

TTC技術レポート
Technical Report

TR-18057

音声ブラウザ環境における CSTA XML の 使用法に関する技術レポート

〔 Technical Report on Using CSTA XML in a Voice Browser
Environment 〕

第1版

2005年2月3日制定

社団法人
情報通信技術委員会

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE



本書は、（社）情報通信技術委員会が著作権を保有しています。

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

目次

< 参考 >	1
1 序文	2
2 技術レポートの概要	2
3 技術レポートの内容	2

付録（和文解説書）目次

1 スコープ (Scope)	1
2 参考文献 (Normative references)	1
3 ECMA-323 の要約 (Brief Overview of ECMA-323)	2
4 基本概念 (Fundamental Concepts)	3
4.1 CSTA コネクション(CSTA Connection)	3
4.2 CSTAコネクション状態モデル (CSTA Connection State Model)	3
4.3 CSTA呼のコネクション 状態遷移 (Connection State Transitions for CSTA Calls)	4
4.3.1 着呼 (Incoming Call)	4
4.3.2 発呼 (Outgoing Call)	4
5 CSTAプロファイル (CSTA Profiles)	5
5.1 Level 1a Voice Browserプロファイル (Level 1a Voice Browser Profile)	6
5.1.1 サービス (Services)	6
5.1.2 イベント (Events)	6
5.2 Level 1b Voice Browserプロファイル (Level 1b Voice Browser Profile)	6
5.2.1 サービス (Services)	6
5.2.2 イベント (Events)	7
5.3 Level 2 Voice Browserプロファイル (Level 2 Voice Browser Profile)	7
5.3.1 サービス (Services)	7
5.3.2 イベント (Events)	8
5.4 基本Telephonyプロファイル (Basic Telephony Profile)	8
5.4.1 サービス (Services)	8
5.4.2 イベント (Events)	9
5.5 他の機能 (Other Features)	9
6 ECMA-323 実例 (ECMA-323 Illustrative Examples)	10
6.1 テレフォニ・プラットフォームの能力把握 (Discovering the Capabilities of a Telephony Platform) ...	11
6.2 状態監視の開始(着呼の認識) (Starting a Monitor (i.e. listening for incoming calls))	11
6.2.1 状態監視開始要求 - サービス要求例 (Monitor Start – Service Request example)	11
6.2.2 状態監視開始要求 - サービス応答例 (Monitor Start – Service Response example)	11
6.3 着呼通知 (Notification of an Inbound Call)	12
6.3.1 Deliveredイベント(Delivered Event example)	12

6.4 着呼応答 (Answering an Inbound Call)	13
6.4.1 応答要求 - サービス要求例 (Answer Call – Service Request example)	13
6.4.2 応答要求 - サービス応答例 (Answer Call – Service Response example)	14
6.5 接続された呼の通知 (Notification of a Connected Call)	14
6.5.1 Establishedイベントの例 (Established Event example)	14
6.6 コネクションの切断 (Clearing a Connection)	15
6.6.1 Clear Connection - サービス要求例 (Clear Connection– Service Request example)	15
6.6.2 切断要求 - サービス応答例 (Clear Connection – Service Response example)	15
6.7 Clearedコネクションの通知 (Notification of a Cleared Connection)	15
6.7.1 Connection Clearedイベント例 (Connection Cleared Event example)	15
6.8 発呼の開始 (Initiating an Outbound Call)	16
6.8.1 Make Call–サービス要求例 (プロファイルを参照) (Make Call – Service Request example (refer to Profile)).....	16
6.8.2 Make Call–サービス応答例 (Make Call – Service Response example)	16
6.9 発呼イベント・シーケンス (Outbound Call Event Sequence)	17
6.9.1 Originatedイベント例 (Originated Event Example)	17
6.9.2 Network Reachedイベント (Network Reached Event)	17
6.9.3 Deliveredイベント (Delivered Event).....	19
6.9.4 Establishedイベント (Established Event)	20
6.10 シングル・ステップ転送 (Single Step Transfer).....	20
6.10.1 シングル・ステップ転送 – サービス要求例 (Single Step Transfer – Service Request example)....	20
6.10.2 シングル・ステップ転送 – サービス応答例 (Single Step Transfer - Service Response example)	20
6.11 Transferred Connectionの通知 (Notification of a Transferred Connection).....	21
6.11.1 Transferredイベント例 (Transferred Event example).....	21
6.12 着信転送 (Deflect)	22
6.12.1 着信転送 – サービス要求例 (Deflect – Service Request example).....	22
6.12.2 着信転送 – サービス応答例 (Deflect – Service Response example).....	22
6.13 着信転送Connectionの通知 (Notification of a Diverted Connection).....	22
6.13.1 Divertedイベントの例 (Diverted Event example).....	22
6.14 シングル・ステップ会議(Single Step Conference).....	23
6.14.1 Single Step Conference – サービス要求 (Single Step Conference – Service Request example).....	23
6.14.2 シングル・ステップ会議 – サービス応答例 (Single Step Conference – Service Response example)	24
6.15 呼に追加されたパーティの通知 (Notification of an Party Added to a call)	24
6.15.1 Conferencedイベント例.....	24
6.16 否定応答例 (Failure Response example)	25
7 SALT/CSTA XMLプログラム例 (SALT/CSTA XML Programming Example)	27
8 CCXML/CSTA XMLプログラム例 (CCXML/CSTA XML Programming Example)	31
9 CSTA呼制御機能 (CSTA Call Control Features)	33

9.1 サービス (Services)	33
9.2 イベント (Events)	34

< 参考 >

1 . 国際勧告等の関連

本技術レポートは、ISO/IEC において制定された「Information technology - Telecommunications and information exchange between systems – Using ECMA-323 (CSTA XML) in a Voice Browser Environment (ISO/IEC TR 18057:2004)」に準拠している。

2 . 上記国際勧告等に対する追加項目等

なし

3 . 改版の履歴

版数	制定日	改版内容
第 1 版	2005 年 2 月 3 日	制定

4 . その他

(1) 引用している勧告、標準等

ISO/IEC TR 18057:2004 (ECMA TR-85:2002)

(2) 参照している勧告、標準等

- ECMA-269:2002 (ISO/IEC 18051:2003)
- ECMA-323:2002 (ISO/IEC 18056:2003)
- SALT Speech language Application Language Tags 1.0 Specification (SALT), SALT Forum, 15 July 2002, (<http://www.saltforum.org>)
- CCXML Voice Browser Call Control: CCXML Version 1.0 - W3C Working Draft, W3C, 11 October 2002, (<http://www.w3c.org/TR/ccxml/>)

(3) 付録（和文解説書）

本技術レポートには、引用している「Information technology - Telecommunications and information exchange between systems – Using ECMA-323 (CSTA XML) in a Voice Browser Environment (ISO/IEC TR 18057:2004)」の理解を助けるために日本語による付録（和文解説書）が添付されている。

5 . 技術レポート作成部門

企業ネットワーク専門委員会 Computer Telecommunication Interface サブワーキング

1 序文

本技術レポートは、「Information technology - Telecommunications and information exchange between systems – Using ECMA-323 (CSTA XML) in a Voice Browser Environment (ISO/IEC TR 18057:2004)」の技術的内容を変更することなく作成したものであり、原国際規格(ISO/IEC TR 18057:2004)の記載項目の同項目の内容を引用する。

2 技術レポートの概要

本技術レポートは、OSI Layer 7 における Computing network と telecommunications network との通信サービスを標準化した「Information technology - Telecommunications and information exchange between systems – Services for Computer Supported Telecommunications Applications (CSTA) Phase III (ISO/IEC 18051:2000)」の XML Protocol を規定した「Information technology - Telecommunications and information exchange between systems – XML Protocol for Computer Supported Telecommunications Applications (CSTA) Phase III (ISO/IEC 18056:2003)」に準拠して記述されたアプリケーションを音声ブラウザ環境上において使用する方法を規定している。

3 技術レポートの内容

本技術レポートにおいて規定する項目は以下の通りである。これらの項目の内容については、原国際規格 (ISO/IEC TR 18057:2004) の同項目の内容を引用する。

1 Scope

2 Normative references

3 Brief Overview of ECMA-323

4 Fundamental Concepts

4.1 CSTA Connection

4.2 CSTA Connection State Model

4.3 Connection State Transitions for CSTA Calls

4.3.1 Incoming Call

4.3.2 Outgoing Cal

5 CSTA Profiles

5.1 Level 1a Voice Browser Profile

5.1.1 Services

5.1.2 Events

5.2 Level 1b Voice Browser Profile

5.2.1 Services

5.2.2 Events

5.3 Level 2 Voice Browser Profile

5.3.1 Services

- 5.3.2 Events
- 5.4 Basic Telephony Profile
 - 5.4.1 Services
 - 5.4.2 Event
- 5.5 Other Features
- 6 ECMA-323 Illustrative Examples
 - 6.1 Discovering the Capabilities of a Telephony Platform
 - 6.2 Starting a Monitor (i.e. listening for incoming calls)
 - 6.2.1 Monitor Start – Service Request example
 - 6.2.2 Monitor Start – Service Response example
 - 6.3 Notification of an Inbound Call
 - 6.3.1 Delivered Event example
 - 6.4 Answering an Inbound Call
 - 6.4.1 Answer Call – Service Request example
 - 6.4.2 Answer Call – Service Response example
 - 6.5 Notification of a Connected Call
 - 6.5.1 Established Event example
 - 6.6 Clearing a Connection
 - 6.6.1 Clear Connection– Service Request example
 - 6.6.2 Clear Connection – Service Response example
 - 6.7 Notification of a Cleared Connection
 - 6.7.1 Connection Cleared Event example
 - 6.8 Initiating an Outbound Call
 - 6.8.1 Make Call – Service Request example (refer to Profile)
 - 6.8.2 Make Call – Service Response example
 - 6.9 Outbound Call Event Sequence
 - 6.9.1 Originated Event Example
 - 6.9.2 Network Reached Event
 - 6.9.3 Delivered Event
 - 6.10 Single Step Transfer
 - 6.10.1 Single Step Transfer – Service Request example
 - 6.10.2 Single Step Transfer – Service Response example
 - 6.11 Notification of a Transferred Connection
 - 6.11.1 Transferred Event example
 - 6.12 Deflect
 - 6.12.1 Deflect – Service Request example
 - 6.12.2 Deflect – Service Response example
 - 6.13 Notification of a Diverted Connection
 - 6.13.1 Diverted Event example

