

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2012-194980

(P2012-194980A)

(43) 公開日 平成24年10月11日(2012.10.11)

(51) Int.Cl. F I テーマコード (参考)
G 0 6 F 1 3 / 0 0 (2006.01) G 0 6 F 1 3 / 0 0 5 4 0 A 5 B 0 8 4

審査請求 有 請求項の数 6 O L (全 81 頁)

(21) 出願番号 特願2012-85792 (P2012-85792) (22) 出願日 平成24年4月4日 (2012.4.4) (62) 分割の表示 特願2011-129692 (P2011-129692) の分割 原出願日 平成11年4月7日 (1999.4.7) (31) 優先権主張番号 09/057,394 (32) 優先日 平成10年4月8日 (1998.4.8) (33) 優先権主張国 米国 (US) (特許庁注：以下のものは登録商標) 1. J A V A	(71) 出願人 591112522 株式会社 A C C E S S 東京都千代田区猿樂町二丁目8番8号 (74) 代理人 100078880 弁理士 松岡 修平 (74) 代理人 100169856 弁理士 尾山 栄啓 (74) 代理人 100148895 弁理士 荒木 佳幸 (72) 発明者 デ・ボーア, アダム アメリカ合衆国カリフォルニア州9450 1, アラメダ, マリナ・ビレッジ・パーク ウェイ 909
--	---

最終頁に続く

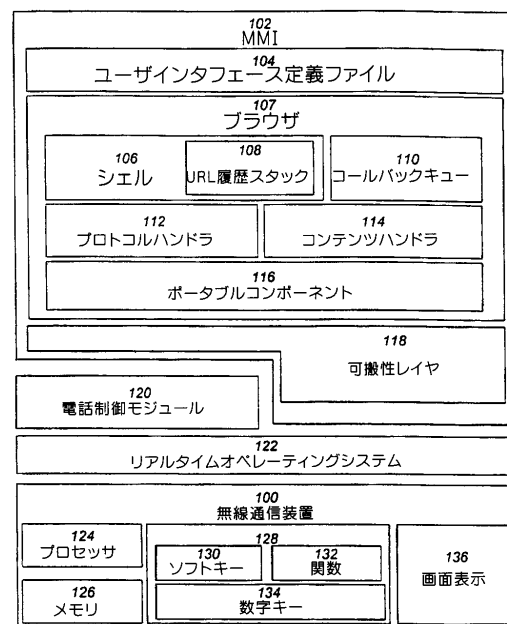
(54) 【発明の名称】 マークアップ言語ベースのマンマシンインタフェースを備えた無線通信装置

(57) 【要約】

【課題】 携帯電話のようにポインティングデバイスを持たない装置において、ユーザに提供するナビゲーション能力を向上させる。

【解決手段】 無線通信装置において使用されるコンピュータプログラムであって、プロトコルコンポーネントおよびデータコンポーネントを有するURLを受信し、プロトコルコンポーネントに従ってデータコンポーネントをプロトコルハンドラに提供し、取得したコンテンツをコンテンツハンドラに提供するシェルと、URLを受信し、データコンポーネントによって特定されるコンテンツを取得してシェルに提供する、またはデータコンポーネントによって特定されるコマンドを実行する、複数のプロトコルハンドラと、取得されたコンテンツを受信し、無線通信装置の画面表示に出力するために処理する、複数のコンテンツハンドラと、を備えることを特徴とする。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

無線通信装置において使用されるコンピュータプログラムであって、該無線通信装置は、メモリと、画面表示と、コンピュータプログラムを実行し、前記無線通信装置の動作を制御するプロセッサと、を備え、前記コンピュータプログラムは、

プロトコルコンポーネントおよびデータコンポーネントを有するURLを受信するためのシェルであって、前記データは、実行されるコマンドまたは取得されるコンテンツを特定し、前記シェルは、前記プロトコルコンポーネントに従って前記データコンポーネントをプロトコルハンドラに提供し、処理のため取得したコンテンツをコンテンツハンドラに提供するシェルと、

URLを受信し、データコンポーネントによって特定されるコンテンツを取得して該取得したコンテンツを前記シェルに提供する、またはデータコンポーネントによって特定されるコマンドを実行する、各々が前記シェルと通信可能に接続される複数のプロトコルハンドラと、

取得されたコンテンツを受信し、前記無線通信装置の画面表示に出力するために該取得したコンテンツを処理する、各々が、前記シェルと通信可能に接続される複数のコンテンツハンドラと

を備える、コンピュータプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、無線通信装置用のマンマシンインタフェースに関し、特に、マークアップ言語から構築されたマンマシンインタフェースに関する。

【背景技術】

【0002】

無線通信装置は、個人通信の必要性によりますます一般的になっている。これら装置としては、例えば、セル式電話、英数字ページャ、「パームトップ」コンピュータ、およびパーソナルインフォメーションマネージャ（PIM）、および他の小型の主にハンドヘルド式の通信装置およびコンピューティング装置が含まれる。無線通信装置は、その特徴において相当に成熟しており、現在では、電話の呼び出しのような基本的な固定通信機能だけでなく、電子メール、ファクシミリの送受信、インターネットアクセス、およびワールドワイドウェブの閲覧等、より進んだ通信機能もサポートする。

【0003】

一般に、無線通信装置は、各種のハンドセット機能および基地局への通信接続を管理するソフトウェアを有する。すべての電話通信機能を管理するソフトウェアは、通常、電話スタックと呼ばれ、画面表示を管理するとともに、キー押下によるユーザ入力进行处理するソフトウェアはマンマシンインタフェース、または「MMI」と呼ばれる。MMIは、無線通信装置のソフトウェアの最上であり、かつ最も目に見えるレイヤである。

【0004】

無線通信装置は概して非常に望ましくかつ小さなフォームファクタに達しているため、成功する装置の主な決定要素は、その機能セットおよび使いやすさにあると思われる。したがって、使いやすく、かつ機能の豊富なセットを有する無線通信装置を迅速に設計し、テストして、配送する能力（互いに相反することが多い属性）が、成功する製品の性能に対して極めて重要である。

【0005】

しかし、無線通信装置は、ノートブックおよびデスクトップのコンピュータ等、同様な通信特徴を有しうる、より大型のプロセッサベースのシステムでは生じない種々のより挑戦的な設計および実施問題を提示する。これら設計における挑戦としては、ユーザインタフェースの設計、特定サービスの運営者のための装置のカスタマイズ、インターネットお

10

20

30

40

50

よびワールドワイドウェブアクセスと他の通信機能性との統合、およびソフトウェア開発プロセスが含まれる。

【0006】

ユーザインタフェース設計制約

デスクトップおよびノートブックのコンピュータとは異なり、無線通信装置は、非常に小さな画面表示サイズを必要とするフォームファクタを有する。通常、デスクトップコンピュータは、少なくとも14インチの画面サイズであり、典型的に640×480から1024×768ピクセル間の解像度を有するディスプレイを備える。それに対して、無線通信装置は通常、25×25mmから80×120mmの間の画面サイズ、すなわちデスクトップまたはノートブックの画面サイズの約3～8%の画面サイズ、および90×60から120×120ピクセルの間の解像度を有する。直接の結果として、無線通信装置のユーザインタフェース設計は、テキスト、イメージ、アイコン等の表示にほんのわずかな表示エリアを使用しながら、電子メール、ファクシミリ、およびウェブ閲覧等、デスクトップコンピュータと本質的に同じ特徴に対するアクセスを提供しなければならない。フォーム等、従来のウェブコンテンツは、従来のデスクトップコンピュータのより大きな画面サイズを想定しているため、これらの特徴を提供するようユーザインタフェースを構築するというこの問題は、ウェブベースのコンテンツを取り扱う場合に特に顕著である。このようなフォームを無線通信装置の小さな画面上に表示すると、結果としてごちゃ混ぜになり、コンテンツの使用が困難になる。

【0007】

無線通信装置の別のユーザインタフェース制約は、ユーザが使用可能な入力セットが厳しく制限されていることである。従来のデスクトップまたはノートブックコンピュータは、コンピュータマウス、トラックボール、ジョイスティック等のカーソルベースのポインティングデバイスと、完全なキーボードとを備えている。これにより、スクロールバーのクリックおよびドラッグ、ハイパーテキストリンクのクリック、およびHTMLフォーム等、フォームのフィールド間のキーボードによるタブ送りでのウェブコンテンツのナビゲートが可能になる。無線通信装置は、通常アップおよびダウンキー、および1から3のソフトキーという、非常に限られた数の入力を有する。

【0008】

したがって、無線通信装置の限られた画面解像度および入力機能性を考慮した、ウェブコンテンツとのユーザインタフェースのカスタマイズおよび使用を可能にする、無線通信装置のMMI用のソフトウェアアーキテクチャを提供することが望ましい。

【0009】

インターネット／ウェブ機能と電話通信との統合

インターネットおよびワールドワイドウェブの出現に伴って、最高性能の無線通信装置は、完全なインターネットアクセスと、ワールドワイドウェブを直接閲覧する能力を提供する。現在の装置は、厳密なモード式インタフェースを通して、インターネットおよびワールドワイドウェブのアクセスを提供する。厳密なモード式インタフェースでは、ユーザが電話をかける、格納した電話帳にアクセスする、ファクシミリを送信する等のために、無線通信装置を使用して本来の通信モードまたは閲覧モードを選択しなければならない。

「閲覧モード」において、ユーザは電話をかけるために電話番号をダイヤルすることができない。同様に、電話通信モードにおいて、ユーザはウェブサイトにはアクセスできない。このため、ユーザは、閲覧中に取得したウェブコンテンツのアイテムをユーザの電話帳または電子メールメッセージ中に組み込むなど、ウェブコンテンツをダウンロードして、それを電話機能のコンテキストで取り扱うことが可能である継ぎ目のない様式で無線通信装置を動作することができない。

【0010】

したがって、インターネットおよびワールドワイドウェブアクセス特徴が、電話通信および無線通信装置の他の制御と継ぎ目なく統合されて、ユーザがいつでも無線通信装置のいかなる特徴にもアクセスすることができるMMIを提供することが望ましい。

【0011】

MMIのソフトウェア工学

通常、MMIは、電話制御機能を管理するより大きなコードにおけるモジュールとして実施される。MMIは、残りの電話制御ソフトウェアと同じコンピュータ言語でコード化される。これは、電話制御ソフトウェア全体の作成に用いられたのと同じプログラミングスキルおよびツールを使用せずに、MMIを変更することを困難にしている。換言すれば、MMIにおけるいずれの変更も、基礎をなす電話通信プログラミングの詳細およびコンピュータ言語に精通した熟練のプログラマのサービスを必要とする。さらに、MMIは電話制御ソフトウェアについてのコードの一体化部分であるため、MMIでの新しい変更を実施するということは、電話制御ソフトウェアすべての新しいイメージをコンパイルし、新しいMMI特徴が他のすべてのコードモジュールと互換性があることを保証するために、その結果をテストすることを意味する。簡潔に言えば、MMIソフトウェアの変更によって導入される問題は、潜在的には、ハンドセットを誤作動させたり、他のユーザに対するネットワーク上のサービスを混乱させたりする可能性がある。変更の程度に応じて、電話制御ソフトウェアの任意の部分を変更した結果、バグが生じたり、および／または無線通信装置全体を新しいタイプへの承認が必要となる可能性がある。したがって、MMI機能性の設計および実施を電話制御ソフトウェアの実施から切り離し、それによって製造業者が、特定の顧客のニーズに合うように、MMI設計を迅速かつ安全にカスタマイズすることが可能なソフトウェアアーキテクチャを提供することが望ましい。

10

【0012】

サービス運営者についてのMMIのカスタマイズ：「ブランド化」

無線通信装置産業において、セルラサービスプロバイダ等のサービス運営者は、精力的に各自の無線通信装置製品をブランド化するとともに、ユーザに対して新しい電話通信特徴およびネットワークサービスを提供することで、顧客を引きつけ、かつ保持することに関心を抱いている。これらの中で重要なことは、上述したように、ボイスメール、電子メッセージング、インターネットアクセス等、ユーザに対する付加価値サービスである。「ブランド化」とは、記章、ロゴ、または他のしるしを無線通信装置のMMIの中に埋め込むこと、およびそれを顧客に対して該サービス運営者から発せられたものであることを識別するその特徴のことである。

20

【0013】

したがって、通常、基本的なハードウェアコンポーネントのみを提供する無線通信装置の製造業者は、サービス運営者に、ソフトウェアプログラミングによりこれらの特徴およびサービスを無線通信装置と統合するための方法を提供するとともに、特徴をブランド化するための機構を提供しなければならない。鍵となる問題は、これらのサービスは、それぞれの機能性および要件が異なっている必要があることであり、ユーザに対してサービスおよび特徴の現行のアレイを提供するというタスクは、困難なものである。

30

【0014】

無線通信装置の製造業者は、従来、無線通信装置を販売しているサービス運営者毎に、各自の通信サービスに関連して無線通信装置制御ソフトウェアの特別バージョンを作製することで、この問題に対処してきた。無線通信装置の特別バージョンはそれぞれ、装置の製造業者のブランド化、運営者のブランド化、およびサービス運営者がサポートする特徴およびサービスが何であれ、それに対するサポートを含む。これらの各バージョンは、新しい特徴またはサービスが消費者に提供される際に、テストし、メンテナンスし、かつ変更すべきである異なるソフトウェアとなる。これは、ソフトウェア開発費用およびメンテナンス問題をかなり増大させる。さらに、無線通信装置の製造業者がサービス運営者にMMIおよび電話制御ソフトウェアのソースコードを提供しない限り、無線通信装置の製造業者が、サービス運営者のブランド化およびMMI設計要件に直接関与する必要がある。したがって、無線通信装置の製造業者が各サービス運営者に対して単一体の電話制御ソフトウェアを提供することを可能にするとともに、各サービス運営者が独立してかつ無線通信装置の製造業者の助力なしで、無線通信装置についてのMMIを設計、実施、かつブラ

40

50

ンド化することを可能にする、MMIについてのソフトウェアアーキテクチャを提供することが望ましい。

【発明の概要】

【発明が解決しようとする課題】

【0015】

本発明は、無線通信装置にマークアップ言語をベースとしたMMIを設けることで、従来の無線通信装置の各種制限を克服する。マークアップ言語は、コンテンツの論理的コンポーネントを示すコンテンツ自体に予め定義されたシンボル、ページまたは画面におけるコンテンツのレイアウトについての命令、またはコンテンツの表示、操作、または変更に関する責任を負うある自動システムによって解釈可能な他のデータを含むことにより、ページのコンテンツまたは画面表示を定義することを可能にするコンピュータプログラミング言語である。特に、本発明は、携帯電話のようにポインティングデバイスを持たない装置において、ユーザに提供するナビゲーション能力を向上させることを目的とする。

【課題を解決するための手段】

【0016】

1つの態様において、マークアップ言語で定義されたユーザインタフェースを備える無線通信装置が提供される。これを作用させるために、本発明は、ユーザインタフェースにおける電話通信機能のユーザ選択に応答して、無線通信装置の電話通信制御およびインターネット上に配置されたコンテンツに関連するデータアイテムのユーザ選択に応答して、ハイパーテキスト転送プロトコル（HTTP）を介したインターネットアクセスの双方を提供するために無線通信装置が使用するマークアップ言語ブラウザを含む。

【0017】

1つの実施形態において、無線通信装置の通信制御および他の機能は、マークアップ言語で書かれた様々なユーザインタフェースページにおいて定義される。各制御機能は、ユニフォームリソースロケータ（URL）に関連づけられるか、またはそれによって起動される。URLは、データアイテムを取得するためのプロトコルおよびいずれのデータアイテムをフェッチまたは操作すべきかを指定するデータアイテムである。ユーザインタフェースページは、無線通信装置のローカルメモリに格納され、ブラウザによってフェッチされ、ブラウザがそれらをデコードして、適切なユーザインタフェース要素を表示する。また、ブラウザは、マークアップ言語ページまたは遠隔に格納された他のコンテンツを、ワールドワイドウェブ等の通信ネットワークを介してかかるページにアクセスすることで、モードレスにフェッチし、かつ同様にこれら遠隔的にアクセスしたページをデコードおよび表示することができる。ユーザインタフェースページが表示されると、ユーザが選択した制御機能は、URLまたはコマンドデータをブラウザに渡す。ブラウザは、受け取ったURLまたはコマンドに応じた通信機能を発効させる。

【0018】

ブラウザは、好ましくは、電話通信プロトコルハンドラ、ローカルファイルプロトコルハンドラ、および遠隔ファイルプロトコルハンドラを含む複数のプロトコルハンドラと、マークアップ言語ハンドラを含む複数のコンテンツハンドラとを含む。電話通信プロトコルハンドラは、電話ダイアリングおよび応答等、電話通信制御特徴についてのURLをデコードし、無線通信装置のハードウェアを制御する電話通信制御ソフトウェアの基礎をなす機能を起動させる。電話通信制御の表示に必要なURLのいかなるコンテンツも、マークアップ言語コンテンツハンドラに提供され、マークアップ言語コンテンツハンドラは、コンテンツを構文解析して、画面表示上に表示する。マークアップ言語コンテンツハンドラは一般に、すべてのユーザインタフェースページを含むフェッチされたマークアップ言語ページをいずれも表示すること、および、フォームおよび他の入力手段を介してこれらのページへのユーザ入力を受け取ることに對して責任を負う。

【0019】

マークアップ言語ハンドラは一般に、2つのソース、すなわちローカルファイルプロトコルハンドラおよび遠隔ファイルプロトコルハンドラからコンテンツを受け取る。遠隔フ

ファイルプロトコルハンドラは、ワールドワイドウェブ上のコンテンツにアクセスするためにURLをデコードし、無線通信装置の画面表示にコンテンツを出力するために、ウェブページ、フォーム、アプレット等のフェッチしたコンテンツをマークアップ言語ハンドラに提供する。1つの適した遠隔ファイルプロトコルハンドラは、HTTPを実施する。ローカルファイルプロトコルハンドラは、ローカルユーザインタフェースファイルにアクセスするためにURLをデコードし、かかるコンテンツをマークアップ言語コンテンツハンドラに提供する。MMIの好ましい実施形態では、ユーザインタフェースはハイパーテキストマークアップ言語即ち「HTML」で定義され、かつブラウザは、HTMLで定義され、整えられたウェブコンテンツおよびユーザインタフェースを双方とも表示するHTMLコンテンツハンドラを含む。

10

【0020】

無線通信装置のMMIを定義するために、マークアップ言語を使用することで、従来のMMIソフトウェアアーキテクチャを凌ぐ多数の利点をもたらされる。まず、マークアップ言語の使用により、インターネットおよびワールドワイドウェブアクセスを無線通信装置の電話通信および他の特徴と完全かつ継ぎ目なしに統合することができる。MMIは、すべての機能的画面の表示に、HTML等のマークアップ言語を用いるため、ワールドワイドウェブコンテンツ（これもまたHTMLで書かれる）は、無線通信装置の他の特徴と同じ外観を有する。より具体的には、MMIのページは、ウェブコンテンツがアクセスされるのと全く同様に、URLを用いてアクセスされる。無線通信装置は、機能的ページを表示する場合にはローカルURLにアクセスし、ウェブコンテンツを表示する場合には、ウェブコンテンツを取得するために、ウェブサーバとの接続を自動的に開始する。このため、マークアップ言語ベースのMMIにより、ユーザが、従来の装置のように無線通信装置を電話通信モードと「閲覧」モードの間で切り替える必要なく、いつでもインターネットおよびワールドワイドウェブにアクセスできるようにするモードレスなユーザインタフェースが可能になる。

