して直接通信することを可能にするためである。個々のアプリケーション プロトコル エンティティまたはアプリケーション プロトコル エンティティとノード コントローラの間でのローカルな通信が起こるかもしれないが、しかし、この標準によりカバーされないローカルなインプリメンテーション問題である。ノード内では、一つ以上のアプリケーション プロトコル エンティティが同じアプリケーション プロトコルに基づく。この場合、それらは、同じアプリケーション プロトコル セッションの部分であり、他のノードの他の同位アプリケーション プロトコル エンティティにだけでなく、ノード内で通信することを可能にするか、またはそれらは別々なアプリケーション プロトコル セッションの部分であり、それらの同位の間で別々に通信することを可能にするが、同じプロトコルを利用する。

この標準の第 7 章において記述されるようなサービス プリミティブは、図 6-2/JT-T124 において示されるような GCC サービス アクセス ポイントとコントロール GCC サービス アクセス ポイントに適用される。この標準の第 8 章において記述された PDU は、コントロール MCS サービス アクセス ポイント (MCSAP) で使用可能な MCS サービス プリミティブで伝達される。

注-この標準が標準的に意図しているのは、会議制御のため、コントロール MCSAP を通して作動している基本オペレーションとデータ交換を行う外部通信シーケンスの手順と中身を、記述することである。図 6-2/JT-T124 において提

起されたノードの内部構造は、GCC プロトコルの機構に良い影響を与えるが、標準的でない。単にローカルのみに影響する GCC サービス プリミティブは、ここで記述された形で、すべてのノードにおいて存在する必要はない。もし外部の通信において同じ結果が、違う内部メカニズムを通して達成できるならば、一定の状況においてノード コントローラまたはアプリケーション プロトコル エンティティが何をするかについての記述は厳密に解釈する必要はない。

注-システムモデルは、いつでもサービス要求が発行できると仮定する。ノード コントローラまたはアプリケーション プロトコル エンティティは、ある要求を発行する時、前に発行した要求に対応する GCC プロバイダからの確認を受け取るのを待つ必要がない。しかし、特殊な実装はより厳密な要求が課される。

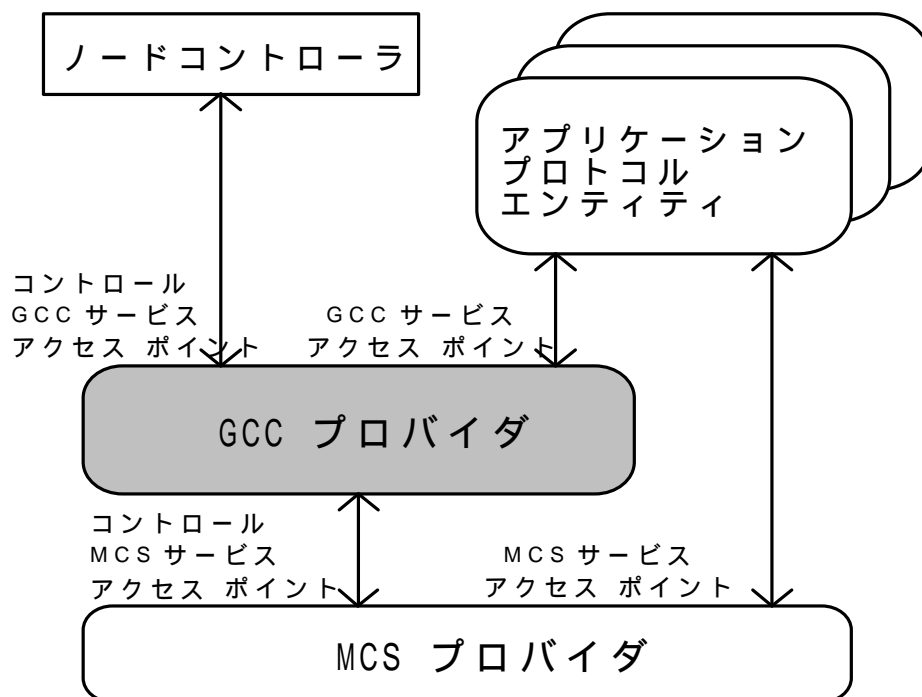


図 6.2/JT-T124 GCC サービス アクセス ポイントと MCS の関連を示すシステムモデル (ITU-T T.124)

6.2.会議の確立と終了

GCC は会議の確立と終了の 1 セットのサービスを提供する。会議は、オーディオグラフィックまたはオーディオビジュアルの情報を交換するために、任意の人数の参加者が参加できる会議室とみなされる。物理的な会議室同様、どんな会議が進行中であるかを見つけ出す、会議に加入する、会議を去る、会議への限定アクセスなどのサービスが、オーディオグラフィックとオーディオビジュアルの会議においても有意である。

ノードの参加者は、会議に加入することに先がけて、加入に必要な情報のすべてを知っていなくてもよい。GCC は、参加者に会議名のリストをみて参加したい会議を選ぶ方法を提供する。このサービスは、よくロビーにある記入された会議スケジュールと相似であり、誰かが、特有の会議が開催されている会議室を見つけることを可能にする。

GCC は、新しい会議を生成するための方法を提供する。これは会議参加者または会議管理者により行われてよい。新しい会議が生成されるとき、会議プロファイルとしてその会議の特徴が会議生成者により指定される。会議プロファイルは次のような情報を含んでいる。会議名、パスワードによりアクセスが限定されているかどうか、自由に参加でき

る（ロックされていない）か、または招待によってのみ参加できるように制限されている（ロックされている）かどうか。

既存の会議の拡張は、加入側ノードまたは会議召集者（または召集者代理ノード）によって初期化できる。もし会議が、パスワードで保護されているならば、加入側ノードは、正しいパスワードの供給を要求される。会議がロックされているならば、召集者または召集者代理ノードの要求があったときのみ加入が許可される。ノードは会議の間、どのような時にでも加入、追加してよい。一つ以上の会議に同時に参加することが可能である。GCC は、参加者を一つの会議から別のものに移動させる方法も提供する。この機能は、二つの会議をマージしたり、会議を一つ以上の会議に分割するために用いられるかもしれない。

ノードはいつでも会議からの切断を要求し、会議を続けることを他のノードに任せることができる。終了方法の選択に依存して、会議はすべてのノードの切断により自動終了するか、または、明示的な終了要求によって手動終了する。召集者または召集者代理ノードは、強制的にいつでも会議全体を終了してよく、また特有のノードを排除してよい。

6.3.会議名簿

一旦、ノードが会議に加入すると、そのノードは会議においてその存在を他のすべてのノードに通知する。会議において個々のノードがその存在を通知するために1つの GCC プリミティブが提供され、完全な会議名簿を個々のノードに提供するために、もう1つの GCC プリミティブが提供される。会議名簿は会議におけるすべてのノードのリストである。個々のノード用に、会議名簿はノードの名前、ノードの参加者のリストなどの情報と、ノード間の通信のために必要な他の情報を含んでいる。ノードは、それが会議名簿に含められるまで、会議の一部としては考慮されない。

6.4.アプリケーション名簿

GCC は、個々のノードでどのアプリケーション プロトコル エンティティが使用可能であるか識別する方法を提供し、さらに、お互いに通信するために、必要な情報を同位のアプリケーション プロトコル エンティティに提供する。会議に加入するとすぐに、個々のノードはアプリケーション プロトコル エンティティのローカルなリストをすべての他のノードに送る。そのローカルなアプリケーション名簿はその後いつでもアップデートされるかもしれない。この情報から、会議アプリケーション名簿がすべてのノードに形成されて、すべてのノードに同報される。この会議名簿の適当な部分は、ノードコントローラだけでなく個々のアプリケーション プロトコル エンティティにローカルに通信される。単純な名簿に加え、アプリケーションプロトコル特定の能力リストを交換される情報に含めるサービスも GCC がアプリケーション プロトコル エンティティに提供する。GCC は、会議においてアプリケーション能力の共通のセットを決定するために、規則の固定のセットをすべてのノードからのこの情報に適用する。この情報はまた、個々のアプリケーションプロトコルエンティティに通信される。

6.5.アプリケーションレジストリ

アプリケーションレジストリは、会議において利用するチャンネル、トークンおよび他の共用資源を管理するために使用されるかもしれないトップ GCC プロバイダにあるアクティブなデータベースである。アプリケーションレジストリは同位アプリケーション プロトコル エンティティ間の通信の確立を援助できる。

6.6.会議議長権

GCC は、ノードに対し会議の議長になることを可能にする方法を提供する。会議が議長制御かまたは非議長制御かを決定するために、トークンが GCC によって使用される。議長トークンをつかんだノードは、会議の議長になる。ノー

ドは議長権を要求しても現在の議長から議長権を受継してもよい。要求すると、GCC は現在の会議議長の識別子を提供する。会議の間、議長モードが許可されないことを、会議生成時に指定してもよい。

議長制御モードは、会議の進行順序を提供する手段として使用可能である。この順序が提供される実際の方法は、アプリケーションプロトコルにより決定される。特にアプリケーションプロトコルを規定している勧告は、会議が議長制御か非議長制御かどうかによって二者択一の手続を定義してもよい。GCC は、アプリケーションプロトコルにより使用されてもよい基本的議長制御オペレーションのメカニズムを提供する。つぎのようなメカニズムが提供されている。ノードは議長ノードから許可を要求してよく、許可が与えられたならば、そのノードのすべてのアプリケーション プロトコル エンティティはアプリケーションプロトコル仕様で規定されたように動作してよい。例えば、アプリケーションプロトコルは議長からの許可を持たないノードに許されるオペレーションにおいて、この許可を持っているノードに対する制限のうちいくつかまたはすべてを削除する間の厳密な制限について規定してもよい。また例えば、アプリケーションプロトコルはいちどすべての許可が議長から与えられると、許可されたオペレーションが実行される前に、もし一つが存在するならば、それ以上の許可は、議長ノードの同位アプリケーション プロトコル エンティティから与えられなければならないことを規定してよい。

6.7. その他の機能

時間制約のある会議を調整するために 1 つの方法が与えられている。時間制約のある会議でノードが残された時間を知るためのメカニズムが提供される。同様に残り時間をすべてのノードにアナウンスするメカニズムも提供される。(多くの場合時間切れをアナウンスする)。さらに、もし可能ならば時間を延長する要求を出すメカニズムが提供される。

また、不特定のオペレータからの支援を要求する方法が提供される。単純なテキストメッセージの伝達を可能にする機能も提供される。

6.8. GCC 抽象サービスのまとめ

表 6-1/JT-T124 は、すべての GCC プリミティブとそれらに関連する PDU のリストである。テーブルは、個々のプリミティブが端末または MCU に対して、必須 (M)、条件付きで必要 (C)、またはオプションである (O) ことを示している。マルチポート端末のための要求条件は、個々のプリミティブにおける端末と MCU の条件で最も制限の強いものである。条件付きで要求されるプリミティブとは、そのノードにおいて一つ以上のアプリケーション プロトコル エンティティのためのアプリケーションプロトコル仕様が、その使用を必須であると要求するものである。テーブルはまた対応する PDU について、これらが伝送 (T) および受信 (R) 方向の双方で必須 (M)、条件付きで必要 (C) であることを示している。マルチポート端末においての要求条件は、その PDU のための端末と MCU で最も制限の強いものである。条件付きで必要な PDU は、対応するプリミティブがそのノードにおいてサポートされる PDU である。プリミティブが必須ではないが、その対応する PDU が必須であるケースは、プリミティブと関連しないプロトコルの一部分が、サポートを要求されているその PDU に依存していることを暗に示している。

表 6-1/JT-T124 – GCC プリミティブと PDU

(ITU-T T.124)

機能単位	プリミティブ	端末	MCU	関連PDUs	方向	端末	MCU
会議の確立 と終了	GCC-CONFERENCE-CREATE 要求	M	O	ConferenceCreateRequest	T	M	C
	GCC-CONFERENCE-CREATE 指示	M	M	ConferenceCreateRequest	R	M	M
	GCC-CONFERENCE-CREATE 応答	M	M	ConferenceCreateResponse	T	M	M

GCC-CONFERENCE-CREATE 確認	M	O	ConferenceCreateResponse UserIDIndication	R T, R	M M	C M
GCC-CONFERENCE-QUERY 要求	M	M	ConferenceQueryRequest	T	M	M
GCC-CONFERENCE-QUERY 指示	M	M	ConferenceQueryRequest	R	M	M
GCC-CONFERENCE-QUERY 応答	M	M	ConferenceQueryResponse	T	M	M
GCC-CONFERENCE-QUERY 確認	M	M	ConferenceQueryResponse	R	M	M
GCC-CONFERENCE-JOIN 要求	M	O	ConferenceJoinRequest	T	M	C
GCC-CONFERENCE-JOIN 指示	O	M	ConferenceJoinRequest	R	C	M
GCC-CONFERENCE-JOIN 応答	O	M	ConferenceJoinResponse	T	C	M
GCC-CONFERENCE-JOIN 確認	M	O	ConferenceJoinResponse UserIDIndication	R T, R	M M	C M
GCC-CONFERENCE-INVITE 要求	O	M	ConferenceInviteRequest	T	C	M
GCC-CONFERENCE-INVITE 指示	M	M	ConferenceInviteRequest	R	M	M
GCC-CONFERENCE-INVITE 応答	M	M	ConferenceInviteResponse	T	M	M
GCC-CONFERENCE-INVITE 確認	O	M	ConferenceInviteResponse UserIDIndication	R T, R	C M	M M
GCC-CONFERENCE-ADD 要求	O	O	ConferenceAddRequest	T	C	C
GCC-CONFERENCE-ADD 指示	O	O	ConferenceAddRequest	R	C	C
GCC-CONFERENCE-ADD 応答	O	O	ConferenceAddResponse	T	C	C
GCC-CONFERENCE-ADD 確認	O	O	ConferenceAddResponse	R	C	C
GCC-CONFERENCE-LOCK 要求	O	O	ConferenceLockRequest	T	C	C
GCC-CONFERENCE-LOCK 指示	O	O	ConferenceLockRequest	R	C	C
GCC-CONFERENCE-LOCK 応答	O	O	ConferenceLockResponse	T	C	C
GCC-CONFERENCE-LOCK 確認	O	O	ConferenceLockResponse	R	C	C
GCC-CONFERENCE-UNLOCK 要求	O	O	ConferenceUnlockRequest	T	C	C
GCC-CONFERENCE-UNLOCK 指示	O	O	ConferenceUnlockRequest	R	C	C
GCC-CONFERENCE-UNLOCK 応答	O	O	ConferenceUnlockResponse	T	C	C
GCC-CONFERENCE-UNLOCK 確認	O	O	ConferenceUnlockResponse	R	C	C
GCC-CONFERENCE-LOCK-REPORT 指示	O	O	ConferenceLockIndication ConferenceUnlockIndication	T, R T, R	C C	C C
GCC-CONFERENCE-DISCONNECT 要求	M	M	-	-	-	-
GCC-CONFERENCE-DISCONNECT 指示	M	M	-	-	-	-
GCC-CONFERENCE-DISCONNECT 確認	M	M	-	-	-	-
GCC-CONFERENCE-TERMINATE 要求	O	O	ConferenceTerminateRequest ConferenceTerminateRequest	T R	C M	C M

	GCC-CONFERENCE-TERMINATE 指示	M	M	ConferenceTerminateIndication	T	M	M
	GCC-CONFERENCE-TERMINATE 確認	O	O	ConferenceTerminateResponse	R	M	M
	GCC-CONFERENCE-EJECT-USER 要求	O	O	ConferenceEjectUserRequest	T	C	C
	GCC-CONFERENCE-EJECT-USER 指示	M	M	ConferenceEjectUserIndication	R	M	M
	GCC-CONFERENCE-EJECT-USER 確認	O	O	ConferenceEjectUserResponse	T	M	M
	GCC-CONFERENCE-TRANSFER 要求	O	M	ConferenceTransferRequest	R	C	M
	GCC-CONFERENCE-TRANSFER 指示	M	M	ConferenceTransferIndication	T	M	M
	GCC-CONFERENCE-TRANSFER 確認	O	M	ConferenceTransferResponse	R	M	M
会議名簿	GCC-CONFERENCE-ANNOUNCE- PRESENCE 要求	M	M	RosterUpdateIndication	T, R	M	M
	GCC-CONFERENCE-ANNOUNCE- PRESENCE 確認	M	M	-	-	-	-
	GCC-CONFERENCE-ROSTER-REPORT 指示	M	M	RosterUpdateIndication	T, R	M	M
	GCC-CONFERENCE-ROSTER-INQUIRE 要求	O	O	-	-	-	-
	GCC-CONFERENCE-ROSTER-INQUIRE 確認	O	O	-	-	-	-
アプリケーション名簿	GCC-APPLICATION-PERMISSION-TO- ENROLL 指示	M	C	-	-	-	-
	GCC-APPLICATION-ENROLL 要求	M	C	RosterUpdateIndication	T, R	M	M
	GCC-APPLICATION-ENROLL 確認	M	C	-	-	-	-
	GCC-APPLICATION-ROSTER-REPORT 指示	M	C	RosterUpdateIndication	T, R	M	M
	GCC-APPLICATION-ROSTER-INQUIRE 要求	O	O	-	-	-	-
	GCC-APPLICATION-ROSTER-INQUIRE 確認	O	O	-	-	-	-
	GCC-APPLICATION-INVOKE 要求	O	O	ApplicationInvokeIndication	T	C	C
	GCC-APPLICATION-INVOKE 指示	O	O	ApplicationInvokeIndication	R	C	C
	GCC-APPLICATION-INVOKE 確認	O	O	-	-	-	-

アプリケーション レジストリ	GCC-REGISTRY-REGISTER-CHANNEL 要求	C	C	RegistryRegisterChannelRequest	T	C	C
				RegistryRegisterChannelRequest	R	M	M
	GCC-REGISTRY-REGISTER-CHANNEL 確認	C	C	RegistryResponse	T	M	M
				RegistryResponse	R	C	C
	GCC-REGISTRY-ASSIGN-TOKEN 要求	C	C	RegistryAssignTokenRequest	T	C	C
				RegistryAssignTokenRequest	R	M	M
	GCC-REGISTRY-ASSIGN-TOKEN 確認	C	C	RegistryResponse	T	M	M
				RegistryResponse	R	C	C
	GCC-REGISTRY-SET-PARAMETER 要求	C	C	RegistrySetParameterRequest	T	C	C
				RegistrySetParameterRequest	R	M	M
	GCC-REGISTRY-SET-PARAMETER 確認	C	C	RegistryResponse	T	M	M
				RegistryResponse	R	C	C
	GCC-REGISTRY-RETRIEVE-ENTRY 要求	C	C	RegistryRetrieveEntryRequest	T	C	C
				RegistryRetrieveEntryRequest	R	M	M
	GCC-REGISTRY-RETRIEVE-ENTRY 確認	C	C	RegistryResponse	T	M	M
				RegistryResponse	R	C	C
	GCC-REGISTRY-DELETE-ENTRY 要求	C	C	RegistryDeleteEntryRequest	T	C	C
				RegistryDeleteEntryRequest	R	M	M
	GCC-REGISTRY-DELETE-ENTRY 確認	C	C	RegistryResponse	T	M	M
				RegistryResponse	R	C	C
	GCC-REGISTRY-MONITOR 要求	C	C	RegistryMonitorEntryRequest	T	C	C
				RegistryMonitorEntryRequest	R	M	M
	GCC-REGISTRY-MONITOR 指示	C	C	RegistryMonitorEntryIndication	T	M	M
				RegistryMonitorEntryIndication	R	C	C
GCC-REGISTRY-MONITOR 確認	C	C	RegistryResponse	T	M	M	
			RegistryResponse	R	C	C	
GCC-REGISTRY-ALLOCATE-HANDLE 要求	C	C	RegistryAllocateHandleRequest	T	C	C	
			RegistryAllocateHandleRequest	R	M	M	
GCC-REGISTRY-ALLOCATE-HANDLE 確認	C	C	RegistryAllocateHandleResponse	T	M	M	
			RegistryAllocateHandleResponse	R	C	C	

会議 議長権	GCC-CONDUCTOR-ASSIGN 要求	O	O	-	-	-	-
	GCC-CONDUCTOR-ASSIGN 指示	C	C	ConductorAssignIndication ConductorAssignIndication	T R	M ¹ C	M ¹ C
	GCC-CONDUCTOR-ASSIGN 確認	O	O	-	-	-	-
	GCC-CONDUCTOR-RELEASE 要求	O	O	ConductorReleaseIndication ConductorReleaseIndication	T R	C M ¹	C M ¹
	GCC-CONDUCTOR-RELEASE 指示	C	C	ConductorReleaseIndication ConductorReleaseIndication	T R	M ¹ C	M ¹ C
	GCC-CONDUCTOR-RELEASE 確認	O	O	-	-	-	-
	GCC-CONDUCTOR-PLEASE 要求	O	O	-	-	-	-
	GCC-CONDUCTOR-PLEASE 指示	O	O	-	-	-	-
	GCC-CONDUCTOR-PLEASE 応答	O	O	-	-	-	-
	GCC-CONDUCTOR-PLEASE 確認	O	O	-	-	-	-
	GCC-CONDUCTOR-GIVE 要求	O	O	-	-	-	-
	GCC-CONDUCTOR-GIVE 指示	O	O	-	-	-	-
	GCC-CONDUCTOR-GIVE 応答	O	O	ConductorAssignIndication	T	C	C
	GCC-CONDUCTOR-GIVE 確認	O	O	-	-	-	-
	GCC-CONDUCTOR-INQUIRE 要求	C	C	-	-	-	-
	GCC-CONDUCTOR-INQUIRE 確認	C	C	-	-	-	-
	GCC-CONDUCTOR-PERMISSION-ASK 要求	O	O	ConductorPermissionAskIndication	T	C	C
	GCC-CONDUCTOR-PERMISSION-ASK 指示	O	O	ConductorPermissionAskIndication	R	C	C
	GCC-CONDUCTOR-PERMISSION-ASK 確認	O	O	-	-	-	-
	GCC-CONDUCTOR-PERMISSION-GRANT 要求	O	O	ConductorPermissionGrantIndication	T	C	C
	GCC-CONDUCTOR-PERMISSION-GRANT 指示	O	O	ConductorPermissionGrantIndication	R	C	C
	GCC-CONDUCTOR-PERMISSION-GRANT 確認	O	O	-	-	-	-
その他の機能	GCC-CONFERENCE-TIME-REMAINING 要求	O	O	ConferenceTimeRemainingIndication	T	C	C

¹ ConductorReleaseIndication PDUs の受信は、トップ GCC プロバイダが特権要求を正しく扱うために必須である。特権は会議が議長有りかなしかに依存する。 トップ GCC プロバイダによる ConductorAssignIndication PDUs と ConductorReleaseIndication PDUs の伝達は新しいノードが会議に参加するために必須である。

GCC-CONFERENCE-TIME-REMAINING 指示	O	O	ConferenceTimeRemainingIndication	R	C	C
GCC-CONFERENCE-TIME-REMAINING 確認	O	O	-	-	-	-
GCC-CONFERENCE-TIME-INQUIRE 要求	O	O	ConferenceTimeInquireIndication	T	C	C
GCC-CONFERENCE-TIME-INQUIRE 指示	O	O	ConferenceTimeInquireIndication	R	C	C
GCC-CONFERENCE-TIME-INQUIRE 確認	O	O	-	-	-	-
GCC-CONFERENCE-EXTEND 要求	O	O	ConferenceTimeExtendIndication	T	C	C
GCC-CONFERENCE-EXTEND 指示	O	O	ConferenceTimeExtendIndication	R	C	C
GCC-CONFERENCE-EXTEND 確認	O	O	-	-	-	-
GCC-CONFERENCE-ASSISTANCE 要求	O	O	ConferenceAssistanceIndication	T	C	C
GCC-CONFERENCE-ASSISTANCE 指示	O	O	ConferenceAssistanceIndication	R	C	C
GCC-CONFERENCE-ASSISTANCE 確認	O	O	-	-	-	-
GCC-TEXT-MESSAGE 要求	O	O	TextMessageIndication	T	C	C
GCC-TEXT-MESSAGE 指示	O	O	TextMessageIndication	R	C	C
GCC-TEXT-MESSAGE 確認	O	O	-	-	-	-
-	-	-	FunctionNotSupported	T, R	M	M

7.GCC サービスの定義

7.1.会議の確立と終了

本章では、会議の確立と終了に必要なプリミティブを記述する。本章にあるすべてのプリミティブは、端末もしくはMCUのノードコントローラにより、使用されることを目的としている。

7.1.1.会議プロファイル (Conference Profile)

会議が生成される時、そして、それぞれのノードが会議に参加し通信しあっている時において、全ての会議は、以下に定義される特徴を有している。これらの特徴は、会議中において不変である。この情報は、ひとまとめにして会議プロファイルとして参照される。

- 会議名 (Conference Name) – 会議を識別する数列とオプションであるユニコードの区 00 のテキスト。もし、会議の生成および会議加入時に、両方の形式の会議名 (Conference Name) が使用されるなら、いずれの形式でも加入すべき会議を指し示すことになる。
- 会議記述 (Conference Description) – 会議を記述するオプションの文字列。リスト記載会議において、この文字列は、GCC-CONFERENCE-QUERY-要求への応答の中の、会議記述子リストの中でも使われる。
- パスワード保護有りかパスワード保護無し (Password Protected vs. Not Password Protected) – 会議が、パスワードで保護されているか、否かの選択。
- リスト記載か非リスト記載 (Listed vs. Unlisted) – 有効な会議のリストを問い合わせた時に与えられる、会議リストに記載するか、しないかの選択。
- 議長制御可能か不可能 (Conductible vs. Non-conductible) – 会議が議長制御モードになり得るか、常に非議長制御モードかの選択。
- 終了方法 (Termination Method) – 会議を陽に終了 (手動終了) するか、全ての参加者が切断するまで会議を継続 (自動終了) するかを選択。
- 特権リスト (Privilege lists) – 通常、会議召集者にのみ利用可能である特権のうち、会議議長にも許される特権、議長制御会議においてどのノードにも許される特権、あるいは、非議長制御会議においてどのノードにも許される特権のリスト。

7.1.2.抽象サービスの記述

以下は、本節で定義するプリミティブのリストで、それぞれの機能の要約である。

- GCC-CONFERENCE-CREATE – 会議の特徴を特定して、新しい会議を生成するために、ノードコントローラにより使用される。
- GCC-CONFERENCE-QUERY – 現在進行中の会議が何であるかの問い合わせ、さらにその会議に加入するために必要な情報の問い合わせのため、ノードコントローラにより使用される。
- GCC-CONFERENCE-JOIN – 存在している会議に加入するために、ノードコントローラにより使用される。
- GCC-CONFERENCE-INVITE – 存在している会議に別のノードを招待するために、ノードコントローラにより使用される。
- GCC-CONFERENCE-ADD – 会議召集者もしくは召集者代理ノードにのみ使用許可される。MCUからのダイヤルアウトによるノードの追加を要求するために、ノードコントローラにより使用される。
- GCC-CONFERENCE-LOCK – 会議召集者もしくは召集者代理ノードにだけ許可される。新たな参加者が、陽に追加される場合を除き、会議に加入することを防ぐために、ノードコントローラにより使用される。

- GCC-CONFERENCE-UNLOCK – 会議召集者もしくは召集者代理ノードにだけ許可される。新たな参加者が、会議に加入することを許可するために、ノードコントローラにより使用される。
- GCC-CONFERENCE-LOCK-REPORT – 会議ロック状態から会議非ロック状態に変化したこと、あるいはその逆が起こったことをノードコントローラに通知する。
- GCC-CONFERENCE-DISCONNECT – 進行中の会議からローカルノードを切断するために、ノードコントローラが使用する。
- GCC-CONFERENCE-TERMINATE – 会議召集者もしくは召集者代理ノードにだけ許可される。すべてのノードを切断し、完全に会議を終了するために、ノードコントローラにより使用される。
- GCC-CONFERENCE-EJECT-USER – 会議召集者もしくは召集者代理ノード(もしくはコネクション階層構造の中で退場ノードの直上のノード)にだけ許可される。進行中の会議から特定のノードを排除するために、ノードコントローラにより使用される。
- GCC-CONFERENCE-TRANSFER – 会議召集者もしくは召集者代理ノードにだけ許可される。会議に加入しているノードを他の会議に移動させるために、ノードコントローラにより使用される。会議のマージや分離のための過程の中で使用されるかもしれない。

7.1.2.1. GCC-CONFERENCE-CREATE

GCC-CONFERENCE-CREATE-要求プリミティブは、ノードコントローラが使用する。そしてローカルノードが自動的に加入することになる新しい会議を、リモートノードで生成する。このプリミティブはいつ発行されてもよい。会議が生成される時、生成要求を受けたノード(GCC-CONFERENCE-CREATE-指示を受信したノード)は自動的に会議に加入し、その会議のトップGCCプロバイダとなる。このノードは、会議が存在し続ける間、トップGCCプロバイダであり続ける。インプリメントの形態によっては、GCCプリミティブを使用することなしでローカルに会議を生成することも可能となりうるであろう。この場合に、会議が生成されたノードはトップGCCプロバイダとなる。表7-1/JT-T124は、このプリミティブの型とパラメータを示している。図7-1/JT-T124は、このプリミティブを使用するときのイベントのシーケンスを示している。

表 7-1/JT-T124 – GCC-CONFERENCE-CREATE – プリミティブの型とパラメータ (ITU-T T.124)

パラメータ	要求	指示	応答	確認
会議名 (Conference Name)	M	M(=)	M(=)	M(=)
会議名修飾子 (Conference Name Modifier)	C		C	
会議 ID (Conference ID)		M	M(=)	M
会議召集者パスワード (Convener Password)	O	O(=)		
パスワード (Password)	O	O(=)		
ロックされた会議 (Conference Locked)	M	M(=)		
リスト記載会議 (Conference Listed)	M	M(=)		

議長制御可能会議 (Conference Conductible)	M	M(=)		
終了方法 (Termination Method)	M	M(=)		
議長特権リスト (Conductor Privilege List)	O	O(=)		
議長制御モード会議の特権リス ト (Conducted-mode Conference Privilege List)	O	O(=)		
非議長制御モード会議の特権リ スト (Non-conducted-mode Conference Privilege List)	O	O(=)		
会議記述 (Conference Description)	O	O(=)		
発信者識別子 (Caller Identifier)	O	O(=)		
発側アドレス (Calling Address)	O	O(=)		
着側アドレス (Called Address)	O	O		
ドメインパラメータ (Domain ParametersDomain Parameters)	M	M	M	M(=)
サービス品質 (Quality of Service)	M	M	M	M(=)
ローカルネットワークアドレス (Local Network Address)	O		O	
ユーザデータ (User Data)	O	O(=)	O	O(=)
結果 (Result)			M	M(=)

会議名 (Conference Name) : 生成される会議を識別する名前。これは、それぞれ 0 から 255 桁までの数列及びオプションのユニコードの区 00 のテキストから構成される。もし双方の形式の会議名 (Conference Name) が使用されており、さらにもしあるノードがこの会議への加入を望むなら、加入要求の中でいずれかの形式の名前を使用することになる。

会議名修飾子 (Conference Name Modifier) : もし要求もしくは応答しているノードが、要求された会議名と同じ会議名 (Conference Name) (数列か文字列かどちらか) の会議にすでに加入していれば、このパラメータが対応する要求もしくは応答プリミティブに含まれる。このパラメータがとる値は、この同一の会議名を持つノードにおいては、全ての会議にわたってユニークである。この修飾子が含まれる際には、対応しているノードとの直接接続により会議に加入しようとしている別のノードが要求する、GCC-CONFERENCE-JOIN-要求の着側ノード会議名修飾子 (Called Node Conference Name Modifier) パラメータとして使用される。この修飾子は、このノードに通知された GCC-CONFERENCE-QUERY への応答にも含まれる。このパラメ

ータは 255 桁までの数列である。

会議 ID (Conference ID) : 新たに生成された会議に、ローカルに割り当てられる識別子。会議に対する後の全ての参照は、ユニークな識別子として、会議 ID (Conference ID) を使用する。会議 ID (Conference ID) は、会議に関連する MCS ドメインを識別するためにローカルに使用される MCS ドメインセクタと同一である。

会議召集者パスワード (Convener Password) : このオプションパラメータには、数列及びオプションのユニコードの区 00 のテキストを含む。これは後の操作で召集者の ID 識別に使用されるが、それによって、召集者は会議を切断したり再接続したりすることが可能になる。このようにして、召集者はその特権を維持する。(ただし、これはトップ GCC プロバイダとの直接接続による会議の再加入に限られる。)。これは、召集者だけに特定の操作を行えるようにするためのプライベートパスワードである (最大 255 桁、255 文字)。このパラメータが NULL の場合は、召集者は会議の切断、再接続などの操作が不可能になり、召集者はその特権を維持することはできない。

パスワード (Password) : これは、会議に参加するためのパスワードとして使用する。数列及びオプションのユニコードの区 00 のテキストからなる (最大 255 桁の数列及び 255 文字)。もし、パスワードが指定されていないければ、会議はパスワード (Password) で保護されない。

注 - もし、会議がパスワードで保護されていれば、ノードコントローラは数列パスワードを指定しなければならない。またこの時、文字列パスワードも指定してよい。文字列エントリ機構を持たないノードを許容するためには、数列パスワードであることが求められる。文字列パスワードが使用される場合でも、ユーザが数列パスワードも生成しなければいけないとの前提はない。装置により生成された数列パスワードを使用する方が、より便利で安全かもしれない。

ロックされた会議 (Conference Locked) : このフラグがセットされれば、ただちに会議がロックされ、誰かが陽に GCC-CONFERENCE-ADD プリミティブを使って追加される以外は、何者もこの会議に加入することが防止される。本プリミティブの発行後いつでも会議のロックを外すため、GCC-CONFERENCE-LOCK プリミティブが使用されてもよい。会議のロックをするため、GCC-CONFERENCE-UNLOCK が使用されてもよい。GCC-CONFERENCE-ADD プリミティブと GCC-CONFERENCE-UNLOCK プリミティブはオプションであるために、このプリミティブをサポートしていない (端末ではなく) MCU がロックされた会議を生成したとしても、あまり役に立たないであろう。この様な MCU は、「ロックされた会議はサポートしていない」との理由で会議生成を拒否するであろう。もし、(GCC-CONFERENCE-INVITE-要求 / 確認と共に)GCC-CONFERENCE-ADD-指示 / 応答あるいは GCC-CONFERENCE-UNLOCK-通知 / 応答プリミティブがサポートされていれば、MCU はロックされた会議の生成を許容し、この拒否理由は使用しない。

リスト記載会議 (Conference Listed) : このフラグを TRUE に設定すると、会議問合せファシリティを使用する際に、この会議はリストに記載されてもよい。このフラグを FALSE に設定すると、この会議はリストに記載されない。

議長制御可能会議 (Conference Conductible) : このフラグを TRUE に設定すると、この会議は、GCC-CONDUCTOR-ASSIGN プリミティブを使用して議長制御モードになってもよい。このフラグを FALSE に設定すると、この会議は非議長制御のみが可能で、議長を割り当てようとしても拒否される。

終了方法 (Termination Method) : このパラメータは、会議召集者もしくは召集者代理ノードが GCC-CONFERENCE-TERMINATE プリミティブを使用して、陽に会議終了させるのか (手動終了)、加入ノードがなくなった時に会議終了させるのか (自動終了) を示す。

議長特権リスト (Conductor Privilege List) : これは、もしあるなら、会議召集者が会議議長に代理使用を許容する

機能のフラグリストである。このリスト中のフラグは、次の操作に対応する：GCC-CONFERENCE-TERMINATE、GCC-CONFERENCE-EJECT-USER、GCC-CONFERENCE-ADD、GCC-CONFERENCE-LOCK、GCC-CONFERENCE-UNLOCK および GCC-CONFERENCE-TRANSFER。

議長制御モード会議の特権リスト (Conducted-mode Conference Privilege List)：これは、議長制御モード会議において、会議召集者がどのノードに対しても代理使用を許容する機能リストフラグである。このリスト中のフラグは、前述と同じ操作に対応している。

非議長制御モード会議の特権リスト (Non-conducted-mode Conference Privilege List)：これは非議長制御モード会議において、会議召集者がどのノードに対しても代理使用を許容する、機能リストのフラグである。このリストのフラグは、前述と同じ操作に対応している。

会議記述 (Conference Description)：会議を記述するために使用されてよい長さ 255 文字までのオプションのユニコード文字列。これは、GCC-CONFERENCE-QUERY-要求への応答の一部として使用するため、会議の GCC プロバイダによって保持される。

発信者識別子 (Caller Identifier)：会議が生成されるノードに、発側ノードを識別させるため、使用されてよいオプションのユニコード文字列 (最大 255 文字)。このノードでの本情報の使用は、本標準の範囲を超えている。例えば、会議の生成を許された会議参加者の中から特定の会議参加者を選ぶことを、そのノードの利用者に許すために使用されるであろう。この文字列としていかなる値を設定することもできるので、会議のセキュリティの追加には必ずしもならない。

発側アドレス (Calling Address)：MCS コネクションを確立する際の、MCS-CONNECT-PROVIDER プリミティブに含まれるオプションのアドレス。このパラメータの解釈については、TTC 標準 JT-T122 を参照のこと。

着側アドレス (Called Address)：MCS コネクションを確立する際の、MCS-CONNECT-PROVIDER プリミティブに含まれるオプションのアドレス。このパラメータの解釈については、TTC 標準 JT-T122 を参照のこと。

ドメインパラメータ (Domain Parameters)：MCS コネクションを確立する際の、MCS-CONNECT-PROVIDER プリミティブに含まれるドメインパラメータ。このパラメータの解釈については、TTC 標準 JT-T122 を参照のこと。

サービス品質 (Quality of Service)：MCS コネクションを確立する際の、MCS-CONNECT-PROVIDER プリミティブに含まれるサービス品質のパラメータ。このパラメータの解釈については、TTC 標準 JT-T122 を参照のこと。

ローカルネットワークアドレス (Local Network Address)：もし要求が応答にこのアドレスが含まれるなら、対応ノードでのローカル GCC プロバイダは、別のノードからの GCC-CONFERENCE-QUERY-要求に応答する際に、会議記述子リスト (Conference Descriptor List) 中にあるネットワークアドレス (Network Address) パラメータとしてこの情報を使用する。

ユーザデータ (User Data)：認証や課金等のような、本標準の範囲外の機能のために使用してもよいオプションのユーザデータ。

結果 (Result)：要求が受け入れられたか拒否されたか、あるいは拒否されたときはその理由の通知。これには以下のおこりうる結果の 1 つが示される：「成功 (successful)」、「ユーザ拒否 (user rejected)」、「リソース不足 (resources not available)」、「対称性破綻による拒否 (rejected for symmetry-breaking)」、「ロックされた会議サポートせず (locked conference not supported)」、「会議名及び会議名修飾子が既に存在 (Conference Name and Conference Name Modifier already exist)」、「ドメインパラメータ受け不可能 (Domain ParametersDomain Parameters unacceptable)」、「非階層的ドメイン (domain not

hierarchical)」、「低位レイヤによる切断 (lower-layer initiated disconnect)」、「原因が不特定の接続失敗 (unspecified failure to connect)」。

GCC-CONFERENCE- CREATE-確認に対する否定的結果は、接続しようとしたノードとの物理的コネクションが切断されていることまでは意味しない。

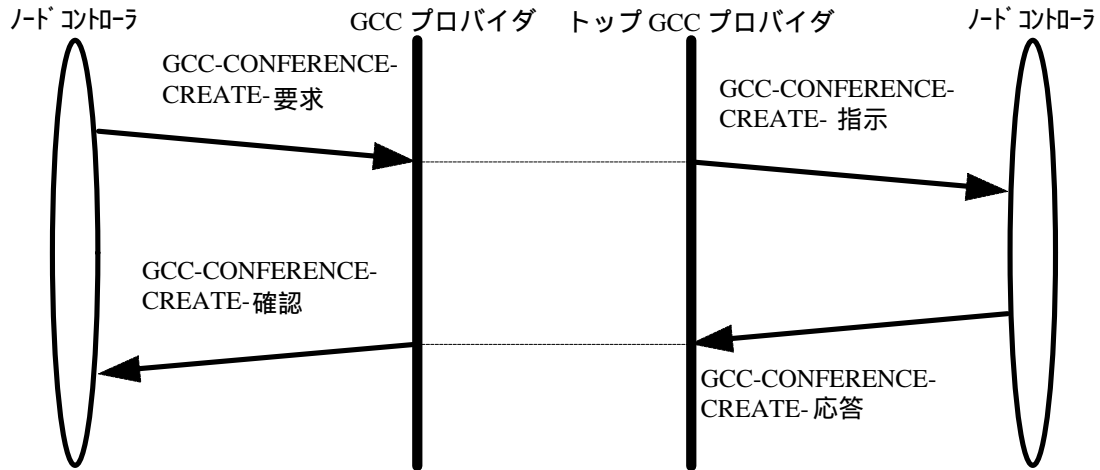


図 7-1/JT-T124 – GCC-CONFERENCE-CREATE – プリミティブのシーケンス

7.1.2.2. GCC-CONFERENCE-QUERY

GCC-CONFERENCE-QUERY-要求プリミティブは、ある MCU で現在存在している会議が何であるかを決定するために、ノードコントローラにより使用されてもよい。表 7-2/JT-T124 は、このプリミティブの型とパラメータを示している。図 7-2/JT-T124 は、このプリミティブが使用されるときに起きるイベントのシーケンスを示している。

表 7-2/JT-T124 – GCC-CONFERENCE-QUERY - プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	指示	応答	確認
ノードタイプ (Node Type)	M	M(=)	M	M(=)
非平衡指示子 (Asymmetry Indicator)	C	C(=)	C	C(=)
会議記述子リスト (Conference Descriptor List)				C
発側アドレス (Calling Address)	O	O(=)		
着側アドレス (Called Address)	O	O		
ユーザデータ (User Data)	O	O(=)	O	O(=)
結果 (Result)			M	M(=)

ノードタイプ (Node Type) : ノードタイプは、端末、MCU、マルチポート端末のいずれかの値である。

非平衡指示子 (Asymmetry Indicator) : これは、単一に接続されているノードのうち、どのノードが発側ノード (物

理的接続の開始者)であり、どのノードが着側ノードであるかの知識が要求される、特定の会議確立手順のため必要とされるフィールドである。このパラメータは、2ノード間を物理的に接続する場合に要求される(つまりTTC標準JT-T123を使用し、PSTN、ISDNもしくはCSDNで2ノード間の接続の場合である)。さもなければ、このパラメータはオプションである。もし、あるノードが発側ノードか着側ノードとしての状態を分かっているならば、この通知子を適切な値に設定する。ある場合には、自ノードが発側ノードか着側ノードか確かでないこともある。この場合には、このパラメータは32ビットの乱数を設定する。もし双方のノードとも状態が確かでないとき通知している時には、会議確立手順を実行するために、乱数を使用してどちらが発側ノードでどちらが着側ノードかを判断する(この判断は真の発側ノード、着側ノードがどちらであることを反映していないかも知れないことに留意のこと)。2つの乱数のうち、最大値を伝達した端末が、会議確立手順を実行するために、発側ノードとみなす。もし双方の乱数が同一なら、この判断が満足されていないとみなされ、要求者は異なる乱数を使用してGCC-CONFERENCE-QUERY-要求を再発行する(応答に対しても、異なる乱数が使用されることになる)。もし双方のノードが、このパラメータに対し実際の値(自ノードが発側ノードか着側ノードかの知識)で応答し、しかしながら、結果が不一致(すなわち両方とも発側ノードと思っているか、両方とも着側ノードと思っているか)なら、この判断もまた満足されていないとみなされ、要求者は乱数を使用してGCC-CONFERENCE-QUERY-要求を再発行することになる(応答にもまた乱数が使用される)。ある接続の片方において、問い合わせ要求に対し未確認なままの間、このノードから同じ物理接続上に発行される全ての問い合わせ要求と問い合わせ応答の中で、このパラメータの値は不変なままとする。乱数は、そのとりうる値の中で一様に分布した値をとるよう発生させるべきである。

会議記述子リスト(Conference Descriptor List)： 加入可能な開催中の会議を通知する可変長リスト。このリストには、会議生成時にリストに記載しないと指定した会議は含まれない。もし加入可能な会議がなければ、リスト中にエントリはないことになる。それぞれの会議記述子は、表7-3/JT-T124に示すパラメータを含む。

表 7-3/JT-T124 – 会議記述子(Conference Descriptor)の内容
(ITU-T T.124)

パラメータ	記述
会議名(Conference Name)	会議の会議名(Conference Name)。もし要求ノードがこの会議に参加することを望めば、このパラメータがGCC-CONFERENCE-JOIN-要求の会議名(Conference Name)パラメータに使用される値である。このパラメータは、最大255桁の数値及びオプションユニコードの区00文字列である。もし双方の名前の形式が与えられていたときは、いずれの形式でも、加入要求の中で指定できるであろう。
会議名修飾子 (Conference Name Modifier) (条件付き)	もし応答を返すノードで、会議名修飾子(Conference Name Modifier)を含む名前がわかっているならば、このパラメータが含まれる。もし要求ノードが、この会議に参加することを望めば、この会議名修飾子(Conference Name Modifier)をGCC-CONFERENCE-JOIN-要求の着側ノード会議名修飾子(Called Node Conference Name Modifier)パラメータとして使用する。このパラメータは、長さ255桁までの数値である。

会議記述 (Conference Description) (条件付き)	会議を記述するため使用される、長さ 255 文字までのオプションユニコード文字列。このパラメータは、会議記述子リスト(Conference Descriptor List)上に同一会議名(Conference Name)が1つ以上記述されており、これらを区別する際に特に有効であろう。
ロ ッ ク / 非 ロ ッ ク (Locked/Unlocked)	会議が現在ロックされているか、いないかを通知するフラグ。
平 文 パ ス ワ ー ド (Password In The Clear Required)	GCC-CONFERENCE-JOIN-要求において使用されると想定され、まだ1度も使用されたことがなく、暗号化されていないパスワードによって、会議が保護されていることを通知する。
ネットワークアドレス (Network Address) (条件付き)	要求ノードに提供されるアドレス情報。これはただオプションのネットワークアドレス (Network Address) パラメータが、そのノードの接続確立プリミティブ GCC-CONFERENCE-CREATE , GCC-CONFERENCE-JOIN もしくは GCC-CONFERENCE-INVITE に含まれているときにのみ提供される。

発側アドレス (Calling Address) : MCS 接続を確立する MCS-CONNECT-PROVIDER プリミティブに含まれるオプションのアドレス。

着側アドレス (Called Address) : MCS 接続を確立する MCS-CONNECT-PROVIDER プリミティブに含まれるオプションのアドレス。

ユーザデータ (User Data) : 認証や課金等のような、本標準の範囲外の機能のために使用されてもよいオプションのユーザデータ。

結果 (Result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む : 「成功 (successful)」、「ユーザ拒否 (user-rejected)」、「ドメインパラメータ受け付け不可能 (Domain Parameters Domain Parameters unacceptable)」、「非階層的ドメイン (domain not hierarchical)」、「低位レイヤによる切断 (lower-layer initiated disconnect)」、「原因が不特定の接続失敗 (unspecified failure to connect)」。

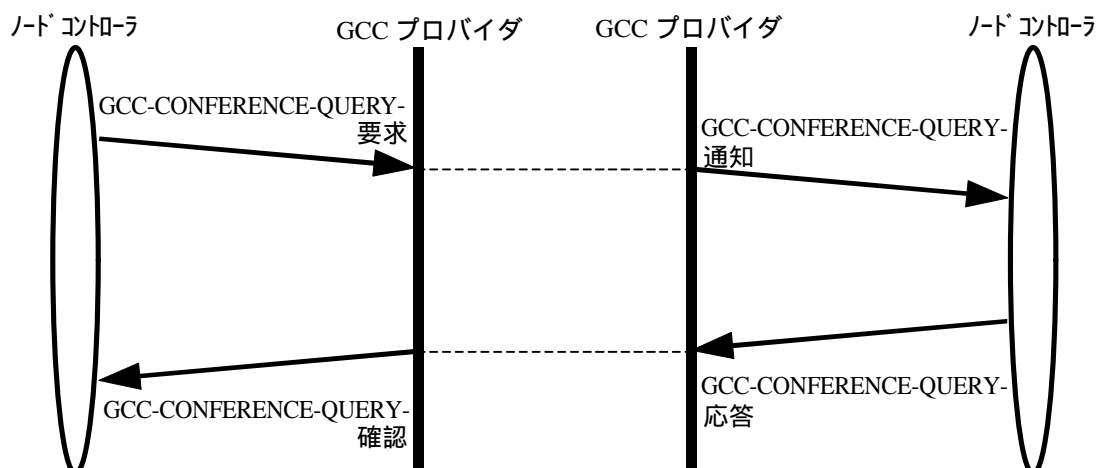


図7-2/JT-T124 – GCC-CONFERENCE-QUERY –プリミティブのシーケンス
(ITU-T T.124)

7.1.2.3. GCC-CONFERENCE-JOIN

ノードコントローラは、GCC-CONFERENCE-JOIN-要求プリミティブを使用して、ローカルノードを現存する会議に加入させる。このプリミティブは、いつ発行されてもよい。1つのノードは、同時に複数の会議に加入できる。会議がパスワードで保護されている場合、パスワードパラメータが正しい情報を含んでいなければ、会議へ加入することはできない。表7-4/JT-T124は、このプリミティブの型とパラメータを示す。図7-3/JT-T124は、このプリミティブを使用する時のイベントのシーケンスを示す。

注: パスワードが正しいかどうか決定するのは、ノードコントローラの責任であり、トップ GCC プロバイダでは決定しない。正しいことの定義は、完全な1文字1文字の一致ほど厳格でなくてもよい。例えば、パスワードがテキスト形式の時には、ノードコントローラは大文字小文字を区別しない一致を使用してもよい。

表7-4/JT-T124 GCC-CONFERENCE-JOIN プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	要求	指示	応答	確認
会議名(Conference Name)	M			M
着側ノード会議名修飾子 (Called Node Conference Name Modifier)	C			C(=RQ)
発側ノード会議名修飾子 (Calling Node Conference Name Modifier)	C			C(=RQ)
会議ID(Conference ID)		M	M(=)	M
召集者パスワード(Convener Password)	O	O(=)		
パスワード>Password)	C	C(=)	C	C(=)
発信者識別子 (Caller Identifier)	O	O(=)		
発アドレス(Calling Address)	O	O(=)		
着アドレス(Called Address)	O	O		
ドメインパラメータ (Domain Parameters)	M			M
サービス品質 (Quality of Service)	M			M
平文パスワード (Password In The Clear Required)				M
ロックされた会議(Conference				M

Locked)				
リスト記載会議 (Conference Listed)				M
議長制御可能会議 (Conference Conductible)				M
終了方法 (Termination Method)				M
議長特権リスト (Conductor Privilege List)				C
議長制御モード会議特権リスト (Conducted-mode Conference Privilege List)				C
非議長制御モード会議特権リスト (Non-conducted-mode Conference Privilege List)				C
会議記述 (Conference Description)				C
ローカルネットワークアドレス (Local Network Address)	O			
ユーザデータ (User Data)	O	O(=)	O	O(=)
結果 (Result)			M	M(=)

会議名 (Conference Name):

加入する会議の名称。要求時には、このパラメータは数字列またはユニコードの区 00 のテキストの文字列で、最大 255 文字である。会議生成時に会議名の数字部分とテキスト部分の両方が使用された場合、どちらかの部分がこの要求に含まれる。会議生成時に数字形式とテキスト形式両方が使用された場合、確認時には、このパラメータは数字形式とテキスト形式を含めた完全な会議名を含む。

着側ノード会議名修飾子 (Called Node Conference Name Modifier):

このパラメータは、加入側ノード(接続確立のために接続が試みられているノード)に直接接続したノードが、会議名の一部に会議名修飾子を含む場合に、要求プリミティブに含まれなければならない。このパラメータは、加入側ノードに直接接続したノードに知られている会議名修飾子を示さなくてはならない。このパラメータは、255 桁までの数字列である。

発側ノード会議名修飾子 (Calling Node Conference Name Modifier):

加入要求を発行しているノードにおいて、加入する会議の会議名と同一の名前で会議が既に存在しているならば、このパラメータが GCC-CONFERENCE-JOIN 要求に含まれ、会議名修飾子を示さなくてはならない。ローカルノードではこの会議名修飾子により会議が知られる。このパラメータはローカルノードが現在加入している会議と同じ会議名を持つ他の会議に既に用いられている会議名修飾子とは異ならなくてはならない。このパラメータが使用された場合、これは、この会議のローカルノードへの接続を通してこの会議への参加を試みる他方のノードにとっては、着側ノード会議名修飾子になる。さらに、会議がリストに記載される場合、このノードからの GCC-CONFERENCE-QUERY-応答に、

会議名修飾子パラメータとしてこの修飾子が含まれる。このパラメータは、255 桁までの数字列である。

会議 ID(Conference ID):

このパラメータは、トップ GCC プロバイダの指示/応答プリミティブにおいて使用され、要求側ノードが加入要求する会議の会議 ID である。このパラメータは、確認プリミティブで GCC が返すもので、ローカルに割り当てられた ID を示す。次以降、この ID が会議への参照をあらわす。会議 ID は、会議に関連した MCS ドメインを識別するためにローカルに使用される MCS ドメインセクタと同一になるべきである。

召集者パスワード(Convener Password):

これはオプションのパラメータで、数字列またはユニコードの区 00 のテキスト文字列である(255 桁の数字または文字)。会議召集者が、切断後に会議に再び加わる時に使用する。この識別子が会議生成時に使用された対応する識別子に一致し、MCU 経由ではなくトップ GCC プロバイダへの直接接続で加入した場合に限り、加入側ノードは召集者の特権を与えられる。正しい召集者パスワードを用いた召集者は、ロックされた会議にさえ加入できる。会議がパスワード保護されている場合は、召集者パスワードに加えて正しいパスワードが必要である。召集者パスワードが存在するが一致しないならば、加入要求は拒否されなくてはならない。召集者パスワードが指定された値と一致するかどうかの基準は、ノードコントローラが決定する。

パスワード>Password):

パスワードパラメータは、パスワード保護された会議にアクセスするために用いられる。「チャレンジ応答要求」という結果が以前のこの会議のための GCC-CONFERENCE-JOIN-確認で受信された場合、または平文パスワードパラメータが以前のこの会議のための GCC-CONFERENCE-QUERY-確認の会議記述子にセットされた場合、このプリミティブの要求形式において、このパラメータはパスワードのみを含むべきである。平文パスワードの場合には、このパラメータは数字列またはユニコードの区 00 のテキストの文字列(255 桁の数字または文字)である。テキストパスワードが数字のパスワードに加えて会議生成時に定義されたならば、テキスト文字列のみが使用されてよい。このパラメータが暗号化されたパスワードを含んでいる場合、パスワードは以前に受け取ったチャレンジにおいて指定されたアルゴリズムの 1 つを用いてエンコードされている。パスワードが平文でも暗号化されていても、チャレンジの応答として送られた場合、このパラメータはチャレンジで受け取ったタグと全く同一のタグも含まなくてはならない。パスワードが GCC-CONFERENCE-QUERY-指示の平文パスワードフラグの応答の場合、タグは必要ない。このプリミティブの要求形式では、このパラメータは受け取るノードへのチャレンジを含んでもよい。チャレンジがこのパラメータに含まれる場合、制約はない。

このプリミティブの応答形式では、このパラメータは、この会議への加入にはパスワードが必要であることを要求者に示すチャレンジを含んでいてもよい。この場合のために、このパラメータは、受信可能なパスワードの形式(平文か非標準暗号化アルゴリズムのリスト)、このチャレンジを識別するために使用される整数タグ、そして暗号化のために要求される他の追加情報などの情報を含む。このパラメータがチャレンジを含む場合、このプリミティブの結果パラメータは「チャレンジ応答要求」をセットされなければならない。この場合、この交換によっては接続は確立しない。このプリミティブの応答形式では、要求ノードによるチャレンジへの応答として、このパラメータがパスワードを含んでもよい。このパスワードは平文でも暗号化されていてもよい。

指示の中のこのパラメータが、受け取るノードに十分なフォーマットになっていない場合、そのノードは結果として「無効なチャレンジ応答」という応答を発行しなければならない。指示の中のこのパラメータが、フォーマットは正しいがパスワード正しくないならば、応答は結果として「無効なパスワード」を含まなければならない。

発信者識別子(Caller Identifier):

あるオプションのユニコードテキスト文字列(最大 255 キャラクタ)。トップ GCC プロバイダのノードで、発側ノード

を識別するために使用される。このノードでこの情報をどう使うかは、この標準の範囲を越えている。例えば、会議への参加者を制限するために使用される。しかしながら、このストリングはどんな値にもセットすることができるので、これを行う会議の安全が増すとは限らない。

発アドレス(Calling Address):

MCS 接続の確立で、MCS-CONNECT-PROVIDER プリミティブに含まれるオプションのアドレス。このパラメータについては TTC 標準 JT-T122 を参照のこと。

着アドレス(Called Address):

MCS 接続の確立時に MCS-CONNECT-PROVIDER プリミティブに含まれるオプションのアドレス。このパラメータについては TTC 標準 JT-T122 を参照のこと。

ドメインパラメータ(Domain Parameters):

MCS 接続の確立時に MCS-CONNECT-PROVIDER プリミティブに含まれるドメインパラメータ。このパラメータについては TTC 標準 JT-T122 を参照のこと。

サービス品質(Quality of Service):

MCS 接続の確立時に MCS-CONNECT-PROVIDER プリミティブに含まれるサービス品質のパラメータ。このパラメータについては TTC 標準 JT-T122 を参照のこと。

平文パスワード>Password In The Clear Required):

これは Boolean パラメータである。これは会議がパスワードで保護されていることを示す。一度もパスワードのためのチャレンジをされていない時に、このパスワードは暗号化せずに GCC-CONFERENCE-JOIN 要求において使用されるものである。加入側ノードが MCU であるならば、この情報は GCC-CONFERENCE-QUERY-指示に対する応答を生成する時に GCC プロバイダが使用する。ノードコントローラはこの情報を使用して、パスワードパラメータが GCC-CONFERENCE-TRANSFER-要求に含まれてもよいかどうか決定しなければならない。

ロックされた会議(Conference Locked):

このフラグは加入した会議がロックされているかどうかを示す。加入側ノードが MCU であるならば、GCC-CONFERENCE-QUERY-指示に対する応答の生成時に、この情報が使用されなければならない。

リスト記載会議(Conference Listed):

このフラグは加入した会議がリストに記載されるかどうかを示す。加入側ノードが MCU であるならば、GCC-CONFERENCE-QUERY-指示に対する応答生成時に、この情報が使用されなくてはならない。特に、リストに記載されないことを指示された会議は、GCC-CONFERENCE-QUERY-応答で供給される会議リストに記載してはならない。

議長制御可能会議(Conference Conductible):

このフラグの TRUE 設定は、GCC-CONDUCTOR-ASSIGN プリミティブを使用して、この会議が議長モードに置かれることを示す。このフラグの FALSE 設定は、この会議は非議長モードのみでなくてはならず、議長を割り当てる試みは拒否されなければならないことを示す。

終了方法(Termination Method):

このフラグは、加入した会議の終了ルールを示す。会議は、手動で、または自動的に終了する。

議長特権リスト(Conductor Privilege List):

これは、召集者が指定するもので、議長による使用が容認された機能を示すフラグのリストである。このリスト内のフラグは、以下の操作に相当する:GCC-CONFERENCE-TERMINATE、GCC-CONFERENCE-EJECT-USER、GCC-CONFERENCE-ADD、GCC-CONFERENCE-LOCK、GCC-CONFERENCE-UNLOCK、GCC-CONFERENCE-TRANSFER。

議長制御モード会議特権リスト(Conducted-mode Conference Privilege List):

これは、議長制御モード会議においてどのノードでも使用できるものとして、召集者が指定した機能を示すフラグのリストである。このリスト内のフラグは、上述のパラメータと同じ操作に相当する。

非議長制御モード会議特権リスト(Non-conducted-mode Conference Privilege List):

これは、非議長制御モード会議においてどのノードでも使用できるものとして、召集者が指定した機能を示すフラグのリストである。このリスト内のフラグは、上述のパラメータと同じ操作に相当する。

会議記述(Conference Description):

このパラメータは、加入している会議について述べた、255文字までのユニコードテキスト文字列である。会議生成時に含まれた場合に確認プリミティブにおいてだけ存在する。

ローカルネットワークアドレス(Local Network Address):

これが要求に含まれた場合、対応するノードのローカル GCC プロバイダは、会議記述子リストのネットワークアドレスパラメータにこの情報を使用しなければならない。この会議子記述リストは、他のノードから GCC-CONFERENCE-QUERY 要求が発行されたときの応答の一部として送るものである。

ユーザデータ(User Data):

課金、認証のようなこの標準の範囲外の機能のために使用してもよいオプションのユーザデータ。

結果(Result):

要求が受け入れられたか拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の1つを含む: 「成功(successful)」、「ユーザ拒否(user rejected)」、「無効な会議(Invalid conference)」、「無効なパスワード(Invalid Password)」、「チャレンジ応答要求(Challenge Response required)」、「無効なチャレンジ応答(Invalid Challenge Response)」、「無効な召集者パスワード(Invalid Convener Password)」、「受け入れ不能なドメインパラメータ(Domain Parameters Domain Parameters unacceptable)」、「非階層ドメイン(domain not hierarchical)」、「低レイヤ初期切断(lower-layer initiated disconnect)」、「不特定の接続失敗(unspecified failure to connect)」。GCC-CONFERENCE-JOIN 確認に含まれる否定的結果は、接続が試みられていたノードへの物理的接続が切断されたことを意味するわけではない。

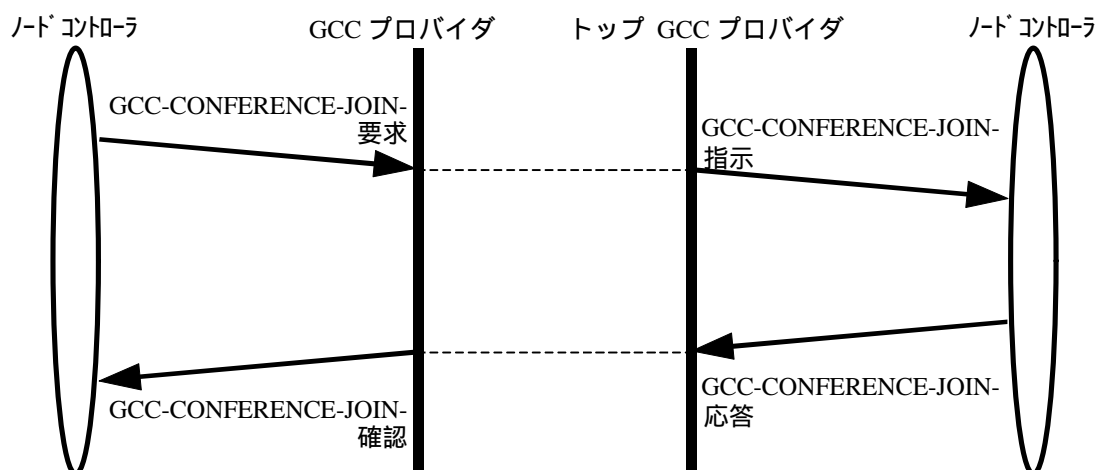


図 7-3/JT-T124 GCC-CONFERENCE-JOIN プリミティブのシーケンス
(ITU-T T.124)

7.1.2.4. GCC-CONFERENCE-INVITE

ノードコントローラは、GCC-CONFERENCE-INVITE-要求プリミティブを用いて、あるノードに会議への招待をおこなう。このプリミティブは、GCC-CONFERENCE-ADD-指示の結果として使用されるか、招待側ノードによって直接発行される。追加実行 MCU は、被追加ノードに GCC-CONFERENCE-INVITE-要求プリミティブを発行することによって、会議に招待する。会議がパスワードで保護されていても、招待されたノードはパスワードなしに会議に加入できる。表 7-5/JT-T124 は、このプリミティブの型とパラメータを示す。図 7-4/JT-T124 は、このプリミティブを使用する時のイベントのシーケンスを示す。

表 7-5/JT-T124 GCC-CONFERENCE-INVITE プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	指示	応答	確認
会議 ID(Conference ID)	M	M	M(=IN)	M(=RQ)
会議名(Conference Name)		M		
会議名修飾子 (Conference Name Modifier)			C	
発信者識別子 (Caller Identifier)	O	O(=)		
発アドレス(Calling Address)	O	O(=)		
着アドレス(Called Address)	O	O		
ドメインパラメータ (Domain Parameters)		M	M	
サービス品質 (Quality of Service)		M	M	
平文パスワード (Password In The Clear Required)		M		
ロックされた会議 (Conference Locked)		M		
リスト記載会議 (Conference Listed)		M		
議長制御可能会議 (Conference Conductible)		M		
終了方法 (Termination Method)		M		
議長特権リスト (Conductor Privilege List)		C		
議長制御モード会議特権リスト (Conducted-mode Conference Privilege List)		C		

非議長制御モード会議 特権リスト (Non-conducted- mode Conference Privilege List)		C		
会議記述 (Conference Description)		C		
ローカルネットワークアドレス (Local Network Address)			O	
ユーザデータ (User Data)	O	O(=)	O	O(=)
結果 (Result)			M	M(=)

会議 ID (Conference ID):

プリミティブが参照する会議の識別子。

会議名 (Conference Name):

会議プロファイルにおいて指定される会議の名前。会議名が数字形式およびテキスト形式を含むならば、両方がこのパラメータに含まれなければならない。

会議名修飾子 (Conference Name Modifier):

被招待側ノードで既に同一の会議名の会議が存在する場合、このパラメータが GCC-CONFERENCE-INVITE-応答に含まれなければならない。このパラメータは、会議名修飾子を示さなければならない。ローカルノードではこの会議名修飾子によりこの会議が知られている。このパラメータは、ローカルノードが現在加入している会議と同じ会議名を持つ他の会議に既に用いられている会議名修飾子とは異ならなくてはならない。このパラメータが使用された場合、これは発側ノード会議名修飾子になる。ローカルノードへの接続を通してこの会議への加入を試みる他のノードは、この発側ノード会議名修飾子により、この会議を参照する。会議がリストに記載される場合、このノードからの GCC-CONFERENCE-QUERY-応答に、会議名修飾子パラメータとしてこの修飾子が含まれる。このパラメータは、255 桁までの数字列である。

発信者識別子 (Caller Identifier):

あるオプションのユニコードテキスト文字列 (最大 255 文字)。会議が作成されたノードで発側ノードを識別するために使用される。このノードでこの情報をどう使うかは、この標準の範囲を越えている。例えば、会議への参加者を制限するために使用される。しかしながら、このストリングはどんな値にもセットすることができるので、これを行う会議の安全が増すとは限らない。

発アドレス (Calling Address):

MCS 接続の確立で、MCS-CONNECT-PROVIDER プリミティブに含まれるオプションのアドレス。このパラメータについては TTC 標準 JT-T122 を参照のこと。

着アドレス (Called Address):

MCS 接続の確立で、MCS-CONNECT-PROVIDER プリミティブに含まれるオプションのアドレス。このパラメータについては TTC 標準 JT-T122 を参照のこと。

ドメインパラメータ (Domain Parameters):

MCS 接続の確立時に MCS-CONNECT-PROVIDER プリミティブに含まれるドメインパラメータ。このパラメータについては TTC 標準 JT-T122 を参照のこと。

サービス品質 (Quality of Service):

MCS 接続の確立時に MCS-CONNECT-PROVIDER プリミティブに含まれるサービス品質のパラメータ。このパラメータについては TTC 標準 JT-T122 を参照のこと。

平文パスワード(Password In The Clear Required):

これは Boolean パラメータである。これは会議がパスワードで保護されていることを示す。一度もパスワードのためのチャレンジをされていない時に、このパスワードは暗号化せずに GCC-CONFERENCE-JOIN 要求において使用されるものである。加入側ノードが MCU であるならば、この情報は GCC-CONFERENCE-QUERY-指示に対する応答を生成する時に GCC プロバイダが使用する。ノードコントローラはこの情報を使用して、パスワードパラメータが GCC-CONFERENCE-TRANSFER-要求に含まれてもよいかどうか決定しなければならない。

ロックされた会議(Conference Locked):