6.14 Single Step Conference

6.14.1 Single Step Conference – Service Request example

6.14.2 Single Step Conference – Service Response example

6.15 Notification of an Party Added to a call

6.15.1 Conferenced Event example

6.16 Failure Response example

7 SALT/CSTA XML Programming Example

8 CCXML/CSTA XML Programming Example

9 CSTA Call Control Features

9.1 Services

9.2 Events

TR-18057

音声ブラウザ環境における CSTA XML の 使用法に関する技術レポート

〔 Technical Report on Using CSTA XML in a Voice Browser
Environment 〕

付録(和文解説書)

付録（和文解説書）目次

1	スコープ (Scope)	1
2	参考文献 (Normative references)	1
3	ECMA-323 の要約 (Brief Overview of ECMA-323)	2
4	基本概念 (Fundamental Concepts)	3
4.1	CSTA コネクション(CSTA Connection)	3
4.2	CSTAコネクション状態モデル (CSTA Connection State Model)	3
4.3	CSTA呼のコネクション 状態遷移 (Connection State Transitions for CSTA Calls)	4
4.3.1	着呼 (Incoming Call)	4
4.3.2	発呼 (Outgoing Call)	4
5	CSTAプロファイル (CSTA Profiles)	5
5.1	Level 1a Voice Browserプロファイル (Level 1a Voice Browser Profile)	6
5.1.1	サービス (Services)	6
5.1.2	イベント (Events)	6
5.2	Level 1b Voice Browserプロファイル (Level 1b Voice Browser Profile)	6
5.2.1	サービス (Services)	6
5.2.2	イベント (Events)	7
5.3	Level 2 Voice Browserプロファイル (Level 2 Voice Browser Profile)	7
5.3.1	サービス (Services)	7
5.3.2	イベント (Events)	8
5.4	基本Telephonyプロファイル (Basic Telephony Profile)	8
5.4.1	サービス (Services)	8
5.4.2	イベント (Events)	9
5.5	他の機能 (Other Features)	9
6	ECMA-323 実例 (ECMA-323 Illustrative Examples)	10
6.1	テレフォニ・プラットフォームの能力把握 (Discovering the Capabilities of a Telephony Platform)	11
6.2	状態監視の開始(着呼の認識) (Starting a Monitor (i.e. listening for incoming calls))	11
6.2.1	状態監視開始要求 - サービス要求例 (Monitor Start – Service Request example)	11
6.2.2	状態監視開始要求 - サービス応答例 (Monitor Start – Service Response example)	11
6.3	着呼通知 (Notification of an Inbound Call)	12
6.3.1	Deliveredイベント(Delivered Event example)	12
6.4	着呼応答 (Answering an Inbound Call)	13
6.4.1	応答要求 - サービス要求例 (Answer Call – Service Request example)	13
6.4.2	応答要求 - サービス応答例 (Answer Call – Service Response example)	14
6.5	接続された呼の通知 (Notification of a Connected Call)	14
6.5.1	Establishedイベントの例 (Established Event example)	14
6.6	コネクションの切断 (Clearing a Connection)	15

6.6.1 Clear Connection - サービス要求例 (Clear Connection– Service Request example)	15
6.6.2 切断要求 - サービス応答例 (Clear Connection – Service Response example)	15
6.7 Clearedコネクションの通知 (Notification of a Cleared Connection)	15
6.7.1 Connection Clearedイベント例 (Connection Cleared Event example)	15
6.8 発呼の開始 (Initiating an Outbound Call)	16
6.8.1 Make Call–サービス要求例 (プロファイルを参照) (Make Call – Service Request example (refer to Profile)).....	16
6.8.2 Make Call–サービス応答例 (Make Call – Service Response example)	16
6.9 発呼イベント・シーケンス (Outbound Call Event Sequence)	17
6.9.1 Originatedイベント例 (Originated Event Example)	17
6.9.2 Network Reachedイベント (Network Reached Event)	17
6.9.3 Deliveredイベント (Delivered Event).....	19
6.9.4 Establishedイベント (Established Event)	20
6.10 シングル・ステップ転送 (Single Step Transfer).....	20
6.10.1 シングル・ステップ転送 – サービス要求例 (Single Step Transfer – Service Request example)....	20
6.10.2 シングル・ステップ転送 - サービス応答例 (Single Step Transfer - Service Response example) 20	
6.11 Transferred Connectionの通知 (Notification of a Transferred Connection).....	21
6.11.1 Transferredイベント例 (Transferred Event example).....	21
6.12 着信転送 (Deflect)	22
6.12.1 着信転送 – サービス要求例 (Deflect – Service Request example).....	22
6.12.2 着信転送 - サービス応答例 (Deflect – Service Response example).....	22
6.13 着信転送Connectionの通知 (Notification of a Diverted Connection).....	22
6.13.1 Divertedイベントの例 (Diverted Event example).....	22
6.14 シングル・ステップ会議(Single Step Conference).....	23
6.14.1 Single Step Conference – サービス要求 (Single Step Conference – Service Request example).....	23
6.14.2 シングル・ステップ会議 - サービス応答例 (Single Step Conference – Service Response example)	24
6.15 呼に追加されたパーティの通知 (Notification of an Party Added to a call)	24
6.15.1 Conferencedイベント例.....	24
6.16 否定応答例 (Failure Response example)	25
7 SALT/CSTA XMLプログラム例 (SALT/CSTA XML Programming Example)	27
8 CCXML/CSTA XMLプログラム例 (CCXML/CSTA XML Programming Example)	31
9 CSTA呼制御機能 (CSTA Call Control Features)	33
9.1 サービス (Services)	33
9.2 イベント (Events)	34

1 スコープ (Scope)

Computer Supported Telecommunication Application (CSTA)のためのサービスは、標準 ECMA-269 によって定義され、それらをサービスするための XML プロトコルは、標準 ECMA-323 によって定義されている。

多くの場合、アプリケーションは、CSTA の中で標準化された機能のわずかなサブセットだけを必要としている。音声ブラウザ環境では、通常（呼制御でない）音声処理がアプリケーションの主な目的である。例えば、CSTA の機能的な観点からみるとアプリケーションは単に着信にตอบสนองして、その後、それを切断する、だけが必要かもしれない。これらの音声中心のアプリケーションとして発展するとともに、付加的に、より進歩した、CSTA 準拠のコミュニケーション・プラットフォームによって提供される CSTA 標準化機能を使用することができる。

ECMA-269 および ECMA-323 が比較的大規模な標準(計 1100 ページ)であるので、基本的な CSTA 機能を実装するために理解しなければならない基本概念がどこにあるかを知ることが、CSTA 標準について最低限の知識を持たないアプリケーション開発者にとっては難問である。

この技術レポートは、ECMA-323 が音声ブラウザ環境の中でどのように使用されるかを説明する。この技術レポートで説明された概念は、CSTA 準拠コミュニケーション・プラットフォームからの非同期イベントをサポートする XML に基づいた読み/書きのメッセージ・インタフェース(つまり CSTA サービス境界)を提供する全ての音声ブラウザ環境に適用することができる。SALT smex メカニズムを使用して、呼制御するための ECMA-323 インタフェースを実装した SALT を利用可能なブラウザは、この能力を備えたブラウザのひとつの例である。

この技術レポートの全体にわたり、用語「ECMA-323 を利用可能な音声ブラウザ」は、一般的な意味で CSTA 準拠 ECMA-323 のインタフェースを提供するブラウザ実装に言及するために使用されている。

SALT が利用可能なブラウザおよび CCXML のようないくつかの異なる環境の中で、ECMA-323 をどのように使用することができるかの例を示す。

2 参考文献 (Normative references)

下記の参考文献は、このドキュメントのアプリケーションにとり不可欠なものである。日付のある参考文献は、引用された版のみ有効であり、日付のない参考文献は、いくつかの修正を含んだ参考文献の最終版が有効である。

この技術レポートは、音声ブラウザ環境で ECMA-323 を使用する方法を提供する。次の ECMA 標準は、CSTA に関する参考文献として用いられるべきである。

- ECMA-269:2002, Services for Computer Supported Telecommunications Applications (CSTA) Phase (ISO/IEC 18051:2003)
- ECMA-323:2002, XML Protocol for Computer Supported Telecommunications Applications (CSTA) Phase (ISO/IEC 18056:2003)

ECMA CSTA 標準は様々な環境の中で呼制御のために使用することができる。次の参考文献は、異なる環境の中で ECMA CSTA 標準を使用することについての追加情報を提供する。

- SALT Speech Language Application Language Tags 1.0 Specification (SALT), SALT Forum, 15 July

2002, (<http://www.saltforum.org>)

- CCXML Voice Browser Call Control: CCXML Version 1.0 - W3C Working Draft, W3C, 11 October 2002, (<http://www.w3c.org/TR/ccxml/>)

3 ECMA-323 の要約 (Brief Overview of ECMA-323)

ECMA-323 は、W3C XML スキーマ言語勧告に基づいた、XML スキーマ・セットから成る。この標準は、ECMA-269 で定義されたたくさんのサービス・カテゴリに対するスキーマを含む。

呼制御は、ECMA-323 のサービス・カテゴリの一つにすぎない。他のサービス・カテゴリの例としては次の通りである。

能力交換サービス、呼ルーチング・サービス、デバイスをコントロールするためのサービス（例えば、メッセージ・ウェイティング、ディスプレイへの表示、フォワード設定）、そして他の多くのものがある。

CSTA は、アプリケーションに対してプロトコルから独立した抽象的なレイヤを提供する。それは、基本的な第一者呼制御に基づいたプラットフォーム、より複雑な第三者呼制御(CTI)プラットフォーム、あるいは両者の組み合わせ（いくつか追加された第三者の呼制御機能を備えた第一者呼制御）を使用することができる、統一的で標準に基づいたメッセージ・インタフェースを提供する。

さらに、CSTA モデリングおよび概念は、SALT 呼制御オブジェクト（SALT 仕様書の 3 章）のように多くの手順とオブジェクト・モデルと互換性を持つ。

4 基本概念 (Fundamental Concepts)

この章は、ECMA-323 を利用可能な音声ブラウザが ECMA-323 メッセージをどのように利用することができるかを説明するために有用なモデル化概念の情報を紹介する。実際の ECMA CSTA 標準は最も新しい版の記述を使用すべきである。