20

【0021】

マークアップ言語ベースのMMIのさらなる利点として、ワールドワイドウェブからのウェブページ、フォーム等のウェブコンテンツにアクセスでき、かつ電話通信、メッセージング、および無線通信装置の他の非インターネットベースの特徴に直接統合することができることがある。例えば、好ましい実施形態において、無線通信装置は、電話番号および氏名を格納した電話帳を有する。従来では、ユーザが無線通信装置のキーパッドを使用してこれらのエントリを手動でキー入力する必要があった。本発明によるMMIを使用した無線通信装置では、ユーザが容易に情報を直接電話帳に格納できるようにするHTMLを含みうる、ワールドワイドウェブ上の電話ディレクトリに単にアクセスするだけで、ユーザは、エントリを電話帳に追加することができる。

30

【0022】

また、マークアップ言語の使用により、特定の無線通信装置についてのMMI作成のソフトウェア工学プロセスの複雑性が低減される。第1に、本発明のMMIはマークアップ言語をベースとしているため、Cまたは他の低レベル言語プログラミングに熟練することをプログラマに要求する従来のMMIとは異なり、完全に形の整ったユーザインタフェースを設計するために、非常に限られた量のプログラミングスキルのみを必要とする。ユーザインタフェースの編集および変更には、完全なアプリケーションプログラミング環境ではなく、単純なマークアップ言語およびイメージ編集ツールのみが必要である。第2に、本発明のマークアップ言語ベースのMMIを使用することで、電話通信制御ソフトウェア全体を再びコンパイルし、かつパッケージ全体を再テストして認証する必要なく、MMIのいずれの特徴も変更することができる。MMIは基礎をなす電話通信制御およびエアインタフェーススタックから切り離されているため、個々に変更あるいは追加されるユーザインタフェースページをテストする必要があるだけである。これは、市場に出すまでの時間を短縮し、かつ新しい特徴およびサービスが利用可能になる際の、MMIの設計、メンテナンス、および変更における容易さを増大させる。また、ユーザインタフェースを変更

40

50

し、その変更をテストするための時間の低減は、従来のMMIを用いた場合よりも短時間でより多くの異なるバージョンのプロトタイプを作成することができることを意味するため、所与のセットのユーザ要件および特徴について最良のユーザインタフェース設計に対する設計探求が促進される。

【0023】

さらに、MMIのユーザインタフェースを作成かつ変更することができる容易さ、および市場に出すまでの時間の短縮により、サービス運営者が、あらゆるターゲット顧客セグメントそれぞれに対して特定の製品ソフトウェアイメージを作成するために装置の製造業者を必要とせずに、特定の顧客セグメントをターゲットにした無線通信装置を迅速に製造することができる。例えば、サービス運営者は、同じ無線通信装置ハードウェアおよび電話通信制御ソフトウェアを、それぞれニーズおよび無線通信装置の特徴を使用する能力が異なる会社役員、ティーンエイジャー、および高齢者向けに設計された異なるユーザインタフェースと共に使用してもよい。

【0024】

例えば、ユーザインタフェースのページを定義するためにマークアップ言語を使用すると、任意のページにおいて以下の任意のアイテムを変更することができる。すなわち、タイトルバーの有無およびテキスト、すべての情動的テキスト、オプションリストテキスト、すべての後続画面へのリンク、ソフトキー割り当て、永久にスクロールするバナーメッセージ、バナー広告、およびヘルプテキストである。

【0025】

また、マークアップ言語ベースのMMIの使用により、異なるサービス運営者に対する無線通信装置のブランド化における利点をもたらされる。ユーザインタフェースはマークアップ言語ページで定義されるため、個々のサービス運営者がサービス運営者に特定のロゴ、アートワーク、およびテキストを容易に追加および変更することが可能である。このため、本無線通信装置は、多くのサービス運営者に対して同一の無線通信装置ハードウェアおよび電話制御ソフトウェアを提供することができ、サービス運営者はそれぞれ、従来のように無線通信装置の製造業者が各運営者毎に異なるユーザインタフェースを実施、テスト、および出荷する必要なく、各自独特なユーザインタフェースにより、無線通信装置を迅速かつ容易にブランド化することができる。

【0026】

マークアップ言語ベースのMMIを備えた無線通信装置の提供において、本発明の一つの実施形態によれば、無線通信装置との作動のために、標準HTMLを特に有用にする多くの拡張を用いて、標準HTMLが強化される。標準HTMLは、大部分の無線通信装置において存在しない特徴である、キーボード、ポインティングデバイス、およびフルサイズの表示画面を備えた従来のコンピュータの存在を想定している。HTMLの最も特筆すべき欠陥は、

- ・フォーム要素（例えば、チェックボックス、ラジオボタン）は、マウスなしではナビゲートしにくい。

- ・今日コンテンツに存在するフォームは、ユーザにとって大きすぎて、ユーザが小さな画面上で記入する際にいくつかのコンテンツを維持することができない傾向がある。フォームが n 個のフォームに分割される場合、ユーザの入力は、 $n-1$ 回クライアントとサーバの間で送信され、かつクライアントに返送されて、帯域幅を無駄にする。さらに、シリーズになったより小さなフォームでは、ユーザは、シリーズにおける各フォーム毎にバックキーを押下するため、トランザクションの終了が煩雑になりうる。

- ・ハイパーリンクは、選択するためのマウスおよびページのコンテンツをスクロールするための別個のスクロールバーなしでは、追跡しにくい。追跡するハイパーリンクを選択し、かつ表示をスクロールするために、アップキーおよびダウンキーのみを有する装置において、アップおよびダウンキーにスクロールまたは選択のいずれかを固定して割り当てることは、必要とされるナビゲーション能力の提供に不十分である。

【0027】

ユーザインタフェースを定義する言語として、HTMLは多くの鍵となる特徴を欠いている。

- ・ソフト機能キー、または実際に装置上のあらゆるキーについてのアクションを特定する能力。

- ・選択のポップアップメニューを定義する能力。

- ・氏名および電話番号等、装置に格納したいデータを表示する、あるいは変更する能力。

- ・空白を埋めるために、Cコードを書かずにテンプレートとして画面を設計する能力。

- ・電波を介した(over the air)コンテンツにより、装置がユーザに提示するインタフェースを拡張またはカスタマイズできるようにする能力。

10

【0028】

本発明の一つの実施形態において、マルチパートフォームの設計、ウェブページのナビゲートおよびハイパーテキストリンクの選択の双方に限られた数のキーの使用、URLを使用しての無線通信装置の任意のキー(キーパッドまたはソフトキー)についてのアクションの定義、ソフトキーについてのオプションメニューの作成、およびテキスト、フォーマット、およびユーザインタフェースガジェット(gadget)の条件付き包含を容易にするHTML言語への拡張が提供される。

【0029】

より具体的には、メニューのキーへのバインディングを含む、特定の機能またはアクションをキーパッドの任意のキーに割り当てることができる「キー」タグが提供される。「keymenu」タグは、メニューアイテムの仕様をキーに対するメニューバウンドに含めることができる。「Template」タグおよび「include」タグは、外部HTMLまたは他のデータディレクトリをページのHTMLに置換または挿入可能とする。

20

「help」タグは、設定された期間の後、ページのタイトルバーにわたって自動的にスクロールするhelpストリングの定義を可能にする。条件付きタグは、例えば変数または装置の構成設定をベースとして、ページ内のHTMLデータを条件付きで表示するための式をテストできるようにする。フォームについての「next」メソッドは、状態を維持するためにクライアントとサーバの間で隠しデータを繰り返し伝送する必要なく、マルチパートフォームの状態を維持できるようにする。改良されたナビゲーションメソッドは、別個のタブおよびエンターキーと、スクロールバーがない状態で、無線通信装置のアップおよびダウンキーで、ページのスクロールおよびユーザインタフェースガジェットおよびハイパーリンクの選択の双方を制御できるようにする。

30

【0030】

本発明の一つの側面により、表示画面と、メモリと、プロセッサとを備える無線通信装置であって、該無線装置のメモリは、無線通信装置の設定情報と、設定情報を取得して無線通信装置の設定を行なうためのユーザインタフェースを定義する、マークアップ言語でエンコードされた複数のユーザインタフェース定義ファイルと、該複数のユーザインタフェース定義ファイルをデコードして表示画面上に表示するための、プロセッサによって実行されるマークアップ言語ブラウザと、を記憶しており、設定情報の一部は、所定のセキュリティレベルを有するユーザインタフェース定義ファイルを介してのみ取得されることを特徴とする無線通信装置が提供される。また、上記設定情報の一部は、無線通信装置のPIN番号を含むよう構成されても良い。

40

【0031】

また、本発明の別の側面により、表示画面と、無線通信装置の設定情報を記憶するメモリと、プロセッサとを備える無線通信装置において、メモリにプログラムとして記憶され、プロセッサによって実行されるブラウザであって、設定情報を取得して無線通信装置の設定を行なうためのユーザインタフェースを定義する、マークアップ言語でエンコードされた複数のユーザインタフェース定義ファイルを読み出し、読み出した複数のユーザインタフェース定義ファイルを、デコードして表示画面上に表示し、表示されたユーザインタフェース定義ファイルが所定のセキュリティレベルを有する場合のみ、設定情報の一部を

50

取得可能であることを特徴とするブラウザが提供される。

【0032】

また、本発明の別の側面により、表示画面と、無線通信装置の設定情報を記憶するメモリと、プロセッサとを備える無線通信装置において実施されるコンピュータ実施方法であって、設定情報を取得して無線通信装置の設定を行なうためのユーザインタフェースを定義する、マークアップ言語でエンコードされた複数のユーザインタフェース定義ファイルを読み出すステップと、読み出した複数のユーザインタフェース定義ファイルを、デコードして表示画面上に表示するステップと、表示されたユーザインタフェース定義ファイルが所定のセキュリティレベルを有する場合のみ、設定情報の一部を取得可能とするステップとからなる方法が提供される。

10

【0033】

上述の構成のように無線通信装置の設定をマークアップ言語ベースのユーザインタフェースを通じて提供することで、サービス運営者が、熟練またはソフトウェア開発環境を必要とせず、または従来のMMIのソフトウェア管理問題なしで、新しいユーザインタフェースおよび異なる無線通信装置用のカスタム特徴セットを容易かつ迅速に生成することが可能となる。また、サービス運営者および製造業者が、個々でもまたサービス運営者毎に異なるMMIソフトウェアを作成する必要なく、望むように無線通信装置を迅速かつ効率的にブランド化することも可能となる。さらに、表示されたユーザインタフェース定義ファイルが所定のセキュリティレベルを有する場合のみ、設定情報の一部を取得可能とすることで、機能またはデータへの無許可のアクセスを防止することができる。

20

【0034】

また、本発明の別の側面により、少なくとも1つのソフトキーを備えた無線通信装置を動作するコンピュータ実施方法であって、マークアップ言語で定義された第1のユーザインタフェース定義ファイルを受信するステップと、該第1のユーザインタフェース定義ファイルを構文解析し、ソフトキーの1つとメニューアイテムのメニューの間の関連を格納するステップであって、各メニューアイテムはURLまたはアクションのいずれかに関連する、ステップと、ソフトキーのユーザ選択の受信に応答して、メニューアイテムのメニューを表示するステップと、アクションに関連する表示されたメニューアイテムのユーザ選択に応答して、該アクションを発効させるステップと、URLに関連するメニューアイテムのユーザ選択に応答して、該URLによって指定される無線通信装置の通信機能を発効させるステップとを含む方法が提供される。

30

【0035】

また、本発明の別の側面により、無線通信装置のプロセッサにより、ブラウザの実行で該無線通信装置の動作を制御するためのブラウザプログラム製品であって、該ブラウザは、マークアップ言語で定義された第1のユーザインタフェース定義ファイルを検索する動作と、該第1のユーザインタフェース定義ファイルを構文解析し、ソフトキーの1つとメニューアイテムのメニューの間の関連を格納する動作であって、各メニューアイテムはURLまたはアクションのいずれかに関連する、動作と、ソフトキーのユーザ選択の受信に応答して、メニューアイテムの前記メニューを表示する動作と、アクションに関連する表示されたメニューアイテムのユーザ選択に応答して、該アクションを発効させる動作と、URLに関連するメニューアイテムのユーザ選択に応答して、該URLによって指定される無線通信装置の通信機能を発効させる動作とを実行するブラウザが提供される。

40

【0036】

さらに、本発明の別の側面により、表示画面と、メモリと、少なくとも一つのソフトキーと、プロセッサとを備える無線通信装置であって、メモリは、マークアップ言語でエンコードされた複数のユーザインタフェース定義ファイルと、プロセッサによって実行されるマークアップ言語ブラウザと、を記憶しており、ブラウザは、ユーザインタフェース定義ファイルを検索する動作と、ユーザインタフェース定義ファイルを構文解析し、ソフトキーの1つとメニューアイテムのメニューの間の関連を格納する動作であって、各メニューアイテムはURLまたはアクションのいずれかに関連する、動作と、ソフトキーのユー

50

ザ選択の受信に応答して、メニューアイテムのメニューを表示する動作と、アクションに関連する表示されたメニューアイテムのユーザ選択に応答して、該アクションを発効させる動作と、URLに関連するメニューアイテムのユーザ選択に応答して、該URLによって指定される無線通信装置の通信機能を発効させる動作と、を含むことを特徴とする無線通信装置が提供される。

【0037】

上述の構成により、ソフトキーと関連付けられるURLに関するメニューアイテムに対して、ユーザ選択がなされた場合に、該URLによって指定される無線通信装置の通信機能を発効させることができる。

【図面の簡単な説明】

10

【0038】

【図1】本発明による無線通信装置のトップレベルのソフトウェアおよびシステムアーキテクチャの図である。

【図2】本発明による無線通信装置のサンプルユーザインタフェースページの図である。

【図3】本発明による無線通信装置のマンマシンインタフェースの詳細なソフトウェアアーキテクチャの図である。

【図4】URL履歴スタックの図である。

【図5】受信したURLの取り扱いにおけるシェルの動作のフローチャートである。

【図6a】ユーザインタフェースガジェットに関連するストリング入力を処理する際のHTMLpコンテンツハンドラの動作のフローチャートである。

20

【図6b】ユーザインタフェースガジェットに関連するストリング入力を処理する際のHTMLpコンテンツハンドラの動作のフローチャートである。

【図7】<KEY>および<KEYMENU>のタグを使用してキーメニューを示すHTMLpファイルおよびページの例である。

【図8】<HELP>タグにより、バナーをスクロールしているヘルプテキストを示すHTMLpファイルおよびページの例である。

【図9】テンプレートファイルについての<TEMPLATE>タグの使用を示すHTMLpファイルおよびページの例である。

【図10】電話帳を編集するためのHTMLpファイルおよびページの例である。

【図11】電話帳を編集するためのHTMLpファイルおよびページの別例である。

30

【図12】CHECKEDおよびSELECTED属性を評価するための、式の使用を示すHTMLpファイルおよびページの例である。

【図13】<INC>タグを含んだHTMLを示すHTMLpファイルおよびページの例である。

【図14】<INPUT>タグについてPHONENUMおよびPHONENAME属性を使用したHTMLpファイルおよびページの例である。

【図15】HTMLpコンテンツハンドラにより入力キーを処理するためのプロセスのフローチャートである。

【図16】複雑なフォームについての従来のHTMLファイルおよびページの例である。

【図17a】図17aはHIDDEN入力データを使用した従来の多重フォームアプローチを示すHTMLファイルおよびページの例である。

40

【図17b】図17bはHIDDEN入力データを使用した従来の多重フォームアプローチを示すHTMLファイルおよびページの例である。

【図18a】図18aはNEXTメソッドとともに使用されるマルチパートフォームを示すHTMLpファイルおよびページの例である。

【図18b】図18bはNEXTメソッドとともに使用されるマルチパートフォームを示すHTMLpファイルおよびページの例である。

【図19】<LINKMENU>タグを使用せずに、ハイパーリンクのメニューのデフォルト作成を示すHTMLpファイルおよびページの例である。

【図20】<LINKMENU>タグの使用を示すHTMLpファイルおよびページの例

50

である。

【図 2 1 a】図 2 1 a は電話番号をダイヤルするための各種ユーザインタフェースページを示す。

【図 2 1 b】図 2 1 b は電話番号をダイヤルするための各種ユーザインタフェースページを示す。

【図 2 1 c】図 2 1 c は電話番号をダイヤルするための各種ユーザインタフェースページを示す。

【図 2 1 d】図 2 1 d は電話番号をダイヤルするための各種ユーザインタフェースページを示す。

【図 2 1 e】図 2 1 e は電話番号をダイヤルするための各種ユーザインタフェースページを示す。 10

【図 2 2 a】図 2 2 a はアクティブコールを取り扱うための各種ユーザインタフェースページを示す。

【図 2 2 b】図 2 2 b はアクティブコールを取り扱うための各種ユーザインタフェースページを示す。

【図 2 2 c】図 2 2 c はアクティブコールを取り扱うための各種ユーザインタフェースページを示す。

【図 2 3】「extra」プロトコルの使用を示す HTML p ファイルおよびページの例である。

【図 2 4】内蔵プロトコルと併用されるアイコンおよびイメージの表である。 20

【図 2 5】<DIAL> タグの使用を示す HTML p ファイルおよびページの例である。

【発明を実施するための形態】

【0039】

A. システムおよびソフトウェアアーキテクチャ

次に図 1 を参照して、本発明によるマークアップ言語ベースの MMI 102 を使用する無線通信装置 100 のシステムおよびソフトウェアアーキテクチャを示す。無線通信装置 100 のハードウェアは、プロセッサ 124 と、メモリ 126 と、画面表示 136 と、キーパッド 128 とを含む。メモリ 126 は、データ長期保存用の ROM、RAM、およびフラッシュメモリを含む。ハードウェア特徴の提供に適した無線通信装置 100 は、Nokia Telecommunications、Inc. 製の Nokia 6100 (30 商標) である。

【0040】

無線通信装置 100 は、パワー、メモリ、スレッド（通信接続）、キーパッド入力、およびタイマアクティビティを管理するためのモジュールを含む従来のリアルタイムオペレーティングシステム 122 をメモリ 126 に格納して、実行する。リアルタイムオペレーティングシステム 122 は、標準的なアプリケーションプログラミングインタフェースを提供し、MMI 102 の上位レベルのコンポーネントが無線通信装置 100 の機能性を要求するとともに、データを送受信できるようにする。