このフラグは加入した会議がロックされているかどうかを示す。加入側ノードが MCU であるならば、GCC-CONFERENCE-QUERY-指示に対する応答の生成時に、この情報が使用される。

リスト記載会議(Conference Listed):

このフラグは加入した会議がリストに記載されるかどうかを示す。加入側ノードが MCU であるならば、GCC-CONFERENCE-QUERY-指示に対する応答生成時に、この情報が使用されなくてはならない。特に、リストに記載されないことを指示された会議は、GCC-CONFERENCE-QUERY-応答で供給される会議リストに記載してはならない。

議長制御可能会議(Conference Conductible):

このフラグの TRUE 設定は、GCC-CONDUCTOR-ASSIGN プリミティブを使用して、この会議が議長モードに置かれることを示す。このフラグの FALSE 設定は、この会議は非議長モードのみでなくてはならず、議長を割り当てる試みは拒否されなければならないことを示す。

終了方法(Termination Method):

このフラグは、加入した会議の終了ルールを示す。会議は、手動で、または自動的に終了する。

議長特権リスト(Conductor Privilege List):

これは、召集者が指定するもので、議長による使用が容認された機能を示すフラグのリストである。このリスト内のフラグは、以下の操作に相当する:GCC-CONFERENCE-TERMINATE、GCC-CONFERENCE-EJECT-USER、GCC-CONFERENCE-ADD、GCC-CONFERENCE-LOCK、GCC-CONFERENCE-UNLOCK、GCC-CONFERENCE-TRANSFER。

議長制御モード会議特権リスト(Conducted-mode Conference Privilege List):

これは、議長制御モード会議においてどのノードでも使用できるものとして、召集者が指定した機能を示すフラグのリストである。このリスト内のフラグは、上述のパラメータと同じ操作に相当する。

非議長制御モード会議特権リスト(Non-conducted-mode Conference Privilege List):

これは、非議長モード会議においてどのノードでも使用できるものとして、召集者が指定した機能を示すフラグのリストである。このリスト内のフラグは、上述のパラメータと同じ操作に相当する。

会議記述(Conference Description):

このパラメータは、会議について述べた、255 文字までのユニコードテキスト文字列である。会議生成時に含まれた場合に確認プリミティブにおいてだけ存在する。

ローカルネットワークアドレス(Local Network Address):

応答に含まれた場合、対応するノードのローカル GCC プロバイダは、会議記述子リストのネットワークアドレスパラメータにこの情報を使用しなければならない。この会議記述子リストは、他のノードから GCC-CONFERENCE-QUERY-要求が発行されたときの応答の一部として送るものである。

ユーザデータ(User Data):

課金、認証のようなこの標準の範囲外の機能のために使用してもよいオプションのユーザデータ。

結果(Result):

要求が受け入れられたか拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の1つを含む: 「成功(successful)」、「ユーザ拒否(user rejected)」、「無効な会議(Invalid conference)」、「受け入れ不能なドメインパラメータ(Domain Parameters Domain Parameters unacceptable)」、「非階層ドメイン(domain not hierarchical)」、「低位レイヤによる切断(lower-layer initiated disconnect)」、「不特定の接続失敗(unspecified failure to connect)」。GCC-CONFERENCE-INVITE-確認に含まれる否定的結果は、接続が試みられていたノードへの物理的接続が切断されたことを意味するわけではない。

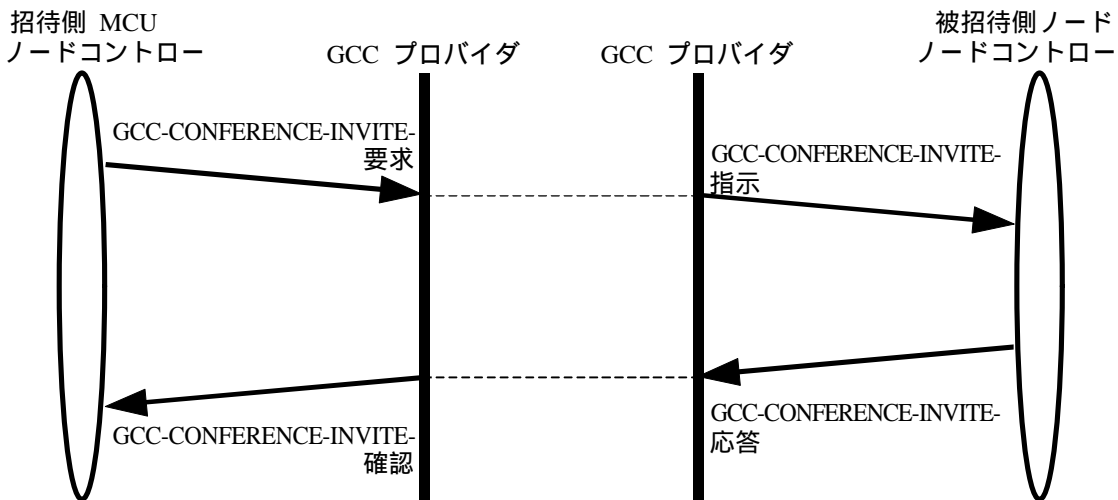


図 7-4/JT-T124 GCC-CONFERENCE-INVITE プリミティブのシーケンス (ITU-T T.124)

7.1.2.5. GCC-CONFERENCE-ADD

ノードコントローラは GCC-CONFERENCE-ADD-要求プリミティブを用いて、MCU に対して指定のノードにダイヤルするように要求し、1つのノードを会議に追加する。このプリミティブは召集者が召集者指名ノードによって発行される場合のみ有効である。追加実行 MCU が指定された場合、プリミティブのシーケンスは要求者と追加実行 MCU の間での直接のやりとりになる。追加実行 MCU が指示されない場合、指示はトップ GCC プロバイダから発行される。この場合でも、追加を行なう能力があるポートがこのノードで利用できるならば、交換は直接おこなわれる。トップ GCC プロバイダのノードコントローラが、会議内の全 MCU のためのポート情報中央データベースにアクセスできるならば、追加実行 MCU を指定し、もう一つの GCC-CONFERENCE-ADD-要求を発行することによって、間接的に追加を行なう。トップ MCU がこの操作に対する召集者指名の特権ノードではない場合もこれは行われる。

被追加ノードへの物理的コネクションをする準備ができると、追加実行 MCU は、被追加ノードを会議に招待するために、GCC-CONFERENCE-INVITE-要求プリミティブを発行する。追加実行 MCU が被追加ノードへの物理的コネクションを既に持っているならば、新しい物理的コネクションを確立せずに、追加実行 MCU は被追加ノードに直接 GCC-CONFERENCE-INVITE-要求を発行する。このコネクションが被追加ノードのネットワークアドレス(Network Address)の知識なしに確立された場合(例えば、被追加ノードが別の会議に接続するために追加実行 MCU への物理的コネクションを持っていた場合)、会議名簿のネットワークアドレス(Network Address)パラメータを使用し、既に接続

されたノードのネットワークアドレス(Network Address)と被追加ノードのネットワークアドレス(Network Address)を比較してこれらのノードが同一であることを確認する。表 7-6/JT-T124 は、このプリミティブの型とパラメータを示す。図 7-5/JT-T124 と図 7-6/JT-T124 は、それぞれ、直接の場合と間接の場合でこのプリミティブを使用する時のイベントのシーケンスを示す。

表 7-6/JT-T124 GCC-CONFERENCE-ADD プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	指示	応答	確認
会議 ID(Conference ID)	M	M	M(=IN)	M(=RQ)
ネットワークアドレス(Network Address)	M	M(=)	M(=)	M(=)
追加実行 MCU ノード ID(Adding MCU Node ID)	O			
要求ノード ID(Requesting Node ID)		M	M(=)	
ユーザデータ(User Data)	O	O(=)	O	O(=)
結果(Result)			M	M(=)

会議 ID(Conference ID):プリミティブが参照する会議の識別子。

ネットワークアドレス(Network Address):このパラメータは、被追加ノードのアドレスを指定するもので、被追加ノードへのネットワークコネクションを作る時に、追加実行 MCU によって使用される。このパラメータは、独立した 1 個以上のアドレスのリストから成る。それぞれが、全体の論理的なコネクションの一部を指定する。他のオーディオや映像のような情報と同様にデータがこのコネクションを通して運ばれる。このリスト中の 1 つのアドレスには、スイッチドコネクション(集合チャネル)やトランスポートコネクションのための番号文字列、あるいは追加実行 MCU が理解すると仮定した非標準フォーマットのアドレスが指定される。スイッチドコネクション(集合チャネル)の場合、アドレスは表 7-7/JT-T124 に記載されたパラメータを含むように指定される。トランスポートコネクションの時には、アドレスは表 7-8/JT-T124 に記載されたパラメータを含むように指定される。

トランスポートアドレスの NSAP 部は、スイッチドコネクションのものがコネクションレスネットワークプロトコルのものかという情報をそれ自身に含んでいる。この場合、リストに記載された要素は集合チャネルまたは非標準のアドレスの独立した選択肢として存在している。一方では、トランスポートアドレスは不完全でもよく、TTC 標準 JT-T123 や非標準の方法で指定されるように、複数メディア多重化集合チャネルに開かれたデータチャネルの共有に暗黙の内に依存してよい。これらのような場合、ローカル指定の NSAP アドレスとトランスポートセクタの両方又は片方が、正しい終端へのデータチャネルの新しいコネクションを操作するために必要とされるかもしれない。NSAP アドレスは、TTC 標準 JT-T123 にあるように、SETUP 中に発側または着側パーティサブアドレス(Calling or Called Party Subaddress)情報要素として含まれ、トランスポートセクタは X.224 コネクション要求に TSAP 識別子として含まれる。

表 7-7/JT-T124 スイッチドコネクション (集合チャネル) 用のネットワークアドレス(Network Address)パラメータ

(ITU-T T.124)

パラメータ	記述
-------	----

<p>転送モード(Transfer Mode)</p>	<p>これは PSTN 会話コネクション、PSTN 音声バンドデータコネクション、56kbps の CSDN デジタルコネクション、64, 128, 192, 256, 320, 384, 512, 768, 1152, 1472, 1536, 1920 kbps の ISDN デジタルコネクション、ISDN フレームモードコネクション、PSDN パケットモードコネクション、ATM コネクションのいずれかである。これらのなかの 1 つ以上が選択されていなければならない。2 つ以上が選択された場合、追加実行 MCU に選択がまかされたと解釈される。</p> <p>注: 64 kbps を越える ISDN 速度は、TTC 標準 JT-H221 の対応するコードポイントに関連がある。それらは、複数の B または H0 よりむしろ H0、H11、H12 のような単一の複数レートコネクションをあらわす。イニシャルチャネルとアディショナルチャネルが異なった番号で指定されるべき場合のみ、複数のアドレス要素(64kbps そしてまたは 384kbps で)が要求される。そうでない場合、チャネルの数は能力交換とコマンド BAS コードを通して解決できる。MCU が映像切換えによって操作する場合、新しいノードは会議に存在するノードと同じ数のチャネルを割り当てられるか、あるいは第 2 ステータスを与えられるだろう。</p>
<p>国際番号 (International Number)</p>	<p>これは、被追加ノードの完全国際番号を表す 16 桁までの数字列である。</p>
<p>サブアドレス (Sub-Address) (オプション)</p>	<p>これは、ISDN 転送モード時のみ有効な被追加ノードの ISDN サブアドレスを表すオプションのパラメータである。これは、40 桁までの数字列である。</p>
<p>エクストラ番号文字列 (Extra Dialing String) (オプション)</p>	<p>これは一度物理的コネクションが確立された被追加ノードに到達するために追加情報が必要とされることを示すオプションのパラメータである。会話または音声バンドデータコネクションの時には、例えば、確立された音声チャネルを通して送られる DTMF トーンを表す。または、エクストラ番号文字列は、仮想プライベートネットワーク番号を表してもよい。これは、数字の 1 から 9 までか、"#", "*", ","(コンマ)のどれかからなる 255 文字までの文字列である。コンマは、追加実行 MCU が後続する文字の前に 1 秒遅延を挿入するという意味を持つ。</p>
<p>高位レイヤコンパチビリティ情報 (High Layer Compatibility Information) (オプション)</p>	<p>これは、ISDN 転送モード時のみ有効なオプションのパラメータで、コネクションのこの部分を使用するための操作モードを示す。この情報はいくつかの国での接続で必要とされる。操作モードは、3kHz 帯域幅の電話、7kHz 帯域幅の電話、ビデオ電話、ビデオ会議、オーディオグラフィック、オーデ</p>

	イオビジュアル、マルチメディアの中の 1 つ以上のものである。これらの2つ以上のものが選択された場合、追加実行 MCU は示されたモードの 1 つを自由に使用してよい。これらの選択肢の定義は追加実行 MCU の操作をする国に依存してよい。
--	---

表 7-8/JT-T124 トランスポート接続のためのネットワークアドレス(Network Address)パラメータ (ITU-T T.124)

パラメータ	記述
NSAP アドレス(NSAP Address)	これは 20 オクテットまでのオクテットストリングである。これは被追加ノードの NSAP アドレスの ITU-T 勧告 X.213 Annex A clause A.8.3.1 による優先バイナリエンコーディングである。
トランスポートセレクタ (Transport Selector) (オプション)	これは、被追加ノードでトランスポートサービスアクセスポイントを選択するために使用されるオプションのパラメータである。

追加実行 MCU ノード ID(Adding MCU Node ID):これは、特定の MCU ノードを指定するために使用される任意のパラメータである。この MCU ノードにより着側ノードが追加される。このパラメータが指定されない場合、着側ノードは会議で利用できる MCU のどれかにより追加される。この場合、指示はトップ GCC プロバイダから発行される。その MCU のノードコントローラは、ローカルにサービスしてもよいし、特定の MCU へもう一つの要求を発行してもよい。

要求ノード ID(Requesting Node ID):要求するノードのノード ID(Node ID)。

ユーザデータ(User Data):課金、認証のようなこの標準の範囲外の機能のために使用してもよいオプションのユーザデータ。

結果(Result): 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。「成功(successful)」、「無効な会議(Invalid conference)」、「無効な追加実行 MCU(invalid adding MCU)」、「召集者でも召集者指名ノードでもない(not convener or convener-designated node)」、「無効ネットワーク型(invalid network type)」、「無効ネットワークアドレス(invalid network address)」、「被追加ノードビジー(added node busy)」、「ネットワークビジー(network busy)」、「接続失敗(connection unsuccessful)」、「利用可能ポートなし(no ports available)」。

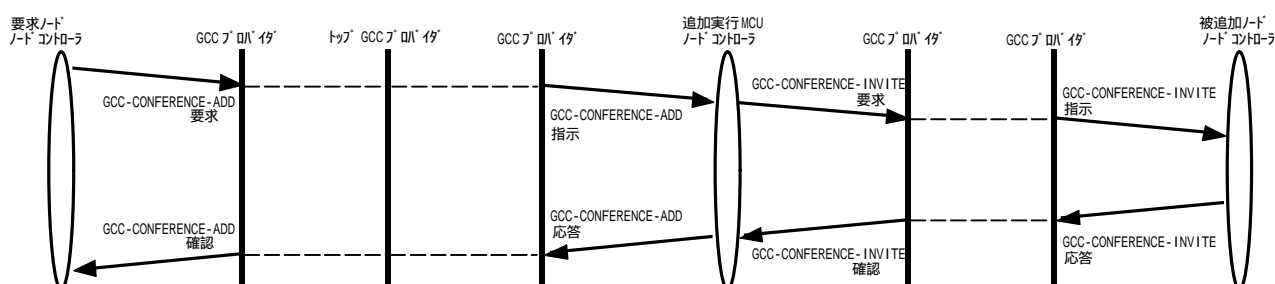


図 7-5/JT-T124 GCC-CONFERENCE-ADD プリミティブのシーケンス(直接の場合)

(ITU-T T.124)

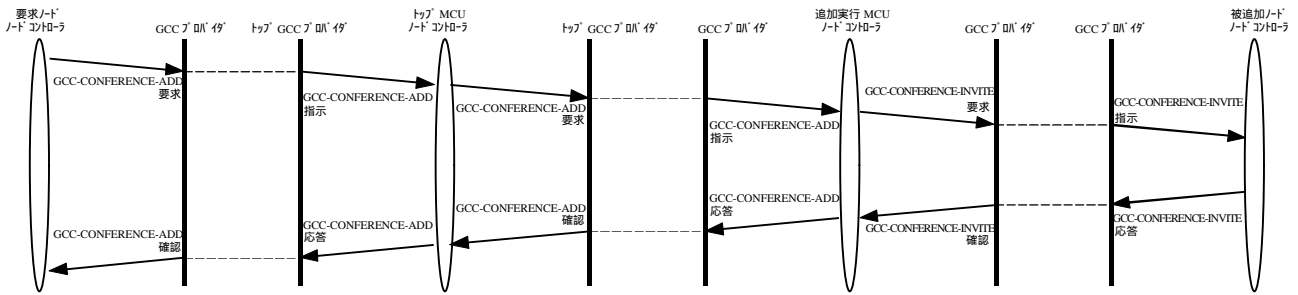


図 7-6/JT-T124 GCC-CONFERENCE-ADD プリミティブのシーケンス(非直接の場合)

(ITU-T T.124)

7.1.2.6. GCC-CONFERENCE-LOCK

GCC-CONFERENCE-LOCK-要求プリミティブはノードコントローラによって会議をロックするために使用され、他のノードが会議に入ることを防ぐ。このプリミティブは召集者が召集者指名ノードが行った時のみ有効である。ロックしている間、参加者は GCC-CONFERENCE-ADD プリミティブのみによって会議に参加する。ノードとトップ GCC プロバイダとの間で交換される GCC-CONFERENCE-LOCK プリミティブと GCC-CONFERENCE-UNLOCK プリミティブの順序は維持される。表 7-9/JT-T124 はこのプリミティブの型とパラメータを示す。図 7-7/JT-T124 はこのプリミティブを使う時のイベントシーケンスである。

表 7-9/JT-T124 GCC-CONFERENCE-LOCK プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	指示	応答	確認
会議 ID(Conference ID)	M	M	M(=IN)	M(=RQ)
発信元ノード ID(Source Node ID)		M	M(=)	
結果(Result)			M	M(=)

会議 ID(Conference ID):プリミティブが参照する会議の識別子。

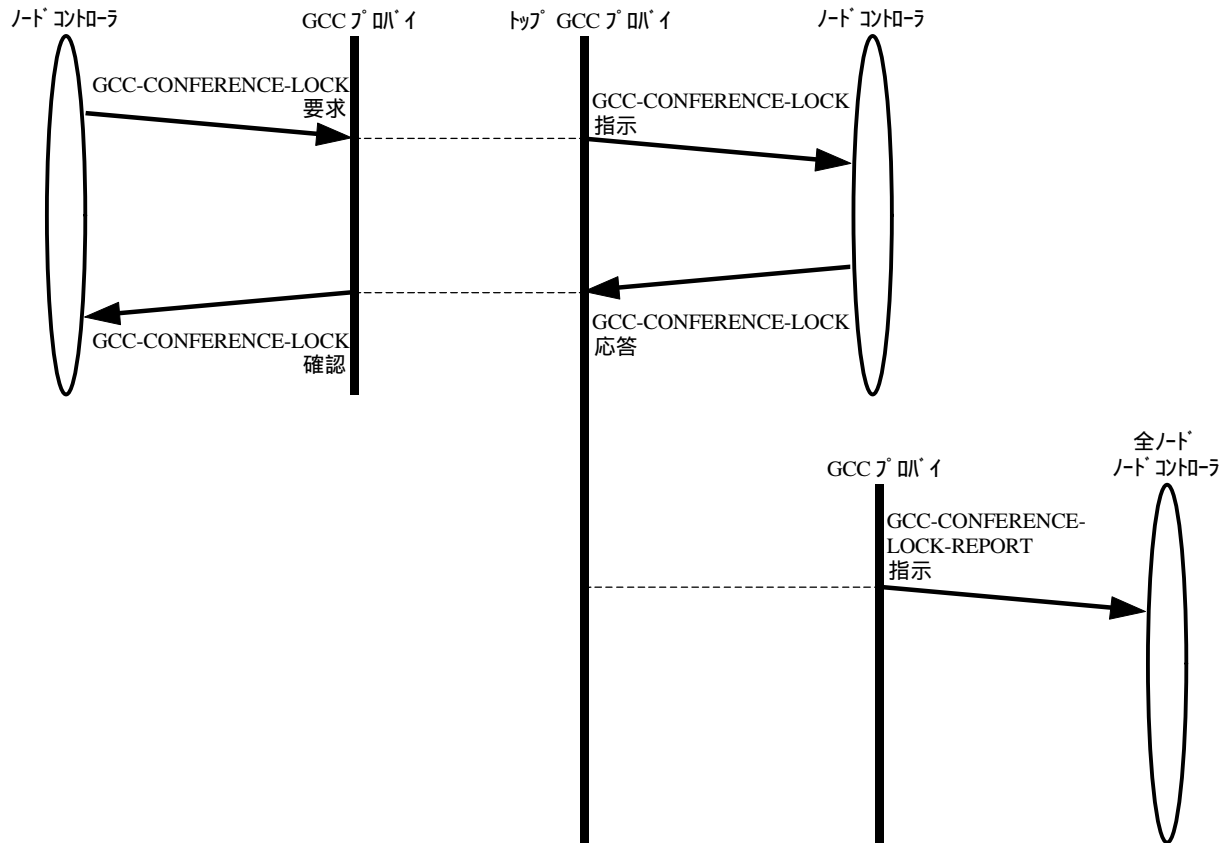
発信元ノード ID(Source Node ID):要求しているノードのノード ID(Node ID)。

結果(Result):要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、リスト中の可能な結果の一つを含む。: 「成功(successful)」、「無効な会議(invalid conference)」、「機能は利用不可(function not available)」、「ロック済み(already locked)」、「召集者でも召集者指名ノードでもない(not convener or convener-designated node)」

図 7-7/JT-T124 GCC-CONFERENCE-LOCK プリミティブのシーケンス
(ITU-T T.124)

7.1.2.7. GCC-CONFERENCE-UNLOCK

GCC-CONFERENCE-UNLOCK-要求プリミティブはノードコントローラによってロックしてある会議を解除するために使われる。会議のアンロックは他のノードがダイアリングによって会議に参加することを認める。もし会議がパ



スワード(Password)プロテクションの使用を指定して創られたなら、Meet-me 形式でアンロックされている会議に参加しようと試みるいかなる参加者もパスワード(Password)を要求される。このプリミティブは召集者か召集者指名ノードが行った時のみ有効である。ノードとトップ GCC プロバイダとの間で交換される GCC-CONFERENCE-LOCK プリミティブと GCC-CONFERENCE-UNLOCK プリミティブの順序は維持される。表 7-10/JT-T124 はこのプリミティブの型とパラメータを示す。図 7-8/JT-T124 はこのプリミティブを使う時のイベントシーケンスである。

表 7-10/JT-T124 GCC-CONFERENCE-UNLOCK プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	要求	指示	応答	確認
会議 ID(Conference ID)	M	M	M(=IN)	M(=RQ)
発信元ノード ID(Source Node ID)		M	M(=)	
結果(Result)			M	M(=)

会議 ID(Conference ID):プリミティブが参照する会議の識別子。

発信元ノード ID(Source Node ID):要求しているノードのノード ID(Node ID)。

結果(Result):要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、リスト中の可能な結果の一つを含む。: 「成功(successful)」、「無効な会議(invalid conference)」、「機能は利用不可(function not available)」、「アンロック済み(already unlocked)」、「召集者でも召集者指名ノードでもない(not convener or convener-designated node)」

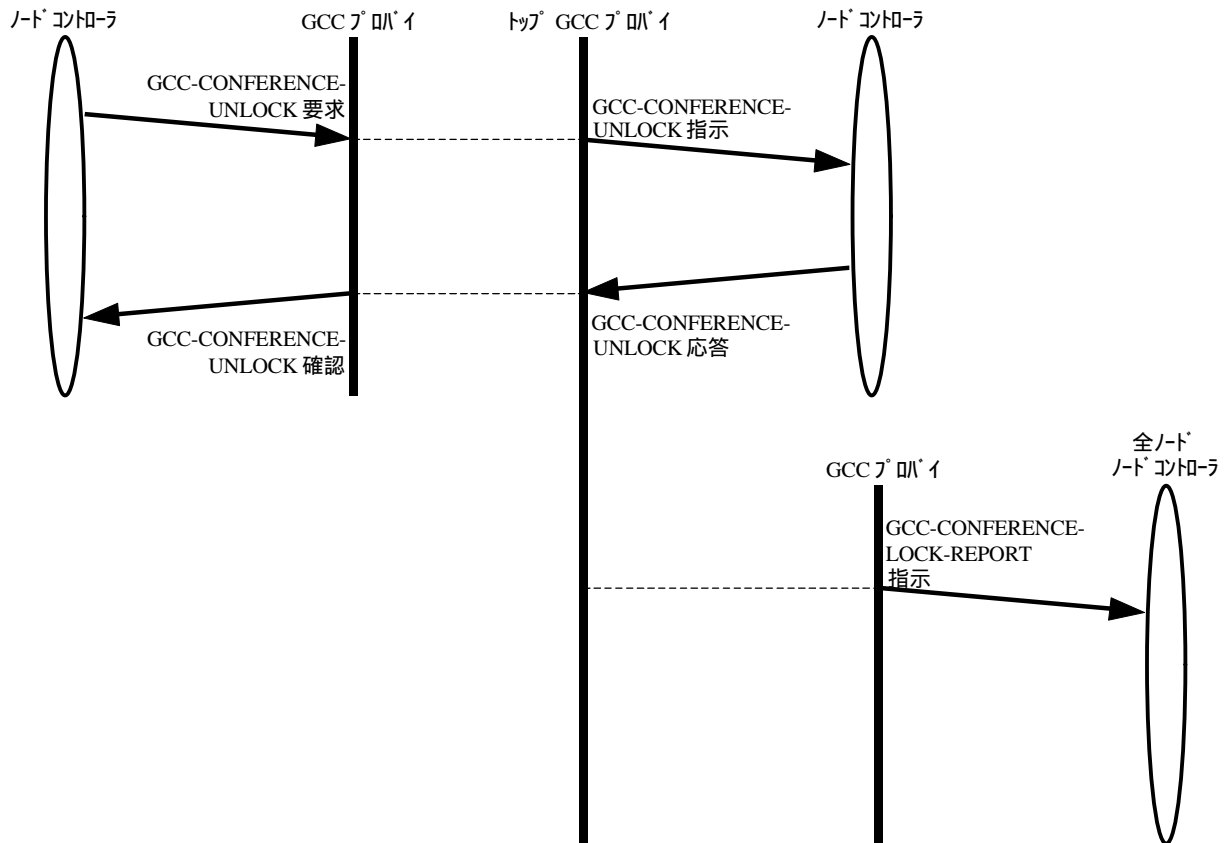


図 7-8/JT-T124 GCC-CONFERENCE-UNLOCK プリミティブのシーケンス (ITU-T T.124)

7.1.2.8. GCC-CONFERENCE-LOCK-REPORT

GCC-CONFERENCE-LOCK-REPORT-指示プリミティブは成功した GCC-CONFERENCE-LOCK-要求や成功した GCC-CONFERENCE-UNLOCK-要求の結果として会議中の全ノードのノードコントローラに通知される。図 7-7/JT-T124 と図 7-8/JT-T124 はこのプリミティブの使用を導くイベントシーケンスを示す。

表 7-11 /JT-T124 GCC-CONFERENCE-LOCK-REPORT プリミティブの型とパラメータ (ITU-T T.124)

パラメータ	指示
会議 ID(Conference ID)	M
ロック / アンロック (Locked/Unlocked)	M

会議 ID(Conference ID):ロックされている会議に対応する MCS ドメインの識別子。

ロック/アンロック(Locked/Unlocked):会議がロック状態とアンロック状態のどちらに切換えられたかを表わすフラグ。

7.1.2.9. GCC-CONFERENCE-DISCONNECT

GCC-CONFERENCE-DISCONNECT-要求プリミティブはノードコントローラによって会議からそれ自身を切断するために使用される。会議からの切断とは対応する物理的なコネクションの切断を必然的に意味する訳ではない。端末は、会議から切断されると、さらに別の会議に参加するかもしれない。GCC は、ノードが異常な手段によって会議から切断した事（例えば、物理的な呼が切断された事）を検出すると、会議に残っている全てのノードに GCC-CONFERENCE-DISCONNECT-指示を送る。表 7-12/JT-T124 はこのプリミティブの型とパラメータを示す。図 7-9/JT-T124 と図 7-13/JT-T124 は切断を始めたクライアントに対してこのプリミティブを使う時のイベントシーケンスを示す。図 7-11/JT-T124 は異常切断を始めた GCC の場合を示す。

表 7-12/JT-T124 GCC-CONFERENCE-DISCONNECT プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
切断理由 (Reason for disconnect)		M	
切断ノード ID(Disconnecting Node ID)		M	
結果(Result)			M

会議 ID(Conference ID):プリミティブが参照する会議の識別子。

切断理由(Reason for disconnect):会議からの切断の理由の表示。「ユーザによる(user-initiated)」、「排除されたノード(ejected node)」、「不明(unknown)」の何れかである。

切断ノード ID(Disconnecting Node ID):切断されたノードに対応するノード ID(Node ID)。

結果(Result):要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、リスト中の可能な結果の一つを含む。: 「成功(successful)」、「無効な会議(invalid conference)」

注: 接続確立の完了の前(例えば GCC-CONFERENCE-JOIN-確認の受信前)にノードコントローラが会議から切断する事を許す必要があるならば、会議を確立している状態で切断される会議の結合のローカルな方法を提供する事が必要とされる。会議 ID(Conference ID)が会議を生成するか会議に参加する後までノードコントローラによって知られていないことがこの理由である。そうするためのメカニズムはこの標準の範囲外のローカルの問題である。

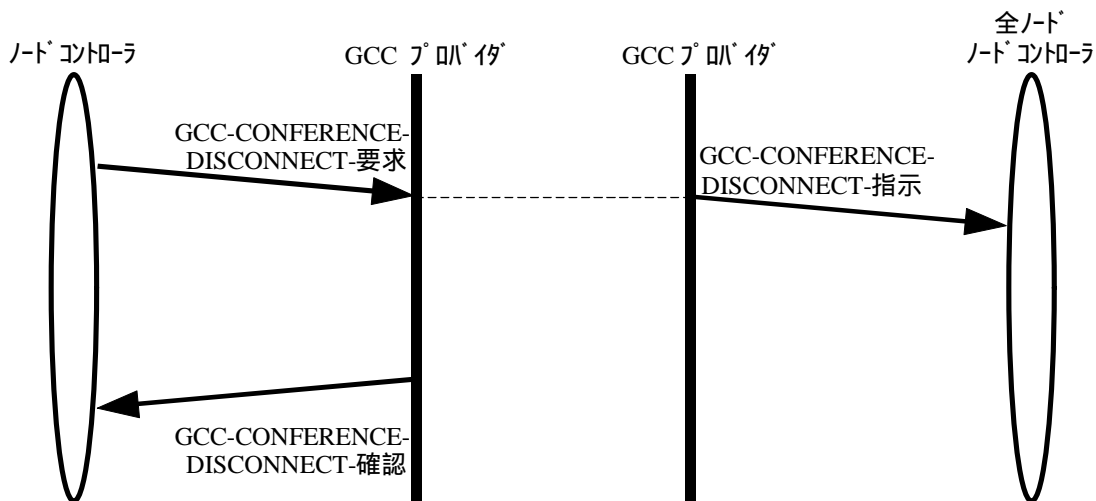


図 7-8/JT-T124 GCC-CONFERENCE-DISCONNECT プリミティブのシーケンス (クライアントからの起動)
(ITU-T T.124)

7.1.2.10. GCC-CONFERENCE-TERMINATE

GCC-CONFERENCE-TERMINATE-要求プリミティブはノードコントローラによって全体の会議を終了するために使われる。このプリミティブは召集者が召集者指名ノードのみが発行する。会議の終了とは対応する物理的な接続の切断を必然的に意味する訳ではない。表 7-13/JT-T124 はこのプリミティブの型とパラメータを示す。図 7-10/JT-T124、図 7-11/JT-T124、図 7-12/JT-T124 はクライアントから始めた場合と GCC から始めた場合のこのプリミティブの使用時のイベントのシーケンスを示す。会議が自動的に終了する会議として生成されていてすべての他ノードがその会議から切断していた場合、GCC から始まるケースはプロトコルの下位レイヤかトップ GCC プロバイダでの異常切断という結果となる。

注：多くの場合、GCC-CONFERENCE-TERMINATE-指示の受信は、全体の会議が終了している事、すなわち、全員が切断された事を意味する。異常終了が理由コードとして与えられる場合では、このプリミティブはローカルなノードが会議から不意に外された事を意味するが、全体の会議が終了した事を必然的に意味する訳ではない。

表 7-13/JT-T124 GCC-CONFERENCE-TERMINATE プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
要求ノード ID(Requesting Node ID)		C	
終了理由 (Reason for termination)	O	O(=)	
結果(Result)			M

会議 ID(Conference ID): プリミティブが参照する会議の識別子。

要求ノード ID(Requesting Node ID): 終了を始めたノードのノード ID(Node ID)。このパラメータは GCC の始めた終了の場合を除いて存在する。

終了理由(Reason for termination):会議の終了の理由の表示。それは、リスト中の可能な結果の一つを含む。: 「正常終了(requested normal termination)」、「時間制限付き会議の終了(requested timed conference termination)」、「自動終了会議での参加者無し(no more participants in automatically terminating conference)」、「異常終了(error termination)」

結果(Result):要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、リスト中の可能な結果の一つを含む。: 「成功(successful)」、「無効な会議(invalid conference)」、「召集者でも召集者指名ノードでもない(not convener or convener-designated node)」

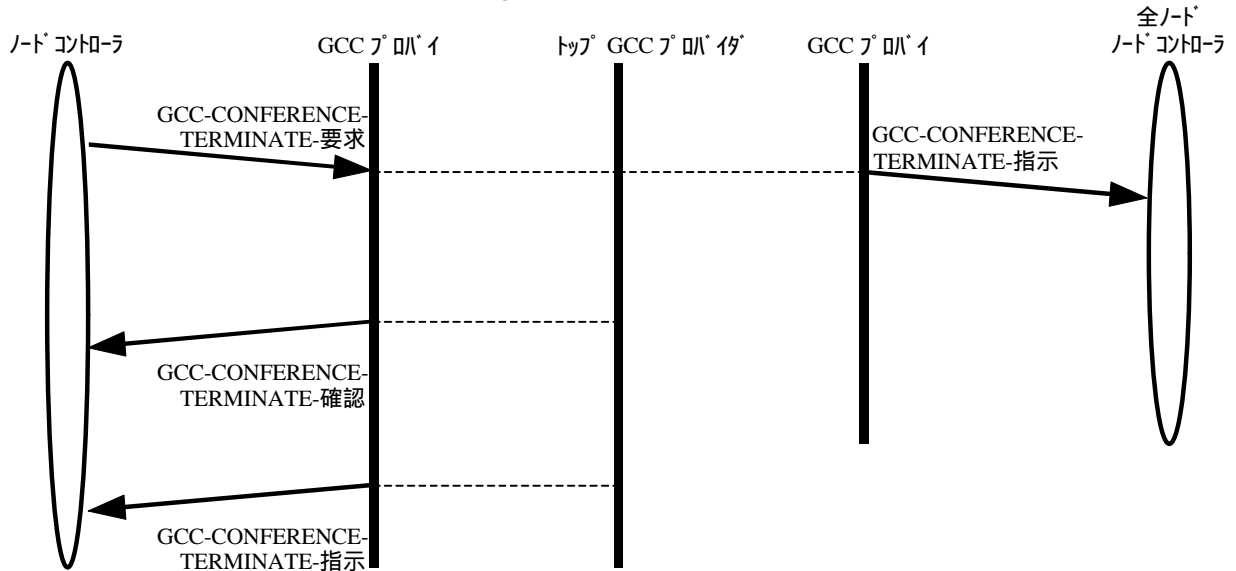


図 7-10/JT-T124 GCC-CONFERENCE-TERMINATE プリミティブのシーケンス (クライアントからの起動)

(ITU-T T.124)

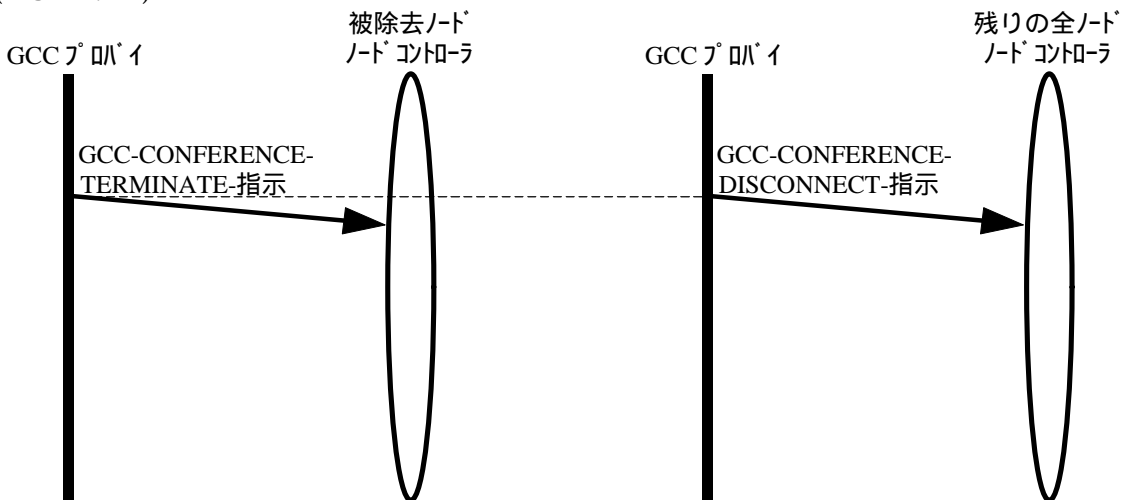


図 7-11/JT-T124 GCC-CONFERENCE-TERMINATE プリミティブのシーケンス (異常終了)

(ITU-T T.124)

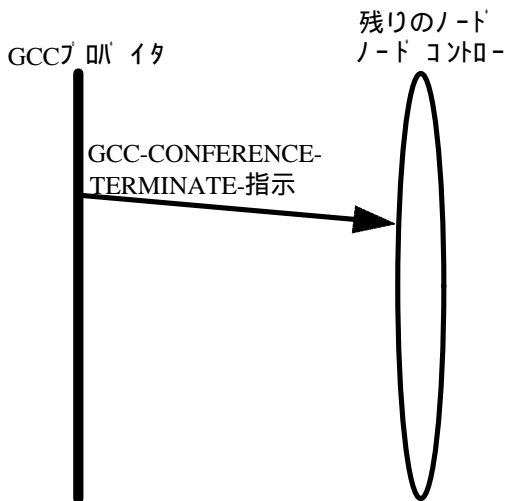


図 7-12/JT-T124 GCC-CONFERENCE-TERMINATE プリミティブのシーケンス (自動終了)
(ITU-T T.124)

7.1.2.11. GCC-CONFERENCE-EJECT-USER

GCC-CONFERENCE-EJECT-USER-要求プリミティブはノードコントローラによって特定のノードを会議から強制的に切断するために使用される。このプリミティブは召集者か召集者指名ノード、またはコネクション階層構造中の被排除ノードの直上ノードのみが発行する。会議から排除される事は物理的なコネクションの切断を必然的に意味する訳ではない。ノードが排除される時、GCC-CONFERENCE-DISCONNECT-指示をそのノードが排除により切断された事を示すために会議に残っている全てのノードのノードコントローラに発行する。表 7-14/JT-T124 はこのプリミティブの型とパラメータを示す。図 7-13/JT-T124 はこのプリミティブを使う時のイベントのシーケンスを示す。

表 7-14/JT-T124 GCC-CONFERENCE-EJECT-USER プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
排除ノード ID(Ejected Node ID)	M	M(=)	M(=)
排除理由(Reason for ejection)	O	O(=)	
結果(Result)			M

会議 ID(Conference ID):プリミティブが参照する会議の識別子。

排除ノード ID(Ejected Node ID):排除されるノード。

排除理由(Reason for ejection):排除の理由の表示: 「ユーザからの起動(user-initiated)」、「高位ノードの切断(higher node disconnected)」、「高位ノードの排除(higher node ejected)」

結果(Result):要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、リスト中の可能な結果の一つを含む: 「成功(successful)」、「無効な会議(invalid conference)」、「排除に無効なユーザ(invalid user to eject)」、「召集者でも召集者指名ノードでもない(not convener or convener-designated node)」

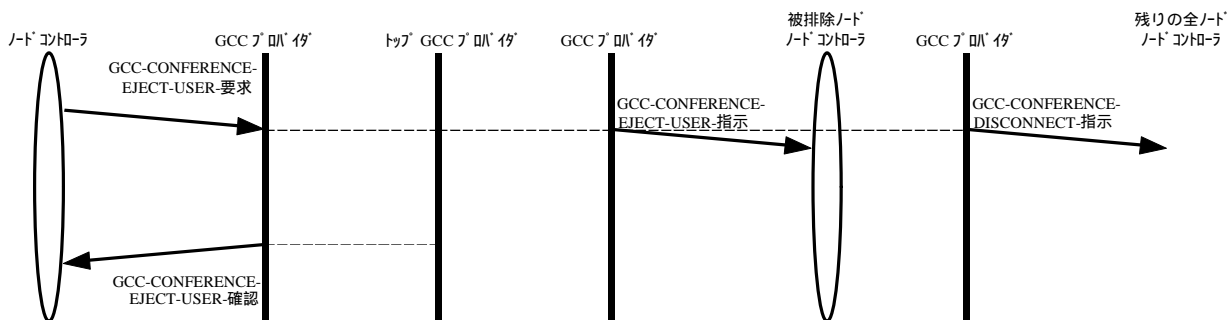


図 7-13/JT-T.124 GCC-CONFERENCE-EJECT-USER プリミティブのシーケンス (クライアントからの起動)
(ITU-T T.124)

7.1.2.12. GCC-CONFERENCE-TRANSFER

GCC-CONFERENCE-TRANSFER-要求プリミティブはノードコントローラによって会議の中の選ばれたノードをその会議から切断し、別の会議に参加させるために使用される。このプリミティブは召集者が召集者指名ノードのみが発行する。会議の中のいくつかの MCU は移動の前に移動元と移動先の両方に含まれているかもしれない。もしそうならば、これらの MCU はこの要求の中の移動先ノードのリストの中に含まれるべきではない。もし MCU が両方の会議には参加していないが、移動後に、両方の会議に参加するノードに接続されるならば、MCU は GCC-CONFERENCE-TRANSFER-要求を発行する前に移動先会議に参加しなければならない。転送ノードのリストに含まれるいかなる MCU も新しいGCC-CONFERENCE-JOIN-指示処理の前に移動作業(元の会議から切断して先の会議に参加する事)を完了しておかなければならない。これはコネクション階層構造の中でその MCU の下のノードからの加入の要求を成功させる。移動処理中で、移動元の会議の GCC-CONFERENCE-TERMINATE-指示を受けるノードは、切断をしていなければ、切断無しに直接移動先の会議に参加する処理に入る。そのノードが接続される MCU もまた移動するように指示されたなら、この状況は起こる。表 7-15/JT-T124 はこのプリミティブの型とパラメータを示す。図 7-14/JT-T124 はこのプリミティブを使用時のイベントのシーケンスを示す。

表 7-15/JT-T124 GCC-CONFERENCE-TRANSFER プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
移動先会議名 (Destination Conference Name)	M	M(=)	M(=)
移動先会議名修飾子 (Destination Conference Name Modifier)	O	O(=)	O(=)
移動先ネットワークアドレス (Destination Network Address)	O	O(=)	
転送ノード(Transferring Nodes (List of Node IDs or null))	O		O(=)
パスワード>Password)	C	C(=)	
結果(Result)			M

会議 ID(Conference ID):指定されたノードが移動操作の前に参加していた会議に対応した MCS Domain の識別子。

移動先会議名(Destination Conference Name):指定されたノードが参加するように指示された会議の名前。これはその MCU か指定されたノードに接続されている MCU においてその会議が知られている名前である。 会議が会議名(Conference Name)を示す数字とテキストの形式を使って生成されたのなら、いずれかの形式がこのプリミティブ中で使用される。

移動先会議名修飾子(Destination Conference Name Modifier):移動先の会議の中の MCU がオプションの会議名修飾子(Conference Name Modifier)を選択したら、この MCU に接続される移動されるべきノードは、無修飾の(若しくは別に修飾された)基本の会議名(Conference Name)を使う MCU に接続されるそれらのノードとは別の GCC-CONFERENCE-TRANSFER プリミティブの交換で、別々に移動されなければならない。要求しているノードが様々な MCU で会議名修飾子(Conference Name Modifier)を知るようになる方法は、この標準の範囲外である。

移動先ネットワークアドレス(Destination Network Address):移動しているノードによって発行されるべき GCC-CONFERENCE-JOIN-要求の中に含まれるべきオプションの着信アドレス(Called Address)パラメータ。

転送ノード(Transferring nodes (List of Node IDs or null)): 指示が送信されるべき GCC プロバイダを識別するノード ID(Node ID)のリストか、会議 ID(Conference ID)で指定された会議中の全ノードに指示が送信されるべきであることを示すための省略。

パスワード>Password): このパラメータは移動するノードが新しい会議に参加する為に GCC-CONFERENCE-JOIN-要求の中で使うパスワード>Password)を指示する。これは数値列であるか、ユニコードの区 00 のテキスト列である(最大 255 桁か文字)。このパラメータはこの会議で平文パスワード>Password In The Clear Required)フラグが設定されている場合に限り使用される。

結果(Result): 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、リスト中の可能な結果の一つを含む。: 「成功(successful)」, 「無効な会議(invalid conference)」, 「召集者でも召集者指名ノードでもない(not convener or convener-designated node)」

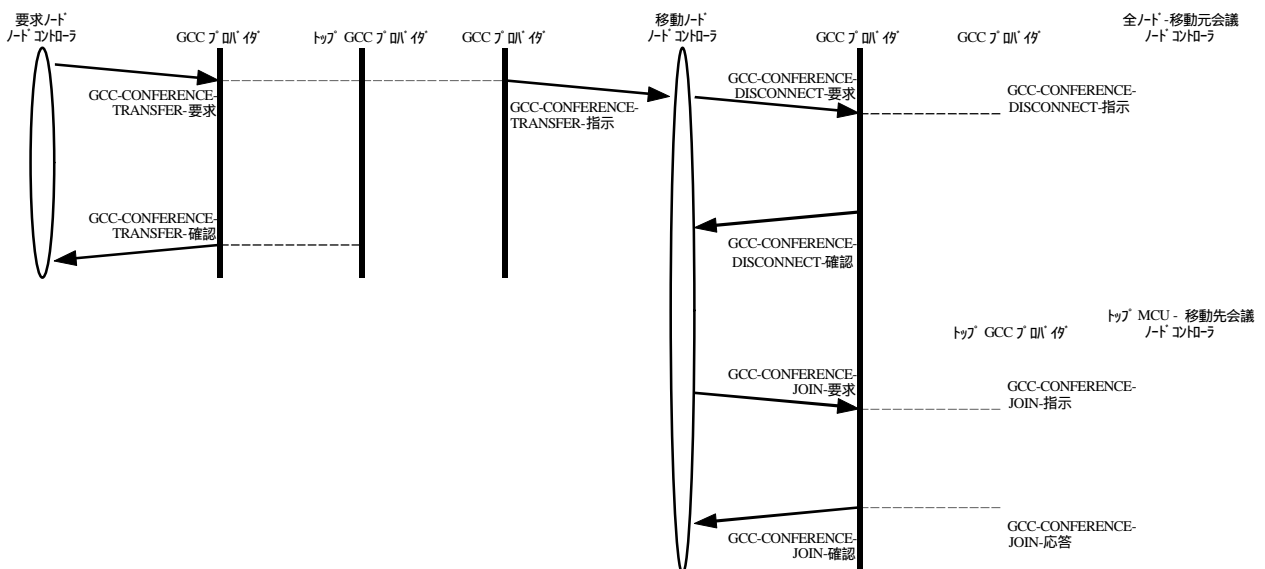


図 7-14/JT-T124 GCC-CONFERENCE-TRANSFER プリミティブのシーケンス (ITU-T T.124)

7.1.3.会議確立条件

物理接続の両側が、お互いに相手の動作を待つ事で GCC 会議設定手順を始めるのに失敗するデッドロック状態となる可能性を回避するために、物理回線を確立する(つまり、2 地点間の接続に TTC 標準 JT-T123 の PSTN,ISDN,CSDN を使う)時の会議確立手順のために、以下の条件が定義される。これらの条件は、事前の取り決めがこの標準の範囲外で定義される情報の交換の何れかによる、関係ノード間の双方の合意によって代えられる。

- まず初めに、どちらが発側ノードでどちらが着側ノードかを物理接続の両端が認識しなければならない。また、お互いのノードタイプ(端末、マルチポート端末、MCU)を認識しなければならない。この情報は GCC-CONFERENCE-QUERY プリミティブを使って得るか、事前情報として通知される。着側ノードか発側ノードかを決定する際、両ノードともどちらに当たるのかを知らないならば、GCC-CONFERENCE-QUERY プリミティブで定義される非平衡指示子パラメータによる決定方法を使用し、任意に一方のノードを会議確立手順のための着側ノードとして選択する。
- 発側ノードは初期会議確立手順(新しい会議の生成要求、着側ノード上の会議への加入、着側ノードを会議に招待の何れか)を開始する責任を負う。一般に、その行動が発側ノードの行動を妨げない事を保証する責任がこの場合あるが、着側ノードが同じ物理接続上の会議を確立する行動をとる事を妨げるものではないことに注意。
- 様々なノードタイプ(端末、マルチポート端末、MCU)の間で呼を確立する時、発側ノードの最初の接続を開始する行動に対して、表 7-16/JT-T124 の制約事項が設定される。表に示される行動の定義 1,2,3,4 は次の通りである:
 1. 発側ノードは着側ノードでの新しい会議の生成を要求する。
 2. 発側ノードは着側ノード上の既存の会議への加入を試みる。
 3. 発側ノードはローカルに会議を生成し、着側ノードを招待する。
 4. 発側ノードは着側ノードを既存の会議に招待する。

表 7-16/JT-T124 会議確立のための発側ノードの行動

(ITU-T T.124)

発側ノード	着側ノード		
	端末	マルチポート端末	MCU
端末	1 又は 3	1 又は 2	1 又は 2
マルチポート端末	3 又は 4	1, 2, 3 又は 4	1 又は 2
MCU	3 又は 4	3 又は 4	1, 2, 3 又は 4

7.1.4. 会議確立手順の例

会議確立は様々な条件下で様々な手法で行われる。最も簡単な会議は呼の中に MCU が含まれないポイントポイント(point-to-point)のものである。1 個かそれ以上の MCU を通して会議が確立される場合には、一般的に呼は(全ての参加者が 1 つの MCU に発呼する)meet-me 型か、(MCU(複数可)が全ての参加者に発呼することにより呼を設定する)call-out 型か、(1 人の参加者が 1 つの MCU に発呼し、MCU に発呼されることによりその他の参加者を加える)call-through 型で行われる。

7.1.4.1. Meet-me 会議確立

meet-me 会議では、会議はある MCU で確立され、端末ノードは(必要ならば他の MCU も同様に)その MCU に発呼し会議に加入する。他の MCU がその会議に加入しているならば、端末ノードはこれらの MCU のいずれかに発呼し、会議に加入できる。

meet-me 会議の最初の生成は何らかの方法で(例えば、MCU でローカルに初期化される)行われるか、MCU に発呼する最初のノードにより生成される。前者では、GCC は関係しない。後者では、会議は、召集者ノードから MCU へのコネクション上に GCC-CONFERENCE-CREATE-要求を発行することにより、MCU で生成される。両方の場合とも、会議を生成したノードがトップ GCC プロバイダとなる。召集ノード(前者では MCU 自身、後者では要求したノード)が会議召集者として特別な立場を認められる。

会議が遠隔的に生成されたとき(GCC-CONFERENCE-CREATE-要求を用いて)、会議名(Conference Name)が要求プリミティブにおいて指定される。その名前がその MCU で既に使われているならば、名前が MCU でローカルにユニークになるように、MCU のノードコントローラによって会議名修飾子(Conference Name Modifier)が割り当てられる。他のノードがその MCU の会議に加入を試みる時、この会議名修飾子(Conference Name Modifier)は加入要求の一部として指定されなければならない。

会議が遠隔的に生成されたとき、オプションとして召集者パスワード(Convener Password)も含むことができる。このパスワードは、召集者が会議から抜けて、その後再加入するとき召集者特権の継続を望む場合にのみ必要である。

一般的に meet-me 会議は他のノードが加入できるようにロックなしに指定される。meet-me 会議は、オプションとして、望まれないノードが会議に加入することを防ぐパスワード>Password)保護を要求するように生成できる。

meet-me 会議に加入するノードは、その会議に加入している MCU のいずれかへのコネクション上に GCC-CONFERENCE-JOIN-要求を発行する。一般的に、ノードは加入する会議の名前を前もって知らない。この場合、会議に加入する前に、ノードは加入可能な会議のリストを MCU に問い合わせることができる。これは、GCC-CONFERENCE-QUERY-要求を発行することにより行われる。応答は GCC-CONFERENCE-QUERY-確認であり、端末が接続するノードの種類(この場合 MCU)と、現在その MCU が加入している全てのリスト記載会議のリストを示す。このリストは、それぞれの会議の会議名(Conference Name)や、パスワード>Password)保護されているかどうか(会議に加入しようと試みる前にユーザからパスワード>Password)を要求するために使われる)などの会議の他の特性と、必要ならば会議名修飾子(Conference Name Modifier)を含む。一旦、会議名が選択されると、所望した会議の会議名(Conference Name)及び必要なら会議名修飾子(Conference Name Modifier)を指定した GCC-CONFERENCE-JOIN-要求を発行することにより、会議に加入できる。会議がパスワード>Password)を要求するならば、それを加入要求に含める。

一般的に、加入する会議の事前知識がない meet-me 会議の場合は、ポイント ポイント会議の場合と等しくなることに注意を要する。すなわち、それぞれの場合で事前知識が同じであるから、会議を立ち上げる時に使われるイベントシーケンスは、ポイント ポイント会議のものと区別することはできない。区別できるのは、GCC-CONFERENCE-QUERY-確認中の識別子が、直接接続しているノードが MCU であり、そこに加入する実行中の会議があることを示し

ているという点からだけである。

何らかの方法で会議名が加入しようとするノードに通知されている場合は、GCC-CONFERENCE-JOIN-要求中で会議名(Conference Name)を指定することによって、加入可能な会議を最初に問い合わせずに、ノードは直接会議に加入できる。ある会議に加入するためにノードが接続できる複数の独立した MCU があるならば、名前の衝突のために、いくつかの MCU で会議名修飾子が必要となることがありうる。この場合、会議に加入するノードは、これらの MCU の 1 つに接続するときに、その MCU の会議名修飾子(Conference Name Modifier)を指定しなくてはならない。GCC-CONFERENCE-QUERY プリミティブを使用せずにこの修飾子を決定するのは難しいので、いくらかでも名前の衝突の可能性がある場合には、meet-me 会議に加入することを試みる前に GCC-CONFERENCE-QUERY を使うことを推奨する。また、meet-me 会議を最初に生成するときには、会議記述(Conference Description)を使うことも推奨する。この場合、同じ会議名(Conference Name)を持つ複数の会議を、会議記述(Conference Description)の違いによって区別できる。しかしながら、一般的に、meet-me 会議を生成するときに、会議名修飾子(Conference Name Modifier)の必要なしに全ての MCU でユニークとなるような会議名(Conference Name)を選択する方がより良い。

7.1.4.2. Call-out 会議確立

call-out 会議確立においては、会議は MCU でローカルに生成され、会議参加者はその MCU によって発呼され会議に招待される。一般的に会議はロックされている会議として生成され、また非リスト記載として指定されることもある。

召集する MCU は、順番に各ノードを会議に招待し、それに続く動作として会議の参加者に通じる各端末への物理コネクションを作る。これは GCC-CONFERENCE-INVITE-要求を招待するノードに発行することにより行われる。MCU で会議が生成されるので、その MCU が call-out 会議のトップ GCC プロバイダとなる。

7.1.4.3. Call-through 会議確立

call-through 会議は、最初に会議が初期化端末により遠隔的に生成されることを除くと、call-out の場合と非常に似ている。この場合、GCC-CONFERENCE-CREATE-要求を使用することにより、端末は MCU に接続し会議を生成する。call-out の場合と同様に、一般的に(会議は)ロックされていて非リスト記載である。一般的に、call-through 会議は、全てのノードが切断した時に終了する自動終了会議として生成される。一般的に、ノードが会議に明示的に加入する必要がないので、call-through 会議には NULL の会議名(Conference Name)が指定される。召集するノードに NULL 会議名の会議が既に存在する場合は、ノードコントローラが会議名(Conference Name)として使用するための任意でユニークな名前を単純に選択する。どちらの場合でも、加入するために用いられることは決してないので、名前を人が読めるようにする必要はない。

7.1.4.4. ポイント ポイント(Point-to-Point)会議確立

MCU が存在せず 2 つの端末だけしか含まれないと言う点でポイント ポイント会議は他種と区別される。どちらの端末が発側端末か(物理コネクションの発起人)、着側端末かが知られている場合、ポイント ポイント会議は、GCC-CONFERENCE-QUERY-要求を発行して、発側端末が着側端末に最初に問い合わせることにより確立される。これにより、そのための事前知識を要求すること無く、端末が他方のノードを端末か、MCU か、マルチポート端末かを判断することができる。ポイント ポイントの呼の場合、この要求の応答として生成された GCC-CONFERENCE-QUERY-確認は、直接接続したノードがユーザ端末であることを示す。一旦、直接接続したノードが端末であると解れば、発側

端末が、新しい会議を生成するために GCC-CONFERENCE-CREATE-要求を発行するか、ローカルに会議を生成し GCC-CONFERENCE-INVITE-要求を発行し他方の端末を招待することにより、会議を確立する。一般的に、会議は"0"のような任意の会議名(Conference Name)で指定され、ロックなし、非リスト記載、自動終了とされる。

どちらの端末が発側端末か、着側端末かをノードが知らない場合は、端末は相手のノードが発側ノードか着側ノードかを判断するために、(すでに、この情報を持つ GCC-CONFERENCE-QUERY-指示を受信していなければ)GCC-CONFERENCE-QUERY-要求を発行する。この要求の中の非平衡指示子(Asymmetry Indicator)パラメータで、ローカルノードが発側ノードか着側ノードか不明であるということを示す。相手のノードが自身についてのこの知識を持っているならば、結果として GCC-CONFERENCE-QUERY-確認の中でそのように示す。そうすれば、どちらが発側か知ることができ、上述のような動作がとられる。両者とも着側/発側状態を知らなければ、確認(または、受信した GCC-CONFERENCE-QUERY-指示の内容)には、不明設定が指定される。このパラメータの不明設定は 32bit の乱数を含む。この場合、その乱数が平衡である(どちらにも決まらない)という結果を解決するために用いられる。2 つの乱数のうち小さい方を生成したノードが着側ノードと考えられ、会議の確立を試みないようにする。大きい数を生成したノードは発側ノードと考えられ、会議を確立しようと試みる。

端末が、呼が 2 つの端末間のポイント ポイントであるという事前知識を持っているならば(かつ、その端末が着側端末か、発側端末かどうかを知っている場合)、端末は GCC-CONFERENCE-QUERY-要求を発行する必要はない。そのかわり、その端末が発側端末ならば、ただちに GCC-CONFERENCE-CREATE-要求か、GCC-CONFERENCE-INVITE-要求を発行する。この事前知識を持つのが着側端末であるならば、GCC-CONFERENCE-QUERY-要求をスキップすることもでき、単に GCC-CONFERENCE-CREATE-指示か GCC-CONFERENCE-INVITE-指示を受信するのを待つ。相手の端末がコネクションのタイプの事前知識を持たないならば、ローカル端末が、応答を要求される GCC-CONFERENCE-QUERY-指示を相手端末から受信することがある。

7.1.4.5. マルチポート端末間の会議確立

一般的に、マルチポート端末は端末として扱われる装置だが、MCU が行うように同時に複数のノードとコネクションを確立する能力を持つ。端末かマルチポート端末が、(事前知識か、GCC-CONFERENCE-QUERY の交換を介してかのどちらかにより)マルチポート端末と解っているノードに発呼するときの動作は、マルチポート端末に使用可能な会議が存在するかどうか、および、これらの会議の特性に依存する。

一般的に、この種のコネクションは、meet-me 会議のようにではなく、むしろポイント ポイントのように自動的に生成されることが望まれる。特に、着側ノードも発側ノードもまだ、それらが他のノードに接続する会議の一部でないならば、呼はまさにポイント ポイントの呼として扱われる。発側ノードが既に実行中の会議を保有しているならば、このノードは単に存在している会議に新しいノードを招待するのが一般的である。

かわりに、着側ノードが実行中の会議を保有している場合、会議がロックされているかロックされていないか、リスト記載されているか非リスト記載か、またパスワード保護されているかどうかには動作は依存する。会議が非リスト記載であると、事前情報なしには発側ノードはその存在を知ることができず、そのノードに会議が存在しないかのように呼を取り扱う(非リスト記載の会議が存在するという事前情報を保有しているならば、その会議には加入できる)。会議がロックされているならば、発側ノードが会議に加入する方法はなく、ここでも、その会議が存在しないかのように呼を

扱う。会議がパスワード保護されているならば、ユーザにパスワードを促し会議に参加させようと試みるオプションがあるが、ここでも、ノードに会議が存在しないかのように、発側ノードが呼を扱うことがありえる。これらのいずれの場合においても、一旦、新しい会議が確立されると、着側ノードは前の会議に接続していたノードを、新しい会議に移動させることを選択することができる。着側ノードに一つの会議しか存在せず、それが"オープン"(ロックされていない、非リスト記載でない、またはパスワード保護されていない)ならば、発側ノードは単にこの会議に加わろうと試みる。発側ノードに1つ以上の"オープン"会議が存在するならば、一般的に、ユーザに参加する会議を明確に決定することを要求する。この場合、手順は完全に自動ではない。

両ノードに会議が存在する場合(着側ノードの"オープン"会議を仮定する)、発側ノードは通常、それ自身の会議に着側ノードを招待するか、着側ノードの会議に加わろうと試みる。どちらの場合でも、相手のマルチポート端末を加えることにより拡大されなかった、既存の会議を持つマルチポート端末は、全ての下位ノードを新しく加入する(または招待する)会議に移動させるべきである。

これらの場合の新しい会議へのノード移動は2つの方法の内の1つにより行われる。GCC-CONFERENCE-TRANSFER プリミティブは各ノードに現在の会議から抜けて新しい会議に加わることを命令するために用いることができる。もちろん、これは新しい会議がロックされておらず、かつ、要求するノードが前の会議に対する移動(Transfer)特権を保有している場合(もしくは、前の会議の召集者である場合)にのみ、行うことができる。別の手段として、GCC-CONFERENCE-INVITE プリミティブを、直接接続されたノードを新しい会議に招待するために使うことができる。一般的に、これは、前の会議からこれらのノードを切断した後に行われる。直接接続されているノードより多くのノードが最初から存在するならば、再帰的に、直接接続したノードを新しい会議に招待することは、直接接続しているノードの責任である。すなわち、招待はノードの物理コネクションの階層に沿って広がっていく。一般的にノードがこれを実行するが、この動作が実行される保証はない。この理由により、実行可能な場合は GCC-CONFERENCE-TRANSFER を使う方がより安全である。

7.2. 会議名簿

会議名簿は、会議に参加しているノードに、他のどんなノードが同じ会議に参加しているかを知ることが許可し、各ノードについての情報を提供する。(会議を生成する、会議に参加する、招待されるのいずれかによって)会議に参加した後直ちに、GCC-CONFERENCE-ANNOUNCE-PRESENCE-要求を発行することにより、そのノードのノードコントローラは会議にその存在を通知する。この要求の結果、会議中のノードの GCC プロバイダは、新しく加わったノードを含めるため会議名簿を更新するのに必要な情報を交換する。GCC-CONFERENCE-ROSTER-REPORT-指示を生成することにより、更新された会議名簿は会議中の全ノードに分配される。それ自身が含まれる会議名簿を受信するまでは、会議に参加しようとしているノードは、その会議の活性メンバーであるとは考えられない。

会議名簿の情報は、ノードコントローラにより会議中はいつでも変更することができる。これは、GCC-CONFERENCE-ANNOUNCE-PRESENCE-要求を再発行することにより行われる。これは、別の GCC-CONFERENCE-ROSTER-REPORT-指示を生成することにより、全てのノードに更新された名簿が分配されるという結果となる。

ノードが何らかの理由により会議を抜けるときにも、GCC-CONFERENCE-ROSTER-REPORT-指示を生成するこ

とにより、GCC によって新しい会議名簿が分配される。

7.2.1. サービス概要記述

下記は、この章で定義されたプリミティブのリストと、それぞれの機能の簡単な要約である。

- GCC-CONFERENCE-ANNOUNCE-PRESENCE: ノードコントローラがノードの存在を会議に通知するために用いる。会議に加入した、あるいは、加えられた直後に、このプリミティブの使用が要求される。
- GCC-CONFERENCE-ROSTER-REPORT: 会議に入るかまたは抜けるノードのための会議名簿の変更の応答として、GCC によって生成される。
- GCC-CONFERENCE-ROSTER-INQUIRE: 会議中を通していつでも現在の会議名簿を検索するためにノードコントローラかアプリケーション プロトコル エンティティによって用いられる。

7.2.1.1. GCC-CONFERENCE-ANNOUNCE-PRESENCE

ノードがいずれかの会議に加わった後直ちに、GCC-CONFERENCE-ANNOUNCE-PRESENCE-要求というプリミティブを使うことにより、会議中の他の全ノードにその存在を通知する。ノードコントローラは、プリミティブを発行する責任がある。このプリミティブは、含まれる情報が変わったならば、会議中いつでも再発行できる。表 7-17/JT-T124 はパラメータとこのプリミティブの型を示す。図 7-15/JT-T124 は、このプリミティブが使われるときのイベントシーケンスを示す。

注- 会議名簿は会議に現在加入している全ての端末や MCU のリストを含んでいる。しかし、名簿をユーザへ表示する場合や、会議へ出入りするノードをユーザに指示する場合は、端末とマルチポート端末を含むだけにすることも可能。このプリミティブのノードタイプ(Node Type)というパラメータは、ユーザへの表示から意味のないノードタイプを除くため、これらのノードタイプの区別をするのに用いられる。さらに、ノードプロパティ(Node Properties)というパラメータもまた、ノードが管理用の装置かどうかや、他のノードの付随的な周辺装置であるかどうかを指定し、装置の用途についての情報を加える。また、これらの特性の両方は、システムの会議参加者表示の中に特定のノードを加えるかどうかを判断するためにも用いられる。一般的に、管理装置も周辺装置もそのようなリストには含まれない。

表7-17/JT-T124 GCC-CONFERENCE-ANNOUNCE-PRESENCE プリミティブの型とパラメータ

(ITU-TT.124)

パラメータ	要求	確認
会議 ID(Conference ID)	M	M(=RQ)
ノードタイプ(Node Type)	M	
ノードプロパティ(Node Properties)	M	
ノード名(Node Name)	O	
参加者名(Participant Name(s))	O	
サイト情報(Site Information)	O	
ネットワークアドレス(Network Address)	O	
代替ノード ID(Alternative Node ID)	O	
ユーザデータ(User Data)	O	
結果(Result)		M

会議 ID(Conference ID):プリミティブを適用する会議の識別子。

ノードタイプ(Node type):ノードタイプは端末か MCU かマルチポート端末のいずれかである。

ノードプロパティ(Node Properties):2 つの独立したフラグを形成する。1 つはノードが管理装置(例えば、予約システム)であるかどうかを示し、もう 1 つはノードが他のノードの付随的周辺装置であるかどうかを示す。

ノード名(Node Name):このノードの名前(例えば、"London")を含むユニコード文字列。最大 255 文字。

参加者名(Participant Name(s)):そのノードに集合している参加者の名前をそれぞれ含むユニコード文字列のリスト。それぞれの文字列に対し最大 255 文字。

サイト情報(Site Information):ノードに関するその他の情報を含むユニコード文字列。例えば、その場所の電話や FAX 番号のようなものを示すために使われる。