4.1 CSTA コネクション(CSTA Connection)

CSTA 呼制御サービスは、CSTA コネクションに適用される。CSTA コネクションは、呼とテレフォニ・エンドポイントの関係を表す。CSTA コネクションは、CSTA コネクション識別子によって参照付けられる。CSTA コネクション識別子は、呼識別子およびデバイス(エンドポイント)識別子から成る。

典型的な第一者呼制御実装では、音声ブラウザ・アプリケーションが、音声ブラウザ・プラットフォームに直接関連した CSTA コネクションだけを操作する。しかしながら、別の呼制御実装は、さらに、CSTA サービス(例えば第三者呼制御による)を利用して、呼に含まれる別のエンドポイントのアプリケーション制御を提供するかもしれない。任意のエンドポイントが音声ブラウザ・アプリケーションによって特定されることを可能にするために、デバイス識別子は CSTA コネクション識別子に含まれている。

4.2 CSTA コネクション状態モデル (CSTA Connection State Model)

音声ブラウザ・アプリケーションを利用可能とする ECMA-323 は、関連するアドレス経由のテレフォニ・エンドポイント上で状態監視をおこなうことにより (ECMA-323 呼制御イベントにより) コネクションの状態遷移が通知される (例えば、これはアプリケーションがどのように着信を「認識する」か、ということである)。

呼における各 CSTA コネクションは、コネクション状態に関係している。CSTA は、次のコネクション状態から成るコネクション状態モデル (ECMA-269、図 6-19 を参照) を指定する。

状態	説明
Alerting	エンドポイントへの着呼を示す。典型的には、そのコネクションは鳴動中かもしれない。あるいは、それが呼出直前(例えば、offered)の状態かもしれない。
Connected	コネクションは、呼と通話中であることを示す。このコネクション状態は、着呼または発呼の結果である。
Failed	呼の経過が立ち往生した事示す。典型的には、発呼を試みて話中のエンドポイントに遭遇したことを表す。
Held	エンドポイントがもはや呼と通話中でないを示す。1つのエンドポイント(つまり回線)毎に複数呼をサポートする実装のため、回線が別の電話をかけている状態(例えば、アナログ回線上のコンサルテーション転送)の間、コネクションは保留される。
Initiated	過渡状態であり、通常は、エンドポイントがサービス(例えば、発信音)を起動していることを示す。
Null	呼とエンドポイントの間に関係がない。
Queued	デバイスで、呼が一時的に待ち行列に入っていることを示す (例えば、呼がパークされた、キャンプオンされた)。

CSTA コネクション状態は ECMA-323 イベントで提供される。

4.3 CSTA 呼のコネクション 状態遷移 (Connection State Transitions for CSTA Calls)

4.3.1 着呼 (Incoming Call)

次の図は、着呼における CSTA のイベントを説明している。音声ブラウザ・プラットフォーム上のエンドポイント (着呼コネクション) のコネクション状態も示す。

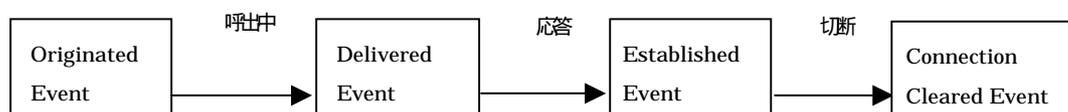
イベント	状態	説明
Delivered Event	Alerting	呼が呼出中であることを示す。「自動応答」される呼は、このイベントを送らない。CSTA Answer Call サービスは呼に応答するために使用される。この結果、Established イベントが生じる。
Established Event	Connected	呼に応答したことを示す。メディア・パスが確立された。CSTA Clear Connection サービスは、呼を切断するために使用される。Connection Cleared イベントは、Clear Connection サービスの結果として生成される。
Connection Cleared Event	Null	コネクションが切断されたことを示す。これは Clear Connection サービスの結果、もしくは呼から任意のパーティが切断された結果としてありえる。



4.3.2 発呼 (Outgoing Call)

次の図は、発呼のための CSTA イベントを説明する。音声ブラウザ・プラットフォーム上のエンドポイント (発呼コネクション) のコネクション状態も示す。このシーケンスは CSTA Make Call サービスの結果である。

イベント	状態	説明
Originated Event	Connected	発側コネクション(音声ブラウザ・プラットフォーム上のエンドポイント)が接続されたことを示す。
Delivered Event	Connected	呼が着呼パーティを呼出中であることを示す。
Established Event	Connected	着呼パーティが呼に応答したことを示す。メディア・パスが確立された。
Connection Cleared Event	Null	コネクションが切断されたことを示す。



5 CSTA プロファイル (CSTA Profiles)

多くの CSTA 機能がオプションであり、また異なる CSTA 実装間のアプリケーション移植性を高めるために、CSTA 標準は、適合基準として機能の最小限のサブセットを要求する。

ECMA-269 は、プロファイルの組を定義する。少なくとも 1 つのプロファイルをサポートされることが必要である。次のプロファイルは、音声ブラウザ・アプリケーションによって必要とされる呼制御のサービスとイベントを綿密に適合させている。

プロファイル名	説明
Level 1a Voice Browser Profile (ECMA-269 第 5 版で追加)	以下の機能を提供する。 <ul style="list-style-type: none"> ・ 着呼に応答する ・ 切断する ・ Single Step Transfer Call サービスを使用して呼を別のエンドポイントへ転送する Get Switching Function Capabilities サービスの機能は、このプロファイルで必要としていない。
Level 1b Voice Browser Profile (ECMA-269 第 5 版で追加)	以下の機能を提供する。 <ul style="list-style-type: none"> ・ 着呼に応答する ・ 切断する ・ Deflect Call サービスを使用して呼を別のエンドポイントに転送する Get Switching Function Capabilities サービスの機能は、このプロファイルで必要としていない。
Level 2 Voice Browser Profile (ECMA-269 第 5 版で追加)	Level 1a Voice Browser Profile、Level 1b Voice Browser Profile のどちらかで要求されるサービスとイベントに加えて、発呼する機能を提供する。 Get Switching Function Capabilities サービスの機能は、このプロファイルで必要とする。
Basic Telephony Profile	以下の機能を提供する。 <ul style="list-style-type: none"> ・ 着呼に応答する ・ 発呼する ・ 切断する Get Switching Function Capabilities サービスの機能は、このプロファイルで必要とする。

注

ネットワーク/デバイス信号が内在するのを見せないネットワーク及び/またはエンドポイントで接続されるテレフォニ・プラットフォームは、それらの全ての CSTA イベントを提供することを期待されていない。例えば、もしテレフォニ・ネットワークが話中の表示を提供しなければ、Failed イベントは要求されない。

5.1 Level 1a Voice Browser プロファイル (Level 1a Voice Browser Profile)

5.1.1 サービス (Services)

次の CSTA サービスは、Level 1a Voice Browser プロファイルに含まれる。

サービス	説明
Answer call	呼出中の呼に応答する。音声ブラウザで、応答デバイスがプラットフォーム上のエンドポイントである。
Clear Connection	コネクションを切断する。音声ブラウザ環境で、切断デバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Single Step Transfer (of a connected call)	別のエンドポイントに呼を転送する。音声ブラウザ環境では、転送デバイスが音声ブラウザ・プラットフォーム上のエンドポイントであり、Single Step Transfer サービスを完了した後、その呼とは関連しない。
Monitor Start	エンドポイント上でデバイス・タイプ状態監視を確立する。音声ブラウザ環境では、状態監視されたデバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Monitor Stop	既存の状態監視を終了する。

5.1.2 イベント (Events)

次の CSTA イベントは、Level 1a Voice Browser プロファイルに含まれる。

イベント	説明
Connection Cleared	エンドポイントが呼から切断されたことを示す。
Delivered	呼がエンドポイントを呼出中であることを示す。
Established	エンドポイントが応答もしくは呼と接続されたことを示す。
Failed	呼が完了されない(例えば、話中のエンドポイントと遭遇した場合)ことを示す。
Transferred	既存の呼がエンドポイント(音声ブラウザ・プラットフォーム上で)から別のエンドポイントに転送され、その呼から切断されたことを示す。これは、転送デバイス・コネクション状態「Null」であることを示し、Connection Cleared イベントが Transferred イベント後に転送デバイスから生成されない。

5.2 Level 1b Voice Browser プロファイル (Level 1b Voice Browser Profile)

5.2.1 サービス (Services)

次の CSTA サービスが Level 1b Voice Browser プロファイルに含まれる。

サービス	説明
Answer call	呼出中の呼に応答する。音声ブラウザで、応答デバイスがプラットフォーム上のエンドポイントである。
Clear Connection	コネクションを切断する。音声ブラウザ環境で、切断デバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Deflect (of a connected call)	着信転送デバイスからコネクションを移動する。音声ブラウザ環境では、着信転送デバイスが音声ブラウザ・プラットフォーム上のエンドポイントであり、Deflect Call サービスが完了した後、その呼とは関連しない。
Monitor Start	エンドポイント上でデバイス・タイプ状態監視を確立する。音声ブラウザ環境では、状態監視されたデバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Monitor Stop	既存の状態監視を終了する。

5.2.2 イベント (Events)

次のCSTA イベントが Level 1b Voice Browser プロファイルに含まれる。

イベント	説明
Connection Cleared	エンドポイントが呼から切断されたことを示す。
Delivered	呼がエンドポイントを呼出中であることを示す。
Diverted	エンドポイント(音声ブラウザ・プラットフォーム上で)が呼を別のエンドポイントにリダイレクトし、そして呼とは関連しなくなったことを示す。
Established	エンドポイントが応答もしくは呼と接続されたことを示す。
Failed	呼が完了されない(例えば、話中のエンドポイントと遭遇した場合)ことを示す。

5.3 Level 2 Voice Browser プロファイル (Level 2 Voice Browser Profile)

5.3.1 サービス (Services)

このプロファイルでは、アプリケーションが CSTA 実装を ECMA-323 Get Switching Function Capabilities サービスにより認識するために、その能力を示すよう要求される。能力は、テレフォニ・プラットフォームによってサポートされる ECMA-323 サービスとイベントのリストと、実装によってサポートされる多くのタイプの動作オプションを含む(プロファイルが提供され、外線発信のためにダイヤル文字列で許可されたディジット・タイプなどである)。多くのパラメータはオプションであり、ECMA-323 メッセージで提供される必要がない。