【0041】

また、電話をかける、電話を受ける、複数の電話線（適用可能であれば）の管理、テキストメッセージング（適用可能であれば）の管理、電話信号の監視、および他の基本的な電話通信機能を含む、主要な電話制御を提供する電話通信制御モジュール 120 もメモリ 126 に格納され、かつリアルタイムオペレーティングシステム 122 と通信する。電話通信制御モジュール 120 は、エアインタフェースプロトコルを実施する従来の電話プロトコルスタックを含む。電話通信制御モジュール 120 は、これらの特徴にアクセスするため、MMI 102 に対するアプリケーションプログラミングインタフェースを提供する。電話通信制御モジュール 120 およびリアルタイムオペレーティングシステム 122 は、通常、無線通信装置 100 の製造業者によって設けられ、それらの特定の実施は、本発明にとって重要ではない。 40

【0042】

画面表示 136 は、ビットマップ LCD であるか、または同様の表示装置である。画面表示 136 は、通常、コンパクトで携帯可能なハンドヘルド式電子装置に適するように、例えば約 90×60 から 120×120 ピクセル（約 0.28 mm のドットピッチで）と、解像度が非常に限られている。表示技術における進歩の結果、かなり高い解像度の画面表示 136 になるものと予想されるが、そうであっても、無線通信装置の人間工学およびフォームファクタ要件により、ノートブックおよびデスクトップコンピュータの画面表示と比べて比較的小さく（例えば、25×25 mm から 80×120 mm の間）、その結果、かかるより大型の画面表示用に設計されたコンテンツを厳密に同じ様式では表示しない。本発明は、ウェブコンテンツの表示時に、かかる画面表示の使いやすさを増大するように適合される。

10

【0043】

無線通信装置 100 は、無線通信装置 100 の定義された機能（例えば、「送信」、「終了」、および「電源」）にアクセスするための多くの固定機能キー 132 と、数字入力用の（かつ適切にエンコードされた場合、他の文字を入力するための）数字キー 134 と、示されているユーザインタフェース 104 の特定の画面表示に応じて変わる様々な機能性を有するプログラマブルソフトキー 130 とを含むキーパッド 128 を備える。

【0044】

無線通信装置 100 は、本発明に従って作成された MMI 102 のインスタンスをメモリ 126 に格納し、実行する。この MMI 102 は、ユーザインタフェース定義ファイル 104 のセットと、ブラウザ 107 と、ポータブルコンポーネント 116 のセットと、可搬性レイヤ 118 とを含む。ブラウザ 107 は、主要なユーザインタフェース機構をユーザに提供し、通信機能およびインターネット／ワールドワイドウェブアクセスの双方にアクセスできるようにする。ポータブルコンポーネント 116 は、グラフィックスプリミティブのセットと、ファイル格納機能と、ブラウザを種々の無線通信装置 100 上で使用可能とするローカリゼーション特徴とを提供する。可搬性レイヤ 118 は、リアルタイムオペレーティングシステム 122 および電話制御モジュール 120 に対するブラウザ 107 およびポータブルコンポーネント 116 についてのインタフェースを提供する。

20

【0045】

MMI 102 は、シングルスレッドアプリケーションとして実行されるとともに、一般に、任意のリアルタイムオペレーティングシステム 122、電話制御モジュール 120、および十分な ROM、RAM、およびフラッシュメモリ、画面表示 136、および基本サービスを提供する無線通信装置 100 上で実行するように設計される。

30

【0046】

B. ブラウザ

ブラウザ 107 は、無線通信装置 100 の基本的なユーザインタフェースを提供し、かつユーザインタフェース定義ファイル 104 によって定義されるコンテンツ、および通信リンクを介して遠隔ウェブサイトアクセスされるウェブコンテンツ等、遠隔サイトから検索することのできるコンテンツを画面表示 136 に表示する責任を負う。ユーザインタフェース定義ファイル 104 は、HTML または後述する好ましいバリエーション型である HTML p 等のマークアップ言語で書かれたコンテンツファイルおよびコードファイルのセットであり、実行可能埋め込みコードオブジェクトを含みうる。本発明は、HTML に制限されず、SGML または XML 等の他の任意のマークアップ言語、または HTML 拡張の Netscape Communication のセット等、HTML の他の拡張された非標準バージョンと共に動作し、それらを拡張しうる。

40

【0047】

本発明の MMI 102 を使用する無線通信装置を提供するサービス運営者はそれぞれ、装置製造業者によって提供されるユーザインタフェース定義ファイル 104 のある部分を変更して、各自特定のユーザインタフェースを設計するであろうことから、ユーザインタフェース定義ファイル 104 の特定のコンテンツは可変であり、かつ本明細書に記載される例示的なユーザインタフェース画面のいずれとも異なるものと予期される。さらに、M

50

M I 1 0 2 は、いかなるユーザインタフェース定義ファイル 1 0 4 もない状態でサービス運営者に提供され、サービス運営者に所望のこれらファイルを作成する余地を残してもよいことが予期される。したがって、ユーザインタフェース定義ファイル 1 0 4 は、本発明の M M I 1 0 2 によって使用されるが、本発明の必須部分ではない。ユーザインタフェース定義ファイル 1 0 4 は、ユーザに提示されるユーザインタフェースを定義するため、サービス運営者が、ユーザインタフェース定義ファイル 1 0 4 を単純に編集するだけで、無線通信装置 1 0 0 を容易かつ迅速に「ブランド化」できるようにしている。このブランド化では、基礎をなすブラウザ 1 0 7、可搬性レイヤ 1 1 8、またはポータブルコンポーネント 1 1 6 を再度コンパイルする必要がなく、したがって、カスタマイズを非常に容易かつ費用効率的なものとする。それはまた、サービス運営者が、プログラミングスキルおよび従来のコード開発環境を必要とすることなく、単純なマークアップ言語編集ツールを使用してユーザインタフェースをカスタマイズし、ブランド化できることも意味する。

10

【0048】

ワールドワイドウェブの専門用語に従い、本明細書では、個々のユーザのインタフェース画面を「ページ」と呼ぶ。次に図 2 を参照して、ブラウザ 1 0 7 によって画面表示 1 3 6 に表示されるページ 1 3 5 の基本的なレイアウトを示す。各ページ 1 3 5 は、一般に、4 つの基本エリアを有する。常に提示されていることが好ましく、かつ信号強度 2 0 2 およびバッテリー強度 2 0 4、待機メッセージインジケータ 2 0 6 等のアイテムを表示するステータスバー 2 0 0。タイトルバー 2 1 0 は、特定画面の名称を表示する（そのように定義されている場合には）。ステータスメッセージエリア 2 1 2 は、かけているまたは答えている電話番号等、現行のコンテンツに特定のステータスメッセージを提示するために使用することができる。コンテンツエリア 2 1 4 は、ユーザインタフェースページの特定のコンテンツ、例えばメッセージのテキスト、電話帳エントリ等を表示するために使用される。ボタンに沿って（他の場所を使用してもよいが）、ソフトキーラベル 2 1 6 があり、これは、ユーザインタフェース定義ファイル 1 0 4 に設けられたキーの定義に従って動的に更新される。スクロールアロー 2 1 5 は、ユーザがスクロールしている現在の方向（上か下のいずれか）を示す。コンテンツエリア 2 1 4 において、フォーカスを有する、すなわち現行のユーザインタフェースガジェットおよび入力フィールドであるコンテンツの特定のアイテムまたはラインを示すために、フォーカスおよび選択アイコン 2 2 0 をオプションとして使用してもよい。モードインジケータ 2 1 8 は、英字（alpha）、数字、または組み合わせた英数字モードのいずれかのテキストエントリのモードを示す。

20

30

【0049】

画面表示 1 3 6 上に表示されたページまたはコンテンツはいずれも、ユーザインタフェース定義ファイル 1 0 4 からローカルに、またはインターネットまたはワールドワイドウェブから遠隔的に取得することができる。ローカルコンテンツの例としては、電話帳、受信したテキストメッセージ、または送信するために作成中のメッセージ、無線通信装置の構成設定等が挙げられる。本発明の特徴の 1 つは、コンテンツがローカルにまたは遠隔的にフェッチされるかが、取得における遅延（もしあれば）を除いて、広くユーザから隠されていることである。この特徴は、継ぎ目なく統合された、通信機能を有するインターネットおよびワールドワイドウェブアクセスの提示を強化する。

40

【0050】

ユーザインタフェースの特徴の大部分は、U R L（ユニバーサルリソースロケータ）によって起動される。公称的には、U R L は、予め定義されても、あるいは U R L にエンコードされる引数をベースとして、必要に応じて生成されてもよい 1 個のデータを識別する手段である。U R L は、以下の形式をとるストリングである。

protocol:data-identifier[?arguments]

【0051】

protocol コンポーネントは、データの検索に使用すべき通信プロトコルを指定する。ワールドワイドウェブ上にあるコンテンツの場合、プロトコルは通常「h t t p」、すなわち、ハイパーテキスト転送プロトコルであり、ユーザインタフェース定義ファイル 1 0 4

50

のローカルコンテンツは、メモリ 126 に格納されたローカルファイルシステムにおけるデータを取得するために、「file」プロトコルを使用して検索された。本発明は、無線通信装置 100 の動作および構成を制御する多数の追加の新しいプロトコルを提供する。

【0052】

data-identifier コンポーネントは、フェッチするのが望ましいコンテンツの仕様である。現在、ワールドワイドウェブ上のコンテンツの場合、data-identifier は通常 2 つの「/」キャラクタの後にマシン名、もう 1 つの「/」キャラクタ、およびある種のパスが続く形式をとる。

【0053】

引数（もしあれば）は、「?」によって data-identifier から分離され、引数名およびその値からなるペアの形式をとる。「=」キャラクタは、引数名をその値から分離する。複数の引数は、引数の値と次の引数名の間の「&」キャラクタによって分離される。

【0054】

アーキテクチャ的には、ブラウザ 107 は、3 つの主な部分、すなわちシェル 106 と、プロトコルハンドラ 112 と、コンテンツハンドラ 114 とを含む。図 3 は、ブラウザ 107 を含む MMI 102 の詳細なソフトウェアアーキテクチャを示す。

【0055】

シェル 106 は、画面表示 136 の全体の部分の維持、URL の部分を正しいプロトコル 112 および URL 用のコンテンツハンドラ 114 に渡すことによる URL の処理、URL の履歴スタック 108 の維持、およびユーザ入力のルーティングに責任を負う。ユーザ入力のルーティングには、ユーザ入力のキーストローク、または数字入力および文字のフォームへの入力、または電話番号のダイアリング等、処理のための他のターゲットエンティティを適切なコンテンツハンドラ 114 に渡すことが含まれる。

【0056】

プロトコルハンドラ 112 は、シェル 106 から URL を受け取るとともに、該 URL に対応するデータをフェッチして、シェル 106 にいずれのコンテンツハンドラ 114 がデータを受け取るべきであることを命令する責任を負う。場合によっては、URL は、プロトコルハンドラ 112 が実行する責任を負う、無線通信装置 100 の特徴を制御するためのコマンドである。

【0057】

コンテンツハンドラ 114 は、フェッチされた URL を表示し、かつユーザと対話する責任を負う。少なくとも 1 つのコンテンツハンドラ 114 が常に現行のコンテンツハンドラ 114 である。新しい URL がいずれもシェル 106 に提供し戻されるのは、現行のコンテンツハンドラからであり、かつデフォルトで他の入力ターゲットのいずれにも送られないすべてのキーストロークを受け取るのも現行のコンテンツハンドラである。シェル 106 についてはさらに、図 4～図 5 に関して後述する。

【0058】

1. プロトコルハンドラ概観

プロトコルハンドラ 112 は、MMI 102 において 2 つの機能を果たす。第 1 に、プロトコルハンドラ 112 は、データをフェッチしてその型を決定する。型の決定により、今度は該データの表示にいずれのコンテンツハンドラを使用するかを決定する。第 2 に、リアルタイムオペレーティングシステム 122 または電話制御モジュール 120 の埋め込みオブジェクトまたは適切な API にアクセスすることにより、無線通信装置 100 においてコマンドを実行する。プロトコルハンドラ 112 は、そのコマンドの結果を戻して、異なる画面を表示させてもよく、または結果を戻さなくてもよい。好ましい実施形態のプロトコルハンドラ 112 は、以下を含む。

【0059】

内蔵プロトコルハンドラ 112 a は、無線通信装置 100 に内蔵されているアイコンへのアクセスを提供する。

10

20

30

40

50

【0060】

構成プロトコルハンドラ112bは、無線通信装置100の構成設定を得て、設定する。

【0061】

エクストラプロトコルハンドラ112cは、ページにおける埋め込みオブジェクトから、あるいは先にアクセスしたページから渡される引数およびフォームデータへのアクセスを提供する。このプロトコルは、ページを動的に生成する必要なく、データを直接ページに渡せるようにする。

【0062】

ファイルプロトコルハンドラ112dは、ROMおよびフラッシュファイルシステムにおけるローカルコンテンツへのアクセスを提供する。一般に、このコンテンツは、ユーザインタフェースのページを定義するユーザインタフェース定義ファイル104である。ファイルプロトコルハンドラ112dは、POSIX仕様等、ファイルシステムアクセスに適した仕様に従って、当業者により実施されうる。

【0063】

HTTPハンドラ112eは、標準ハイパーテキスト転送プロトコルを使用して、ワールドワイドウェブ上に格納された遠隔コンテンツへのアクセスを提供する遠隔ファイルプロトコルハンドラである。HTTPハンドラ112eは、RFC2068：ハイパーテキスト転送プロトコル—HTTP/1.1に定義される仕様に従って当業者によって実施されうる。遠隔ファイルプロトコルハンドラを実施するために、他の遠隔ファイルアクセス仕様を使用してもよい。

【0064】

メッセージプロトコルハンドラ112fは、メッセージ送信、格納しておいたメッセージの削除、新規または格納しておいたメッセージを読む、または格納したメッセージのロックを含む、各種メッセージング機構を起動する。

【0065】

電話プロトコルハンドラ112gは、電話をかける、電話を受ける、電話帳の表示、電話帳エントリの編集、および新しい電話帳エントリの作成を含む、各種電話機能を起動する。

【0066】

構成プロトコルハンドラ112b（可搬性レイヤ118の一部として図示）は、無線通信装置100の各種構成設定を検索、かつ設定する。

【0067】

a) 基本プロトコルハンドラAPI

一般に、プロトコルハンドラ112は、実行することができる十分に定義された機能セットを有するという点でデバイスドライバと同様であり、各プロトコルハンドラ112は、同じ機能をサポートすることにより、これらの機能を各自の様式でサポートする。各プロトコルハンドラ112は、3つの機能を実施する。

GetURL：URLおよびフェッチすべき該URLを含むページのセキュリティレベルが与えられると、該URLに関連するデータおよびそのデータの特権レベルを戻す。URLが実際に、データへのリファレンスではなくコマンドである場合には、データを戻す必要はない。

BuildURL：完全なURLおよび部分的なURL（protocol:要素がない）が与えられると、完全なURLを戻す。これは、主にHTMLページ内のリファレンスに使用される。

PutURL：URL、URLの下で格納されるべきデータストリーム、および「put」メソッドを含むページのセキュリティレベルが与えられると、セキュリティレベルがデータを格納することができるほど十分に高い場合に、データを格納する。

【0068】

プロトコルハンドラ112によってサポートされる各種埋め込みオブジェクトおよびコ

マンドURLについては、後述する。

【0069】

2. コンテンツハンドラ概観

コンテンツハンドラ114は、フェッチしたURLに対応するページのコンテンツデータをデコードして、コンテンツを表示するか、そうでなければコンテンツを操作することに対して責任を負う。コンテンツハンドラ114は、通常、受信したコンテンツをデコードして、1ページまたはその一部を画面表示136に提示する。コンテンツハンドラ114によっては、メモリ126から、または通信リンクを介して受け取ったデータからページを構築するものもある一方で、無線通信装置100の状態を表示するか、または他のいくつかの管理機能を果たすものもある。ブラウザ107の好ましい実施形態では、コンテンツハンドラ114は以下を含む。

10

【0070】

コールマネージャコンテンツハンドラ114bは、ユーザインタフェース定義ファイル104において定義された着信画面を管理して、ユーザが着信を受信できるようにする。コールマネージャコンテンツハンドラ114bは、埋め込みオブジェクトを通して他の機能性を提供する。

【0071】

HTMLpコンテンツハンドラ114cは、メモリ126における適切なユーザインタフェース定義ファイル104にアクセスすることで、大半のユーザインタフェースを表示する。HTMLpコンテンツハンドラ114cは、HTML3.2互換パーサを含み、HTMLをデコードして、HTMLタグに従ってテキストおよびグラフィックスを表示するために必要なデータ構造およびオブジェクトを作成することができる。さらに、HTMLpコンテンツハンドラ114cは、HTML3.2の特徴および機能性に多くの有利な拡張を提供する後述の、本明細書では「HTMLp」と呼ばれるHTML3.2の変形形式を許容する。

20

【0072】

ユーザインタフェースとしてより良好に機能するため、かつユーザインタフェースの設計において柔軟性を追加するために、HTMLpコンテンツハンドラ114cは、Cまたは他のプログラミング言語で書かれたオブジェクトをHTMLまたはHTMLpページに埋め込んで、無線通信装置100にある異なる型のデータを表示できるようにする。しかし、HTMLpコンテンツハンドラ114cは、標準HTMLパーサと異なり、まず現行ページでユーザが選択したURLを任意の埋め込みオブジェクトに渡して、URLを、ユーザが選択または入力したもので変更するか、あるいはシェル106に与えられ、処理される前に、完全に処理できるようにする。

30

【0073】

好ましい実施形態において、利用可能な埋め込みオブジェクトとしては、以下が含まれる。

【0074】

電話帳オブジェクトは、氏名、関連する電話番号、住所、電子メールアドレス、短縮ダイヤルキー選択、呼び出し音、およびデータの他の定義可能なフィールドのレコードを格納する。電話帳オブジェクトは、レコードのフィールドを取得してセットするためのメソッドを含む。電話帳オブジェクトは、ページに埋め込んで、電話プロトコルの適切なURLによって起動させることができる。

40

【0075】

最近および受け損なった電話リストオブジェクトは、最近かけた電話番号、受けた電話番号、または受けなかった電話番号の継続して更新されるリストを格納する。呼リストオブジェクトは、リスト上の電話番号を削除したり、またダイヤルしたりするためのメソッドを含む。呼リストは、ページに埋め込んで、電話プロトコルの適切なURLによって起動させることができる。

【0076】

50

短縮ダイヤル番号リストオブジェクトは、キーの選択により短縮ダイアリング機能性を提供するように、キーパッド128のキーに電話番号および関連事項のセットを格納する。このリストオブジェクトは、短縮ダイヤル番号をセットし、取得し、ダイヤルするためのメソッドを含む。