ネットワークアドレス(Network Address):このノードのネットワークアドレスを示すためのオプションパラメータ。このパラメータは、実際のネットワークアドレスに続き、ネットワークタイプの情報を示すサブ領域を含む。このパラメータは、もし後で、GCC-CONFERENCE-ADD プリミティブを介して、このノードを別の会議に追加しようとすることがあるならば、そのときに使用される。これは、MCU が、追加されるノードに対する物理コネクションがすでに存在するかどうかを判断するときに、追加されるノードのネットワークアドレス(Network Address)と比較するためのネットワークアドレス情報となる。この情報が含まれないならば、同じ物理コネクション上で、後でこのノードを別の会議に追加することは不可能である。また、ポイント ポイントの会議にさらにほかのノードを追加するために、MCU を経由するように自動的に再ルーティングしようとする場合にも、このパラメータは使用される。このパラメータによって、MCU を通した GCC-CONFERENCE-ADD プリミティブを使用して元々の 2 つのノード同士を再接続することができる。

代替ノード ID(Alternative Node ID):この領域は、通知ノード(及び、それに相当するノード ID)と、その他の目的のために定義される代替ノード ID を関連付けするために使われる。この代替ノード ID は、ノード ID と同じ範囲の数字からではなく、むしろこの標準によって指定されていない別の番号付け機構による ID を表すことを意図する。例えば、ISDN の場合、TTC 標準 JT-H243 をサポートしているノードに対し代替ノード ID は 2 オクテット長の H.243 サイト ID(H.243 siteID)とすることができる。

ユーザデータ(User Data):この標準の範囲外の機能に使われるオプションのユーザデータ。このデータは会議名簿の一部としてトップ GCC プロバイダに蓄えられることに注意を要する。したがって、このパラメータは大量の情報を含むようには意図されていない。そのようにすることは、会議から不本意に退去させられるリスクを伴う。

結果 (result):要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。

それは、以下のリスト中の可能な結果の 1 つを含む。

「成功(successful)」、「無効な会議(invalid conference)」

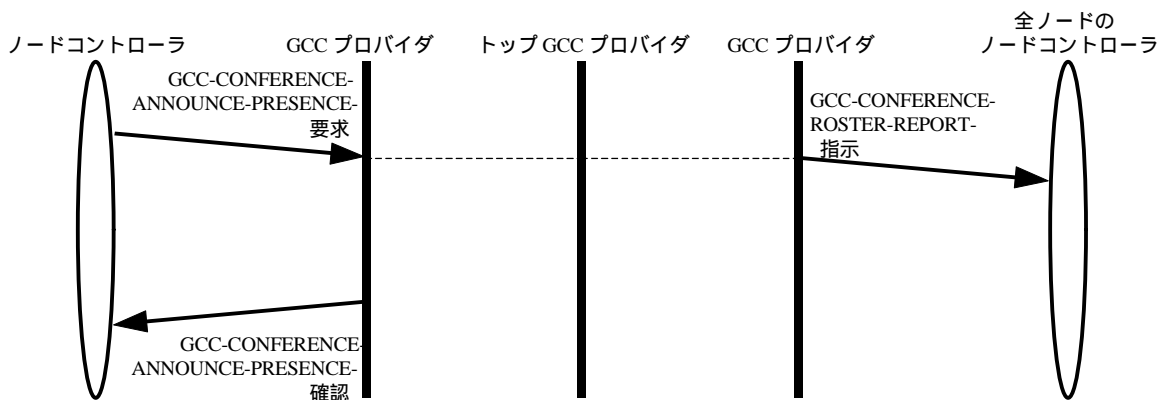


図7-15/JT-T124 GCC-CONFERENCE-ANNOUNCE-PRESENCE プリミティブのシーケンス (ITU-TT.124)

7.2.1.2. GCC-CONFERENCE-ROSTER-REPORT

何等かの理由(会議に新しいノードが加わる、ノードが会議から退出する、名簿中の情報の更新)により、会議名簿が変更される時はいつでも、GCC が各ノードのノードコントローラに GCC-CONFERENCE-ROSTER-REPORT プリミティブを発行することにより、名簿は全ノードに分配される。表 7-18/JT-T124 は、このプリミティブの型とパラメータを示す。図 7-16/JT-T124 はこのプリミティブが使われるときのイベントシーケンスを示す。

表7-18/JT-T124 GCC-CONFERENCE-ROSTER-REPORT プリミティブの型とパラメータ (ITU-TT.124)

パラメータ	指示
会議 ID(Conference ID)	M
会議名簿(Conference Roster)	M

会議 ID(Conference ID):プリミティブを適用する会議の識別子。

会議名簿(Conference Roster):指定された会議に加入している各ノードのリスト。会議名簿(Conference Roster)のフォーマットは表 7-19/JT-T124 に示される。

表 7-19/JT-T124 会議名簿(Conference Roster)パラメータの内容 (ITU-TT.124)

パラメータ	記述
会議ノードのリスト (List of Conference Nodes)	各ノードについての情報と、会議に加入しているノードのリスト。このリスト中の各項目の内容を表 7-20/JT-T124 に示す。
インスタンス番号 (Instance Number)	会議名簿に対するインスタンス番号。会議名簿の内容が変更される度に、モジュロ 2^{16} で増加する 16bit 数である。
ノード追加フラグ (Nodes Added Flag)	最後のインスタンスから 1 つ以上のノードが会議名簿に加わったかどうかを示すフラグ。このフラグはノード削除フラグ(Node Removed Flag)と互いに排他的ではない。

ノード削除フラグ (Nodes Removed Flag)	最後のインスタンスから 1 つ以上のノードが会議名簿から抜けたかどうかを示すフラグ。このフラグはノード追加フラグ(Node Added Flag)と互いに排他的ではない。
----------------------------------	---

表 7-20/JT-T124 会議ノードのリスト(List of Conference Nodes)中の各エントリの内容
(ITU-T T.124)

パラメータ	記述
ノード ID(Node ID)	このノードの GCC プロバイダの MCS ユーザ ID。
上位ノードのノード ID (Node ID of Superior Node) (条件付き)	もし存在するならば、コネクション階層上でこのノードの直接上位のノードの GCC プロバイダの MCS ユーザ ID。(トップ GCC プロバイダには存在しない)
ノードタイプ(Node Type)	ノードタイプは、端末か MCU かマルチポート端末かである。
ノードプロパティ (Node Properties)	2 つの独立したフラグを形成する。1 つはノードが管理装置(例えば、予約システム)であるかどうかを示し、もう 1 つはノードが他のノードの付随的周辺装置であるかどうかを示す。
ノード名(Node Name) (条件付き)	このノードの名前(例えば、"London")を含むユニコード文字列。最大 255 文字。
参加者名(Participant Names) (条件付き)	そのノードに集合している参加者の名前をそれぞれ含むユニコード文字列のリスト。それぞれの文字列に対し最大 255 文字。
サイト情報(Site Information) (条件付き)	ノードに関するその他の情報を含むユニコード文字列。例えば、その場所の電話や FAX の番号のようなものを示すために使われる。
ネットワークアドレス (Network Address) (条件付き)	このノードのネットワークアドレス。このパラメータは、実際のネットワークアドレスに続き、ネットワークの型情報を示すサブ領域を含む。
代替ノード ID (Alternative Node ID) (条件付き)	この領域は、通知ノード(及び、それに相当するノード ID)と、他の目的のために定義される代替ノード ID を関連付けするために使われる。この代替ノード ID は、ノード ID と同じ範囲の数字からではなく、むしろこの標準によって指定されていない別の番号付け機構による ID を表すことを意図する。例えば、ISDN の場合、TTC 標準 JT-H243 をサポートしているノードに対し代替ノード ID は 2 オクテット長の H.243 サイト ID(H.243 siteID)とすることができる。
ユーザデータ (User Data) (条件付き)	この標準の範囲外の機能に使われるオプションのユーザデータ。



図 7-16/JT-T124 GCC-CONFERENCE-ROSTER-REPORT プリミティブのシーケンス (ITU-T T.124)

7.2.1.3. GCC-CONFERENCE-ROSTER-INQUIRE

GCC-CONFERENCE-ROSTER-INQUIRE-要求プリミティブは、指定された会議に対する完全な会議名簿を返す。このプリミティブはノードコントローラと同様にアプリケーション プロトコル エンティティからも使用でき、それぞれが個別にローカルな GCC プロバイダから会議名簿を得ることを可能とする。

表 7-21/JT-T124 はこのプリミティブの型とパラメータを示す。図 7-17/JT-T124 はこのプリミティブのイベントシーケンスを示す。

表 7-21/JT-T124 GCC-CONFERENCE-ROSTER-INQUIRE プリミティブの型とパラメータ (ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M(=)
会議名(Conference Name)		M
会議記述(Conference Description)		C
会議名簿(Conference Roster)		M
結果(Result)		M

会議名(Conference Name):会議プロファイルに含まれる会議名(Conference Name)。

会議記述(Conference Description):会議プロファイルに含まれる会議記述(Conference Description)。会議が生成されたときに会議記述(Conference Description)を定義した場合のみ、このパラメータは存在する。

会議 ID(Conference ID):プリミティブを適用する会議の識別子。

会議名簿(Conference Roster):指定された会議に加入している各ノードのリスト。会議名簿(Conference Roster)のフォーマットは表 7-22/JT-T124 に示される。

表 7-22/JT-T124 会議名簿(Conference Roster)パラメータの内容

(ITU-T T.124)

パラメータ	記述
会議ノードのリスト (List of Conference Nodes)	各ノードについての情報と、会議に加入しているノードのリスト。このリスト中の各項目の内容は表 7-20/JT-T124 に示されている。
インスタンス番号 (Instance Number)	会議名簿に対するインスタンス番号。会議名簿の内容が変更される度に、モジュロ 2^{16} で増加する 16bit 数である。

結果 (result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。

それは、以下のリスト中の可能な結果の 1 つを含む。

「成功(successful)」、「無効な会議(invalid conference)」

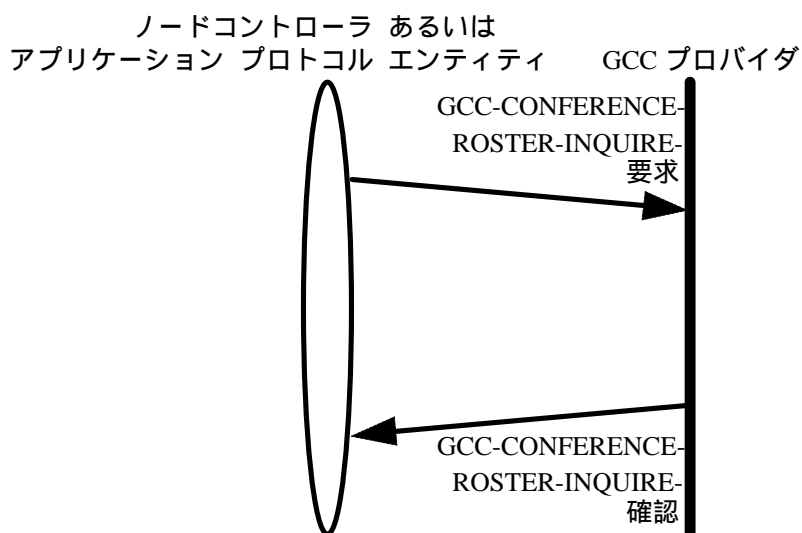


図 7-17/JT-T124 GCC-CONFERENCE-ROSTER-INQUIRE プリミティブのシーケンス

(ITU-T T.124)

7.3. アプリケーション名簿

アプリケーション名簿は会議に参加しているノードにどのアプリケーション プロトコル エンティティが有効か教え、また同位アプリケーション プロトコル エンティティ間で直接通信を開始するために十分な情報を供給する。名簿は標準のアプリケーション プロトコルあるいは、非標準のアプリケーションプロトコルに基づいたアプリケーション プロトコル エンティティを含む。

7.3.1. アプリケーション名簿の内容

各会議のために、GCC プロバイダは各アプリケーション プロトコル エンティティに付随する情報リストを編集する。これらのリストは、ローカル アプリケーション名簿を形成し、それは、アプリケーション名簿交換手順の一部として

他のノードと交換される。この交換の結果として、会議アプリケーション名簿として知られている会議全体におけるすべてのアプリケーション プロトコル エンティティに付随する情報リストとなる。

ローカル アプリケーション名簿は、各 GCC プロバイダに管理され、そのローカル データベースに保管される。会議セットアップの適当な時期に、ノードはローカル アプリケーション名簿をすべての会議ノードと交換し、GCC プロバイダはノード コントローラを含む、すべてのアプリケーション プロトコル エンティティを示す会議アプリケーション名簿を受け取る。この交換は GCC 内部で行われ、アプリケーション プロトコル エンティティには生成されるプリミティブを介してのみ明確となる。

ローカル/会議アプリケーション名簿は、以下のコンポーネントからなる。

- ・ローカル アプリケーション名簿

ローカルの GCC プロバイダに登録された各アプリケーション プロトコル エンティティのセッション キー (Session Key)、アプリケーション レコード (Application Record)、およびアプリケーション能力リスト (Application Capabilities List) 等から構成される。

- ・会議アプリケーション名簿

各アプリケーション プロトコル セッションの、そのセッションのセッション キー (Session Key)、セッションのネゴシエーションされた能力セットを含むアプリケーション能力リスト (Application Capabilities List)、セッションに登録されたアプリケーション プロトコル エンティティのリスト (アプリケーション プロトコル エンティティが登録されたノードのノード ID (Node ID)、ローカルノード内で、あるアプリケーション プロトコル エンティティを識別する識別子、アプリケーション レコードを含む。) 等から構成される。

同位アプリケーション プロトコル エンティティのセット (単一のアプリケーション プロトコル セッション中) はセッション キーを同じ値で登録することにより定義される。セッション キーは以下のように定義される。

- ・セッション キー (Session Key)

アプリケーション プロトコル セッションをユニークに識別するための識別子。セッション キーは二つのコンポーネントから構成される。一つはアプリケーション プロトコルを識別するコンポーネント。もう一つはオプションで、そのアプリケーション プロトコルの、個別のセッションを識別する。

セッション キー (Session Key) の一つ目のコンポーネントであるアプリケーション プロトコル キーは、標準および非標準のアプリケーション プロトコル仕様を識別する。キーは標準あるいは非標準のいずれかのアプリケーション プロトコルを重複することなくユニークに定義されるように構成される。キーは以下の二つの型の内の一つにより特定される。一つ目の型のキーは ASN.1 型オブジェクト識別子により特定される。この型は標準のアプリケーション プロトコルを定義するために使用される。標準あるいは勧告されたオブジェクト識別子の割り当ては、ITU-T 勧告 X.680 ANNEX B と C に詳述されている。オブジェクト識別子は ISO や ITU によって直接あるいは間接的に承認された、国もしくはプライベートな管理機関の定める非標準のアプリケーション プロトコルの特定にも使用される。

2つ目の型のキーはASN.1型オクテット スtringで表される TTC 標準 JT-H221 に採用されている方法と同様の方法で解釈され非標準のコマンドや能力を表す。この場合、最初の2オクテットが国別コードを、つぎの2オクテットがメーカー別コードを示す。国別コードの最初のオクテットはITU-T 勧告 T.35 に従って割り当てられ次のオクテットとメーカー別コードは各国で割り当てられる。これ以降のオクテットは各メーカーや国別に自由に選択される。

セッション キー (Session Key) のもう一つのコンポーネントはオプションのセッション ID である。セッション ID はアプリケーション プロトコル セッションのユニークな識別子として使用される、その (アプリケーション プロトコル仕様によって判断される) セッションに参加しているアプリケーション プロトコル エンティティによって通信チャネルとして使用される MCS チャネル ID である。ユニークさを保証するため、このチャネル ID はセッション期間中は保持する。活性状態のアプリケーション プロトコル エンティティのセッション キーでセッション ID のない場合は特別のセッションつまりデフォルト セッションとみなされる。

非活性状態のアプリケーション プロトコル セッションの場合、セッション ID は必要でない。この場合、指示されているアプリケーション プロトコルと、一つもしくはそれ以上のアプリケーション プロトコル エンティティをどれかのセッションの一部として呼び出す能力をサポートしていることと解釈される。

各アプリケーション レコードは以下のパラメータを持つ (一部はオプション)

・活性/非活性フラグ

アプリケーション プロトコル エンティティが登録されたが、しかしデータ受信がまだ出来ない (例: まだ適当なチャネルに加わっていない) 状態。この状態を他のノードに示すため、アプリケーション プロトコル エンティティはこのフラグを非活性状態に設定して登録する。これはアプリケーション プロトコル エンティティが GCC サービスを使用できるようにするために行われるか、他のノードのアプリケーション プロトコル エンティティが存在することが認知される以前に、起動することなしに、このアプリケーション プロトコル エンティティの存在を知らせることを可能にする。起動されていないアプリケーション プロトコル エンティティが起動される時、このフラグを活性状態に設定して再登録する。このフラグはアプリケーション プロトコル エンティティにより随時再登録によって変更される。

・アプリケーション ユーザ ID

セッション キー (Session Key) に付随する MCS ユーザ ID。そのアプリケーション プロトコル セッションのためにプライベート チャネルで会議する時に、参加者として別のノードを加えることができるノードのことを意味する。このパラメータは非活性状態として登録されたアプリケーション プロトコル エンティティ用にオプションであるが、活性状態のアプリケーション プロトコル エンティティには必須である。アプリケーション プロトコル エンティティがまだ MCS ドメインにアタッチしていないために含まれていない場合、後でアプリケーション プロトコル エンティティが再登録によってアタッチする時に付加される。

・議長動作能力

このフラグは、(もし、そのアプリケーション プロトコル仕様が議長エンティティの行うべき手続きを定めているならば)、このアプリケーション プロトコル エンティティが議長のアプリケーション プロトコル エンティティとして動作する能力があるかどうかを表す。各ノードの GCC プロバイダは、交換するアプリケーション名簿中で、このフラ

グが設定される同位アプリケーション プロトコル エンティティがアプリケーション プロトコル セッション当たり一つ以下になるようにしなければならない。あるノードが会議議長になった場合、指名されたアプリケーション プロトコル エンティティが(もしあれば)、各アプリケーション プロトコル セッションの議長指名アプリケーション プロトコル エンティティとなる。このフラグは、活性状態のアプリケーション プロトコル エンティティにのみ適用する。登録時に活性/非活性フラグが非活性状態に設定されている場合、GCC プロバイダは、議長動作能力フラグを無視し、FALSE であるとみなす。

・非集約型能力リスト

これはオプションのパラメータでアプリケーション プロトコル エンティティがその能力(標準か非標準か)をリストするためにある。アプリケーション能力リストに示された能力をある規則により格納するというよりはむしろ各アプリケーション プロトコル エンティティのアプリケーション レコードの一部として名簿中に保持される。

・アプリケーション能力リスト

これはオプションのパラメータ リストであり、アプリケーション プロトコル エンティティの能力をリストするためのものである。その能力はアプリケーション プロトコルに特異なものであるが、アプリケーション プロトコル セッションの共通能力セットを判断する情報を GCC によりリストされ、このセットの同位アプリケーション プロトコル エンティティに通知する。これは、すべてのノード間で、全能力リストを交換することを省く。リスト中の各能力は、クラス特定子が付加される。クラス特定子は、共通能力セットに適用される規則を決定する。

表 7-23/JT-T124 能力の組み合わせルール
(ITU-T T.124)

クラス (Class)	説明
Logical	会議中のいずれかの同位アプリケーション エンティティがこの能力の使用を示している場合、最後の能力リストはこの能力を持つことを指示している同位アプリケーションエンティティの数を示す。
Unsigned-MIN	パラメータ オクテットは一つの符号なし整数 (unsigned integer) として扱われる。この能力を示す同位アプリケーション プロトコル エンティティの数と、この能力を指示する全ての同位アプリケーション プロトコル エンティティの中での最小値が最後の能力リストに示される。
Unsigned-MAX	パラメータ オクテットは一つの符号なし整数 (unsigned integer) として扱われる。この能力を示す同位アプリケーション プロトコル エンティティの数と、この能力を指示する全ての同位アプリケーション プロトコル エンティティの中での最大値が最後の能力リストに示される。

これら基本的な分類に加えて能力をネストすることも可能である。ネストを明確に定義はしないが、3つの能力クラスの適切な解釈によって行われる。例えば、論理値 (Logical) 能力 X に依存するアプリケーション プロトコル特有の能力 Y を定義したとする。能力 X がある場合のみ能力 Y を利用できるとアプリケーション プロトコル仕様内で適用され

るとすると、最終的な能力セット中の Y の値が論理値 (Logical) 能力の場合は、能力 X を持つものの中で能力 Y を持つ同位アプリケーション プロトコル エンティティの総数として解釈でき、Y が数値 (Numerical) 能力の場合、能力 X を持つ同位アプリケーション プロトコル エンティティの間の能力 Y の最小値もしくは最大値と解釈できる。Y が数値 (Numerical) 能力の場合、もし能力 Y を示す同位アプリケーション プロトコル エンティティの数が能力 X を示す同位アプリケーション プロトコル エンティティの数より少なかったならば、いくつかの同位アプリケーション プロトコル エンティティがそのデフォルト値を超えた Y の値をサポートしていないことを示している。この場合、ネゴシエーションされた Y 値は無視されるであろう。ネゴシエーションされた Y 値は、能力 X と能力 Y を持つ同位アプリケーション プロトコル エンティティの数が等しい場合に限り有効である。

7.3.2. アプリケーション名簿交換手順

各アプリケーション プロトコル エンティティは GCC プロバイダと通信する GCCSAP を通してその存在をローカルの GCC プロバイダに知らせる。GCCSAP の生成と管理は個別の事柄であって、この標準の範囲外である。

アプリケーション プロトコル エンティティは GCC-APPLICATION-PERMISSION-TO-ENROLL によりそのノードが加わった会議の存在を知らされる。一旦アプリケーション プロトコル エンティティが会議の存在を知らされると、GCC-APPLICATION-ENROLL-要求を GCC プロバイダに対し発行する。この要求プリミティブはアプリケーション プロトコル エンティティが会議に登録したことを示す (登録/登録解除フラグ - 設定 登録) か、または会議に登録していないことを示す (登録/登録解除フラグ - 設定 登録解除)。後者の場合、アプリケーション レコード パラメータはそのプリミティブのなかに含まれる必要はない。それ以降、登録許可が取り消されるまで (許可/取り消しフラグが取り消しに設定された GCC-APPLICATION-PERMISSION-TO-ENROLL プリミティブを受信するまで)、最初から会議に登録することを選ばなかったアプリケーション プロトコル エンティティは GCC-APPLICATION-ENROLL-要求を自ら発行することにより登録する。

GCC-APPLICATION-ENROLL-要求にはいくつかのパラメータがあり、使用時にはそれを発行する前に踏まなければならない段階がある。一つ目が、活性状態に設定された活性/非活性フラグとともに登録要求する場合に必須であるアプリケーション ユーザ ID である。アプリケーション ユーザ ID を得るためには、アプリケーション プロトコル エンティティは GCC-APPLICATION-PERMISSION-TO-ENROLL-指示の中に示されている MCS ドメインに最初にアタッチしなければならない。これは MCS プリミティブ MCS-ATTACH-USER-要求を発行することによって行われる。MCS-ATTACH-USER-確認の受信により、アプリケーション プロトコル エンティティはアプリケーション ユーザ ID が割り当てられる。この ID は GCC-APPLICATION-ENROLL-要求に含まれているかも知れない。その調停の間にドメインへのコネクションを失うことによりドメインへのアタッチに失敗した場合、アプリケーション プロトコル エンティティはそのノードがもはや会議の一部でなくなったものと仮定し、別の GCC-APPLICATION-PERMISSION-TO-ENROLL-指示を受信するまで待つ。ノードが会議から切り離される時、アプリケーション プロトコル エンティティはそのアタッチはもはや有効でないことを MCS から直接通知される。アプリケーション プロトコル エンティティはそのノードがもはや会議の一部ではなくなったものと仮定し、別の GCC-APPLICATION-PERMISSION-TO-ENROLL-指示を受信するまで待つ。GCC-APPLICATION-PERMISSION-TO-ENROLL プリミティブによって登録許可を取り消された場合アプリケーション プロトコル エンティティはそのドメインへのアタッチを試みてはならない。

二つ目のパラメータはオプションでセッションキーの一部であるセッション ID である。セッション ID は MCS チャンネル ID (ユーザ ID ではない) から作成される。起動されたセッションへ登録するとき、セッション ID は GCC-APPLICATION-INVOKE-指示から得られる。セッションへの参加を起動している最中は、セッション ID は最後に受信したアプリケーション名簿から得る。新しいセッションを生成する場合、新しいチャンネル ID は MCS-CHANNEL-JOIN プリミティブもしくは MCS-CHANNEL-CONVENE プリミティブを用いて割り当てられる。

GCC-APPLICATION-ENROLL-要求 プリミティブに従って、GCC プロバイダは新しいエントリをそのローカルのアプリケーション名簿中に生成する。アプリケーション プロトコル エンティティ用のエントリがすでに存在する場合、ローカルアプリケーション名簿エントリの内容は新しい情報を含むため変更されるか、もしくは登録解除/登録フラグが設定されている場合には、そのエントリは名簿から削除される。

会議の開始時点で、GCC プロバイダは GCC-APPLICATION-PERMISSION-TO-ENROLL プリミティブを発行後、GCCSAP を通して GCC プロバイダへ接続されたアプリケーション プロトコル エンティティが登録か登録しない旨の明示的な決定を示す GCC-APPLICATION-ENROLL プリミティブが発行されるのを待つ。一旦すべてのアプリケーション プロトコル エンティティが応答したら、GCC プロバイダは会議の他のノードとローカル アプリケーション名簿を交換する。会議が進行中、ローカルアプリケーション名簿の変更は、即座に他のノードと交換される。

どちらの場合においても、新しい名簿の交換が発生したときは、どのノードから生成されたものでも、新しい会議アプリケーション名簿が生成されすべてのノードへ配られる。GCC プロバイダは全体の名簿をノードコントローラへ発行し、名簿の一部を GCC-APPLICATION-ROSTER-REPORT-指示 プリミティブを使用している登録されたアプリケーション プロトコル エンティティへ発行する。アプリケーション プロトコル エンティティへの、名簿報告は会議アプリケーション名簿の特にそのアプリケーション プロトコル セッションに適用される部分を含む(同様に他の部分も含むこともできる)。これは、各セッションの、リスト中に存在する。リスト中の各エントリはノード ID とアプリケーション プロトコル エンティティ ID を含み、それらはアプリケーション プロトコル エンティティを特定し、そしてまたそのアプリケーション プロトコル エンティティ用のアプリケーション レコードも含まれる。またアプリケーション プロトコル セッション用のアプリケーション能力リストも含まれる。これは、すべての同位アプリケーション プロトコル エンティティによって公開された能力を制御する能力クラスのアプリケーションに基づいた、このアプリケーション プロトコル セッションのための共通能力のリストである。ノードコントローラの場合、GCC-APPLICATION-ROSTER-REPORT-指示は、会議アプリケーション名簿全体を含む。リスト中の各エントリ用に、それはノード識別子を含み、また、そのノードに登録されたアプリケーション プロトコル用のアプリケーション レコードも含む。また、各アプリケーション プロトコル 用に、アプリケーション能力リストを含む。

アプリケーション プロトコル エンティティはアプリケーション名簿に含まれることを示す GCC-APPLICATION-ROSTER-REPORT-指示を受信するまで会議の一部とは認められない。いつでも、先に登録されたアプリケーション プロトコル エンティティがもはや含まれていないことを示す GCC-APPLICATION-ROSTER-REPORT-指示を受信すると、そのアプリケーション プロトコル エンティティはもはや登録されていないとみなされる。アプリケーション プロトコル エンティティは GCC-APPLICATION-ENROLL-要求を発行することにより、再登録を試みるかも知れない。

いつでもアプリケーション プロトコル エンティティは登録解除に設定された登録/登録解除フラグを伴った GCC-

APPLICATION-ENROLL-要求 プリミティブを発行することによりアプリケーション名簿から自ら削除する。これはアプリケーション プロトコル エンティティ用のアプリケーション レコードをローカル アプリケーション名簿から削除する。これは会議のすべてのノードの会議アプリケーション名簿から削除するのと同様である。

会議中いつでも、アプリケーション プロトコル エンティティもしくはノード コントローラは、会議アプリケーション名簿の一部を要求するかも知れない。ある会議における各ノードのために、アプリケーション レコードで GCC-APPLICATION-ROSTER-INQUIRE-要求で与えられたセッションキーと一致するものは全て返される。NULL のキーを使用することで、会議アプリケーション名簿は完結する。

アプリケーション プロトコル エンティティもしくはノード コントローラは他のノードからアプリケーション プロトコル エンティティを起動することを試みる。この要求はオプションとして宛先ノードの特定リストを含む（もしくは宛先が指定されていない場合すべてのノード）。ノード コントローラはオプションとしてこの指示に対し、自身を会議に登録することになる同位アプリケーション プロトコル エンティティを起動することにより応答する。

7.3.3. サービスの概要記述

以下はこの節で定義されるプリミティブのリストと各機能の要約である。

・GCC-APPLICATION-PERMISSION-TO-ENROLL

GCC により生成され、ローカルノードが会議に参加する時はいつでも、GCC プロバイダにそのアプリケーション プロトコル エンティティの存在を知らせたアプリケーション プロトコル エンティティに対し発行される。これは、アプリケーション プロトコル エンティティが登録をすることを示す。ノードが会議にもはや加わらない時は、取り消しフラグを設定することで登録する権利を取り消す。

・GCC-APPLICATION -ENROLL

アプリケーション プロトコル エンティティによって使用され、会議における他の同位アプリケーション プロトコル エンティティとの通信を確立する。このプリミティブの使用は他のノードと交換されるアプリケーション名簿中にエントリを生成（もしくは変更、削除）する。

・GCC-APPLICATION-ROSTER-REPORT

アプリケーション プロトコル エンティティの登録/登録解除、ノードの会議への参加/退場によるアプリケーション名簿の変更に対する応答のために GCC により生成される。

・GCC-APPLICATION-ROSTER-INQUIRE

会議中に随時現在のアプリケーション名簿のすべてもしくは一部を取り出すためにノード コントローラもしくはアプリケーション プロトコル エンティティによって使用される。

・GCC-APPLICATION -INVOKE

アプリケーション プロトコルの特定のセッション用にアプリケーション プロトコル エンティティを起動するために

会議における他ノード セットへの信号用に使用される。

7.3.3.1. GCC-APPLICATION-PERMISSION-TO-ENROLL

GCC-APPLICATION-PERMISSION-TO-ENROLL-指示 プリミティブはローカルのアプリケーション プロトコル エンティティに対しローカル ノードが指定した会議に参加することや、アプリケーション プロトコル エンティティ自身がその会議に登録することを通知する。またこのプリミティブはローカル ノードがもはや会議に加わらない場合の許可の取り消しのためにも使用される。MCS に接続されたアプリケーション プロトコル エンティティは、そのノードにおいて会議が終了したとき、MCS からの通知を受信する。一方、接続もしくは登録していないアプリケーション プロトコル エンティティはその通知を受信する必要はない。このプリミティブはもはやその会議に登録しないか対応する MCS ドメインに接続しないことをアプリケーション プロトコル エンティティに示す。この指示はノードが会議への参加に成功した後にのみ、GCC プロバイダにそれらの存在を知らせたすべてのアプリケーション プロトコル エンティティに対して GCC プロバイダによって発行される。表 7-24/JT-T124 にこのプリミティブの型とパラメータを示す。図 7-18/JT-T124 にこのプリミティブが使用される時のイベントのシーケンスを示す。

表 7-24/JT-T124 GCC-APPLICATION-PERMISSION-TO-ENROLLプリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	指示
会議ID (Conference ID)	M
許可/取り消しフラグ (Grant/Revoke Flag)	M

会議 ID (Conference ID) : アプリケーション プロトコル エンティティが登録される会議の識別子。そのアプリケーション プロトコル エンティティがアタッチする MCS ドメインに等しい。

許可/取り消しフラグ (Grant/Revoke Flag) : このフラグはアプリケーション プロトコル エンティティが登録許可を与えられている、もしくは許可を取り消されているかを示す。

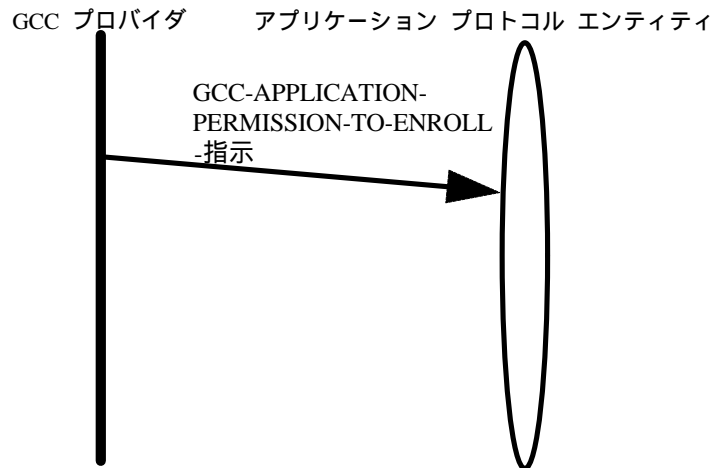


図7-18/JT-T124 GCC-APPLICATION-PERMISSION-TO-ENROLL プリミティブのシーケンス
(ITU-T T.124)

7.3.3.2. GCC-APPLICATION-ENROLL

GCC-APPLICATION-ENROLL-要求 プリミティブは、アプリケーション プロトコル エンティティが指定した会議に自分自身を登録するために発行される。このプリミティブは、GCC-PERMISSION-TO-ENROLL-指示に応答して発行されなければならない。この際、アプリケーション プロトコル エンティティがまだ登録したくない場合は、登録/登録解除フラグを登録解除に設定してもよい。登録の許可が取り消されるまでの間は、アプリケーション プロトコル エンティティは登録、再登録（自身のアプリケーション レコードのエントリを変更するため）、登録解除をすることができる。既に登録解除されているアプリケーション プロトコル エンティティが登録解除を発行した場合、要求の結果は「成功 (successful)」になる。しかし、そのアプリケーション プロトコル エンティティの状態は何も変更されない。アプリケーション プロトコル エンティティが登録されると、関連するパラメータからアプリケーション名簿交換中に会議に放送されるアプリケーション レコードが作られる。登録していないアプリケーション プロトコル エンティティは、ローカルまたはリモートの参加者から利用できず、GCC-APPLICATION-ROSTER-REPORT-指示を受信することもない。

このプリミティブの型とパラメータを表 7-25/JT-T124 に示す。

このプリミティブを使用したときのシーケンスを図 7-19/JT-T124 に示す。

表 7-25/JT-T124 GCC-APPLICATION-ENROLL プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M (=)
セッション キー (Session Key)	M	M (=)
活性/非活性 (Active or Inactive)	M	
アプリケーション ユーザ ID (Application User ID)	C	
議長動作能力 (Conducting Operation Capable)	O	
スタートアップ チャネル (Start-up Channel)	O	
非集約型能力リスト (Non-Collapsing Capabilities List)	O	
アプリケーション能力リスト (Application Capabilities List)	O	
登録/登録解除 (Enroll or Un-enroll)	M	
アプリケーション プロトコル エンティティ ID (Application Protocol Entity ID)		C
ノード ID (Node ID)		C
結果 (Result)		M

会議 ID (Conference ID) : プリミティブを適用する会議の識別子。

セッション キー (Session Key) : アプリケーション プロトコル セッションを特定するユニークな識別子。アプリケーション プロトコルの識別子に相当する。セッション ID (Session ID) を任意に含めることができる。

活性/非活性 (Active or Inactive) : このフラグは、登録しようとするアプリケーション プロトコル エンティティが、完全に活性状態(そのアプリケーション プロトコル仕様によって課せられる仕事が処理できる状態)であるか、非活性状態であるかを表す。ある GCC サービスを利用するために登録しなければならないが完全に動作可能になっていない場合(例えば、動的チャネルには未加入だが、まず最初にレジストリをチェックして加入すべきチャネルを決める必要がある場合など)、アプリケーション プロトコル エンティティは非活性状態であることを自分自身で表明する可能性がある。あるいは、そのアプリケーション プロトコルをサポートしていることを公表するために非活性状態で登録し、同様のアプリケーション プロトコル エンティティが他のノード上に登録されたことがわかるまでエンティティを活性化させずにおく、という使い方も考えられる。このフラグの具体的な意味は、各アプリケーション プロトコルの仕様で定義される。

アプリケーション ユーザ ID (Application User ID) : このアプリケーション プロトコル エンティティに与えられた MCS ユーザ ID。このパラメータは、非活性状態のアプリケーション プロトコル エンティティではオプシ

オンであるが活性状態のアプリケーション プロトコル エンティティでは必須である。ある MCS ユーザ ID で登録済みのアプリケーション プロトコル エンティティの再登録時に異なる MCS ユーザ ID を指定するとエラーになる。このパラメータは登録解除の時は不要である。

議長動作能力 (Conducting Operation Capable) : このフラグは、(もし、そのアプリケーション プロトコル仕様が議長エンティティの行うべき手続きを定めているならば)、このアプリケーション プロトコル エンティティが議長のアプリケーション プロトコル エンティティとして動作する能力があるかどうかを表す。各ノードの GCC プロバイダは、交換するアプリケーション名簿中で、このフラグが設定される同位アプリケーション プロトコル エンティティがアプリケーション プロトコル セッション当たり一つ以下になるようにしなければならない。あるノードが会議議長になった場合、指名されたアプリケーション プロトコル エンティティが(もしあれば)、各アプリケーション プロトコル セッションの議長指名アプリケーション プロトコル エンティティとなる。登録しようとしているアプリケーション プロトコル エンティティは、自分のエントリにこのフラグが設定されたアプリケーション名簿を受信するまでは、自分自身に議長の役割が指名されたときのみ適用する。このフラグは、活性状態のアプリケーション プロトコル エンティティにのみ適用する。登録時に活性/非活性フラグが非活性に設定されている場合、GCC プロバイダは、議長動作能力フラグを無視し、FALSE であるとみなす。

スタートアップ チャンネル (Start-up Channel) : オプションパラメータであり、その値は、「静的 (Static)」、「動的マルチキャスト (Dynamic Multicast)」、「動的プライベート (Dynamic Private)」、「動的ユーザ ID (Dynamic UserId)」のいずれかである。このパラメータは、アプリケーション プロトコル エンティティが一連の起動動作で使用する MCS チャンネルのタイプを指定する。厳密なこのパラメータの実装と使用上の要件はアプリケーション プロトコル固有の仕様による。特定のアプリケーション プロトコルでは、これらのチャンネルタイプの中のいくつかが無効でないということも有り得る。

非集約型能力リスト (Non-Collapsing Capabilities List) : オプションパラメータであり、アプリケーション能力リストのようにある規則に従って集約されることなく、アプリケーション名簿中でアプリケーション レコードの一部として保存される、各アプリケーション プロトコル エンティティの(標準か非標準の)能力リストである。このリストの各エントリは、(標準か非標準の)能力 ID と、アプリケーション プロトコル固有のデータフィールドを含む。

アプリケーション能力リスト (Application Capabilities List) : オプションの、アプリケーション プロトコル固有の能力のリストである。各能力には、GCC がこのアプリケーション プロトコル セッションにおける共通能力を確定する(集約する)ときに適用される規則を識別するためのクラス識別子が付けられる。

登録/登録解除 (Enroll or Un-enroll) : このフラグは、そのアプリケーション プロトコル エンティティが、新規追加あるいは自分のアプリケーション レコードの変更(登録)を希望しているのか、自分のアプリケーション レコードの削除(登録解除)を希望しているのかを表す。アプリケーション プロトコル エンティティは、GCC-PERMISSION-TO-ENROLL-指示を受信したならば、GCC-APPLICATION-ENROLL-要求で応答しなければならない。そのとき、もしそのアプリケーション プロトコル エンティティが登録を希望していなければ、この

フラグを登録解除 (Un-enroll) に設定する。

アプリケーション プロトコル エンティティ ID (Application Protocol Entity ID) : 新規登録が成功した場合のみ存在する (既に登録済みのとき、登録が不成功のとき、登録解除のときには存在しない)。これは、ローカルノード内でアプリケーション プロトコル エンティティを一意に識別するための識別子である。この値は、アプリケーション名簿の各エントリ中の同パラメータと一致し、以降のアプリケーション名簿の更新情報にそのアプリケーション プロトコル エンティティが含まれているかどうかを判断するために利用される。

ノード ID (Node ID) : 登録が成功した場合のみ存在する (登録が不成功または登録解除のときには存在しない)。これは、登録したアプリケーション プロトコル エンティティに、そのローカル ノードのノード ID を与えるものである。これにより、アプリケーション プロトコル エンティティは受信したアプリケーション名簿の中から自分に関連するエントリを見つけることができ、会議に実際に登録されたときを判断する役に立つ。ノード ID とアプリケーション プロトコル エンティティ ID の組み合わせで、そのアプリケーション プロトコル エンティティを一意に判別することができる。

結果 (Result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。「成功 (successful)」、「無効な会議 (invalid conference)」、「MCS ユーザ ID の変更不許可 (not permitted to change MCS User ID)」。

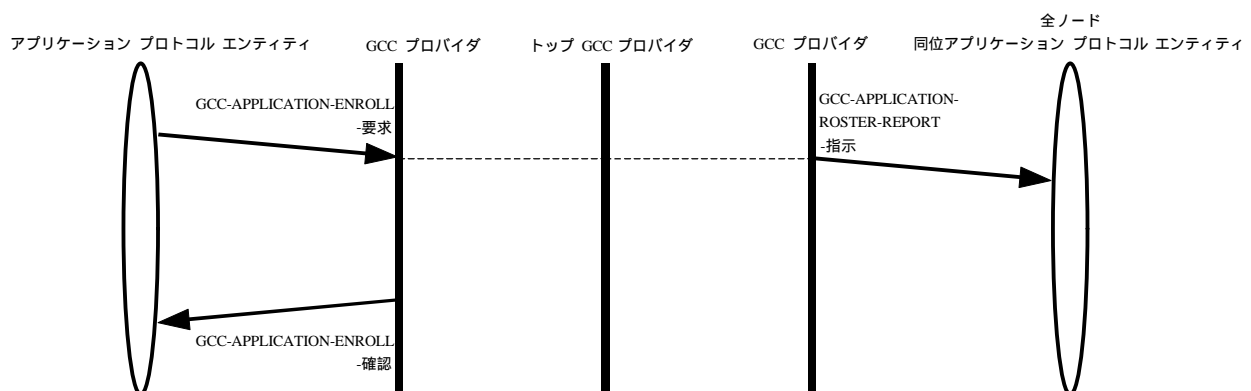


図 7-19/JT-T124 GCC-APPLICATION-ENROLL プリミティブのシーケンス (ITU-T T.124)

7.3.3.3. GCC-APPLICATION-ROSTER-REPORT

このプリミティブは、一部あるいは全ての会議アプリケーション名簿をノードコントローラと登録アプリケーション プロトコル エンティティの各々に送るために GCC プロバイダによって利用される。GCC プロバイダは、名簿に変更があったことを発見すると、GCC-APPLICATION-ROSTER-REPORT-指示を自動的に生成する。これは、GCC-APPLICATION-ENROLL-要求の結果として、あるいは、会議からのアプリケーション プロトコル エンティティのデタッチや、ノードの会議からの離脱を検出したときなどに起こる。任意のアプリケーション プロトコル エンティティに関する名簿の任意の部分が変更されたとき、GCC-APPLICATION-ROSTER-REPORT-指示がノードコントローラへ発行される。任意の同位アプリケーション プロトコル エンティティに関する名簿の任意の部分が変更されたとき、GCC プロバイダは活性状態のアプリケーション プロトコル エンティティに対して GCC-APPLICATION-ROSTER-

REPORT-指示を発行する。この場合、名簿のうち関連するアプリケーション プロトコル セッションの部分だけを含める必要がある。GCC プロバイダは他の場合にもアプリケーション プロトコル エンティティに対して GCC-APPLICATION-ROSTER-REPORT-指示を発行してもよい(同位アプリケーション プロトコル エンティティ以外のエントリが変わったときなど)。その場合、指示プリミティブの名簿の部分に同位アプリケーション プロトコル エンティティ以外に関する部分が含まれる。非活性状態のアプリケーション プロトコル エンティティに対しても、特定のセッション ID を指定して登録された場合には、活性状態のアプリケーション プロトコル エンティティと同様のルールが適用される。セッション ID なしで登録された非活性状態のアプリケーション プロトコル エンティティに対しては、その登録アプリケーション プロトコル エンティティと同じアプリケーション プロトコルをベースとする全てのアプリケーション プロトコル セッションについて名簿の内容が変更されたとき、GCC プロバイダは GCC-APPLICATION-ROSTER-REPORT を発行する。

このプリミティブの型とパラメータを表 7-26/JT-T124 に示す。ノードコントローラに対して発行される場合、アプリケーション プロトコル固有の各パラメータ(会議 ID(Conference ID)以外の全て)が、それぞれのアプリケーション プロトコル セッションについて同様に存在する。

このプリミティブを使用したときのシーケンスを図 7-20/JT-T124 に示す。

表 7-26/JT-T124 GCC-APPLICATION-ROSTER-REPORT プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	指示
会議 ID (Conference ID)	M
更新アプリケーション名簿 (Updated Application Roster)	M

会議 ID(Conference ID): 指定された会議に対応する MCS ドメインの識別子。

更新アプリケーション名簿(Updated Application Roster): 更新されたアプリケーション名簿。指示で指定する各アプリケーション プロトコル セッションについて、表 7-27/JT-T124 で示す情報を含む。

表 7-27/JT-T124 各アプリケーション プロトコル セッションについての更新アプリケーション名簿の内容
(ITU-T T.124)

パラメータ	記述
セッション キー (Session Key)	アプリケーション プロトコル セッションを特定するセッション キー(もしあれば、セッション ID を含む)。
アプリケーション プロトコル エンティティ リスト更新フラグ (Application Protocol Entity List Updated Flag)	このセッション キー(Session Key)に対応するアプリケーション プロトコル エンティティ リスト(Application Protocol Entity List)が更新されたかどうかを示すフラグ。更新されていた場合は、後述するようなアプリケーション プロトコル エンティティ リスト(Application Protocol Entity List)が含まれる。
アプリケーション プロトコル	このアプリケーション プロトコル セッションの一部として

エンティティリスト (Application Protocol Entity List) (条件付き)	会議に登録されたアプリケーション プロトコル エンティティのリスト。それぞれのアプリケーション レコードを含む。このリストの各エントリの内容を表 7-28/JT-T124 に示す。
インスタンス番号 (Instance Number)	このセッション キー(Session Key)に関するアプリケーション名簿のインスタンス番号。16 ビットの数字で、アプリケーション名簿が変更されるごとにモジュロ 2^{16} でインクリメントされる。これにより、アプリケーション プロトコル エンティティは、変更途中にある特定の能力セットの操作を競合なしに処理できる。
同位エンティティ追加フラグ (Peer Entities Added Flag)	前回のインスタンス以降、どこかのノードでアプリケーション名簿に一つ以上の同位アプリケーション プロトコル エンティティが追加されたかどうかを示すフラグ。このフラグは、同位エンティティ削除フラグ(Peer Entities Removed Flag)と互いに排他的でない。
同位エンティティ削除フラグ (Peer Entities Removed Flag)	前回のインスタンス以降、どこかのノードでアプリケーション名簿から一つ以上の同位アプリケーション プロトコル エンティティが削除されたかどうかを示すフラグ。このフラグは、同位エンティティ追加フラグ(Peer Entities Added Flag)と互いに排他的でない。
アプリケーション能力リスト更新フラグ (Application Capabilities List Updated Flag)	このセッション キー(Session Key)に対応するアプリケーション能力リストが更新されたかどうかを示すフラグ。更新されていた場合、後述するようなアプリケーション能力リストが含まれる。
アプリケーション能力リスト (Application Capabilities List) (条件付き)	このセッション キー(Session Key)に対する集約されたアプリケーション能力リスト全体。

表 7-28/JT-T124 アプリケーション プロトコル エンティティ リストの内容

(ITU-T T.124)

パラメータ	記述
ノード ID (Node ID)	そのアプリケーション プロトコル エンティティが登録されたノードを識別するノード ID。
アプリケーション プロトコル エンティティ ID (Application Protocol Entity ID)	ノード ID(Node ID)で指定されたノード内でアプリケーション プロトコル エンティティをユニークに識別するための識別子。
活性 / 非活性 フラグ (Active/Inactive Flag)	登録されたアプリケーション プロトコル エンティティが非活性状態か活性状態かを示すフラグ。
アプリケーション ユーザ ID (Application User ID) (条件付)	登録されたアプリケーション プロトコル エンティティに関連づけられた MCS ユーザ ID。

き)		
議長動作能力 (Conducting Operation Capable)	Operation	このフラグは、(もし、そのアプリケーション プロトコル仕様が議長エンティティの行うべき手続きを定めているならば)、アプリケーション プロトコル エンティティが議長のアプリケーション プロトコル エンティティとして動作する能力があるかどうかを表わす。1 ノードあたり二つ以上の同位アプリケーション プロトコル エンティティにこのフラグがセットされることはない。あるノードが会議議長になった場合、そのノード上の指名されたアプリケーション プロトコル エンティティが(もしあれば)、このアプリケーション プロトコル セッションの議長指名アプリケーション プロトコル エンティティとなる。
スタートアップ チャンネル(Start-up Channel) (条件付き)		このパラメータがもしあるならば、その値は、「静的(Static)」、「動的マルチキャスト(Dynamic Multicast)」、「動的プライベート(Dynamic Private)」、「動的ユーザ ID(Dynamic UserId)」のいずれかである。このパラメータは、アプリケーション プロトコル エンティティが一連の起動動作で使用する MCS チャンネルのタイプを示す。厳密なこのパラメータの実装はアプリケーション プロトコル固有の仕様による。特定のアプリケーション プロトコルでは、これらのチャンネルタイプの中のいくつかが無効でないということも有り得る。
非集約型能力リスト (Non-Collapsing Capabilities List) (条件付き)		アプリケーション名簿にアプリケーション レコードの一部として保存される各アプリケーション プロトコル エンティティの(標準か非標準の)固有の能力リストである。

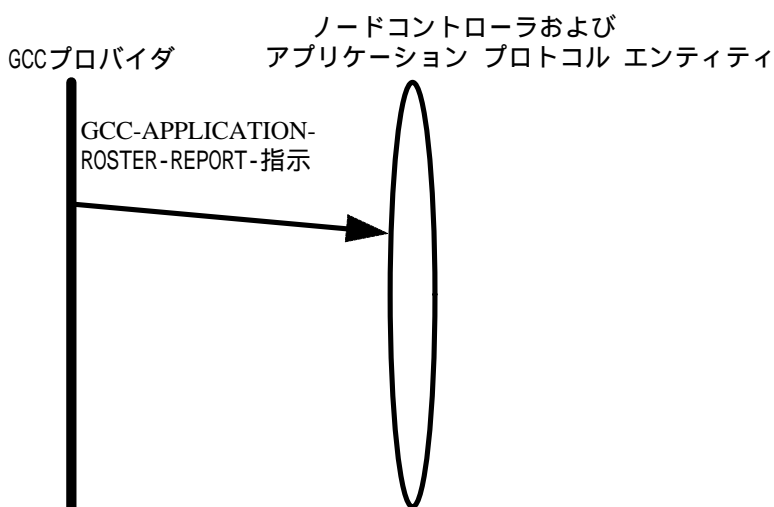


図 7-20/JT-T124 GCC-APPLICATION-ROSTER-REPORT プリミティブのシーケンス (ITU-T T.124)

7.3.3.4. GCC-APPLICATION-ROSTER-INQUIRE

アプリケーション プロトコル エンティティは、GCC-APPLICATION-ROSTER-INQUIRE-要求プリミティブを用いることにより、一つまたは複数のアプリケーション プロトコル セッション、または全てのアプリケーション プロトコル エンティティに対応する会議アプリケーション名簿の一部分を要求することが出来る。この情報は GCC-APPLICATION-ROSTER-INQUIRE-確認プリミティブによって GCC から返送される。

このプリミティブの型とパラメータを表 7-29/JT-T124 に示す。

このプリミティブを使用したときのシーケンスを図 7-21/JT-T124 に示す。

表 7-29/JT-T124 GCC-APPLICATION-ROSTER-INQUIRE プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M(=)
セッション キー (Session Key)	M	M(=)
アプリケーション 名簿 (Application Roster)		M
結果 (Result)		M

会議 ID(Conference ID): プリミティブを適用する会議の識別子。

セッション キー(Session Key): アプリケーション プロトコル セッションにユニークな識別子。全てのアプリケーション プロトコル エンティティについてのアプリケーション情報を要求することを表わすために、セッション キーの一部が NULL のキーを指定してもよい。

アプリケーション名簿(Application Roster): セッション キー(Session Key)で指定されるアプリケーション プロトコル セッションについての名簿エントリのリストを含む。要求中のセッション キー(Session Key)に指定された部分(その長さまで)が一致するセッション キーに対する名簿エントリ。もし、セッション キー(Session Key)の一部としてセッション ID(Session ID)が指定された場合は、同じセッション ID を持つ名簿エントリ。一致した各エントリについて返送される情報を表 7-30/JT-T124 に示す。

表 7-30/JT-T124 各アプリケーション プロトコル セッションについてのアプリケーション名簿の内容

(ITU-T T.124)

パラメータ	記述
セッション キー (Session Key)	そのアプリケーション プロトコル セッションを特定するセッション キー(もしあれば、セッション ID を含む)。
アプリケーション プロトコル エンティティ リスト (Application Protocol Entity)	このアプリケーション プロトコル セッションの一部として会議に登録されたアプリケーション プロトコル エンティティのリスト。それぞれのアプリケーション レコードを含む。

List)	このリストの各エントリの内容を表 7-28/JT-T124 に示す。
インスタンス番号 (Instance Number)	このセッションキー(Session Key)に関するアプリケーション名簿のインスタンス番号。16 ビットの数字で、アプリケーション名簿が変更される毎にモジュロ 2^{16} でインクリメントされる。これにより、アプリケーション プロトコル エンティティは、変更途中にある特定の能力セットの操作を競合なしに処理できる。
アプリケーション能力リスト (Application Capabilities List)	このセッション キーに対する集約されたアプリケーション能力リスト全体。

結果(Result): 要求が受け入れられたか拒否されたか、および拒否の場合はその理由を示す。以下のいずれかの値である。

「成功(successful)」、「無効な会議(invalid conference)」。

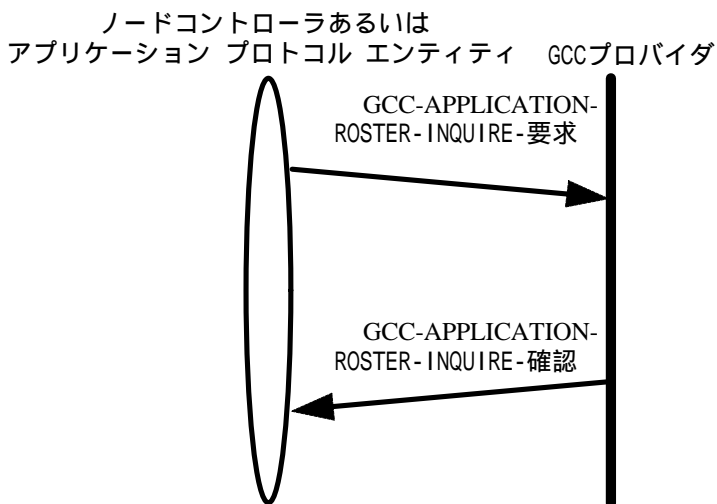


図 7-21/JT-T124 GCC-APPLICATION-ROSTER-INQUIRE プリミティブのシーケンス (ITU-T T.124)

7.3.3.5. GCC-APPLICATION-INVOKE

このプリミティブは、アプリケーション プロトコル エンティティがノード コントローラによって発行され、指定されたいくつかのノードにおいて、一つもしくは複数のアプリケーション プロトコル エンティティを起動することを要求するのに用いられる。指定されたノードでは、このプリミティブの指示が、指定されたアプリケーション プロトコル エンティティを起動するノードコントローラに対して発行される。この指示は、指定された各アプリケーション プロトコル エンティティについて、要求中で指定されたセッション キーを持つ新しいアプリケーション プロトコル エンティティを生成し、それから、会議に活性状態で登録して対応するアプリケーション プロトコル セッションの一部になることを要請する。もし、同一セッション キーを持つ非活性のアプリケーション プロトコル エンティティが宛先ノードに既に存在するならば、この指示の受信はそのアプリケーション プロトコル エンティティに活性状態で再登録することを要請する。もし、宛先ノードに同一セッション キーを持つアプリケーション プロトコル エンティティ

が既に活性状態で存在するならば、この指示は無視してもよい。

このプリミティブの型とパラメータを表 7-31/JT-T124 に示す。

このプリミティブを使用したときのシーケンスを図 7-22/JT-T124 に示す。

表 7-31/JT-T124 GCC-APPLICATION-INVOKE プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID (Conference ID)	M	M	M(=)
起動アプリケーション プロトコ ル エンティティ リスト (Application Protocol Entity Invoke List)	M	M(=)	M(=)
宛先ノード(ノード ID のリスト または NULL) (Destination Nodes (List of Node IDs or NULL))	O		
起動ノード ID (Invoking Node ID)		M	
結果 (Result)			M

会議 ID(Conference ID): プリミティブを適用する会議の識別子。

起動アプリケーション プロトコル エンティティ リスト(Application Protocol Entity Invoke List): 起動される(一つまたはいくつかの)アプリケーション プロトコル エンティティのリスト。このリストの各エントリの内容を表 7-32/JT-T124 に示す。

表 7-32/JT-T124 アプリケーション プロトコル エンティティ リストの各エントリの内容

(ITU-T T.124)

パラメータ	記述
セッション キー (Session Key)	起動するアプリケーション プロトコル セッションを指定するセッション キー(もしあれば、セッション ID を含む)。
期待される能力セット (Expected Capability Set)	これは(アプリケーション能力リストと同一形式の)アプリケーション能力のリストでオプションである。起動されるアプリケーション プロトコル エンティティが持つべき能力セットを示す。指定されたノードでアプリケーション プロトコル エンティティがこれを満たすことができなければ、そのノードで起動されるべきではない。このパラメータが含まれていなければ、起動されるアプリケーション プロトコル エンティティに制約はない。能力クラスの定義は、この「期待され

	<p>る能力セット(Expected Capability Set)」を解釈するために利用されるが、GCC が「アプリケーション能力リスト(Application Capabilities List)」について解釈する場合と同様ではないことに注意を要する。Unsigned-MAX クラスの能力では、起動されるアプリケーション プロトコル エンティティは指定された能力以下を持たなければならない。Unsigned-MIN クラスでは、起動されるアプリケーション プロトコル エンティティは指定された能力以上を持たなければならない。Logical クラスの能力では、起動されるアプリケーション プロトコル エンティティは全く同じ能力を持たなければならない。</p>
<p>スタートアップ チャンネル (Start-up Channel) (条件付き)</p>	<p>「静的 (Static)」、「動的マルチキャスト (Dynamic Multicast)」、「動的プライベート (Dynamic Private)」、「動的ユーザ ID (Dynamic UserId)」のいずれか。このパラメータは、アプリケーション プロトコル エンティティが起動時に使用する MCS チャンネルのタイプを示す。特定のアプリケーション プロトコルでは、これらのチャンネルタイプの中のいくつかが無効でないということも有り得る。もし、無効なタイプが使われた場合には、そのアプリケーション プロトコル エンティティは起動するべきではない。</p>
<p>必須 / オプション フラグ (Mandatory/Optional Flag)</p>	<p>このフラグは、このリスト中の他のアプリケーション プロトコル エンティティを起動するために、このアプリケーション プロトコル エンティティを起動しなければならないことを示す。このフラグは、「必須 (Mandatory)」とマークされた全てのアプリケーション プロトコル エンティティの起動に成功した場合のみ、このリスト中のアプリケーション プロトコル エンティティを起動することを宛先ノードに指示するために使用される。</p>

宛先ノード(ノード ID のリストまたは NULL) (Destination Nodes(List of Node IDs or NULL)): ノード ID のリストの場合は、要求を送る宛先の GCC プロバイダを指示する。NULL の場合は、会議 ID(Conference ID)で指定される会議の全ノードに要求を送ることを示す。

起動ノード ID(Invoke Node ID): GCC-APPLICATION-INVOKE-要求を発行したノードのノード ID。

結果(Result): 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは以下のリスト中の可能な結果の一つを含む。

「成功(successful)」、「無効な会議(Invalid conference)」。

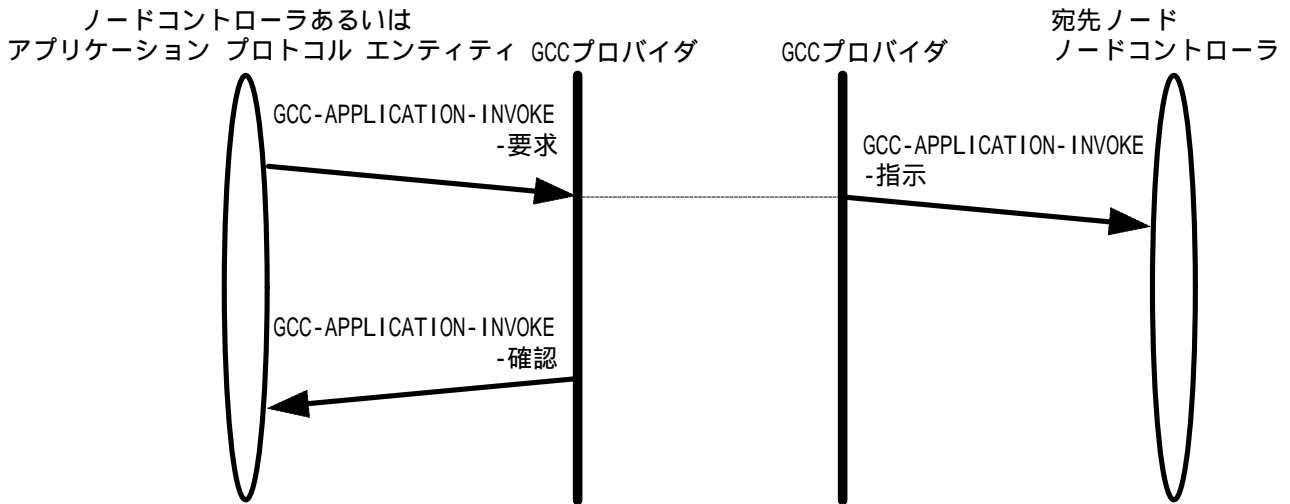


図 7-22/JT-T124 GCC-APPLICATION-INVOKE プリミティブのシーケンス
(ITU-T T.124)

7.4. アプリケーション レジストリ

アプリケーション レジストリは GCC の機能構成要素である。レジストリは、トップ GCC プロバイダに置かれた中央データベースを操作するアプリケーション プロトコル エンティティに機能セットを提供する。レジストリの内容は単一の会議中でユニークである。レジストリデータベースに蓄積された情報の意味はアプリケーション プロトコルによって定義される。中央の格納庫がアプリケーション プロトコル エンティティ間のコミュニケーション確立の助けとなる。つまり、通信すべき相手の共通チャンネル ID や、排他アクセスを制御するための共通のトークン ID、あるいは共通のパラメータ値（パラメータはアプリケーション プロトコルの特定の用途のレジストリエントリである。）を知るのに役立つ。このことにより、チャンネル ID、トークン ID と他のパラメータ値を静的に予約しておく必要がなくなる。動的取得機能をサポートすることにより、アプリケーション レジストリでは、オーディオ グラフィックとオーディオ ビジュアル会議をさらに強化するための新規機能の導入が、標準あるいは非標準に関わらず容易になる。レジストリはまた、会議の中で全体的にユニークな数のハンドルの割り当てのための多目的サービスも含んでいる。

7.4.1. レジストリ キー

レジストリ データベース内のエントリを保存、読み出しするために使用されるインデックスは、アプリケーション プロトコルによって規定されたキーである。標準、非標準いずれのキーを使用してもよい。標準キーがこの標準などによって割り当てられる。非標準キーは製造者に特化したものでよい。両方ともキー値の選択で重複を避けるよう構成される。

レジストリ キーは、7.3.1 節で定義される、セッション キーとリソース ID との組合せで構成される。レジストリ キーを必要とする、どのレジストリ制御にも使用されたセッションキーは、アプリケーション名簿（活性あるいは非活性メンバー）に示すアプリケーション プロトコル セッションのセッション キーと同じでなければならない。リソース ID は ASN.1 型のオクテット スtring である。リソース ID により、1 つのアプリケーション プロトコル エンティティに複数のレジストリ キーを使用することが可能となる。リソース ID の値はそれを利用するアプリケーション プロトコルの仕様によって定義される。

7.4.2. 所有権と持続性

レジストリによるレジストリ キーの所有権は、GCC-REGISTRY-REGISTER-CHANNEL、GCC-REGISTRY-ASSIGN-TOKEN、GCC-REGISTRY-SET-PARAMETER のいずれかの要求プリミティブによって、ある項目を保存するように要求した最初のアプリケーション プロトコル エンティティに与えられる。

パラメータ レジストリ項目の場合は、パラメータが生成される時、その所有者がそのエントリの値の変更を許可するアプリケーション プロトコル エンティティの範囲を規定してもよい。所有者は、「所有者 (owner)」、「セッション (session)」、「パブリック (public)」のいずれかの変更権を規定してもよい。そのパラメータが「所有者 (owner)」の場合は所有者のみによって (チャンネルとトークンの場合のように)、「セッション (session)」の場合は同位のアプリケーション プロトコル エンティティ (同じアプリケーション プロトコル セッション内のもの) によって、あるいは「パブリック (public)」の場合は会議中のどのアプリケーション プロトコル エンティティによってでも、それぞれ変更可能である。所有者以外のアプリケーション プロトコル エンティティによるパラメータの内容の変更は、そのエントリの所有者の識別子を変更しない。パラメータの所有者はいつでも変更権の範囲を再定義できる。もし、他のノードが GCC-REGISTRY-SET-PARAMETER-要求を発行することで変更権を変更しようと試みた場合、要求者が変更権を持っていれば、パラメータは要求された値に設定されるが、権利は変更されない。

エントリを削除するためには、どのレジストリ項目タイプ (チャンネル、トークン、あるいはパラメータ) でも所有権が必要となる。一旦、エントリが削除されると、所有者はなくなり、そして、異なったアプリケーション プロトコル エンティティが引き継ぐことができる。

所有者が会議からデタッチする時、レジストリ エントリは自動的に削除されない。その内容はいつまでも変化せず持続する。しかし、エントリの所有権は、所有者がデタッチする時、取り去られる。この機構が、残されたアプリケーション プロトコル エンティティが変更 (パラメータ エントリのみ) したり、もし無用になったらそのエントリを削除することを可能とする。残されたパラメータエントリの項目を保存することを、最初に要求したアプリケーション プロトコル エンティティは、その新しい所有者になる。

レジストリ エントリのレジストリ キーを形成するために使用されたセッション キーに対応するセッションで、すべてのアプリケーション プロトコル エンティティが未登録になる時、そのレジストリ エントリは自動的に削除される。

7.4.3. 動的な割り当て

MCS はスタティックとダイナミック チャンネル ID を区別する。後者はユーザ ID、プライベートチャンネル ID と割り当てチャンネル ID から構成される。ダイナミック チャンネル ID は、レジストリ サービスの使用とは別に、アプリケーション プロトコル エンティティによって直接生成、削除される。アプリケーション レジストリは、他のアプリケーション プロトコル エンティティによる検索のために、ダイナミック チャンネル ID を中央の格納庫にしまっておく。それは MCS プリミティブを用いてこれらのチャンネル ID をテストしたり、あるいは操作したりしない。

MCS はスタティックとダイナミック トークン ID を区別しない。しかし、意味的に類似したものにするため、GCC によって意図的な区別が行われる。トークン ID の 1~16383 まではスタティックとし、他の仕様によって割り当てられるために予約されている。トークン ID の 16384~65535 まではダイナミックとし、レジストリ データベースでエントリを生成することの一部として、要求に基づいてトップ GCC プロバイダによって割り当てられる。関連づけられたエントリが削除される時、それが持っていたトークン ID は再割り当てのために利用可能とする。レジストリは、割り当てたトークン ID に対して、獲得と解放のような、MCS プリミティブを起動しない。それはただ特定の ID 値を選択して、それを他に広めるだけである。アプリケーション プロトコル エンティティは、それぞれのロジックに従って、割り当てたトークン ID に基づいた自由な動作を行ってよい。