次のCSTA サービスが Level 2 Voice Browser プロファイルに含まれる。

サービス	説明
Answer Call	呼出中の呼に応答する。音声ブラウザで、応答デバイスがプラットフォーム上のエンドポイントである。
Clear Connection	コネクションを切断する。音声ブラウザ環境で、切断デバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Make Call	2つのデバイス間の呼を確立する。音声ブラウザ環境で、発呼デバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Monitor Start	エンドポイント上でデバイス・タイプ状態監視を確立する。音声ブラウザ環境では、状態監視されたデバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Monitor Stop	既存の状態監視を終了する。

さらに、次のサービスの少なくとも1つを、エンドポイントから接続された呼を移動させるためにサポートする必要がある。

サービス	説明
Single Step Transfer (of a connected call)	別のエンドポイントに呼を転送する。音声ブラウザ環境では、転送デバイスが音声ブラウザ・プラットフォーム上のエンドポイントであり、Single Step Transfer サービスを完了した後、その呼とは関連しない。
Deflect (of a connected call)	着信転送デバイスからコネクションを移動する。音声ブラウザ環境では、着信転送デバイスが音声ブラウザ・プラットフォーム上のエンドポイントであり、Deflect Call サービスが完了した後、その呼とは関連しない。

5.3.2 イベント (Events)

次の CSTA のイベントが Level 2 Voice Browser プロファイルに含まれる。

イベント	説明
Connection Cleared	エンドポイントが呼から切断されたことを示す。
Delivered	呼がエンドポイントを呼出中であることを示す。
Established	エンドポイントが応答もしくは呼と接続されたことを示す。
Failed	呼が完了されない(例えば、話中のエンドポイントと遭遇した場合)ことを示す。
Network Reached	発呼において、呼が網インタフェース・デバイス経由で外部の網に接続されたことを示す。
Originated	発呼において、発側エンドポイント(音声ブラウザ・プラットフォーム上の)が呼に接続されたことを示す。
Diverted	(もし Deflect サービスが接続された呼を移動するために使用された場合)
Transferred	(もし Single Step Transfer サービスが接続された呼を転送するために使用された場合)

5.4 基本 Telephony プロファイル (Basic Telephony Profile)

5.4.1 サービス (Services)

このプロファイルでは、アプリケーションが CSTA 実装を ECMA-323 Get Switching Function Capabilities サービスにより認識するために、その能力を示すよう要求される。能力は、テレフォニ・プラットフォームによってサポートされる ECMA-323 サービスとイベントのリストと、実装によってサポートされる多くのタイプの動作オプションを含む(プロファイルが提供され、外線発信のためにダイヤル文字列で許可されたディジット・タイプなどである)。多くのパラメータはオプションであり、ECMA-323 メッセージで提供される必要がない。

次の CSTA サービスが基本 Telephony プロファイルに含まれる。

サービス	説明
Answer Call	呼出中の呼に回答する。音声ブラウザで、応答デバイスがプラットフォーム上のエンドポイントである。
Clear Connection	コネクションを切断する。音声ブラウザ環境で、切断デバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Make Call	2つのデバイス間の呼を確立する。音声ブラウザ環境で、発呼デバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Monitor Start	エンドポイント上でデバイス・タイプ状態監視を確立する。音声ブラウザ環境では、状態監視されたデバイスが音声ブラウザ・プラットフォーム上のエンドポイントである。
Monitor Stop	既存の状態監視を終了する。

5.4.2 イベント (Events)

次の CSTA イベントが基本 Telephony プロファイルに含まれる。

イベント	説明
Connection Cleared	エンドポイントが呼から切断されたことを示す。
Delivered	呼がエンドポイントを呼出中であることを示す。
Established	エンドポイントが応答もしくは呼と接続されたことを示す。
Failed	呼が完了されない(例えば、話中のエンドポイントと遭遇した場合)ことを示す。
Network Reached	発呼において、呼が網インタフェース・デバイス経由で外部の網に接続されたことを示す。
Originated	発呼において、発側エンドポイント(音声ブラウザ・プラットフォーム上の)が呼に接続されたことを示す。
Service Initiated	エンドポイント(音声ブラウザ・プラットフォーム上の)がサービス(例えば、発信音)を要求していることを示す。いくつかのエンドポイントがこの状態を飛び越えて、このイベント(例えば、発信音を提供しないエンドポイント)を提供しないはずである。

5.5 他の機能 (Other Features)

これらのプロファイルの中で定義されたサービスは CSTA で標準化された呼制御機能の非常にわずかなサブセットである。テレフォニ・プラットフォームは、ECMA-323 拡張メカニズムを用いることによって ECMA-323 の標準化されていない機能を提供したり、たくさんの追加 ECMA-323 サービスを提供することができる。

次の ECMA-323 サービスとイベントは、さらに音声ブラウザ・アプリケーションに非常に役立つかもしれない。

サービスとイベント	説明
Single Step Conference Service	このサービスは既存の呼に別のデバイスを加える。Single Step Transfer サービスとは異なり、このサービスの結果として呼からどのデバイスも切り離されない。
Conferenced Event	デバイスが既存の呼に加えられ、既存の呼からデバイスが取り除かれなかったことを示す。
Get Switching Function Devices Service	このサービスは、テレフォニ・プラットフォームによって提供されるエンドポイント識別子のリストを提供する。もしもこのサービスがサポートされなければ、音声ブラウザ・アプリケーションは ECMA-323 インタフェースの外部でエンドポイント識別子を決定しなければならない。

6 ECMA-323 実例 (ECMA-323 Illustrative Examples)

本章では、音声ブラウザを使用したシナリオのコンテキストにおいて ECMA-323 XML メッセージの使用例を示す。

本章におけるイベントシーケンス例

- ・ テレフォニ・プラットフォームによってサポートされる機能の把握
- ・ デバイス上で状態監視の確立 (すなわち、着呼の認識)
- ・ 着呼呼出コネクションの通知
- ・ 着呼コネクションに応答する、接続されたコネクションの通知
- ・ コネクションの切断、切断されたコネクションの通知
- ・ 発呼の開始
- ・ single step transfer シナリオ
- ・ single step conference シナリオ

6.1 テレフォニ・プラットフォームの能力把握 (Discovering the Capabilities of a Telephony Platform)

音声ブラウザ・アプリケーションは、ECMA-323 Get Switching Function Capabilities サービスを使用することにより、サポート能力を把握することができる。詳細は ECMA-323 11.1.3 を参照のこと。

6.2 状態監視の開始(着呼の認識) (Starting a Monitor (i.e. listening for incoming calls))

ECMA-323 呼制御イベントを受信するためには、ECMA-323 Monitor Start サービスを介して、CSTA 状態監視をテレフォニ・エンドポイントに対して設定しなければならない。

6.2.1 状態監視開始要求 - サービス要求例 (Monitor Start – Service Request example)

この例では、音声ブラウザ・アプリケーションが識別子 22343 に関連付けられたエンドポイントに対してどのように状態監視を設定すればよいかを示している。ECMA-323 のオプション・パラメータに関しては省略しており、最も簡略した状態監視 (デバイス上のデバイス・タイプ状態監視) を要求している。

Monitor Start サービス要求の必須パラメータ

パラメータ	説明
monitorObject	エンドポイントのデバイス識別子。これは、ECMA-323 もしくは ECMA-323 の外部のメカニズムによって得られる。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<MonitorStart xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorObject>
    <deviceObject>22343</deviceObject>
  </monitorObject>
</MonitorStart>
```

6.2.2 状態監視開始要求 - サービス応答例 (Monitor Start – Service Response example)

この例は Monitor Start 要求の応答を示す。99 の monitorCrossRefID は、この状態監視要求に関連する全ての ECMA-323 イベントで提供される。

Monitor Start サービス応答の必須パラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、この状態監視要求と次の ECMA-323 イベントを関連づけるためのハンドルを提供する。各々の ECMA-323 イベントはこのハンドルを含むので、これによりイベントと Monitor Start 要求を発行した音声ブラウザ・アプリケーションを関連づけるために使用される。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<MonitorStartResponse xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
</MonitorStartResponse>
```

6.3 着呼通知 (Notification of an Inbound Call)

状態監視されたエンドポイントに着呼した場合、ECMA-323 Delivered イベントがテレフォニ・プラットフォームにより提供される。

Delivered イベントは、例えば網によって提供される ANI や DNIS 情報のような呼についての情報を含む。この情報は、音声ブラウザ・アプリケーションに、呼に応答するか拒否するかなど、呼をどのように扱えばよいかを決定することを可能にする。

6.3.1 Delivered イベント(Delivered Event example)

Delivered イベントに含まれるパラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、Monitor Start 応答で提供されるハンドルを表す。
connection	このパラメータは、このコネクションを使用する CSTA サービス (Answer Call) で使用されるコネクション識別子を含んでいる。
alertingDevice	このパラメータは、呼出中コネクションのエンドポイント識別子を示している。着呼では、音声ブラウザ上のエンドポイントのことであり、発呼では、呼出中の着呼デバイスを示す。
callingDevice	このパラメータは、発呼デバイスの自局内における表現を意味する。テレフォニ・プラットフォームの機能 (転送、フォワード、会議など) のために、発呼デバイスは音声ブラウザ・プラットフォームにたどり着くまでに変更されることも考えられる。
calledDevice	このパラメータは、着呼デバイスの自局内における表現を意味する。テレフォニ・プラットフォームの機能 (転送、フォワード、会議など) のために、着呼デバイスは音声ブラウザ・プラットフォームにたどり着くまでに変更されることも考えられる。
lastRedirectionDevice	呼が以前にリダイレクトされたかどうかを示す。
localConnectionInfo	自局コネクションのコネクション状態が、「alerting」であることを示す。これは典型的に音声ブラウザ・プラットフォーム上の状態監視されたデバイスと関連したコネクションである。
cause	イベントの原因を示す。この例の場合は「new」呼を表す。
networkCallingDevice	着呼において、網によって提供された発呼デバイス情報 (ANI など) を表す。この情報は、呼とともに常に存在し (Calling デバイスと異なり) 決して変更されない。
networkCalledDevice	着呼において、網によって提供された着呼デバイス情報 (DNIS など) を表す。この情報は、呼とともに常に存在し (Called デバイスと異なり) 決して変更されない。
associatedCallingDevice	このパラメータは、着呼の場合に提供される。この例では、網インタフェース・デバイス識別子が 023 であることを意味する。これはまた「notKnown」の値を提供されることも可能である。