【0077】

電話番号索引オブジェクトは、電話帳オブジェクトにアクセスして、入力または選択された氏名または氏名の一部に関連する電話番号を戻す。

【0078】

電話帳氏名索引オブジェクトは、電話帳オブジェクトにアクセスして、入力または電話番号の一部の選択された電話番号に関連する氏名を戻す。

【0079】

受信または送信したテキストメッセージ／英数字ページのリストオブジェクト。メッセージリストオブジェクトは、電子メールまたはショートメッセージサービスメッセージを含むメッセージのリストを格納するとともに、メッセージを見る、格納する、編集する、削除する、および送信するためのメソッドを含む。メッセージリストは、ページに埋め込んで、メッセージプロトコルの適切なURLによって起動させることができる。

【0080】

メインコンテンツハンドラ114dは主に、HTMLpを使用して、無線通信装置100のメイン画面を表示するという点において、HTMLpコンテンツハンドラ114cのフロントエンドである。

【0081】

広告コンテンツハンドラ114aは、無線通信装置100のアイドル時にいずれの広告ページを表示するかを選択し、HTMLpコンテンツハンドラ114cにそれを表示するよう命令する、HTMLpコンテンツハンドラ114cのフロントエンドである。さらに、応答していたまたは失効した広告をオプションとして削除するために、キーストロークおよびユーザインタフェースガジェット起動を中断する。

【0082】

a) 基本コンテンツハンドラAPI

プロトコルハンドラ112のように、コンテンツハンドラ114は、シェル106がそれらと通信するために使用する、十分に定義されたインタフェースを有する。インタフェースは、画面表示135の周囲で、表示されている画面内に定義されたコンテンツエリアがあるという意味において、コンテンツハンドラ114の対話のニーズに合わせられる。各コンテンツハンドラ114がサポートする4つの機能とは、以下のものである。

Content Open：これは、コンテンツハンドラ114に画面表示136のコンテンツエリア214、ソフトキー130およびソフトキーラベル216、タイトルバー210、およびステータスメッセージエリア212の制御を与えるコールである。各コンテンツハンドラ114は、Content Open関数が呼び出されると、以下の4つの情報を受け取る。

1. データをフェッチしたプロトコルハンドラ112により戻されるデータストリーム。これは、表示されるデータである。

2. データをどこに表示すべきかを示す、コンテンツエリア214へのハンドル。

3. コンテンツハンドラ114が以前にこのデータを表示したことがあるか否かを示すフラグ。

4. エンティティがシェル106にURLをフェッチするよう求めたものにより渡されたエクストラデータへのポインタであって、表示されているページにエクストラデータを入力できるようにするポインタ。エクストラデータの使用については、HTMLpの<TEMPLATE>タグ、および「extra」プロトコルに関してさらに後述する。

Content Close：ユーザがページを閉じるよう求めるか、あるいは異なるページを開くよう求めると、現行のコンテンツハンドラ114が閉じる。これは、ページがURL履歴スタック108に維持されているか否か、またはページが該スタックから

10

20

30

40

50

永久に削除されてしまっているかを示すフラグを受け取る。

ContentProcessKey：キーストロークを処理する他に何もない状態において、シェル106は、デフォルトによりキーストロークを現行のコンテンツハンドラ114に送る。現行のコンテンツハンドラ114とは、そのURLがURLスタックの最上部にあるコンテンツを表示しているものである。

ContentActivate：ユーザがソフトキー130を押下するか、あるいはメニューからアイテムを選択すると、そのキーまたはメニューにバインディングされたストリングが、この機能を介して現行のコンテンツハンドラ114に渡される。場合によっては、ストリングが、フェッチすべきシェル106に直接渡すことのできるURLである場合があり、また、ストリングが、ユーザが望むものの指示であり、コンテンツハンドラ114は、自身のアクションを実行するか、あるいは画面表示136上でユーザが選択したアイテムを使用して、シェル106に与えることができるURLを生成する場合もある。例えば、ユーザが電話帳におけるエントリを選択して、「編集」とラベルされたソフトキー130を押下する場合、HTMLpコンテンツハンドラ114cが、ソフトキー130に関連するストリングを取り、それを埋め込み電話帳オブジェクトに渡す。埋め込み電話帳オブジェクトは、埋め込みオブジェクトにおいてユーザがいずれの電話帳エントリを選択したのかに基づいて、該ストリングをシェル106に渡す実際のURLを生成するためのテンプレートとして使用する。

【0083】

コンテンツハンドラ114の特定の機能性については、さらに後述する。

【0084】

3. 制御フロー

ブラウザ107の好ましい実施形態は、他の環境において使用されるイベントループの代わりである、単一のコールバックキュー110の周辺に編成される。MMI102のいずれの部分も、コールバックキュー110に関数要求を追加することで、関数を後に呼び出すよう要求することができる。

【0085】

コールバックキュー110は、それぞれ呼び出すべき関数へのポインタと、ルーチンに渡される2つの32ビット引数を有する、多数の要素を有する。関数ポインタは、システムにおけるいずれのモジュールであってもよい。

【0086】

本質的に、システム全体は、一番上の制御ループを実行する。

1. コールバックキュー110における次のアイテムを呼び出す。

2. その呼び出しが行ったすべての変更により、画面表示136を更新する。このステップは、画面にグラフィックス要素を描画し（例えば、ステータスメッセージおよびアイコンをスクロール、更新する）、キーストロークを表示し、URLに関連する機能または特徴のユーザ起動に応答して、新しいページを表示する。

3. ステップ1に戻る。

【0087】

コールバックキュー110についてのアイテムは、主にキーストローク（上または下）、アクティブコールにおける変化、タイマ経過、および入りテキストメッセージ等の非同期イベントによって追加される。

【0088】

ある延長（protracted）動作もまた、コールバックキュー110を使用して、他のアクション（ユーザ入力等）をなお処理可能としながら、動作を継続する。例えば、動画化したGIFイメージのフレーム読み出しは、2つの概念的な段階に分けられる。

1. 最初のフレームを読み出して、次のフレームの読み出す呼び出しを待ち行列に入れる。

2. 次のフレームの読み出しを試みる。成功した場合には、次のフレームを読み出す呼び出しを待ち行列に入れる。

【0089】

このようにして、動画を含むページを、残りの動画を効率的にバックグラウンドにロードしながら、可能な限り高速に表示することができる。

【0090】

4. シェル

シェル106は、入力キーストロークおよびより下位のレイヤからの他の入力の処理と、かかる入力の適切なプロトコルハンドラ112およびコンテンツハンドラ114への引き渡しを提供する。シェル106の関数リストを、付録Aに提供する。

【0091】

a) キーパッド入力

キーパッド入力は、可搬性レイヤ118からシェル106に自発的に到来する。シェル106は、特に現在表示されているページのユーザインタフェースオブジェクトである、キーストロークを処理することができるエンティティのリストであるキーストロークターゲットリストを維持する。キーストロークが到来すると、シェル106は、該キーストロークをキーストロークターゲットリストの最初のエンティティにShellprocesskeyを介して渡す。そのエンティティが該キーストロークを処理しないと決定する場合には、シェル106を呼び出して、キーストロークをリスト内の次のエンティティに与える(ShellPreviousInput)。リストの初期化時にシェル106によって配置される、リストの最後のエンティティは、キーストロークを現行のコンテンツハンドラ114に分配する。現行のコンテンツハンドラ114は、システム全体でのキーストロークデフォルトを実施するシェル106におけるデフォルト処理ルーチンに渡すよう選択するか、またはキーストロークを所望のように処理することができる。

【0092】

シェル106は、エンティティ(通常、ユーザインタフェースオブジェクト)をキーストロークターゲットリストに登録する(ShellGrabInput)と共に、リストからエンティティを削除する(ShellReleaseInput)関数を含む。

【0093】

b) ソフトキー

特殊な機能を有するキーの一種は、ソフトキー130である。ソフトキー130は、そのラベルが表示画面136におけるページに表示されるとともに、その目的が、ユーザインタフェース定義ファイル104において定義されたパラメータに従って、ページ毎に変わるキーである。シェル106は、通常1から3の間であるが、無線通信装置100に応じて可変であるいくつかのソフトキー130を管理する。各ソフトキー130は、ストリングまたはストリングを指定するメニューアイテムを有するメニューにバウンディングされてもよく、または、URLスタックから1つまたは複数のエントリをポップ(pop: 取り出)しても、または何も行わないように設定されても良い。

【0094】

ソフトキー130、またはストリングにバインディングされたメニューアイテムが起動されると、ストリングが、ContentActivate関数を介して現行のコンテンツハンドラ114に渡される。いくつかの場合では、ソフトキー130にバインディングされるストリングは、フェッチすべきURLである。この例では、URLはコンテンツハンドラ114によってシェル106に渡され、適切なプロトコルハンドラ112およびコンテンツハンドラ114によって処理され(ShellGetURL)、フェッチされる。他の場合では、バインドストリングは、コンテンツハンドラ114がページを変えずに処理するコマンドである。最後に、バインドストリングは、フェッチされる前に、ユーザからのある入力に基づいて、コンテンツハンドラ114によって変更されるテンプレートURL(以下参照)を2つ組み合わせたものであってもよい。

【0095】

上述したように、ContentActivate関数の一部として、HTMLpコンテンツハンドラ114cは、ストリング自体を検査する前に、ユーザ選択のユーザインタ

フェースエンティティにバインディングされたストリングを現行ページにおける任意の埋め込みオブジェクトに渡す。埋め込みオブジェクトによっては、この方法でこのようにして表示しているデータに対して動作するというコマンドを単に取るだけのものもあるが、その一方で、ユーザが選択したデータのある部分に置換される、ストリング内の特殊なエスケープを探してURLをもたらし、それから該URLをシェル106に渡すものもある。HTMLpコンテンツハンドラ114cはまた、ハイパーリンクからURLをただ受け入れるだけというよりもむしろ、受け入れるある特別なコマンドも有する。

【0096】

c) URL処理

上述した制御フローで画面表示136を更新するステップは、新しく表示するページを取得するというコンテキストにおいて行われる場合、URLをShellGetURL関数を介してシェル106に渡すことによって達成される。図4は、URL処理をサポートするために、シェル106によって使用されるURL履歴スタック108を示す。URL履歴スタック108は、LIFOスタックである。各エントリ402は、URL404と、URLを処理する特定のコンテンツハンドラ114についての関数の関数表412へのポインタ406と、「エクストラ」プロトコルによって検索されるか、またはURLについてコンテキストハンドラによって使用されるよう、URLと共に渡されるエクストラデータ408（もしあれば）と、ページが動作している特権レベル414と、URLの優先度416と、シェル106がコンテンツハンドラ114の代わりに維持する状態ブロック418へのポインタと、を含む。

【0097】

図5を参照して、URL処理におけるシェル106の動作のフローチャートを示す。シェル106は、最初の「:」キャラクタの前の、URLストリングの最初の部分を抽出し（500）、それを既知のプロトコルハンドラ112のリストと比較して、URLデータのフェッチに適したハンドラを識別する（502）。プロトコルハンドラ112が見つからない場合、シェル106は、現在、URLスタックの一番上にあるURLについてのプロトコルのBuildURL関数を呼び出すことで、完全なURLを作成し（508）、それを現在一番上にあるURLとフェッチされているURLとに渡す。プロトコルハンドラ112は、渡された相対的なURLの代わりに使用すべき絶対的なURLを戻す。

【0098】

プロトコルハンドラ112が見つかる場合、シェル106は、このプロトコルハンドラ112のGetURL関数を呼び出し（504）、URLの残りを渡す。プロトコルハンドラ112は、URLをフェッチして、戻されるデータの型を示すストリングと、データを解釈すべき特権レベルと、実際のデータを含む（まだデータを提示する必要はないが、ストリームからの読み出しは、到来したままデータを戻さなければならない）Streamへのポインタとを含むContentStream構造へのポインタを戻すものと予期される。いずれのストリームも戻されない場合（506）、シェル106はエラーを戻す（510）。

【0099】

シェル106は、データ型ストリングを既知のコンテンツハンドラ112のリストとマッチングさせる（512）。マッチングがない場合、シェル106は、エラーを戻す（520）。マッチングがある場合（514）、このコンテンツハンドラ114が現行のコンテンツハンドラになる。シェル106は、ソフトキー130、コンテンツエリアサイズ214、タイトルバー210、およびステータスメッセージエリア212を各自のデフォルト状態にリセットする（516）。シェル106は、現行のコンテンツハンドラ114のContentOpen関数を呼び出して、ContentStreamおよびコンテンツ表示エリア214の制御の双方を、コンテンツ表示のために、現行のコンテンツハンドラ114に渡す。

【0100】

オープン呼び出しが成功した場合（522）、シェル106は、URL、現行のコンテ

ンツハンドラ 1 1 4 関数へのポインタ、任意のエクストラデータをスタックに配置して URL 履歴スタック 1 0 8 を更新し、URL を要求しているエンティティに成功を戻し (5 2 4)、そうでなければ、シェル 1 0 6 が URL 履歴スタック 1 0 8 からの先の URL に応答して (5 2 6)、エラーを戻す。

【0 1 0 1】

5. セキュリティ

電波を介して受信するコンテンツは、電話特徴を起動するとともに、電話帳エントリ等の電話データにアクセスすることができることから、本発明は、機能またはデータへの無許可のアクセスを防止するために、セキュリティに関する機構を提供する。

【0 1 0 2】

URL が、Get URL 関数の一部としてプロトコルハンドラ 1 1 2 によってフェッチされると、そのデータがプロトコルハンドラによって特権レベルに割り当てられる。コンテンツが特権ソースから来たものであれば、プロトコルハンドラ 1 1 2 は、最高の特権レベルを割り当てる。例えば、無線通信装置 1 0 0 の ROM 1 2 6 に格納されるユーザインタフェース定義ファイル 1 0 4 からのページはすべて、最高の特権レベルを割り当てられる。フォーマットしたテキストメッセージには、ユーザにより受信されたか、あるいは作成されたかに係わらず、最低の特権レベルが割り当てられる。特権レベルが割り当てられていないコンテンツは、シェル 1 0 6 によって自動的に最低の特権レベルが与えられる。例えば、ワールドワイドウェブからのコンテンツには (そうでなくても、予め割り当てられていない場合には)、最低の特権レベルが与えられる。コンテンツのアイテムの特権レベルは、その URL と共に URL 履歴スタック 1 0 8 に格納される。

【0 1 0 3】

無線通信装置 1 0 0 の選択された関数は、無線通信装置 1 0 0 の製造業者またはサービス運営者のいずれかによって、特権に敏感なように構成される。かかる関数は、呼び出されると、URL 履歴スタック 1 0 8 内のページの URL からコンテンツを要求しているページの特権レベルを決定する。要求しているページの特権レベルが要求されているコンテンツの特権レベルよりも高い場合に、コンテンツがアクセスされる。要求しているページの特権レベルが要求されているコンテンツの特権レベルよりも低い場合には、該関数は、即座にアクセスを拒むか、あるいはユーザに動作を確認することができる。例えば、より低い特権ページが Call Manager を介して電話をかけるよう要求する場合、ユーザには実際の番号をダイヤルするという警告がなされ、ユーザは、電話をダイヤルする前に、要求を確認しなければならない。

【0 1 0 4】

6. コンテンツハンドラ

以下の項では、個々のコンテンツハンドラ 1 1 4 がどのようにして、各コンテンツハンドラ 1 1 4 への API における 4 つの機能を実施するかが概説される。

【0 1 0 5】

a) HTML p コンテンツハンドラ

(1) HTML p API

HTML p コンテンツハンドラ 1 1 4 c は、HTML 3. 2 仕様を用いて、完全な HTML 準拠パーサを提供する。このパーサは、必要に応じて、画面表示 1 3 6 の表示、およびかかるエンティティに関連する各データの格納、ユーザインタフェースエンティティを作成するために、コンテンツのページの構文解析中に、HTML p コンテンツハンドラ 1 1 4 c によって起動される。

【0 1 0 6】

上述したように、各コンテンツハンドラ 1 1 4 は、HTML p Open、HTML p Close、HTML p Activate、および HTML p Process Key という 4 つの関数の外部インタフェースを提供する。HTML p Open および HTML p Close についてここに説明する。理解しやすいように、HTML p Activate および HTML p Process Key については、HTML p の新しいタグの説明後に、後

述する。

【0107】

(a) HTMLpOpen

この関数は、コンテンツのページを表示するため、シェル106によって呼び出され、画面表示136のコンテンツエリア214の制御をHTMLpコンテンツハンドラ114cに与える。上述したように、HTMLpコンテンツハンドラ114cは、シェル106から表示すべきデータのストリームと、コンテンツエリア214へのハンドルと、該コンテンツデータ（ページ）が以前表示されたことがあるか否かを示す表示フラグと、該ページに関連するあらゆるエクストラデータへのポインタと、を受け取る。

【0108】

該関数は、表示フラグから、そのページが以前表示されたことがあるか否かを決定し、またそうであれば、ページの任意の埋め込みオブジェクトがキャッシュされたか否かを決定する。この場合、ページは再度表示されて、任意の埋め込みオブジェクトは、以前の状態を再確立するために呼び出されるRestoreState関数を有する。

【0109】

ページの埋め込みオブジェクトがキャッシュされていなかった、あるいはページが表示されるのは今回が初めてである場合、該ページのコンテンツストリームが基礎をなすHTMLパーサに渡されて、HTMLpコードとして解釈される。パーサは、ウィンドウと、必要に応じてユーザインタフェースエンティティとを作成し、テキストをラッピングし、必要に応じてソフトキー130を更新するとともに割り当てる。ページは、完全に構文解析されると、ユーザに表示される。ユーザインタフェースエンティティを作成する際に、HTMLパーサは、ユーザインタフェース要素（キー132、ソフトキー130、メニューアイテム等を含む）およびこれらのエンティティにバインディングされたURL（ローカルまたは遠隔に係わらず）との間の関連表を確立する。関連は、特定のユーザインタフェースエンティティと、エンティティがユーザによって選択されるか、そうでなければ起動される場合、フェッチされるURLとを識別する。これらの関連は、ページが受け取ったキーストロークを、HTMLpProcessKey関数により続けて処理する場合に、用いられる。他のページをすべて保持するメインウィンドウへのハンドルは、ShellSetStateを呼び出すことにより、そのページについて状態ブロック418としてセットされる。

【0110】

(b) HTMLpClose

この関数は、ユーザが現行ページを閉じるか、または異なるページに切り替える際に呼び出される。シェル106は、該ページをURL履歴スタック108から削除すべきか、あるいはURL履歴スタック108に残りうるかを示すフラグを渡す。ページを削除すべきではなく、かつページがキャッシュ不可能とマークされていない場合、メインページが見えないようセットされ、ユーザから効果的にそれを隠すが、アクティブページとして残す。

【0111】

ページがキャッシュ不可能であるか、またはページがURL履歴スタック108から削除されるべきである場合、ページウィンドウおよびそれに関連するデータ構造が破壊される。ページは、ページが表示されている間に潜在的に変化しうるそれ自身の外部にあるデータを参照する場合に、キャッシュ不可能であるとみなされる。例えば、ページが<INC>（「include」）タグを含むか、構成設定を表示するために、後述する構成機構を使用する場合には、キャッシュ不可能であると考えられる。条件付き<IF>タグ、および<TEMPLATE>タグの使用は、これらのタグについてDYNAMIC属性がセットされているか否か、および%[url]エスケープが<TEMPLATE></TEMPLATE>ブロック内にあるか否かに応じて、ページをキャッシュ不可能にさせる場合もあり、させない場合もある。これらの特徴については、さらに後述する。