7.4.4. サービス概要記述

以下に、この節で定義するプリミティブのリストと各機能の要約を示す。

- GCC-REGISTRY-REGISTER-CHANNEL : MCS ダイナミック チャネルのチャンネル ID を登録するためにアプリケーション プロトコル エンティティによって使用される。アプリケーション プロトコル エンティティは GCC-REGISTRY-RETRIEVE-ENTRY を使用して、ノードが既にそのチャンネル ID(レジストリ キー)を登録しているかどうかを判断し、もし登録しているならそのチャンネル ID 値を検索し、エントリを調べることができる。
- GCC-REGISTRY-ASSIGN-TOKEN : ダイナミック トークンを割り当て、その割当トークン ID を登録するためにアプリケーション プロトコル エンティティによって使用される。アプリケーション プロトコル エンティティは GCC-REGISTRY-RETRIEVE-ENTRY を使用して、ノードが既にそのトークン ID(レジストリ キー) を登録しているかどうかを判断し、もし登録しているならそのトークン ID 値を検索し、エントリを調べることができる。
- GCC-REGISTRY-SET-PARAMETER : レジストリ データベース中の会議中のいずれかのノードによって調査あるいは変更されている値を設定するために、アプリケーション プロトコル エンティティによって使用される。
- GCC-REGISTRY-RETRIEVE-ENTRY : いずれかのレジストリ エントリの現在の内容を抽出するために、アプリケーション プロトコル エンティティによって使用される。
- GCC-REGISTRY-DELETE-ENTRY : レジストリ データベース中のエントリを削除するために、アプリケーション プロトコル エンティティによって使用される。
- GCC-REGISTRY-MONITOR : レジストリ エントリの監視を有効(あるいは無効)にするために、アプリケーション プロトコル エンティティによって使用される。一旦有効にされると、このプリミティブの指示型はエントリの(削除を含む)内容のどのような変更も、要求したアプリケーション プロトコル エンティティに通知する。
- GCC-REGISTRY-ALLOCATE-HANDLE : 単一会議の範囲の中でユニークな 32 ビットのハンドルを生成させるために使用される。

特定の要求者についての、レジストリ要求プリミティブと、トップ GCC プロバイダでの(成功したことによる)動作結果と、それらに関連する確認プリミティブとの順序は維持されている。

7.4.4.1. GCC-REGISTRY-REGISTER-CHANNEL

GCC-REGISTRY-REGISTER-CHANNEL-要求プリミティブは、特定の MCS チャネルがレジストリ キーによって示された方法でアプリケーション プロトコル エンティティによって使用するために指定されていることを他のノードのアプリケーション プロトコル エンティティに通知するために、アプリケーション プロトコル エンティティによって発行されるかもしれない。いずれかのアプリケーション プロトコル エンティティが一旦特定なレジストリ キーの使用を登録すれば、他のノードにおけるアプリケーション プロトコル エンティティは問題のレジストリ キーを指定して GCC-REGISTRY-RETRIEVE-ENTRY-要求プリミティブを発行することによって、チャンネルがこのキーのために登録されているどうか(登録されていたら、チャンネル ID の値を)を検出してもよい。一旦チャンネルが登録されると、アプリケーション プロトコル エンティティが同じレジストリ キーを使用して登録しよう(所有者を含む)とした場合、その試みはキーが既に使用されているという指摘と共に拒否される。表 7-33/JT-T124 にプリミティブの型とパラメータを示す。図 7-23/JT-T124 に、このプリミティブを使用するときのイベントシーケンスを示す。

表 7-33/JT-T124 GCC-REGISTRY-REGISTER-CHANNEL プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M(=)
レジストリ キー (Registry Key)	M	M(=)
チャンネル ID (Channel ID)	M	
レジストリ 項目 (Registry Item)		C
所有者 (Owner)		C
結果 (Result)		M

会議 ID (Conference ID) : プリミティブが参照する会議の識別子。

レジストリ キー (Registry Key) : 項目を格納するデータベースのインデックス。

チャンネル ID (Channel ID) : アプリケーション プロトコル エンティティによって指定されるダイナミック チャンネル ID (ユーザ ID 、プライベート、あるいは割り当て)。

レジストリ項目 (Registry Item) : 要求の有効後のエントリ値。要求が成功なら、これはその(要求された)パラメータの値となる。矛盾した型あるいは既に所有されたインデックスのため不成功なら、これは要求前の値のままである。それは、「チャンネル ID(Channel ID)」、「トークン ID(Token ID)」、「パラメータ値(Parameter Value)」であるか、または、エントリが空なら、「なし(None)」である。(型と値の両方(もしあれば)が、このパラメータによって示される。)ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

所有者 (Owner) : このパラメータはこのレジストリ エントリの現在の所有者を示す。もし所有されるなら、このパラメータは所有者が存在するノードの「ノード ID (Node ID)」と所有しているアプリケーション プロトコル エンティティの「エンティティ ID (Entity ID)」を含む。エントリが所有されていない場合、このパラメータは示される。ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

結果 (Result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、次の中のいずれか一つを含む。「成功 (successful)」、「無効な会議 (invalid conference)」、「レジストリ フル (registry full)」、「インデックスが既に存在する (index already exists)」、「矛盾した型 (inconsistent type)」、「無効な要求者 (invalid requester)」。

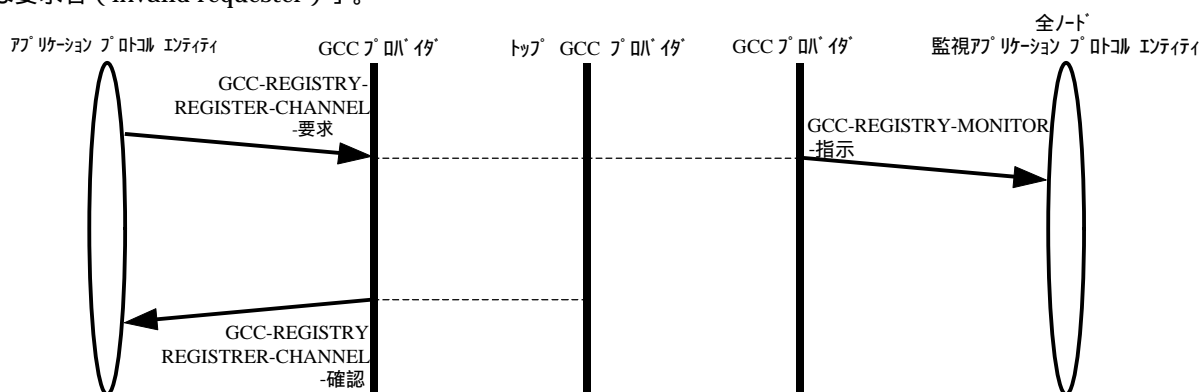


図 7-23/JT-T124 GCC-REGISTRY-REGISTER-CHANNEL プリミティブのシーケンス (ITU-T T.124)

7.4.4.2. GCC-REGISTRY-ASSIGN-TOKEN

GCC-REGISTRY-ASSIGN-TOKEN-要求プリミティブはトークンを特定なレジストリ キーと結び付けるため、アプリ

ケーション プロトコル エンティティによって発行されるかもしれない。要求が成功なら、トークン ID は確認 プリミティブのパラメータで返送される。一旦アプリケーション プロトコル エンティティが指定したトークンが特定なレジストリ キーを使用中になると、他のノードのアプリケーション プロトコル エンティティは問題のレジストリ キーを規定する GCC-REGISTRY-RETRIEVE-ENTRY-要求プリミティブを発行することによって、トークンがこのキーに割り当てられたかどうか（割り当てられていたら、トークン ID の値を）を検出してもよい。一旦トークンが割り当てられると、アプリケーション プロトコル エンティティがトークンを同じレジストリ キーを使用して登録しよう（所有者を含む）とした場合、その試みは既に割り当てられているという指摘と共に拒否される。表 7-34/JT-T124 にパラメータとこのプリミティブの型を示す。図 7-24/JT-T124 に、このプリミティブを使用したイベントシーケンスを示す。

表 7-34/JT-T124 GCC-REGISTRY-ASSIGN-TOKEN プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M(=)
レジストリ キー (Registry Key)	M	M(=)
トークン ID (Token ID)		C
レジストリ 項目 (Registry Item)		C
所有者 (Owner)		C
結果 (Result)		M

会議 ID (Conference ID) : プリミティブが参照する会議の識別子。

レジストリ キー (Registry Key) : 項目を格納するデータベースのインデックス。

トークン ID (Token ID) : トップ GCC プロバイダによって割り当てられたダイナミック トークン ID (16384 以上)。

ローカルに生成されたエラー状態の確認ではこのパラメータは存在しない。

レジストリ項目 (Registry Item) : 要求の有効後のエントリ値。要求が成功なら、これはその(要求された)パラメータの値となる。矛盾した型あるいは既に所有されたインデックスのため不成功なら、これは要求前の値のままである。それは、「チャンネル ID(Channel ID)」、「トークン ID(Token ID)」、「パラメータ値(Parameter Value)」であるか、または、エントリが空なら、「なし(None)」である(型と値の両方(もしあれば)が、このパラメータによって示される)。

ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

所有者 (Owner) : このパラメータはこのレジストリ エントリの現在の所有者を示す。もし所有されるなら、このパラメータは所有者が存在するノードのノード ID と所有しているアプリケーション プロトコル エンティティのエンティティ ID を含む。エントリが所有されていない場合、このパラメータは示される。ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

結果 (Result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、次の中のいずれか一つを含む。「成功(successful)」、「無効な会議(invalid conference)」、「レジストリ フル(registry full)」、「インデックスが既に存在する(index already exists)」、「矛盾した型(inconsistent type)」、「無効な要求者(invalid requester)」。

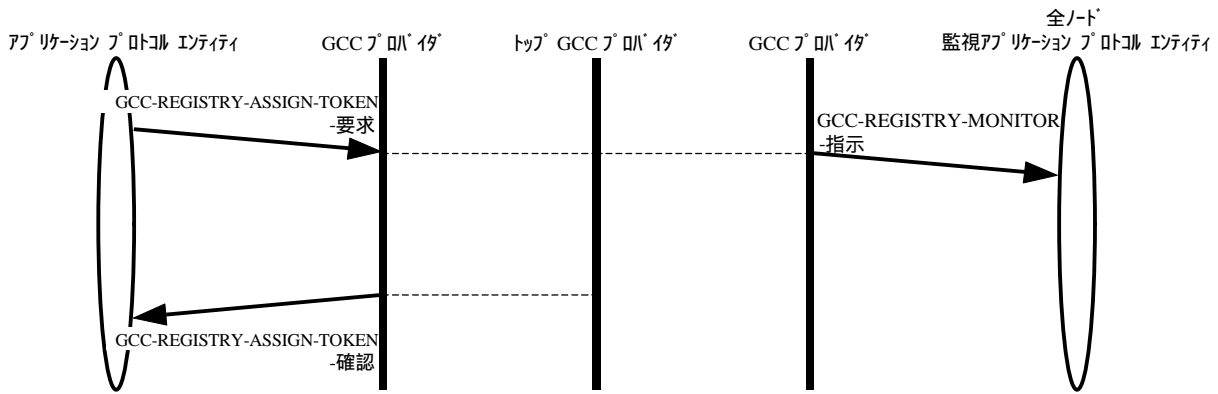


図 7-24/JT-T124 GCC-REGISTRY-ASSIGN-TOKEN プリミティブのシーケンス
(ITU-T T.124)

7.4.4.3. GCC-REGISTRY-SET-PARAMETER

GCC-REGISTRY-SET-PARAMETER-要求プリミティブはレジストリ パラメータの値を設定または変更するために、アプリケーション プロトコル エンティティによって発行されるかもしれない。レジストリ エントリが GCC-REGISTRY-MONITOR-要求プリミティブによって監視状態に設定されている場合、REGISTRY-SET-PARAMETER-要求が成功すると、その度ごとに会議内の全ノードにおけるアプリケーション プロトコル エンティティに対して、GCC-MONITOR-INDICATION が送られることとなる。特定キーに対するレジストリ エントリが存在する場合、パラメータをセットする要求が承認されるのは、そのエントリが既にパラメータであった場合のみである。表 7-35/JT-T124 にパラメータとこのプリミティブの型を示す。図 7-25/JT-T124 にこのプリミティブを使用するときのイベントシーケンスを示す。

表 7-35/JT-T124 GCC-REGISTRY-SET-PARAMETER プリミティブの型とパラメータ
(ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M(=)
レジストリ キー (Registry Key)	M	M(=)
パラメータ値 (Parameter Value)	M	
レジストリ 項目 (Registry Item)		C
所有者 (Owner)		C
変更権 (Modification Rights)	O	C
結果 (Result)		M

会議 ID (Conference ID) : プリミティブが参照する会議識別子。

レジストリ キー (Registry Key) : 項目を格納するデータベースのインデックス。

パラメータ値 (Parameter Value) : アプリケーション プロトコル エンティティで規定されるオクテット列。

レジストリ項目 (Registry Item) : 要求の有効後のエントリ値。要求が成功なら、これはその(要求された)パラメータの値となる。矛盾した型のため不成功なら、これは要求前の値のままである。それは、「チャンネル ID(Channel ID)」、

「トークン ID(Token ID)」、「パラメータ値(Parameter Value)」であるか、または、エントリが空なら、「なし(None)」である(型と値の両方(もしあれば)が、このパラメータによって示される)。ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

所有者 (Owner) : このパラメータはこのレジストリ エントリの現在の所有者を示す。もし所有されるなら、このパラメータは所有者が存在するノードのノード ID と所有しているアプリケーション プロトコル エンティティのエンティティ ID を含む。エントリが所有されていない場合、このパラメータは示される。このプリミティブの成功的な完了の結果、要求者がこのエントリの所有者になるなら、要求者は確認プリミティブで新しい所有者として示される。ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

変更権(Modification Rights): この任意のパラメータはこのレジストリ エントリ値を変更することを許されたアプリケーション プロトコル エンティティの範囲を規定する。このパラメータ値は「所有者 (owner)」、「セッション (session)」、あるいは「パブリック (public)」のいずれかである。「所有者 (owner)」は、その所有者だけがこのエントリを変更してもよい(エントリが所有される限り)ことを規定する。「セッション (session)」は、所有者と同じアプリケーション プロトコル セッションの一部のどのアプリケーション プロトコル エンティティでもこのエントリを変更してもよいことを規定する。「パブリック (public)」は、会議に参加登録したどのアプリケーション プロトコル エンティティでも、このエントリを変更してもよいことを規定する。エントリが最初に生成される時、このパラメータが指定されていないなら、「パブリック (Public)」の値となる。他の時で、このパラメータが指定されていない時は、その値は変化しない。このレジストリ エントリの所有者だけ(エントリが所有される限り)が変更権を変えることができる。非所有者が変更権を変えようと試みても、変更されない。しかしながら、要求者が変更権を持つならばパラメータ エントリそのものの変更は可能である。確認プリミティブでは、このパラメータの要求プリミティブの表現に依存せず、実際の変更権を示す。もし確認 がローカルに生成されたエラー状態なら、このパラメータは存在しない。

結果 (Result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、次のうちのいずれか一つを含む。「成功 (successful)」、「無効な会議 (invalid conference)」、「レジストリ フル (registry full)」、「矛盾した型 (inconsistent type)」、「無効な要求者 (invalid requester)」。

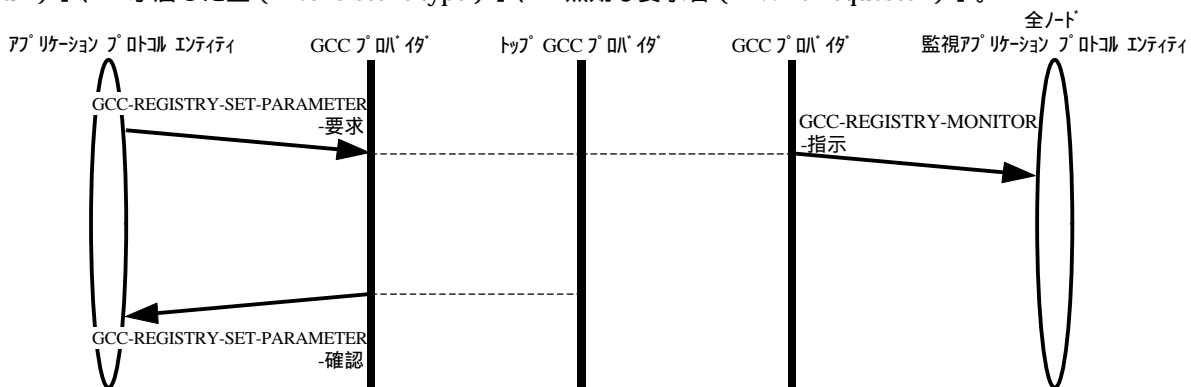


図 7-25/JT-T124 GCC-REGISTRY-SET-PARAMETER プリミティブのシーケンス (ITU-T T.124)

7.4.4.4. GCC-REGISTRY-RETRIEVE-ENTRY

GCC-REGISTRY-RETRIEVE-ENTRY-要求プリミティブは単一レジストリ エントリの内容を決定するためにアプリケーション プロトコル エンティティによって発行されるかもしれない。このプリミティブはいつでも発行されてよく、エントリがチャンネル ID、トークン ID、パラメータ、あるいは空のいずれかであることを示す。表 7-36/JT-T124 にパラ

メータとこのプリミティブの型を示す。図 7-26/JT-T124 にこのプリミティブを使用するときのイベントのシーケンスを示す。

表 7-36/JT-T124 GCC-REGISTRY-RETRIEVE-ENTRY プリミティブの型とそのパラメータ (ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M(=)
レジストリ キー (Registry Key)	M	M(=)
レジストリ 項目 (Registry Item)		C
所有者 (Owner)		C
変更権 (Modification Rights)		C
結果 (Result)		M

会議 ID (Conference ID): プリミティブが参照する会議識別子。

レジストリ キー (Registry Key): 項目を格納するデータベースのインデックス。

レジストリ 項目 (Registry Item): 「チャンネル ID(Channel ID)」、「トークン ID(Token ID)」、「パラメータ値(Parameter Value)」であるか、または、エントリが空なら、「なし(None)」である。(型と値の両方(もしあれば)が、このパラメータによって示される。)ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

所有者 (Owner): このパラメータはこのレジストリ エントリの現在の所有者を示す。もし所有されるなら、このパラメータは所有者が存在するノードの「ノード ID (Node ID)」と所有しているアプリケーション プロトコル エンティティの「エンティティ ID (Entity ID)」を含む。エントリが所有されていない場合、このパラメータは示される。ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

変更権 (Modification Rights): このパラメータは、パラメータ エントリの場合にのみ含まれ、このレジストリ エントリの値を変更することを許されたアプリケーション プロトコル エンティティの範囲を示す。このパラメータ値は「所有者 (owner)」、「セッション (session)」、あるいは「パブリック (public)」のいずれかである。「所有者 (owner)」は、その所有者だけがこのエントリを変更してもよい(エントリが所有される限り)ことを規定する。「セッション (session)」は、所有者と同じアプリケーション プロトコル セッションの一部のどのアプリケーション プロトコル エンティティでもこのエントリを変更してもよいことを規定する。「パブリック (public)」は、会議に参加登録したどのアプリケーション プロトコル エンティティでも、このエントリを変更してもよいことを規定する。エントリが最初に生成される時、このパラメータが指定されていないなら、「パブリック (Public)」の値となる。もし確認がローカルに生成されたエラー状態なら、このパラメータは存在しない。

結果 (Result): 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、次のうちのいずれか一つを含む。「成功 (successful)」、「エントリ不明 (entry not found)」、「無効な会議 (invalid conference)」。

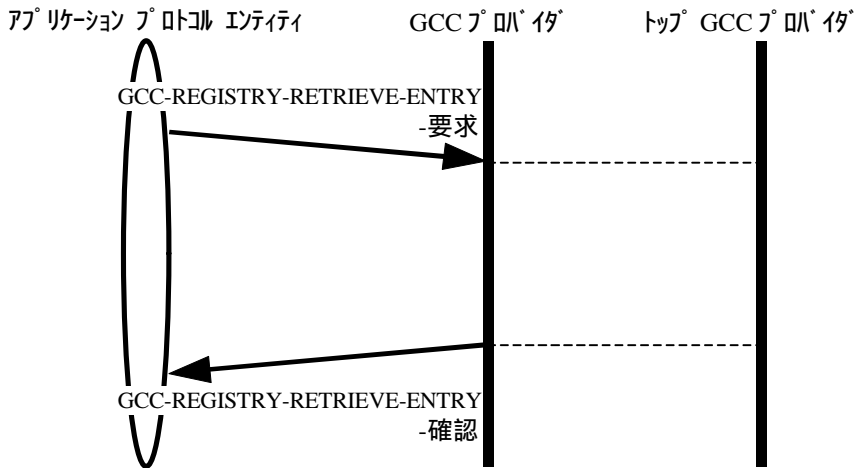


図 7-26/JT-T124 GCC-REGISTRY-RETRIEVE-ENTRY プリミティブのシーケンス (ITU-T T.124)

7.4.4.5. GCC-REGISTRY-DELETE-ENTRY

GCC-REGISTRY-DELETE-ENTRY-要求プリミティブはレジストリ エントリを消去するためにアプリケーション プロトコル エンティティによって発行されるかもしれない。レジストリ エントリの消去はそのエントリを所有する（最後の所有者が会議から切断せず、新しい所有者が割り当てられなかったなら）アプリケーション プロトコル エンティティにのみ許可されている。レジストリ エントリが GCC-REGISTRY-MONITOR-要求プリミティブの使用により監視されるようになっていた場合、GCC-REGISTRY-DELETE-ENTRY-要求が成功となる度に、会議内の全ノードにおけるアプリケーション プロトコル エンティティにエントリが削除されたことを示すために、空きレジストリ項目パラメータを伴った GCC-REGISTRY-MONITOR 指示が送られることとなる。表 7-37/JT-T124 にパラメータとこのプリミティブの型を示す。図 7-27/JT-T124 に、このプリミティブを使用したイベント シーケンスを示す。

表 7-37/JT-T124 GCC-REGISTRY-DELETE-ENTRY プリミティブの型とそのパラメータ (ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M(=)
レジストリ キー (Registry Key)	M	M(=)
結果 (Result)		M

会議 ID (Conference ID)：プリミティブが参照する会議識別子。

レジストリ キー (Registry Key)：項目を格納するデータベースのインデックス。

結果 (Result)：要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、次の中のいずれか一つを含む。「成功 (successful)」、「無効な会議 (invalid conference)」、「他の GCC プロバイダによってすでに所有されているインデックス (index already owned by another GCC Provider)」。

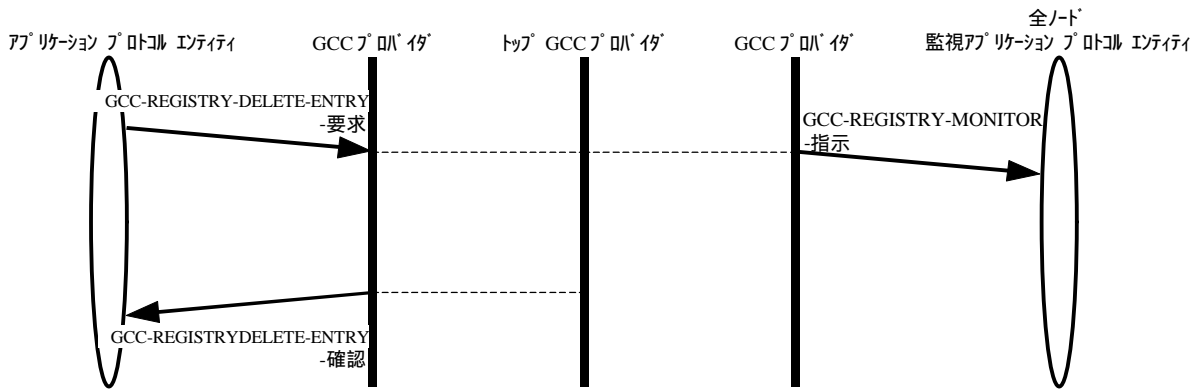


図 7-27/JT-T124 GCC-REGISTRY-DELETE-ENTRY プリミティブのシーケンス
(ITU-T T.124)

7.4.4.6. GCC-REGISTRY-MONITOR

GCC は、特定のレジストリ エントリを継続して監視する機構を提供し、これによりそのエントリが変化したか（変更された、削除された、所有者が変わった、パラメータ エントリの場合は、変更権が変わった、など）ということを持続したデータ収集を行うことなく判断できる。GCC-REGISTRY-MONITOR-要求プリミティブは、特定なレジストリ エントリの監視を有効にする（あるいは無効にする）ために、アプリケーション プロトコル エンティティによって使用されるかもしれない。有効の間は、要求したアプリケーション プロトコル エンティティは GCC-REGISTRY-MONITOR-指示プリミティブによってこのエントリの変更の全てが通知される。指示はレジストリ エントリの内容のどのような変更や、あるいはエントリの削除によって生成される。存在するレジストリ エントリだけが監視されるかもしれない。一旦エントリが削除され、それが再び生成されたならば、監視を始めるためには、監視要求は再度発行されなければならない。表 7-38/JT-T124 にパラメータとこのプリミティブの型を示す。図 7-28/JT-T124 にこのプリミティブの要求型を使用したイベントシーケンスを示す。

（注）特定の GCC プロバイダのインプリメントでは、各エントリのために、アプリケーション プロトコル エンティティが監視を有効あるいは無効とした形跡を保持しないことも可能である。このような場合、監視を要求していないエントリや、監視を無効に設定したエントリについて、アプリケーション プロトコル エンティティが、GCC-REGISTRY-MONITOR-指示を受信することもありえる。

表 7-38/JT-T124 GCC-REGISTRY-MONITOR プリミティブの型とそのパラメータ
(ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID (Conference ID)	M	M(=)	M(=)
有効 / 無効 (Enable/Disable)	M		M(=)
レジストリ キー (Registry Key)	M	M(=)	M(=)
レジストリ 項目 (Registry Item)		M	
所有者 (Owner)		M	
変更権 (Modification Rights)		C	

結果 (Result)			M
-------------	--	--	---

会議 ID (Conference ID): プリミティブが参照する会議識別子。

有効/無効 (Enable/Disable): 真 (TRUE) = レジストリ内容の指示を届ける。偽 (FALSE) = 指示を抑制する。

レジストリ キー (Registry Key): 項目を格納するデータベースのインデックス。

レジストリ項目 (Registry Item): 「チャンネル ID(Channel ID)」、「トークン ID(Token ID)」、「パラメータ値(Parameter Value)」であるか、または、エントリがすでに削除されているなら、「なし(None)」である(型と値の両方(もしあれば)が、このパラメータによって示される)。

所有者 (Owner): このパラメータはこのレジストリ エントリの現在の所有者を示す。もし所有されるなら、このパラメータは所有者が存在するノードの「ノード ID (Node ID)」と所有しているアプリケーション プロトコル エンティティの「エンティティ ID (Entity ID)」を含む。エントリが所有されていない場合、このパラメータは示される。ローカルに生成されたエラー状態の確認なら、このパラメータは存在しない。

変更権 (Modification Rights): このパラメータは、パラメータ エントリの場合にのみ含まれ、このレジストリ エントリの値を変更することを許されたアプリケーション プロトコル エンティティの範囲を示す。このパラメータ値は「所有者 (owner)」、「セッション (session)」、あるいは「パブリック (public)」のいずれかである。「所有者 (owner)」は、その所有者だけがこのエントリを変更してもよい(エントリが所有される限り)ことを規定する。「セッション (session)」は、所有者と同じアプリケーション プロトコル セッションの一部のどのアプリケーション プロトコル エンティティでもこのエントリを変更してもよいことを規定する。「パブリック (public)」は、会議に参加登録したどのアプリケーション プロトコル エンティティでも、このエントリを変更してもよいことを規定する。エントリが最初に生成される時、このパラメータが指定されていないなら、「パブリック (Public)」の値となる。もし確認がローカルに生成されたエラー状態なら、このパラメータは存在しない。

結果 (Result): 要求を受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、次の中のいずれか一つを含む。「成功 (successful)」、「無効な会議 (invalid conference)」、「レジストリ エントリが存在しない (registry entry does not exist)」。

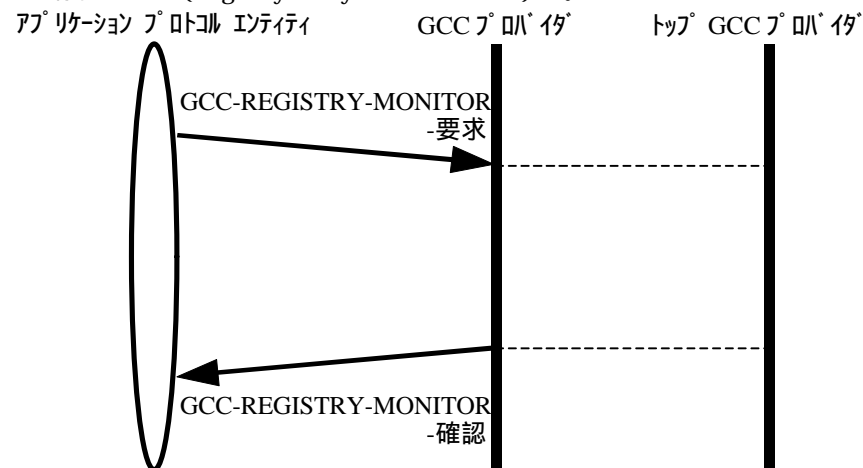


図 7-28/JT-T124 GCC-REGISTRY-MONITOR プリミティブのシーケンス (ITU-T T.124)

7.4.4.7. GCC-REGISTRY-ALLOCATE-HANDLE

GCC-REGISTRY-ALLOCATE-HANDLE-要求プリミティブは、単一会議の範囲内で全体的にユニークな数値 (numerical value) (あるいは値のリスト) がそのアプリケーション プロトコル エンティティに充てられることを要求

するために、アプリケーション プロトコル エンティティによって発行されるかもしれない。ハンドルは、要求が受信された順序で昇順の数値(increasing numerical order)が、トップ GCC プロバイダによって割り当てられる。ブロックのハンドルもまた、昇順(increasing numerical order)で割り当てられる。結果として、「ハンドル数(Number of Handles)」が 1 より大きければ、ブロックの最初のハンドルだけが、返送されることになる。表 7-39/JT-T124 にパラメータとこのプリミティブの型を示す。図 7-29/JT-T124 にこのプリミティブを使用したイベントシーケンスを示す。

表 7-39/JT-T124 GCC-REGISTRY-ALLOCATE-HANDLE プリミティブの型とそのパラメータ (ITU-T T.124)

パラメータ	要求	確認
会議 ID (Conference ID)	M	M(=)
ハンドル数 (Number of Handles)	M	M(=)
最初のハンドル (First Handle)		M
結果 (Result)		M

会議 ID (Conference ID) : プリミティブが参照する会議識別子。

ハンドルの数 (Number of Handles) : 割当を要求したハンドルの数。確認プリミティブでも返送される。1 から 1024 までの値。

最初のハンドル (First Handle) : 32 ビットの符号無し整数(unsigned integer)。要求したハンドルの数が 1 なら、割り当てられたハンドル値である。ハンドル数が 1 以上なら、割り当てられたハンドルは、最初のハンドルの値(モジュロ 2^{32}) から、(最初のハンドル + ハンドル数)の値(モジュロ 2^{32})までの連続する値である。

結果 (Result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、次のうちのいずれか一つを含む。「成功 (successful)」、「無効な会議 (invalid conference)」、「ハンドル利用不可 (no handles available)」、「要求ハンドル過多 (too many handles requested)」。

アプリケーション プロトコル エンティティ GCC プロバイダ トップ GCC プロバイダ

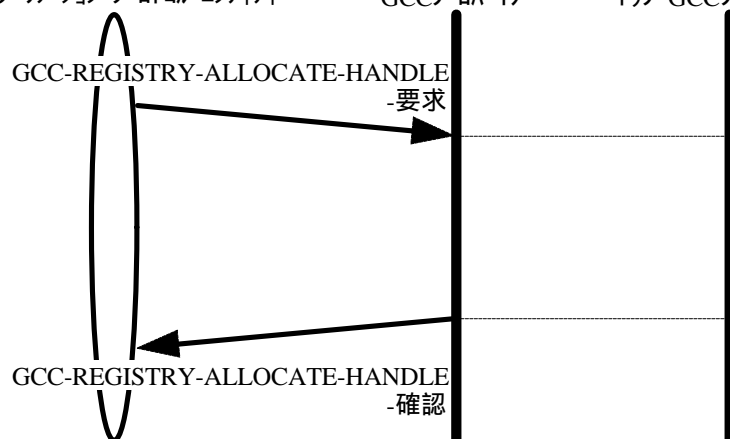


図 7-29/JT-T124 GCC-REGISTRY-ALLOCATE-HANDLE プリミティブのシーケンス (ITU-T T.124)

7.5. 会議議長権

GCC は、ノードが会議の議長になることを許可する方法を提供する。GCC は、会議が議長ありか、議長なしであるか

どうかを決定するためにトークンを使用する。ノードにおけるノードコントローラは、GCC-CONDUCTOR-ASSIGN-要求 プリミティブを発行して議長トークンを取得する。いずれかのノードが議長トークンの取得に成功したとき、その会議は議長制御モードに移行する。会議が議長制御モードに切り替わるとき、その会議中の全ノードでのノードコントローラと全ての登録されたアプリケーション プロトコル エンティティは GCC-CONDUCTOR-ASSIGN-指示によって、その切り替えが通知される。この指示はまた、どのノードが議長になったのかも示す。

特定の会議が議長制御モードで動作してもよいかどうかは、その会議を生成するときに決定される。議長制御可能会議フラグがセットされていたならば、その会議は、議長制御モードに移行できる。そうでなければ、議長制御モードにするためのいかなる試みも拒否されるであろう。

ノードは、GCC-CONDUCTOR-RELEASE-要求 プリミティブを発行することによって、議長権を解放し、その会議を非議長制御モードへ戻す。会議が非議長制御モードへ切り替わるとき、その会議中の全ノードでのノードコントローラと全ての登録されたアプリケーション プロトコル エンティティは GCC-CONDUCTOR-RELEASE-指示によって、その切り替えが通知される。

現在会議の議長であるノードは、希望する議長権受け取りノードを指定した GCC-CONDUCTOR-GIVE-要求を発行することによって、その遷移期間中、会議を非議長制御モードに移行することなく、他のノードに直接議長権を渡す。受け取りノードが議長権を受け入れるならば、その会議中の全ノードでのノードコントローラと全ての登録されたアプリケーション プロトコル エンティティは、どのノードが新しい議長となるかを指示する GCC-CONDUCTOR-ASSIGN-指示によって、この遷移が通知される。ノードは、GCC-CONDUCTOR-PLEASE-要求 プリミティブを発行することによって、議長からの議長権の譲渡を明示的に要求する。現在の議長は、要求ノードに議長権を譲渡するか、または要求を無視するかを選択する。

また、その会議中の全ノードでのノードコントローラと全ての登録されたアプリケーション プロトコル エンティティは、現在どのノードが議長トークンを保持しているかを GCC-CONDUCTOR-INQUIRE-要求 プリミティブを使用して尋ねることができる。

会議に参加しているアプリケーション プロトコル エンティティが議長制御モードであることを通知されると、直ちに対応するアプリケーションプロトコルの仕様によって規定された議長制御モードの動作に従って動作し始めるであろう。逆に、会議に参加しているアプリケーション プロトコル エンティティが非議長制御モードであることを通知されると、直ちに対応するアプリケーションプロトコルの仕様によって規定された非議長制御モードの動作に従って動作し始めるであろう。例えば、典型的なアプリケーション プロトコル仕様では、議長制御モードにあるどのノードのアプリケーション プロトコル エンティティも、いかなる動作を行なう前に議長の同位アプリケーション プロトコル エンティティからの許可を要求しなければならないのに対し、非議長制御モードにおいては、そのような許可は不必要である、と記述するであろう。しかしながら、議長制御モードあるいは非議長制御モード中のアプリケーション プロトコル仕様によって定義される動作は、この標準の範囲外である。

7.5.1. サービスの概要記述

以下のリストは、この節の中で定義されたプリミティブと各々の機能の簡単な概要を記述している。

GCC-CONDUCTOR-ASSIGN - 会議の議長権を要求するために、ノード コントローラによって使用される。新しいノードが議長になるとき、その会議中の全ノードでのノードコントローラと全ての登録されたアプリケーション プロトコル エンティティに、このことを通知するために、このプリミティブの指示形式が使用される。

GCC-CONDUCTOR-RELEASE - 会議の議長権を解放するために、ノード コントローラによって使用される。いずれかのノードが議長権を解放するとき、その会議中の全ノードでのノードコントローラと全ての登録されたアプリケ

ーション プロトコル エンティティに、このことを通知するために、このプリミティブの指示形式が使用される。

GCC-CONDUCTOR-PLEASE - 議長権を現在の議長から譲渡してもらうことを要求するためにノード コントローラによって使用される。

GCC-CONDUCTOR-GIVE - 指定されたノードに議長権を譲渡するためにノード コントローラによって使用される。

GCC-CONDUCTOR-INQUIRE - 現在、会議が議長制御モードであるのか否かを確認するために、また、もし議長制御モードであるならば、現在の議長のノード ID を確認するために、ノードコントローラがアプリケーション プロトコル エンティティによって使用される。

GCC-CONDUCTOR-PERMISSION-ASK - 議長制御モード中において、同じノードのアプリケーション プロトコル エンティティのために、その動作の許可を議長に要求するためにノードコントローラによって使用される。

GCC-CONDUCTOR-PERMISSION-GRANT - 会議中のどのノードに議長制御モード許可が認められているのかを指示するために、議長ノードのノードコントローラによって使用される。

7.5.1.1. GCC-CONDUCTOR-ASSIGN

非議長制御モードの間、どのノードのノードコントローラも、GCC-CONDUCTOR-ASSIGN-要求 プリミティブを発行することによって、議長になるための要求を発行することができる。受信した GCC-CONDUCTOR-ASSIGN-確認 プリミティブが、そのプリミティブ内の「結果(Result)」パラメータに基づいて、その要求者が議長になったかどうかを示す。成功である GCC-CONDUCTOR-ASSIGN-要求により、会議が議長制御モードになったことを通知するとともに、議長の識別子情報を含んだ GCC-CONDUCTOR-ASSIGN-指示がその会議中の全ノードでのノードコントローラと全ての登録されたアプリケーション プロトコル エンティティへ発行される。また、このプリミティブの指示形式は、会議が議長制御モードになることを知らせるために会議へ新たに登録しているアプリケーション プロトコル エンティティにも発行される。GCC-CONDUCTOR-ASSIGN-指示と GCC-CONDUCTOR-RELEASE-指示の順序は、議長権移行の実際の順序を表す。表 7-40/JT-T124 に、このプリミティブの型とパラメータを示す。図 7-30/JT-T124 に、このプリミティブ を使用する際のイベントのシーケンスを示す。

注 - 議長が何らかの理由により会議から切断するならば、別のノードが議長になるための要求を出すまで、その会議は非議長制御モードに戻る。

表 7-40/JT-T124 GCC-CONDCUTOR-ASSIGN プリミティブの型とパラメータ

(ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
要求ノード ID(Requesting Node ID)		M	
結果(Result)			M

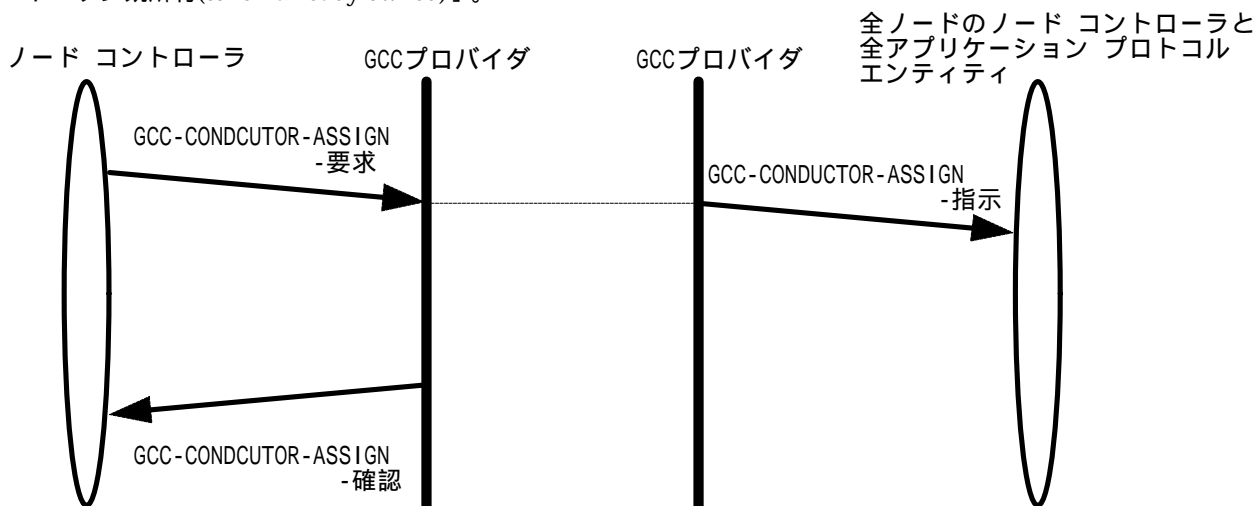
会議 ID(Conference ID) : このプリミティブが参照する会議の識別子。

要求ノード ID(Requesting Node ID) : 要求ノードのノード ID。

結果(Result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。

「成功(successful)」、「無効な会議(invalid conference)」、「議長制御が不可能な会議(non-conductible conference)」

「トークン既所有(token already owned)」。



図

7-30/JT-T124 GCC-CONDUCTOR-ASSIGN プリミティブのシーケンス

(ITU-T T.124)

7.5.1.2. GCC-CONDUCTOR-RELEASE

議長から一般の参加者への状態変更のために、議長ノードでのノードコントローラは、GCC-CONDUCTOR-RELEASE-要求 プリミティブを発行する。議長が議長権の解放を要求したらすぐに、その会議中の全ノードでのノードコントローラとすべての登録されたアプリケーションプロトコルエンティティは、GCC-CONDUCTOR-RELEASE-指示によって動作モードの変更を通知される。GCC-CONDUCTOR-ASSIGN-指示とGCC-CONDUCTOR-RELEASE-指示の順序は、議長権移行の実際の順序を表す。GCC-CONDUCTOR-RELEASEを発行している参加者が、現在の議長でないなら、否定の結果と理由の記述を含むGCC-CONDUCTOR-RELEASE-確認がGCCより返される。この状況では、指示は他の参加者へは送られない。また、このユーザが非議長制御モードへの移行を始める以外に、例えば議長が会議から切断したことによる議長権の解放もGCC自身によって始めることができる。さらに、このプリミティブの指示形式は、会議が非議長制御モードになることを知らせるために会議へ新たに登録しているアプリケーションプロトコルエンティティにも発行される。表 7-41/JT-T124 に、このプリミティブの型とパラメータを示す。両方の状況に属しているプリミティブのシーケンスを、それぞれ図 7-31/JT-T124 と図 7-32/JT-T124 に示す。

表 7-41/JT-T124 GCC-CONDUCTOR-RELEASE プリミティブの型とパラメータ

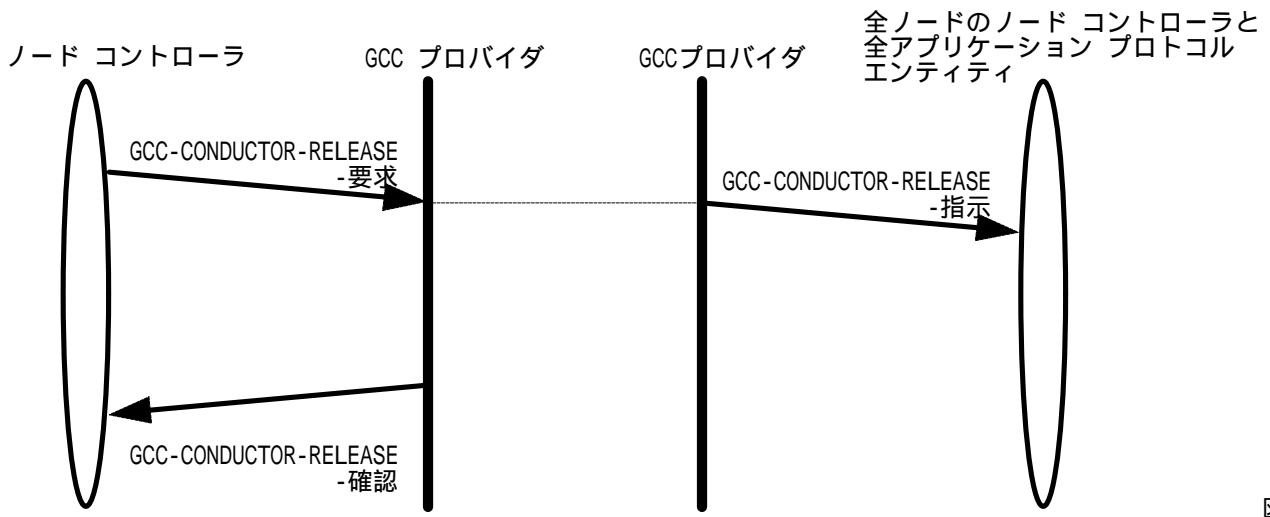
(ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
結果(Result)			M

会議 ID(Conference ID)：このプリミティブが参照する会議の識別子。

結果(Result)：要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。

「成功(successful)」、「無効な会議(invalid conference)」、「トークン非所有(did not own token)」。



7-31/JT-T124 GCC- CONDUCTOR-RELEASE(ユーザ要求)プリミティブのシーケンス (ITU-T T.124)

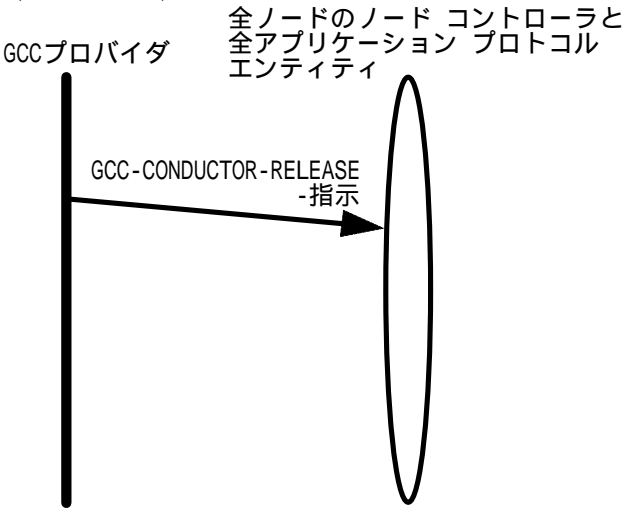


図 7-32/JT-T124 GCC- CONDUCTOR-RELEASE (GCC 要求)プリミティブのシーケンス (ITU-T T.124)

7.5.1.3. GCC-CONDUCTOR-PLEASE

GCC-CONDUCTOR-PLEASE-要求 プリミティブは、現在の議長に対し議長権の譲渡を依頼するためにノード コントローラによって発行される。GCC-CONDUCTOR-PLEASE-指示は、現在の議長ノードへ転送される。そのとき、現在の議長は、GCC-CONDUCTOR-GIVE プリミティブを使用することによってその要求者に議長権を譲渡することを選択するかもしれない。このプリミティブの確認は、要求がローカルの GCC プロバイダによって受け入れられたローカルな確認にすぎない。議長からの確認は、直接与えられない。表 7-42/JT-T124 に、このプリミティブの型とパラメータを示す。図 7-33/JT-T124 に、このプリミティブを使用する際のイベントのシーケンスを示す。

表 7-42/JT-T124 GCC-CONDUCTOR-PLEASE プリミティブの型とパラメータ (ITU-T T.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
要求ノード ID(Requesting Node)		M	

ID)			
結果(Result)			M

会議 ID(Conference ID)：このプリミティブが参照する会議の識別子。

要求側ノード ID(Requesting Node ID)：要求ノードのノード ID。

結果(Result)：要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。

「成功(successful)」、「無効な会議(invalid conference)」、「議長制御モードでない(not in conducted mode)」。

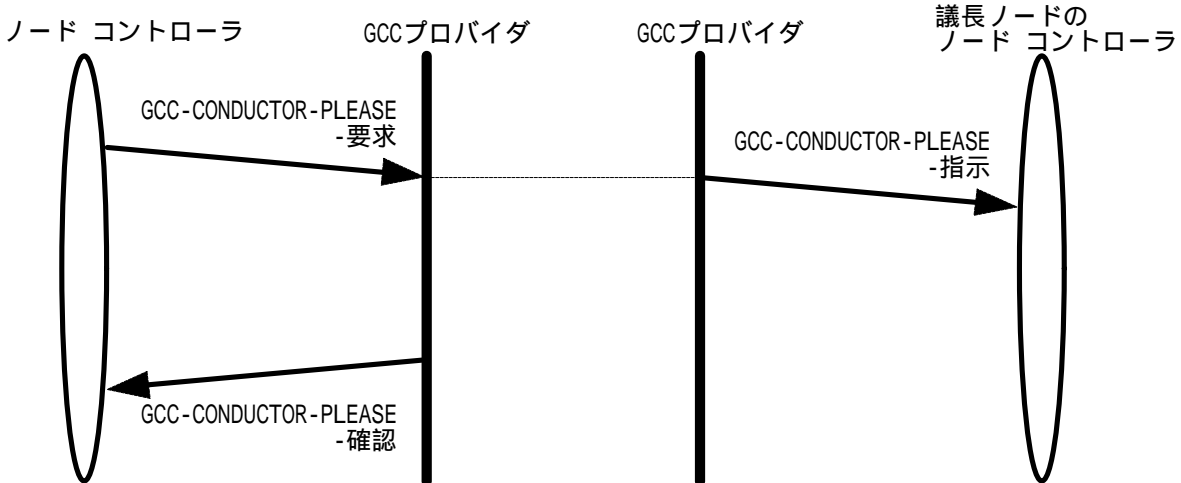


図 7-33/JT-T124 GCC-CONDUCTOR-PLEASE プリミティブのシーケンス (ITU-T T.124)

7.5.1.4. GCC-CONDUCTOR-GIVE

GCC-CONDUCTOR-GIVE-要求 プリミティブは、特定のノードへ議長権を移すためにノードコントローラによって発行される。その議長権が予定された受け入れノードによって受け入れられないならば、会議議長権は、もとの議長によって保持され続ける。表 7-43/JT-T124 に、このプリミティブの型とパラメータを示す。図 7-34/JT-T124 に、このプリミティブを使用する際のイベントのシーケンスを示す。

注 - 他のノードに会議議長になる機会を与えることなく、特定のノードが議長権を取得することを保証することは、このプリミティブ(他のプリミティブと共に)を利用することで可能である。その手順は、会議召集者がまず、ロックされた会議を生成する。それから GCC-CONDUCTOR-ASSIGN プリミティブを使用して議長権を取得する。その召集者が会議議長になったなら、それが GCC-CONFERENCE-UNLOCK プリミティブを使用してロックを解除することによって、あるいは、他のノードを直接付加することによって、他のノードの会議への参加を許す。予定していた議長ノードがその会議に参加したならば、議長権は GCC-CONDUCTOR-GIVE プリミティブによってそのノードに直接渡される。

表 7-43/JT-T124 GCC-CONDUCTOR-GIVE プリミティブの型とパラメータ (ITU-T T.124)

パラメータ	要求	指示	応答	確認
会議 ID(Conference ID)	M	M	M(=IN)	M(=RQ)
受け入れノード ID(Recipient Node ID)	M			M(=)

結果(Result)			M	M(=)
------------	--	--	---	------

会議 ID(Conference ID) : このプリミティブが参照する会議の識別子。

受け入れノード ID(Recipient Node ID) : 議長権が移行されているノードのノード ID。

結果(Result) : 要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。

「成功(successful)」、「無効な会議(invalid conference)」、「議長でない(not conductor)」、「受け入れてないのに譲渡された(give not accepted)」。

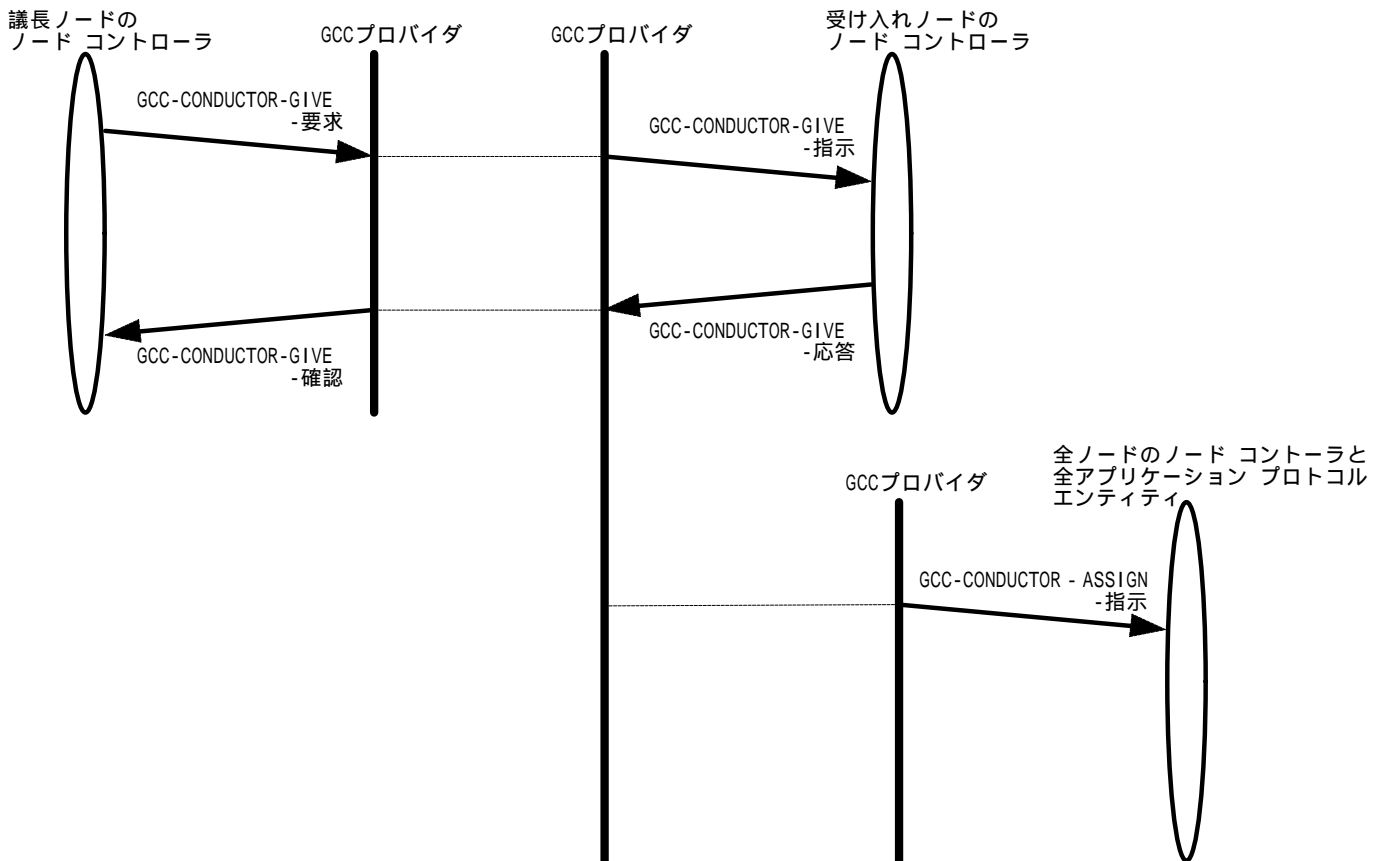


図 7-34/JT-T124 GCC-CONDUCTOR-GIVE プリミティブのシーケンス (ITU-T T.124)

7.5.1.5. GCC-CONDUCTOR-INQUIRE

GCC-CONDUCTOR-INQUIRE-要求 プリミティブが、その会議が議長有りかそうでないか、有りであるなら、どのノードがその議長であるか、そして、要求ノードが、議長制御モード許可を与えられているかを見つけるために、いつでも、ノードコントローラかアプリケーション プロトコル エンティティによって発行される。表 7-44/JT-T124 に、このプリミティブの型とパラメータを示す。図 7-34/JT-T124 に、このプリミティブを使用する際のイベントのシーケンスを示す。

表 7-44/JT-T124 GCC-CONDUCTOR-INQUIRE プリミティブの型とパラメータ

パラメータ	要求	確認
-------	----	----

会議 ID(Conference ID)	M	M(=)
議長制御 / 非議長制御フラグ (Conducted vs. Non-Conducted Flag)		M
議長ノード ID (Conductor Node ID)		C
許可フラグ(Permission flag)		C
結果(Result)		M

会議 ID(Conference ID)：このプリミティブが参照する会議の識別子。

議長制御 / 非議長制御フラグ(Conducted vs. Non-Conducted Flag)：指示された会議が、現在議長制御モードか非議長制御モードであるかどうかを示すフラグ。

議長ノード ID(Conductor Node ID)：現在議長であるノードのノード ID。現在非議長制御モード中であれば、存在しない。

許可フラグ(Permission flag)：議長制御モード中であるならば、このフラグは、ローカルなノードが議長制御モード許可を与えられたかどうかを示す。

結果(Result)：要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。

「成功(successful)」、「無効な会議(invalid conference)」。

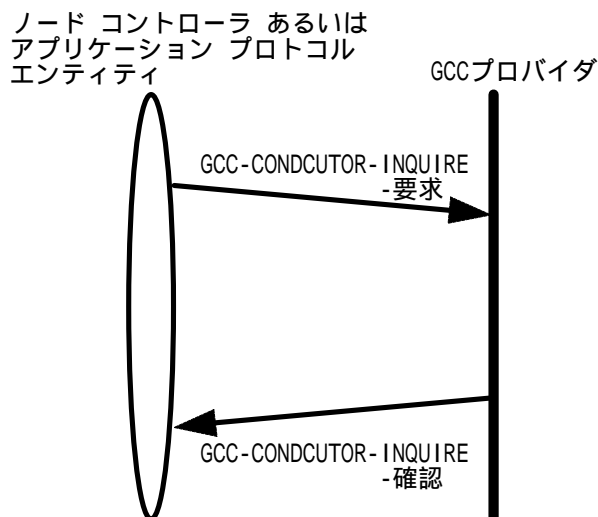


図 7-35/JT-T124 GCC-CONDUCTOR-INQUIRE プリミティブのシーケンス

7.5.1.6. GCC-CONDUCTOR-PERMISSION-ASK

GCC-CONDUCTOR-PERMISSION-ASK-要求 プリミティブは、要求ノードでのアプリケーション プロトコル エンティティがいかなる動作を実行するのに、現在の議長に許可を問い合わせるためにノードコントローラによって発行される。このような許可を必要とする特定の動作の定義は、定義する個々のアプリケーション プロトコル仕様の問題である。GCC-CONDUCTOR-PERMISSION-ASK-指示は、現在の議長ノードの方へ転送される。一つのノードからのGCC-CONDUCTOR-PERMISSION-ASK-指示の順序は、そのノードから実際に出した要求の順序である。それから、

現在の議長は、GCC-CONDUCTOR-PERMISSION-GRANT プリミティブを使用して、要求ノードに対し会議制御モード許可を与えるか（あるいは解放するか）、あるいは要求を無視するかを選択できる。この確認プリミティブは、要求がローカルな GCC プロバイダによって受け入れられたことのローカルに確認にすぎない。議長からの確認は、直接与えられない。表 7-45/JT-T124 に、このプリミティブの型とパラメータを示す。図 7-36/JT-T124 に、このプリミティブを使用する際のイベントのシーケンスを示す。

表 7-45/JT-T124 GCC-CONDUCTOR-PERMISSION-ASK プリミティブの型とパラメータ (ITU-TT/124)

Parameter	要求	指示	確認
	会議 ID(Conference ID)	M	M
譲渡/解放フラグ(Grant/Release Flag)	M	M(=)	M(=)
要求ノード ID(Requesting Node ID)		M	
結果(Result)			M

会議 ID(Conference ID)：このプリミティブが参照する会議の識別子。

譲渡/解放フラグ(Grant/Release Flag)：このフラグは、要求者が議長制御モード許可を与えられるのを希望するかどうか、あるいは、要求者が議長制御モード許可を解放するのを希望するかどうかを示す。

要求ノード ID(Requesting Node ID)：要求ノードのノード ID。

結果(Result)：要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。

「成功(successful)」、「無効な会議(invalid conference)」、「議長制御モードでない(not in conducted mode)」。

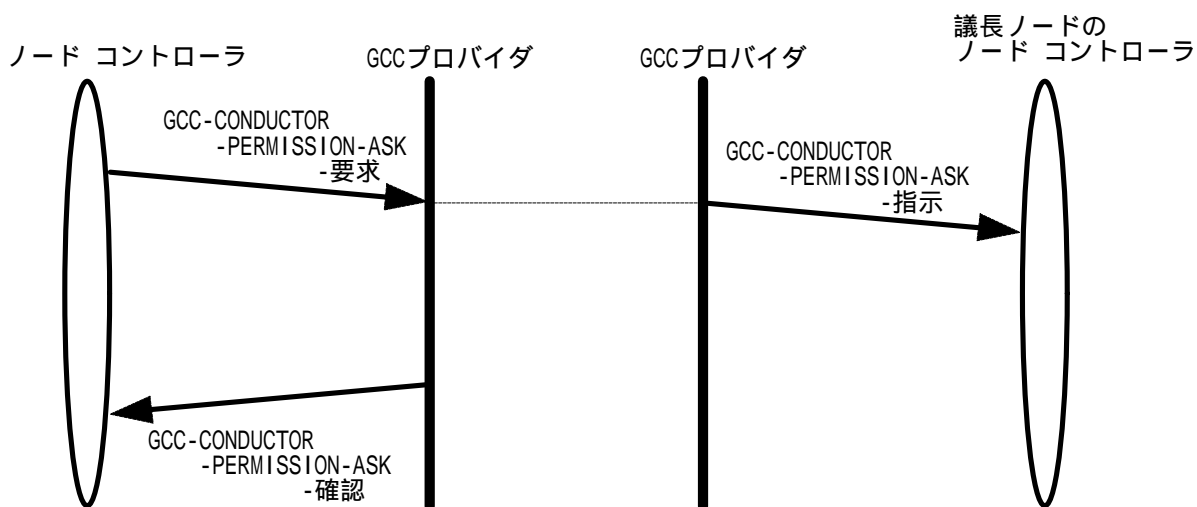


図 7-36/JT-T124 GCC-CONDUCTOR-PERMISSION-ASK プリミティブのシーケンス

7.5.1.7. GCC-CONDUCTOR-PERMISSION-GRANT

GCC-CONDUCTOR-PERMISSION-GRANT-要求 プリミティブは、議長ノードのノードコントローラによって、会議中に一つ以上のノードから議長制御モード許可を与えるか、あるいは無効にするために発行される。対応する GCC-CONDUCTOR-PERMISSION-GRANT-指示が、その会議中のあらゆるノードへ放送され、どのノードが、現在議長制御モード許可を持っているか、また要求したがまだ許可を待っているかを示す。議長が許可を与える順序に従ってリストが作成される。指示は、各ノードでのノードコントローラとすべての登録されたアプリケーション プロトコル エンティティに与えられる。複数の GCC-CONDUCTOR-PERMISSION-GRANT-指示の順番は、議長から出された実際の要求の順番を示す。すなわち、最も最近に受信した指示プリミティブが有効となる。議長制御モード許可は、典型的には GCC-CONDUCTOR-PERMISSION-ASK に対する応答内で与えられるが、また、その議長によって自発的に与えられるかもしれない。議長ノード自身は、許可を与えられたノードのリストに明示的に記載されているかどうかは別にして、議長制御モード許可を持つ。会議が議長制御モードになるとき、最初は何のノードも許可を持っていない。例えば、その前の会議が議長制御モードで行われたとしてもである。議長権の所有ノードが GCC-CONDUCTOR-GIVE シーケンスにより替わるならば、許可の状態は元の議長によって最後に放送されたときのままとなる。新しいノードが会議に加わるとき、そのノードは許可を持っていないとみなされる。この場合、議長ノードでのノードコントローラは、新しいノードに会議中の他のノードの許可状態を知らせるために、GCC-CONDUCTOR-PERMISSION-GRANT-要求を再送信することによって許可リストを再放送する。この確認プリミティブは、要求がローカルな GCC プロバイダによって受け入れられたことのローカルな確認にすぎない。表 7-46/JT-T124 に、このプリミティブの型とパラメータを示す。図 7-37/JT-T124 に、このプリミティブを使用する際のイベントのシーケンスを示す。

表 7-46/JT-T124 GCC-CONDUCTOR-PERMISSION-GRANT プリミティブの型とパラメータ (ITU-TT.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
許可を与えられたノードのリスト (List of Nodes Granted Permission)	M	M(=)	
許可を待っているノードのリスト (List of Nodes Waiting for Permission)	O	O(=)	
許可フラグ(Permission Flag)		M	
結果(Result)			M

会議 ID(Conference ID)：このプリミティブが参照する会議の識別子。

許可を与えられたノードのリスト(List of Nodes Granted Permission)：ノード ID のリスト。議長が議長制御モード許可を与えた各ノードのリスト。このリストが空であるならば、どのノードも議長制御モード許可を持たない。

許可を待っているノードのリスト(List of Nodes Waiting for Permission)：ノード ID の順番リスト。議長によって、議長制御モード許可を与えられることを考慮されているが、それをまだ与えられていない各ノードのリスト。議長は、許可を与える順序に従ってリストを作成するであろう。そうであるならば、そのリストの中の 1 番目のノードは、許可を受ける次のノードとみなされる。

許可フラグ(Permission Flag)：このフラグは、ローカルなノードが許可を与えられたノードのリストに載っているか否かを示す。これにより、アプリケーション プロトコル エンティティは、送られて来たリストを調査する必要なしに動作モードを容易に決定することができる。

結果(Result)：要求が受け入れられたか、拒絶されたかどうか、もし拒絶されたなら、その理由を示す。それは、以下のリスト中の可能な結果の一つを含む。

「成功(successful)」、「無効な会議(invalid conference)」、「議長でない(not conductor)」。

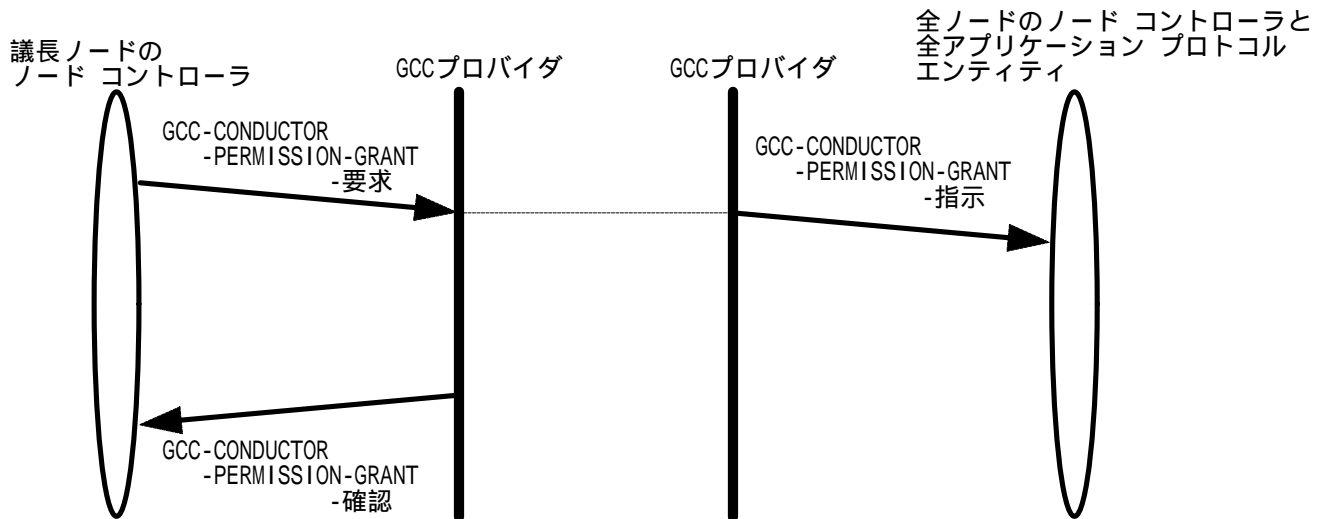


図 7-37/JT-T124 GCC-CONDUCTOR-PERMISSION-GRANT プリミティブのシーケンス

7.6. その他の機能

7.6.1. サービスの概要記述

次に掲げるものはこの章で定義されるプリミティブとそれぞれの機能の簡単な要約である。

- GCC-CONFERENCE-TIME-REMAINING - 会議召集者のノードコントローラがすべてのノードに対して時間制約のある会議の終了までの残り時間を通知することを可能にする。
- GCC-CONFERENCE-TIME-INQUIRE- 任意のノードのノードコントローラが時間制約のある会議の残り時間を知ることを可能にする。
- GCC-CONFERENCE-EXTEND- 任意のノードのノードコントローラが会議の召集者に対して時間制約のある会議を割り当てられた時間を超えて延長するよう要求することを可能にする。
- GCC-CONFERENCE-ASSISTANCE- 会議オペレータから非定型のアシスタンスを要求するために使われる。
- GCC-TEXT-MESSAGE - 任意のテキストメッセージをほかのノードに対して送り、それらのノードのユーザに対して表示するために使われる。

7.6.1.1. GCC-CONFERENCE-TIME-REMAINING

GCC-CONFERENCE-TIME-REMAINING-要求プリミティブはノードコントローラがすべてのノードに対して、時間制約のある会議の残り時間を通知するために使用することができる。また、すべてのノードではなく、特定のノードに対しての残り時間を通知するためにも使用できる。この要求は、召集者によって発行されることを意図しているが、同様にほかのノードによって発行されてもよい。典型的には、このプリミティブは会議の終わり近くにおいて、会議がほとんど終わろうとしていることを通知するために一度発行される。表 7-47/JT-T124 にこのプリミティブの型とパラメ

ータを示す。図 7-38/JT-T124 にこのプリミティブを用いたときのイベントのシーケンスを示す。

表 7-47/JT-T124- GCC-CONFERENCE-TIME-REMAINING- プリミティブの型とパラメータ (ITU-TT.124)

パラメータ	要求	指示	確認
会議 ID(Conferencing ID)	M	M	M(=RQ)
残り時間(Time Remaining)	M	M(=)	
ノード ID(Node ID)	O	O(=)	
ソースノード ID(Source Node ID)		M	
結果(Result)			M

会議 ID: プリミティブが参照する会議の識別子

残り時間: 秒単位で表された会議の残り時間を示す。

ノード ID: オプションパラメータもし存在すれば、指示された残り時間は特定のノードに対してのみ適用されることを示す。もしなければ、残り時間は会議のすべてのノードに適用される。

ソースノード ID: 要求を発行したノードのノード ID。

結果(Result): 要求が受諾されたか拒絶されたかを示す。もし拒絶されたならその理由を示す。それは以下のリスト中の可能な結果の1つを含む。「成功(successful)」、「無効な会議(invalid conference)」。

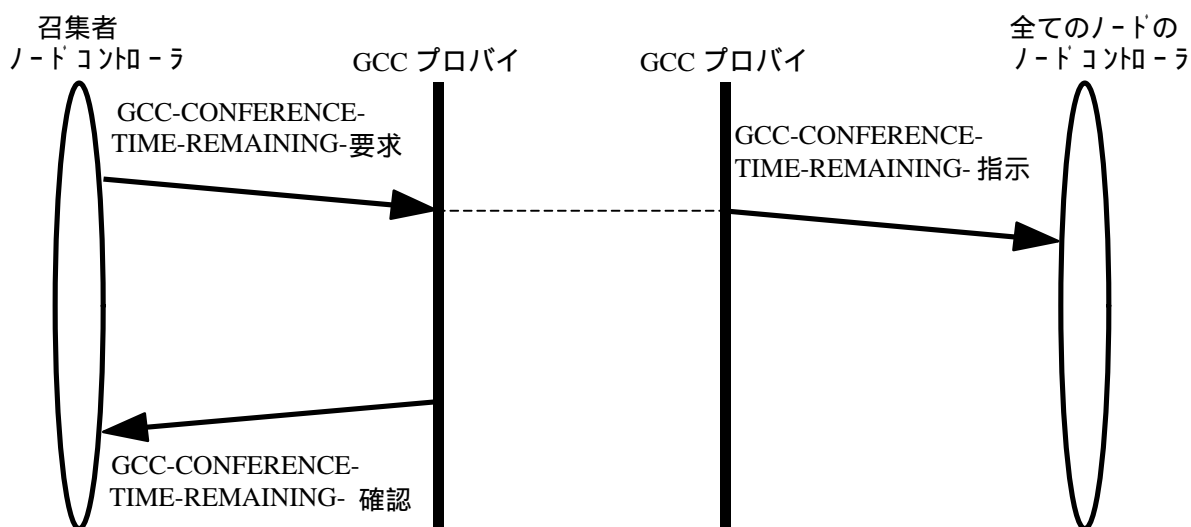


図 7-38/JT-T124 - GCC-CONFERENCE-TIME-REMAINING- プリミティブのシーケンス

7.6.1.2. GCC-CONFERENCE-TIME-INQUIRE

GCC-CONFERENCE-TIME-INQUIRE-要求プリミティブは、任意のノードが、会議の召集者から会議において残り時間がどれくらいか知るために使われる。会議の召集者がこのプリミティブの通知を受け取ると（もし召集者がこのプリミティブをサポートしていれば）、GCC-CONFERENCE-TIME-REMAINING プリミティブを使って、会議の残り時間をすべてのノードに対して放送する。表 7-48/JT-T124にこのプリミティブの型とパラメータを示す。図 7-39/JT-T124 にこのプリミティブを使ったときのイベントのシーケンスを示す。

表 7-48/JT-T124 – GCC-CONFERENCE-TIME-INQUIRE-プリミティブの型とパラメータ (ITU-TT.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
会議全体または特定ノードのフラグ (Conference-Wide vs. Node-Specific Time Flag)	M	M(=)	
要求ノード ID(Requesting Node ID)		M	
結果(Result)			M

会議 ID: プリミティブが参照する会議の識別子

会議全体または特定ノードフラグ: このフラグは要求が会議全体の残り時間か、もし違うのならば要求しているノードの残り時間を知らうとしているのを示す。もし、会議の召集者が会議全体の終了時間のみを考慮しているのならばこのフラグは無視される。

要求ノード ID: 要求しているノード ID .