サンプル :

```

<?xml version="1.0" encoding="UTF-8"?>
<DeliveredEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <connection>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </connection>
  <alertingDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </alertingDevice>
  <callingDevice>
    <deviceIdentifier>14085551212</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </calledDevice>
  <lastRedirectionDevice>
    <notRequired/>
  </lastRedirectionDevice>
  <localConnectionInfo>alerting</localConnectionInfo>
  <cause>newCall</cause>
  <networkCallingDevice>
    <deviceIdentifier>14085551212</deviceIdentifier>
  </networkCallingDevice>
  <networkCalledDevice>
    <deviceIdentifier>18001234567</deviceIdentifier>
  </networkCalledDevice>
  <associatedCallingDevice>
    <deviceIdentifier>023</deviceIdentifier>
  </associatedCallingDevice>
</DeliveredEvent>

```

6.4 着呼応答 (Answering an Inbound Call)

音声ブラウザ・アプリケーションが着呼を通知された後、その呼に応答するために ECMA-323 Answer Call サービスを使用する。

6.4.1 応答要求 - サービス要求例 (Answer Call – Service Request example)

Answer Call サービス要求の必須パラメータ

パラメータ	説明
callToBeAnswered	呼び出し接続の接続識別子。アプリケーションは、Delivered イベントの中で提供される接続・パラメータの内容を使用する。

サンプル :

```

<?xml version="1.0" encoding="UTF-8"?>
<AnswerCall xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <callToBeAnswered>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </callToBeAnswered>
</AnswerCall>

```

6.4.2 応答要求 - サービス応答例 (Answer Call – Service Response example)

テレフォニ・プラットフォームは Answer Call サービス要求に対して肯定応答を提供する。必須サービス応答パラメータは無い。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<AnswerCallResponse xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2"/>
```

6.5 接続された呼の通知 (Notification of a Connected Call)

Answer Call サービスの結果、ECMA-323 Established イベントが接続状態になったことを示すために、テレフォニ・プラットフォームによって提供される。テレフォニ・プラットフォームに関して、呼に自動応答するように設定されていた場合に、Established イベントが音声ブラウザ・アプリケーションに送出される最初のイベントになる。

6.5.1 Established イベントの例 (Established Event example)

Established イベントは、Delivered イベントと類似した情報を含むので、パラメータ詳細説明は省略する。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<EstablishedEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <establishedConnection>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </establishedConnection>
  <answeringDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </answeringDevice>
  <callingDevice>
    <deviceIdentifier>14085551212</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </calledDevice>
  <lastRedirectionDevice>
    <numberDialed/>
  </lastRedirectionDevice>
  <localConnectionInfo>connected</localConnectionInfo>
  <cause>normal</cause>
  <networkCallingDevice>
    <deviceIdentifier>14085551212</deviceIdentifier>
  </networkCallingDevice>
  <networkCalledDevice>
    <deviceIdentifier>18001234567</deviceIdentifier>
  </networkCalledDevice>
  <associatedCallingDevice>
    <deviceIdentifier>023</deviceIdentifier>
  </associatedCallingDevice>
</EstablishedEvent>
```

6.6 コネクションの切断 (Clearing a Connection)

一旦、音声ブラウザ・アプリケーションが呼から切り離されると、アプリケーションは、ECMA-323 Clear Connection サービスをテレフォニ・プラットフォームに送出する。

6.6.1 Clear Connection - サービス要求例 (Clear Connection- Service Request example)

パラメータ	説明
connectionToBeCleared	切断されたコネクション識別子。アプリケーションは以前のイベントでテレフォニ・プラットフォームにより提供されたコネクション識別子を使用する。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<ClearConnection xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <connectionToBeCleared>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </connectionToBeCleared>
</ClearConnection>
```

6.6.2 切断要求 - サービス応答例 (Clear Connection - Service Response example)

テレフォニ・プラットフォームは Clear Connection サービス要求に対して肯定応答を通知する。必須サービス応答パラメータは無い。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<ClearConnectionResponse xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2"/>
```

6.7 Cleared コネクションの通知 (Notification of a Cleared Connection)

Clear Connection サービスの結果として、ECMA-323 Connection Cleared イベントは、コネクションが切断されたことを示すために、テレフォニ・プラットフォームに対して送出される。

6.7.1 Connection Cleared イベント例 (Connection Cleared Event example)

Connection Cleared イベントに含まれるパラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、Monitor Start 応答で提供されるハンドルを表す。
droppedConnection	このパラメータは、切断されたコネクションのコネクション識別子を含む。
releasingDevice	このパラメータは、解放デバイスのエンドポイント識別子を示す。
localConnectionInfo	自局コネクションのコネクション状態が、「null」であることを示す。これは典型的に音声ブラウザ・プラットフォーム上の状態監視されたデバイスと関連したコネクションである。
cause	イベントの原因を示す。この例の場合は「normal」切断を表す。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<ConnectionClearedEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <droppedConnection>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </droppedConnection>
  <releasingDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </releasingDevice>
  <localConnectionInfo>null</localConnectionInfo>
  <cause>normalClearing</cause>
</ConnectionClearedEvent>
```

注

音声ブラウザ・アプリケーションは、相手局エンドポイントが呼を解放することを示す *Connection Cleared* イベントを受信するために準備しなければならない(すなわち相手が最初に切断した場合)。

6.8 発呼の開始 (Initiating an Outbound Call)

音声ブラウザ・アプリケーションは、ECMA-323 Make Call サービスを利用することにより、発呼を開始することができる。発呼を開始する前に、アプリケーションは、呼の経過を監視するために発呼デバイスの状態監視をすべきである。

6.8.1 Make Call-サービス要求例(プロファイルを参照) Make Call – Service Request example (refer to Profile)

この例では、アプリケーションは(音声ブラウザ上の)デバイス識別子 22343 から電話番号 18005551212 と関連づけられた外部デバイスへの呼を確立させたい。

Make Call サービス要求の必須パラメータ

パラメータ	説明
callingDevice	音声ブラウザ・プラットフォーム上の発呼デバイスに関連したデバイス識別子(22343)。
calledDevice	電話番号 18005551212 に関連付けられたエンドポイントを表す。
autoOriginate	「doNotPrompt」の値を設定することによって、発呼デバイスのコネクションが自動的に応答されることを示す。このモードは、常に音声ブラウザ・プラットフォームに関連付けられたデバイスのために使用されるべきである。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<MakeCall xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <callingDevice>22343</callingDevice>
  <calledDirectoryNumber>18005551212</calledDirectoryNumber>
  <autoOriginate>doNotPrompt</autoOriginate>
</MakeCall>
```

6.8.2 Make Call-サービス応答例 (Make Call – Service Response example)

この例は Make Call 要求に対する応答例を示す。

Make Call サービス要求応答の必須パラメータ

パラメータ	説明
callingDevice	このパラメータは、次に続くサービスで参照するために使用する接続識別子を提供する。これはまた接続に関連付けられた後のイベントにも提供される。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<MakeCallResponse xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <callingDevice>
    <callID>2</callID>
    <deviceID>22343</deviceID>
  </callingDevice>
</MakeCallResponse>
```

6.9 発呼イベント・シーケンス (Outbound Call Event Sequence)

この例は、Make Call サービス要求の結果として生成されるイベントのシーケンスを示す。このイベント・シーケンスは、Originated、Network Reached、Delivered 及び Established イベントを含む。

6.9.1 Originated イベント例 (Originated Event Example)

Originated イベントは、(音声ブラウザに関連付けられた)発呼デバイスが呼に接続されたことを示す。

Originated イベントに含まれるパラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、Monitor Start 応答で提供されるハンドルを表す。
originatedConnection	このパラメータは、音声ブラウザ・プラットフォーム上の発呼接続の接続識別子を含んでいる。これは、この接続を適用した CSTA サービスで利用される。
callingDevice	これは音声ブラウザ・プラットフォームに関連した発呼デバイスのローカル表現である。
calledDevice	これは着呼デバイスの表現である。この例では電話番号 18005551212 である。
localConnectionInfo	発側接続で「connected」の接続状態を示す。これは着呼デバイスの接続の接続状態でないことに注意する。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<OriginatedEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <originatedConnection>
    <callID>2</callID>
    <deviceID>22343</deviceID>
  </originatedConnection>
  <callingDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <deviceIdentifier>18005551212</deviceIdentifier>
  </calledDevice>
  <localConnectionInfo>connected</localConnectionInfo>
  <cause>makeCall</cause>
</OriginatedEvent>
```

6.9.2

Network Reached イベントは、呼が網インタフェース・デバイス (公衆網やトランクなど) に遭遇した

ことを示す。

Network Reached イベントに含まれるパラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、Monitor Start 応答で提供されるハンドルを表す。
outboundConnection	このパラメータは、網インタフェース・デバイスに関連付けられた発呼コネクションのコネクション識別子を含む。
networkInterfaceUsed	網インタフェース・デバイスのエンドポイント識別子を示す。
callingDevice	これは音声ブラウザ・プラットフォームに関連した発呼デバイスのローカル表現である。
calledDevice	これは着呼デバイスの表現である。この例では電話番号 18005551212 である。
lastRedirectionDevice	「notKnown」は不明な転送番号を示すために提供される。
localConnectionInfo	自局コネクション（典型的には originatedConnection）で「connected」のコネクション状態を示す。これは着呼デバイスのコネクションのコネクション状態でないことに注意する。
cause	「normal」が原因として提供される。

サンプル：

```
<?xml version="1.0" encoding="UTF-8"?>
<NetworkReachedEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <outboundConnection>
    <callID>2</callID>
    <deviceID>023</deviceID>
  </outboundConnection>
  <networkInterfaceUsed>
    <deviceIdentifier>023</deviceIdentifier>
  </networkInterfaceUsed>
  <callingDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <deviceIdentifier>18005551212</deviceIdentifier>
  </calledDevice>
  <lastRedirectionDevice>
    <notRequired/>
  </lastRedirectionDevice>
  <localConnectionInfo>connected</localConnectionInfo>
  <cause>normal</cause>
</NetworkReachedEvent>
```