【0112】

(2) HTML への拡張: HTML p

本発明は、完全にHTMLと互換性のあるMMI 102を提供する。しかし、コンテンツを単に表示するだけでなく、無線通信装置100のユーザインタフェースをさらに洗練させて、通信機能および小型画面表示136に関してより機能的にするとともに、無線通信装置を装置の製造業者およびサービス運営者の双方により容易かつ迅速にカスタマイズできるようにするHTMLの拡張セットを提供することが望ましい。

【0113】

HTML pを構成する拡張は、以下の通りである。

(a) A. ユーザインタフェース拡張

この項では、ユーザインタフェースを豊かにする、HTML pの拡張について説明する。これらの各種タグは、HTML pコンテンツハンドラ114cがシェル106からコンテンツのページを受け取ったときに、HTML pコンテンツハンドラ114cによってデコードされる。

【0114】

(i) 関数へのバインディングキー: <Key>タグ

通常の無線通信装置100は、1から3つのソフトキー130、標準数字キー134、およびいくつかの特殊目的キー132を有する。HTML pという新しい拡張により、キーパッド128の任意のキーを<KEY>タグを使用してHTML ページにバインディングすることができる。構文は、<KEY KEY=keyLABEL=stringACTION=url|POP|DONE|CLEAR|MENU|GO|NONE>である。

【0115】

KEY属性の値は、以下のうちの1つである。

[1, . . . m] バインディングするソフトキーを指定する。キーは左から右に、または上から下に（それらが電話上のどこにあるかに応じて）番号付けされる。通常 ($m \leq 3$) であるが、これは装置毎に可変でありうる。

[send] 「Send」または「Talk」キーを指定する。

[back] 「Back」キーを指定する。すべての装置がこのキーを備えるわけではない。このキーをバインディングするためには、ページは、特権付与されなくてはならない。

[#n] ダイアルキーのうちの1つを指定するが、但し、nはキー上のラベル（0～9、*、または#）である。すべてのダイアルキーを同じアクションにバインディングするためには、#xを指定する。

[end] 「End」キーを指定する。このキーをバインディングするためには、ページは、特権付与されなくてはならない。

[mode] 「Mode」または「ABC」キーを指定する。

[clear] 「Clear」または「Del」キーを指定する。

[up] アップアローキーまたはダウンアクションを指定する。

[down] ダウンアローキーまたはダウンアクションを指定する。

[left] レフトアローキーを指定する。

[right] ライトアローキーを指定する。

[select] select キーまたは選択アクションを指定する。

[power] power キーを指定する。バインディングするためには、ページは、特権付与されなくてはならない。

[default] 標準アクション以外がとられる場合、アクションを常に明示的に指定しなければならない、back、end、およびpower キーは例外として、他のアクションが指定されていないいずれのキーに対しても1つのアクションを指定する。

【0116】

キーがソフトキー130である場合、LABEL属性の値は、該キーの画面上のラベルに現れるストリングである。ストリングが長すぎて、割り当てられたスペースに収まらない場合には、切り捨てられる (truncate)。LABEL属性は、ソフトキー130に対し

10

20

30

40

50

てのみ有効である。

【0117】

ACTION属性の値は、バインドキーが押下されたときに何が起こるべきかを指定する。

[URL] フェッチすべきURL、またはページにおける埋め込みオブジェクトのためのある他のコマンドを指定する。URLは、処理のために、HTMLpコンテンツハンドラのHTMLpActive関数に渡される。

[POP] 前ページを表示するよう要求する。

[DONE] 最も最近の<TOP>ページの前のページを表示するよう要求する。

[CLEAR] すべてのページをURL履歴スタック108からクリアして、メイン画面を表示するよう要求する。

[MENU] ソフトキー130がメニューを繰り出すよう指定する。メニューにおけるアイテムは、<HEAD>セクションにおける<KEYMENU>タグによって指定される。これは、ソフトキー130についてのみ有効である。

[GO] 現在選択されているリンクをフェッチするよう求める。これは、<LINKMENU>属性を有するページについてのみ有効である。

[NONE] キー押下時に、いずれのアクションも取らないよう求める。これは、オーバーライドすべきキーについてデフォルトアクションを可能にする。

【0118】

HTMLpコンテンツハンドラ114cは、<KEY>タグを有するページをロードすると、キー、ラベル（もしあれば）、およびアクションの関連を格納するキーバインディング表を作成する。

【0119】

ACTION属性にバインディングされたストリングは、HTMLpActivate関数により、以下のように処理される。

【0120】

(ii) HTMLpActivate

HTMLpActivate関数は、キー、ソフトキー、メニューアイテム、ハイパーリンク等のユーザインタフェースエンティティにバインディングされたストリングに対する適切な応答を決定するために使用される。入力は、シェル106からのストリング、より具体的には、KEYまたはKEYMENUタグのACTION属性、またはAタグのHREF属性にバインディングされたストリングである。新しいHTMLpタグにより、一般に、無線通信装置100の任意のキーまたはメニューアイテムを特定のアクションまたはURLに関連づけることが可能になる。

【0121】

図6a～図6bを参照して、HTMLpActivate関数の一実施形態のフローチャートを示す。HTMLpコンテンツハンドラ114cは、現行ページにおける埋め込みオブジェクトのリストを維持する。ページに埋め込みオブジェクトがある場合(602)、該関数は、処理のためにストリングを埋め込みオブジェクトに渡す(604)。埋め込みオブジェクトは、ストリングが処理されたか否か、およびストリングをさらに処理する必要はないことを示すブールを戻す。ストリングが埋め込みオブジェクトによって処理された(606)場合、関数は制御をシェル106に戻す(608)。埋め込みオブジェクトへのストリングの処理(604)については、さらに後述する。

【0122】

ストリングが処理されていなかった場合、HTMLpActivate関数は、その値に従うストリングをACTION属性として、プロセス610に進む。

【0123】

ストリングが「POP」である場合(612)、シェルのShellGoBack(POP)関数が呼び出される(614)。この関数は、URL履歴スタック108の最上部のURLをポップして、先のURLをロードさせる。

【0124】

同様に、ストリングが「DONE」である場合（616）、ShellGoBack（DONE）関数が呼び出される（618）。これは同様であるが、最も最近の<TOP>タグの前のページを表示する。<TOP>タグは、マルチパートフォームの最初のパートを識別するが、これについてはさらに後述する。

【0125】

HTMLpコンテンツハンドラ114cは、フォームまたはハイパーリンクにおける入力フィールドでありうる、ページにおける現行オブジェクトへのポインタを維持する。この現行オブジェクトは、入力フォーカスが配置される場所である。

【0126】

ストリングが「GO」であり（620）、かつ現行オブジェクトがハイパーリンクではない（622）場合には、何も起こらない。現行オブジェクトがハイパーリンクである場合には、コンテンツハンドラ114がハイパーリンクに関連したURLストリングを取得して、そのURLストリングをShellGetURLに渡し（624）、これが実際のコンテンツをフェッチする。

【0127】

ストリングが「CLEAR」である場合（626）、シェル106のShellGoBack（CLEAR）が呼び出される（628）。この関数は、URL履歴スタック108をクリアして、デフォルトのメインページを表示させる。

【0128】

ストリングが「resetformid」の形式を有する場合（630）、該関数はフォーム番号（form number）がformidである入力要素を各自の元の状態に戻す（632）。このアクションは、<INPUT TYPE=reset>ガジェットについての任意のソフトキー130またはソフトキーメニューにバインディングされる。

【0129】

ストリングが形式「submitformid, label」である場合（634）、該関数は、フォーム番号formidを定義したFORMタグのMETHODおよびACTION属性に従って、フォーム番号formidを提出する（636）。labelは、もしあれば、ユーザが起動したのはいずれの<INPUT TYPE=submit>ガジェットであるかを示し、そのため、そのネーム値の対を、フォームにおける残りのガジェットからの値と共に提出することができる。

【0130】

ストリングが「SELECT」である場合（638）、該関数は、以下のガジェット型に従って、ユーザが選択したユーザインタフェースガジェットを起動する（640）。

<INPUT TYPE=radio> そのラジオボタンを選択し、同じ名称の他のラジオボタンを選択しない。

<INPUT TYPE=checkbox> チェックボックスをチェックするか、またはチェックしない。

<SELECT> オプションが表示される場合、ユーザが選択したオプションを選択する。SELECTリストがポップダウンリストである場合、ポップダウンを閉じる。ハイパーリンクリンクをたどる。

【0131】

ストリングが「NONE」である場合（642）、いずれのアクションも取られない。

【0132】

これらすべての条件をパスした後、ストリングがURL等の任意の他の値である場合（644）、ストリングは処理のためにShellGetURLに渡される（646）。URLが引数を持たない（その中に「?」がない）場合には、エクストラデータとしてこのページに渡されたパラメータはいずれもまた、ShellGetURLへの呼び出しにおいて渡される。

【0133】

HTMLpActivate関数および埋め込みオブジェクトに関するその特定の利点

10

20

30

40

50

の例について、さらに以下に説明する。

【0134】

<KEY>タグを使用したActivate関数による処理の例は、次の通りである。ページが以下のKEYタグ、<KEY KEY="send" ACTION=phone:dial>を含むと想定する。

【0135】

このページがロードされると、HTMLpコンテンツハンドラ114cは、Sendキーと、そのキーバインディング表におけるURL「phone:dial」の間の関連を格納する。この格納データは、ユーザがSendキーを押下したときに、電話プロトコルハンドラ112gの電話ダイアル機能を起動するために使用される。HTMLpコンテンツハンドラ114cは、現行のコンテンツハンドラ114である。

10

【0136】

ある後の時点でユーザがSendキーを押下すると仮定する。可搬性レイヤ118は、キーが押下されたことを示すシェルのShellProcessKey関数を呼び出して、Sendキーについてのキー番号、および押下されたことを示すフラグを渡す。上述したように、シェル106は、キーストロークターゲットリストを維持する。ShellProcessKeyは、受信したキーをスタック上の最初のターゲットに送る。このターゲットが該キーについて目的を持たない場合、ターゲットはシェル106のPreviousInput関数を呼び出して、Sendキーインデックスを渡す。シェル106は、キーストロークターゲットリストにおける次のアイテムを見つける。このプロセスは、キーがHTMLpコンテンツハンドラ114cに渡されるまで繰り返される。URL履歴スタック108の最上部にあるURLはHTMLpコンテンツハンドラ114cへのポインタ406を含むため、これは、シェル106が現行のコンテンツハンドラ114のProcessKey関数を呼び出すときに発生する。

20

【0137】

次に、シェル106は、HTMLpProcessKey（「Send」）を呼び出す。この関数は、SendキーとURL「phone:dial」の間の関連を含むキーバインディング表を見る。HTMLpコンテンツハンドラ114cは、ShellActivate（phone:dial）を呼び出し、これは、HTMLpコンテンツハンドラ114cのHTMLpActivate関数を呼び出す。

30

【0138】

ここで、ページに埋め込みオブジェクトがないと仮定する。次に、該関数は、入力ストリングをPOP、DONE、GO、CLEAR、およびNONE等の様々な他のアクションに対してテストする。ストリング「phone:dial」はこれらのいずれともマッチしないため、HTMLpActivate関数は、シェルのShellGetURL（phone:dial）関数を呼び出す。

【0139】

シェル106は、この関数を図5のフローチャートに示すように処理する。例を続けて、シェルは、URLが電話プロトコルハンドラ112gについてのものであると決定し、「dial」の部分処理のために電話プロトコルハンドラ112gに渡す。このプロトコルハンドラ112gは、ダイアルすべき番号を含む「CallManager」型（コールマネージャコンテンツハンドラ114bに定められた）のコンテンツストリームを戻す。シェル106はHTMLpコンテンツハンドラ114cを閉じるが、該URLをURL履歴スタック108から削除しない。シェル106は、URL「phone:dial」をURL履歴スタック108の最上部に配置する。シェル106は、電話プロトコルハンドラ112gが返したコンテンツストリームから、コンテンツハンドラ114のストリング名を取得して、ストリームを処理する。シェル106は、その表におけるストリングを索引して、URL履歴スタック108の最上部エントリのCONTENTフィールドを更新する。最後に、シェル106は、新しい現行のコンテンツハンドラ114のOpen関数を呼び出して、コンテンツストリームデータを渡す。コールマネージャコンテンツハ

40

50

ンドラ114bのOpenルーチンにおいて、コールマネージャコンテンツハンドラ114bは、ストリームから電話番号を検索して、電話制御モジュール120における、通話を確立するために必要な関数を呼び出す。

【0140】

(iii) メニュー構築

ソフトキー130が<KEY>タグを使用してメニューにバインディングされ、かつACTION="menu"である場合、メニューへのエントリは、ページのHEADセクションにおける<KEYMENU>タグを使用して指定される。<KEYMENU>タグは、以下の構文を有する。

```
<KEYMENU KEY=nLABEL=stringACTION=url|POP|DONE|CLEAR|GO>
```

10

【0141】

エントリは、入力(encounter)された順序でメニューに表示される。しかし、メニューエントリがすべて共にヘッダにある必要はない。

【0142】

KEY属性の値は、いずれのメニューにエントリを追加すべきかを指定する。これは、存在すべきメニューを指定した<KEY>タグに与えられる値と同じである。

【0143】

LABEL属性の値は、メニューエントリに現れるストリングである。メニューは、すべてのエントリを保持するのに必要な幅である。しかし、ラベルは、画面よりも幅のある場合には、切り捨てられる(truncate)。

20

【0144】

ACTION属性の値は、エントリが選択される場合に何が起こるべきかを指定する。可能な値は、

[URL] フェッチすべきURL、またはページにおける埋め込みオブジェクトのためのある他のコマンドを指定する。

[POP] 前ページを表示するよう要求する。

[DONE] 最も最近の<TOP>ページの前のページを表示するよう要求する。

[CLEAR] すべてのページをURLスタックからポップして、メイン画面を表示するよう要求する。

[GO] 現在選択されているリンクをフェッチするよう要求する。これは、<LINKMENU>属性を有するページについてのみ有効である。

30

【0145】

これらのACTION属性は、上述したHTMLpActivate関数における<KEY>タグの属性と同じ様式で処理される。

【0146】

図7は、メニューが表示されるよう選択されるとき、キーメニューが定義されているページ、および表示画面136についてのHTMLpソースコードの例を示す。HTMLpコードを左側に示し、その結果得られるページを右側に示す。4行目は、最初のソフトキー(KEY=1)についてのメニューを定義する。なお、メニューが開いている場合、メニューについてのソフトキーラベル216は「Select」に変わり、ユーザがソフトキー130を再度押下すると、選択されたエントリが起動されることを示し、このラベル216は、代わりにメニューを閉じるよう変わってもよく、選択を起動するために、Sendキーを残す。5~7行目は、このメニューについてのメニューアイテムを定義し、それぞれ、適切なページを表示して、メッセージ、最近の通話、または電話設定を検索するために、ユーザインタフェース定義ファイル104のうちの1つを指定する各自のACTION属性を有する。「Select」とマークされたソフトキーを押下するか、あるいはエントリの左側のアイコンにマッチする数字キーを押下することにより、メニューにおけるエントリの1つを選択すると、KEYMENUにおいて指定されたACTIONが実行される。この例において、適切なHTMLpユーザインタフェース定義ファイル104は、ROMからフェッチされる。

40

50

【0147】

(iv) 遅延ヘルプ

コンピュータについての多くのユーザインタフェースは、ある形式のオンライン文脈特定ヘルプテキストを提供する。従来、これらのヘルプ画面は、ユーザがデータを入力しようとしているユーザインタフェースの部分に重なった各自のウィンドウに表示される。さらに、ヘルプ画面は通常受け身的であり、ユーザがヘルプを要求するという直接的なアクションに応答してのみ、現れる。しかし、非常に小さな画面表示を備える無線通信装置では、コンテンツエリアの上にヘルプ画面を重ねると、ユーザの妨げになる。さらに、ユーザがヘルプを利用することができることを知らない場合もあるため、ユーザがヘルプを要求するのを受け身的に待つということは、ユーザによっては、補助が不十分である可能性がある。

10

【0148】

ページを見ているときに、次に何をすべきかが不確かでありうるユーザを補助するため、かつユーザがデータを入力しようとしている、ページのコンテンツエリア214を隠さずに、かかるヘルプを提供するために、本発明は、ユーザによるキーストローク入力なしで、ある量の時間が経過した後、ページのタイトル210の代わりに、ヘルプテキストを画面にわたって自動的にスクロールできるようにする。1つよりも多くのヘルプテキストストリングを指定してもよく、この場合、ストリングは、互いの間に適した間隔をおいて、連続して表示される。ヘルプテキストストリングは、ページが見られる度に、それぞれ一度だけ表示される。

20

【0149】

ページが1つまたは複数のヘルプストリングを指定できるようにするためには、新しい<HELP>タグがドキュメントの<HEAD>セクションにおいて指定される。<HELP>タグの構文は、

<HELP>helpstring</HELP>

であり、ここで、helpstringとは、表示すべきヘルプテキストである。

【0150】

<HELP>タグを含むページがロードされると、HTMLpコンテンツハンドラ114cは、ヘルプテキストストリングを含む、表等の構造を構築する。この表は、シェル106に渡され、シェル106が後の使用のために表を格納する。そして、<HELP>タグの機能性は、アイドル時処理中に主にシェル106によって処理される。

30

【0151】

シェル106は、最後のキーストロークからの秒数カウンタを維持している。カウンタは、通常各ShellProcessKey毎にクリアされる。リアルタイムオペレーティングシステム122は、バックグラウンドで実行されるタイマを有すると共に、カウンタを増分する、シェル106におけるルーチンを呼び出す。カウンタが閾値の秒数に達すると、シェル106がスクローリングバナーオブジェクトを作成し、これに対して、表中の最初のヘルプテキストストリングを表示するよう命令する。スクローリングバナーオブジェクトは、いつ、ヘルプテキストストリングが完全に画面表示136からスクロールし終わるかを内部で決定して、これをシェル106に通知し、シェル106はタイトル210を再び表示する。第2の閾値が、テキストヘルプストリング表示のためにセットされる。閾値は、ヘルプテキストストリングを表示している望ましい時間の長さを基にして、シェル106によって予め決定される。

40

【0152】

図8は、2つのHELPタグを含むページについてのHTMLpソース、およびページがロードされると、渡される画面表示136の結果得られるシーケンスの例を示す。HTMLpコードの7~8行目は、表示すべきヘルプテキストを定義する。2番目および3番目の画面イメージは、最初のヘルプテキストストリングがタイトルの代わりにスクロールされているところを示す。