結果(Result): 要求が受諾されたか拒絶されたかを示す。もし拒絶されたならその理由を示す。それは以下のリスト中の可能な結果の1つを含む。「成功(successful)」、「無効な会議(invalid conference)」。

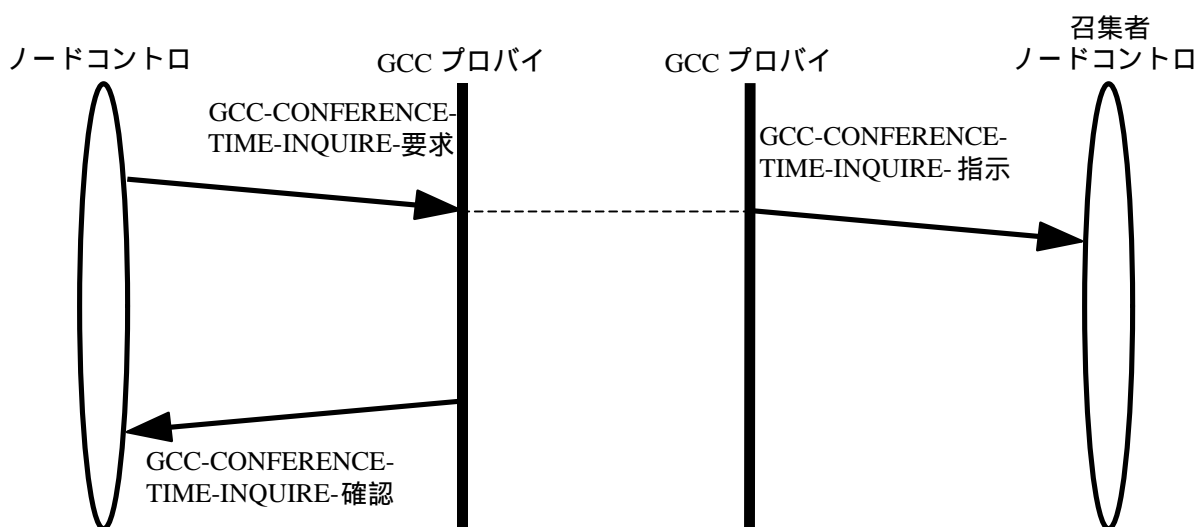


図 7-39 /JT-T124- GCC-CONFERENCE-TIME-INQUIRE-プリミティブのシーケンス

7.6.1.3. GCC-CONFERENCE-EXTEND

GCC-CONFERENCE-EXTEND-要求プリミティブはあるノードのノードコントローラが会議の召集者に対して時間制約のある会議の時間の延長を要求するために使われる。会議の召集者がこのプリミティブの通知を受け取ると（もし召集者がこのプリミティブをサポートしていれば）GCC-CONFERENCE-TIME-REMAINING プリミティブを使用してすべてのノードに対して会議の新しい残り時間を（実際に延長されていない場合でも）放送する。召集者が実際に残り時間に追加した時間と要求された時間は一致しなくてもよい。表 7-49/JT-T124 にこのプリミティブの型とパラメータを示す。図 7-40/JT-T124 にこのプリミティブを用いたときのイベントのシーケンスを示す。

表 7-49/JT-T124 – GCC-CONFERENCE-EXTEND -プリミティブ型とパラメータ
(ITU-TT.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
要求された時間 (Time Requested)	M	M(=)	M(=)
会議全体または特定ノードのフラグ (Conference-Wide vs. Node-Specific Time Flag)	M	M(=)	
要求ノード ID(Requesting Node ID)		M	
結果(Result)			M

会議 ID: プリミティブが参照する会議の識別子

要求された時間: 要求と指示プリミティブの中において、秒単位で表された希望延長時間を示す。

会議全体または特定ノードフラグ: このフラグは要求が会議全体の残り時間か、もしちがうのならば要求しているノードの残り時間を知ろうとしているのを示す。もし、会議の召集者が会議全体の終了時間のみを考慮しているのならばこのフラグは無視される。

要求ノード ID:要求しているノード ID .

結果(Result) : 要求が受諾されたか、拒絶されたかを示す。もし拒絶されたなら、その理由を示す。それは以下のリスト中の可能な結果の1つを含む。「結果は成功(successful)」、「無効な会議(invalid conference)」、「議長でない(not conductor)」。

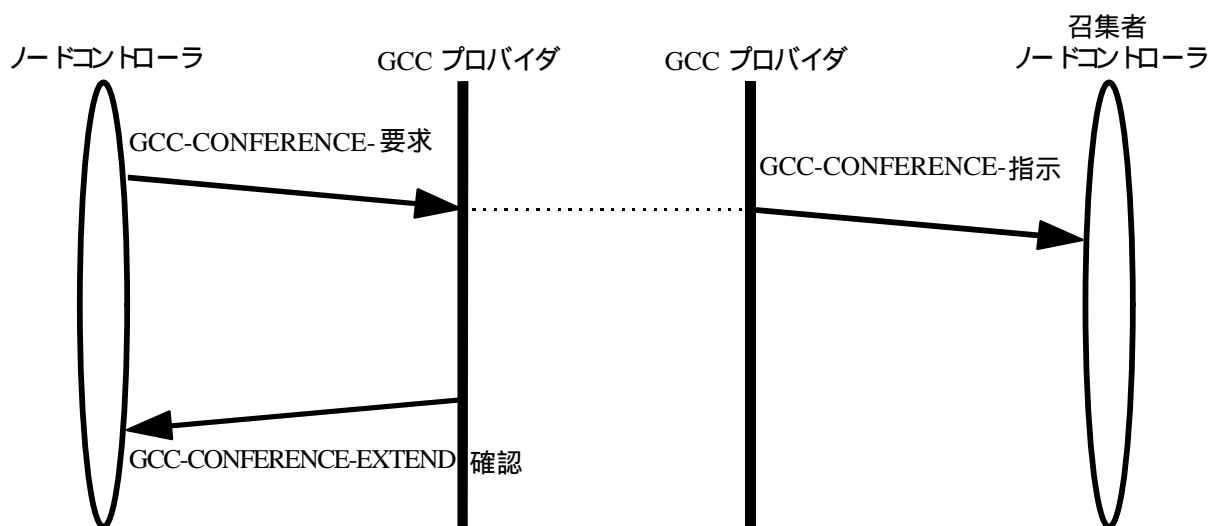


図 7-40/JT-T124 – GCC-CONFERENCE-EXTEND- プリミティブのシーケンス

7.6.1.4. GCC-CONFERENCE-ASSISTANCE

GCC-CONFERENCE-ASSISTANCE プリミティブは会議オペレータによる何らかのアシスタンスを要求するための簡単な手順を提供する。GCC-CONFERENCE-ASSISTANCE-要求プリミティブを発行すると、このプリミティブをサ

ポートしているすべてのノードのノードコントローラに対して GCC-CONFERENCE-ASSISTANCE-指示が放送される。このプリミティブに対するレスポンスは規定されておらず、またこの標準の範囲外である。表 7-50/JT-T124 にこのプリミティブの型とパラメータを示す。図 7-50/JT-T124 にこのプリミティブを用いたときのイベントのシーケンスを示す。

表 7-50 /JT-T124- GCC-CONFERENCE-ASSISTANCE- プリミティブの型とパラメータ (ITU-TT.124)

パラメータ	要求	指示	確認
会議 ID	M	M	M(=RQ)
ユーザデータ	O	O(=)	
ソースノード ID		M	
結果			M

会議 ID: プリミティブが参照する会議の識別子

ユーザデータ: 任意のデータ.

ソースノード ID: **GCC-CONFERENCE-ASSISTANCE** を要求したソースノードのノードID.

結果(Result): 要求が受諾されたか拒絶されたかを示す。もし拒絶されたならその理由を示す。それは以下のリスト中の可能な結果の1つを含む。「成功(successful)」、「無効な会議(invalid conference)」。

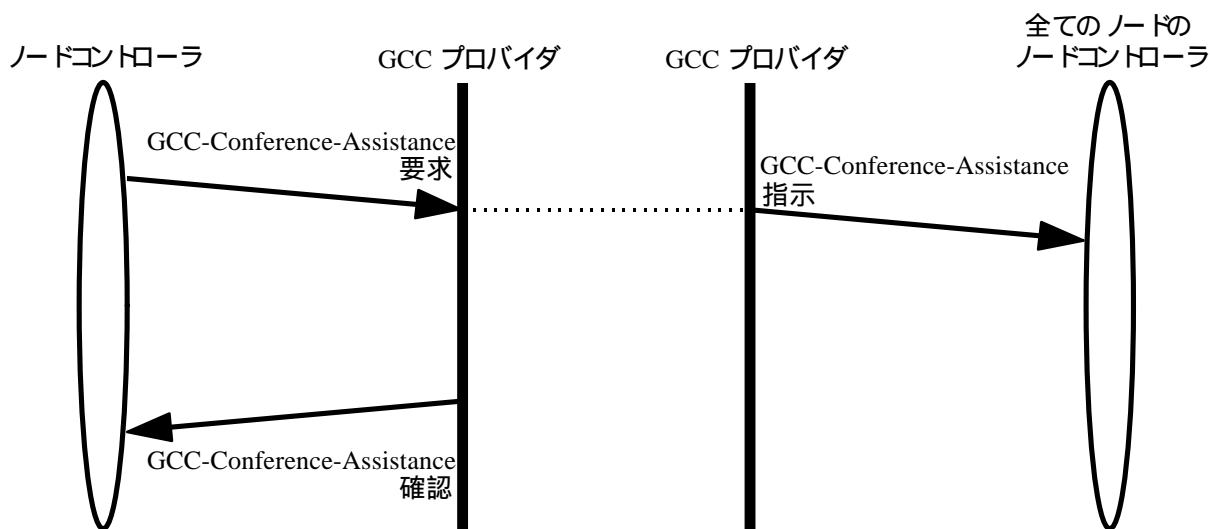


図 7-41/JT-T124 - GCC-CONFERENCE-ASSISTANCE- プリミティブのシーケンス

7.6.1.5. GCC-TEXT-MESSAGE

GCC-TEXT-MESSAGE プリミティブは任意のテキストメッセージを送受するための簡単な方法を提供する。GCC-TEXT-MESSAG 要求を発行すると GCC-TEXT-MESSAG-指示がこのプリミティブをサポートする会議のすべてのノードコントローラに対して放送されるか、あるいは、単一のノードに対して送られる。このプリミティブに対するレスポンスは、その会議の参加者に対して、この標準に規定されていない何らかの方法によってテキストメッセージを表示することである。表 7-51/JT-T124 にこのプリミティブの型とパラメータを示す。図 7-42/JT-T124 にこのプリミティブを用いたときのイベントのシーケンスを示す。

表 7-51/JT-T124 – GCC-TEXT-MESSAGE- プリミティブの型とパラメータ
(ITU-TT.124)

パラメータ	要求	指示	確認
会議 ID(Conference ID)	M	M	M(=RQ)
テキストメッセージ (Text Message)	M	M(=)	
相手先ノード ID(Destination Node ID)	O		
ソースノード ID(Source Node ID)		M	
結果(Result)			M

会議 ID: プリミティブが参照する会議の識別子

テキストメッセージ: 「ユニコード(Unicode)」 テキストメッセージ。

相手先ノード ID: このテキストメッセージを受け取るべき単一のノードのノード ID。もしノードが決められていなければ、メッセージはその会議のすべてのノードに放送される。

ソースノード ID: GCC-TEXT-MESSAGE のソースのノード ID

結果(Result): 要求が受諾されたか拒絶されたかを示す。もし拒絶されたならその理由を示す。それは以下のリスト中の可能な結果の1つを含む。「成功(successful)」、「無効な会議(invalid conference)」。

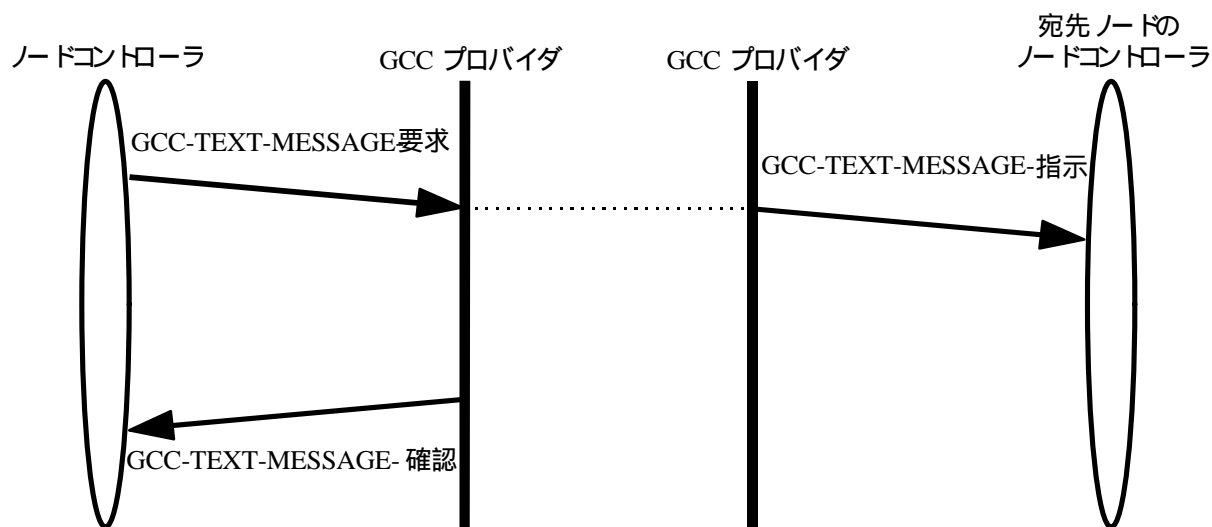


図 7-42/JT-T124 – GCC-TEXT-MESSAGE- プリミティブのシーケンス

8. GCC プロトコル仕様

8.1. 一般操作

いかなるノードにおける GCC プロバイダも、MCS コントロールアプリケーションであり、コントロール MCSAP を介して MCS と通信を行う。初期化に際し、GCC プロバイダは、ローカルな手段でコントロール MCSAP を用いて、MCS と通信接続を確立する。

いかなる場合も、会議は、MCS ドメインを生成することにより確立される。会議は、一つの MCS ドメインと、1 対 1 に対応する。MCS ドメインは、現実にはローカルな手段で生成される。会議は、MCS-CONNECT-PROVIDER-プリミティブを使用して、生成もしくは加入される。

特定の会議（会議 ID（Conference ID）により指定される）に関わる全ての要求プリミティブに対し、これを受信するローカルな GCC プロバイダは、そのノードが現在加入している会議に関するものかどうかを判断する。もしそうなら、GCC プロバイダは、以下に記述されたそれぞれのプリミティブに従って進む。もし違うなら、要求は拒否され、拒否理由が「無効会議（invalid-conference）」との確認プリミティブを発行する。無効会議に関わるいかなる応答プリミティブも、GCC プロバイダは、無視する（もしくは、このエラーは何らかの不特定なローカルな手段で取り扱われる）。

GCC プロバイダは、GCC プロトコル データ ユニット（GCC PDU）で、お互いに通信し合う。GCC PDU は、2 つの MCS データ サービス プリミティブ（MCS-SEND-DATA もしくは MCS-UNIFORM-SEND-DATA、表 9-4/JT-T124 参照）、あるいはコネクション設定の間に使用される GCC PDU のための MCS-CONNECT-PROVIDER-プリミティブのどちらかを用いて伝達される。

全ての GCC PDU は、要求、応答、指示のいずれかのクラスに分類される。要求 PDU は、対応する応答 PDU を返信要求する PDU と定義される。指示 PDU は、応答を要求しない（もしくは、ある場合には、応答が間接的に与えられる）PDU と定義される。必須の機能に対応していない要求 PDU に対して、一般的な応答 PDU（FunctionNotSupportedResponse）が提供される。これにより、特別な応答 PDU のフォーマットの知識を必要とせず、GCC プロバイダが、要求ノードに応答できる。要求ノードに、この PDU が、何に対する応答であるか知らせるために、要求 PDU の内容の全てが FunctionNotSupportedResponse PDU の中に含まれる。FunctionNotSupportedResponse PDU は、受信した要求 PDU と同じプライオリティで送信される。

注 - この用語（要求、応答、指示）は、プリミティブの定義で使用された要求、応答、指示、確認の定義とは違う。この用語は、実行する機能の観点から、PDU タイプの第一義的な用途に対応するように選ばれた。この理由から、確認という用語は、PDU を記述するためには必要なかった。つまり、応答 PDU は応答プリミティブを生成源とし、確認プリミティブとしての結果をもたらす。

GCC プロトコルは、非標準拡張のためのサポートを含む。nonStandardRequest PDU 受信時に、GCC プロバイダ

はその要求を理解できなければ、返信として FunctionNotSupportedResponse PDU を発行する。nonStandardResponse PDU と nonStandardIndication PDU は、GCC プロバイダより無視されるかもしれない。

8.2. 会議の確立と終了

8.2.1. 会議生成 (Conference Creation)

GCC-CONFERENCE-CREATE-要求プリミティブ受信時に、GCC プロバイダは、表 8-1/JT-T124 に示されるパラメータを持つ MCS-CONNECT-PROVIDER-要求プリミティブを発行する。ローカル GCC プロバイダは、会議 ID を割り当てる。これは、生成した会議に対応するローカル MCS ドメイン セレクタとして使用される。会議 ID (Conference ID) は、発側ドメイン セレクタ (Calling Domain Selector) として含まれている。これは、後のプリミティブでこの会議を識別する手段として、ローカル GCC プロバイダにより保持される。

もし、会議名 (Conference Name) パラメータと会議名修飾子 (Conference Name Modifier) パラメータの組合せが、ローカル ノードが既に加入している会議のパラメータ (数値もしくは文字形式による名前) と同一ならば、要求は直ちに拒否されて、会議名が既に存在 (conference-name-already-exists) という GCC-CONFERENCE-CREATE-応答が発行される。同一でなければ、GCC プロバイダは、会議問合せや会議加入や会議招待の開始への応答手順で使用するため、(ConferenceCreateRequest PDU に会議名 (Conference Name) を含むと共に) 会議名 (Conference Name) と会議名修飾子 (Conference Name Modifier) を保持する。

表 8-1/JT-T124 – ConferenceCreateRequest PDU の MCS-CONNECT-PROVIDER-要求パラメータ

(ITU-T T.124)

パラメータ	内容
発側アドレス (Calling Address)	要求プリミティブから
発側ドメイン セレクタ (Calling Domain Selector)	GCC プロバイダにより選択される会議 ID
着側アドレス (Called Address)	要求プリミティブから
着側ドメイン セレクタ (Called Domain Selector)	NULL
Upward/Downward フラグ (Upward/Downward Flag)	Up
ドメイン パラメータ (Domain ParametersDomain Parameters)	要求プリミティブから
サービス品質 (Quality of Service)	要求プリミティブから
ユーザ データ (User Data)	T.124 オブジェクト識別子 ConferenceCreateRequest PDU (表 8-2/JT-T124 参照)

MCS-CONNECT-PROVIDER-要求のユーザ データ (User Data) パラメータは、オブジェクト識別子 (Object

Identifier) を含む構造をとる。これにより、本標準に従う PDU を識別する。この構造の詳細は、9.6 節で定義する。本 PDU の内容を、表 8-2/JT-T124 に示す。

表 8-2/JT-T124 – ConferenceCreateRequest GCC PDU

(ITU-T T.124)

内容	ソース	シンク
会議名 (Conference Name)	要求	指示と宛先 GCC プロバイダ
会議召集者パスワード (Convener Password)(オプション)	要求	指示
パスワード(Password)(オプション)	要求	指示
ロックされた会議のフラグ (Locked Conference Flag)	要求	指示と宛先 GCC プロバイダ
リスト記載会議のフラグ (Listed Conference Flag)	要求	指示と宛先 GCC プロバイダ
議長制御可能会議フラグ (Conductible Conference Flag)	要求	指示と宛先 GCC プロバイダ
終了方法 (Termination Method)	要求	指示と宛先 GCC プロバイダ
議長特権リスト (Conductor Privilege List) (オプション)	要求	指示と宛先 GCC プロバイダ
議長制御モード会議特権リスト (Conducted-Mode Conference Privilege List)(オプション)	要求	指示と宛先 GCC プロバイダ
非議長制御モード会議特権リスト (Non-Conducted-Mode Conference Privilege List)(オプション)	要求	指示と宛先 GCC プロバイダ
会議記述 (Conference Description) (オプション)	要求	指示と宛先 GCC プロバイダ
発信者識別子 (Caller Identifier) (オプション)	要求	指示
ユーザ データ (User Data) (オプション)	要求	指示

T.124 オブジェクト識別子と ConferenceCreateRequest PDU を含む MCS-CONNECT-PROVIDER-指示受信時に、GCC プロバイダは、特定された ConferenceCreateRequest PDU のパラメータを有する GCC-CONFERENCE-CREATE-指示プリミティブを生成する。このプリミティブは、コントロール GCCSAP に発行される。GCC プロバイダもまた、このプリミティブに含まれるローカルでユニークな文字列である会議 ID (Conference ID) を割り当てる。もし会議生成に成功すれば、GCC プロバイダは、この会議 ID (Conference ID) を、後のプリミティブでこの会議を

識別する手段として使用する。もし、GCC が新しい会議を生成するために必要なリソースを持たなければ、GCC-CONFERENCE-CREATE-指示を生成せずに自動的に否定応答を生成してもよい。さもなければ、コントロール GCCSAP から成功の GCC-CONFERENCE-CREATE- 応答受信時に、GCC プロバイダ(今ではこの会議のトップ GCC プロバイダである)は、MCS-ATTACH-USER-要求を発行する。割り当てられたノード ID (Node ID) を含む MCS-ATTACH-USER-確認受信時に、GCC プロバイダは、MCS-CHANNEL-JOIN-要求を発行して、対応する NODE-ID-CHANNEL に加入する。GCC プロバイダは、MCS-CHANNEL-JOIN-要求を発行して GCC-BROADCAST-CHANNEL にも加入する(前の確認を受け取る前に、連続してそのような要求を発行してもよい)。アタッチの前に、新しいドメインが生成されたことを、MCS プロバイダへローカルに通知することが必要かもしれない。これを行うために必要などんな情報交換も、この標準の範囲を超えたローカルな事項とみなされる。

もし、GCC-CONFERENCE-CREATE-応答が会議名修飾子 (Conference Name Modifier) パラメータを含むなら、GCC プロバイダ(今ではトップ GCC プロバイダである)は、会議問合せや会議加入や会議招待の手順の中で使用するのために、この会議名修飾子を保持する。

GCC プロバイダは、MCS-CONNECT-PROVIDER-応答を生成する。この応答は、GCC-CONFERENCE-CREATE-応答プリミティブの結果 (Result) パラメータが成功か失敗かの指示内容に応じて、「成功 (success)」²か「ユーザ拒否 (user-rejected)」の結果を含む。ユーザ データ (User Data) パラメータは、T.124 オブジェクト識別子と ConferenceCreateResponse PDU を含む。接続プロバイダ プリミティブの内容を、表 8-3/JT-T124 に示す。会議生成が成功する場合、MCS-CONNECT-PROVIDER-指示を受信するノード上の GCC プロバイダは、会議のトップ GCC プロバイダになる。

表 8-3/JT-T124 – ConferenceCreateResponse PDU の MCS-CONNECT-PROVIDER-応答パラメータ

(ITU-T T.124)

パラメータ	内容
ドメインパラメータ (Domain ParametersDomain Parameters)	応答プリミティブから
サービス品質 (Quality of Service)	応答プリミティブから
結果 (Result)	TTC 標準 JT-T122 に定める
ユーザデータ (User Data)	T.124 オブジェクト識別子 ConferenceCreateResponsePDU (表 8-4/JT-T124 参照)

ConferenceCreateResponse PDU を、表 8-4/JT-T124 に示す。ノード ID (Node ID) パラメータは、この PDU を生成源とする GCC プロバイダによって与えられる。本パラメータは、GCC プロバイダが発行する MCS-ATTACH-USER-要求に対する応答の中で、MCS が割り当てられたユーザ ID (User ID) である。タグ (Tag) パラメータは、

² ITU-T勧告T.124では『success』と記述されているが、TTC標準JT-T122のMCS-CONNECT-PROVIDER-応答の結果(Result)パラメータには『「成功(success)」』が無く、代わりに『「成功(successful)」』が存在する。したがって、ここは『「成功(successful)」』とする。(TTC標準独自脚注)

発行元 GCC プロバイダによりローカルでユニークに割り当てられる。これは、返信された UserIDIndication PDU を識別するために使用される。結果 (Result) パラメータは、GCC-CONFERENCE-CREATE-応答プリミティブの結果 (Result) パラメータから直接生成される、GCC 特有の失敗情報を含む。もし、結果 (Result) パラメータが「成功 (successful)」以外の何かであれば、MCS-CONNECT-PROVIDER-応答の結果 (Result) パラメータには「ユーザ拒否 (user-rejected)」と設定する。

表 8-4/JT-T124 – ConferenceCreateResponse GCC PDU

(ITU-T T.124)

内容	ソース	シンク
ノード ID (Node ID)	トップ GCC プロバイダ	宛先 GCC プロバイダ
タグ (Tag)	トップ GCC プロバイダ	宛先 GCC プロバイダ
結果 (Result)	応答	確認
ユーザ データ (User Data) (オプション)	応答	確認

ConferenceCreateResponse PDU 受信時に、もし PDU が成功の結果を指示していれば、GCC プロバイダは、まず MCS-ATTACH-USER-要求を発行する。割り当てられたノード ID を含む MCS-ATTACH-USER 確認受信時に、GCC プロバイダは、MCS-CHANNEL-JOIN-要求を発行することによって、その NODE-ID-CHANNEL に加入する。GCC プロバイダは、MCS-CHANNEL-JOIN-要求を発行することによって、GCC-BROADCAST-CHANNEL にも加入する。GCC プロバイダは、MCS-CHANNEL-JOIN-要求を発行することによって、GCC-CONVENER-CHANNEL (もし、このチャンネルを使う要求のための、全ての機能をサポートしていれば) に加入してもよい。いったん GCC プロバイダが、それぞれのチャンネル加入要求から MCS-CHANNEL-JOIN-確認を受け取ると (前の確認を受け取る前に、連続した要求が発行されるかも知れない)、GCC プロバイダは、MCS-SEND-DATA-要求を発行することによって、トップ GCC プロバイダに、UserIDIndication PDU を発行する。その MCS-SEND-DATA-要求は、受信した ConferenceCreateResponse PDU に含まれる、トップ GCC プロバイダのノード ID (Node ID) を、チャンネル ID (Channel ID) として特定する。その ConferenceCreateResponse PDU は、トップ データの優先度を特定し、データフィールド内に PDU を含む。UserIDIndication PDU の内容を、表 8-5/JT-T124 に示す。タグ (Tag) パラメータは、ConferenceCreateResponse PDU の中で受け取られた、対応するパラメータの値で満たされる。

表 8-5/JT-T124 – UserIDIndication GCC PDU

(ITU-T T.124)

内容	ソース	シンク
タグ (Tag)	発行元 GCC プロバイダ	宛先 GCC プロバイダ

次に、GCC プロバイダは、GCC-CONFERENCE-CREATE-確認プリミティブを生成して、コントロール GCCSAP に発行する。このプリミティブは、元の要求プリミティブからの会議名 (Conference Name)、修飾会議名 (Modified Conference Name) (もしあるなら)、受信 PDU からの結果 (Result) パラメータ、ローカルに割り当てられた会議 ID (Conference ID) を含む。もし、受信 PDU が、失敗の結果を通知すれば、あるいは、GCC プロバイダが、成功の GCC-CONFERENCE-CREATE-確認を発行する前に、この接続に対して MCS-DISCONNECT-PROVIDER-指示を受信すれば、GCC-CONFERENCE-CREATE-確認プリミティブが即座に発行され、失敗の結果を通知する。その際には、アタッチ ユーザやチャンネル加入要求は発行せず、UserIDIndication PDU も送信しない。PDU の結果 (Result) パラ

メータおよび、MCS-CONNECT-PROVIDER-確認で伝達される結果 (Result) パラメータ (あるいは MCS-DISCONNECT-PROVIDER の理由 (Reason) パラメータ) にもとづき、GCC-CONFERENCE-CREATE-確認プリミティブで伝達される結果が生成される。もし、MCS-CONNECT-PROVIDER-確認の結果 (Result) パラメータが、「ユーザ拒否 (user-rejected) 」なら、PDU の結果 (Result) パラメータにより、伝達された結果を判断する。さもなければ、MCS-CONNECT-PROVIDER の結果 (Result) パラメータは、直接使用される。

UserIDIndication 受信時に、会議が生成されたノードは、タグ (Tag) パラメータと、未解決なタグ (Tag) のリストを比較する。もし、タグ (Tag) がそれらのいずれかと一致すれば、コネクション階層構造の真下にあるノードのノード ID (Node ID) のデータベースに、(MCS-SEND-DATA-指示から引き出された) 発信元ノードのユーザ ID (User ID) を保存する。もし、タグ (Tag) がどの未解決なタグ (Tag) とも一致しなければ、PDU は無視される。

もし、この会議のために新たに生成されたトップ GCC プロバイダが、会議生成中のいかなる時 (これは、GCC-CONFERENCE-CREATE-指示がコントロール GCCSAP に対して発行され、UserIDIndication が受信されるまで) でも、確立されたコネクションに対する MCS-DISCONNECT-PROVIDER-指示を受信すれば、コントロール GCCSAP に対して GCC-CONFERENCE-TERMINATE-指示を発行する。GCC-CONFERENCE-TERMINATE-指示では、MCS-DISCONNECT-PROVIDER-指示によって伝えられる理由が「ユーザによる (user-initiated) 」³によるものであれば正常終了、それ以外は異常終了とする。GCC-CONFERENCE-TERMINATE-指示は、このコネクション確立手順に関する以降の処理を中断する。もし、MCS-DISCONNECT-PROVIDER-指示が、コントロール GCCSAP に対する GCC-CONFERENCE-CREATE-指示を発行する前に受信されたなら、GCC プロバイダは、このコネクション確立手順に関する以降の処理を中断し、さらなる動作を行わない。

会議生成が成功する場合のイベントのシーケンスを、図 8-1/JT-T124 に示す。

注 - もし、会議が、他方のノードからの要求によらず、ローカルな指示により生成されれば、そのノードの GCC プロバイダは、図の中で MCS-ATTACH-USER-要求から始まるトップ GCC プロバイダにより実行するイベントのシーケンスをとる。すなわち、MCS-ATTACH-USER-要求を発行することによって MCS にアタッチし、確認を待つ。その後、NODE-ID-CHANNEL と、GCC-BROADCAST-CHANNEL に加入する。この場合、このノード自身会議召集者であるために、オプションとして GCC-CONVENER-CHANNEL に加入することになる。この際、もし後で他方のノードが (正しい召集者パスワード (Convener Password) を発行して) 、召集者であるとして会議に加入したなら、トップ GCC プロバイダは、新しいノードに対する召集者特権を放棄して、接続を受け入れることを選んでもよい。この場合、MCS ドメイン パラメータ (MCS Domain Parameters) は、最初のコネクションが確立された時間ではなく、生成の時間に設定される、と想定されている事にも注意するべきだ。

³ ITU-T 勧告 T.124 では『user-initiated』と記述されているが、TTC 標準 JT-T122 の MCS-DISCONNECT-PROVIDER-指示の理由 (Reason) パラメータには『「ユーザによる (user-initiated) 」』が無く、代わりに『「ユーザ要求 (user-requested) 」』が存在する。したがって、ここは『「ユーザ要求 (user-requested) 」』とする。(TTC 標準独自脚注)

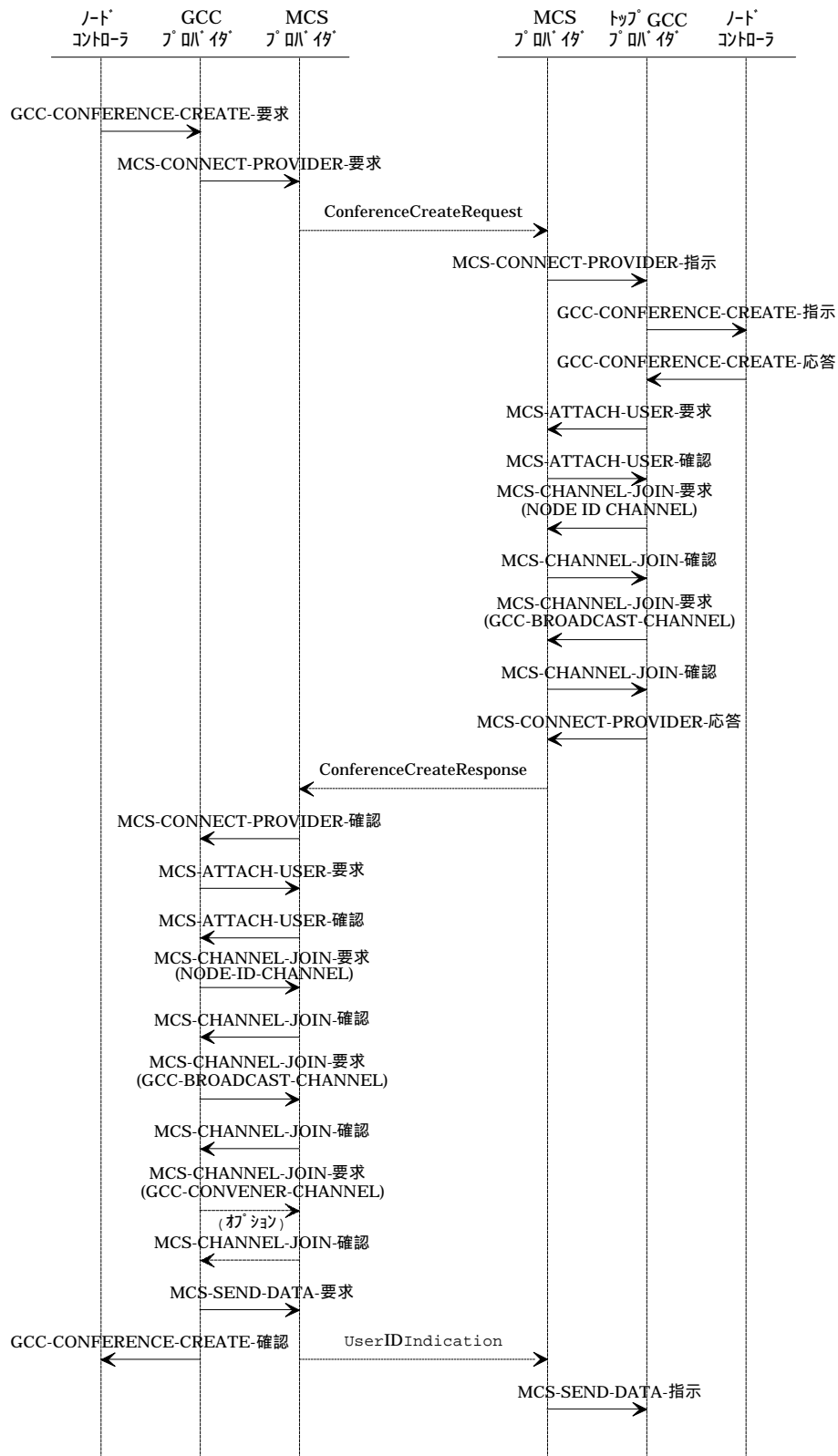


図 8-1/JT-T124 – 会議の生成
(ITU-T T.124)

8.2.2. 会議問合せ (Querying Conference)

GCC-CONFERENCE-QUERY-要求プリミティブ受信時に、GCC プロバイダは、表 8-6/JT-T124 に示すパラメータを

持つ MCS-CONNECT-PROVIDER-要求プリミティブを発行する。

表 8-6/JT-T124 – ConferenceQueryRequest PDU のための MCS-CONNECT-PROVIDER-要求パラメータ (ITU-TT.124)

パラメータ	内容
発側アドレス (Calling Address)	要求プリミティブから
発側ドメインセクタ (Calling Domain Selector)	NULL
着側アドレス (Called Address)	要求プリミティブから
着側ドメインセクタ (Called Domain Selector)	NULL
Upward/Downward フラグ (Upward/Downward Flag)	Up
ドメインパラメータ (Domain ParametersDomain Parameters)	デフォルトドメインパラメータ
サービス品質 (Quality of Service)	デフォルトサービス品質パラメータ
ユーザデータ (User Data)	T.124 オブジェクト識別子 (Object Identifier)
	ConferenceQueryRequest PDU (表 8-7/JT-T124 参照)

MCS-CONNECT-PROVIDER-要求のユーザデータ (User Data) パラメータは、オブジェクト識別子 (Object Identifier) を含む構造をとる。これにより、本標準に従う PDU を識別する。この構造の詳細は、9.6 節で定義する。本 PDU の内容を、表 8-7/JT-T124 に示す。

表 8-7/JT-T124 – ConferenceQueryRequest GCC PDU (ITU-T T.124)

内容	発行元	格納先
ノードタイプ (Node Type)	要求	指示
非平衡指示子 (Asymmetry Indicator) (条件付き)	要求	指示
ユーザデータ (User Data) (オプション)	要求	指示

T.124 オブジェクト識別子 (Object Identifier) と ConferenceQueryRequest PDU を含む MCS-CONNECT-PROVIDER-指示受信時に、GCC プロバイダは、特定された Conference Query Request PDU のパラメータを有する GCC-CONFERENCE-QUERY-指示プリミティブを生成する。このプリミティブは、コントロール GCCSAP に発行される。コントロール GCCSAP からの GCC-CONFERENCE-QUERY-応答受信時に、GCC プロバイダは、結果が拒否(会議がセットアップされていない)であって、そのユーザデータ (User Data) フィールドの中に

ConferenceQueryResponse PDU を含む、MCS-CONNECT-PROVIDER- 応答を送信する。MCS-CONNECT-PROVIDER- 応答のパラメータを、表 8-8/JT-T124 に示す。

表 8-8/JT-T124 – ConferenceQueryResponse PDU のための MCS-CONNECT-PROVIDER- 応答パラメータ (ITU-T T.124)

パラメータ	内容
ドメインパラメータ (Domain ParametersDomain Parameters)	デフォルトドメインパラメータ
サービス品質 (Quality of Service)	デフォルトサービス品質パラメータ
結果 (Result)	ユーザ拒否
ユーザデータ (User Data)	T.124 オブジェクト識別子 (Object Identifier) ConferenceQueryResponse PDU (表 8-9/JT-T124 参照)

Conference Query Response PDU を、表 8-9/JT-T124 に示す。会議記述子リスト (Conference Descriptor List) には、問い合わせたノードで現在加入している各リスト記載会議に関するエントリを含んでいる。会議記述子リスト (Conference Descriptor List) は、ConferenceQueryResponse PDU を生成する GCC プロバイダによってつくられる。この記述子は、GCC プロバイダが加入した各会議の会議名 (Conference Name)、もしあるなら各会議でローカルに保持されている会議名修飾子 (Conference Name Modifier)、もしあるなら会議記述 (Conference Description)、ロック / 非ロック (Locked/Unlocked) フラグ、平文パスワード (Password In The Clear Required) 指示子、および、もしあるならネットワークアドレス (Network Address) フィールドから構成されている。ネットワークアドレス (Network Address) フィールドは、もし存在するなら、このノードが会議にどのように加入したかに応じて、以下のプリミティブにおけるローカルネットワークアドレス (Local Network Address) パラメータからつくられる。すなわち、GCC-CONFERENCE-CREATE-要求、GCC-CONFERENCE-CREATE-応答、GCC-CONFERENCE-JOIN-要求、あるいは GCC-CONFERENCE-INVITE-応答である。結果 (Result) パラメータは、もし問合せ要求を満足させることができれば (たとえ会議リスト (Conference List) が空でも) 成功を、あるいは、もし応答プリミティブに「ユーザ拒否 (user-rejected)」と示されていればその旨を示す。もし、MCS-CONNECT-PROVIDER-指示を処理する GCC プロバイダが、処理中のいかなる時でも、同じ接続に対する MCS-DISCONNECT-PROVIDER-指示を受信すれば、GCC プロバイダは、この手順に関する以降の処理を中断し、さらなる動作は行わない。

表 8-9/JT-T124 – ConferenceQueryResponse GCC PDU (ITU-T T.124)

内容	ソース	シンク
ノードタイプ (Node Type)	応答	確認
非平衡指示子 (Asymmetry Indicator) (条件付き)	応答	確認
会議記述子リスト	発行元 GCC プロバイダ	確認

(Conference Descriptor List)		
結果 (Result)	応答	確認
ユーザデータ (User Data) (オプション)	応答	確認

T.124 オブジェクト識別子 (Object Identifier) と ConferenceQueryResponse PDU を含む MCS-CONNECT-PROVIDER-応答受信時に、GCC プロバイダは、GCC-CONFERENCE-QUERY-確認プリミティブを生成し、コントロール GCCSAP に発行する。確認プリミティブの内容は、ConferenceQueryResponse PDU のパラメータから得られる。本プリミティブの結果 (Result) パラメータは、特に受信 PDU の結果 (Result) パラメータから厳密に得られる。MCS-CONNECT-PROVIDER-確認の結果 (Result) パラメータは成功の場合でも「ユーザ拒否(user-rejected)」を設定されるので無視される。

もし、受信された PDU が、不成功の結果を示したならば、あるいはもし、GCC プロバイダが、成功の GCC-CONFERENCE-QUERY-確認を発行する前に、この接続に対して MCS-DISCONNECT-PROVIDER-指示を受信すれば、GCC-CONFERENCE-QUERY-確認プリミティブが即座に発行され不成功の結果を示す。PDU の結果 (Result) パラメータ、および、MCS-CONNECT-PROVIDER-確認で伝達される結果 (Result) パラメータ (あるいは MCS-DISCONNECT-PROVIDER の理由 (Reason) パラメータ) にもとづき、GCC-CONFERENCE-QUERY-確認プリミティブで伝達される結果が生成される。もし、MCS-CONNECT-PROVIDER-確認の結果パラメータが「ユーザ拒否 (user-rejected)」なら、PDU の結果 (Result) パラメータが伝達された結果の判断に使用される。さもなければ、MCS-CONNECT-PROVIDER の結果 (Result) パラメータが直接使用される。

8.2.3. 会議への加入

GCC-CONFERENCE-JOIN-要求プリミティブを受信した場合、GCC プロバイダは表 8-10/JT-T124 において示されたパラメータと共に MCS-CONNECT-PROVIDER-要求プリミティブを発行する。ローカル GCC プロバイダは、会議 ID(Conference ID)を割り当てる。この会議 ID(Conference ID)は会議に関連したローカル MCS ドメインセレクタとして使用される。会議 ID(Conference ID)は、MCS-CONNECT-PROVIDER-要求の中の発側ドメインセレクタ(Calling Domain Selector)として送信される。

表 8-10/JT-T124 ConferenceJoinRequest PDU のための MCS-CONNECT-PROVIDER-要求パラメータ (ITU-T T.124)

パラメータ	内容
発アドレス(Calling Address)	要求プリミティブより
発側ドメインセレクタ (Calling Domain Selector)	GCC プロバイダによって選択される会議 ID(Conference ID)
着アドレス(Called Address)	要求プリミティブより
着側ドメインセレクタ (Called Domain Selector)	NULL
Upward/Downward フラグ (Upward/Downward Flag)	Up
ドメインパラメータ	要求プリミティブより

(Domain ParametersDomain Parameters)	
サービス品質(Quality of Service)	要求プリミティブより
ユーザデータ(User Data)	T.124 オブジェクト識別子 (Object Identifier)
	ConferenceJoinRequest PDU (表 8-11/JT-T124 参照)

MCS-CONNECT-PROVIDER-要求のユーザデータ(User Data)パラメータは、オブジェクト識別子(Object Identifier)の入った構造を含んでいる。このオブジェクト識別子(Object Identifier)は本標準にもとづく PDU を識別するものである(注：付属資料 B 参照)。この構造の詳細は、9.6 節において定義される。PDU の内容を表 8-11/JT-T124 に示す。発側ノードからのものにはタグ(Tag)パラメータを含まない。会議名(Conference Name)と会議名修飾子(Conference Name Modifier)は、要求プリミティブから取り出される。PDU における会議名修飾子(Conference Name Modifier)パラメータは、プリミティブの着側ノード会議名修飾子(Called Node Conference Name Modifier)パラメータから取り出される。同様に要求プリミティブから取り出される発側ノード会議名修飾子(Calling Node Conference Name Modifier)は、会議加入要求と会議問い合わせ要求への応答のために会議の識別子としてローカル GCC プロバイダで維持管理される。

表 8-11/JT-T124 ConferenceJoinRequestGCCPDU

(ITU-T T.124)

内容	ソース	シンク
会議名 (Conference Name) (条件付)	要求	MCS-CONNECT-PROVIDER- 指示を受け取る GCC プロバイダ
会議名修飾子(Conference Name Modifier) (オプション)	要求	MCS-CONNECT-PROVIDER- 指示を受け取る GCC プロバイダ
タグ(Tag) (条件付き)	MCS-CONNECT-PROVIDER- 指示を受け取る GCC プロバイダ	トップ GCC プロバイダ
パスワード>Password) (オプション)	要求	指示
召集者パスワード (Convener Password) (オプション)	要求	指示
発信者識別子(Caller Identifier) (オプション)	要求	指示
ユーザデータ(Usr Data) (オプション)	要求	指示

T.124 オブジェクト識別子(Object Identifier)と ConferenceJoinRequest PDU を含む MCS-CONNECT-PROVIDER-指示を受信した場合、GCC プロバイダの動作は、それが会議名(Conference Name)と会議名修飾子(Conference Name Modifier)によって指定された会議のトップ GCC プロバイダであるかどうか依存する。GCC プロバイダは自身の加入している会議が加入される会議であるかどうか、会議名(Conference Name)と会議名修飾子(Conference Name Modifier)をこの加入している会議のリストと比較することで決定する。ConferenceJoinRequest PDU で受信した会議名(Conference Name)は、会議名(Conference Name)の数字形式とテキスト形式のどちらを含んでいてもよい。GCC プロバイダは、存在する会議名の対応する内容と指示された名前的一致を試す。会議名(Conference Name)と会議名修飾子(Conference Name Modifier)の両方が、現在の会議のそれに一致しなければならない。会議名修飾子(Conference Name Modifier)が要求に含まれない場合、GCC プロバイダは会議名修飾子(Conference Name Modifier)を持たない会議とのみ一致を試す。

MCS-CONNECT-PROVIDER-指示を受信した GCC プロバイダが、要求を発行したノードが加入しようとする会議に加入しているがトップ GCC プロバイダではない場合、GCC プロバイダはトップ GCC プロバイダに ConferenceJoinRequest PDU をフォワードする。PDU のフォワードは、チャンネル ID(Channel ID)がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが Top で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を送信することによりおこなわれる。この場合、ローカルにユニークな識別子が割り当てられ、PDU のタグ(Tag)パラメータにその識別子を含められる。この番号は、トップ GCC プロバイダから返される応答 PDU を識別するために使用される。会議名(Conference Name)と会議名修飾子(Conference Name Modifier)パラメータは、トップ GCC プロバイダに必要なものとして受信した PDU から取り除いてもよい。

トップ GCC プロバイダがこの PDU を受信したとき、会議がロックされていないならば、対応する会議のための会議 ID(Conference ID)と ConferenceJoinRequest PDU で指定されたパラメータを指定した GCC-CONFERENCE-JOIN-指示プリミティブを生成する。このプリミティブはコントロール GCCSAP に発行される。会議名(Conference Name)と会議名修飾子(Conference Name Modifier)パラメータはトップ GCC プロバイダではなく、MCS-CONNECT-PROVIDER-指示を受信したノードによってのみ使用される。会議がロックされている場合、トップ GCC プロバイダは、ConferenceJoinResponse PDU をもともとの要求者ではなく ConferenceJoinRequest PDU を含む MCS-SEND-DATA の発信者に送信する。PDU の送信は、チャンネル ID(Channel ID)がそのノードのユーザ ID(User ID)で、データプライオリティが Top で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を送信することによりおこなわれる。この PDU は、結果コードとして「ロックされた会議 (locked-conference)」を示す。一方、GCC-CONFERENCE-JOIN-応答を受信した場合、GCC プロバイダは、ConferenceJoinResponsePDU をもともとの要求者ではなく ConferenceJoinRequestPDU を含んだ MCS-SEND-DATA-要求の発信者に送信する。PDU の送信は、チャンネル ID(Channel ID)がそのノードのユーザ ID(User ID)で、データプライオリティが Top で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することによりおこなわれる。ConferenceJoinResponse PDU の内容を表 8-13/JT-T124 に示す。ノード ID(Node ID)は応答中には含まれないが、トップノード ID(Top Node ID)パラメータ、タグ(Tag)、結果(Result)パラメータ、会議プロファイル(Conference Profile)に関連したパラメータは含まれる。タグ(Tag)は、受信した ConferenceJoinRequest PDU の対応するパラメータと同じ値を持つ。

この PDU を含む MCS-SEND-DATA-指示を受信した場合、タグ(Tag)パラメータがローカルに保持された未解決加入要求リストのタグ(Tag)パラメータと一致するならば、GCC プロバイダは MCS-CONNECT-PROVIDER-応答を生成する。この MCS-CONNECT-PROVIDER-応答には、受信した PDU 指示の結果(Result)パラメータが成功、失敗のどちらになったかにあわせて、「成功(success)」、「ユーザ拒否(user-rejected)」のどちらかの結果が含まれる。ユーザデータ(User Data)パラメータは、ConferenceJoinResponse PDU と同様に T.124 オブジェクト識別子(Object Identifier)

を含む。ConferenceJoinResponse PDU のノード ID(Node ID)パラメータは、発側ノードに直接接続されたノードのノード ID(Node ID)になる。コネクトプロバイダプリミティブの内容を表 8-12/JT-T124 に示す。

最初の MCS-CONNECT-PROVIDER-指示の受信者がトップ GCC プロバイダであり、会議がロックされていない場合、指定された会議の会議 ID(Conference ID)と ConferenceJoinRequest PDU で指定されたパラメータと同じ値を指定して GCC-CONFERENCE-JOIN-指示プリミティブを生成する。このプリミティブはコントロール GCCSAP へ発行する。会議がロックされている場合、トップ GCC プロバイダは代わりに MCS-CONNECT-PROVIDER-応答を生成する。これに含まれる ConferenceJoinResponse PDU の結果コードは「ロックされた会議(locked-conference)」になる。一方、GCC-CONFERENCE-JOIN-応答を受信した場合、GCC プロバイダは MCS-CONNECT-PROVIDER-応答を生成する。この MCS-CONNECT-PROVIDER-応答は ConferenceJoinResponse PDU を含む。この PDU の結果(Result)パラメータは、GCC-CONFERENCE-JOIN-応答プリミティブの結果(Result)パラメータの値が成功か失敗かに依存して「成功(success)」または「ユーザー拒否(user-rejected)」になる。

最初の MCS-CONNECT-PROVIDER-指示の受信者がどの会議にも加入していない場合、または GCC-CONFERENCE-JOIN-指示プリミティブをまったくサポートしていない場合、加入要求を拒絶してもよい。加入要求の拒絶は、MCS-CONNECT-PROVIDER-応答を発行することによっておこなう。これは、結果コードとして「ユーザー拒否(user-rejected)」を含み、PDU の理由コードが「無効な会議(invalid-conference)」である ConferenceJoinResponse を含む。

最初の MCS-CONNECT-PROVIDER-指示の受信者が ConferenceJoinRequest PDU の会議名(Conference Name)と会議名修飾子(Conference Name Modifier)パラメータで指定された会議に加入していない場合、「指定ドメインなし(no-such-domain)」という理由で MCS-CONNECT-PROVIDER-応答を発行することにより要求を拒絶しなければならない。

表 8-12/JT-T124 ConferenceJoinResponse PDU のための MCS-CONNECT-PROVIDER-応答パラメータ (ITU-T T.124)

パラメータ	内容
ドメインパラメータ (Domain ParametersDomain Parameters)	以前に保存された GCC プロバイダより
サービス品質(Quality of Service)	以前に保存された GCC プロバイダより
結果(Result)	TTC 標準 JT-T122 で指定されたもの
ユーザデータ(User Data)	T.124 オブジェクト識別子 (Object Identifier) ConferenceJoinResponse PDU (表 8-13/JT-T124 参照)

ConferenceJoinResponse PDU を、表 8-13/JT-T124 に示す。ノード ID(Node ID)パラメータは GCC プロバイダによって発行される MCS-ATTACH-USER-要求の応答で MCS によって割り当てられるユーザ ID(User ID)であり、発側ノードに直接接続されたノードで GCC プロバイダによって供給される。タグ(Tag)パラメータは、発側ノードに直接接続されたノードで GCC プロバイダによりローカルでユニークな値を与えられる。この値は返却された UserIDIndication

PDU を識別するために使用される。このタグ(Tag)パラメータの値は、このパラメータがこのノードによって ConferenceJoinRequest PDU で送信するために割り当てられて以降、トップ GCC プロバイダから受信したパラメータと同じ値になってもよい。このため、どちらかの目的のために使用されるタグ(Tag)はローカルでユニークでなければならない。この PDU の他のパラメータはトップ GCC プロバイダによってセットされる。これは、会議プロファイル (Conference Profile)に関連するパラメータと同様に、トップ GCC プロバイダのノード ID(Node ID)であるトップノード ID(Top Node ID)を含む。会議別名(Conference Name Alias)もトップ GCC プロバイダによりセットされる。これは会議名(Conference Name)が数字形式とテキスト形式の両方を含むかどうかによって依存して条件付きで含まれる。含まれる場合、会議別名(Conference Name Alias)は、ConferenceJoinRequest PDU に含まれない形式の会議名(Conference Name)になる。MCS-CONNECT-PROVIDER メッセージの結果(Result)パラメータが「ユーザー拒否(user-rejected)」の場合、結果(Result)パラメータは GCC 特有の失敗情報を含む。この情報は、GCC-CONFERENCE-CREATE-応答 PDU の結果(Result)パラメータから入手される。

表 8-13/JT-T124 ConferenceJoinResponse GCC PDU

(ITU-TT.124)

内容	ソース	シンク
ノード ID(Node ID)(条件付き)	発側ノードに直接接続されたノードの GCC プロバイダ	あて先 GCC プロバイダ
トップノード ID(Top Node ID)	トップ GCC プロバイダ	あて先 GCC プロバイダ
タグ(Tag)	トップ GCC プロバイダ	あて先 GCC プロバイダ
会議別名(Conference Name Alias) (条件付き)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
平文パスワードフラグ (Password In The Clear Required Flag)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
ロックされた会議フラグ (Locked Conference Flag)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
リスト記載会議フラグ (Listed Conference Flag)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
議長制御可能会議フラグ (Conductible Conference Flag)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
終了方法 (Termination Method)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
議長特権リスト(Conductor Privilege List) (オプション)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
議長制御モード会議特権リスト (Conducted-Mode Conference)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ

Privilege List) (オプション)		
非議長制御モード会議特権リスト(Non-Conducted-Mode Conference Privilege List) (オプション)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
会議記述(Conference Description) (オプション)	トップ GCC プロバイダ	確認とあて先 GCC プロバイダ
パスワード>Password) (オプション)	要求	指示
結果(Result)	応答	確認
ユーザデータ(User Data) (オプション)	応答	確認

ConferenceJoinResponse PDU を受信し PDU の結果が成功の場合、GCC プロバイダは、MCS-ATTACH-USER-要求を最初に発行する。割り当てられた ノード ID(Node ID)を含む MCS-ATTACH-USER-確認を受信した場合、GCC プロバイダは MCS-CHANNEL-JOIN-要求を発行することによって対応する NODE-ID-CHANNEL を加入させる。GCC プロバイダは、MCS-CHANNEL-JOIN-要求を発行することによって、GCC-BROADCAST-CHANNEL も加入させる。最初の GCC-CONFERENCE-JOIN-要求が召集者パスワード(Convener Password)を指定していた場合、ノードが会議召集者(Conference Convener)としての役割を取り戻そうとしていることを示している。この場合、GCC プロバイダは MCS-CHANNEL-JOIN-要求を発行することによって、GCC-CONVENER-CHANNEL も(このチャンネルが使用する機能をサポートするならば)加入させてよい。GCC プロバイダが各チャンネル加入要求の成功した MCS-CHANNEL-JOIN-確認を受信すると、UserIDIndication PDU を直接接続したノードの GCC プロバイダに送信する。この送信は、受信した ConferenceJoinResponse PDU に含まれた直接接続したノードのノード ID(Node ID)を指定し、データプライオリティが Top で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することによりおこなう。UserIDIndication PDU の内容は表 8-5/JT-T124 に示す。タグ(Tag)パラメータは、ConferenceJoinResponse PDU で受信した対応するパラメータの値をセットされる。

この時 GCC プロバイダは、GCC-CONFERENCE-JOIN-確認プリミティブを生成し、コントロール GCCSAP へ発行する。このプリミティブは、ローカルに割り当てられた会議 ID(Conference ID)とともに、受信した PDU から取り出した結果(Result)パラメータを含む。受信した PDU が失敗を示した場合、または GCC プロバイダが成功の意味の結果を含む GCC-CONFERENCE-JOIN-確認を発行する前にこのコネクションの MCS-DISCONNECT-PROVIDER-指示を受信した場合、失敗を示す GCC-CONFERENCE-JOIN-確認 プリミティブが即座に発行される。この時は、アタッチユーザ要求やチャンネル加入要求は発行しないし、また UserIDIndicationPDU の送信もしない。PDU の結果(Result)パラメータは、MCS-CONNECT-PROVIDER-確認の結果(Result)パラメータや MCS-DISCONNECT-PROVIDER の理由(Reason)パラメータとともに GCC-CONFERENCE-JOIN-確認プリミティブの結果に使用される。MCS-CONNECT-PROVIDER-確認の結果(Result)パラメータが「ユーザー拒否(user-rejected)」の場合、結果を決定するために PDU における結果(Result)パラメータが使用される。そうでない場合、MCS-CONNECT-PROVIDER の結果(Result)パラメータが直接使用される。

ConferenceJoinResponse PDU で会議別名(Conference Name Alias)を受信した場合、この会議別名(Conference Name Alias)は、GCC-CONFERENCE-JOIN-確認で返却する完全な会議名(Conference Name)を形成するために、ConferenceJoinRequest PDU に含まれた会議名(Conference Name)に加えられる。この完全な会議名(もしあれば、テキスト形式と数字形式の両方)はローカル GCC プロバイダでこの会議の識別子として維持管理される。これは、会議加入要求、会議問い合わせ要求に対する応答の手続および会議への招待の開始の時に使用される。

UserIDIndication を受信したとき、発側ノードに直接接続されたノードは、タグ(Tag)パラメータとこの PDU が関係する会議におけるノードのタグ(Tag)パラメータのリストを比較する。タグ(Tag)がリストの中の 1 つと一致する場合、MCS-SEND-DATA-指示から抽出される発行側ノードのユーザ ID(User ID)を、ノード ID(Node ID)データベースに保存しなければならない。このデータベースはコネクション階層構造で自身の直下のノードのノード ID(Node ID)を保存するものである。タグ(Tag)がどの未解決なタグ(Tag)とも一致しないならば、その PDU を無視する。

トップ GCC プロバイダが、加入操作が成功した結果としての UserIDIndication を受信し、召集者パスワード(Convener Password)パラメータが ConferenceJoinRequest PDU に含まれていた場合、トップ GCC プロバイダは有効な会議召集者として UserIDIndication で示された発側ノードのノード ID(Node ID)を保持する。その後の要求で召集者だけが行なってもよいものについて確認を取るために、これを使用する。

コネクション確立のための加入処理の間に、発側ノードに直接接続されたノードの GCC プロバイダが MCS-DISCONNECT-PROVIDER-指示を受信した場合、GCC プロバイダはこの処理を停止し、これ以上の処理はおこなわない。

トップ GCC プロバイダに直接接続されたノードからの会議加入が成功する場合のイベントのシーケンスを図 8-2/JT-T124 に示す。発側ノードがトップ GCC プロバイダに直接接続されない場合のものを図 8-3/JT-T124 に示す。