6.9.3 Delivered イベント (Delivered Event)

もし内在するシグナリングが、着呼デバイスが呼出中であることを検出できるならば、テレフォニ・プラットフォームは Delivered イベントを提供する。

Delivered イベントに含まれるパラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、Monitor Start 応答で提供されるハンドルを表す。
connection	このパラメータは、呼出中の相手局デバイスに関連づけられたコネクション識別子を含む。発呼において、これは着呼デバイスを示す網インタフェース・デバイスへのコネクションである。
alertingDevice	このパラメータは、呼出中コネクションのエンドポイント識別子を提供する。発呼において、呼出中デバイスは着呼デバイスが呼出中であることを示す。
callingDevice	これは発呼デバイスのローカル表現である（音声ブラウザ・プラットフォームに関連づけられたデバイス）。
calledDevice	これは着呼デバイスである。
lastRedirectionDevice	呼が以前にリダイレクトされたかどうかを示す。この例では、「NotRequired」が提供されている。
localConnectionInfo	音声ブラウザ・プラットフォーム上でコネクションに対する「connected」のコネクション状態を示す。これは着呼デバイスのコネクションのコネクション状態でないことに注意する。
cause	イベントの原因を示す。この場合、「networkSignal」は、網インタフェース・デバイスに対する相手局デバイスであることをイベントが表している。
associatedCalledDevice	このパラメータは、発呼の場合に提供される。この例では、網インタフェース・デバイス識別子が 023 であることを意味する。これはまた「notKnown」の値を提供されることも可能である。

サンプル：

```
<?xml version="1.0" encoding="UTF-8"?>
<DeliveredEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <connection>
    <callID>2</callID>
    <deviceID>023</deviceID>
  </connection>
  <alertingDevice>
    <deviceIdentifier>18005551212</deviceIdentifier>
  </alertingDevice>
  <callingDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <deviceIdentifier>18005551212</deviceIdentifier>
  </calledDevice>
  <lastRedirectionDevice> <notRequired/>
</lastRedirectionDevice>
  <localConnectionInfo>connected</localConnectionInfo>
  <cause>networkSignal</cause>
  <associatedCalledDevice>
    <deviceIdentifier>023</deviceIdentifier>
  </associatedCalledDevice>
</DeliveredEvent>
```

6.9.4 Established イベント (Established Event)

Established イベントは、着呼パーティが呼に応答したことをテレフォニ・プラットフォームが検出したときに通知される。着呼デバイスが応答したことを検出する能力は、内在するシグナリング能力に依存する。

Established イベントは Delivered イベントに類似しているため、例を示さない。

6.10 シングル・ステップ転送 (Single Step Transfer)

音声ブラウザ・アプリケーションは、音声ブラウザ・プラットフォームから別のエンドポイントへ発呼者を転送するため、ECMA-323 で定義された Single Step Transfer サービスを使用することができる。呼を転送するプロセスでは、音声ブラウザ・プラットフォームの転送元の接続は呼から切断される。

6.10.1 シングル・ステップ転送 - サービス要求例 (Single Step Transfer - Service Request example)

Single Step Transfer サービス要求に含まれるパラメータ

パラメータ	説明
activeCall connection	転送元接続の接続識別子 (音声ブラウザ・プラットフォームでの接続)。
transferredTo	転送先のエンドポイントに関連づけられたデバイス識別子。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<SingleStepTransferCall xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <activeCall>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </activeCall>
  <transferredTo>333333</transferredTo>
</SingleStepTransferCall>
```

6.10.2 シングル・ステップ転送 - サービス応答例 (Single Step Transfer - Service Response example)

テレフォニ・プラットフォームは、Single Step Transfer サービス要求に対する肯定応答を通知する。

Single Step Transfer サービス応答に含まれるパラメータ

パラメータ	説明
transferredCall connection	transferredTo 接続の接続識別子。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<SingleStepTransferCallResponse xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <transferredCall>
    <callID>2</callID>
    <deviceID>333333</deviceID>
  </transferredCall>
</SingleStepTransferCallResponse>
```

6.11 Transferred Connection の通知 (Notification of a Transferred Connection)

Single Step Transfer サービスの結果により、転送元接続が切断され、保留されたデバイスが転送先デバイスに接続されたことを示す、ECMA-323 の Transferred イベントがテレフォニ・プラットフォームによって通知される。

6.11.1 Transferred イベント例 (Transferred Event example)

このイベントには以下のパラメータが含まれる。

Transferred Event に含まれるパラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、Monitor Start 応答で提供されるハンドルを表す。
primaryOldCall connection	このパラメータは、音声ブラウザ・プラットフォーム上の転送元エンドポイントの接続の接続識別子を含む。
transferringDevice	このパラメータは、音声ブラウザ・プラットフォームにおける転送元デバイスのエンドポイント識別子を提供する。
transferredToDevice	このパラメータは、転送先デバイス（すなわち呼の中で新たに加わったデバイス）を表す。
transferredConnections	Single Step Transfer では、これは、転送の結果切断された転送元デバイスの旧接続を含む。
localConnectionInfo	音声ブラウザ・プラットフォームで、自局接続の接続状態が、「null」であることを示す。
cause	イベントの原因を示す。この場合、転送が Single Step Transfer であることを示す。

サンプル：

```
<?xml version="1.0" encoding="UTF-8"?>
<TransferredEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <primaryOldCall>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </primaryOldCall>
  <transferringDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </transferringDevice>
  <transferredToDevice>
    <deviceIdentifier>333333</deviceIdentifier>
  </transferredToDevice>
  <transferredConnections>
    <connectionListItem>
      <oldConnection>
        <callID>1</callID>
        <deviceID>22343</deviceID>
      </oldConnection>
    </connectionListItem>
  </transferredConnections>
  <localConnectionInfo>null</localConnectionInfo>
  <cause>singleStepTransfer</cause>
</TransferredEvent>
```

音声ブラウザ・アプリケーションは、Transferred イベントの後の Transferred デバイスに対する、Connection Cleared イベントを無視する。Transferred イベントは転送元デバイスが呼から既に切断されたことを示す。

6.12 着信転送 (Deflect)

音声ブラウザ・アプリケーションは、音声ブラウザ・プラットフォームから別のエンドポイントへ呼を移動するために ECMA-323 で定義された Deflect サービスを使用できる。

6.12.1 着信転送 - サービス要求例 (Deflect - Service Request example)

Deflect サービス要求に含まれるパラメータ

パラメータ	説明
callToBeDiverted connection	移動されるコネクション識別子 (音声ブラウザ・プラットフォームにおけるコネクション)。
newDestination	新しい着信先に関連づけられたデバイス識別子。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<DeflectCall xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <callToBeDiverted>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </callToBeDiverted >
  <newDestination>333333</ newDestination>
</DeflectCall>
```

6.12.2 着信転送 - サービス応答例 (Deflect - Service Response example)

テレフォニ・プラットフォームは、Deflect サービス要求に対する肯定応答を通知する。必須サービス応答パラメータは無い。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<DeflectCallResponse xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2"/>
```

6.13 着信転送 Connection の通知 (Notification of a Diverted Connection)

Deflect サービスの結果により、着信転送されたコネクションは切断され、呼は別のデバイスへ移動させられたことを示す ECMA-323 で定義された Diverted イベントが、テレフォニ・プラットフォームによって通知される。

6.13.1 Diverted イベントの例 (Diverted Event example)

Diverted イベントに含まれるパラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、Monitor Start 応答で提供されるハンドルを表す。
connection	このパラメータは、別のエンドポイントへ移動させられた音声ブラウザ・プラットフォーム上のコネクションのコネクション識別子を含む。
divertingDevice	このパラメータは、音声ブラウザ・プラットフォームにおける着信転送デバイスのエンドポイント識別子を提供する。
newDestination	このパラメータは、着信転送先デバイス (すなわち呼の中で新たに加わったデバイス) を表す。
localConnectionInfo	音声ブラウザ・プラットフォームで、自局コネクションのコネクション状態が、「null」であることを示す。
cause	イベントの原因を示す。この場合、着信転送であることを示す。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<DivertedEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <connection>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </connection>
  <divertingDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </divertingDevice>
  <newDestination>
    <deviceIdentifier>333333</deviceIdentifier>
  </newDestination>
  <localConnectionInfo>null</localConnectionInfo>
  <cause>redirected</cause>
</DivertedEvent>
```

音声ブラウザ・アプリケーションは、Diverted イベントの後の着信転送デバイスに対する Connection Cleared イベントを無視する。Diverted イベントは着信転送デバイスが呼から既に切断されたことを示す。

6.14 シングル・ステップ会議(Single Step Conference)

音声ブラウザ・プラットフォームにおいて音声ブラウザ・アプリケーションは、既に存在する呼に別のデバイスを加えるため、ECMA-323 で定義された Single Step Conference を使用することができる。このサービスの結果により、呼から抜けるデバイスはない。

6.14.1 Single Step Conference – サービス要求 (Single Step Conference – Service Request example)

Single Step Conference サービス要求に含まれるパラメータ

パラメータ	説明
activeCall connection	会議通話中のコネクションのコネクション識別子 (音声ブラウザ・プラットフォームにおけるコネクション)。
deviceToJoin	既に存在する呼に加えられるデバイスに関連づけられたデバイス識別子。
participationType	加えられるデバイスの参加種別 (聴話での参加、会議通話での参加など) を指定する。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<SingleStepConferenceCall xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <activeCall>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </activeCall>
  <deviceToJoin>55555</deviceToJoin>
  <participationType>active</participationType>
</SingleStepConferenceCall>
```

6.14.2 シングル・ステップ会議 - サービス応答例 (Single Step Conference – Service Response example)

テレフォニ・プラットフォームは、Single Step Conference サービス要求に対する肯定応答を通知する。

Single Step Conference サービス応答に含まれるパラメータ

パラメータ	説明
conferencedCall connection	deviceToJoin のコネクションのコネクション識別子 (deviceToJoin がテレフォニ・プラットフォーム外のデバイスの場合、このコネクションはネットワーク・インタフェースに関連づけられたデバイスのコネクションを表すことに注意する)。

サンプル :

```
<?xml version="1.0" encoding="UTF-8"?>
<SingleStepConferenceCallResponse xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <conferencedCall>
    <callID>1</callID>
    <deviceID>55555</deviceID>
  </conferencedCall>
</SingleStepConferenceCallResponse>
```

6.15 呼に追加されたパーティの通知 (Notification of an Party Added to a call)

Single Step Conference サービスの結果により、新しいデバイスが既存の呼に追加されたことを示す ECMA-323 で定義された Conferenced イベントが、テレフォニ・プラットフォームによって通知される。