【0153】

50

(v) テンプレートとしてのページ

本発明には、標準HTMLエディタを使用して、フォームエントリフィールドを完全にするために、Cコードによって渡される引数を使用して、またはページにおけるフォームの初期状態を決定するために、装置からオンザフライにフェッチされるデータを指定することで、ページを設計できるようにする多数のHTMLの拡張がある。これらの拡張としては、テンプレート、条件付きHTML、構成設定性能、および「included（含まれた）」HTMLが含まれる。

【0154】

一般に、埋め込みオブジェクトのためのCコードは、表示すべきパラメータを有するが、表示の形式はページについてのHTMLで定義されることが望ましい。例えば、着信にある形式のデータを表示しているページは、電話番号およびそれに関連する氏名を表示することが好ましい。したがって、着信表示用のHTMLページは、パラメータ（電話番号および発呼者氏名）を取り、必要であればそれらをフォーマットすることが可能であるべきである。

【0155】

しかし、従来のHTML 3.2は、この所望の用途に、データを直接ページに渡す機構を提供せず、最高でも、HTMLに、ページを必要に応じて作成することを可能にするだけである。HTMLの生成は遅くかつコンピュータ集約的であり、ページ生成のための実行可能スクリプトは通常、生成されるページよりも多くの格納領域を必要とし、そのため、ページ自体を格納するよりも効率を低くしている。引数を間接的に渡せるようにすることで、本発明は、実行時にHTMLを生成する必要性をなくす。これにより、ページをROM 126に格納することが可能になり、かつ必要なメモリは、必要に応じてHTMLを生成するためのコードよりも少ない。

【0156】

(a) テンプレートとしてのページの使用

データをページに渡せるようにする最初の拡張は、<TEMPLATE>タグである。<TEMPLATE>タグは、ページのどこに現れてもよい。ドキュメントの構造において、適切な場所にある対応する</TEMPLATE>タグとマッチしなければならない。

【0157】

<TEMPLATE>タグと</TEMPLATE>タグの間のテキストはすべて、%(url)、 %[url]、または%<url>の形式のエスケープについて検査される。これは、タグ内の、さらには引用された属性値内のテキストを含む。このようなエスケープが見つかり、URLについてのデータがフェッチされ、普通テキストまたはHTMLである場合には、まるで最初からそこにあったのかのように、エスケープの代わりにHTMLドキュメントに挿入される。<TEMPLATE>と</TEMPLATE>の間のテキストに%キャラクタを含めるには、「%%」のようにもう一つの%が前にこなければならない。

【0158】

フォーム%<url>は、urlについてのデータとして戻されたデータを、別のURLについての引数として使用するためにエンコードさせる。URL引数は、制限されたキャラクタセットを有し、該キャラクタセット外のものはいずれも、後に2つの16進数字が続く%としてエンコードされる。

【0159】

%(url)と %[url]の間の違いは、HTML pコンテンツハンドラ 114cのキャッシングの振る舞いにある。通常、ページについてのデータは、構文解析され、ページのレンダリングに必要な情報が、URLがURL履歴スタック 108に残っている限り、保存される。これにより、すでにフェッチされているページを迅速に再表示することができる。しかし、 %[url]エスケープがテンプレートセクションに見られる場合には、ユーザが最後にそのページを見たときからの、URLデータにおける変更をいずれも取り込

むために、URLについてのデータが、ページの再表示が必要ときにページを再度構文解析することを保証するのに十分動的であることが示される。

【0160】

図9は、TEMPLATEタグの例を示す。この例において、19行目～25行目の<TEMPLATE>タグの間のテキストは、テンプレートテキストを定義し、20行目のエスケープは、URL「extra:name」をフェッチさせる結果となり、これにより、テキストを、何であれ、変数「name」のもとに格納されているデータと置換する。画面表示は、これをテキスト「Adam M」として示す。

【0161】

HTML4.0の仕様は、ページにおける埋め込みオブジェクトの使用に備えている。通常、埋め込みオブジェクトは、各自のコンテンツの構築および表示に責任を負う、ページ上のある場所に位置決めされたコードのアイテムである。埋め込みオブジェクトは、HTMLソースの所望の位置に配置された、OBJECTタグについてURLにおいて指定される。通常、URLは、ActiveX制御、Javaアプレット、Cコード等のコードエンティティを指定する。HTML4.0の仕様では、URLは単にサーバに渡されて、サーバが所望のエンティティを戻すだけである。しかし、HTML4.0では、埋め込みオブジェクトを有するページが一旦ロードされると、埋め込みオブジェクトに対する引数のさらなる処理または引き渡しを行うことはできない。特に、ユーザが、埋め込みオブジェクトを含むページ上のハイパーリンク（URLに関連するユーザインタフェースガジェット）を選択する場合、埋め込みオブジェクトは、URLを処理する機会を何等持たず、その代わり、URLは単にリンクされたページに続くだけである。

【0162】

しかし、本発明は、処理のために、ハイパーリンクに関連するURLまたはユーザインタフェースガジェットをまず埋め込みオブジェクトに提供することで、埋め込みオブジェクトの機能性を拡張する。これは、サーバがページを更新する必要なく、埋め込みオブジェクトに直接HTML形式で提供される引数に応答させるものである。この埋め込みオブジェクト機能性の実施は、図6aおよび図6bに関して上述したように、HTMLpコンテンツハンドラ114cのHTMLpActivate関数において提供される。

【0163】

上述したように、HTMLpコンテンツハンドラ114cがURLであるストリングを処理する際、ページに埋め込みオブジェクトがある場合には、HTMLpコンテンツハンドラ114cはURLを埋め込みオブジェクトに渡す。本発明によれば、埋め込みオブジェクトは、URL処理の際に、3つのアクションのうちの1つを行う。

- ・URLをコマンドとして処理する。
- ・URLにおいてエスケープシーケンスを探して、URLをフェッチすべきシェル106に渡す前に、それらシーケンスをユーザが選択したデータからの情報で置換する。
- ・URLを処理せずにHTMLpコンテンツハンドラ114cに返し、これは、HTMLpActivateの残りに従って処理される。

【0164】

第2のタイプの処理の例として、HTMLpページにおける電話帳埋め込みオブジェクトのインスタンスは、「@n」または「@i」等のエスケープシーケンスを探して、それをユーザが選択した電話帳レコードの名称またはレコードIDで置換することができる。電話帳オブジェクトは、レコードの名称あるいは任意のフィールドのいずれかを編集することができる編集メソッドを含む。URL「phone:edit?Name=@n」を電話帳オブジェクトに渡すことで、名称が選択された電話帳レコードからすでにセットされている状態で、もう一つの番号を選択された電話帳レコードに追加するという処理が、電話帳エントリ作成プロセスに入ることによって開始され、URL「phone:edit?id=@i」を電話帳オブジェクトに渡すことで、レコードのすべてのフィールドが編集される。

【0165】

この拡張機能性の利点は、Cでハードコードされたページを有する代わりに、ユーザが

実行することのできる機能の仕様はHTMLで行ないながら、Cでコードされたデータアクセスおよび動的振る舞いを要求する要素を有することができることである。これは、埋め込みオブジェクトがユーザと対話する唯一の方法は、ここでも、埋め込みオブジェクトが実施される言語でハードコードされた各自のユーザインタフェースを提示することによるものであるため、HTML 4. 0では不可能であり、したがってサービス運営者による変更またはブランド化が容易ではない。

【0166】

図10および図11は、可能な電話帳ページの2つの例を提供するとともに、埋め込みオブジェクトが、本発明の拡張処理により提供することができる柔軟性を示す。

【0167】

図10において、ユーザは、新しいエントリを作成できるか、または短縮ダイアルキー、呼び出し音、または番号を変更するために、既存の電話帳エントリのフィールドの1つを修正することができる。5行目は、表示されて図示されており、ユーザが最初のソフトキー130を押下することで起動されるキーメニューを定義する。メニューエントリ（6～12行目）は、現在の選択からのデータについてのエスケーププレースホルダを有するURLにバインディングされる。特に、6行目は、メニューアイテムについてのACTIONを、埋め込み電話帳オブジェクトにおいて現在選択されている名称にすでに指定されている、電話帳エントリ作成プロセスの最初のページに変更させる、埋め込み電話帳オブジェクトについてのURLとなるように定義する。一般に、本発明における他のURLは、選択された電話帳レコードを用いて、各自の引数を埋めることで起動可能であり、これらURLはいずれも、メニューエントリ、ソフトキー、またはキーボード上の他のキーにバインディングすることができる。特殊なエスケープシーケンスについて、電話プロトコルのphone: list URLに関して後述する。

【0168】

図11において、ユーザは、新しいエントリを作成するか、または別個のページに進み、特定のエントリについてのパラメータすべてを表示して、ユーザが望む場合にはそれらを変更することができる（これは、別個の画面pbedit. htmlを介して行われ、phone: list埋め込みオブジェクトへのediturl引数における「@」エスケープは、pbedit. htmlへ渡すために、関連するすべてのエントリを抽出する）。さらに、ここでは、グラフィカルタイトルバーが、無線通信装置の製造業者のロゴをバナーとしたタイトル付き背景と共に使用される。埋め込まれるオブジェクトは、電話リストオブジェクトへのURLと共に10行目において指定される。

【0169】

HTMLpコンテンツハンドラ114cが<OBJECT>タグに遭遇すると、シェル106に、OBJECTタグへのCODE属性として与えられるURLに関連するデータをフェッチするよう要求する。シェル106は、「Object」型でなければならないこのデータをコンテンツストリームとして返す。ストリームには、以下を含む構造がある。

- ・オブジェクトへのポインタ（HTMLpページウィンドウの子となるべきウィンドウ）
- ・与えられているストリングで呼び出すことができるHTMLpActivate関数へのポインタ
- ・オブジェクトの現在状態をフェッチする関数へのポインタ。これは、HTMLpCloseにより呼び出される。
- ・フェッチされた状態を受け入れ、オブジェクトをその状態に回復する関数へのポインタ。これは、ページが以前に表示されたことがあると分かる場合、HTMLpOpenによって呼び出される。
- ・オブジェクトの「値」をストリングとして戻す関数へのポインタ。これは、オブジェクトが、提出されているフォームの一部である場合に呼び出される。
- ・オブジェクトをセットすべき「値」を受け入れる関数へのポインタ。該値は、ストリングである。関数は、オブジェクトがフォームの一部であり、かつオブジェクトと同じ名称

10

20

30

40

50

を有するエクストラデータ（NAME属性によってOBJECTタグに与えられる）が引き渡された場合に呼び出される。

【0170】

（b）装置設定へのアクセス

無線通信装置100の各種構成可能なパラメータは、さらに後述する構成プロトコルを介してアクセス可能である。フォームがジェットを使用し、各設定について可能な値を指定して、これらの設定を調整できるページを提供することが望ましい。しかし、これを行うためには、これらフォームがジェットの初期状態を、影響するものと思われる設定の現行値にマッチするようセット可能でなければならない。

【0171】

装置設定の値の設定に使用する最も好ましいフォームガジェットは、ラジオボタン、チェックボックス、およびスクローリングリストである。ラジオボタンおよびチェックボックスは双方とも、RADIOあるいはCHECKBOXいずれかのTYPE属性を有するINPUTタグを介してアクセスされる。これらの入力要素について、HTML3.2において、入力要素がフォーム上で選択されるか否かを定義する選択属性を指定することが可能である。選択属性は、CHECKEDおよびSELECTEDを含む。CHECKED属性は、ラジオボタンまたはチェックボックスを初めに選択すべきであることを示す。同様に、スクローリングリストは、<OPTION>タグを内部に有する<SELECT>タグによって指定され、そのうちの任意のものは、選択リストのオプションを初めに選択すべきであることを示すために、SELECTED属性を有しうる。

【0172】

通常、これらの属性はブールであり、CHECKEDまたはSELECTED属性がソースに存在する場合、ラジオボタン、チェックボックス、またはオプションが選択される一方で、存在しない場合には、選択されない。しかし、本発明では、これらの属性は、値を受け入れるように拡張されている。値は、式の形式を採り、これは、真と評価される場合、アイテムを最初に選択させる。

【0173】

式は、URLからデータをフェッチし、それを真または偽についてチェックされるブール値として扱うか、あるいは同等かまたは不等かについてストリングと比較される。

【0174】

構文は、

ATTRIBUTE=[!]url

であるか、または

ATTRIBUTE=url[!]=string

である。

【0175】

ATTRIBUTEは、CHECKEDまたはSELECTEDのいずれかである。ここで、URLは、構成プロトコルへのものであり、「config:setting」の形式を採るが、但しSettingは、構成プロトコルハンドラ112bがアクセスする関心のある特定の設定である。

【0176】

最初の構文形式は、フェッチされたURLデータをブール値として扱い、テキストを整数に変換して、それが0か、または0ではないかを見る。先行する「！」が存在し、かつ値が0である場合、アイテムが選択され、また先行する「！」が存在せず、かつ値が0ではない場合、アイテムが選択される。url自体が等符号を含む場合には、それを括弧でくくらなければならない。

【0177】

2番目の構文形式は、stringとURLについて検索したデータの間にストリングの比較を行う。「!=」が与えられる場合、ストリングが等しくなければアイテムが選択され、また=だけが与えられる場合、ストリングが等しければアイテムが選択される。

【0178】

URLが戻したデータが普通テキストでも、またHTMLでもない場合、式は常に偽と評価し、アイテムはチェックされないままである。

【0179】

図12は、この式の使用例を提供し、HTMLpソースおよび結果得られるページの双方を示す。9～14行目は各種構成設定を指定し、CHECKED属性が、構成プロトコルハンダーへのURLである式によってセットされているところを図示している。図示のページは、これら設定を構成するための結果得られるユーザインタフェースを示す。この例において、ユーザには、3つのラジオボタンと、テキスト入力フィールドとが提示される。ユーザは、画面表示136のバックライトを、装置の使用時のみオンするか、または使用時か、プラグインされているときにオンするか、あるいは全くオンにしないかを指定することができる。さらに、ユーザは、入力することなく、それが過ぎた後は無線通信装置がもはや使用されていないと見なされる秒数をセットすることができる。これらの設定は、「バックライト」および「バックライト遅延」設定それぞれに格納することができる。画面が最初に現れると、現行設定に相当するラジオボタンがチェックされ、テキスト入力フィールドが現在の遅延を含む。

【0180】

一般に、この拡張された拡張属性を有する<INPUT>タグがロードされると、HTMLpコンテンツハンドラ114cがURLをフェッチすべきシェル106に渡す。HTMLpコンテンツハンドラ114cは、フェッチされたデータをその構文形式に従って上述したように評価し、URLデータの評価に従って選択属性の値を確立する。「including（包含）」HTML。

【0181】

「including（包含）」は、HTMLまたはHTMLpコードのブロックをページにおいて参照できるようにする、HTMLに対する拡張である。包含されたHTMLはいずれも、包含されたHTML自身が他のHTMLを含むことができるように、再帰的に解かれる。現在、HTML4.0は、ページからページにHTMLのブロックを包含する機構を提供していない。包含HTMLの利点は、多くのページが現れるよう設計される、ページのヘッダまたはフッタ、ナビゲーションツールバー等の共通のHTMLコンポーネントを容易に包含ページへの単一のリファレンスに組み込むことができる。

【0182】

本発明は、任意のソースからのHTML（普通テキストも含む）を、そのURLを与えることにより、含むための機構を提供する。これは、主にテンプレートページを介して装置設定を表示するために使用されるが、複数のページに共通する要素をコンテンツアーカイブの別個部分に配置して、それをその他のページに含めることにより、コンテンツサイズを低減するためにも使用することができる。該機構は、以下の構文を有する<INC>タグによって提供される。

```
<INC SRC=urlDYNAMIC>
```

【0183】

SRC（ソース）属性は存在しなければならず、そのデータをその時点にドキュメントに包含すべきURLを指定する。<INC>タグがHTMLpコンテンツハンドラ114cによって解かれると、URLに関連するデータがフェッチされ、<INC>タグの場所に挿入される。

【0184】

<INC>タグは、別のタグの中を除いて、ページのどこに用いてもよい。例えば、「<A HREF=<INC SRC=file:commonurl.txt>>」は許されない。

【0185】

図13は、<INC>タグの例を示す。この例において、最初のページのinfo.htmlは、第2のファイルbody.htmlを参照する2行目に<INC>タグを有し、body.html自身は、第3のファイルstdlogo.htmlを参照する

10

20

30

40

50

8 行目に< I N C > タグを有する。最初のページがロードされると、H T M L p コンテンツハンドラ 1 1 4 c は、すべての< I N C > タグを完全に解き、図示のような結果得られるページを生成する。ファイル i n f o . h t m l は、ファイル e n d b o d y . h t m l を参照する 1 3 行目に< I N C > タグを有し、この後者のファイルは、完全に異なるファイル b b o d y . h t m l に現れる< B O D Y > タグを適切に閉じる< / B O D Y > タグを含む。

【0186】

一般に、ページの構文解析時、H T M L p コンテンツハンドラ 1 1 4 c は、< I N C > タグを識別する際に、リファレンス U R L をフェッチして、ファイルからのデータを現在のファイルのソースに直接挿入し、必要に応じて、ページを表示するためにそれを解く。

10

【0187】

D Y N A M I C 属性が< I N C > タグについて指定される場合、それは、ページが一時的に閉じられるときにキャッシュされた表示命令を使用するのではなく、ユーザがそれを戻したときに、ページを再構築させる。一般に、D Y N A M I C 属性は、S R C として使用される u r l が、ページが一時的に閉じられている間に変更されうるデータを参照していることを示し、そのため、かかるあらゆる変更を調整するよう、それを再び読み込んで、ページを再構築しなければならない。

【0188】

(d) 条件付き H T M L

テンプレート H T M L p の各ページの表示は、ファイルプロトコル 1 1 2 d についての U R L における「？」に続く引数（ファイルプロトコル 1 1 2 d についての U R L において指定される引数が、e x t r a : p r o t o c o l を使用して、ファイル内の値として使用可能である）として直接、参照している U R L における M E T H O D = N E X T フォームからのフォームデータとして、あるいは本発明における C コードからのパラメータとして、テンプレートページについての U R L とともに渡されるパラメータによって制御することができる。

20

【0189】

条件付き H T M L は、ページのいずれの要素を表示するかを制御するために、条件付き式をページの H T M L ソースにおいて直接エンコードすることを許さない。本発明は、この欠点を新しい< I F > タグにより克服する。< I F > タグは、パラメータまたは装置設定等の式のテストを可能にして、ページの表示を制御する。構文は、以下の通りである。

30

```
<IF TEST=expressionDYNAMIC>html<ELSE
    TEST=expression>html<ELSE>html</IF>
```

【0190】

T E S T 属性における式は、C H E C K E D および S E L E C T E D 属性について上述したのとまったく同様に評価される。

【0191】

例えば、以下の H T M L p ソースを考慮する。

```
<IF TEST=extra:conf>
    <KEY KEY=1 LABEL=Conference
ACTION=phone:conference>
<ELSE>
    <KEY KEY=1 LABEL="Pick
Up"ACTION=phone:answer>
</IF>
```