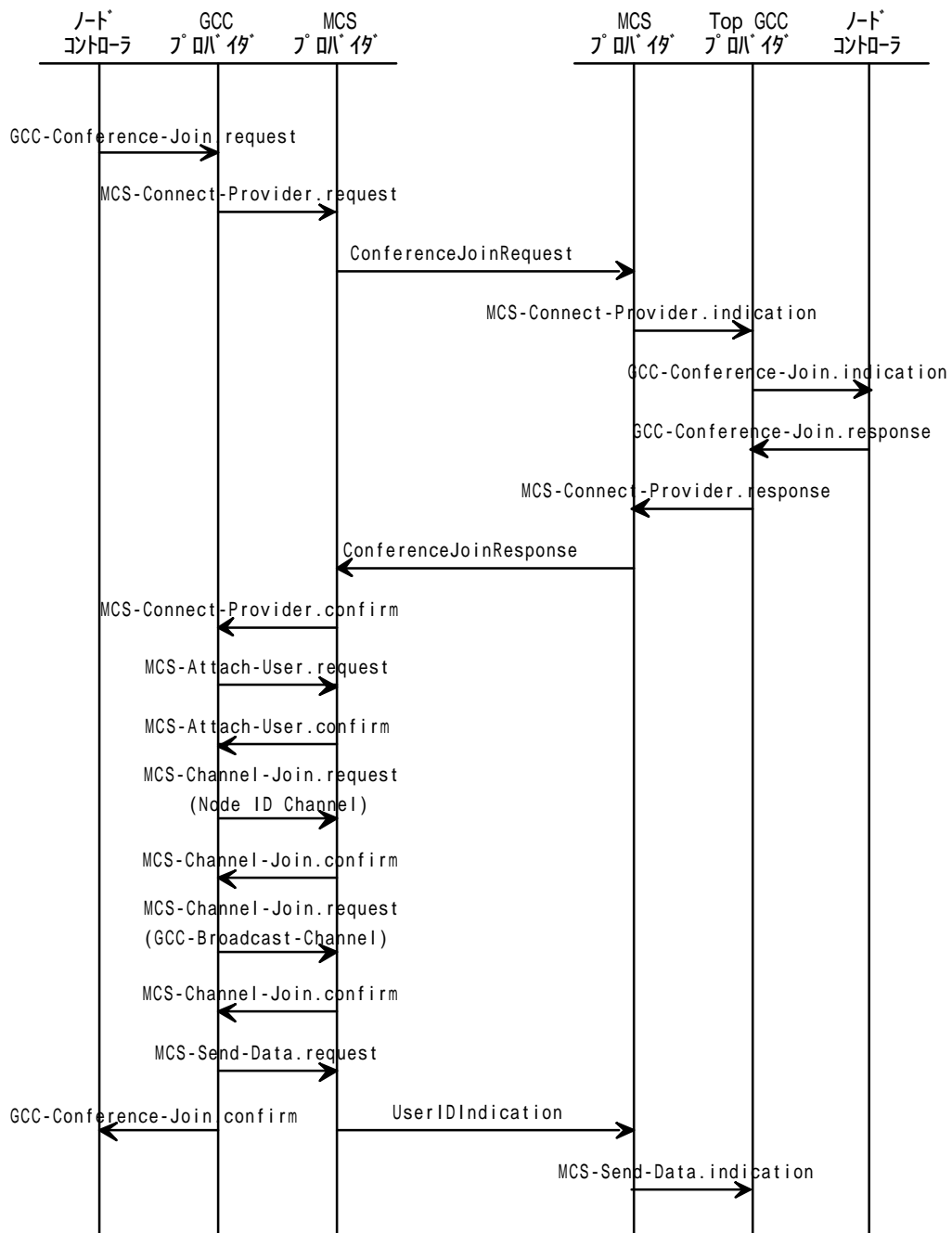


図 8-2/JT-T124 トップ GCC プロバイダに直接接続する場合の会議への加入

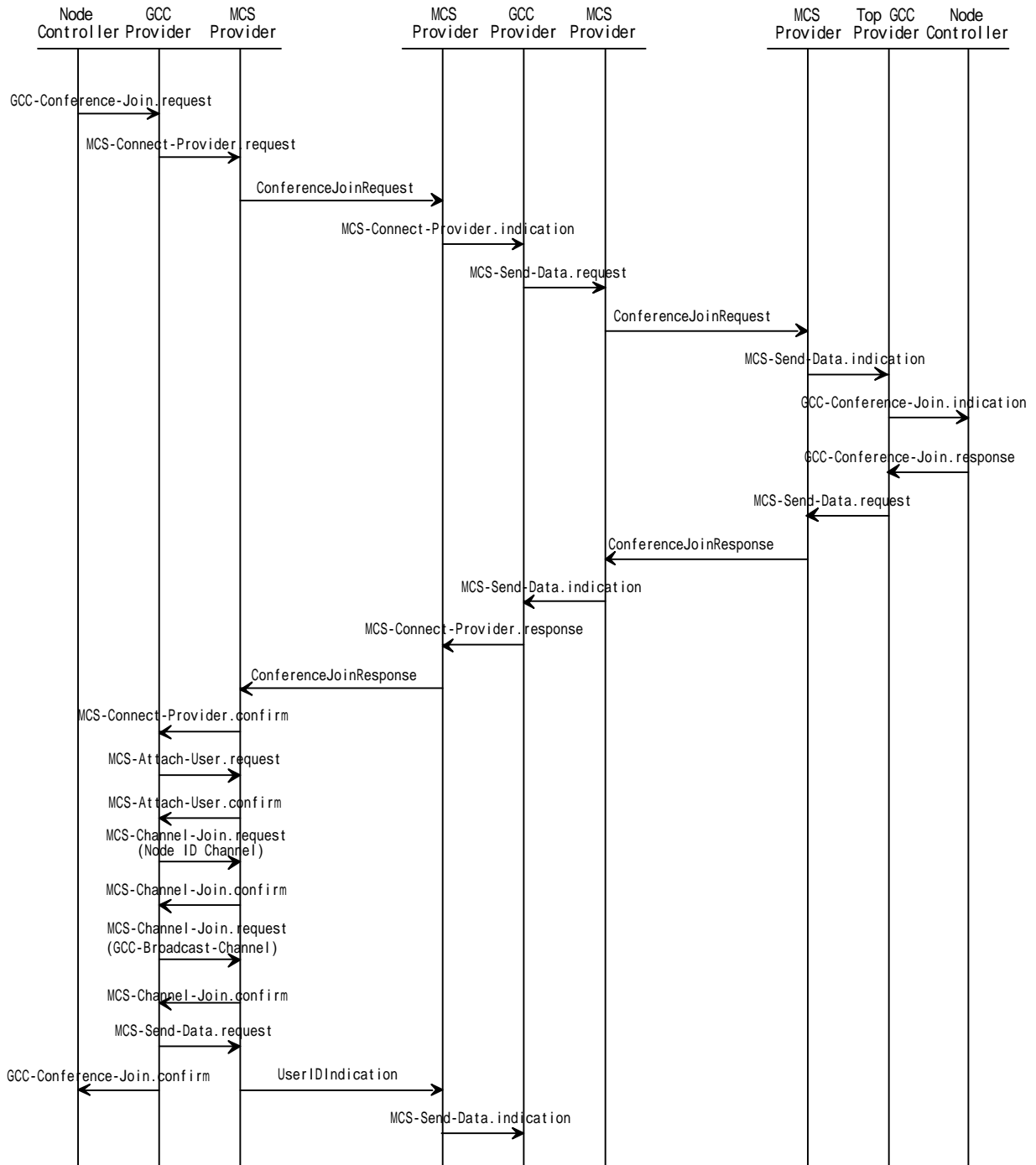


図 8-3/JT-T124 トップ GCC プロバイダに間接的に接続する場合の会議への加入

8.2.4. 会議へのノードの招待

GCC-CONFERENCE-INVITE-要求プリミティブを受信した場合、GCC プロバイダは表 8-14/JT-T124 で示すパラメータで MCS-CONNECT-PROVIDER-要求プリミティブを発行しなくてはならない。

表 8-14/JT-T124 ConferenceInviteRequest PDU のための MCS-CONNECT-PROVIDER-要求パラメータ (ITU-T T.124)

パラメータ	内容
発アドレス(Calling Address)	要求プリミティブより
発側ドメインセクタ(Calling Domain Selector)	要求プリミティブからの会議 ID
着アドレス(Called Address)	要求プリミティブ より
着側ドメインセクタ(Called Domain Selector)	NULL
Upward/Downward フラグ (Upward/Downward Flag)	Down
ドメインパラメータ(Domain Parameters)	GCC プロバイダが以前に保存したものより
サービス品質(Quality of Service)	GCC プロバイダが以前に保存したものより
ユーザデータ(User Data)	T.124 オブジェクト識別子 (Object Identifier) ConferenceInviteRequest PDU(表 8-15/JT-T124 参照)

MCS-CONNECT-PROVIDER-要求のユーザデータ(User Data)パラメータは、オブジェクト識別子(Object Identifier)の入った構造を含んでいる。このオブジェクト識別子(Object Identifier)は本標準にもとづく PDU を識別するものである(注：付属資料 B 参照)。この構造の詳細は、9.6 節において定義される。PDU の内容を表 8-15/JT-T124 に示す。会議名(Conference Name)は、ローカルな会議プロファイルに保存されるもので、要求プリミティブの会議 ID(Conference ID)によって指定された会議の名前である。ノード ID(Node ID)パラメータは GCC プロバイダによって発行される MCS-ATTACH-USER-要求の応答で MCS によって割り当てられるユーザ ID である。ノード ID(Node ID)パラメータは、発信元 GCC プロバイダによって供給されなければならない。トップノード ID(Top Node ID)は、トップ GCC プロバイダのノード ID である。招待側 GCC プロバイダによって以前に保存されたものである。タグ(Tag)パラメータは、ローカルにユニークなもので、発信元 GCC プロバイダによって割り当てられる。それは、戻された UserIDIndication PDU を識別するために使用される。

表 8-15/JT-T124 ConferenceInviteRequestGCCPDU
(ITU-T T.124)

内容	ソース	シンク
会議名(Conference Name)	発信元 GCC プロバイダ	あて先 GCC プロバイダと指示
ノード ID(Node ID)	発信元 GCC プロバイダ	あて先 GCC プロバイダ
トップノード ID(Top Node ID)	発信元 GCC プロバイダ	あて先 GCC プロバイダ
タグ(Tag)	発信元 GCC プロバイダ	あて先 GCC プロバイダ

平文パスワードフラグ (Password In The Clear Required Flag)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
ロックされた会議フラグ (Locked Conference Flag)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
リスト記載会議フラグ(Listed Conference Flag)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
議長制御可能会議フラグ (Conductible Conference Flag)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
終了方法(Termination Method)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
議長特権リスト (Conductor Privilege List)(オプション)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
議長制御モード会議特権リスト (Conducted-Mode Conference Privilege List) (オプション)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
非議長制御モード会議特権リス ト (Non-Conducted-Mode Conference Privilege List) (オプ ション)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
会議記述 (Conference Description) (オプション)	発信元 GCC プロバイダ	指示とあて先 GCC プ ロバイダ
発信者識別子 (Caller Identifier)(オプション)	要求	指示
ユーザデータ(User Data)(オプ ション)	要求	指示

T.124 オブジェクト識別子(Object Identifier)と ConferenceInviteRequest PDU を含む MCS-CONNECT-PROVIDER-指示を受信した場合、GCC プロバイダは、GCC-CONFERENCE-INVITE-指示プリミティブを生成し、コントロール GCCSAP へ発行しなければならない。このプリミティブの会議 ID(Conference ID)は、GCC プロバイダによってローカルに割り当てられ、ローカル MCS ドメインセクタとして使用されなければならない。GCC が会議に加入するのに必要な資源を持たない場合、GCC-CONFERENCE-INVITE-指示の生成をせず、自動的に否定的な応答を生成してもよい。GCC-CONFERENCE-INVITE-応答を受信した場合、GCC プロバイダは MCS-CONNECT-PROVIDER-応答を生成しなければならない。この MCS-CONNECT-PROVIDER-応答には、GCC-CONFERENCE-INVITE-応答プリミティブの結果パラメータが成功か失敗かに依存して「成功(success)⁴」、あるいは「ユーザ拒否(user-rejected)」のどちらかの結果を含む。そしてこの結果は ConferenceInviteResponsePDU に含まれることになる。

⁴ ITU-T勧告T.124では『success』と記述されているが、TTC標準JT-T122のMCS-CONNECT-PROVIDER-応答の結果(Result)パラメータには『「成功(success)」』が無く、代わりに『「成功(successful)」』が存在する。したがって、ここは『「成功(successful)」』とする。(TTC標準独自脚注)

ユーザデータ(User Data)パラメータは、ConferenceInviteResponse PDU と同様に T.124 オブジェクト識別子(Object Identifier)を含む。コネクトプロバイダプリミティブの内容を、表 8-16/JT-T124 に示す。

成功したという応答に会議名修飾子(Conference Name Modifier)パラメータが含まれるならば、GCC プロバイダは受信した ConferenceInviteRequest PDU からの会議名(Conference Name)パラメータと同様にこれを維持管理しなければならない。この会議名(Conference Name)にはテキスト形式と同様に数字形式も含まれることがある。この会議名(Conference Name)は、会議のローカル ID とともに、会議加入要求、会議問い合わせ要求に対する応答の手続および会議への招待の開始の時に使用される。会議名修飾子が存在しないならば、会議名だけが維持管理される。

MCS-CONNECT-PROVIDER-応答を送った後で、招待が成功であるならば、GCC プロバイダは MCS-ATTACH-USER-要求を発行しなければならない。割り当てられたノード ID を含んでいる MCS-ATTACH-USER-確認を受信した時、GCC プロバイダは MCS-CHANNEL-JOIN-要求を発行することにより、対応する NODE-ID-CHANNEL に加入しなければならない。さらに、GCC プロバイダは、MCS-CHANNEL-JOIN-要求を発行することによって、GCC-BROADCAST-CHANNEL に加入しなければならない。GCC プロバイダが、成功したそれぞれのチャンネル加入要求に対する MCS-CHANNEL-JOIN-確認を受信すると、直接接続したノードの GCC プロバイダに UserIDIndication PDU を送信しなければならない。この送信は、受信した ConferenceInviteRequest PDU に含まれていた直接接続したノードのノード ID を指定し、データプライオリティが Top で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することでおこなう。UserIDIndication PDU の内容は、表 8-5/JT-T124 に示す。タグパラメータは、ConferenceInviteRequest PDU で受信した対応するパラメータの値をセットされる。

被招待側ノードの GCC プロバイダが、接続が確立されて会議加入の処理をしている間に(この時間は、GCC-CONFERENCE-INVITE-指示がコントロール GCCSAP に発行された時間を含み、UserIDIndication が転送される時間までであるが)MCS-DISCONNECT-PROVIDER-指示を受信した場合、GCC-CONFERENCE-TERMINATE-指示をコントロール GCCSAP に発行しなければならない。MCS-DISCONNECT-PROVIDER-指示で理由が「ユーザによる(user-initiated)⁵」になっていた場合、この GCC-CONFERENCE-TERMINATE-指示は、理由として「要求された正常終了(requested normal termination)」を示す。その他の場合は「異常終了(error termination)」を示す。この時、このコネクションの確立処理は中止される。コントロール GCCSAP へ GCC-CONFERENCE-INVITE-指示を発行する前に MCS-DISCONNECT-PROVIDER-指示受信した場合、GCC プロバイダはこのコネクションの確立処理を止め、それ以上の処理をしないようにする必要がある。

表 8-16/JT-T124 ConferenceInviteResponse PDU のための MCS-CONNECT-PROVIDER-応答パラメータ (ITU-T T.124)

パラメータ	内容
ドメインパラメータ (Domain Parameters)	応答プリミティブより
サービス品質(Quality of Service)	応答プリミティブより
結果(result)	TTC 標準 JT-T122 の仕様による
ユーザデータ(User Data)	T.124 オブジェクト識別子

⁵ ITU-T勧告T.124では『user-initiated』と記述されているが、TTC標準JT-T122のMCS-DISCONNECT-PROVIDER-指示の理由(Reason)パラメータには『「ユーザによる(user-initiated)」』が無く、代わりに『「ユーザ要求(user-requested)」』が存在する。したがって、ここは『「ユーザ要求(user-requested)」』とする。(TTC標準独自脚注)

	(Object Identifier)
	ConferenceInviteResponse PDU (表 8-17/JT-T124 参照)

ConferenceInviteResponse PDU の内容は表 8-17/JT-T124 に示す。

表 8-17/JT-T124 ConferenceInviteResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
結果(Result)	応答	確認
ユーザデータ(User Data) (オプション)	応答	確認

ConferenceInviteResponse PDU を受信し、MCS-CONNECT-PROVIDER の結果パラメータが「成功(successful)」だった場合、GCC プロバイダはユーザデータ(User Data)パラメータを記録しなければならない。これは、UserIDIndication を受信したときに生成する GCC-CONFERENCE-INVITE-確認プリミティブに格納するためである。結果パラメータが「失敗(unsuccessful)」だった場合、または GCC プロバイダが成功した GCC-CONFERENCE-INVITE-確認を発行する前に MCS-DISCONNECT-PROVIDER-指示を受信した場合、GCC プロバイダは即座に GCC-CONFERENCE-INVITE-確認プリミティブを生成し、コントロール GCCSAP に発行しなければならない。MCS-CONNECT-PROVIDER-確認の結果(Result)パラメータと同様に、この PDU の結果(Result)パラメータ(もしくは MCS-DISCONNECT-PROVIDER-指示の結果パラメータ)は GCC-CONFERENCE-INVITE-確認プリミティブの結果を生成するために使用される。MCS-CONNECT-PROVIDER-確認の結果(Result)パラメータが「ユーザ拒否(user-rejected)」だった場合、PDU の結果(Result)パラメータが使用される。そうでない場合、MCS-CONNECT-PROVIDER の結果(Result)が直接使用される。

UserIDIndication を受信した時、被招待側ノードに直接接続されたノードはそのタグ(Tag)パラメータとこの PDU が関係する会議におけるノードのタグパラメータのリストを比較しなければならない。タグ(Tag)がリストの中の 1 つと一致する場合、MCS-SEND-DATA-指示から抽出される発行側ノードのユーザ ID を、ノード ID データベースに保存しなければならない。このデータベースはコネクション階層構造構造で自身の直下のノードのノード ID を保存するものである。タグ(Tag)がどの未解決のタグとも一致しないならば、その PDU を無視しなければならない。

タグ(Tag)が未解決な招待に関するものである場合、UserIDIndication を受信したならば、GCC プロバイダは GCC-CONFERENCE-INVITE-確認プリミティブを生成し、結果(Result)パラメータに「成功(successful)」を入れてコントロール GCCSAP に発行しなければならない。

会議への招待の成功した場合のイベントのシーケンスを図 8-4/JT-T124 に示す。

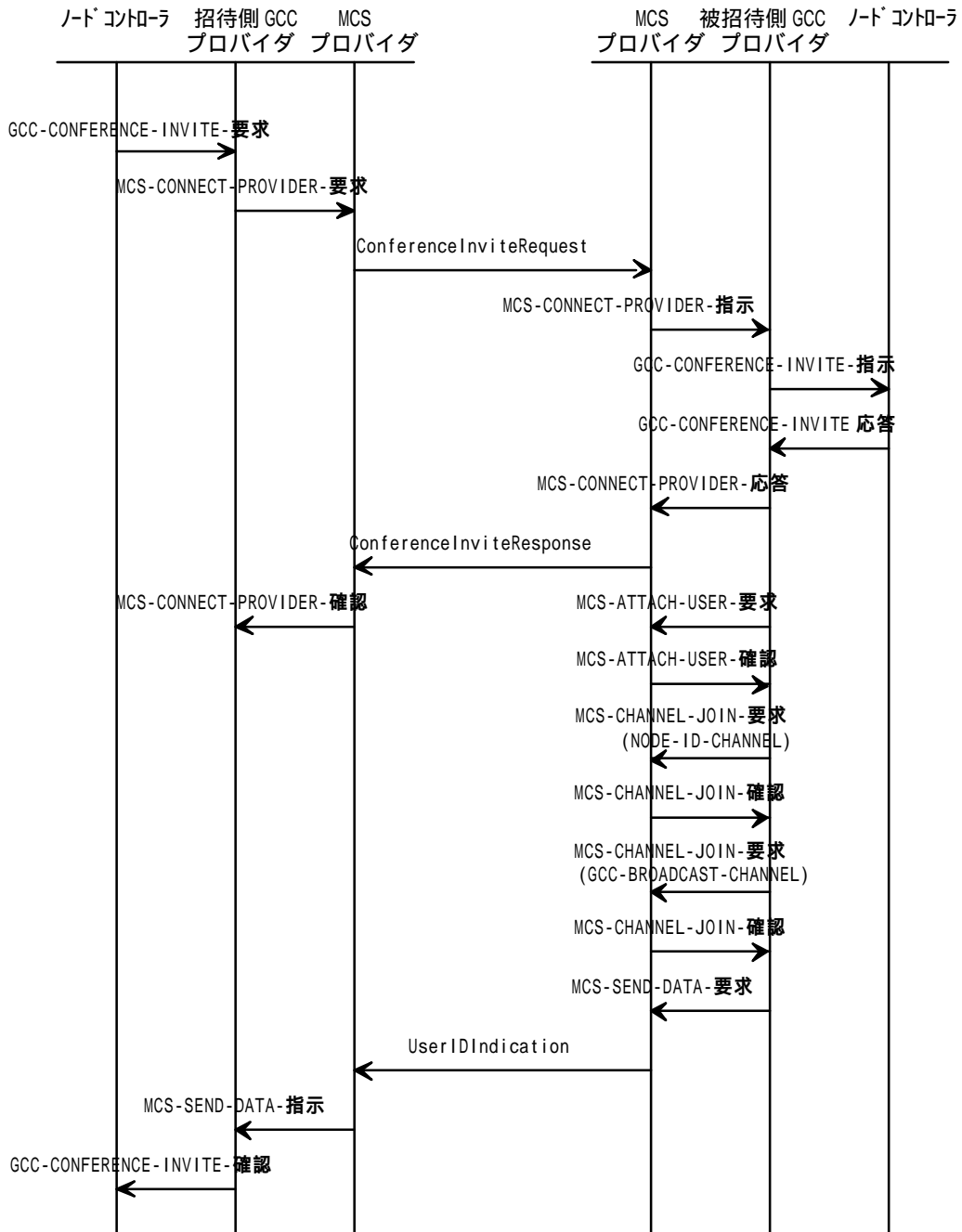


図 8-4/JT-T124 会議へのノードの招待
(ITU-T T.124)

8.2.5. 会議へノードを追加するための要求

GCC-CONFERENCE-ADD-要求プリミティブの受信時に、GCC プロバイダはトップ GCC プロバイダに ConferenceAddRequest PDU を送信する。PDU の送信は、チャンネル ID がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceAddRequest PDU の内容を表 8-18/JT-T124 に示す。その内容は、GCC-CONFERENCE-ADD-要求プリミティブの中を通ったパラメータから構成される。タグ(Tag)パラメータは要求する GCC プロバイダによってローカルにユニークな値を割り当てられる。この番号は対応する応答 PDU が戻ってきた時に確認するために使われる。

表 8-18/JT-T124 ConferenceAddRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
ネットワーク アドレス (Network Address)	要求	指示
要求側ノード(Requesting Node)	要求側 GCC プロバイダ	指示
タグ(Tag)	要求側 GCC プロバイダ	宛先 GCC プロバイダ
追加実行 MCU(Adding MCU)(オプション)	要求	トップ GCC プロバイダ
ユーザ データ(User Data)(オプ ション)	要求	指示

ConferenceAddRequest PDU の受信により、トップ GCC プロバイダは会議プロファイルに示されるようにノードを会議に追加するのに必要な特権を持つノードかどうかを最初にチェックする。もしなければ、それは結果として「無効な要求者(invalid-requester)」を示す ConferenceAddResponsePDU を要求側ノードへ送信する事によって要求を拒否しなければならない。ConferenceAddResponse PDU の内容を表 8-19/JT-T124 に示す。もし、要求側ノードが十分な特権を持っていたならば、トップ GCC プロバイダは、その時、追加実行 MCU(Adding MCU)パラメータが存在するかどうかを検査する。もし存在しないなら、もしくは追加実行 MCU(Adding MCU)識別子がトップ GCC プロバイダのノード ID と同じなら、例えば GCC-CONFERENCE-ADD-機能をサポートするトップ GCC プロバイダは、GCC-CONFERENCE-ADD-指示プリミティブを生成し、それをコントロール GCCSAP に対して発行する。プリミティブの内容は、受信した PDU の内容から構成される。もしそのノードがこの機能をサポートしないならば、それは、応答の中に受信した ConferenceAddRequest を含む FunctionNotSupportedResponse PDU を発行する事によって要求を拒否するかもしれない。

もし、オプションの追加実行 MCU(Adding MCU)パラメータが存在し、トップ GCC プロバイダのノード ID のそれより他の値を設定してあるならば、トップ GCC プロバイダは追加実行 MCU に ConferenceAddRequest PDU を送信する。PDU の送信は、チャンネル ID が追加実行 MCU のノード ID で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。その PDU の内容は受信した要求 PDU のそれと同じである。ConferenceAddRequest PDU の受信により、トップ GCC プロバイダ以外のノードは、もし、GCC-CONFERENCE-ADD 機能をサポートしているなら、受信した MCS-SEND-DATA 指示からユーザ ID を検査する事により、トップ GCC プロバイダから受信した事を最初にチェックする。もしそれがトップ GCC プロバイダから受信したなら、それは GCC-CONFERENCE-ADD-指示プリミティブを生成し、コントロール GCCSAP に対して発行する。プリミティブの内容は、受信した PDU の内容から構成される。もし受信した PDU のユーザ ID がトップ GCC プロバイダのノード ID と一致しないならば、受信した PDU は無視され、以後の動作を行わない。もしそのノードがこの機能をサポートしないなら、それは、応答の中に受信した ConferenceAddRequest を含む FunctionNotSupportedResponse PDU を発行する事によって要求を拒絶するかもしれない。

GCC-CONFERENCE-ADD- 応答プリミティブの受信時に、GCC プロバイダは要求側ノードに ConferenceAddResponse PDU を送信する。PDU の送信は、チャンネル ID が要求側ノードのノード ID で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。要求が他のノードへトップ GCC プロバイダを通して発送されたなら、応答は、要求 PDU の要求側ノード(Requesting

Node)パラメータによって示されるオリジナルの要求側ノードに発行される。ConferenceAddResponse PDU の内容を表 8-19/JT-T124 に示す。その内容は、GCC-CONFERENCE-ADD-応答プリミティブの中を通ったパラメータから構成される。そのタグ(Tag)は受信した ConferenceAddRequest PDU の対応するパラメータと同じ値を持つ。

表 8-19/JT-T124 ConferenceAddResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
タグ(Tag)	ソース GCC プロバイダ	宛先 GCC プロバイダ
結果(Result)	応答	確認
ユーザ データ(User Data)(オプション)	応答	確認

ConferenceAddResponse PDU の受信により、GCC プロバイダは GCC-CONFERENCE-ADD-確認プリミティブを生成し、コントロール GCCSAP に発行する。そのプリミティブの内容は受信した PDU のパラメータから得られる。

8.2.6. 会議のロック

GCC-CONFERENCE-LOCK-要求プリミティブの受信時に、GCC プロバイダは ConferenceLockRequest PDU を送信する。PDU の送信は、チャンネル ID (Channel ID)がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceLockRequest PDU の内容を表 8-20/JT-T124 に示す。この PDU はパラメータを含まない。

表 8-20/JT-T124 ConferenceLockRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
パラメータなし		

ConferenceLockRequest PDU の受信時に、トップ GCC プロバイダが会議ロック能力をサポートするならば、それは会議が生成された時に定義されたロック/アンロック特権に基づく会議をロックするために必要な特権を要求ノードが持っているかを最初に判断しなければならない。もしそうならば、トップ GCC プロバイダは GCC-CONFERENCE-LOCK-指示プリミティブを生成し、それをコントロール GCCSAP に発行しなければならない。プリミティブ中に記述された発信元ノード(Source Node)は MCS-SEND-DATA-指示の中の送信ユーザ ID(Sender User ID)から得られる。GCC-CONFERENCE-LOCK-応答プリミティブの受信時に、GCC プロバイダは ConferenceLockResponse PDU を送信する。PDU の送信は、チャンネル ID (Channel ID)が応答中の発信元ノードで、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceLockResponse PDU の内容を表 8-21/JT-T124 に示す。結果(Result)パラメータは GCC-CONFERENCE-LOCK-応答で返送された結果(Result)より生成される。

要求側ノードがこの動作をサポートするための特権を持っていないならば、要求は GCC-CONFERENCE-LOCK-指示の生成をせずに直ちに拒否される。これは、結果として「無効な要求者 (invalid-requester)」を記述した ConferenceLockResponse PDU を生成することによって行われる。

トップ GCC プロバイダは、コントロール GCCSAP からの応答プリミティブと転送された ConferenceLockResponse PDU、ConferenceUnlockResponse PDU の間の順序とともに、受信した ConferenceLockRequest PDU、

ConferenceUnlockRequest PDU とコントロール GCCSAP に発行した対応する指示プリミティブの順序を保存する。

要求側ノードにおいては、受信した ConferenceLockResponse PDU、ConferenceUnlockResponse PDU と対応する確認プリミティブの間の順序とともに、要求プリミティブと対応して転送された ConferenceLockRequest PDU、ConferenceUnlockRequest PDU の間の順序も保存する。

トップ GCC プロバイダが会議ロック能力をサポートしないならば、ConferenceLockRequest PDU の受信時に、応答として受信した ConferenceLockRequest PDU を含む FunctionNotSupportedResponse PDU を直ちに生成しなければならない。

表 8-21 /JT-T124 ConferenceLockResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
結果(Result)	応答	確認

ConferenceLockResponse PDU の受信時に、GCC プロバイダは GCC-CONFERENCE-LOCK-確認プリミティブを生成し、コントロール GCCSAP へ発行しなければならない。確認プリミティブの中の結果(Result)パラメータは PDU の中の結果(Result)フィールドから得られる。

トップ GCC プロバイダが成功の結果を示す ConferenceLockResponse を送信するなら、トップ GCC プロバイダは ConferenceLockIndication PDU を生成し、それを全てのノードに送信する。PDU の送信は、チャンネル ID(Channel ID) が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。 ConferenceLockIndication PDU の内容を表 8-22/JT-T124 に示す。この PDU はパラメータを含まない。

表 8-22 /JT-T124 ConferenceLockIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
パラメータなし		

ConferenceLockIndication PDU の受信時に、オプションのロック指示をサポートする GCC プロバイダは GCC-CONFERENCE-LOCK-REPORT-指示を生成し、コントロール GCCSAP に発行できる。そうする前に受信した MCS-UNIFORM-SEND-DATA-指示の中で指示されるユーザ ID を検査し、トップ GCC プロバイダのノード ID(Node ID) と比較しなければならない。GCC-CONFERENCE-LOCK-REPORT プリミティブは、ユーザ ID がトップ GCC プロバイダのノード ID(Node ID) と一致するならば、生成される。さもなければ受信した PDU は無視されて、何も以後の動作が起こらない。

トップ GCC プロバイダが会議名簿の中のその存在によって会議に入っている新しいノードに気づき、会議が現在はロックされていて、ノードがその会議に加入した(または招かれた)時、アンロックされていた可能性があるならば、トップ GCC プロバイダは ConferenceLockIndication PDU を生成し、新しいノードにそれを送信する。PDU の送信は、チャンネル ID (Channel ID)がそのノード ID(Node ID)で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。選択的に、この PDU を全てのノードに送信できる。その場合の PDU の送信は、チャンネル ID (Channel ID)が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。会議がロックかアンロックであるかを示すフラグが接続確立プロセスの一部として新しく参加したノードに提供さ

れた情報に含まれるが、これは、接続が確立してから会議がそのロック状態を変更したならば、ノードが正確にその変更を通知されることを保証する。会議へのノードの加入(又は招待)以降にロック状態が変化しなかった可能性がないなら(例えば、会議の生成以降ロック状態が変化していないなら)、トップ GCC プロバイダはこの PDU の送信を必要としない。

8.2.7. 会議のアンロック

GCC-CONFERENCE-UNLOCK-要求プリミティブの受信時に、GCC プロバイダは ConferenceUnlockRequest PDU を送信する。PDU の送信は、チャンネル ID (Channel ID) がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceUnlockRequest PDU の内容を表 8-23/JT-T124 に示す。この PDU にパラメータは存在しない。

表 8-23/JT-T124 ConferenceUnlockRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
パラメータなし		

ConferenceUnlockRequest PDU の受信時に、トップ GCC プロバイダが会議アンロック能力をサポートするならば、要求ノードが会議が生成された時に定義されたロック/アンロック特権に基づく会議をアンロックするために必要な特権を持っているかを最初に判断しなければならない。もしそうならば、トップ GCC プロバイダは GCC-CONFERENCE-UNLOCK-指示プリミティブを生成し、それをコントロール GCCSAP に発行しなければならない。プリミティブ中に記述された発信元ノード(Source Node)は MCS-SEND-DATA-指示の中の送信ユーザ ID(Sender User ID) から得られる。GCC-CONFERENCE-UNLOCK-応答プリミティブの受信時に、GCC プロバイダは ConferenceUnlockResponse PDU を送信する。PDU の送信は、チャンネル ID (Channel ID) が応答中の発信元ノードで、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceUnlockResponse PDU の内容を表 8-24/JT-T124 に示す。結果(Result)パラメータは GCC-CONFERENCE-UNLOCK-応答で返送された結果(Result)より生成される。

要求側ノードがこの動作をサポートするための特権を持っていないならば、要求は GCC-CONFERENCE-UNLOCK-指示の生成をせずに直ちに拒否される。これは、結果として「無効な要求者 (invalid-requester)」を記述した ConferenceUnlockResponse PDU を生成することによって行われる。

トップ GCC プロバイダは、コントロール GCCSAP からの応答プリミティブと転送された ConferenceLockResponse PDU、ConferenceUnlockResponse PDU の間の順序とともに、受信した ConferenceLockRequest PDU、ConferenceUnlockRequest PDU とコントロール GCCSAP に発行した対応する指示プリミティブの順序を保存する。

要求側ノードにおいては、受信した ConferenceLockResponse PDU、ConferenceUnlockResponse PDU と対応する確認プリミティブの間の順序とともに要求プリミティブと対応して転送された ConferenceLockRequest PDU、ConferenceUnlockRequest PDU の間の順序も保存する。

トップ GCC プロバイダが会議ロック能力をサポートしないならば、ConferenceUnlockRequest PDU の受信時に、応答として受信した ConferenceUnlockRequest PDU を含む FunctionNotSupportedResponse PDU を直ちに生成しなければならない。

表 8-24/JT-T124 ConferenceUnlockResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
結果(Result)	応答	確認

ConferenceUnlockResponse PDU の受信時に、GCC プロバイダは GCC-CONFERENCE-UNLOCK-確認プリミティブを生成し、コントロール GCCSAP へ発送しなければならない。確認プリミティブの中の結果(Result)パラメータは PDU の中の結果(Result)フィールドから得られる。

トップ GCC プロバイダが成功の結果を示す ConferenceUnlockResponse を送信するなら、トップ GCC プロバイダは ConferenceUnlockIndication PDU を生成し、それを全てのノードに送信する。PDU の送信は、チャンネル ID (Channel ID) が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。ConferenceUnlockIndication PDU の内容を表 8-25/JT-T124 に示す。この PDU はパラメータを含まない。

表 8-25/JT-T124 ConferenceUnlockIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
パラメータなし		

ConferenceUnlockIndication PDU の受信時に、オプションのアンロック指示をサポートする GCC プロバイダは GCC-CONFERENCE-LOCK-REPORT-指示を生成し、コントロール GCCSAP に発行できる。そうする前に受信した MCS-UNIFORM-SEND-DATA-指示の中で指示されるユーザ ID を検査し、トップ GCC プロバイダのノード ID(Node ID)と比較しなければならない。GCC-CONFERENCE-LOCK-REPORT プリミティブは、ユーザ ID がトップ GCC プロバイダのノード ID(Node ID)と一致するなら、生成される。さもなければ受信した PDU は無視されて、何も以後の動作が起こらない。

トップ GCC プロバイダが会議名簿の中のその存在によって会議に入っている新しいノードに気づき、会議が現在ではアンロックされていて、ノードがその会議に加入した(または招かれた)時ロックされていた可能性があるならば、トップ GCC プロバイダは ConferenceUnlockIndication PDU を生成し、新しいノードにそれを送信する。PDU の送信は、チャンネル ID (Channel ID) がそのノード ID(Node ID)で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。選択的に、この PDU を全てのノードに送信できる。その場合の PDU の送信は、チャンネル ID (Channel ID) が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。会議がロックかアンロックかであるかを示すフラグが接続確立プロセスの一部として新しく参加したノードに提供された情報に含まれるが、これは、接続が確立してから会議がロック状態を変更したならば、ノードが正確にその変更を通知される事を保証する。会議への参加(又は招待)以降にロック状態が変化しなかった可能性がないなら(例えば、会議の生成以降にロック状態が変化していないなら)、トップ GCC プロバイダはこの PDU の送信を必要としない。

8.2.8. 会議からの切断

GCC-CONFERENCE-DISCONNECT-要求の受信時に、存在するならば、GCC プロバイダはコネクション階層構造における直接下のノードを排除することを最初に試みる。そのような各ノードに対して、排除するノード(Node To Eject)パラメータに特定の下位のノードのノード ID(Node ID)を設定し、理由が高位ノードの切断(higher-node-disconnected)の ConferenceEjectUserIndication PDU を送信することでこれを行うべきである。PDU の送信は、チャンネル ID (Channel ID) が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに

PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。ConferenceEjectUserIndication PDU の内容を表 8-31/JT-T124 に示す。GCC プロバイダは下位の各コネクションから MCS-DISCONNECT-PROVIDER を受信するまで待たなければならない。これに続き、上位のコネクション(唯一の残っているコネクション)に直接向けられた MCS-DISCONNECT-PROVIDER-要求にしたがって、最初に MCS-DETACH-USER-要求を発行することによって会議から切断しなければならない。いくつかの理由で GCC プロバイダが(ローカルに決定された)十分な時間内に各低位ノードから MCS-DISCONNECT-PROVIDER-指示を受信しなかったならば、GCC プロバイダは正常ケースとして上位のコネクションへ向かう MCS-DETACH-USER-要求と MCS-DISCONNECT-PROVIDER-要求に続き、残っている下位の各コネクションに向かって MCS-DISCONNECT-PROVIDER-要求を発行することによってそれらのコネクションを自ら切断し始める。どちらかのケースでも、もしくは下位ノードが無いなら、GCC プロバイダは GCC-CONFERENCE-DISCONNECT-確認プリミティブを生成し、コントロール GCCSAP へ発行しなければならない。GCC プロバイダはその時、この会議に関連した全てのデータベース情報を削除しなければならない。

MCS-DETACH-USER-指示の受信時に、その会議中の各 GCC プロバイダはその指示の中で指示されているユーザ ID を検査し、会議名簿のローカルなコピーの中のノード ID(Node ID)のリストとそれを比較する。ユーザ ID が自分自身のノード ID(Node ID)とは違うノード ID(Node ID)に一致しているなら、GCC プロバイダは GCC-CONFERENCE-DISCONNECT-指示を生成し、コントロール GCCSAP へ発行しなければならない。その指示の中の切断ノード(Disconnecting Node)パラメータは受信した指示の中のユーザ ID と対応しなければならない。受信した指示の理由コードが「ユーザによる(user-initiated)」なら、排除するノード(Node To Eject)フィールド中に MCS-DETACH-USER-指示と同じユーザ ID を含んでいる ConferenceEjectUserIndication PDU が先に受信されたかどうかによって依存して、GCC-CONFERENCE-DISCONNECT-指示の理由コードは「ユーザによる(user-initiated)」か「排除されたノード(ejected node)」のどちらかになる。さもなければ、その理由は「不明(unknown)」として示される。

8.2.9. 会議の終了

GCC-CONFERENCE-TERMINATE-要求プリミティブの受信時に、GCC プロバイダはトップ GCC プロバイダに ConferenceTerminateRequest PDU を送信する。PDU の送信は、チャンネル ID (Channel ID)がトップ GCC のノード ID(Node ID)のチャンネルで、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceTerminateRequest の内容を表 8-26/JT-T124 に示す。理由コードは要求プリミティブの中の対応するパラメータから得られる。

表 8-26/JT-T124 ConferenceTerminateRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
理由(Reason)	要求	トップ GCC プロバイダ

ConferenceTerminateRequest PDU の受信時に、トップ GCC プロバイダは、要求ノードが会議が生成された時に定義される終了特権に基づいた会議を終了させるために必要な特権を持っているかどうかを最初に判断しなければならない。もし無いなら、要求は拒否され、ConferenceTerminateResponse は要求者に返送される。PDU の送信は、チャンネル ID (Channel ID)が要求者のユーザ ID で、データプライオリティが High で、データフィールドにその PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceTerminateResponse PDU の内容を表 8-27/JT-T124 に示す。この場合、結果(Result)パラメータは拒否の理由として「無効な要求者(Invalid-Requester)」を示すように設定される。

要求者が会議を終了させるための特権を持っていたら、その時は、ConferenceTerminateResponse は成功の結果を示して要求者に返送される。更に、ConferenceTerminateIndication は会議中の全ノードに送信される。PDU の送信は、チャンネル ID (Channel ID)が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU をふくめた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。ConferenceTerminateIndication PDU の内容を表 8-28/JT-T124 に示す。理由コードは要求 PDU 中の理由コードから得られる。

表 8-27/JT-T124 ConferenceTerminateResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
結果(Reason)	トップ GCC プロバイダ	確認

ConferenceTerminateResponse PDU の受信時に、GCC プロバイダは GCC-CONFERENCE-TERMINATE-確認プリミティブを生成し、コントロール GCCSAP に発送しなければならない。プリミティブに示される結果はその PDU 中の結果(Result)パラメータから直接得られる。

表 8-28/JT-T124 ConferenceTerminateIndecation GCCPDU

(ITU-T T.124)

内容	ソース	シンク
理由(Reason)	トップ GCC プロバイダ	指示

トップ GCC プロバイダより送られた ConferenceTerminateIndication PDU の受信時に、GCC プロバイダは、最初に、コネクション階層構造の直接下位の各コネクションから MCS-DISCONNECT-PROVIDER-指示を受信するまで待たなければならない。これに続き、上位のコネクション(唯一残っているコネクション)に向けられた MCS-DISCONNECT-PROVIDER-要求を発生することによって会議から切断しなければならない。いくつかの理由で、GCC プロバイダが(ローカルに決定された)十分な時間内に各低位ノードから MCS-DISCONNECT-PROVIDER-指示を受信しなかったならば、GCC プロバイダは正常ケースとして上位のコネクションへ向かう MCS-DISCONNECT-PROVIDER-要求に続き、残っている下位の各コネクションに向かって MCS-DISCONNECT-PROVIDER-要求を発行することによってそれらのコネクションを自ら切断し始める。どちらかのケースでも、GCC プロバイダはその時 GCC-CONFERENCE-TERMINATE-指示プリミティブを生成し、コントロール GCCSAP へ発行しなければならない。そのプリミティブで示された理由は PDU 中の理由(Reason)パラメータから直接得られる。

GCC プロバイダが上位の MCS コネクションに対応するローカルな MCS プロバイダから MCS-DISCONNECT-PROVIDER-指示を受信するなら、これは MCS コネクションが MCS 内部の異常のために終了したことを示し、GCC プロバイダは GCC-CONFERENCE-TERMINATE-指示プリミティブを生成し、コントロール GCCSAP へ発行されなければならない。理由コードはこれが異常終了であることを示す。

トップ GCC プロバイダがローカルなノードを除いて会議名簿に何も記入されていない結果の(PDU か MCS-DETACH-USER-指示を経由して)切断指示を受信したら、プロバイダは会議が手動終了であるか自動終了であるかを判断するために会議プロファイルをチェックする。手動終了ならば、以後の動作は何も起こらない。自動終了ならば、GCC プロバイダは GCC-CONFERENCE-TERMINATE-指示を生成し、コントロール GCCSAP に発行することによって、その会議が終了されたことをローカルなノードコントローラに通知しなければならない。理由コードは自動終了会議に加入するノードのないことを示す。GCC プロバイダはその時この会議と関連した全てのデータベース情報を削除し

なくてはならない。

8.2.10. 会議からのノードの排除

GCC-CONFERENCE-EJECT-USER-要求プリミティブの受信時に、GCC プロバイダは、排除するノード(Node To Eject)パラメータをコネクション階層構造内で直接下位のノードのノード ID(Node ID)を比較します。排除するノード(Node To Eject)が直接下位のノードでないならば、GCC プロバイダはトップ GCC プロバイダに ConferenceEjectUserRequest PDU を送信する。PDU の送信は、チャンネル ID (Channel ID)がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが Top で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceEjectUserRequest PDU の内容は表 8-29/JT-T124 に示される。PDU の内容は要求プリミティブから得られる。

排除するノード(Node To Eject)がコネクション階層構造内のローカルノードの直接下位のノードならば、GCC プロバイダは代わりに、PDU 内に排除されるノードのノード ID(Node ID)と要求プリミティブ内で指示された理由を指定する ConferenceEjectUserIndication PDU をすべてのノードに送信します。PDU の送信は、チャンネル ID(Channel ID)が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-UNIFORM-DATA-要求を発行することにより行われる。ConferenceEjectUserIndication の内容は表 8-31/JT-T124 に示されている。それから GCC プロバイダは排除されるノードに対応するコネクションから MCS-DISCONNECT-PROVIDER-指示を受信するまで待つ。いくつかの理由により、GCC プロバイダが(ローカルに決定された)十分な時間内に、排除されるノードから MCS-DISCONNECT-PROVIDER-指示を受信しなければ、GCC プロバイダは、排除されるノードに直接 MCS-DISCONNECT-PROVIDER-要求を発行することにより、それらのコネクションの切断を自ら始める。それから GCC プロバイダは成功を示す GCC-CONFERENCE-EJECT-USER 確認プリミティブ⁶を生成する。

表 8-29/JT-T124 ConferenceEjectUserRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
排除されるノード(Node To Eject)	要求	トップ GCC プロバイダ
理由(Reason)	要求	トップ GCC プロバイダ

ConferenceEjectRequest PDU の受信時に、トップ GCC プロバイダは最初に、要求ノードが会議が生成された時に定義されるユーザ排除特権(eject-user-privileges)に基づいたユーザを排除するのに必要な特権を持つかどうかを確認する。持たなければ要求は拒否され、ConferenceEjectUserResponse PDU を要求者に返送される。PDU の送信は、チャンネル ID (Channel ID)が要求者の NODE-ID-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含む MCS-SEND-DATA-要求を発行することにより行われる。ConferenceEjectUserResponse PDU は表 8-30/JT-T124 に示される。この場合、結果パラメータは拒否の理由として「無効な要求者(invalid-requester)」を設定する。

⁶ 原文ではGCC-CONFERENCE-EJECT-USER-確認PDUとあるが、これは明らかに「PDU」ではなく「プリミティブ」の誤りであると思われる。(TTC標準独自脚注)

要求者が、ユーザを排除するための適当な特権を保有しているならば、全ノードに対して ConferenceEjectUser Indication PDU が放送される。PDU の送信は、チャンネル ID (Channel ID)が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。ConferenceEjectUser IbdicationPDU の内容は表 8-31/JT-T124 に示される。この PDU 中の理由パラメータは要求 PDU から得られる。排除されるノードが妥当ならば、応答は要求者に返送され成功結果を示す。要求されたノードを排除するのが不可能ならば、応答は否定結果を含めて送信される。

表 8-30/JT-T124 ConferenceEjectUserResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
排除されるノード (Node To Eject)	トップ GCC プロバイダ	確認
理由 (Reason)	トップ GCC プロバイダ	確認

ConferenceEjectUserResponse PDU の受信時に、GCC プロバイダは GGC-CONFERENCE-EJECT-USER-確認プリミティブを生成し、コントロール GCCSAP にそれを発行する。確認プリミティブの内容は ConferenceEjectUserResponse PDU から得られる。

表 8-31/JT-T124 ConferenceEjectUserIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
排除されるノード (Node To Eject)	トップ GCC プロバイダ	確認
理由 (Reason)	トップ GCC プロバイダ	指示

ConferenceEjectUserIndication PDU の受信時に、GCC プロバイダは排除するノード(Node To Eject)パラメータと自身のノード ID(Node ID)を比較する。同じならば、それから、MCS-SEND-DATA-指示に示されているユーザ ID を、トップ GCC プロバイダのノード ID(Node ID)とコネクション階層構造中の直接上位のノードのそれとを比較する。送信元のノード ID(Node ID)がそれらのいずれかと同じならば、コネクション階層構造中の直接下位のノードを排除しようとする最初の試みがあれば、これによって直ちに会議から切断する。これは、排除するノード(Node To Eject)パラメータを特定の下位ノードのノード ID(Node ID)に設定し、理由として「上位ノードの排除(higher-node-ejected)」を指定する ConferenceEjectUserIndication PDU を、そのような各ノードに対して送信することによってなされる。PDU の送信はチャンネル ID (Channel ID)が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含む MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。それから、GCC プロバイダは各下位コネクションから MCS-DISCONNECT-PROVIDER-指示を受信するまで待つ。続いて、上位のコネクション (唯一残っているコネクション)に向けられた MCS-DISCONNECT-PROVIDER-要求が発行された後、MCS-DETACH-USER-要求を最初に発行することによって会議から切断する。いくつかの理由により、(ローカルに決定される)十分な時間内に各下位ノードから MCS-DISCONNECT-PROVIDER-指示を受信しなかったならば、正常ケース

として、上位のコネクションに向かう MCS-DISCONNECT-PROVIDER-要求に続き、最初の MCS-DETACH-USER-要求を発行し、残っている各下位コネクションに向かう MCS-DISCONNECT-PROVIDER-要求を発行することにより、GCC プロバイダは自らこれらのコネクションの切断を進める。どちらの場合でも、また下位ノードが無いならば、それから GCC プロバイダは GCC-CONFERENCE-EJECT-USER-指示プリミティブを生成し、コントロール GCCSAP にそれを発行する。トップ GCC プロバイダまたはコネクション階層構造中の直接上位のノードのノード ID(Node ID) と一致しないユーザ ID で、PDU が受信されるならば、さらなる動作をなされず PDU は無視される。

受信する GCC が、コネクション階層構造上で排除されるノードの上位に直接接続している MCU ならば、相当する MCS コネクションに対し MCS-DISCONNECT-PROVIDER-要求を発行することにより会議から排除されるノードを任意に切断できる。このような動作が取られる前に、受信した PDU 中に示されるユーザ ID がトップ GCC プロバイダのノード ID(Node ID)と同じであることを確かめる。

トップ GCC プロバイダが会議名簿中に排除されるノードの上位として登録されているノードのノード ID(Node ID) により送信された ConferenceEjectUserIndication PDU を受信したとき、排除されるノードを除く全ノードは、このイベントを記録し、MCS-DETACH-USER-指示が排除されるノードに着いたならば、あとでそれを参照する。その時、8.2.8節で示されているように、ノードはGCC-CONFERENCE-DISCONNECT-指示を生成し、コントロール GCCSAP にそれを発行する。示される理由は「排除されたノード(ejected-node)」である。

8.2.11. 会議間のノードの移動

GCC-CONFERENCE-TRANSFER-要求プリミティブ受信時に、GCC プロバイダはトップ GCC プロバイダに、ConferenceTransferRequest PDU を送信しなければならない。PDU の送信は、チャンネル ID(Channel ID)がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceTransferRequest PDU の内容を表 8-32/JT-T124 に示す。この PDU のすべてのパラメータは要求プリミティブの内容から直接得られる。

表 8-32/JT-T124 ConferenceTransferRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
会議名(Conference Name)	要求	トップGCCプロバイダ
会議名修飾子 (Conference Name Modifier) (オプション)	要求	トップGCCプロバイダ
ネットワークアドレス (Network Address) (オプション)	要求	トップGCCプロバイダ
移動ノード(Transferring Nodes) (オプション)	要求	トップGCCプロバイダ
パスワード>Password) (オプション)	要求	トップGCCプロバイダ

ConferenceTransferRequest PDUを受信時に、トップGCCプロバイダはまず、会議が生成されたときに定義された移動特権(transfer-privileges)に基づき、要求したノードが移動を要求するために必要な特権を持つかどうかを確認する。もし持たなければ、要求は拒否され、要求者に ConferenceTransferResponse を返送する。PDUの送信は、チャンネルID(Channel ID)が要求者のNODE-ID-CHANNELで、データプライオリティがHighで、データ(Data)フィールドにPDUを含めたMCS-SEND-DATA-要求を発行することにより行われる。ConferenceTransferResponse PDUの内容を表8-33/JT-T124に示す。この場合、結果パラメータは拒否の理由として「無効な要求者(invalid-requester)」が設定される。

要求者が、移動を要求するための適当な特権を保有しているならば、その後、トップGCCプロバイダは上述のように要求者に ConferenceTransferResponse PDUを返送する。ただし、成功を示す結果パラメータを持つ。それから、チャンネルID(Channel ID)がGCC-BROADCAST-CHANNELで、データプライオリティがHighで、データ(Data)フィールドにPDUを含めたMCS-UNIFORM-SEND-DATA-要求を発行することによって、ConferenceTransferIndication PDUを会議内のすべてのノードに放送する。ConferenceTransferIndication PDUの内容は表8-34/JT-T124に示す。このPDUのパラメータは受信した ConferenceTransferRequest PDUから得られる。

表 8-33/JT-T124 ConferenceTransferResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
会議名(Conference Name)	トップGCCプロバイダ	確認
会議名修飾子 (Conference Name Modifier) (オプション)	トップGCCプロバイダ	確認
移動ノード(Transferring Nodes) (オプション)	トップGCCプロバイダ	確認
結果(Result)	トップGCCプロバイダ	確認

ConferenceTransferResponse PDU を受信時に、GCC プロバイダは GCC-CONFERENCE-TRANSFER-確認プリミティブを生成し、コントロール GCCSAP に発行する。確認プリミティブの内容は、ConferenceTransferResponse PDU から得られる。

表 8-34/JT-T124 ConferenceTransferIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
会議名(Conference Name)	トップGCCプロバイダ	指示
会議名修飾子 (Conference Name Modifier) (オプション)	トップGCCプロバイダ	指示
ネットワークアドレス (Network Address) (オプション)	トップGCCプロバイダ	指示
移動ノード(Transferring Nodes) (オプション)	トップGCCプロバイダ	指示
パスワード>Password) (オプション)	トップGCCプロバイダ	指示

ConferenceTransferIndication PDU を受信時に、GCC-CONFERENCE-TRANSFER-指示プリミティブをサポートする GCC プロバイダは、宛先ノードのリストを調べる。ローカルなノード ID が宛先ノードのリスト中に見つかるか、宛先ノードのリストが NULL ならば、それから、GCC プロバイダは MCS-UNIFORM-SEND-DATA-指示が示すユーザ ID がトップ GCC プロバイダのノード ID に一致しているかを調べる。これらが一致しているならば、GCC-CONFERENCE-TRANSFER-指示プリミティブを生成し、コントロール GCCSAP にそれを発行する。GCC プロバイダが GCC-CONFERENCE-TRANSFER-指示プリミティブをサポートしていない場合や、宛先ノードのリスト上にローカルノードがない場合、または、受信した PDU がトップ GCC プロバイダからのものでない場合、さらなる動作はとられず PDU は無視される。

8.3. 会議名簿とアプリケーション名簿

8.3.1. 概要

会議名簿とアプリケーション名簿は両方とも、同じ組み合わせの PDU を用いることにより、ノード間で伝達される。ただ一つの PDU、RosterUpdateIndication PDU がこの交換の全ての局面に関係している。この PDU は完全な、または部分的な名簿情報を会議内の他のノードに送信するために使われる。

会議に加入し、各ノード(トップノード以外)が最初にその会議情報を告知するとき、それは、階層上で直接上位のノードに RosterUpdateIndication を送ることによってなされる。会議名簿かアプリケーション名簿のいずれかの一部の以降の変更は、新しい情報を含む RosterUpdateIndication を再発行することにより、告知される。

ノードがその名簿の一部を更新する（または、最初に告知する）とき、その情報は、会議名簿とアプリケーション名簿の全体を構成し会議内の全ノードにそれらを分配することに責任を持つトップ GCC プロバイダに到達するまで、ノードからコネクション階層構造上の上位のノードへ伝搬する。

名簿情報がコネクション階層構造の上位に伝搬するとき、各中間ノード（MCU）は会議名簿とアプリケーション名簿全体に対するサブセットを構成する責任を持つ。このサブセットはそのノードとコネクション階層構造内の下位のすべてのノードを含む。すなわち、会議名簿はそのような全てのノードのノードレコードを含み、各アプリケーション プロトコル セッションに対するアプリケーション名簿のエントリは同位のアプリケーション プロトコル エンティティが登録されたそのような各ノードに対するアプリケーション レコードを含み、各アプリケーション プロトコル セッションに対するアプリケーション能力リストはそのようなすべてのノードの集約された能力情報を含む。下位のノードからの RosterUpdateIndication を受信したとき、中間ノードは名簿のサブセットに適当な変更をなし、それから、RosterUpdateIndication を発行することにより、次の上位のノードにこの情報を渡す。

下位のノードからの更新が少なくとも 1 つのアプリケーション プロトコル セッションに対するアプリケーション能力リストの再計算を必要とする変更を含んでいるならば、コネクション階層構造中の直接下位の各ノードに対する個々のアプリケーション能力リストが知られているときのみ（新しい更新情報に集約ルールを再適用するため）、名簿サブセットのための新しい名簿情報を生成することができる。各ノードの GCC プロバイダはローカルにこの情報を維持しなければならない。

名簿更新がトップ GCC プロバイダに到達したならば、その後新しい名簿情報は会議内のすべてのノードに放送される。これは、すべてのノードに RosterUpdateIndication PDU を放送することによってなされる。

RosterUpdateIndication PDU を用いて設定される名簿情報は、3 つの方法のうちの 1 つで送信される。

- ・名簿は会議名簿及びアプリケーション名簿の全体のリフレッシュとして送信され、存在する全ての名簿エントリを置き換える。
- ・名簿の一部が、会議名簿、および（または）、1 つ以上のセッション キーに関係したアプリケーション名簿の一部がリフレッシュされることによって送信される。この場合、名簿の送信された部分のすべてのエントリは置き換えられるが、送信されなかった部分は変更されないまま残る。
- ・追加、変更、削除された名簿のこれらの特定の要素だけ（例えば、個々のアプリケーション レコード）を送信することによって、名簿の一部の変更は更新として送信され、他のエントリ全てを変更しないで残す。

トップ GCC プロバイダが新しい名簿情報を放送するときは、最初の 2 つの方法だけが用いられる。少なくとも 1 つのノードが会議に加わったために変更された新しい会議情報を、トップ GCC プロバイダが放送するとき、新しいノードは以前の名簿情報を持っていないので、フル リフレッシュとしてのみ、この情報を放送する。コネクション階層構造を上方に名簿情報が伝搬しているとき、これらの 3 つの方法のどれでも使うことができる。方法は、変更される情報の範囲によって選択され、理想的には、最少伝送時間のために PDU のサイズの最少化の基本によって選択される。

注- 記述された機構には、会議内の全ての GCC プロバイダが、コネクション階層構造内の直接下位のノードと同様に自身のノードに対する会議名簿とアプリケーション名簿のサブセットを蓄えていることが必要である。加えて、種々

の機能は、GCC プロバイダがトップ GCC プロバイダからの放送として会議名簿とアプリケーション名簿の全てをも蓄えることを要求する。何等かの理由で、GCC プロバイダがこの情報の幾つか、または全てを保持するのに失敗したならば、唯一の手段は会議から切断し、可能ならば、その後再加入することである。

8.3.2. ノードの会議への参加

会議の生成か、加入か、招待かを介して、ノードが会議に加わるとき、GCC プロバイダは、ローカルに GCC プロバイダに対してその存在を示した全てのアプリケーション プロトコル エンティティの GCCSAP に、GCC-APPLICATION-PERMISSION-TO-ENROLL-指示を発行する。既に会議に加入しているとき、GCC プロバイダが追加のアプリケーション プロトコル エンティティを認識した場合、GCC プロバイダは対応する GCCSAP に、この会議の存在を示す GCC-APPLICATION-PERMISSION-TO-ENROLL-指示を発行する。

さらなる動作がとられる前に、GCC プロバイダは、GCC-APPLICATION-PERMISSION-TO-ENROLL-指示を送信した全てのアプリケーション プロトコル エンティティからの GCC-APPLICATION-ENROLL-要求と、コントロール GCCSAP からの GCC-CONFERENCE-ANNOUNCE-PRESENCE-要求を受信するまで待たねばならない。これらの登録要求のいくつかはアプリケーション プロトコル エンティティが登録するつもりがないこと（登録/登録解除(enroll/un-enroll)フラグが登録解除に設定されている）を示しているかもしれない。登録フラグが設定されている GCC-APPLICATION-ENROLL-要求を受信するたびに、プリミティブで供給される、そのアプリケーション プロトコル エンティティに関する情報をローカルアプリケーション名簿に加える。

GCC プロバイダはまた、ローカルに生成したエンティティ ID を登録した各アプリケーション プロトコル エンティティに割り当てる。これは、RosterUpdateIndication PDU 中の各アプリケーション プロトコル エンティティの識別子として GCC プロバイダによって用いられる。また、割り当てられたアプリケーション プロトコル エンティティ ID(Application Protocol Entity ID)は、ローカルノードのノード ID とともに、登録した各アプリケーション プロトコル エンティティに返される GCC-APPLICATION-ENROLL-確認プリミティブにも含まれる。エンティティ ID は、1 つのノードにおいて登録されている全アプリケーション プロトコル エンティティ間でユニークな 16 ビットの整数値である。あるアプリケーション プロトコル エンティティが登録解除するならば、そのエンティティ ID の値は、登録解除のときにアプリケーション プロトコル エンティティに割り当てられていなかった他の全ての値が割り当てられてしまったのでなければ、再利用してはならない。

GCC プロバイダは、活性/非活性(Active/Inactive)フラグが「活性(Active)」になっている登録された各アプリケーション プロトコル エンティティの議長動作能力(Conducting Operation Capable)フラグを調査する（非活性なアプリケーション プロトコル エンティティは議長動作の能力が無いと仮定される）。このフラグを設定して登録されたそのようなアプリケーション プロトコル エンティティが 1 つ以上あるならば、GCC プロバイダは RosterUpdateIndication PDU で送信されるアプリケーション名簿情報中でこのフラグを設定するために、それらの 1 つを選択しなければならない。これらのいずれを選択するかという規則はローカルな項目であり、この標準では指定しない。典型的な規則は、最初に登録されたそのようなアプリケーション プロトコル エンティティが選択されるものである。

コントロール GCCSAP からの GCC-CONFERENCE-ANNOUNCE-PRESENCE-要求と、全アプリケーション プロトコル エンティティからの GCC-APPLICATION-ENROLL-要求を受信したとき、トップ GCC プロバイダでない

GCC プロバイダはコネクション階層構造上の直接上位の GCC プロバイダに RosterUpdateIndication PDU を送信する。これは、チャンネル ID(Channel ID)が宛先ノードの NODE-ID-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することによってなされる。コネクション階層構造上の直接上位のノードのノード ID は、会議に接続するとき、その会議の加入方法によって ConferenceCreateResponse、ConferenceJoinResponse または ConferenceInviteRequest のいずれかの PDU から決定される。上位ノードのノード ID の値は、上位ノード(Superior Node)パラメータとして PDU 中の会議名簿にもまた含まれる。RosterUpdateIndication PDU の内容を表 8-35/JT-T124 に示す。この場合、名簿はフル リフレッシュとして送信され、ローカルノードレコードと、すべての登録アプリケーション プロトコル エンティティに対するアプリケーション レコードと、登録アプリケーション プロトコル エンティティに相応する全アプリケーション プロトコル セッションに対するアプリケーション能力リストを含んでいる。あるアプリケーション プロトコル セッションにローカルに登録されたアプリケーション プロトコル エンティティが 1 つ以上ある場合には、そのアプリケーション プロトコル セッションに対する RosterUpdateIndication PDU 中に含まれるアプリケーション能力リスト(Application Capabilities List)を生成するために、GCC プロバイダは同位のアプリケーション プロトコル エンティティ間のアプリケーション能力リストの集約を実行する。この集約は、8.3.8 節で記述する手順によってなされる。RosterUpdateIndication を受信したノードによってとられる動作は 8.3.5 節に記述する。

会議にちょうど加わったばかりで GCC-CONFERENCE-ANNOUNCE-PRESENCE-要求を受信したトップ GCC プロバイダであるノードは、それに格納される情報をその会議名簿データベースに含めなければならないが、PDU を送信する前に、会議中の他の少なくとも 1 つのノードから RosterUpdateIndication PDU を受信するのを待ってもよい。そのような指示を受信したとき、その会議名簿及びアプリケーション名簿のデータベースを更新し、それから会議内の全ノードに RosterUpdateIndication PDU を送信することにより全ての会議名簿とアプリケーション名簿を放送する。これは、チャンネル ID(Channel ID)が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。新しいノードが会議に加わった場合、RosterUpdateIndication PDU はフル リフレッシュとして送信され、会議に加入している全ノードからのノードレコードと、会議内の各アプリケーション プロトコル セッションに対する集約したアプリケーション能力リストと、そのセッションを構成する同位アプリケーション プロトコル エンティティのリスト(および、それぞれのアプリケーション レコード)を含む。集約したアプリケーション能力リストを含む、全体の会議名簿とアプリケーション名簿を生成する規則は、8.3.5 節に記述する。

表 8-35/JT-T124 RosterUpdateIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
フル リフレッシュ フラグ (Full Refresh Flag)	発行元GCCプロバイダ	宛先GCCプロバイダ
会議情報 (Conference Information) (オプション)	発行元GCCプロバイダ (要求、または、受信PDU 中の情報)	宛先GCCプロバイダ
アプリケーション情報 (Application Information) (オプション)	発行元GCCプロバイダ (要求、または、受信PDU 中の情報)	宛先GCCプロバイダ

8.3.3.アプリケーション プロトコル エンティティの登録

GCC-APPLICATION-ENROLL-要求プリミティブの受信時、まず最初に GCC プロバイダは、指定された会議のローカルアプリケーション名簿のなかに、要求者の GCCSAP に対して割り当てられたエントリがすでに存在するかどうかを確認しなければならない。存在しない場合で、また、要求中の登録/登録解除(Enroll/un-enroll)フラグが「登録(Enroll)」にセットされている場合、要求プリミティブに指定されている情報を含んでいる新しいエントリをローカルアプリケーション名簿の中に生成する。この場合、GCC プロバイダは、8.3.2 節に記述されているように、新たに登録されたアプリケーション プロトコル エンティティにエントリ ID を割り当てなければならない。登録/登録解除(Enroll/Un-enroll)フラグが「登録解除(Un-enroll)」にセットされた場合、その要求は即座に GCC-APPLICATION-ENROLL-確認を生成し要求者の GCCSAP へ発行することにより確認される。この場合、それ以上の動作はとられない。

ローカルアプリケーション名簿中にエントリが既に存在し、そして登録/登録解除(Enroll/un-enroll)フラグが「登録(Enroll)」にセットされている場合、存在するエントリの内容は指定された新しい値を反映すべく変更される。登録/登録解除(Enroll/un-enroll)フラグが「登録解除(Un-enroll)」にセットされた場合、GCC プロバイダはローカルアプリケーション名簿から対応するエントリを削除しなければならない。

GCC プロバイダが指定された会議のトップ GCC プロバイダでない場合、GCC プロバイダは新しいローカルアプリケーション名簿を RosterUpdateIndication PDU を用いてコネクション階層構造の上位のノードに直接送る。それは、チャンネル ID(Channel ID)が上位ノードの NODE-ID-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに選択された PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。コネクション階層構造上の直接上位の GCC プロバイダのノード ID は、会議に接続するとき、その会議の加入方法によって ConferenceCreateResponse , ConferenceJoinResponse , ConferenceInviteRequest のいずれかの PDU から決定され

る。RosterUpdateIndication PDU の内容を表 8-35/JT-T124 に示す。

登録が RosterUpdateIndication PDU の最初の転送以前に起こった場合、新しい、もしくは、変更されたレコードが最初の名簿情報中に含まれ、8.3.2 節に記述されているフル リフレッシュとして送られる。アプリケーション プロトコル エンティティが最初の転送以前に登録解除された場合、そのレコードエントリは名簿から削除され、転送されるいずれの PDU にも含まれない。

RosterUpdateIndication PDU の最初の転送後に登録が起こった場合、新規、変更、もしくは、削除されるレコードはすでに存在する名簿への更新として扱われる。この場合、GCC プロバイダは会議及びアプリケーション名簿の相応する部分を名簿全体で再送するか、または、更新として情報を送る(変更したレコードに関する情報のみを送る)かのどちらかを選ぶ。PDU のサイズや転送時間を最小化するために後者の方法が好ましい。前回の更新以降に一つ以上の変更がされた場合(一つ以上の GCC-APPLICATION-ENROLL 要求を受信)、これらの変更は一つの PDU へまとめてもよい。あるアプリケーション プロトコル セッションについて、そのアプリケーション プロトコル セッションの最後のインスタンス以降にアプリケーション レコード情報のみ変更された場合、アプリケーション能力リストは RosterUpdateIndication PDU に含まれる必要はなく、変更されたアプリケーション レコードだけでよい。あるアプリケーション プロトコル セッションについて、そのアプリケーション能力リストの更新は以下の場合に必要となる。

- ・新たに登録されたアプリケーション プロトコル エンティティが、ローカルノードかまたはコネクション階層構造上のローカルノード以下のノードで、すでに登録されている同位アプリケーション プロトコル エンティティ用のアプリケーション プロトコル セッションに属す場合。
- ・ローカルノードかまたはコネクション階層構造上のローカルノード以下のノードで他にまだ登録されたままの同位アプリケーション プロトコル エンティティがあるときに、ノードが登録解除された場合。