Conferenced イベントの後で、新しいデバイスでの呼の進行状況を反映するために、追加のイベントが生成されるかも知れない (例えば、追加されたデバイスが応答した時の Established イベント)。

6.15.1 Conferenced イベント例

このイベントは以下のパラメータを含む。

Conferenced イベントに含まれるパラメータ

パラメータ	説明
monitorCrossRefID	このパラメータは、Monitor Start 応答で提供されるハンドルを表す。
primaryOldCall connection	このパラメータは、音声ブラウザ・プラットフォーム上の会議主催者のエンドポイントのコネクションのコネクション識別子を含む。
conferencingDevice	このパラメータは、音声ブラウザ・プラットフォームにおける会議主催デバイスのエンドポイント識別子を提供する。
addedParty	このパラメータは、追加されたデバイス (すなわち、呼の中で新たに加わったデバイス) を表す。
conferenceConnections	Single Step Conference では、これは最終結果の呼に含まれるデバイスのコネクション識別子のリストを含む。この例では最終結果の呼に含まれる3つのエンドポイント (22343, 33333, 55555) を示す (リストには各エンドポイントのコネクション識別子とそれに関連づけられたデバイス識別子が示される)。
localConnectionInfo	音声ブラウザ・プラットフォーム上のコネクションのコネクション状態を示す。この例では、コネクション状態「Connected」が通知される。
cause	イベントの原因を示す。この場合、会議が Single Step Conference であることを示す。

サンプル :

```

<?xml version="1.0" encoding="UTF-8"?>
<ConferencedEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <monitorCrossRefID>99</monitorCrossRefID>
  <primaryOldCall>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </primaryOldCall>
  <conferencingDevice>
    <deviceIdentifier>22343</deviceIdentifier>
  </conferencingDevice>
  <addedParty>
    <deviceIdentifier>55555</deviceIdentifier>
  </addedParty>
  <conferenceConnections>
    <connectionListItem>
      <newConnection>
        <callID>1</callID>
        <deviceID>22343</deviceID>
      </newConnection>
      <Endpoint>
        <deviceID>22343</deviceID>
      </Endpoint>
    </connectionListItem>
    <connectionListItem>
      <newConnection>
        <callID>1</callID>
        <deviceID>33333</deviceID>
      </newConnection>
      <Endpoint>
        <deviceID>33333</deviceID>
      </Endpoint>
    </connectionListItem>
    <connectionListItem>
      <newConnection>
        <callID>1</callID>
        <deviceID>55555</deviceID>
      </newConnection>
      <Endpoint>
        <deviceID>55555</deviceID>
      </Endpoint>
    </connectionListItem>
  </conferenceConnections>
  <localConnectionInfo>connected</localConnectionInfo>
  <cause>singleStepConference</cause>
</ConferencedEvent>

```

6.16 否定応答例 (Failure Response example)

CSTACode メッセージは、ECMA-323 で定義されたサービス要求に対する否定応答を示すために使用される。この例は Monitor Start 要求に対するエラー応答を示す。

CSTACode の必須パラメータは、以下の通り。

- ・ 7つのエラー・カテゴリの選択肢がある。この例では、エラー・カテゴリ「operation」が特定のエラー値「invalidMonitorObject」として示される。これは Monitor Start 要求で指定されたデバイス識別子(22343)が無効であることを示す。

サンプル：

```
<?xml version="1.0" encoding="UTF-8"?>
<CSTAErroCode xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <operation>invalidMonitorObject</operation>
</CSTAErroCode>
```

CSTAErroCode メッセージはあらゆる ECMA-323 で定義されたサービス要求に対する否定応答を示すために使用されることに注意する。全ての有効なエラーコードについては、ECMA-323 の 9.19 節を参照のこと。

7 SALT/CSTA XML プログラム例 (SALT/CSTA XML Programming Example)

次の例は、SALT 1.0 仕様から得られる。その例は、SALT の ECMA-323 の使用を実証する。

この例の主な目的は、発信者に転送先電話番号を尋ねて、呼を転送するというシンプルなものである。SALT アプリケーションは論理上次のセクションから構成することができる。

アプリケーション用データ

```
<input name="transferTarget" />
<input name="callerID" />
<input name="callID" />
<input name="deviceID" />
<input name="monitorObject" type="hidden" value="2234" />
<input name="monitorCrossRefID" />
```

英語 (自然言語によって唯一影響されたセクション) の音声オブジェクト

```
<listen id="recNumber" onreco="procRecNumber()"
onnoreco="procNoReco()" onsilence="procNoReco()">
  <grammar src="..." />
</listen>

<listen id="recYesNo" onreco="procYesNo()" onnoreco="procNoReco()"
onsilence="procNoReco()">
  <grammar src="..." />
</listen>

<prompt id="sayWelcome">Hello! Please say the phone number to transfer to. </prompt>
<prompt id="askAgain">Sorry, I missed that. Please say the number again. </prompt>
<prompt id="confirm">Did you say <value href="transferTarget"/>? </prompt>
<prompt id="sayBye">Thank you. Your call is being transferred. </prompt>
<prompt id="tryAgain">
The number, <value href="transferTarget"/>, cannot be
reached for transfer. Please try again later.
</prompt>
```

ECMA スクリプトの音声イベント・ハンドラ (ダイアログ・ロジック)

```
<script><!--
function procRecNumber() {
  var msg = event.srcElement.recoresult;
  transferTarget.value = msg.SelectSingleNode("*/phoneNumber").nodeValue;
  // read recognized phone number
  var confidence = msg.selectSingleNode("/@confidence").nodeValue;
  if (confidence < 0.5) {
    confirm.Start(); recYesNo.Start();
  } else {
    sayBye.Start(); ccTransfer();
  }
}
}
```

```

function procYesNo() {
    var answer = event.srcElement.recoresult.SelectSingleNode(
        "*/yes[@confidence>0.5]");
        // accept only yes with confidence
    if (answer == null) {
        procNoReco();
    } else {
        sayBye.Start(); ccTransfer();
    }
}

function procNoReco() {
    transferTarget.value = "";
    askAgain.Start(); recNumber.Start();
}
--></script>

```

呼制御セクション (場面やダイアログ・ロジックに影響されない)

```

<smex id="callControl" onreceive="ccHandler()">...</smex>
<script><!--
    // The chandler handles the ECMA-323 events.
    //
    // Once the connection is answered, a welcome prompt
    // is played and the transfer target telephone number is solicited.
    //
    // When the speech event handler detects and confirms the correct
    // speech input, an ECMA-323 SingleStepTransfer service is used to
    // transfer the caller to the new transfer target.
    //
function ccHandler() {
    var msg = event.srcElement.received;
    if (msg.nodeName == "DeliveredEvent") { // incoming call notification
        //
        // If the connection is alerting (DeliveredEvent, ECMA-323, 15.2.5) the
        // connection information from the Delivered event is saved
        // called.value and deviceID.value) and the call is answered by using the
        // ECMA-323 AnswerCall service with the saved connection information.
        // If the application needed the ANI and DNIS, it could also obtain
        // this information from this event.
        //
        callID.value = msg.selectSingleNode(
            "./connection/callID").nodeValue;
        deviceID.value = msg.selectSingleNode(
            "./connection/deviceID").nodeValue;
        ccAnswer();
    } else if (msg.nodeName == "EstablishedEvent") { // call answered //

        // Once the connection is answered (EstablishedEvent, ECMA-323, 15.2.8)
        // a welcome prompt is played and the transfer target telephone number
        // is solicited.
        //
        callerID.value = msg.selectSingleNode(
            "./callingDevice/DeviceIdentifier").nodeValue;
        sayWelcome.Start(); recNumber.Start();
    } else if (msg.nodeName == "TransferredEvent") { // call transferred

```

```

//
// The TransferredEvent (ECMA-323, 15.2.18) is received when
// the transfer has been completed. ccCleanup is called to clean up
// the application data.
//
    ccCleanup();
} else if (msg.nodeName == "ConnectionClearedEvent") { // user hang up
//
// A user hang up is indicated by a ConnectionClearedEvent (ECMA-323,
// 15.2.4) which flushes the prompt queue and cleans the application
// data. This could happen at any time during the call.
//
    promptQueue.Flush();
    ccCleanup();
} else if (msg.nodeName == "CSTAErrorCode") { // service failure event
//
// The ccError function handles any failure responses from any of the
// ECMA-323 services that may have failed.
//
    ccError();
} // feel free to handle other events here
}

function ccTransfer() { // transferring a call
//
// The SingleStepTransferCall service (ECMA-323, 15.1.24) is used to
// invoke the transfer. There are two elements provided. The first
// element is the connection information that was obtained from
// the DeliveredEvent. The second element is the transfer target that
// was solicited from the caller.
//
callControl.sent = "<SingleStepTransferCall" +
" xmlns='http://www.ecma.ch/standards/ecma-323/csta/ed2'>" +
"<activeCall><callID>" +
callID.value + "</callID><deviceID>" +
deviceID.value + "</deviceID></activeCall><transferredTo>" +
transferTarget.value+"</transferredTo></SingleStepTransferCall>";
}

function ccStartListening() { // listening for call events
/
// The MonitorStart service (ECMA-323, 13.1.2) is used to place a
// monitor on a device so that events can be generated when activity
// happens at that device. The single element provided indicates the
// identifier of the device that is to be monitored. In this example
// it was part of the application data.
//
callControl.sent = "<MonitorStart" +
" xmlns='http://www.ecma.ch/standards/ecma-323/csta/ed2'>" +
"<monitorObject><deviceObject>" +
monitorObject.value +
"</deviceObject></monitorObject></MonitorStart>";
}

function ccAnswer() { // answering a call
//
// The AnswerCall service (ECMA-323, 15.1.3) is used to answer the
// alerting connection. The single element provided is the connection
// information that was obtained in the Delivered event.
//
callControl.sent = "<AnswerCall" +
" xmlns='http://www.ecma.ch/standards/ecma-323/csta/ed2'>" +
"<callToBeAnswered><callID>" +
callID.value + "</callID><deviceID>" +
deviceID.value + "</deviceID></callToBeAnswered></AnswerCall>";
}