40

```
<KEY KEY=send ACTION=phone:answer>
```

```
<IF TEST=config:anykey>
```

50

<KEY KEY=default ACTION=phone:answer>

<KEY KEY=back ACTION=phone:answer>
<ELSE>

<KEY KEY=default ACTION=none>

<KEY KEY=back ACTION=phone:ignore>
</IF>

【0192】

10

最初の<IF>タグにおいて、TESTは、ページをロードしたCコードによって、ページに渡されるエクストラデータ「conf」に関して評価される。このデータは、HTMLpコンテンツハンドラ114cに対して利用可能な変数に格納される。ページがロードされると、「conf」がTRUE（真）に評価される場合、最初のソフトキー130（KEY=1）が「Conference」とラベルされ、キーバインディングテーブルにおいてURL「phone:Conference」にバインディングされ、ユーザが電話の会議特徴を起動できるようにする。「conf」がFALSE（偽）と評価される場合、ソフトキーは、代わりに「Pick Up」とラベルされ、キーは異なるURLにバインディングされる。

【0193】

20

第2の<IF>タグにおいて、テストされるデータは、「config:anykey」URLによってアクセスされる無線通信装置の構成設定である。この設定についての装置設定に応じて、Backキーを含むすべてのキーが「phone:answer」関数にバインディングされるか、あるいはBackキー（および、特殊なバインディングを有する他の任意のキー）以外のすべてのキーは何も行わない一方で、Backキーが「phone:ignore」関数にバインディングされる。

【0194】

<IF>タグは、テンプレートモードにおいて使用される形式%[url]と同様に、パーサに、使用するURLが動的データを生成し、ページが再度構文解析されるべきであると教えるDYNAMIC属性を有し、urlが、ページが再度見えるようになったときに、ページを再構築するよう要求するのに十分動的なデータを参照することを通知する。

30

【0195】

(vi) 電話番号エントリフィールド

HTML4.0は、汎用言語として設計され、無線通信装置、特に電話をかけ、かつ電話番号および関連する氏名を格納することができるものにおける使用に特に適合させる特徴を何等含んでいない。

【0196】

HTMLをこのような無線通信装置100により適合させるために、本発明は、「phonenum」および「phonename」を<INPUT>タグのTYPE属性についての新しい値として指定する。一般に、<INPUT>タグは、チェックボックス、ラジオボタン、テキスト、またはイメージ等のデータ入力タイプの仕様を可能にするものである。

40

【0197】

本発明の新しい入力タイプは、ユーザが電話番号または人の氏名を入力できるようにする。ユーザタイプとして、入力フィールドは、入力データを使用して、電話帳データ構造においてマッチングするレコードを索引する。そして、マッチングしたレコードが、電話帳表示に使用されるのとまったく同じフォーマットで入力フィールド下のリストに表示される。マッチングしたレコードが一旦表示されると、ユーザは、リストにおけるアイテムを選択して、そのアイテムをフォームを完成させるために使用することができる。

【0198】

50

より具体的には、入力タイプが「`phonenum`」である場合、入力数字が電話帳にあるすべての電話番号と比較され、マッチした電話がリストに表示される。ユーザがリストにおけるマッチした電話番号の1つを選択すると、これが、入力とマッチした部分に下線を引いた状態で、入力フィールドを完全に選択された電話番号で置換させる。マッチングにおいて、一桁の数字（0～9）または二桁の数字（00～99）が短縮ダイアルリストのみに対してマッチングされ、マッチした短縮ダイアル番号を表示する。

【0199】

入力タイプが「`phonename`」である場合、入力キャラクタが電話帳における氏名と比較され、マッチしたエントリが、マッチしたキャラクタに下線が引かれた状態で表示される。氏名の最初の語におけるマッチは、後続する語におけるマッチよりも優先され、リストはそれに従って格納される。ユーザがマッチした氏名の1つを選択すると、これにより、入力フィールドを完全にマッチした氏名で置換させる。

10

【0200】

図14は、HTMLソースおよび結果得られるページの例を示す。ここで、7行目は、電話番号をダイアルするための電話プロトコルを指定し、入力タイプは「`phonenum`」である。ユーザは最初に「2」をタイプしており、これが短縮ダイアルリストに対してマッチングされ、マッチした氏名を表示する。次のイメージにおいて、ユーザは、「995」とタイプしており、これが電話帳リストに対してマッチングされ、マッチングした1つの氏名を表示する。3番目のイメージでは、ユーザが氏名を選択しており、これにより、前にある数字および残りの数字を含む、マッチした電話番号全体が、ダイアリングでの使用のために入力フィールドに挿入される。

20

【0201】

(b) マルチパートフォーム

上述したように、デスクトップおよびその大型画面向けに定められた通常のHTMLページでは、従来のフォームは多くの入力フィールドを使用する。仮に、かかるフォームが通常の無線通信装置100の小さな画面表示136上に表示されると、ユーザが、いずれの所与のときでも大部分が画面表示136からスクロールして消えていくフォームのコンテキストを見失うか、またはユーザが入力しているデータの大部分を見ることができないため、ユーザは、フォームにおいて非常に迷いやすい。図16は、無線通信装置100の表示画面136において使いにくい従来のHTMLフォームの例を示す。

30

【0202】

1つの解決策は、単一のフォームを、それぞれ必要なアイテムの1つまたは2つを集めた一連のフォームに分割することである。しかし、この場合、従来のHTMLでは、次のフォームをフェッチするURLの一部として、各フォームからデータをサーバに伝送する必要がある。そして、サーバは、URLで渡されたデータを取り、ページを戻す。該ページは、先行フォームから渡されたデータが、戻されたページ中のフォームに「隠し」タイプ入力要素として含む状態で、オンザフライで生成されなければならない。この様式では、ユーザが入力するデータは、フォーム全体が記入されるまで、複数回送信および返信される。このプロセスは、非常に帯域幅集約的であり、時間がかかるとともに、費用がかさむ。

40

【0203】

1つのフォームを複数のフォームに分割することの他の欠点は、ユーザが途中で結局フォームを完成させたくないと決心した場合に、ユーザが踏まなければならない手順である。このような場合、ユーザは、記入したサブフォームそれぞれについて、「End」または「Back」キーを一回押下する必要がある。これは、それぞれが別個のページであるためである。ユーザが急いでいる場合には、フォームをキャンセルしているときに、行き過ぎやすく、実際に望むところが消えてなくなる結果に終わる可能性がある。

【0204】

図17aおよび図17bは、1つのHTMLページでクライアントコンピュータから受信したパラメータが、クライアントコンピュータにアップロードされる前に、サーバコン

50

ピュータにより次のHTMLページにおけるHIDDENタイプの入力として挿入される、上述した従来のマルチフォーム方法を示す。図からわかるように、クライアントとサーバの間でこのデータを連続して往復して渡すために、複数のページを動的に作成しなければならない。

【0205】

サーバとクライアントの間でデータを往復して送信するためのこのプロトコルは、クライアントとサーバの間の各トランザクションがステートレス (stateless) であり、したがって現行状態を完成するために、先行状態からのデータには暗黙的に依拠しなくてもよいという標準HTMLの基本的な仮定から生じるものである。

【0206】

本発明は、HTMLのこれらの欠点を、クライアントは各自の状態を十分に保存するとともに、この情報を次の状態の決定に用いることができるという事実を利用するよう設計された、フォームについての新しい「NEXT」メソッド、および新しい<TOP>タグにより、克服する。

【0207】

(vii) 「NEXT」フォームメソッド

HTMLページにおけるフォームは、ユーザからデータを収集するための1つまたは複数の入力要素からなる。データは、それぞれやってきた入力要素の名称でタグ付けされており、FORMタグ内の2つのパラメータ、METHODおよびACTIONに従ってサーバに提出される。ACTIONは、データが通常添付されるURLであり、METHODはGETまたはPOSTのいずれかである。GETは、データに基づいて別のページをフェッチするために使用され、POSTは通常、トランザクションを締めくくろう、データを送信するために使用される。

【0208】

これら2つのメソッドに対して、本発明は、新しいNEXTメソッドを追加する。一般に、NEXTメソッドは、フォームを、各パートがフォーム全体のすべての入力フィールドのサブセットを含み、フォーム全体のデータを外部データ構造に格納するマルチパートで指定できるようにする。

【0209】

NEXTメソッドは、以下の効果を有する。

1. ACTION

URLで使用するメソッドは、URLを何等変更しない「GET」であり、ページは単純にサーバからフェッチされる。GetURLは、プロトコルハンドラ112について呼び出される関数である。

【0210】

2. フェッチされたページにおける任意のフォームは、現行ページ上のフォームにおいてアクティブなname/valueデータセットで始まる。これは、先行ページから渡されたいずれのname/valueデータも含む。これは、「隠し」入力フィールドの使用を取り替え、データをサーバまで転送し、かつサーバにデータを再び転送させて、ページを返送する必要があることの帯域幅ペナルティを回避する。その代わり、ユーザがサービスを契約したときに、トランザクションを構成するその他のページと共にダウンロードされたページは関連するCGIスクリプトなしでサーバに常駐することができるか、または装置に常駐することができる。

【0211】

図18a～図18bは、図示した従来のHTMLページと同じ情報を取り込むが、画面表示136のはるかに有用であり、かつ使いやすいフォーマットを提示するマルチパートフォームのHTMLpソースおよびページを示す。最初の3つのページ、「purchaseform.html」、「addr.html」、および「credit.html」はすべて、<FORM>タグにおいてNEXTメソッドを用いて、さらなる入力を取得するために、次にロードされるページを指定する。最後のページ「confirm.ht

10

20

30

40

50

m l」は、マルチパートフォームに蓄積されたすべてのデータを処理するためのC G I スクリプトを指定するために、従来のA C T I O N値を用いている。この様式では、クライアント（無線通信装置100）が先行入力ページの状態データを維持するため、フォームへのすべての入力を取得して、それらをサーバに送信し戻すために、サーバとクライアントの間に必要な帯域幅は、はるかに低い程度である。氏名、住所、クレジットカード番号等のこの状態データは、メモリ126内の無線通信装置100の内部データ構造内で維持されるため、従来のH T M LのようにH I D D E Nタイプの入力要素に埋め込む必要がない。この内部データ構造は、最初のページがH T M L p コンテンツハンドラ114cによって構文解析される際に作成され、マルチページフォームがロードされる際にそれぞれの新しいページとして更新される。

10

【0212】

「c o n f i r m . h t m l」ページで使用される< T O P >タグおよび< D O N E >アクションについては、次の項で説明する。

【0213】

（v i i i）複雑な対話

デスクトップブラウザに表示される標準H T M Lでのマルチ要素フォームであったものを、ここでは、それぞれ1つまたは2つの要素フォーム（画面表示136に楽に収まるために）を含む、H T M L pでの複数ページに変形できるとすると、ユーザは、各フィールドについてデータを提供している途中で、自分がどこにいるのかを容易に知ることができ、かつプロセス全体を終了したいと容易に決定することができよう。ユーザが先行ページに戻ることはできない場合には、これは、トランザクションを締めくくるための遅くかつ骨の折れるプロセスであろう。

20

【0214】

より容易にマルチページを終了させるために、本発明は、ユーザが完全に終了したいと望む複雑な対話の開始として、ページをマークできるようにする。これは、対話の「開始（top）」の所望の場所に、< T O P >タグ：

< T O P >

を提供することによって行われる。このような対話は、マルチパートフォームであっても、またはページの任意の他の複雑な群であってもよい。

【0215】

対話の最初にアクセスするには、ソフトキー130がA C T I O N属性が「D O N E」である< K E Y >タグを用いて定義される。このソフトキー130がユーザにより選択されると、ユーザは、< T O P >でマークされた最も新しいページとしてユーザに参照されるページに戻ることになる。A C T I O NがD O N E値であるこの処理は、上述したように、H T M L p A c t i v a t e 関数中に行われる。

30

【0216】

（b）ナビゲーション

従来の無線通信装置100でのH T M Lの表示は、キーボードが厳しく制限され、かつポインティングデバイスがまったくないことにより、さらに妨げられる。通常のH T M L環境には、クリックまたはドラッグ可能なスクロールバーと、フォームのフィールド間をシフトするためのタブキーと、ユーザが辿りたいと望めば、コンテンツに含まれるハイパーリンクを選択することができるマウスとがある。

40

【0217】

しかし、本発明が有利に働く無線通信装置100では、ナビゲーション制御を提供するために頼ることのできるのは、アップキー、ダウンキー、およびソフトキー130のうちの可能な1つだけである。豊富なセットのナビゲーション能力を提供するために、本発明は以下の特徴をH T M L pに提供する。

【0218】

（i）フォームエントリ

ページがフォーム要素を含む場合には、アップおよびダウンアローが、以下の様式でフ

50

フォームフィールド間を移動できるようにするために、多重定義される。

・フォームにおいて次の（ダウンアロー）または前の（アップアロー）フィールドが見える場合、アクティブフォームフィールドになる。これは、グラフィカル選択インジケータによって表され、テキスト入力要素もまた、明滅するテキストカーソルを得る。

・フォームにおける次の（または前の）フィールドが画面上では見えない場合、ページの次の（または前の）行が画面上に入るように、画面がスクロールされる。次の（または前の）フィールドが画面上に入れられた行にある場合、アクティブフォームフィールドになる。そうでなければ、現行フォームフィールドがアクティブのままである。

【0219】

画面表示がスクロールするにつれ、現行フォームフィールドが、ユーザが直接該フィールドを選択する必要なく、連続して更新される。この様式において、ユーザは、記入すべき次のフィールドまで続く説明テキストを予想通りに読むことが可能でありながら、単にスクロールするだけでフィールド間を容易に切り替えることができる。

【0220】

これに加えて、最初のソフトキー130がページによってある他の目的にバインディングされていない場合、現行フォームフィールドが変化すると、キーバインディング表におけるアクションが以下のように変化する。

【0221】

【表1】

ソフトキーナビゲーションデフォルト

フォーカス付き ガジェット	ソフトキーラベル	アクション
テキストフィールド	ラベルなし	—
スクローリングリスト (1つだけ選択可能)	OK	選択および生成 提出
スクローリングリスト (複数選択可能)	Select	アイテムの選択/選択 解除
ポップダウン (閉)	Open	ポップダウンを開く
ポップダウン (開)	Select	ポップダウンを閉じ て、選択を行う
チェックボックス (選択済み)	Clear	選択解除
チェックボックス (未選択)	Select	選択
ラジオボタン (未選択)	Select	ボタン選択
ラジオボタン (選択済み)	ラベルなし	—
標準リンク	Select	リンクを起動
電話：ダイヤル・リンク	Call	番号を表示するダイ アル画面に進む (Sen dボタンで起動するこ ともできる)

【0222】

表1に定義される各種アクションは、現行フォームフィールドであるユーザインタフェースガジェットのタイプに応じて、ソフトキー130の適切で動的に可変な振る舞いについて提供される。これらのアクションは、ユーザがページをスクロールし、ガジェット間のフォーカスを変えることで、現行フォームフィールドが変わる際に、動的に割り当てられる。例えば、現行フォームフィールドがハイパーリンクである場合には、ソフトキーはリンクのURLに自動的に割り当てられ、ソフトキーを選択すると、ハイパーリンクが自動的にフェッチされる。フォームフィールドが、リスト、ポップダウン、チェックボックス、またはラジオボタン等、あるタイプの選択デバイスである場合には、ソフトキーは、適宜選択デバイスからアイテムを選択するか、選択解除するか、あるいは提出する。例えば、スクローリングリストにおいて、SELECTED属性（本発明の式評価特徴を有す

るかまたは有さない)を使用してアイテムが予め選択されている場合、ソフトキー130は、該アイテムを選択解除するよう定義される。

【0223】

ページナビゲーションに関するこの機能性は、HTMLpProcessKey関数において実施される。この関数は、現行ページの他のエントリが、入力キーストロークを受けるために選定されていない場合、シェル106によって呼び出される。該関数への入力には、キーパッド128のキーを示すキー番号と、該キーが押下されているか、解放されているかを示すブールである。図15を参照して、HTMLpProcessKey関数の一実施形態のフローチャートを示す。

【0224】

該キーに関連するURLがある場合、ProcessKey関数606は、シェル106のGetURL関数を呼び出して、関連するURLを渡す(1502)。シェル106は、図5に示すようにこのURLを処理して、該URLの処理に適切なプロトコルハンドラ112およびコンテンツハンドラ114を決定する。

【0225】

該関数は、キー番号から、該キーが、<KEY>タグ、あるいは<A>または<INPUT>タグのKEY属性を用いて、あるアクションにバインディングされているか否かを決定する(1502)。バインディングされている場合には、キーはソフトキー130(シェル106によって処理される)ではなく、アクションは、キーが解放されたときに、ShellActivateに渡される(1504)。

【0226】

そうではない場合には、アップおよびダウンキーのみが特別に処理され(1506)、他はすべて、各自のデフォルト処理を受けるために、ShellDefaultProcessKeyに渡される(1508)。

【0227】

アップおよびダウンキーの振る舞いは、ページに選択可能なユーザインタフェースガジェットがあるか否かによって左右される。選択可能なガジェットは、フォーム入力フィールド(INPUT、SELECT、TEXTAREA、またはOBJECTタグから)か、あるいはハイパーリンク(LINKMENUTagが存在する場合には)のいずれかである。ページに選択可能なガジェットがない場合(1509)、関数は所与の方向に次の行を見えるようにする(1514)。

【0228】

ページに選択可能なガジェットがある場合(1509)、アップおよびダウンキーに対する反応は、以下の通りである。選択された方向での次のユーザインタフェース下ジェットが見えている場合(1510)、関数はそのガジェットを現行ガジェットとする(1516)。次のユーザインタフェースガジェットが見えていない場合、所与の方向での次の行が見えるように、コンテンツエリア214がスクロールされる(1512)。これにより、次のガジェットが見えるようになると(1513)、それが現行ガジェットとなる(1516)。いずれのユーザインタフェースガジェットも見えない場合には、いずれのガジェットも現行ではない。

【0229】

(iii)メニューとしてのコンテンツ

フォームを含まないコンテンツは、大まかに2つのクラスにグループ化することができる。すなわち、1)読まれることを意図した情報コンテンツと、2)さらなる情報を得るために、またはあるアクションを行うために、ユーザがリストから何かを選択できるようにするメニューコンテンツと、である。

【0230】

ユーザが必要とするスクローリングおよびリンク選択の振る舞いは、これらのタイプのコンテンツそれぞれについて異なる。情報コンテンツでは、スクロールにより、次のテキストを読めるようにすべきであり(コンテンツのフォーカスとして)、リンクはいずれも

10

20

30

40

50

メニューから選択される（リンクは、二番目に重要である）。しかし、メニューコンテンツの場合、重要性が逆になり、テキストはリンクを説明するという役割を果たすが、ユーザが見る必要があるのはリンク自体である。このように、リンクが常に選択されるべきであり、発生するスクローリングはいずれも、次のリンクへ行き着くというコンテキストにおいて発生すべきである。従来のHTMLは、コンテンツのこれらのタイプを区別せず、それらの相違に適応するよう、コンテンツを表示するコンピュータのナビゲーション特徴を変更するための機能を提供しない。