登録後または登録解除以前に一つ以上の同位アプリケーション プロトコル エンティティがある場合、GCC プロバイダは RosterUpdateIndication PDU の中に格納する前にそのアプリケーション プロトコル セッションの集約されたアプリケーション能力リスト(Application Capabilities List)を再生成しなければならない。この集約は 8.3.8 節に記述されている手続きによりなされなければならない。

登録するアプリケーション プロトコル エンティティの活性/非活性(Active/Inactive)フラグが「活性(Active)」にセットされている場合、GCC プロバイダは RosterUpdateIndication PDU を送る以前に議長動作能力(Conducting Operation Capable)フラグをしらべなければならない。このフラグがセットされている場合、GCC プロバイダは、アプリケーション名簿中にこのフラグがセットされる同位アプリケーション プロトコル エンティティが、このノードからはアプリケーション プロトコル セッション当たり一つだけになることを確実にしなければならない。そのような同位アプリケーション プロトコル エンティティがそれまでにアプリケーション名簿中に選択されていなかった場合、あらたに登録しようとしているアプリケーション プロトコル エンティティはこのフラグがセットされている状態で、転送される RosterUpdateIndication PDU に含まれる。このフラグがセットされた状態でアプリケーション名簿に含まれている同位アプリケーション プロトコル エンティティがすでにこのノードにある場合、GCC プロバイダはフラグなしで RosterUpdateIndication PDU の中にあらたに登録するアプリケーション プロトコル エンティティを含めるか、もしくは、このフラグがセットされた状態で登録するアプリケーション プロトコル エンティティを PDU に含ませ、

そして以前に示された議長能力アプリケーション プロトコル エンティティのこのフラグを FALSE にする。どの同位アプリケーション プロトコル エンティティにこのフラグをセットしてアプリケーション名簿に含めるかという選択規則はこの標準の範囲外のローカルな問題である。登録するアプリケーション プロトコル エンティティの活性/非活性 (Active/Inactive) フラグが「非活性(Inactive)」にセットされている場合、この要求プリミティブ中の議長動作能力 (Conducting Operation Capable) フラグは無視されなければならない。そして更新された名簿のこれに対応するフィールドは FALSE にセットしなければならない。

トップ GCC プロバイダの場合、ローカルアプリケーション名簿中のエントリの正しい変更は、会議アプリケーション名簿の直接の変更となる。新しい会議アプリケーション名簿の会議のすべてのノードへの通知のための手続きは 8.3.5 節に記述される。

成功時、GCC プロバイダは、その成功結果を知らせる GCC-APPLICATION-ENROLL-確認プリミティブを生成し、要求者の GCCSAP へ発行する。

8.3.4. 会議名簿エントリの更新

GCC-CONFERENCE-ANNOUNCE-PRESENCE-要求が最初の名簿の転送後受信された場合、このレコードに含まれる新しいノードレコードは会議名簿への更新としてあつかわれる。アプリケーション レコードの更新の場合のように、GCC プロバイダは RosterUpdateIndication PDU を用い、コネクション階層構造上の直接上位のノードへ新しいノードレコードを送る。それは、チャンネル ID(Channel ID)が上位ノードの NODE-ID-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに選択された PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。RosterUpdateIndication PDU の内容を表 8-35/JT-T124 に示す。

登録の場合のように、GCC プロバイダは会議及びアプリケーション名簿の相応する部分を名簿全体で再送するか、または、更新として新たなノードレコードのみを送るかのどちらかを選ぶ。PDU サイズや転送時間を最小にするため、後者の方法が好ましい。

GCC プロバイダが、ローカルに登録されたアプリケーション プロトコル エンティティのユーザ ID に関する MCS-DETACH-USER-指示を受信検知した場合、アプリケーション プロトコル エンティティを削除するために名簿を更新し、またそのアプリケーション プロトコル エンティティに対応する GCCSAP へ GCC-PERMISSION-TO-ENROLL-指示を発行しなければならない。

8.3.5. トップ GCC プロバイダへの名簿更新の伝搬

コネクション階層構造の直接下位にあたるノードからの RosterUpdateIndicationPDU の受信に従って、GCC プロバイダは最初に会議及びアプリケーション名簿のサブセットを更新する。会議の各ノードは、そのノードとコネクション階層構造のすべての下位ノードにも対応する、会議及びアプリケーション名簿のサブセットを維持管理する。各アプリケーション プロトコル エンティティにとって、レコードに対応するノードのノード ID を含んだアプリケーション レコードのリストや、一部集約されたアプリケーション能力リストは、これらノードのいずれかに登録されたものである。これは、このノードのコネクション階層構造の直接下位のノードのアプリケーション能力リストに適用される規則セッ

トの結果に対応しているアプリケーション能力リストである。

GCC プロバイダがその名簿のサブセットを通常更新する手順は、受信した RosterUpdateIndication に使用される更新の方法に依存する。更新がフル リフレッシュで示される場合（PDU のフル リフレッシュ フラグで示される）、以下の手順に従う。まず、その PDU の送信元のノードに対応するものか、もしくは、コネクション階層構造においてそのノードより下位にあることがすでにわかっているノードであれば、すべてのノード レコードとアプリケーション レコードは、会議及びアプリケーション名簿から削除される。その後、PDU にリストされている、すべての会議およびアプリケーション名簿が、その名簿に追加される。このノードにストアされている要求ノードに対応するアプリケーション能力リストの全体が削除される。PDU に指定されたアプリケーション能力リストがそれに置き換えられる。全ての現在登録されているか、もしくは現在のノードの下位にあるアプリケーション プロトコル エンティティと、受信した PDU に示されている（この PDU で特定されないものを含む）いずれかの新しいアプリケーション プロトコル エンティティのために、新しい集約アプリケーション能力リストが直接下位のノードからの各アプリケーション能力リストから計算される。この集約は 8.3.8 節に記述されている手順によって行われる。

RosterUpdateIndication がフル リフレッシュを示していない場合、異なった手順に従う。まず、ノード レコード情報が変更されていることが示されていて、変更がノード レコードのリストのリフレッシュとして行われている場合、PDU の送信元ノードに対応するノードレコード全てか、もしくは、コネクション階層構造において下位のノードとして知られているものは、ローカルの会議名簿のサブセットから削除される。その後、PDU にリストされているすべてのノード レコードは、その名簿に追加される。変更がノードレコードの一連の更新として行われた場合は、追加指示レコードは名簿サブセットに追加され、置き換え指示は該当するレコードを置き換え、削除指示は削除が行われる。すでに存在するレコードの変更や、存在しないレコードの変更・削除は無視される。いずれの場合も、名簿サブセットにかつて存在しすでに削除されたノードのために、それらのノードに対応するすべてのアプリケーションレコードは削除される（PDU のアプリケーション名簿部に対応する更新レコードが含まれているか否かにかかわらず）。いずれかのアプリケーション レコード情報が変更を示されている場合は、各アプリケーション プロトコル セッションのために同様な手順に従う。各アプリケーション プロトコル セッションのために、変更がリフレッシュとして行われた場合は、PDU の送信元ノード、または、コネクション階層構造において下位のノードとして、あらかじめ知られているノードに対応するすべてのアプリケーション レコードは、ローカルのアプリケーション名簿のサブセットから削除される。その後、PDU にリストされている全てのアプリケーション レコードが名簿に追加される。与えられたアプリケーション プロトコルセッションに登録されたアプリケーション プロトコル エンティティを持つノードがないことを示すことができる。変更がアプリケーションレコードの一連の更新として行われた場合、追加指示レコードは名簿サブセットに追加され、置き換え指示は該当するレコードの置き換え、そして、削除指示は削除が行われる。すでに存在するレコードの変更や、存在しないレコードの変更・削除は無視される。アプリケーション能力リストのために、フル リフレッシュ用に記述されたものと、同様な手順に従う。しかしながらこの場合、変更指示された一連の同位アプリケーション プロトコル エンティティ用に新しいアプリケーション能力リストが再計算されるだけである。

上記のいずれの場合も、会議名簿とアプリケーション名簿中の各アプリケーション プロトコル セッションの更新情報は一つのインスタンス番号を含む。会議情報が受信した更新によって変更された場合、会議情報のインスタンス番号はモジュロ 2¹⁶ でインクリメントされるべきである。同様に、各アプリケーション プロトコル セッションで、いずれかの情報が変更されるもの（アプリケーション レコードやアプリケーション能力リスト）について、そのインスタンス

番号はモジュロ 2^{16} でインクリメントされるべきである。複数の変更が単一の更新として他ノードへ送られる場合、これらの変更はインスタンス番号の一つのインクリメントとしてカウントされる。インスタンス番号はローカルに管理されその名簿のローカルサブセットに対応する。すなわち、そのノードとコネクション階層構造における下位ノードである。トップ GCC プロバイダの場合、会議及びアプリケーション名簿にインスタンス番号が適用され、GCC-APPLICATION-ROSTER-REPORT プリミティブにより、全ノードへ放送され通知される。また、会議情報は、最後のインスタンスから追加・削除されたノードかどうかを示すフラグを含む。同様に、各アプリケーション プロトコル セッションのために、フラグはその一連のアプリケーション プロトコル エンティティに最後のインスタンスから追加・削除されたノードかどうかをフラグが示す。この場合、追加もしくは削除されるノードは、あるノードでアプリケーション プロトコル エンティティが登録もしくは登録を解除されることを示す。どちらの場合も、ノード全体が会議から追加か削除されることを必要とする訳ではない。

一旦、会議とアプリケーション名簿のどちらかあるいは両方の情報が再計算されたら、トップ GCC プロバイダでない GCC プロバイダは、RosterUpdateIndicationPDU を生成し、コネクション階層構造の上位の GCC プロバイダへ直接送るべきである。この PDU に含まれる情報のフォーマットは、フル リフレッシュ、いくつかの同位アプリケーション プロトコル エンティティと会議情報のどちらかか両方の更新、個々の更新のいずれかである。更新情報を送信するフォーマットの選択は、GCC プロバイダに任される。PDU サイズ及び転送時間を最小にするフォーマットを使用することが望ましい。

8.3.6. 会議及びアプリケーション名簿の分配

トップ GCC プロバイダが RosterUpdateIndication を受信した時、その名簿サブセットの更新がなされる中間 MCU 用に記述された同様の方法で管理される会議及びアプリケーション名簿情報を変更する（それは、この場合、サブセットというより、全会議及びアプリケーション名簿）。一旦、完全に名簿が更新されたら、トップ GCC プロバイダは RosterUpdateIndicationPDU を送付することにより、更新された名簿情報を全ノードへ放送する。これはチャンネル ID が GCC-BROADCAST-CHANNEL で、high データプライオリティで、データフィールドに PDU を含んだ、MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。

会議名簿が新しいノードを加えるために変更された場合、RosterUpdateIndication をフル リフレッシュとして放送する。すなわち、会議及びアプリケーション名簿の全てを全ノードに転送する。新しいノードの追加を含まない名簿への変更の場合、トップ GCC プロバイダは以下のいずれかの方法を選択する。更新の送信をフル リフレッシュにて行う方法。ノード レコードのリストのリフレッシュ、あるいは、同位アプリケーション プロトコル エンティティ セットの一部もしくは全部、あるいは、アプリケーション プロトコル セッションの一部もしくは全部のアプリケーション能力リスト、の内の一つあるいは組み合わせにて更新を行う方法。

GCC-BROADCAST-CHANNEL の RosterUpdateIndication の受信に従って、各 GCC プロバイダは GCC-CONFERENCE-ROSTER-REPORT-指示を生成し、受信した PDU が会議名簿の変更を示していれば、それをコントロール GCCSAP へ発行する。受信した PDU がいくつかもしくはすべてのアプリケーション名簿の変更を示していれば、GCC プロバイダは一連の GCC-APPLICATION-ROSTER-REPORT-指示を生成し、PDU で受信したアプリケーション名簿更新のためにアプリケーション プロトコル セッションに対応する登録済みアプリケーション プロトコル

エンティティに関連する GCCSAP へそれらを発行する。それはまた GCC-APPLICATION-ROSTER-REPORT-指示を他の GCCSAP に対し発行するかもしれないが、その必要性はこの標準の範囲外の問題である。そのアプリケーション プロトコル エンティティ用のセッション キーに関連する会議アプリケーション名簿の該当部分のみが対応するプリミティブに含まなければならない。GCC プロバイダは他のセッション キーに対応する名簿の各部分を含めることを選択するかもしれないが、その必要性はこの標準の範囲外の問題である。セッション ID なしで非活性に登録されたアプリケーション プロトコル エンティティの場合、GCC プロバイダは、アプリケーション プロトコル エンティティと同じ基本アプリケーション プロトコルとともに、いずれかのアプリケーション プロトコル セッションに対応する名簿の部分のために、対応する GCCSAP へ GCC-APPLICATION-ROSTER-REPORT-指示を発行する。GCC プロバイダはまた、コントロール GCCSAP へ、そこには受信した PDU により変更されたことを示された一連の同位アプリケーション プロトコル エンティティすべてのためのアプリケーション名簿の変更を含んだ、GCC-APPLICATION-ROSTER-REPORT-指示を発行する。

8.3.7. ノードの会議からの退場

MCS-DETACH-USER-指示の受信に従って、コネクション階層構造において下位のノードをもつ MCU ノードは、示されたユーザ ID を確認して、そしてコネクション階層構造において直接下位のノードのノード ID にそれが対応するか判断する。対応する場合、このノードに対応する全てのエントリとそのノードの下位に接続されているいずれかのノードを会議及びアプリケーション名簿から削除する。その後、8.3.8 節に記述されている方法ですべての同位アプリケーション プロトコル エンティティのためにアプリケーション能力リストを再計算する。一旦ノードの退場を反映するために名簿のサブセットが完全に更新されたら、その GCC プロバイダは会議の他のノードへこの更新を伝搬するために 8.3.5 節に記述されている手順に従うべきである。

会議から切断されたノードにおいて（切断、終了、除外のいずれかによる）、GCC プロバイダは対応する会議への登録許可を取り消す GCC-PERMISSION-TO-ENROLL-指示を生成し、これをすべての非コントロール GCCSAP へ発行する。

8.3.8. 集約アプリケーション能力リスト

各 GCC プロバイダは、各アプリケーション プロトコル エンティティのために、コネクション階層構造の下位のノードとローカルノードで登録された同位アプリケーション プロトコル エンティティに対応する集約アプリケーション能力リストを形成する責任がある。ノードはコネクション階層構造において直接下位であるなら、ローカル GCC プロバイダは、全ての下位ノードからのアプリケーション能力リストよりはむしろ、それらのノードからの集約アプリケーション能力リストを知ることのみが必要である。ローカルノードにおいて同位アプリケーション プロトコル エンティティからのアプリケーション能力リストと同様に、これらの集約アプリケーション能力リストがローカルアプリケーション名簿から入力として与えられ、GCC プロバイダは以下の手順により集約アプリケーション能力リストを作成する。

- ・ 各リスト中のそれぞれの能力項目に対して、能力クラスを判断する。
- ・ いずれかのクラスに対しても、集約アプリケーション能力リスト中のエントリにたいするカウントパラメータ値は、入力セットの各アプリケーション能力リストの対応するエントリに示されているカウントの合計をセットする。

- ・ Unsigned-MIN や Unsigned-MAX クラスに対し、新たなエントリは入力セット中の各アプリケーション能力リストのエントリに対応するそれぞれの最小か最大の値を含む。

8.3.9. アプリケーション及び会議名簿問い合わせ

GCC-APPLICATION-ROSTER-INQUIRE-要求プリミティブの受信に従って、このプリミティブをサポートする GCC プロバイダは GCC-APPLICATION-ROSTER-INQUIRE-確認プリミティブを生成し要求者の GCCSAP へ発行することで応答する。確認プリミティブの内容はローカルに管理された会議アプリケーション名簿データベースから生成され、要求の中のセッション キー (Session Key) が要求で与えられたキーの長さ上に正確にエントリのセッション キー (Session Key) に合うエントリのみを名簿に含めるべきである。

GCC-CONFERENCE-ROSTER-INQUIRE-要求 プリミティブの受信に従って、このプリミティブをサポートする GCC プロバイダはGCC-CONFERENCE-ROSTER-INQUIRE-確認プリミティブを生成し要求者のGCCSAPへ発行することで応答する。確認プリミティブの内容は会議に存在することが知られているすべてノードをアカウントしているローカル会議名簿データベースから生成される。

8.3.10. 名簿更新の例

図 8-5/JT-T124 に会議中に発行される会議アプリケーション名簿の更新の原因になる GCC-APPLICATION-ENROLL-要求の例を示す。この例において、ノードはコネクション階層構造の第 3 層に位置する。それは更新指示を次の上位のノードに発行し、そのノードはその後更新指示をフォーマットし、そのまた上位のノードに送信する。この例で、それはコネクション階層構造上のトップノードに該当する。トップノードは完全なアプリケーション名簿を組み立て、会議のすべてのノードに更新されたアプリケーション プロトコル セッションに関連する部分を放送する。それは GCC-APPLICATION-REPORT プリミティブをすべてのノードのノードコントローラや存在する場合は同位アプリケーション プロトコル エンティティへ発行することとなる。

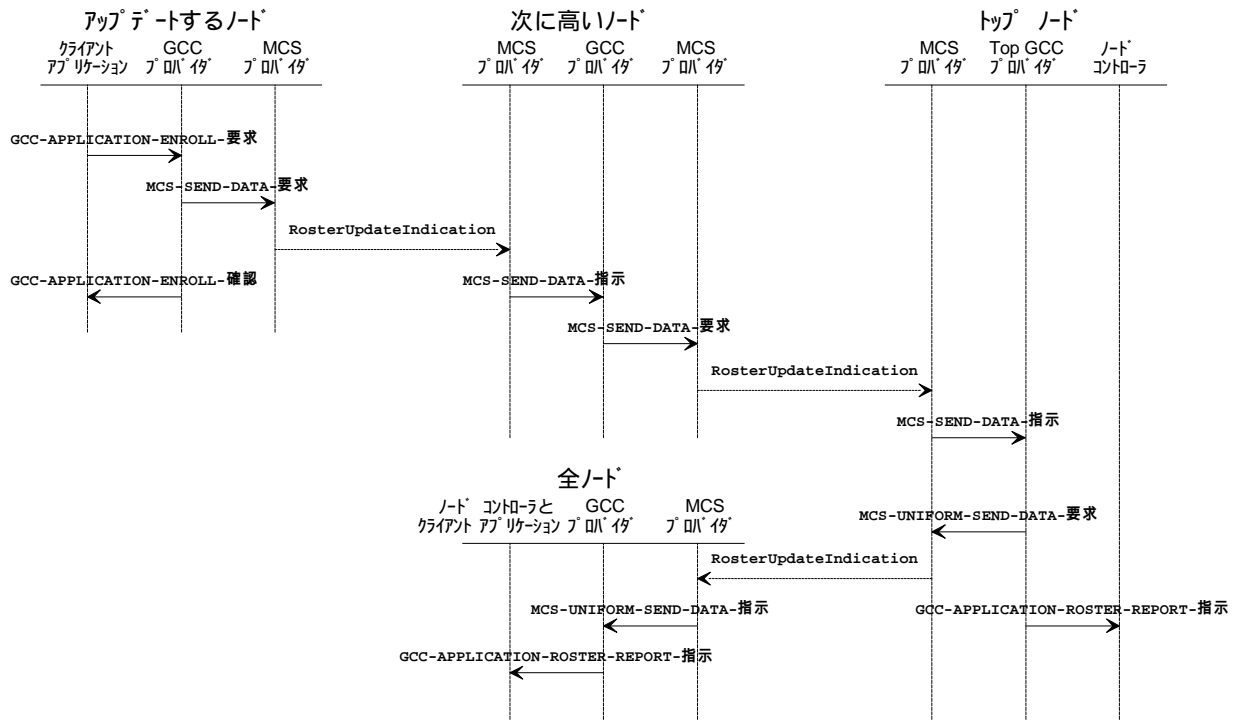


図 8-5/JT-T124 アプリケーション名簿更新例
(ITU-T T.124)

8.3.11. アプリケーション プロトコル エンティティの遠隔起動

コントロール GCCSAP が通常の GCCSAP いずれか経由の GCC-APPLICATION-INVOKE-要求プリミティブの受信に従って、このプリミティブをサポートする GCC プロバイダは特定された会議における全てのノードに ApplicationInvokeIndicationPDU を放送する。これは、チャンネル ID が GCC-BROADCAST-CHANNEL で、high データプライオリティで、データフィールドに PDU を含む、MCS-SEND-DATA-要求か、MCS-UNIFORM-SEND-DATA-要求を発行することによって行われる。別の方法として、同じ high データプライオリティで、データフィールドに PDU を含む、チャンネル ID としてリストされた宛先ノードのひとつのノード ID を指定した、一つまたはそれ以上の MCS-SEND-DATA-要求を発行することにより行われる場合もある。この場合、PDU 中の宛先ノード リスト (Destination Node List) パラメータを Null にしておく。ApplicationInvokeIndication の内容を表 8-36/JT-T124 に示す。GCC プロバイダはその後、要求が正しく実行されたかどうかを示す、GCC-APPLICATION-INVOKE-確認プリミティブを生成し要求者の GCCSAP へ発行する。

表8-36/JT-T124 ApplicationInvokeIndication GCCPDU
(ITU-T T.124)

内容	ソース	シンク
アプリケーション プロトコル エンティティ リスト (Application Protocol Entity List)	要求	指示
宛先ノード リストまたはNull (Destination Node List or NULL)	要求	宛先GCCプロバイダ

ApplicationInvokeIndicationPDU の受信に従って、GCC-APPLICATION-INVOKE プリミティブをサポートする GCC プロバイダは、まず宛先ノードのリスト上のローカルノードかどうか判断する。リストが Null か、ローカルノード

ド ID がリストに存在する場合、GCC プロバイダは GCC-APPLICATION-INVOKE-指示 プリミティブを生成しコントロール GCCSAP へ発行するかも知れない。起動ノードパラメータは、受信した MCS-SEND-DATA-指示か MCS-UNIFORM-SEND-DATA-指示の送信者ユーザ ID フィールドで満たされる。ローカルノードが宛先ノードリスト (Destination Node List) にない場合 (明示的にリストされた場合)、それ以上の処理は行われない。

8.4. アプリケーション レジストリ

アプリケーション レジストリは会議のためにトップ GCC プロバイダに常駐するデータベースである。あらゆる MCU ノードはこのデータベースを操作するためにレジストリ サービスの全機能をサポートしなければならない。それに対し、ターミナル ノードはそのターミナルでサポートするアプリケーション プロトコルが要求するレジストリ サービスの機能だけを選択してサポートすればよい。

会議が生成されたとき、トップ GCC プロバイダにあるレジストリ データベースは、全てのレジストリ エントリが空で、監視されているレジストリ エントリが無い状態に初期化される。これはトップ GCC プロバイダが NODE-ID-CHANNEL に加入する前に行われる。

会議中でのレジストリの操作中、空でないレジストリ エントリは以下の情報を含む：

- ・ エントリを識別するレジストリキー (Registry Key)。
- ・ 含まれている情報のタイプ - 「チャンネル ID(Channel ID)」、「トークン ID(Token ID)」、「パラメータ(Parameter)」のいずれか。
- ・ 監視状態 - 「監視が有効(monitoring enabled)」、「監視が無効(monitoring disabled)」のいずれか。
- ・ エントリの値 - 「チャンネル ID(Channel ID)」、「トークン ID(Token ID)」、「パラメータ(Parameter)」の実際の値。
- ・ エントリの所有者 - 所有者なしか、もしくは所有するアプリケーション プロトコル エンティティのノード ID およびエンティティ ID のいずれか。
- ・ エントリの変更権 (パラメータ型のエントリのみ) - 「所有者(Owner)」、「セッション(Session)」、「パブリック(Public)」のいずれか。

一つのノードから複数のレジストリ要求を受信した場合、トップ GCC プロバイダでの処理は要求を受信した順に正確に行われる。要求ノードへの応答 PDU の順番も、ノードから要求を受信した順序を維持される。

要求ノードにおいても、受信した PDU とそれに対応する確認プリミティブの関係と同様に、要求プリミティブとそれに対応する送信 PDU の順番を維持される。

8.4.1. チャンネルの登録

GCC-REGISTRY-REGISTER-CHANNEL-要求 プリミティブ受信時に、GCC プロバイダはトップ GCC プロバイ

ダに RegistryRegisterChannelRequest PDU を送信する。PDU の送信は、チャンネル ID(Channel ID)がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。

RegistryRegisterChannelRequest PDU の内容は、表 8-37/JT-T124 で示される。

表 8-37/JT-T124 RegistryRegisterChannelRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
エンティティ ID(Entity ID)	発行元 GCC プロバイダ	トップ GCC プロバイダ
キー(Key)	要求(Request)	トップ GCC プロバイダ
チャンネル ID(Channel ID)	要求	トップ GCC プロバイダ

RegistryRegisterChannelRequest PDU の受信時に、トップ GCC プロバイダは受信した MCS-SEND-DATA-指示の送信ユーザ ID(Sender User ID)と PDU のエンティティ ID(Entity ID)が、現在のアプリケーション名簿中のアプリケーション プロトコル エンティティのノード ID およびエンティティ ID と正確に一致するかどうかを最初に比較する。同一であれば、該当するレジストリ エントリを作成して、そのチャンネルを登録しようとする。まず、レジストリ データベースに指定されたキー(Key)に一致するエントリが既に存在するかどうかを調べる。もしレジストリ エントリが存在しなければ、トップ GCC プロバイダはこのキー(Key)のエントリを生成し、PDU に含まれているエンティティ ID(Entity ID)と受信した MCS-SEND-DATA-指示の送信ユーザ ID(Sender User ID)で示されるノード ID とともにエントリの所有者に設定する。そしてエントリのタイプを「チャンネル ID(Channel ID)」とし、PDU で指定されたチャンネル ID(Channel ID)をその値として、このエントリをデータベースに登録する。

トップ GCC プロバイダは、RegistryResponse PDU を要求ノードへ送り、そのチャンネルが正しく登録されたことを通知する。PDU の送信は、チャンネル ID(Channel ID)が要求者のノード ID で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。RegistryResponse PDU の内容は、表 8-38/JT-T124 で示される。要求が成功した場合、結果(Result)パラメータは「成功(Successful)」となる。また、変更権(Modification Rights)フィールドは記入されない(このフィールドはパラメータ タイプのエントリのためにだけある)。

レジストリ エントリが既に存在するか、要求者がアプリケーション名簿に無いため無効か、あるいは利用できる資源の制限によりレジストリ エントリが生成できない場合、レジストリは変更されず、RegistryResponse PDU はその失敗の理由を示す否定の結果(Result)で返信される。この場合、そのエントリを修正しようとする前の値がレジストリ項目として RegistryResponse PDU で返される。

表 8-38/JT-T124 RegistryResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
エンティティ ID(Entity ID)	トップ GCC プロバイダ	宛先 GCC プロバイダ

プリミティブの型(チャンネルの登録、トークンの割当て、パラメータのセット、エントリの検索、エントリの削除、エントリの監視) (Primitive type (register-channel, assign-token, set-parameter, retrieve-entry, delete-entry, monitor-entry))	トップ GCC プロバイダ	宛先 GCC プロバイダ
キー(Key)	トップ GCC プロバイダ	確認
レジストリ項目(Registry Item)	トップ GCC プロバイダ	確認
所有者(Owner)	トップ GCC プロバイダ	確認
変更権 (Modification Rights)(オプション))	トップ GCC プロバイダ	確認
結果(Result)	トップ GCC プロバイダ	確認

既存のレジストリ エントリの変更に成功した場合、トップ GCC プロバイダはそのレジストリ エントリが監視されていたかどうかをチェックする。監視されていなかった場合、トップ GCC プロバイダは何もしない。エントリが監視されていた場合、トップ GCC プロバイダは、会議中の全ノードに対して RegistryMonitorEntryIndication PDU を送信する。PDU の送信は、チャンネル ID(Channel ID)が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。RegistryMonitorEntryIndication PDU の内容は、表 8-44/JT-T124 で示される。

この型の RegistryResponse PDU 受信時に、GCC プロバイダは要求が成功したかどうかを示す RegistryResponse PDU の結果(Result)パラメータに示された内容で GCC-REGISTRY-REGISTER-CHANNEL-確認プリミティブを生成し、エンティティ ID(Entity ID)によって示されるアプリケーション プロトコル エンティティの GCCSAP に発行する。GCC プロバイダが対応するエンティティ ID のアプリケーションプロトコル エンティティが現在登録されていないことを認識していれば、指示を無視し、特に何もしない。

8.4.2. トークンの割当て

GCC-REGISTRY-ASSIGN-TOKEN-要求 プリミティブ受信時に、GCC プロバイダはトップ GCC プロバイダに RegistryAssignTokenRequest PDU を送信する。PDU の送信は、チャンネル ID(Channel ID)がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。

RegistryAssignTokenRequest PDU の内容は、表 8-39/JT-T124 で示される。

表 8-39/JT-T124 RegistryAssignTokenRequest GCCPDU
(ITU-T T.124)

内容	ソース	シンク
エンティティ ID(Entity ID)	発行元 GCC プロバイダ	トップ GCC プロバイダ
キー(Key)	要求	トップ GCC プロバイダ

RegistryAssignTokenRequest PDU の受信時に、トップ GCC プロバイダは受信した MCS-SEND-DATA-指示の送信ユーザ ID(Sender User ID)と PDU のエンティティ ID(Entity ID)が、現在のアプリケーション名簿中のアプリケーション プロトコル エンティティのノード ID およびエンティティ ID と正確に一致するかどうかを最初に比較する。同一であれば、トップ GCC プロバイダは適切なレジストリ エントリを生成し、トークン ID を生成してトークンの割当てを試みる。

まず、レジストリ データベースに指定されたキー(Key)に一致するエントリが既に存在するかどうかを調べる。もしレジストリ エントリが存在しなければ、トップ GCC プロバイダは動的トークン ID の領域(16384 から 65535 まで)から、最初に新しいトークン ID を割り当てる。その後、トップ GCC プロバイダはこのキーのためのエントリを生成し、受信した MCS-SEND-DATA-指示の送信ユーザ ID(Sender User ID)で指定されたノード ID と PDU に含まれるエンティティ ID(Entity ID)をエントリの所有者、エントリ種別を「トークン ID(Token ID)」と、および生成されたトークン ID をエントリに対する値とする。

トップ GCC プロバイダは、要求ノードに RegistryResponse PDU を送信し、トークンが正しく割当てられたこととトークン ID の値を要求者に通知する。PDU の送信は、チャンネル ID(Channel ID)が要求者のノード ID で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。RegistryResponse PDU の内容は、表 8-38/JT-T124 で示される。要求が成功した場合、結果(Result)パラメータは「成功(Successful)」となる。また変更権(Modification Rights)フィールドは記入されない(このフィールドはパラメータ型のエントリのためにだけある)。

もしレジストリエントリが既に存在するか、要求者がアプリケーション名簿に存在しないか、利用できる資源の制限によりレジストリ エントリが生成できないか、利用できる動的トークン ID が無かった場合は、レジストリは修正されず、失敗の理由を示す否定の RegistryResponse PDU が通知される。この場合、修正しようとする前のエントリの値がレジストリ項目として RegistryResponse PDU の中で通知される。

既存のレジストリ エントリの修正に成功した場合、トップ GCC プロバイダはそのレジストリ エントリが監視されていたかどうかをチェックする。監視されていない場合は、トップ GCC プロバイダは何もしない。そのエントリが監視されていれば、トップ GCC プロバイダは RegistryMonitorEntryIndication PDU を会議中のすべてのノードへ送信する。PDU の送信は、チャンネル ID(Channel)が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。RegistryMonitorEntryIndication PDU の内容は表 8-44/JT-T124 で示される。

このタイプの RegistryResponse PDU の受信時に、GCC プロバイダは要求が成功したかどうかを示す RegistryResponse PDU の結果(Result)パラメータに示された内容で GCC-REGISTRY-ASSIGN-TOKEN-確認 プリミティブを生成する。もし成功していれば、生成されたトークン ID を含む。そしてエンティティ ID(Entity ID)によって

示されるアプリケーション プロトコル エンティティの GCCSAP に発行される。GCC プロバイダが対応するエンティティ ID のアプリケーションプロトコル エンティティが現在登録されていないことを認識していれば、指示を無視し、特に何もしない。

8.4.3. パラメータのセット

GCC-REGISTRY-SET-PARAMETER-要求 プリミティブ受信時に、GCC プロバイダはトップ GCC プロバイダに RegistrySetParameterRequest PDU を送信する。PDU の送信は、チャンネル ID(Channel ID)がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。

RegistrySetParameterRequest PDU の内容は表 8-40/JT-T124 で示される。

表 8-40/JT-T124 RegistrySetParameterRequest GCCPDU
(ITU-T T.124)

内容	ソース	シンク
エンティティ ID(Entity ID)	発行元 GCC プロバイダ	トップ GCC プロバイダ
キー(Key)	要求	トップ GCC プロバイダ
パラメータ(Parameter)	要求	トップ GCC プロバイダ
変更権 (Modification Rights) (オプション)	要求	トップ GCC プロバイダ

RegistrySetParameterRequest PDU の受信時に、トップ GCC プロバイダは受信した MCS-SEND-DATA-指示中の送信ユーザ ID(Sender User ID)と PDU に含まれるエンティティ ID(Entity ID)が、現在の会議名簿中のアプリケーション プロトコル エンティティのノード ID およびエンティティ ID と正確に一致するかどうかを最初に比較する。同一であれば、パラメータを既存のレジストリエントリにセットしようとするか、指定されたパラメータで新しいエントリを作ろうとする。まず、指定されたキー(Key)に対応しているレジストリ エントリがレジストリ データベース中に既に存在するかどうかを調べる。もしエントリが存在すれば、エントリのタイプと、所有者と、現在の変更権を決定する。エントリが存在しなければトップ GCC プロバイダはこのキーのためのエントリを生成する。受信した MCS-SEND-DATA-指示の送信ユーザ ID で指定されたノード ID と PDU に含まれるエンティティ ID をエントリの所有者、エントリ種別をパラメータ、および PDU に指定されたパラメータ値をエントリに対する値として、このエントリをデータベースに加える。レジストリエントリが既に存在し、そのエントリがパラメータ タイプならば、GCC プロバイダはまずその要求者がこのエントリを修正する権利を持つかどうかをチェックする。要求者が変更権を持つならば、レジストリ エントリの値は、型をパラメータ、および PDU の中で指定されるパラメータ値をエントリに対する値として反映するように修正される。セットに先立ち、エントリが所有されていないかったら、エントリは新しい所有者を示すように修正される。

以下の規則により、要求者が変更権を有するかどうかを決定する。もしエントリの変更権属性が「所有者(Owner)」にセットされていた場合、エンティティ ID(Entity ID)と受信した MCS-SEND-DATA-指示の送信ユーザ ID(Sender User ID)中のノード ID で示される要求者は、要求を成功させるためには、そのエントリの所有者が、あるいはエント

リが現在非所有でなければならない。現在のエントリの変更権属性が「セッション(Session)」にセットされていた場合、要求者がこの要求を成功させるためには、同じアプリケーション セッションの一部でなければならない。これは要求者のためのアプリケーション名簿のエントリを見つけ、セッションキーを調べることによって決定される。セッションキーは、その所有者のセッションに同一でなければならない。また、エントリが所有されなければ、この制限は適用されない。エントリの現在の変更権属性が「パブリック(Public)」にセットされていれば、要求を成功させるのに要求者に制限はない。

パラメータエントリが最初に作られるときに、そのエントリの変更権の状態が決定される。変更権パラメータがそのエントリを作成した要求 PDU に含まれていたなら、変更の種別は指示された値にセットされる。そうでなければ変更権は「パブリック(Public)」と仮定される。パラメータをセットする要求で、変更権エントリがないならば変更権の状態は変更されない。もしあれば、要求者がエントリの所有者か、あるいはそのエントリが非所有の場合だけ、GCC プロバイダは PDU で指定された状態に変更権の状態を修正する。さもなければ、変更権の状態は変更されない。後者の場合、パラメータ設定操作は正常なものとして行われ、その結果には影響されない。

上記のいずれの場合でも、トップ GCC プロバイダは要求ノードに RegistryResponse PDU を送信し、パラメータが正しくセットされたことを通知する。PDU の送信は、チャンネル ID(Channel ID)が要求者の NODE-ID-CHANNEL で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。RegistryResponse PDU の内容は表 8-38/JT-T124 で示される。要求が成功した場合、結果(Result)パラメータは、「成功(Successful)」となる。変更権パラメータは、応答 PDU に含まれる。

利用可能な資源の制限によりレジストリ エントリが生成できない、既にエントリは存在するが要求者が変更権を持っていない、要求者がアプリケーション名簿に登録されていない、あるいはエントリがパラメータ型でない場合、そのレジストリは修正されず、RegistryResponse PDU はその失敗の理由を示す否定の結果で返される。この場合、そのエントリ修正前の値が「レジストリ項目(Registry Item)」として RegistryResponse PDU で返される。な変更権が不適切な場合、結果(Result)パラメータは「他に属している(belongs to other)」となる。

既存のレジストリ エントリの変更に成功した場合、トップ GCC プロバイダはそのレジストリ エントリが監視されていたかどうかをチェックする。監視されていなかった場合、トップ GCC プロバイダは何もしない。エントリが監視されていた場合、トップ GCC プロバイダは会議中の全ノードに対して RegistryMonitorEntryIndication PDU を送信する。PDU の送信は、チャンネル ID(Channe ID)が GCC-BROADCAST-CHANNEL のチャンネル ID で、データプライオリティが High で、データ(Data)フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。

RegistryMonitorEntryIndication PDU の内容は表 8-44/JT-T124 で示される。

このタイプの RegistryResponse PDU の受信時に、GCC プロバイダは要求が成功したかどうかを示す RegistryResponse PDU の結果(Result)パラメータに示された内容で GCC-REGISTRY-SET-PARAMETER-確認 プリミティブを生成し、エンティティ ID(Entity ID)によって示されるアプリケーション プロトコル エンティティの GCCSAP を発行する。GCC プロバイダが対応するエンティティ ID のアプリケーションプロトコル エンティティが現在登録されていないことを認識していれば、指示を無視し、特に何もしない。

8.4.4. エントリの検索

GCC-REGISTRY-RETRIEVE-ENTRY-要求 プリミティブ受信時に、GCC プロバイダはトップ GCC プロバイダに RegistryRetrieveEntryRequest PDU を送信する。PDU の送信は、チャンネル ID がトップ GCC プロバイダのノード ID チャンネルで、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。

RegistryRetrieveEntryRequest PDU の内容は表 8-41/JT-T124 で示される。

表 8-41/JT-T124 RegistryRetrieveEntryRequest GCCPDU
(ITU-T T.124)

内容	ソース	シンク
エンティティ ID(Entity ID)	発信元 GCC プロバイダ (Source GCC Provider)	トップ GCC プロバイダ (Top GCC Provider)
キー(Key)	要求(Request)	トップ GCC プロバイダ (Top GCC Provider)

RegistryRetrieveEntryRequest PDU の受信時に、トップ GCC プロバイダはキーによって指定されたレジストリエントリの内容を調べる。レジストリエントリは、以下の 4 つの可能な状態のいずれかである：「空(empty)」、「チャンネル ID(containing a Channel ID)」、「トークン ID(Token ID)」、「パラメータ(Parameter)」。これらのいずれの場合でも、トップ GCC プロバイダはチャンネル ID がその要求者のノード ID で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を送信することによって、RegistryResponse PDU を要求ノードへ送り、要求者にそのエントリの状態を通知する。RegistryResponse PDU の内容は表 8-38/JT-T124 で示される。レジストリ項目パラメータはそのエントリの状態を含み、そして空でなければそのエントリが含む値も含む。「結果(Result)」パラメータは、もしエントリが空でなければ「成功(successful)」、エントリが空であれば「エントリが見つからない (entry-not-found)」となる。もしエントリがパラメータ タイプならば、変更権パラメータは応答 PDU に格納される。

この型の RegistryResponse PDU の受信時に、GCC プロバイダは要求が成功したかどうかを示す RegistryResponse PDU の「結果(Result)」パラメータに示された内容で GCC-REGISTRY-RETRIEVE-ENTRY-確認プリミティブを生成し、エンティティ ID によって示されるアプリケーション プロトコル エンティティの GCCSAP に発行する。GCC プロバイダが対応するエンティティ ID のアプリケーションプロトコル エンティティが現在登録されていないことを知っていれば、指示を無視し、特に何もしない。

8.4.5. エントリの削除

GCC-REGISTRY-DELETE-ENTRY-要求 プリミティブ受信時に、GCC プロバイダはトップ GCC プロバイダに RegistryDeleteEntryRequest PDU を送信する。PDU の送信は、チャンネル ID がトップ GCC プロバイダのノード ID

チャンネルで、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。

RegistryDeleteEntryRequest PDU の内容は表 8-42/JT-T124 で示される。

表 8-42/JT-T124 RegistryDeleteEntryRequest GCCPDU
(ITU-T T.124)

内容	ソース	シンク
エンティティ ID(Entity ID)	発信元 GCC プロバイダ (Source GCC Provider)	トップ GCC プロバイダ (Top GCC Provider)
キー(Key)	要求(Request)	トップ GCC プロバイダ (Top GCC Provider)

RegistryDeleteEntryRequest PDU の受信時に、トップ GCC プロバイダは指定されたレジストリ エントリを削除しようとする。まず指定された「キー(Key)」に対応するレジストリ エントリがレジストリデータベース中に存在するか調べ、存在すれば所有者を調べる。もしレジストリエントリが存在し、かつ要求者が所有している（要求者と所有者のノード ID とエンティティ ID が同一である）か、あるいはそのエントリが現在所有されていないか、トップ GCC プロバイダは、そのエントリの内容を削除し、状態を空にセットし、監視を無効にする。

レジストリエントリの削除が成功、あるいはそのエントリが既に存在しなければ、トップ GCC プロバイダはチャンネル ID が要求者のノード ID で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を送信することにより、RegistryResponse PDU を要求ノードへ送り、そのエントリが正しく削除されたことを通知する。RegistryResponse PDU の内容は表 8-38/JT-T124 で示される。成功した場合、「結果(Result)」パラメータは「成功(Successful)」に指定される。またレジストリ項目は空に設定される。

レジストリエントリは既に存在するが、別のノードによって所有されていたら、そのレジストリは修正されない。しかし、RegistryResponse PDU がその失敗の理由を示す否定の「結果(Result)」で返される。この場合、レジストリ項目は削除を試みる前のそのエントリの値である。

既存のレジストリエントリの削除に成功した場合、トップ GCC プロバイダは（削除に先立ち）監視状態にセットされていたかどうかをチェックしておく。エントリが監視されていない場合はトップ GCC プロバイダは特に何もしない。エントリが監視されていれば、トップ GCC プロバイダはその会議中の全ノードへ RegistryMonitorEntryIndication PDU を送信する。PDU の送信は、チャンネル ID が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を送信することにより行われる。

RegistryMonitorEntryIndication PDU の内容は表 8-44/JT-T124 で示される。

トークン型のレジストリ エントリの削除に成功した場合、トップ GCC プロバイダは対応するトークンを削除してもよい。すなわち、トップ GCC プロバイダは後で RegistryAssignTokenRequest に応じて削除したエントリに含めたトークン ID を再利用してもよい。

この型の RegistryResponse PDU の受信時に、GCC プロバイダは要求が成功したかどうかを示す RegistryResponse PDU の「結果(Result)」パラメータに示された内容で GCC-REGISTRY-DELETE-ENTRY-確認 プリミティブを生成する。そしてエンティティ ID によって示されるアプリケーション プロトコル エンティティの GCCSAP に発行される。GCC プロバイダが対応するエンティティ ID のアプリケーションプロトコル エンティティが現在登録されていないことを知っていれば、指示を無視し、特に何もしない。

8.4.6. エントリの監視

「有効/無効フラグ(Enable/Disable flag)」を「有効(Enable)」にセットした GCC-REGISTRY-MONITOR-要求 プリミティブの受信時に、GCC プロバイダはトップ GCC プロバイダに RegistryMonitorEntryRequest PDU を送信する。PDU の送信は、チャンネル ID がトップ GCC プロバイダの NODE-ID-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。RegistryMonitorEntryRequest PDU の内容は表 8-43 で示される。

GCC プロバイダは、このエントリに対応する RegistryMonitorEntryIndication PDU を受信した時に GCC-REGISTRY-MONITOR-指示を生成するかどうかを判断するために、要求アプリケーション プロトコル エンティティがこのエントリの監視を有効にしたことをローカルに記録しておく。

表 8-43/JT-T124 RegistryMonitorEntryRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
エンティティ ID(Entity ID)	発信元 GCC プロバイダ	トップ GCC プロバイダ
キー(Key)	要求	トップ GCC プロバイダ

RegistryMonitorEntryRequest PDU の受信時に、トップ GCC プロバイダはレジストリエントリの監視状態を変えようと試みる。まず、指定されたキーに対応するレジストリエントリがレジストリデータベースの中に既に存在するかどうかを調べる。レジストリエントリが存在すれば、指定されたエントリの監視状態を有効にセットする。

注：監視が有効にされると、会議中にエントリの変更が行われるたび、またはエントリが削除されるまで、監視指示は放送(broadcast)され続ける。

適当な処理を行った後、トップ GCC プロバイダは、トップ GCC プロバイダに RegistryResponse PDU を送る。PDU の送信は、チャンネル ID が要求者の NODE-ID-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-SEND-DATA-要求を送信することにより行われる。RegistryResponse PDU の内容は表 8-38/JT-T124 で示される。成功した場合、結果(Result)パラメータは「成功(Successful)」となる。要求されたエントリが存在しなければ、結果(Result)パラメータはエントリが見つからないとして指定される。もしエントリがパラメータタイプに指示されていれ、変更権パラメータが応答 PDU に含まれる。

レジストリエントリが存在しなければ、そのレジストリは修正されない。そして RegistryResponse PDU は失敗の理由を示す否定の結果(Result)で返却される。

このタイプの RegistryResponse PDU の受信時に、GCC プロバイダは要求が成功したかどうかを示す RegistryResponse PDUの結果(Result)パラメータに示された内容でGCC-REGISTRY-MONITOR-確認 プリミティブを生成する。そしてエンティティ ID によって示されるアプリケーション プロトコル エンティティの GCCSAP に発行される。GCC プロバイダが対応するエンティティ ID のアプリケーションプロトコル エンティティが現在登録されていないことを知っていれば、指示を無視し、特に何もしない。

レジストリエントリが監視されている間、レジストリエントリの内容が変更（パラメータの設定、エントリの削除、所有権の変更、変更権状態の変更）されると、トップ GCC プロバイダは会議中の全ノードに RegistryMonitorEntryIndication PDU を送信する。PDU の送信は、チャンネル ID が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データフィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行う。 RegistryMonitorEntryIndication PDU の内容は表 8-44/JT-T124 で示される。

表 8-44/JT-T124 RegistryMonitorEntryIndication GCCPDU
(ITU-T T.124)

内容	ソース	シンク
キー(Key)	トップ GCC プロバイダ	指示
レジストリ項目(Registry Item)	トップ GCC プロバイダ	指示
所有者(Owner)	トップ GCC プロバイダ	確認
変更権 (Modification Rights) (オプション)	トップ GCC プロバイダ	確認

RegistryMonitorEntryIndication PDU の受信時、レジストリの監視機能をサポートしている GCC プロバイダは GCC-REGISTRY-MONITOR-指示プリミティブを生成する。そして以前と同じ「キー(Key)」を指定して（このフラグでより最近受信されたものに Disable にセットさせることなく）「有効/無効フラグ(Enable/Disable flag)」を「有効(Enable)」にセットした GCC-REGISTRY-MONITOR-要求を出した全ての GCCSAP にこのプリミティブを送信する。もし GCC プロバイダが対応するエンティティ ID でその時点で登録されたアプリケーション プロトコル エンティティを知らなければ、指示を無視し特に何もしない。指定された「キー(Key)」のための要求者がこのノードにいないならば、指示 PDU は GCC プロバイダによって無視される。GCC-REGISTRY-MONITOR-要求を受信したとき、いつでも更新される、要求した GCCSAP のローカルなデータベースを保つことは、各々ローカルな GCC プロバイダの役割である。

注：ローカルな問題として、どのアプリケーション プロトコル エンティティがどのエントリの監視を有効あるいはは無効にしたかを記録せず、その代わりに RegistryMonitorEntryIndication PDU を受信したら全ての登録アプリケーション プロトコルエンティティに対応する GCCSAP に GCC-REGISTRY-MONITOR-指示を発行する様に実装するかもしれない。

8.4.7. ユニークなハンドルの割り当て

GCC-REGISTRY-ALLOCATE-HANDLE-要求プリミティブ受信時に、GCC プロバイダはトップ GCC プロバイダに RegistryAllocateHandleRequest PDU を送信する。PDU の送信は、チャンネル ID がトップ GCC プロバイダのノード ID チャンネルで、データプライオリティが High で、データ フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。RegistryAllocateHandleRequest PDU の内容を表 8-45 に示す。

表 8-45/JT-T124 RegistryAllocateHandleRequest GCCPDU

(ITU-T T.124)

内容	ソース	シンク
エンティティ ID (Entity ID)	発信元 GCC プロバイダ	トップ GCC プロバイダ
ハンドル数 (Number of Handles)	要求	トップ GCC プロバイダ

RegistryAllocateHandleRequest PDU の受信時に、トップ GCC プロバイダは要求された数のハンドルのユニークなセットを生成する。各々の割り当ては、32 ビットの状態変数をモジュロ 2^{32} でインクリメントすることによってなされる。次に要求されたハンドルの割り当てには、インクリメントされた値を使用する。可能なハンドルのフルセットが既に割り当てられているなら、要求は「ハンドル利用不可 (no handles are available) 」が指示されて拒否される。ハンドル数が十分に利用可能なら、割り当てたハンドルの数と、最初に割り当てたハンドルを規定することによってハンドルが要求者に返送される。トップ GCC プロバイダは RegistryAllocateHandleResponse PDU を要求者に送信して結果を返送する。これは、チャンネル ID が要求者のノード ID で、データプライオリティが High で、データ フィールドにその PDU を含んだ MCS-SEND-DATA-要求を発行することにより行われる。RegistryAllocateHandleResponse PDU の内容を表 8-46/JT-T124 に示す。結果が「成功 (successful) 」なら、パラメータにハンドルのリストが含まれる。

表 8-46/JT-T124 RegistryAllocateHandleResponse GCCPDU

(ITU-T T.124)

内容	ソース	シンク
エンティティ ID (Entity ID)	トップ GCC プロバイダ	移動先 GCC プロバイダ
ハンドル数 (Number of Handles)	トップ GCC プロバイダ	確認
最初のハンドル (First Handle)	トップ GCC プロバイダ	確認
結果 (Result)	トップ GCC プロバイダ	確認

RegistryAllocateHandleResponse 指示の受信時に、GCC プロバイダは GCC-REGISTRY-ALLOCATE-HANDLE 確認プリミティブを生成し、エンティティ ID によって指示されたアプリケーション プロトコル エンティティの GCCSAP にそれを発行する。GCC プロバイダが、対応するエンティティ ID で現在登録されているアプリケーション プロトコル エンティティがないことを知っているなら、指示は無視され、それ以上の動作は行わない。

8.4.8. 所有権の変更とレジストリの整理

アプリケーション名簿が更新された時はいつも、トップ GCC プロバイダはアプリケーション プロトコル エンティティが名簿から削除されたかどうか判断する。アプリケーション プロトコル エンティティが名簿から削除された場合、切断するアプリケーション プロトコル エンティティがいずれかのレジストリ エントリの所有者であるかを判断するために、レジストリ データ ベース全体を調査する。切断するアプリケーション プロトコル エンティティがいずれかのレジストリ エントリの所有者である場合、これらのエントリは現在の所有者がいないことを示すように変更される。アプリケーション名簿の変更の結果が、あるセッション中のすべてのアプリケーション プロトコル エンティティが未登録になったことを示すなら、トップ GCC プロバイダはレジストリ キーのセッション キー部分がそのセッションのセッション キーに整合するかどうか判断するために、すべてのレジストリ エントリを調査する。削除されたセッションに対応するエントリについては、トップ GCC プロバイダはそのレジストリ エントリを削除する。そのレジストリ エントリがトークン型エントリに対応していたなら、トップ GCC プロバイダはその後の再利用を許すためにトークンの割り当てを解放してもよい。

8.5. 会議議長権

会議議長権は会議議長トークンの使用によって制御されている。このトークンが「フリー (free)」の時、会議は非議長制御モードである。このトークンが「獲得 (grabbed)」の時、会議は 議長制御モードであり、トークンの獲得者は会議議長である。

会議議長トークンの取得に際して、会議議長 (それがトップ GCC プロバイダ自身ではないなら) は、トップ GCC プロバイダを受取人として、MCS-TOKEN-GIVE-要求を送信して、その譲渡を申し出ることによって、トップ GCC プロバイダに対しトークンを所有していることを表明する。トップ GCC プロバイダは、会議議長の所有するトークンをそのままとするために、MCS-TOKEN-GIVE-指示の結果として否定応答する。トークンが現在の会議議長によって後で再び譲渡されるなら、トップ GCC プロバイダはそれを GCC-CONDUCTOR-GIVE 動作の一部と解釈し、自動的に拒否する必要はない。

8.5.1. 議長権の獲得

GCC-CONDUCTOR-ASSIGN-要求 プリミティブの受信時に、GCC プロバイダは会議が議長制御可能であるかどうか判断するために、最初に会議プロファイルを調査する。議長制御可能でなければ、会議が議長制御が不可能であることを示す失敗の結果を伴った GCC-CONDUCTOR-ASSIGN-確認 プリミティブを生成する。会議が議長制御可能なら、GCC プロバイダはそれが現在会議議長であるかどうか判断するために、そのローカルなデータベースを調査する。議長であるなら、「成功 (successful)」の結果を伴った GCC-CONDUCTOR-ASSIGN-確認 プリミティブを生成して、コントロール GCCSAP にそれを発行する。議長でなければ、獲得するトークン ID として会議議長トークンを指定した MCS プリミティブ MCS-TOKEN-GRAB-要求 を発行することによって議長トークンを獲得しようと試みる。

MCS-TOKEN-GRAB-確認 プリミティブの受信時に、GCC プロバイダは結果パラメータを調査する。結果が「成功 (successful)」なら、GCC プロバイダは、送出トークン ID として会議議長トークンを、そしてトークンを受け取るユーザ ID としてトップ GCC プロバイダのノード ID を指定した MCS プリミティブ MCS-TOKEN-GIVE-要求 を発行する。MCS-TOKEN-GIVE-確認 プリミティブの受信時に、GCC プロバイダは結果パラメータを調査する。結果が不成功なら、GCC プロバイダは「成功 (successful)」の結果を示す GCC-CONDUCTOR-ASSIGN-確認 プリミティブをローカルに生成して、コントロール GCCSAP にそれを発行する。

現在の会議議長としてトップ GCC プロバイダが承認していない提供者ユーザ ID から、提供するトークン ID として会

議議長トークンを指定した MCS-TOKEN-GIVE-指示を受信したときは、トップ GCC プロバイダは会議議長として提供者を承認するためにそのデータベースを更新する。それはまた、チャンネル ID が GCC-BROADCAST-CHANNEL で、データプライオリティが Top で、そしてデータ フィールドに ConductorAssignIndication PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することによって、会議中のすべてのノードに送信する。ConductorAssignIndication PDU の内容を表 8-47/JT-T124 に示す。

MCS-TOKEN-GRAB-確認の結果パラメータが失敗の結果であるか、MCS-TOKEN-GIVE-確認の結果パラメータが「成功 (successful)」の結果であるなら、GCC プロバイダは不十分な結果を反映している否定的な結果を示す GCC-CONDUCTOR-ASSIGN-確認 プリミティブを生成し、コントロール GCCSAP にそれを発行する。

表 8-47/JT-T124 ConductorAssignIndication GCCPDU
(ITU-T T.124)

内容	ソース	シンク
議長ノード (Conducting Node)	トップ GCC プロバイダ	目的 GCC プロバイダ

8.5.2. 議長権の解放

GCC-CONDUCTOR-RELEASE-要求 プリミティブの受信に従って、GCC プロバイダは最初に、それが現在会議議長であるかどうか判断するために、そのローカルのデータベースを調査する。会議議長でなければ、それは否定的な結果を示す GCC-CONDUCTOR-RELEASE-確認 プリミティブを生成して、コントロール GCCSAP にそれを発行する。それが現在の議長なら、チャンネル ID が GCC-Broadcast-Channel で、データプライオリティが Top で、データ フィールドに ConductorReleaseIndication PDU を含んだ MCS-UNIFORM-SEND-DATA-要求を発行することによって、会議中のすべてのノードに送信する。ConductorReleaseIndication PDU の内容を表 8-48/JT-T124 に示す(この PDU にはパラメータがない)。それから、解放されるトークン ID として会議議長トークンを指定した MCS-TOKEN-RELEASE-要求 プリミティブを発行して、その議長トークンを解放する。

MCS-TOKEN-RELEASE-確認 プリミティブの受信に従って、GCC プロバイダは「成功 (successful)」の結果を示す GCC-CONDUCTOR-RELEASE-確認 プリミティブをローカルに生成して、コントロール GCCSAP にそれを発行する。

表 8-48/JT-T124 ConductorReleaseIndication GCCPDU
(ITU-T T.124)

内容	ソース	シンク
-- パラメータなし --		

8.5.3. 議長割り当てと解放の指示

議長制御モードと非議長制御モードで違う動作を規定するアプリケーション プロトコルをサポートするノードは ConductorAssignIndication PDU と ConductorReleaseIndication PDU の受信に応答する。このようなアプリケーション プロトコルをサポートしないノードは、これらの指示を無視してもよい。

ConductorAssignIndication PDU を含んだ MCS-UNIFORM-SEND-DATA-指示を受信した時に、GCC プロバイダは会議が議長制御可能であるかどうか判断するために会議プロファイルを調査する。議長制御可能でなければ、この PDU を無視する。会議が議長制御可能であるなら、GCC-CONDUCTOR-ASSIGN-指示 プリミティブを生成して、そ

れをコントロール GCCSAP とすべての登録されたアプリケーション プロトコル エンティティの GCCSAP に発行する。最初に、MCS-UNIFORM-SEND-DATA-指示の「送信ユーザ ID(Sender user id)」フィールドが、トップ GCC プロバイダによって PDU が伝送されたことを示しているかどうかを判断する。トップ GCC プロバイダによって PDU が伝送されたことを示していなければ、PDU は無視され、指示プリミティブは生成されない。PDU がトップ GCC プロバイダによって伝送されたなら、プリミティブのノード ID パラメータは受信 PDU 中の議長ノード パラメータで指示される値に設定される。議長制御権プリミティブをサポートするノードのために、GCC プロバイダはまた、議長のノード ID と、会議が現在議長制御モードである事をそのローカルのデータベースに格納する。

ConductorReleaseIndication PDU を含んだ MCS-UNIFORM-SEND-DATA-指示を受信した時に、GCC プロバイダは GCC-CONDUCTOR-RELEASE-指示 プリミティブを生成して、それをコントロール GCCSAP とすべての登録されたアプリケーション プロトコル エンティティの GCCSAP に発行する。最初に、MCS-UNIFORM-SEND-DATA-指示の「送信ユーザ ID(Sender user id)」フィールドが、トップ GCC プロバイダによって、あるいは、現在の議長として記録されているノードによって PDU が伝送されたことを示しているかどうか判断する。もし示していなければ、PDU は無視され、指示プリミティブは生成されない。議長権プリミティブをサポートするノードのために、GCC プロバイダはまた、会議がもう議長制御モードでないという事をそのローカルのデータベースに格納する。GCC プロバイダはまた、議長制御モードの許可が与えられなかったことを指示するためにそのローカルに格納された許可フラグを設定する。GCC プロバイダは、対応する指示プリミティブの生成過程で受信した ConductorAssignIndication と ConductorReleaseIndication PDU の順序を維持する。

議長権プリミティブをサポートするノードの GCC プロバイダは、会議が既に確立された後、新たに登録するアプリケーション プロトコル エンティティと関連付けられた GCCSAP に GCC-CONDUCTOR-ASSIGN あるいは GCC-CONDUCTOR-RELEASE-指示を発行する。GCC プロバイダによって周知の現在の議長権の状態がこれらの2つのプリミティブのいずれを発行するべきかを判断する。

8.5.4. 議長権の委譲依頼

GCC-CONDUCTOR-PLEASE-要求 プリミティブの受信時に、GCC プロバイダはまず、会議が議長制御モードであるかどうか判断するために、そのローカルのデータベースを調査する。議長制御モードでなければ、会議が議長制御モードでないことを示す否定的な結果の GCC-CONDUCTOR-PLEASE-確認を生成して、それをコントロール GCCSAP に発行する。議長制御モードなら、GCC プロバイダは、要求されるトークンとして会議議長トークンを指定した MCS-TOKEN-PLEASE-要求を発行する。また、「成功 (successful)」の結果を示す GCC-CONDUCTOR-PLEASE-確認を生成して、それをコントロール GCCSAP に発行する。

トークン ID として会議議長トークンを指定された MCS-TOKEN-PLEASE-指示の受信時に、GCC プロバイダは、MCS-TOKEN-PLEASE-指示で受信したユーザ ID パラメータから得た、要求ノードのノード ID を示している GCC-CONDUCTOR-PLEASE-指示 プリミティブを生成して、それをコントロール GCCSAP に発行する。

8.5.5. 議長権の委譲

GCC-CONDUCTOR-GIVE-要求 プリミティブの受信時に、GCC プロバイダはまず、それが現在の会議議長であるかどうか判断するために、そのローカルのデータベースを調査する。議長でなければ、ノードが現在議長でないことを示す、否定的な結果の GCC-CONDUCTOR-GIVE-確認を生成して、それをコントロール GCCSAP に発行する。さもなければ、GCC プロバイダは、トークン ID が 会議議長トークンで、受取人ノードが GCC-CONDUCTOR-GIVE-要求で指定された、MCS-TOKEN-GIVE-要求を発行することによって、指定されたノードに議長権を委譲しようと試みる

べきである。

MCS-TOKEN-GIVE-確認 プリミティブの受信時に、GCC プロバイダは結果パラメータを調査する。結果が「成功 (successful)」なら、GCC プロバイダはローカルに、「成功 (successful)」結果を示す GCC-CONDUCTOR-GIVE-確認 プリミティブを生成して、それをコントロール GCCSAP に発行する。MCS-TOKEN-GIVE-確認の結果パラメータが失敗の結果を示しているなら、GCC プロバイダは MCS-TOKEN-GRAB-確認 (注: MCS-TOKEN-GIVE-確認と思われる) によって指示された結果を反映して、否定的な結果を示す GCC-CONDUCTOR-GIVE-確認 プリミティブを生成し、それをコントロール GCCSAP に発行する。

トップ GCC プロバイダ以外の GCC プロバイダは、会議議長トークンを指定した MCS-TOKEN-GIVE-指示 プリミティブの受信時に GCC-CONDUCTOR-GIVE-指示 プリミティブを生成して、それをコントロール GCCSAP に発行する。GCC-CONDUCTOR-GIVE-応答の受信時に、GCC プロバイダは結果パラメータを調査する。結果が「成功 (successful)」なら、GCC プロバイダはトークンが受け入れられたことを示す MCS-TOKEN-GIVE-応答を発行する。GCC-CONDUCTOR-GIVE-応答がトークンが受け入れられないことを示すなら、GCC プロバイダはトークンが「ユーザ拒否(user rejected)」であることを示す MCS-TOKEN-GIVE-応答を発行する。

議長権プリミティブをサポートしないノードが MCS-TOKEN-GIVE-指示を受信したら、GCC プロバイダは、「ユーザ拒否(user rejected)」の結果を示す MCS-TOKEN-GIVE-応答で、応答する。

トップ GCC プロバイダが現在の議長として記録していない提供者ユーザ ID からの会議議長トークンを規定する MCS-TOKEN-GIVE-指示 プリミティブの受信時に、トップ GCC プロバイダは、自動的にトークンを拒否して、8.5.1 節で規定されるように、応答する。しかしながら、それが既に提供者を議長として認識するなら、トップ GCC プロバイダは、他のいずれかの GCC プロバイダのように、上述した GCC-CONDUCTOR-GIVE-指示 プリミティブを生成する。

8.5.6. 議長権の状態の取得

GCC-CONDUCTOR-INQUIRE-要求の受信時に、GCC プロバイダはまず、会議が現在議長制御モードであるかどうか判断するためにそれ自身のローカルのデータベースを調査する。議長制御モードでなければ、GCC プロバイダは会議が議長制御モードでないことを示す GCC-CONDUCTOR-INQUIRE-確認を生成して、それをコントロール GCCSAP に発行する。さもなければ、それは会議が議長制御モードであることを示す確認プリミティブを生成し、現在の議長としてそのローカルのデータベースに格納されているノードを、(確認プリミティブの中の)議長ノードのノード ID として指定する。この場合、もしノードがちょうど会議に参加して、そしてまだ議長 (注: トップ GCC プロバイダと思われる) から ConductorAssignIndication PDU を受信していないなら (そして、そのローカルのデータベースでまだ議長のノード ID を持たないなら)、GCC プロバイダは、GCC-CONDUCTOR-INQUIRE-確認を生成する前に、この PDU が受け取られるまで待つべきである。

もし会議が議長制御モードなら、GCC プロバイダは GCC-CONDUCTOR-INQUIRE-確認に含まれる許可フラグの設定を判断しなければならない。その判断は、ローカルのノードが現在の議長であるかどうか (その場合許可が与えられていると考えられる)、あるいは、議長でないなら、非議長制御モードから議長制御モードへの最後の移行後に受信した最も最近の ConductorPermissionGrantIndication PDU の許可リストにローカルノードのノード ID が記されているかどうかによる。このリストにローカルのノードがある場合、このフラグは TRUE に設定される。そうでないか、または非議長制御モードから議長制御モードへの最も最近の移行後に受信した ConductorPermissionGrantIndication PDU がなかったなら、このフラグは FALSE に設定される。

8.5.7. 新ノードの会議参加時の議長権アナウンス

トップ GCC プロバイダが、会議に新しいノードがあることを示すフラグセットを伴った会議名簿のアップデートを受信し、かつ、会議が現在議長制御モードであるなら、トップ GCC プロバイダは会議中のすべてのノードへ ConductorAssignIndication PDU を送信する。PDU の送信はチャンネル ID が GCC-BROADCAST-CHANNEL で、データプライオリティが Top で、そしてデータ フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。ConductorAssignIndication PDU の内容を表 8-47/JT-T124 に示す。議長ノード パラメータは現在の議長のノード ID を示す。

トップ GCC プロバイダが、会議に新しいノードがあることを示すフラグセットを伴った会議名簿のアップデートを受信し、かつ、会議が現在議長制御モードでないなら、トップ GCC プロバイダは、すべてのノードに ConductorReleaseIndication PDU を送信する。PDU の送信は、チャンネル ID が GCC-BROADCAST-CHANNEL で、データプライオリティが Top で、データ フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。ConductorReleaseIndication PDU の内容を表 8-48/JT-T124 に示す。

トップ GCC プロバイダが、自身が送信した ConductorAssignIndication PDU を受信する前に、他のノードから現在の議長権の状態の変更を示す ConductorReleaseIndication PDU を受信した場合、トップ GCC プロバイダは新しい状態を反映するために ConductorReleaseIndication PDU を再度発行する。

会議に加入したばかりのノードは、ConductorAssignIndication PDU あるいは ConductorReleaseIndication PDU を受信するまで、議長関連の要求プリミティブのいずれの処理も延期する。これは、すべての議長制御関連の要求プリミティブは、ローカルのノードがそれ自身のローカルの議長権の状態のデータベースをチェックすることを必要とするが、そのデータベースにはこれらの PDU の一つを受信するまで会議の実際の状態と一致する状態がないからである。

8.5.8. 予期せぬ議長の切断

会議の議長が、会議から不意に切断するなら、それは会議が現在非議長制御モードであることを示す ConductorReleaseIndication PDU をすべてのノードに送信する機会がないかも知れない。それぞれのノード（もしそれが議長権プリミティブをサポートするなら）において、GCC プロバイダが受信した MCS-DETACH-USER-指示をモニタして、その指示プリミティブのユーザ ID を現在の議長であるノードのノード ID と比較する。それが一致するなら、GCC プロバイダは、もはや会議が議長制御モードでないことをそのローカルのデータベースに格納する。そして、GCC プロバイダは GCC-CONDUCTOR-RELEASE-指示を生成して、それをコントロール GCCSAP とすべての登録されたアプリケーション プロトコル エンティティの GCCSAP に発行する。GCC プロバイダはまた、議長制御モードの許可が与えられなくなったことを示すように許可フラグをローカルに設定する。