```

```

function ccCleanUp() {
    callerID.value = ""; transferTarget.value = ""; callID.value = "";
    recNumber.Stop(); recYesNo.Stop(); ...
}

function ccHangup() { // clearing a connection
    //
    // The ClearConnection service (ECMA-323, 15.1.8) is used to clear
    // a connection. In this example, this is used when the transfer is
    // unable to be completed.
    //
    callControl.sent = "<ClearConnection " +
        "xmlns=http://www.ecma.ch/standards/ecma-323/csta/ed2"> " +
        "<connectionToBeCleared> <callID>" +
        callID.value + "</callID><deviceID>" +
        deviceID.value + "</deviceID></connectionToBeCleared"> " +
        "</ClearConnection>";
}

function ccError() {
    //
    // The ccError function is called to handle any failure responses to
    // ECMA-323 service requests. If there was an error starting a
    // monitor, the application logs an error. If there was an error
    // response to the SingleStepTransfer service, a message is played
    // for the caller and the connection is cleared.
    //
    var request = callControl.sent.substr(1, 7);
        // read the first 7 characters of service requested
    if (request == "Monitor") { // error starting a monitor
        logMessage("ccError", callControl.sent);
    } else if (request == "SingleS") { // error in transfer
        tryAgain.Start();
        ccHangup();
    } // feel free to handle other errors here
}
--> </script>

```

```

Putting it all together:
<html>
...
<body>
// data section here
<div xmlns="http://www.saltforum.org/2002/SALT" style="visibility:hidden">
    // put the speech objects here
</div>
// speech event handlers here
// call control section here
// finally, when the page is loaded...
<script>
ccStartListening();
</script>
</body>
</html>

```

8 CCXML/CSTA XML プログラム例 (CCXML/CSTA XML Programming Example)

次の例は、CCXML を用いた ECMA-323 の利用方法について説明する。前の例で利用された ECMA-323 サービスとイベントがどのように CCXML によって作成され、さらされるかを示す。

```
<?xml version="1.0" encoding="UTF-8"?>
<ccxml version="1.0"
xmlns="http://www.w3.org/2002/09/ccxml"
xmlns:ccxml="http://www.w3.org/2002/09/ccxml"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta/ed2">
  <var name="state0"/>
  <!-- CSTA Vars -->
  <var name="callID"/>
  <var name="deviceID"/>
  <eventhandler statevariable="state0">
    <transition event="ccxml.loaded">
      <!-- inline ECMA-323 -->
      <csta:MonitorStart>
        <monitorObject>
          <deviceObject>
            9999
          </deviceObject>
        </monitorObject>
      </csta:MonitorStart>
      <!-- ECMA-323 as a external event source -->
      <script>
        var monitorRequest = "<MonitorStart"
          + " xmlns='http://www.ecma.ch/standards/ecma-323/csta/ed2'"
          + "<monitorObject><deviceObject>" + 9999
          + "</deviceObject></monitorObject></MonitorStart>";
      </script>
      <send dest="http://www.csta-webservice.com/ecma323" name="sendRequest"
        namelist="monitorRequest" />
    </transition>
    <transition event="csta.DeliveredEvent" name="evt">
      <!-- Simple CCXML style shortcut -->
      <accept/>
      <!-- save off event props -->
      <!-- simple object mapping -->
      <assign name="callID" expr="evt.connection.callID"/>
      <assign name="deviceID" expr="evt.connection.deviceID"/>
      <!-- DOM API -->
      <script>
        callID.value = evt.selectSingleNode("./connection/callID").value;
        deviceID.value =
          evt.selectSingleNode("./connection/deviceID").value;
      </script>
      <!-- Big inline ECMA-323 style request -->
      <AnswerCall xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
        <callToBeAnswered>
          <callID> <ccxml:value expr="evt.connection.callID"/> </callID>
          <deviceID> <ccxml:value expr="evt.connection.deviceID"/>
          </deviceID>
        </callToBeAnswered>
      </AnswerCall>
    </transition>
  </eventhandler>
</ccxml>
```

```

<!-- ECMA-323 as a external event source -->
<script>
  var answerRequest = "<AnswerCall"
    + " xmlns='http://www.ecma.ch/standards/ecma-323/csta/ed2'"
    + "<callToBeAnswered><callID>" + evt.connection.callID
    + "</callID><deviceID>" + evt.connection.deviceID +
    "</deviceID></callToBeAnswered></AnswerCall>";
  </script>
  <send dest="http://www.csta-webservice.com/ecma323" name="sendRequest"
  namelist="answerRequest" />
</transition>
<transition event="csta.EstablishedEvent" name="evt">
  <dialogstart src="hello.vxml"/>
</transition>
<transition event="dialog.exit">
  <disconnect/>
  <!-- Big inline ECMA-323 style request -->
  <ClearConnection xmlns="http://www.ecma.ch/standards/ecma-323/csta/ed2">
    <connectionToBeCleared>
      <callID><ccxml:value expr="callID"/></callID>
      <deviceID><ccxml:value expr="deviceID"/></deviceID>
    </connectionToBeCleared>
  </ClearConnection>
  <!-- ECMA-323 as a external event source -->
  <script>
    var clearRequest = "<ClearConnection"
      + " xmlns='http://www.ecma.ch/standards/ecma-323/csta/ed2'"
      + "<connectionToBeCleared><callID>" + evt.connection.callID
      + "</callID><deviceID>" + evt.connection.deviceID +
      "</deviceID></connectionToBeCleared></ClearConnection>";
    <send dest="http://www.csta-webservice.com/ecma323" name="sendRequest"
    namelist="clearRequest" />
  </script>
</transition>
<transition event="csta.ConnectionClearedEvent" name="evt">
  <exit/>
</transition>
<transition event="error.*" name="evt">
  <log expr="an error has occurred (' + evt.error + ')" />
  <exit/>
</transition>
</eventhandler>
</ccxml>

```

9 CSTA 呼制御機能 (CSTA Call Control Features)

いくつかの ECMA-323 準拠プラットフォームは、より高度な呼制御機能を提供するかもしれない。この章は、CSTA 呼制御機能の完全なセットの概念を提供する。ECMA-269 についての機能の詳細仕様は ECMA-269 を参照のこと。

9.1 サービス (Services)

サービス名	説明
Accept Call	Offered Call を鳴動あるいは、呼出中状態の分配モードに入れるよう状態を遷移させる。
Alternate Call	既存呼を保留して、次に、同じデバイスで以前保留されたか呼出された呼を保留解除する。
Answer Call	鳴動している、待ち行列に入ったもしくはデバイスに対して着呼している呼に応答する。
Call Back Call-Related	本来の着呼デバイスが発呼デバイスへの呼返し要求をコンピューティング機能に許可する。
Call Back Message Call Related	着呼デバイスが発呼デバイスを呼ぶという、あらかじめ定義されたメッセージ要求することを、スイッチング機能に指示することをコンピューティング機能に許可する。
Camp On Call	デバイスが利用可能になるまで、話中デバイスの呼を待ち行列に入れる。
Clear Call	特定の呼に関連するすべてのデバイスを解放する。
Clear Connection	呼から特定のデバイスを解放する。
Conference Call	会議デバイスで既存の保留呼と別の通話中の呼の会議を提供する。2つの呼を会議デバイスで1つの呼にマージする。
Consultation Call	デバイスで既存の通話中の呼を保留状態にし、同じデバイスから新しい呼を発信させる。
Deflect Call	別のデバイスへ呼を着信転送する。
Dial Digits	すでに起動された呼に対して電話番号をダイヤルする。
Directed Pickup Call	指定された呼をピックアップする (移動して、指定された呼出中または待ち行列にいる呼と接続する)。
Group Pickup Call	指定されたピックアップ・グループから呼をピックアップする (移動して、ピックアップ・グループ内のいくつかの呼出中の呼が他のデバイスに接続する)。
Hold Call	特定の接続を保留状態にする。
Intrude Call	話中の着側デバイスにおいて、呼に発側デバイスを加えることをコンピューティング機能に許可する。
Join Call	既存の呼にデバイスを結合させることをデバイスに代わって要求することをコンピューティング機能に許可する。
Make Call	2つのデバイス間の呼を確立する。
Make Predictive Call	2つのデバイス間の呼を確立する。着側デバイスが鳴動、もしくは呼に応答した後にはのみ、その呼が発側デバイスに送られる。
Park Call	指定されたデバイスで呼をパークする (移動して、接続された呼を別のデバイスの待ち行列に入れる)。
Reconnect Call	既存の接続を切断し、次に、同じデバイスで以前保留された接続と接続する。
Retrieve Call	以前に保留状態に置かれた呼と接続する。
Single Step Conference Call	既存の呼に対してデバイスを加える。
Single Step Transfer Call	既存の呼にあるデバイスを別のデバイスと入れ替える。
Transfer Call	保留された呼を協議パーティに転送する。

9.2 イベント (Events)

イベント名	説明
Bridged	非通話中モード(つまり待ち行列状態)に置かれている、共有されたブリッジ・デバイス構成でのアピランスを示す。
Call Cleared	全てのデバイスが既存の呼から切り離されたことを示す。
Conferenced	会議デバイスが既存の呼でそれ自身もしくは別のデバイスを会議状態においたことを示す。
Connection Cleared	呼のデバイスが切断されたか呼から途中放棄されたことを示す。
Delivered	呼が鳴動中あるいは、呼出中状態の分配モードにあるデバイスへ送られようとしていることを示す。
Digits Dialed	デバイスから呼を発信もしくは機能を試みようとし、ダイヤリング・シーケンスの一部が完了したことを示す。
Diverted	呼がデバイスから着信転送されたことを示す。
Established	デバイスが呼に対して応答または接続されたことを示す。
Failed	呼が完了することができない、そして/または、コネクションが失敗状態になったことを示す。
Held	既存の呼が保留されたことを示す。
Network Capabilities Changed	公衆またはプライベート・ネットワークにおける呼の進行の間に、そのシグナリング能力(つまり相互ネットワーキング)を修正するような状況が発生したことを示す。
Network Reached	呼が網インタフェース・デバイス(例えば、トランク、公衆網)を用いて、外部ネットワークに接続されたことを示す。
Offered	呼がデバイスで引渡し前状態(例えば、鳴動や呼出音が聞こえる前)にあることを示す。
Originated	デバイスから呼を発信しようとしていることを示す。
Queued	呼が待ち行列に入ったことを示す。
Retrieved	以前に保留された呼が解除されたことを示す。
Service Initiated	デバイスがサービスのためにオフフックされた、またはオフフックするように促されたことを示す。
Transferred	既存の呼が別のデバイスに転送され、転送デバイスが呼から抜けたことを示す。