8.5.9. 議長制御モード許可の問い合わせ

コントロール GCCSAP 経由の GCC-CONDUCTOR-PERMISSION-ASK-要求 プリミティブの受信時に、このプリミティブをサポートする GCC プロバイダは最初に、会議が議長制御モードであるかどうか判断する。議長制御モードでなければ、それは結果が「議長制御モードでない間 (not-in-conducted-mode)」を示す GCC-CONDUCTOR-PERMISSION-ASK-確認 プリミティブを生成して、それをコントロール GCCSAP に発行する。議長制御モードであるなら、その会議の議長に ConductorPermissionAskIndication PDU を送信する。PDU の送信は、チャンネル ID が GCC-BROADCAST-CHANNEL で、データプライオリティが High で、データ フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。ConductorAskIndication PDU の内容を表 8-49/JT-T124 に示す。成功で完了される時、GCC プロバイダは「成功 (successful)」の結果を示す GCC-CONDUCTOR-

PERMISSION-ASK-確認 プリミティブを生成して、それをコントロール GCCSAP に発行する。

表 8-49/JT-T124 ConductorPermissionAskIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
獲得/解放フラグ (Grant/Release Flag)	要求	指示

ConductorPermissionAskIndication PDU の受信時に、GCC プロバイダは自身が現在会議議長であるかどうか判断する。議長でなければ、PDU を無視して、それ以上の動作を行なわない。議長で、GCC プロバイダが対応するプリミティブをサポートするならば、GCC-CONDUCTOR-PERMISSION-ASK-指示 プリミティブを生成して、それをコントロール GCCSAP に発行する。要求者パラメータのノード ID は、受信した MCS-UNIFORM-SEND-DATA-指示の送信ユーザ ID (Sender User Id) パラメータがセットされるべきである。GCC プロバイダは、対応する指示 プリミティブの生成において、受信した ConductorPermissionAskIndication PDU の順序を維持する。

8.5.10. 議長制御モード許可の授与

コントロール GCCSAP 経由の GCC-CONDUCTOR-PERMISSION-GRANT-要求の受信時に、このプリミティブをサポートする GCC プロバイダはまず、自身が指定された会議の現在の議長であるかどうか判断する。議長でなければ、拒否の理由(reason)として「議長でない(not-conductor)」を指定した GCC-CONDUCTOR-PERMISSION-GRANT-確認を生成して、それをコントロール GCCSAP に発行する。その時、それ以上の動作を行なわない。ノードが現在の会議の議長であるならば、GCC プロバイダは、会議中のすべてのノードに ConductorPermissionGrantIndication PDU を放送する。それは、チャンネル ID が GCC-BROADCAST-CHANNEL で、データプライオリティが Top で、データフィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行う。ConductorAssignIndication PDU の内容を表 8-49/JT-T124 に示す。許可と待ちリストのパラメータは、要求プリミティブの対応するパラメータで供給されたノード ID のリストの内容がセットされる。両方のリストの順序は維持されている。成功で完了される時、GCC プロバイダは「成功(successful)」の結果を示す GCC-CONDUCTOR-PERMISSION-GRANT-確認 プリミティブを生成して、それをコントロール GCCSAP に発行する。

表 8-50/JT-T124 ConductorPermissionGrantIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
許可リスト (Permission List)	要求	指示
待ちリスト (Waiting List) (オプション)	要求	指示

ConductorPermissionGrantIndication PDU の受信時に、対応するプリミティブをサポートする GCC プロバイダはまず、会議が現在議長制御モードであるかどうか判断する。議長制御モードでなければ、受信した PDU を無視して、それ以上の動作を行なわない。会議が議長制御モードであるならば、この PDU が現在の会議の議長であるノードから受信したかどうか判断する。もし議長ノードから受信したものでなければ、同様にこの PDU を無視して、それ以上の動作

を行わない。もし送信元が現在の議長であるなら、それは GCC-CONDUCTOR-PERMISSION-GRANT-指示 プリミティブを生成して、それをコントロール GCCSAP とすべての登録されたアプリケーション プロトコル エンティティの GCCSAP に発行する。指示プリミティブ中のパラメータ「許可を与えられたノードのリスト (List of Nodes Granted Permission)」と「許可を待っているノードのリスト (List of Nodes Waiting for Permission)」は、これらのリストの順序を維持したまま、PDU の対応するパラメータ値が直接セットされる。GCC プロバイダは「許可リスト (Permission List)」パラメータを調査して、ローカルのノードがこのリストに含まれているかどうか判断する。もしローカルのノードが含まれていれば、指示プリミティブの「許可フラグ (Permission Flag)」パラメータを TRUE に設定する。さもなければ、このフラグを FALSE に設定する。GCC プロバイダが、対応する指示プリミティブの生成において受信した ConductorPermissionGrantIndication PDU の順序を維持する。

8.6. その他の機能

8.6.1. 時間制約のある会議

GCC-CONFERENCE-TIME-REMAINING-要求 プリミティブ受信時に、GCC プロバイダは指定された会議中のすべてのノードに ConferenceTimeRemainingIndication PDU を送信する。PDU の送信はチャンネル ID(Channel ID)が GCC-BROADCAST-CHANNEL で、データプライオリティ(data priority)が high で、データ(Data)フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。ConferenceTimeRemainingIndication PDU の内容を表 8-51 /JT-T124 に示す。残り時間(Time Remaining)パラメータの値は、要求プリミティブの内容から得られる。また、GCC プロバイダは「成功(successful)」の結果を指示する GCC-CONFERENCE-TIME-REMAINING-確認 プリミティブを生成して、それをコントロール GCCSAP に発行する。

表 8-51/JT-T124 ConferenceTimeRemainingIndication GCCPDU
(ITU-T T.124)

内容	ソース	シンク
残り時間 (Time Remaining)	要求	指示
ノード ID (Node Identifier) (オプション)	要求	指示

ConferenceTimeRemainingIndication PDU 受信時に、GCC プロバイダは、選択的に GCC-CONFERENCE-TIME-REMAINING-指示 プリミティブを生成し、それをコントロール GCCSAP に発行する。残り時間(Time Remaining)パラメータは、PDU の内容から得られる。発信元ノード(Source Node)パラメータは、受信した MCS-UNIFORM-SEND-DATA-指示の中の送信ユーザ ID(Sender User ID)に記述される。

GCC-CONFERENCE-TIME-INQUIRE-要求 プリミティブ受信時に、GCC プロバイダは会議の召集者に ConferenceTimeInquireIndication PDU を送信する。PDU の送信はチャンネル ID(Channel ID)が GCC-CONVENER-CHANNEL で、データプライオリティ(data priority)が High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceTimeInquireIndication PDU の内容を表 8-52/JT-T124 に示す。GCC プロバイダは GCC-CONFERENCE-TIME-INQUIRE-確認 プリミティブを生成して、それをコントロール GCCSAP に発行する。

表 8-52/JT-T124 ConferenceTimeInquireIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
ノード固有タイムフラグ (Node-Specific Time Flag)	要求	指示

ConferenceTimeInquireIndication PDU の受信時に、その GCC プロバイダ (会議の召集者) は、選択的に GCC-CONFERENCE-TIME-INQUIRE-指示プリミティブを生成し、それをコントロール GCCSAP に発行する。MCS-SEND-DATA-指示の発信元ノードパラメータの中で示された、要求者のノード ID が、要求ノードのノード ID(Node ID) パラメータとして、GCC-CONFERENCE-TIME-INQUIRE-指示プリミティブの中で使用される。

GCC-CONFERENCE-EXTEND-要求プリミティブの受信時に、GCC プロバイダは会議の召集者に ConferenceTimeExtendIndication PDU を送信する。PDU の送信はチャンネル ID(Channel ID)が GCC-CONVENER-CHANNEL で、データプライオリティ(data priority)が High で、データ(Data)フィールドに PDU を含めた MCS-SEND-DATA-要求を発行することにより行われる。ConferenceTimeExtendIndication PDU の内容を表 8 - 53/JT-T124 に示す。GCC プロバイダは GCC-CONFERENCE-EXTEND-確認 プリミティブを生成し、それをコントロール GCCSAP に発行する。

表 8-53/JT-T124 ConferenceTimeExtendIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
延長時間 (Time to Extend)	要求	指示
ノード固有タイムフラグ (Node-Specific Time Flag)	要求	指示

ConferenceTimeExtendIndication PDU の受信時に、GCC プロバイダ (会議の召集者) は、選択的に GCC-CONFERENCE-EXTEND-指示プリミティブを生成し、それをコントロール GCCSAP に発行する。MCS-SEND-DATA-指示の発信元ノードパラメータの中で示される、要求者のノード ID は、GCC-CONFERENCE-EXTEND-指示プリミティブ中の「要求ノード(RequestingNode)」パラメータのノード ID として使われる。

8.6.2. 会議アシスタンスの要求

GCC-CONFERENCE-ASSISTANCE-要求 プリミティブの受信時に、GCC プロバイダは指定された会議中のすべてのノードに ConferenceAssistanceIndication PDU を送信する。PDU の送信は「チャンネル ID(Channel ID)」が GCC-BROADCAST-CHANNEL で、データプライオリティ(data priority)が High で、「データ(Data)」フィールドに PDU を含めた MCS-UNIFORM-SEND-DATA-要求 あるいは MCS-SEND-DATA-要求を発行することにより行われる。ConferenceAssistanceIndication PDU の内容を表 8-54/JT-T 124 に示す。GCC プロバイダは、動作が成功したかどうかを指示する GCC-CONFERENCE-ASSISTANCE-確認 プリミティブを生成する。

表 8-54/JT-T124 ConferenceAssistanceIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
ユーザ データ (User Data) (オプション)	要求	指示

ConferenceAssistanceIndication PDU の受信時に、この機能をサポートするノードは、GCC-CONFERENCE-ASSISTANCE-指示 プリミティブを生成し、それをコントロール GCCSAP に発行するかもしれない。発信元ノードパラメータは、受信した MCS-UNIFORM-SEND-指示の中の「送信ユーザ ID(Sender User ID)」から得られる。

8.6.3. テキストメッセージの放送

GCC-TEXT-MESSAGE-要求 プリミティブの受信時に、宛先ノードパラメータが要求プリミティブに含まれていないなら、GCC プロバイダは指定された会議中のすべてのノードに TextMessageIndication PDU を送信する。PDU の送信は「チャンネル ID(Channel ID)」が GCC-BROADCAST-CHANNEL で、「データプライオリティ(data priority)」が「High」で、「データ(Data)」フィールド内に PDU を含める MCS-SEND-DATA-要求 あるいは MCS-UNIFORM-SEND-DATA-要求を発行することにより行われる。宛先ノードが指示されたならば、GCC プロバイダは同一 PDU を送信する。PDU の送信は、要求ノードのノード ID を指定しているチャンネル ID(Channel ID)で、データプライオリティ(data priority)が High で、データ(Data)フィールド内に PDU を含めた MCU-SEND-DATA-要求を発行することにより行われる。TextMessageIndication PDU の内容を表 8-55/JT-T124 に示す。それから、GCC プロバイダは、その動作が成功したかどうかを指示をする GCC-TEXT-MESSAGE-確認 プリミティブを生成する。

表 8-55/JT-T124 TextMessageIndication GCCPDU

(ITU-T T.124)

内容	ソース	シンク
メッセージ (Message)	要求	指示

TextMessageIndication PDU の受信時に、この機能をサポートするノードは、GCC-TEXT-MESSAGE-指示 プリミティブを生成し、それをコントロール GCCSAP に発行する。発信元ノードパラメータは、受信した MCS-UNIFORM-SEND-指示の中の「送信ユーザ ID(Sender User ID)」から得られる。

8.7. GCCPDU 定義

GCCPDU の構造は、ITU-T 勧告 X.680 の ASN.1 表記を使用して次のように記述される。すべての GCCPDU は、ベーシックな割当とは異なる方法を用いた ITU-T 勧告 X.691 のパケット符号化規則を適用することによって伝送時に符号化される。

注 - GCC プロトコル定義中の AutomaticTag の使用は、この定義内に含まれる SEQUENCE 構造と CHOICE 構造の順序が実際の符号化値に影響することを意味する。

GCC-PROTOCOL DEFINITIONS AUTOMATIC TAGS ::= BEGIN

-- Export all symbols

-- =====
-- Part 1: Elements of which messages are composed
-- =====

ChannelID ::= INTEGER (1..65535)
StaticChannelID ::= INTEGER (1..1000) -- Those assigned by specifications
DynamicChannelID ::= INTEGER (1001..65535) -- Those created and deleted by MCS
UserID ::= DynamicChannelID

TokenID ::= INTEGER (1..65535)
StaticTokenID ::= INTEGER (1..16383) -- Those assigned by specifications
DynamicTokenID ::= INTEGER (16384..65535) -- Those assigned by the registry

Time ::= INTEGER (-2147483648..2147483647) -- Time in seconds
Handle ::= INTEGER (0..4294967295) -- 32-bit value

H221NonStandardIdentifier ::= OCTET STRING (SIZE (4..255))
-- First four octets shall be country code
and
-- Manufacturer code, assigned as specified
in
-- H.221 Annex A for NS-cap and NS-comm

Key ::= CHOICE -- Identifier of a standard or non-standard
object
{
 object OBJECT IDENTIFIER,
 h221NonStandard H221NonStandardIdentifier
}

NonStandardParameter ::= SEQUENCE
{
 key Key,
 data OCTET STRING
}

TextString ::= BMPString (SIZE (0..255)) -- Basic Multilingual Plane of ISO/IEC
10646-1 (Unicode)

simpleTextFirstCharacter UniversalString ::= {0, 0, 0, 0}

simpleTextLastCharacter UniversalString ::= {0, 0, 0, 255}

SimpleTextString ::= BMPString (SIZE (0..255)) (FROM
(simpleTextFirstCharacter..simpleTextLastCharacter))

SimpleNumericString ::= NumericString (SIZE (1..255)) (FROM ("0123456789"))

DialingString ::= NumericString (SIZE (1..16)) (FROM ("0123456789"))

SubAddressString ::= NumericString (SIZE (1..40)) (FROM ("0123456789"))

ExtraDialingString ::= TextString (SIZE (1..255)) (FROM ("0123456789#*))

UserData ::= SET OF SEQUENCE

```
{  
  key      Key,  
  value    OCTET STRING OPTIONAL  
}
```

Password ::= SEQUENCE

```
{  
  numeric  SimpleNumericString,  
  text     SimpleTextString OPTIONAL,  
  ...  
}
```

PasswordSelector ::= CHOICE

```
{  
  numeric  SimpleNumericString,  
  text     SimpleTextString,  
  ...  
}
```

ChallengeResponseItem ::= CHOICE

```
{  
  passwordString PasswordSelector,  
  responseData   UserData,  
  ...  
}
```

ChallengeResponseAlgorithm ::= CHOICE

```
{  
  passwordInTheClear  NULL,  
  nonStandardAlgorithm NonStandardParameter,  
  ...  
}
```

ChallengeItem ::= SEQUENCE

```
{  
  responseAlgorithm ChallengeResponseAlgorithm,  
  challengeData     UserData,  
  ...  
}
```

ChallengeRequest ::= SEQUENCE

```
{  
  challengeTag      INTEGER,  
  challengeSet      SET OF ChallengeItem, -- Set of algorithms offered for response  
  ...  
}
```

```

ChallengeResponse ::= SEQUENCE
{
    challengeTag          INTEGER,
    responseAlgorithm    ChallengeResponseAlgorithm,
                                -- Specific algorithm selected from the set
of
    responseltem        ChallengeResponseltem,
                                -- items presented in the ChallengeRequest
    ...
}

```

```

PasswordChallengeRequestResponse ::= CHOICE
{
    passwordInTheClear   PasswordSelector,
    challengeRequestResponse SEQUENCE
    {
        challengeRequest  ChallengeRequest OPTIONAL,
        challengeResponse  ChallengeResponse OPTIONAL,
        ...
    },
    ...
}

```

```

ConferenceName ::= SEQUENCE
{
    numeric   SimpleNumericString,
    text     SimpleTextString OPTIONAL,
    ...
}

```

```

ConferenceNameSelector ::= CHOICE
{
    numeric   SimpleNumericString,
    text     SimpleTextString,
    ...
}

```

```

ConferenceNameModifier ::= SimpleNumericString

```

```

Privilege ::= ENUMERATED
{
    terminate   (0),
    ejectUser   (1),
    add         (2),
    lockUnlock  (3),
    transfer    (4),
    ...
}

```

```

TerminationMethod ::= ENUMERATED
{
    automatic   (0),
    manual      (1),
    ...
}

```

```

NetworkAddress ::= SEQUENCE (SIZE (1..64)) OF CHOICE -- Listed in order of use
{
    aggregatedChannel SEQUENCE

```

```

{
  transferModes      SEQUENCE      -- One or more
  {
    speech           BOOLEAN,
    voice-band       BOOLEAN,
    digital-56k      BOOLEAN,
    digital-64k      BOOLEAN,
    digital-128k     BOOLEAN,
    digital-192k     BOOLEAN,
    digital-256k     BOOLEAN,
    digital-320k     BOOLEAN,
    digital-384k     BOOLEAN,
    digital-512k     BOOLEAN,
    digital-768k     BOOLEAN,
    digital-1152k    BOOLEAN,
    digital-1472k    BOOLEAN,
    digital-1536k    BOOLEAN,
    digital-1920k    BOOLEAN,
    packet-mode      BOOLEAN,
    frame-mode       BOOLEAN,
    atm              BOOLEAN,
    ...
  },

  internationalNumber DialingString,
  subAddress          SubAddressString OPTIONAL,
  extraDialing        ExtraDialingString OPTIONAL,

  highLayerCompatibility SEQUENCE
  {
    telephony3kHz    BOOLEAN,
    telephony7kHz    BOOLEAN,
    videotelephony   BOOLEAN,
    videoconference  BOOLEAN,
    audiographic     BOOLEAN,
    audiovisual      BOOLEAN,
    multimedia       BOOLEAN,
    ...
  } OPTIONAL,
  ...
},

transportConnection SEQUENCE
{
  nsapAddress        OCTET STRING (SIZE (1..20)),
  transportSelector  OCTET STRING OPTIONAL
},

nonStandard         NonStandardParameter,
...
}

NodeType ::= ENUMERATED
{
  terminal           (0),
  multiportTerminal (1),
  mcu                (2),
  ...
}

```



```

NodeProperties ::= SEQUENCE
{
    managementDevice    BOOLEAN,           -- Is the node a device such as a reservation
system
    peripheralDevice    BOOLEAN,           -- Is the node a peripheral to a primary node
    ...
}

AsymmetryIndicator ::= CHOICE
{
    callingNode         NULL,
    calledNode          NULL,
    unknown             INTEGER (0..4294967295) -- Uniformly distributed 32-bit random
number
}

AlternativeNodeID ::= CHOICE
{
    h243NodeID         OCTET STRING (SIZE (2)),
    ...
}

ConferenceDescriptor ::= SEQUENCE
{
    conferenceName      ConferenceName,
    conferenceNameModifier  ConferenceNameModifier OPTIONAL,
    conferenceDescription  TextString OPTIONAL,
    lockedConference     BOOLEAN,
    passwordInTheClearRequired  BOOLEAN,
    networkAddress      NetworkAddress OPTIONAL,
    ...
}

NodeRecord ::= SEQUENCE
{
    superiorNode        UserID OPTIONAL,     -- Not present only for the Top GCC Provider
nodeType
    nodeType            NodeType,
nodeProperties
    nodeProperties      NodeProperties,
nodeName
    nodeName           TextString OPTIONAL,
participantsList
    participantsList    SEQUENCE OF TextString OPTIONAL,
siteInformation
    siteInformation     TextString OPTIONAL,
networkAddress
    networkAddress      NetworkAddress OPTIONAL,
alternativeNodeID
    alternativeNodeID   AlternativeNodeID OPTIONAL,
userData
    userData            UserData OPTIONAL,
    ...
}

SessionKey ::= SEQUENCE
{
    applicationProtocolKey  Key,
sessionID
    sessionID           ChannelID OPTIONAL
}

ChannelType ::= ENUMERATED
{
    static              (0),
dynamicMulticast
    dynamicMulticast    (1),
dynamicPrivate
    dynamicPrivate      (2),
}

```

```

    dynamicUserId      (3)
}

ApplicationRecord ::= SEQUENCE
{
    applicationActive      BOOLEAN,      -- Active/Inactive flag
    conductingOperationCapable  BOOLEAN,  -- Maximum one per node per session
    startupChannel        ChannelType OPTIONAL,
    applicationUserID     UserID OPTIONAL, -- User ID assigned to the Application Protocol Entity
    nonCollapsingCapabilities
    {
        capabilityID      CapabilityID,
        applicationData    OCTET STRING OPTIONAL
    } OPTIONAL,
    ...
}

CapabilityID ::= CHOICE
{
    standard              INTEGER (0..65535),  -- Assigned by Application Protocol
specifications
    nonStandard          Key
}

CapabilityClass ::= CHOICE
{
    logical              NULL,
    unsignedMin          INTEGER (0..MAX),      -- Capability value
    unsignedMax          INTEGER (0..MAX),      -- Capability value
    ...
}

EntityID ::= INTEGER (0..65535)

ApplicationInvokeSpecifier ::= SEQUENCE
{
    sessionKey           SessionKey,
    expectedCapabilitySet
    SET OF SEQUENCE
    {
        capabilityID      CapabilityID,
        capabilityClass    CapabilityClass,
        ...
    } OPTIONAL,
    startupChannel      ChannelType OPTIONAL,
    mandatoryFlag      BOOLEAN,      -- TRUE indicates required Application
Protocol Entity
    ...
}

RegistryKey ::= SEQUENCE
{
    sessionKey           SessionKey,
    resourceID           OCTET STRING (SIZE (0..64))
}

RegistryItem ::= CHOICE
{
    channelID           DynamicChannelID,
    tokenID             DynamicTokenID,
}

```

```

    parameter    OCTET STRING (SIZE (0..64)),
    vacant       NULL,
    ...
}

RegistryEntryOwner ::= CHOICE
{
    owned        SEQUENCE
    {
        nodeID    UserID,           -- Node ID of the owning node
        entityID  EntityID         -- Entity ID of the owning
    },
    notOwned     NULL              -- Application Protocol Entity
                                   -- There is no current owner
}

RegistryModificationRights ::= ENUMERATED
{
    owner        (0),
    session      (1),
    public       (2)
}

-----
-- Part 2: PDU Messages
-----

UserIDIndication ::= SEQUENCE
{
    tag          INTEGER,
    ...
}

ConferenceCreateRequest ::= SEQUENCE
{
    conferenceName      ConferenceName,
    convenerPassword    Password OPTIONAL,
    password            Password OPTIONAL,
    lockedConference    BOOLEAN,
    listedConference    BOOLEAN,
    conductibleConference    BOOLEAN,
    terminationMethod   TerminationMethod,
    conductorPrivileges SET OF Privilege OPTIONAL,
    conductedPrivileges SET OF Privilege OPTIONAL,
    nonConductedPrivileges SET OF Privilege OPTIONAL,
    conferenceDescription    TextString OPTIONAL,
    callerIdentifier      TextString OPTIONAL,
    userData             UserData OPTIONAL,
    ...
}
-- MCS-Connect-Provider request user data

ConferenceCreateResponse ::= SEQUENCE
{
    data
    {
        nodeID          UserID,           -- Node ID of the sending node
        tag              INTEGER,
        result           ENUMERATED
        {
            success      (0),
            userRejected (1),
        }
    }
}
-- MCS-Connect-Provider response user

```

```

        resourcesNotAvailable          (2),
        rejectedForSymmetryBreaking    (3),
        lockedConferenceNotSupported   (4),
        ...
    },
    userData          UserData OPTIONAL,
    ...
}

ConferenceQueryRequest ::= SEQUENCE
{
    nodeType          NodeType,
    asymmetryIndicator AsymmetryIndicator OPTIONAL,
    userData          UserData OPTIONAL,
    ...
}

ConferenceQueryResponse ::= SEQUENCE
{
    data
    nodeType          NodeType,
    asymmetryIndicator AsymmetryIndicator OPTIONAL,
    conferenceList    SET OF ConferenceDescriptor,
    result            ENUMERATED
    {
        success        (0),
        userRejected    (1),
        ...
    },
    userData          UserData OPTIONAL,
    ...
}

ConferenceJoinRequest ::= SEQUENCE
{
    ...
    -- MCS-Connect-Provider request user data
    as well as
    ...
    -- MCS-Send-Data on Node ID Channel of
    Top GCC sent
    ...
    -- by the receiver of the MCS-Connect-
    Provider
    conferenceName    ConferenceNameSelector OPTIONAL,
    ...
    -- Required when part of MCS-Connect-
    Provider
    conferenceNameModifier ConferenceNameModifier OPTIONAL,
    tag              INTEGER OPTIONAL, -- Filled in when sent on Node ID Channel of
    Top GCC
    password          PasswordChallengeRequestResponse OPTIONAL,
    convenerPassword PasswordSelector OPTIONAL,
    callerIdentifier  TextString OPTIONAL,
    userData          UserData OPTIONAL,
    ...
}

ConferenceJoinResponse ::= SEQUENCE
{
    ...
    -- MCS-Connect-Provider response user
    data as well as
    ...
    -- MCS-Send-Data on Node ID Channel of
    -- the receiver of the MCS-Connect-Provider
    nodeId           UserID OPTIONAL, -- Node ID of directly connected node only
}

```

```

topNodeID          UserID,          -- Node ID of Top GCC Provider
tag                INTEGER,
conferenceNameAlias ConferenceNameSelector OPTIONAL,
passwordInTheClearRequired BOOLEAN,
lockedConference   BOOLEAN,
listedConference   BOOLEAN,
conductibleConference BOOLEAN,
terminationMethod TerminationMethod,
conductorPrivileges SET OF Privilege OPTIONAL, -- No privilege shall be listed
more than once
conductedPrivileges SET OF Privilege OPTIONAL, -- No privilege shall be listed
more than once
nonConductedPrivileges SET OF Privilege OPTIONAL, -- No privilege shall be listed
more than once
conferenceDescription TextString OPTIONAL,
password              PasswordChallengeRequestResponse OPTIONAL,
result                ENUMERATED
{
    success                (0),
    userRejected           (1),
    invalidConference       (2),
    invalidPassword        (3),
    invalidConvenerPassword (4),
    challengeResponseRequired (5),
    invalidChallengeResponse (6),
    ...
},
userData              UserData OPTIONAL,
...
}

```

ConferenceInviteRequest ::= SEQUENCE

```

{
    conferenceName      ConferenceName,
    nodeID              UserID,          -- Node ID of the sending node
    topNodeID           UserID,          -- Node ID of Top GCC Provider
    tag                 INTEGER,
    passwordInTheClearRequired BOOLEAN,
    lockedConference    BOOLEAN,
    listedConference     BOOLEAN,
    conductibleConference BOOLEAN,
    terminationMethod   TerminationMethod,
    conductorPrivileges SET OF Privilege OPTIONAL, -- No privilege shall be listed
more than once
    conductedPrivileges SET OF Privilege OPTIONAL, -- No privilege shall be listed
more than once
    nonConductedPrivileges SET OF Privilege OPTIONAL, -- No privilege shall be listed
more than once
    conferenceDescription TextString OPTIONAL,
    callerIdentifier      TextString OPTIONAL,
    userData              UserData OPTIONAL,
    ...
}

```

ConferenceInviteResponse ::= SEQUENCE

```

{
    data
    result                ENUMERATED
    {

```

```

        success          (0),
        userRejected    (1),
        ...
    },
    userData            UserData OPTIONAL,
    ...
}

```

ConferenceAddRequest ::= SEQUENCE

```

{
    Top GCC or
    specified
    networkAddress      NetworkAddress,
    requestingNode      UserID,
    tag                 INTEGER,
    addingMCU           UserID OPTIONAL,
    userData            UserData OPTIONAL,
    ...
}
-- MCS-Send-Data on Node ID Channel of
-- Node ID Channel of Adding MCU if

```

ConferenceAddResponse ::= SEQUENCE

```

{
    requester
    tag                 INTEGER,
    result              ENUMERATED
    {
        success          (0),
        invalidRequester (1),
        invalidNetworkType (2),
        invalidNetworkAddress (3),
        addedNodeBusy    (4),
        networkBusy      (5),
        noPortsAvailable (6),
        connectionUnsuccessful (7),
        ...
    },
    userData            UserData OPTIONAL,
    ...
}
-- MCS-Send-Data on Node ID Channel of

```

ConferenceLockRequest ::= SEQUENCE

```

{
    Top GCC
    -- No parameters
    ...
}
-- MCS-Send-Data on Node ID Channel of

```

ConferenceLockResponse ::= SEQUENCE

```

{
    requester
    result              ENUMERATED
    {
        success          (0),
        invalidRequester (1),
        alreadyLocked    (2),
        ...
    },
}
-- MCS-Send-Data on Node ID Channel of

```

```

...
}

ConferenceLockIndication ::= SEQUENCE
{
Broadcast-Channel
    -- No parameters
    ...
}

ConferenceUnlockRequest ::= SEQUENCE
{
Top GCC
    -- No parameters
    ...
}

ConferenceUnlockResponse ::= SEQUENCE
{
requester
    result      ENUMERATED
    {
        success          (0),
        invalidRequester (1),
        alreadyUnlocked  (2),
        ...
    },
    ...
}

ConferenceUnlockIndication ::= SEQUENCE
{
Broadcast-Channel
    -- No parameters
    ...
}

ConferenceTerminateRequest ::= SEQUENCE
{
Top GCC
    reason      ENUMERATED
    {
        userInitiated          (0),
        timedConferenceTermination (1),
        ...
    },
    ...
}

ConferenceTerminateResponse ::= SEQUENCE
{
requester
    result      ENUMERATED
    {
        success          (0),
        invalidRequester (1),
        ...
}

```

-- MCS-Uniform-Send-Data on GCC-
-- or MCS-Send-Data on Node ID Channel
-- MCS-Send-Data on Node ID Channel of
-- MCS-Send-Data on Node ID Channel of
-- MCS-Uniform-Send-Data on GCC-
-- or MCS-Send-Data on Node ID Channel
-- MCS-Send-Data on Node ID Channel of
-- MCS-Send-Data on Node ID Channel of

```

    },
    ...
}

ConferenceTerminateIndication ::= SEQUENCE
{
Broadcast-Channel
    reason      ENUMERATED
    {
        userInitiated          (0),
        timedConferenceTermination (1),
        ...
    },
    ...
}

ConferenceEjectUserRequest ::= SEQUENCE
{
Top GCC
    nodeToEject      UserID,
    reason           ENUMERATED
    {
        userInitiated      (0),
        ...
    },
    ...
}

ConferenceEjectUserResponse ::= SEQUENCE
{
requester
    nodeToEject      UserID,
    result           ENUMERATED
    {
        success          (0),
        invalidRequester (1),
        invalidNode      (2),
        ...
    },
    ...
}

ConferenceEjectUserIndication ::= SEQUENCE
{
Broadcast-Channel
    nodeToEject      UserID,
    reason           ENUMERATED
    {
        userInitiated          (0),
        higherNodeDisconnected (1),
        higherNodeEjected      (2),
        ...
    },
    ...
}

ConferenceTransferRequest ::= SEQUENCE
{
Top GCC

```

-- MCS-Uniform-Send-Data on GCC-

-- MCS-Send-Data on Node ID Channel of

-- Node ID of the node to eject

-- MCS-Send-Data on Node ID Channel of

-- Node ID of the node to eject

-- MCS-Uniform-Send-Data on GCC-

-- Node ID of the node to eject

-- MCS-Send-Data on Node ID Channel of


```

        conferenceName      ConferenceNameSelector, -- Name of conference to transfer to
        conferenceNameModifier ConferenceNameModifier OPTIONAL,
        networkAddress      NetworkAddress OPTIONAL,
        transferringNodes    SET (SIZE (1..65536)) OF UserID OPTIONAL,
        password            PasswordSelector OPTIONAL,
        ...
    }

ConferenceTransferResponse ::= SEQUENCE
{
    requester                -- MCS-Send-Data on Node ID Channel of
    conferenceName          ConferenceNameSelector, -- Name of conference to transfer to
    conferenceNameModifier ConferenceNameModifier OPTIONAL,
    transferringNodes        SET (SIZE (1..65536)) OF UserID OPTIONAL,
    result                  ENUMERATED
    {
        success              (0),
        invalidRequester     (1),
        ...
    },
    ...
}

ConferenceTransferIndication ::= SEQUENCE
{
    Broadcast-Channel        -- MCS-Uniform-Send-Data on GCC-
    conferenceName          ConferenceNameSelector, -- Name of conference to transfer to
    conferenceNameModifier ConferenceNameModifier OPTIONAL,
    networkAddress          NetworkAddress OPTIONAL,
    transferringNodes        SET (SIZE (1..65536)) OF UserID OPTIONAL,
    -- List of Node IDs,
    -- not present if destined for all nodes
    password                PasswordSelector OPTIONAL,
    ...
}

RosterUpdateIndication ::= SEQUENCE
{
    Broadcast-Channel        -- MCS-Send-Data on Node ID Channel or
    -- MCS-Uniform-Send-Data on GCC-
    fullRefresh              BOOLEAN, -- Conference Roster and all
    -- ApplicationProtocol Sessions refreshed
    nodeInformation          SEQUENCE
    {
        nodeRecordList      CHOICE
        {
            noChange        NULL,
            refresh          SET (SIZE (1..65536)) OF SEQUENCE
            -- One for each node in the conference;
            -- no node shall be listed more than once
            {
                nodeID      UserID, -- Node ID of the node
                nodeRecord  NodeRecord
            },
            update          SET (SIZE (1..65536)) OF SEQUENCE
            -- One for each node changing its node
            record;
            -- no node shall be listed more than once
        }
    }
}

```

```

        nodeID          UserID,      -- Node ID of the node
        nodeUpdate     CHOICE
        {
            addRecord   NodeRecord,
            replaceRecord NodeRecord,
            removeRecord NULL,
            ...
        }
    },
    ...
},
rosterInstanceNumber INTEGER (0..65535),
nodesAdded           BOOLEAN,      -- Nodes have been added since last
instance
nodesRemoved        BOOLEAN,      -- Nodes have been removed since last
instance
...
},
applicationInformation SET (SIZE (0..65536)) OF SEQUENCE
-- One for each Application Protocol
Session;
-- all Application Protocol Sessions if full
refresh;
-- no Application Protocol Session shall be
-- listed more than once
{
    sessionKey        SessionKey,
    applicationRecordList CHOICE
    {
        noChange      NULL,
        refresh       SET (SIZE (1..65536)) OF SEQUENCE
-- One for each node with the
-- Application Protocol Session enrolled;
-- no node shall be listed more than once
        {
            nodeID      UserID,      -- Node ID of node
            entityID    EntityID,    -- ID for this Application Protocol Entity at
this node
            applicationRecord ApplicationRecord
        },
        update         SET (SIZE (1..65536)) OF SEQUENCE
-- One for each node modifying its
Application Record;
-- no node shall be listed more than once
        {
            nodeID      UserID,      -- Node ID of node
            entityID    EntityID,    -- ID for this Application Protocol Entity at
this node
            applicationUpdate CHOICE
            {
                addRecord   ApplicationRecord,
                replaceRecord ApplicationRecord,
                removeRecord NULL,
                ...
            }
        },
        ...
    },
    ...
},
applicationCapabilitiesList CHOICE

```

```

    {
        noChange refresh NULL, SET OF SEQUENCE
        {
            capabilityID CapabilityID,
            capabilityClass CapabilityClass,
            numberOfEntities INTEGER (1..65536),
                -- Number of Application Protocol Entities
                -- which issued the capability
            ...
        },
        ...
    },
    rosterInstanceNumber INTEGER (0..65535),
    peerEntitiesAdded BOOLEAN, -- Peer Entities have been added since last instance
    peerEntitiesRemoved BOOLEAN, -- Peer Entities have been removed since last instance
    ...
},
...
}

ApplicationInvokeIndication ::= SEQUENCE
{
    Data -- MCS-Send-Data or MCS-Uniform-Send-Data
        -- on GCC-Broadcast-Channel or Node ID
    Channel
        applicationProtocolEntiyList SET (SIZE (1..65536)) OF ApplicationInvokeSpecifier,
        destinationNodes SET (SIZE (1..65536)) OF UserID OPTIONAL,
            -- List of Node IDs,
            -- not presernt if destined for all nodes
        ...
}

RegistryRegisterChannelRequest ::= SEQUENCE
{
    Top GCC -- MCS-Send-Data on Node ID Channel of
        entityID EntityID,
        key RegistryKey,
        channelID DynamicChannelID,
        ...
}

RegistryAssignTokenRequest ::= SEQUENCE
{
    Top GCC -- MCS-Send-Data on Node ID Channel of
        entityID EntityID,
        key RegistryKey,
        ...
}

RegistrySetParameterRequest ::= SEQUENCE
{
    Top GCC -- MCS-Send-Data on Node ID Channel of
        entityID EntityID,
        key RegistryKey,
        parameter OCTET STRING (SIZE (0..64)),
}

```

```

        modificationRights RegistryModificationRights OPTIONAL,
    ...
}

RegistryRetrieveEntryRequest ::= SEQUENCE
{
    Top GCC -- MCS-Send-Data on Node ID Channel of
        entityID      EntityID,
        key            RegistryKey,
    ...
}

RegistryDeleteEntryRequest ::= SEQUENCE
{
    Top GCC -- MCS-Send-Data on Node ID Channel of
        entityID      EntityID,
        key            RegistryKey,
    ...
}

RegistryMonitorEntryRequest ::= SEQUENCE
{
    Top GCC -- MCS-Send-Data on Node ID Channel of
        entityID      EntityID,
        key            RegistryKey,
    ...
}

RegistryMonitorEntryIndication ::= SEQUENCE
{
    Broadcast-Channel -- MCS-Uniform-Send-Data on GCC-
        key            RegistryKey,
        item           RegistryItem, -- Contents: channel, token, parameter, or
empty
        owner          RegistryEntryOwner,
        modificationRights RegistryModificationRights OPTIONAL,
    ...
}

RegistryAllocateHandleRequest ::= SEQUENCE
{
    Top GCC -- MCS-Send-Data on Node ID Channel of
        entityID      EntityID,
        numberOfHandles INTEGER (1..1024),
    ...
}

RegistryAllocateHandleResponse ::= SEQUENCE
{
    requester -- MCS-Send-Data on Node ID Channel of
        entityID      EntityID,
        numberOfHandles INTEGER (1..1024),
        firstHandle   Handle,
        result        ENUMERATED
        {
            successful (0),
            noHandlesAvailable (1),
            ...
        },
}

```

```

    ...
}

RegistryResponse ::= SEQUENCE
{
    requester                                     -- MCS-Send-Data on Node ID Channel of
    entityID                                     EntityID,                               -- Entity ID of the requesting Application
    Protocol Entity
    primitiveType                               ENUMERATED
    {
        registerChannel      (0),
        assignToken          (1),
        setParameter         (2),
        retrieveEntry        (3),
        deleteEntry          (4),
        monitorEntry         (5),
        ...
    },
    key                                         RegistryKey,                            -- Database index
    item                                       RegistryItem,                          -- Contents: channel, token, parameter, or
    vacant
    owner                                     RegistryEntryOwner,
    modificationRights                       RegistryModificationRights OPTIONAL,
    result                                    ENUMERATED
    {
        successful           (0),
        belongsToOther       (1),
        tooManyEntries       (2),
        inconsistentType     (3),
        entryNotFound        (4),
        entryAlreadyExists   (5),
        invalidRequester     (6),
        ...
    },
    ...
}

ConductorAssignIndication ::= SEQUENCE
{
    Broadcast-Channel
    conductingNode    UserID,
    ...
}

ConductorReleaseIndication ::= SEQUENCE
{
    Broadcast-Channel
    -- No parameters
    ...
}

ConductorPermissionAskIndication ::= SEQUENCE
{
    Broadcast-Channel
    grantFlag        BOOLEAN,
    to release
    ...
}

```

```

ConductorPermissionGrantIndication ::= SEQUENCE
{
    Broadcast-Channel          -- MCS-Uniform-Send-Data on GCC-
    permissionList            SEQUENCE (SIZE (0..65536)) OF UserID,
                               -- Node ID of nodes granted permission
    waitingList                SEQUENCE (SIZE (1..65536)) OF UserID OPTIONAL,
                               -- Node ID of nodes waiting form permission
    ...
}

ConferenceTimeRemainingIndication ::= SEQUENCE
{
    Channel                    -- MCS-Send-Data on GCC-Broadcast-
    timeRemaining              Time,
    nodeID                     UserID OPTIONAL,
    ...
}

ConferenceTimeInquireIndication ::= SEQUENCE
{
    Channel                    -- MCS-Send-Data on GCC-Convener-
    nodeSpecificTimeFlag      BOOLEAN,
                               -- FALSE for conference-wide, TRUE for
    node-specific
    ...
}

ConferenceTimeExtendIndication ::= SEQUENCE
{
    Channel                    -- MCS-Send-Data on GCC-Convener-
    timeToExtend              Time,
    nodeSpecificTimeFlag      BOOLEAN,
                               -- FALSE for conference-wide, TRUE for
    node-specific
    ...
}

ConferenceAssistanceIndication ::= SEQUENCE
{
    Broadcast-Channel          -- MCS-Uniform-Send-Data on GCC-
    userData                   UserData OPTIONAL,
    ...
}

TextMessageIndication ::= SEQUENCE
{
    Data                        -- MCS-Send-Data or MCS-Uniform-Send-
    message                    TextString,
                               -- on GCC-Broadcast-Channel or Node ID
    Channel
    ...
}

FunctionNotSupportedResponse ::= SEQUENCE
{
    request                    RequestPDU
}

NonStandardPDU ::= SEQUENCE
{
    data                       NonStandardParameter,
    ...
}

```

}

=====
-- Part 3: Messages sent as MCS-Connect-Provider user data
=====

ConnectData ::= SEQUENCE

```
{
  t124Identifier    Key,    -- This shall be set to the value {itu recommendation t 124
version(0) 1}
  connectPDU       OCTET STRING
}
```

ConnectGCCPDU ::= CHOICE

```
{
  conferenceCreateRequest      ConferenceCreateRequest,
  conferenceCreateResponse     ConferenceCreateResponse,
  conferenceQueryRequest       ConferenceQueryRequest,
  conferenceQueryResponse      ConferenceQueryResponse,
  conferenceJoinRequest        ConferenceJoinRequest,
  conferenceJoinResponse       ConferenceJoinResponse,
  conferenceInviteRequest      ConferenceInviteRequest,
  conferenceInviteResponse     ConferenceInviteResponse,
  ...
}
```

=====
-- Part 4: Messages sent using MCS-Send-Data or MCS-Uniform-Send-Data
=====

GCCPDU ::= CHOICE

```
{
  request           RequestPDU,
  response          ResponsePDU,
  indication        IndicationPDU
}
```

RequestPDU ::= CHOICE

```
{
  conferenceJoinRequest      ConferenceJoinRequest,
  conferenceAddRequest       ConferenceAddRequest,
  conferenceLockRequest      ConferenceLockRequest,
  conferenceUnlockRequest    ConferenceUnlockRequest,
  conferenceTerminateRequest ConferenceTerminateRequest,
  conferenceEjectUserRequest ConferenceEjectUserRequest,
  conferenceTransferRequest  ConferenceTransferRequest,
  registryRegisterChannelRequest RegistryRegisterChannelRequest,
  registryAssignTokenRequest RegistryAssignTokenRequest,
  registrySetParameterRequest RegistrySetParameterRequest,
  registryRetrieveEntryRequest RegistryRetrieveEntryRequest,
  registryDeleteEntryRequest RegistryDeleteEntryRequest,
  registryMonitorEntryRequest RegistryMonitorEntryRequest,
  registryAllocateHandleRequest RegistryAllocateHandleRequest,
  nonStandardRequest        NonStandardPDU,
  ...
}
```

ResponsePDU ::= CHOICE

```
{  
  conferenceJoinResponse           ConferenceJoinResponse,  
  conferenceAddResponse           ConferenceAddResponse,  
  conferenceLockResponse          ConferenceLockResponse,  
  conferenceUnlockResponse        ConferenceUnlockResponse,  
  conferenceTerminateResponse     ConferenceTerminateResponse,  
  conferenceEjectUserResponse     ConferenceEjectUserResponse,  
  conferenceTransferResponse      ConferenceTransferResponse,  
  registryResponse                RegistryResponse,  
  registryAllocateHandleResponse  RegistryAllocateHandleResponse,  
  functionNotSupportedResponse    FunctionNotSupportedResponse,  
  nonStandardResponse             NonStandardPDU,  
  ...  
}
```

IndicationPDU ::= CHOICE

```
{  
  userIDIndication                UserIDIndication,  
  conferenceLockIndication         ConferenceLockIndication,  
  conferenceUnlockIndication       ConferenceUnlockIndication,  
  conferenceTerminateIndication    ConferenceTerminateIndication,  
  conferenceEjectUserIndication    ConferenceEjectUserIndication,  
  conferenceTransferIndication     ConferenceTransferIndication,  
  rosterUpdateIndication           RosterUpdateIndication,  
  applicationInvokeIndication      ApplicationInvokeIndication,  
  registryMonitorEntryIndication   RegistryMonitorEntryIndication,  
  conductorAssignIndication        ConductorAssignIndication,  
  conductorReleaseIndication       ConductorReleaseIndication,  
  conductorPermissionAskIndication ConductorPermissionAskIndication,  
  conductorPermissionGrantIndication ConductorPermissionGrantIndication,  
  conferenceTimeRemainingIndication ConferenceTimeRemainingIndication,  
  conferenceTimeInquireIndication  ConferenceTimeInquireIndication,  
  conferenceTimeExtendIndication   ConferenceTimeExtendIndication,  
  conferenceAssistanceIndication   ConferenceAssistanceIndication,  
  textMessageIndication            TextMessageIndication,  
  nonStandardIndication            NonStandardPDU,  
  ...  
}
```

END

9. 多地点通信サービスの利用

全ての GCC 通信は、TTC 標準 JT-T122 において詳述される多地点通信サービス(MCS)を通じて行なわれる。このセクションでは、GCC がどのようにして、MCS サービス、チャンネル割り当て、トークン割り当て、データプライオリティを利用するかを詳細に述べる。

9.1. MCS サービス

GCC は、表 9-1/JT-T124 において示された MCS サービスを仮定する。“M” がマークされた全てのプリミティブとパラメータは、GCC の必須条件である。単に、“O” がマークされた項目は、GCC のオプション条件である。

表 9-1/JT-T124 – GCC によって使用される MCS サービス

(ITU-T T.124)

プリミティブ	利用	パラメータ	利用
MCS-CONNECT-PROVIDER-要求	M	発側アドレス(Calling Address)	O
MCS-CONNECT-PROVIDER-指示	M	発側ドメインセクタ(Calling Domain Selector)	M
MCS-CONNECT-PROVIDER-応答	M	着側アドレス(Called Address)	O
MCS-CONNECT-PROVIDER-確認	M	着側ドメイン セクタ(Called Domain Selector)	-
		Upward/Downward フラグ	M
		ドメイン パラメータ(Domain ParametersDomain Parameters)	M
		サービス品質(Quality of Service)	M
		結果(Result)	M
		ユーザデータ(User Data)	M
MCS-DISCONNECT-PROVIDER-要求	M	理由(Reason)	M
MCS-DISCONNECT-PROVIDER-指示	M		
MCS-ATTACH-USER-要求	M	ドメイン セクタ(Domain Selector)	M
MCS-ATTACH-USER-確認	M	結果(Result)	M
		ユーザ ID(User ID)	M
MCS-DETACH-USER-要求	M	理由(Reason)	M
MCS-DETACH-USER-指示	M	ユーザ ID(User ID)	M
MCS-CHANNEL-JOIN-要求	M	加入するチャンネル(Channel to Join)	M
MCS-CHANNEL-JOIN-確認	M	結果(Result)	M
MCS-CHANNEL-LEAVE-要求	-	離脱するチャンネル(Channel to Leave)	-
MCS-CHANNEL-LEAVE-指示	-	理由(Reason)	-
MCS-CHANNEL-CONVENE-要求	-	結果(Result)	-
MCS-CHANNEL-CONVENE-確認	-	チャンネル(Channel)	-
MCS-CHANNEL-DISBAND-要求	-	チャンネル(Channel)	-
MCS-CHANNEL-DISBAND-指示	-	理由(Reason)	-

MCS-CHANNEL-ADMIT-要求	-	チャンネル(Channel)	-
MCS-CHANNEL-ADMIT-指示	-	マネージャユーザ ID(Manager User ID) ユーザ ID リスト(List of User IDs)	-
MCS-CHANNEL-EXPEL-要求	-	チャンネル(Channel)	-
MCS-CHANNEL-EXPEL-指示	-	ユーザ ID リスト(List of User IDs) 理由(Reason)	-
MCS-SEND-DATA-要求	M	プライオリティ (Priority)	M
MCS-SEND-DATA-指示	M	チャンネル ID(Channel ID) 送信ユーザ ID(Sender User ID) データ(Data)	M M M
MCS-UNIFORM-SEND-DATA-要求	M	プライオリティ (Priority)	M
MCS-UNIFORM-SEND-DATA-指示	M	チャンネル ID(Channel ID) 送信ユーザ ID(Sender User ID) データ(Data)	M M M
MCS-TOKEN-GRAB-要求	O	トークン ID(Token ID)	O
MCS-TOKEN-GRAB-確認	O	結果(Result)	O
MCS-TOKEN-INHIBIT-要求	O	トークン ID(Token ID)	O
MCS-TOKEN-INHIBIT-確認	O	結果(Result)	O
MCS-TOKEN-GIVE-要求	O	譲渡ユーザ ID(User ID Giving)	M
MCS-TOKEN-GIVE-指示	M	受取ユーザ ID(User ID to Receive)	O
MCS-TOKEN-GIVE-応答	M	トークン ID(Token ID)	M
MCS-TOKEN-GIVE-確認	O	結果(Result)	M
MCS-TOKEN-PLEASE-要求	O	要求者のユーザ ID(User ID Requesting)	O
MCS-TOKEN-PLEASE-指示	O	トークン ID(Token ID)	O
MCS-TOKEN-RELEASE-要求	O	トークン ID(Token ID)	O
MCS-TOKEN-RELEASE-確認	O	結果(Result)	O
MCS-TOKEN-TEST-要求	O	トークン ID(Token ID)	O
MCS-TOKEN-TEST-確認	O	トークン状態(Token Status)	O

9.2. チャンネル割り当て

GCC は、2 つの静的なチャンネルを、独占的に使用するために予約しておく。一方のチャンネル、GCC-CHANNEL-0 は、会議における全ての GCC プロバイダによって加入される。もう一方のチャンネル、GCC-CHANNEL-1 は、単に会議召集者によって加入される。さらに、各々の GCC プロバイダは、MCS によって割り当てられた NODE-ID-CHANNEL に加入する。表 9-2/JT-T124 は、GCC によるチャンネルの使用法を示す。

表 9-2/JT-T124 – GCC チャンネル使用方法
(ITU-T T.124)

チャンネルID	型	ニーモニック	記述
GCC-CHANNEL-0	静的(Static)	GCC-BROADCAST-CHANNEL	全ての GCC プロバイダに対する会議における、あらゆる GCC プロバイダからの通信のため。
GCC-CHANNEL-1	静的(Static)	GCC-CONVENER-CHANNEL	会議召集者に対する会議における、あらゆる GCC プロバイダからの通信のため。
-	動的(Dynamic)	NODE ID CHANNEL	会議中のあらゆる GCC プロバイダから特別のノードの GCC プロバイダへ、の通信のため。各々のノードは、そのノード ID によって見分けられる。

9.3. トークン割り当て

GCC は、一つのトークンを、独占的に使用するために予約しておく。このトークン、GCC-TOKEN-0 は、会議議長トークンとして使用される。さらに、GCC は、GCC がアプリケーション プロトコル エンティティによって使用するために割り当てる、動的なトークンとして使用するために、トークン 16384 から 65535 までを予約しておく。表 9-3/JT-T124 は、GCC によるトークンの使用法を示す。

表 9-3/JT-T124 – GCC トークン使用方法

(ITU-T T.124)

トークン ID	ニーモニック	記述
GCC-TOKEN-0	会議議長トークン (CONFERENCE-CONDUCTORSHIP-TOKEN)	会議議長になるために GCC プロバイダによって掴まれる。
16384 から 65535	動的トークン(Dynamic Tokens)	GCC-Registry-Assign-Token プリミティブを使用しているアプリケーション プロトコル エンティティによって使用されるために、GCC によって配置される。

9.4. MCS データ伝送サービスの使用

表 9-4/JT-T124 は、各 GCCPDU の MCS データ伝送サービス(MCS-SEND-DATA と MCS-UNIFORM-SEND-DATA)の用途を要約する。各 GCCPDU の表は、使用された MCS データサービスのタイプ、チャンネル ID、要求されたデータプライオリティである。

表 9-4/JT-T124 – GCCPDU のための MCS 送信データの使用

(ITU-T T.124)

GCCPDU	チャンネル	送信データタイプ	プライオリティ
UserIDIndication	直接接続されたノードの NODE-ID-	MCS-SEND-	Top

	CHANNEL (NODE ID CHANNEL of directly connected node)	DATA	
ConferenceCreateRequest	-	-	-
ConferenceCreateResponse	-	-	-
ConferenceQueryRequest	-	-	-
ConferenceQueryResponse	-	-	-
ConferenceJoinRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND- DATA	Top
ConferenceJoinResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND- DATA	Top
ConferenceInviteRequest	-	-	-
ConferenceInviteResponse	-	-	-
ConferenceAddRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND- DATA	High
	追加実行 MCU の NODE-ID-CHANNEL (Node ID Channel of adding MCU)	MCS-SEND- DATA	High
ConferenceAddResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND- DATA	High
ConferenceLockRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND- DATA	High
ConferenceLockResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND- DATA	High
ConferenceLockIndication	GCC-BROADCAST-CHANNEL	MCS-SEND- DATA	High
	NODE ID CHANNEL	MCS-SEND- DATA	High
ConferenceUnlockRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND- DATA	High
ConferenceUnlockResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND- DATA	High
ConferenceUnlockIndication	GCC-BROADCAST-CHANNEL	MCS-SEND- DATA	High
	NODE ID CHANNEL	MCS-SEND- DATA	High
ConferenceTerminateRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND- DATA	High
ConferenceTerminateRespon	要求者の NODE-ID-CHANNEL	MCS-SEND-	High

e	(Node ID Channel of requester)	DATA	
ConferenceTerminateIndication	GCC-BROADCAST-CHANNEL	MCS-SEND-DATA	High
ConferenceEjectUserRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND-DATA	Top
ConferenceEjectUserResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND-DATA	High
ConferenceEjectUserIndication	GCC-BROADCAST-CHANNEL	MCS-SEND-DATA	Top
ConferenceTransferRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND-DATA	High
ConferenceTransferResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND-DATA	High
ConferenceTransferIndication	GCC-BROADCAST-CHANNEL	MCS-SEND-DATA	High
RosterUpdateIndication	受領者の NODE-ID-CHANNEL (Node ID Channel of recipient)	MCS-SEND-DATA	High
	GCC-BROADCAST-CHANNEL	MCS-SEND-DATA	High
ApplicationInvokeIndication	GCC-BROADCAST-CHANNEL	MCS-SEND-DATA	High
	受領者の NODE-ID-CHANNEL (Node ID Channel of recipient)	MCS-SEND-DATA	High
RegistryRegisterChannelRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND-DATA	High
RegistryAssignTokenRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND-DATA	High
RegistrySetParameterRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND-DATA	High
RegistryRetrieveEntryRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND-DATA	High
RegistryDeleteEntryRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND-DATA	High
RegistryMonitorEntryRequest	トップ GCC の NODE-ID-CHANNEL (Node ID Channel of Top GCC)	MCS-SEND-DATA	High
RegistryMonitorEntryIndication	GCC-BROADCAST-CHANNEL	MCS-SEND-DATA	High
RegistryAllocateHandleRequest	トップ GCC の NODE-ID-CHANNEL	MCS-SEND-	High

st	(Node ID Channel of Top GCC)	DATA	
RegistryAllocateHandleResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND-DATA	High
RegistryResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND-DATA	High
ConductorAssignIndication	GCC-BROADCAST-CHANNEL	MCS-UNIFORM-SEND-DATA	Top
ConductorReleaseIndication	GCC-BROADCAST-CHANNEL	MCS-UNIFORM-SEND-DATA	Top
ConductorPermissionAskIndication	GCC-BROADCAST-CHANNEL	MCS-UNIFORM-SEND-DATA	High
ConductorPermissionGrantIndication	GCC-BROADCAST-CHANNEL	MCS-UNIFORM-SEND-DATA	Top
ConferenceTimeRemainingIndication	GCC-BROADCAST-CHANNEL	MCS-UNIFORM-SEND-DATA	High
ConferenceTimeInquireIndication	GCC-CONVENER-CHANNEL	MCS-SEND-DATA	High
ConferenceTimeExtendIndication	GCC-CONVENER-CHANNEL	MCS-SEND-DATA	High
ConferenceAssistanceIndication	GCC-BROADCAST-CHANNEL	MCS-SEND-DATA or MCS-UNIFORM-SEND-DATA	High
TextMessageIndication	GCC-BROADCAST-CHANNEL	MCS-SEND-DATA or MCS-UNIFORM-SEND-DATA	High
	受領者の NODE-ID-CHANNEL (Node ID Channel of recipient)	MCS-SEND-DATA	High
FunctionNotSupportedResponse	要求者の NODE-ID-CHANNEL (Node ID Channel of requester)	MCS-SEND-DATA	要求と同じ (Same as request)

NonStandardPDU	未定義(Not defined)	未定義 (Not defined)	未定義 (Not defined)
----------------	------------------	----------------------	----------------------

9.5. MCS プリミティブ中の PDU のコード化

この標準において定義された全ての PDU は、あるデータフィールド内で、符号化され配置される。これらは、MCS-CONNECT-PROVIDER か、MCS-SEND-DATA か、または MCS-UNIFORM-SEND-DATA のいずれかである。これらの場合のいずれにおいても、ASN.1 符号化の結果のビット列が、各オクテット、MSB 位置に配置される頭のビット、LSB 位置に配置される続くビットの順番に、MCS によって使用されるオクテット スtringの中に配置される。

MCS-Connect-Provider の場合は、PDU それ自身は、MCS データパラメータで直接配置されるのではなく、9.6 節において定義された、囲み構造の中に含まれている。この場合、記述は、PDU それ自身よりむしろ、外側の構造に適用する。

9.6. MCS-CONNECT-PROVIDER のユーザデータパラメータのフォーマット

TTC 標準 JT-T124 で使用されている、MCS-CONNECT-PROVIDER と MCS-DISCONNECT-PROVIDER プリミティブのユーザデータパラメータは、(TTC 標準 JT-T122 において定義した)MCS コントローラをユニークに識別する形式である。MCS コントローラは、(TTC 標準 JT-T124 を支持するような)標準の型であってもよい、あるいは非標準の型であってもよい。これらを区別するために、ユーザデータは、PDU がたどった MCS コントローラの型の識別子を含む ASN.1 抽象型から構成される。識別子フィールドは、オブジェクト識別子かまたは、H221NonStandardIdentifier (TTC 標準 JT-H221 付属資料 A NS-cap,NS-comm 参照)のどちらかを選択する型キーである (ASN.1 参照)。構成要素と同様に、全体の抽象型は、8.7 節において定義される。全体構造は、ITU-T 勧告 X.691 において定義されるパッキング符号化規則(Packed Encoding Rules)(Aligned variant)を使用して、符号化される。MCS コントローラがこの標準(TTC 標準 JT-T124)を支持する時、このヘッダーは値(TTC 標準 JT-T124 バージョン(0)を示す値 1)を伴って、オブジェクト識別子 CHOICE にセットされることになっている。

この構造の中で、PDU は直接符号化されるのではなく、むしろ、オクテット スtring型のデータフィールドに含まれる。さらに、この標準で定義された全ての他の PDU と共に、パッキング符号化規則(Packed Encoding Rules)(Aligned variant)を使用して、PDU は別々に符号化される。結果として生じる符号化されたビット列は、各オクテット、MSB 位置に配置される頭のビット、LSB 位置に配置される続くビットの順番に、オクテット スtringの中に配置される。(パッキング符号化規則(Packed Encoding Rules)を使用して、常に符号化される)MCS コントローラ型をユニークに符号化され識別している間、異なる PDU 符号を使用する非標準 MCS コントローラや、将来の勧告を考慮に入れるために、PDU と囲み構造は個々に符号化される。

9.7. MCS ドメインセクタの割り込み

MCS-CONNECT-PROVIDER プリミティブの使用において、TTC 標準 JT-T124 は、ドメイン セクタの使用において次の仮定をする。

発側ドメイン セクタは、(MCS アタッチメントを作成する上で、アプリケーションプロトコルによって、更に使用される)会議 ID と同等のローカルドメインセクタを、常に正確に反映する。

着側ドメイン セクタは、TTC 標準 JT-T124 によって決して使用されない。その代わりに、MCS プロバイダは、それらの内容に関わらずコントロール MCSAP へ MCS-CONNECT-PROVIDER を渡す、と推測される。接続が確立される必要があるかどうか、接続がドメインと関係するかどうかを決定することは、GCC プロバイダの役割である。これ

は、MCS-CONNECT-PROVIDER-要求を生成する4つのTTC標準JT-T124プリミティブで、わずかに異なっている。

GCC-CONFERENCE-CREATE: この場合、接続をアタッチできたMCS-CONNECT-PROVIDER-指示の受領より前に、ドメインは存在しない。有効なMCS-CONNECT-PROVIDER-指示を受け取った上で、GCCプロバイダは、新しいMCSドメインを確立しなければならない。また、そのドメインが新しい接続と関連することをMCSへ示さなければならない。その時、ドメインセクタは、ローカルの会議IDとして、GCCプロバイダによって選択される。これらの行為の両方を行なうための手段は、TTC標準JT-T122では明示されていないローカルな事柄である。

GCC-CONFERENCE-QUERY: この場合、MCS-CONNECT-PROVIDER-応答が、要求がユーザ拒否されたことを指示するので、いかなる接続も確立されない。結果として、いかなるMCSドメインにも要求を関連させる必要はない。

GCC-CONFERENCE-JOIN: この場合、ドメインは、受け取りノードで既に存在する。加入要求がGCCプロバイダによって受け入れられるならば、GCCプロバイダは、接続が関連する領域のMCSプロバイダへ指示しなければならない。会議名(及び会議名修飾子)は、どのドメインを指示するかを決定するGCCプロバイダによって使用される。ドメインセクタは、その会議のローカルの会議IDである。MCSプロバイダへのドメインを示すための手段は、TTC標準JT-T122では明示されていないローカルな事柄である。

GCC-CONFERENCE-INVITE: この場合、受け取りノードは、Createの場合のように、ドメインを確立されてはいない。GCCプロバイダは、Inviteが受け入れられるならば、ドメインセクタとして使用されることで、会議IDを選択する。MCSプロバイダは、このノードで、このドメインの確立と特定の接続が、このドメインに関連することを示さなければならない。Createの場合におけるように、これらの行為の両方を行なうための手段は、TTC標準JT-T122では明示されていないローカルな事柄である。

付属資料 A

静的チャンネルとトークン ID 割り当て

(この付属資料は、この標準の不可欠な部分である)

A.1. 静的チャンネル ID 割り当て

表 A-1/JT-T124 は、この標準で使用するために割り当てられた静的なチャンネルのための、静的なチャンネル ID 数の割り当てのリストである。静的なチャンネル ID 数の割り当ては、ITU-T 勧告 T.120 に集中することが考えられる。しかし、ITU-T 勧告 T.120 が完成するまで、ここに含まれる。

表 A-1 /JT-T124- 静的チャンネル ID 割り当て

(ITU-T T.124)

シンボル名	チャンネル ID
GCC-CHANNEL-0	1
GCC-CHANNEL-1	2

A.2. 静的トークン ID 割り当て

表 A-2/JT-T124 は、この標準によって使用するために割り当てられた静的なトークンのための、静的なトークン ID の数に関する割り当てのリストである。静的なトークン ID の数に関する割り当ては、ITU-T 勧告 T.120 に集中することが考えられる、しかし、ITU-T 勧告 T.120 が完成するまで、ここに含まれる。

表 A-2/JT-T124 – 静的トークン ID 割り当て

(ITU-T T.124)

シンボル名	トークン ID
GCC-TOKEN-0	1

付属資料 B

オブジェクト識別子割り当て

(この付属資料は、この標準の不可欠な部分である)

表 B-1/JT-T124 は、この標準で使用するために定義されたオブジェクト識別子の割り当てのリストである。

表 B-1/JT-T124

(ITU-T T.124)

対象 ID 値	記述
(TTC 標準 JT-T124 バージョン(0)1)	このオブジェクト識別子は、MCS コントローラとして使用中のこの標準のバージョンを指示するために使用される。この時、定義された単一の標準化されたバージョンが存在する。

執筆作成協力者 (1996年10月31日時点)

第四部門委員会

部門委員長	齋藤 幸男	日本電信電話 (株)
副部門委員長	長尾 朋	国際電信電話 (株)
副部門委員長	渡辺 芳明	日本アイ・ビー・エム (株)
委員	須川 毅	住友電気工業 (株)
〃	堀 潔洋	(株) 東芝
〃	伊藤 政司	(株) リコー
〃	菅野 昌志	松下電器産業 (株)
〃	羽柴 善安	東京電力 (株)
〃	青島 一倫	(第1 専門委員会 専門委員長)
〃	伊藤 均	(第1 専門委員会 副専門委員長)
〃	小池 淳	(第2 専門委員会 専門委員長)
〃	藤野 博文	(第2 専門委員会 副専門委員長)
〃	大橋 康	(第3 専門委員会 専門委員長)
〃	青山 敬	(第3 専門委員会 副専門委員長)
〃	松本 充司	(第4 専門委員会 専門委員長)
〃	近藤 貴士	(第4 専門委員会 副専門委員長)
〃	千田 昇一	(オブジェクト特別専門委員会 専門委員長)
〃	岩本 裕司	(オブジェクト特別専門委員会 副専門委員長)

第四部門委員会 第2 専門委員会

専門委員長	小池 淳	国際電信電話 (株)
副専門委員長	藤野 博文	沖電気工業 (株)
委員	松村 浩知	国際電信電話 (株)
〃	亀井 喜宏	日本電信電話 (株)
〃	松原 俊幸	シャープ (株)
〃	磯山 芳一	住友電気工業 (株)
〃	池田 強	(株) 田村電機製作所
〃	井澤 庄次	(株) テック
〃	笹野 潤	(株) 東芝
〃	佐藤 貴	日本電気 (株)
〃	羽鳥 瑛	日本無線 (株)
〃	岩田 吉隆	(株) 日立製作所
〃	前井 佳博	富士ゼロックス (株)
〃	鈴木 卓	松下電送 (株)
〃	石井 薫	三菱電機 (株)
〃	望月 薫雅	(株) 明電舎
〃	柴田 博	(株) リコー
〃	畑下 眞廣	村田機械 (株)
〃	吉沢 健一	三菱電機 (株)

第四部門委員会 第2 専門委員会 SWG 6

リーダー	吉沢 健一	三菱電機 (株)
サブリーダー	前野 和俊	日本電気 (株)
委員	山口 博幸	日本電信電話 (株)
〃	宝生 真行	沖電気工業 (株)
〃	上野 裕治	シャープ (株)
〃	若松 久仁男	(株) 東芝
〃	八田 智明	日本アイ・ビー・エム (株)
〃	清水 雅子	日本電気 (株)
〃	右馬 伸一	(株) 日立製作所
〃	松田 幸成	三菱電機 (株)
〃	望月 薫雅	(株) 明電舎
〃	藤岡 進	(株) リコー
オブザーバ	藤本 宏明	日本電信電話 (株)
〃	池上 貴則	(株) 東芝

事務局 近藤 康生 (第4 技術部)