【0231】

コンテンツのこれら2つのタイプを区別し、かつ望ましいナビゲーション制御を提供するために、本発明は、新しい<LINKMENU>タグを提供する。<LINKMENU>タグは、コンテンツが選択メニューであることを示すために、ヘッダに与えることができる。構文は、次の通りである。

<LINKMENU TARGET=nameNOSCROLL>

【0232】

TARGET属性は、ページ上のすべてのリンクのNAME属性とマッチする値を有する。name値がTARGET値とマッチするリンクは、ページが表示されるときに初期的に選択されているリンクである。TARGET値がURL（値の最初の部分が、コロンが後に続く英字キャラクタである）である場合、URLがフェッチされて、戻されたコンテンツがページ上のすべてのリンクのNAME属性と比較される。

【0233】

NOSCROLL属性が存在する場合、該属性は、選択されたリンクを見えるようにするために、表示をスクロールしないよう要求する。

【0234】

<LINKMENU>の指定は、以下の効果を有する。まず、リンクは、従来のHTMLのように、グラフィック的に目立たない（例えば、下線を付されない）。次に、ページの最初のリンクが選択されているものとしてマークされる（TARGET属性が与えられていない限り）。最後に、HTMLprocessKeyメソッドにおいて説明したように、アップおよびダウンキーが現行のユーザインタフェースガジェットを、見えていた前のまたは次のリンクにセットする。次のリンクは、現行リンクの下にあり、かつ水平距離にして最短距離にあるリンクとして定義される。これにより、リンクの列を優雅に、かつ予期されるように処理することができる。前のまたは次のリンクが見えていない場合、画面は適切な方向に一行スクロールする。そして、所望のリンクが見えるようになると、そのリンクが選択される。そうではない場合には、現行リンクが、現在見えている限り、アクティブのまま残る。

【0235】

<LINKMENU>がページに与えられていない場合、コンテンツは以下のように扱われる。まず、リンクに下線を付すことで、リンクを目立たせる。次に、最後のソフトキーがいずれにもバインディングされていなければ、ページにおけるすべてのリンクが、そのソフトキーにバインディングされたメニューに収集される。最後に、アップおよびダウンセレクトが一度にテキスト一行ずつページをスクロールする。

【0236】

<LINKMENU>タグのこの機能性は、HTMLpコンテンツハンドラ114cにより、該ハンドラがページを構文解析して、キーバインディング表およびメニューをセットするときに、発効される。

【0237】

図19は、<LINKMENU>が指定されない第2のタイプの振る舞いの例を示す。この例では、iguana.gifおよびlizards/blue-tongued-iguana.html等のファイルまたはデータベースデータを含む他のコンテンツへのリンクはすべて、2番目のソフトキー130にバインディングされる「Links」という名称のメニューに自動的に配置される。この

10

20

30

40

50

メニューは、ソフトキー 130 が押下されると、2 番目のイメージに示すように表示される。

【0238】

図 20 は、<LINKMENU> が指定される第 1 のタイプの振る舞いを示す。なお、ページにおけるリンク、例えばUA 909には、従来の HTML でのように、下線が付されない。むしろ、アップおよびダウンアローがこれらのリンクを順序通りに動かして選択する。

【0239】

(iii) リンクのキーへのバインディング

本発明は、<A> タグに新しい属性を提供する。これらの属性は、標準 HTML ブラウザは該属性を無視するため、無線通信装置 100 およびデスクトップコンピュータの双方にコンテンツを提供する互換性のある方法を提供する。これらの属性は、KEY および LABEL 属性である。

【0240】

指定される場合、KEY 属性は、指示されたキーが HREF についてこれにより指定された URL を辿るよう求める。「back」キーは、このようにしてバインディングしていなくてもよい。したがって、この属性は、特定のキーを特定の URL にバインディングするための手段を提供する。

【0241】

指定される場合、LABEL スtring は、2 つの方法の 1 つで使用される。

KEY も提供され、かつこれはソフトキー 130 である場合、String は、画面表示 136 上に表示されるソフトキー 130 についてのラベルである。

KEY が提供されない場合、String は、LINKMENU タグが与えていないときに、HTML p コンテンツハンドラ 114c によって自動的に構築された、リンクのメニューに使用されるラベルである。LABEL 属性がない場合には、リンクのテキストが使用される（あてはまるかぎり）。

【0242】

(iv) キーの入力要素へのバインディング

従来の HTML は、<INPUT> タグに SUBMIT および RESET 属性を提供する。しかし、これらの属性は、リターンキー、またはユーザインタフェースガジェット上のマウスクリックのいずれかにハードコードされる。

【0243】

本発明は、バインディングしたい所望のキーを指定する<INPUT> タグの KEY 属性を使用して、特定のキーにもこれらをバインディング可能にすることで、SUBMIT および RESET 入力要素の使用を拡張する。

【0244】

好ましい実施形態において、デフォルトにより、SUBMIT 要素は第 2 のソフトキー 130 にバインディングされ、かつ RESET 要素は 3 番目のソフトキー 130 にバインディングされる。装置が 2 つのソフトキー 130 しか持たない場合、RESET 要素にはアクセス不可能である。しかし、これらの装置の形式が単純であるとすれば、通常、このことは問題ではない。

【0245】

複数の SUBMIT 要素を有するフォームは、明示的なキーバインディングが与えていない限り、それらすべてを第 2 のソフトキー 130 上のメニューに配置される。同じキーにバインディングされた複数の SUBMIT または RESET 要素は、そのキー上でメニューに組み合わせられる。

【0246】

(c) 特殊化されたコンテンツ

本発明はまた、特殊なタイプのコンテンツをサポートするか、または既存のワールドワイ

10

20

30

40

50

ドウェブでの実施を無線通信装置 100 のより小型の画面表示 136 にマッピングするための方法を提供するために、HTML へのさらなる拡張を含む。

【0247】

(i) 電話のダイアリング

他の電話型製品のように、無線通信装置 100 は、DTMF ベースのネットワークサービス、またはボイスメールシステム等の、機能性制御に DTMF トーンを使用する他のシステムにアクセスすることができる。したがって、HTML p は、かかるシステムと容易にインタフェースするために、ページのフェッチ時に、DTMF トーンの生成を非常に容易にする新しいタグを含む。これは、新しい<DIAL>タグを使用して達成される。

<DIALNAME=stringICON=numberNOSCREENCHANGE>(n|n@t;)+</DIAL>

10

【0248】

n は番号または特別なダイアルコード (pause の p のように) であり、@t ; は、もしあれば、10 秒単位で継続期間を指定する。DTMF トーン生成と、新しく電話をかけることとの間の選択は、通話が現在アクティブ (保留されていない) であるか否かに基づいて行われ、通話がアクティブである場合には、DTMF トーンが生成される。

【0249】

NAME および ICON 属性は、電話をかけている間に表示されているページおよびダイアルした番号を容易に電話帳に配置可能とする追従ページにおいて使用するため、電話をかける相手を指定する。ICON 属性もまた、通話接続ページにおいて使用され、その主な目的は、ユーザがダイアルした番号を電話帳に格納できるようにする通話に追従するページにある。電話番号は、埋め込み電話帳オブジェクトおよび他の場所にある各自のアイコンによって識別され、実際の番号入力時にユーザがアイコンを変更しない場合には、ICON 属性が使用するアイコンを指定する。

20

【0250】

NOSCREENCHANGE 属性は、通話が成功した後に、表示が、図 22a ~ 図 22c 等の標準の通話管理ページに変わるのではなく、このページに戻るべきであると指示する。

【0251】

非特権ページによる実際に新しく電話をかけようとする試みはいずれも、不当なコンテンツが望ましくない電話をかけないようにするため、ユーザに確認される。

30

【0252】

このタグは、ユーザインタフェースページが、既存の DTMF ベースのコマンドツリーにグラフィカルな命令シートを提供できるようにする。ユーザは、画面上の機能を起動し、それから、画面が、ユーザが要求したアクションを実行するために適した DTMF トーンを生成すると共に、新しい状態でユーザが利用可能な操作を表示する新しいページを提示する。図 25 は、このタイプの使用例を示す。「VoiceBox」というタイトルのファイルは、ボイスメールシステムにアクセスするためのユーザインタフェースページである。2 行目の<DIAL>タグは、ボイスメールシステムのための電話番号と、ユーザパスワード「4416722」とを含む。4 ~ 8 行目は、該システムの各種機能を起動するためのキーパッド 128 のキーを定義する。「Listen」というタイトルのファイルでは、2 行目の<DIAL>タグはまず、ボイスメールシステムをトリガして、再生モードに入らせる番号「5」に対応する DTMF トーンを生成する。4 ~ 8 行目はここで、各番号キー 134 に、それぞれボイスメールシステムのさらなる機能を生成する特定の番号をダイアルするための URL である各種アクションを割り当てる。したがって、<DIAL>タグを使用することで、正確な基礎をなす DTMF 信号を提供しながら、ユーザがグラフィカル的にボイス応答システムのコマンドツリーをナビゲートできるようにする。

40

【0253】

<DIAL> は、HTML p コンテンツハンドラ 114c によってシェル 106 に提供され、シェル 106 は次に、それを処理および URL に提供された番号に対応する DTMF トーン生成のために、電話プロトコルハンドラ 112g に提供する。

50

【0254】

(i i) 広告コンテンツ

既存のワールドワイドウェブコンテンツは、広くグラフィカル広告バナーによってサポートされている。しかし、無線通信装置100の画面表示136の帯域幅および画面サイズは限られており、従来のイメージ集約的な広告バナーが無線通信装置100上に適切に表示されないため、この種の広告を問題のあるものになっている。

【0255】

本発明は、既存の<MARQUEE>タグに対する拡張を提供することで、この制限を克服する。このタグは、通常、画面にわたってスクロールすべきであるテキストを指定するが、ページの<BODY>セクションに限られている。従来、<MARQUEE>タグが<HEAD>セクションに配置される場合、従来のブラウザはタグを無視する。

10

【0256】

本発明において、HTML p コンテンツハンドラ114cは、<MARQUEE>タグをドキュメントの<HEAD>セクションに配置できるようにする。そのように用いられると、<MARQUEE>タグにおける付随の広告テキストがページのタイトルおよび<HELP>タグで指定されているいずれの遅延ヘルプに取って代わる。この機能性は、遅延ヘルプを表示していなときをHTML p コンテンツハンドラ114cに通知するよう求められるシェル106からの呼び出しに応答した、HTML p コンテンツハンドラ114cにより、実施される。通知されると、HTML p コンテンツハンドラ114cは、シェル106にタイトルエリア210にわたって広告テキストをスクロールするよう命令する。

20

【0257】

要約すれば、HTML p コンテンツハンドラ114cおよびHTML p 拡張は、従来のHTMLに存在しない多くの有利な特徴および関数を提供する。

【0258】

(b) 広告マネージャ

広告マネージャコンテンツハンドラ114aは、アイドル時に表示する広告を選択するとともに、古い広告、または要求した回数だけ応答されたか、あるいは実行された広告を削除する。広告は、HTMLまたはHTML p ページとして定義され、メモリ126にユーザインタフェース定義ファイル104の一部として格納される。広告は通常、操作者によりスケジュールをベースにして無線通信装置100にダウンロードされる。ユーザが広告に応答するか、または広告がある回数だけ表示された後、あるいはより重要な広告が来た場合、ダウンロードされた広告は自動的に削除される。

30

【0259】

広告マネージャコンテンツハンドラ114aは、以下のような基本的なコンテンツハンドラ関数を実施する。

【0260】

(1) AdvertOpen

広告マネージャコンテンツハンドラ114aがページを一時的なものとする、と、広告ページは再度実行されることは決してなく、したがってURL履歴スタック108に格納されない。これは、現れる任意の他のページ（例えば、着信、低バッテリー等）がURL履歴スタック108において広告ページと置換されることを意味する。

40

【0261】

AdvertOpenは、呼び出されると、使用可能な広告ページのセットを見て、表示する1つを選択する。それは、ファイルを開いて、ストリームを表示のためにHTML p コンテンツハンドラ114cに渡す。

【0262】

(2) AdvertClose

この関数は、表示していた広告ページをチェックして、削除するようマークされている場合には（応答されたか、あるいは要求した回数だけ表示されたため）、該広告ページを

50

削除する。

【0263】

(3) AdvertActivate

ストリングは、HTMLpコンテンツハンドラ114cに渡されて、ユーザが応答したために、現在の広告を削除するものとしてマークした後に、HTMLpActivateにおいて処理される。

【0264】

(4) AdvertProcessKey

そうでなければ広告ページにバインディングされていない任意のキーを押下すると、ページを閉じることになり、かつキー押下を広告ページが現れる前にアクティブであったページにより再度処理されることになる。

10

【0265】

c) コールマネージャ

コールマネージャコンテンツハンドラ114bは、2つの目的のために使用される。

1. アクティブコールを表示するため。
2. 発信の接続進行を表示するため。

【0266】

コールマネージャコンテンツハンドラ114bを現在使用している、URL履歴スタック108におけるページは常に1つしかなく、該ページは、これら2つのモードの1つである。コールマネージャコンテンツハンドラ114bは、以下のような基本的なコンテンツハンドラ関数を実施する。

20

【0267】

(1) CallManagerOpen

該ページが再度実行される場合、この関数は、ページの表示前に作成したウィンドウを再び見えるようにし、そのコンテンツを再度表示する。

【0268】

これが、ページが開かれる最初である場合、CallManagerOpenは、URLについてのデータストリームを検査して、ダイアルすべき電話番号があるか否かを見る。ストリームにおいて、ストリングの形式は：

num=string&name=string&icon=n

30

である。

【0269】

これは、関数に、ダイアルする番号（もしあれば、DTMFトーンが続く）、番号について表示する氏名（番号自身が電話帳にない場合）、および氏名と共に表示するアイコンを教える。

【0270】

ダイアルする番号がある場合、コールマネージャコンテンツハンドラ114bは、電話制御モジュール120によって提供されるインタフェースを使用してダイアリングモードに入り、電話をかける際の進行を示すダイアリングページを表示する。このページは、いずれのDTMFトーンをダイアルしているかについてのフィードバックを提供するとともに、もしエラーが生じたなら、エラーを報告する。図21a～図21eは、「connecting（接続中）」、「line is busy（お話中）」等の接続状態を示すダイアリングページ例を図示する。図20c～図20eでは、ダイアルしている電話番号全体を示し、ダイアルしている現在の数字の下に移動するインジケータを付けている。

40

【0271】

ダイアルする番号がない場合、コンテンツハンドラ114bのCallManagerOpen関数は、アクティブコールのリストならびにそれら进行操作するためのインタフェースを表示する。アクティブコールが1つだけしかない場合には、より単純なインタフェースが表示される。図21a～図21cは、これらのインタフェースを示す。図21aは

50

、アクティブコールが1つの場合のインタフェースを示し、図21bは、複数のアクティブコールがある場合のインタフェースを示し、図21cは、図21bと同じインタフェースを示すが、電話会議にするか、または呼を保留するかを選択できるようにするソフトウェア130のメニューを備える。

【0272】

(2) CallManagerClose

ページが永久的に閉じられ、かつURL履歴スタック108から削除される場合、これは、アクティブコールを表示するために割り当てていたリソースを解放することになる。アクティブコールがまだある場合、この関数は、ユーザ入力がないある量の時間後、アクティブコールのリストが再度表示されるように、ShellAddIdleHookを呼び出す。

【0273】

(3) CallManagerActive

この関数は、選択された呼に関してどのアクションが利用可能であるかに応じて、ソフトウェア130にバインディングされた以下のコマンドを探す。

[Conference] 多数共同呼(multi-party call)において他の呼(保留)を現在の呼と合わせる。

[Spilit] 参加中の多数共同呼から選択された呼を除去する。

[Hold] 選択した呼または多数共同呼を保留にする。

[Pick Up] 保留していた呼または多数共同呼をアクティブにする。

[Back] コールマネージャ画面を閉じる。

[Stop] 現行のDTMFシーケンスを停止して、進行中の通話画面の代わりにアクティブコールのリストを表示する。

【0274】

(4) CallManagerProcessKey

この関数は、多数共同呼の場合にアップおよびダウンキーを処理して、ユーザ選択のある呼から次の呼へ移動させる。選択された呼は、アクティブコールとなり、旧アクティブコールが保留になる。

【0275】

数字キー134は、アクティブコールが選択される場合にDTMFトーンを生成し(現在選択された呼をアクティブにしないことは不可能であることを示すように見えるアップおよびダウンキーへの応答に係わらず、ユーザがただ保留するよう求め、選択を変えない場合には、選択した呼を保留することが可能である)、そうでない場合にはダイアラー画面が提示され、電話をかける最初の番号として数字が入力される。

【0276】

Endキーは、現在の呼を終了する。呼が多数共同呼の一部である場合には、終了する前に会議(conference)から除去される。

【0277】

Sendキーは、ダイアラー画面を提示するが、ユーザがSendを押下して次に電話をかけるまで、現行の呼に影響を与えない。

【0278】

d) メインコンテンツハンドラ

メインコンテンツハンドラ114dは、装置のメインページを表示するために、HTMLコンテンツハンドラ114cのフロントエンドとしての役割を広く果たす。

【0279】

(1) MainOpen

ユーザがキーを押下し始めると、番号をダイアルし始めるように、入力モードを数字およびシフト状態にリセットする。

そして、メインページへのストリームを開き、ストリームをHTMLコンテンツハンドラ114cに渡して、表示する。メインページのサンプルを図22aから図22cに示す

10

20

30

40

50

。

【0280】

(2) MainClose

同じ引数を有するHTMLpCloseを呼び出す。

【0281】

(3) MainActive

同じ引数を有するHTMLpActivateを呼び出す。

【0282】

(4) MainProcessKey

キーがEndであり、かつアクティブコールがある場合、この関数はアクティブコール画面を提示し、それが現れるのを待つ必要なく、ユーザが電話を切ることができるようにする。

10

【0283】

任意の他のキー押下は、HTMLpProcessKeyに渡されて、処理される。

【0284】

7. プロトコルハンドラ

本発明によれば、無線通信装置100の機能性は、いくつかのプロトコル、およびかかるデータまたは関数を識別するURLに応答して、データをフェッチまたは通知(post)したり、要求された関数を実行したりするプロトコルハンドラ112を介してアクセスされる。

20

【0285】

好ましい実施形態では、無線通信装置100と対話するための3つのメインプロトコル、すなわち、電話、メッセージ、および構成がある。4番目のプロトコルである「エクストラ」プロトコルは、<TEMPLATE>タグおよび他の特徴とともに記載したように、HTMLpテンプレートページを大部分のユーザインタフェースに使用できるようにする。

【0286】

各プロトコルについて、サポートされたURLは、HTMLまたはHTMLpページに埋め込まれるオブジェクトを返すもの、およびコンテンツを表示するか、または無線通信装置100の機能を起動するものに分けられる。

30

【0287】

a) 電話プロトコル

これは、装置の特徴、およびHTMLよりはむしろCで書かれた各種埋め込みオブジェクトにアクセスするための主要なプロトコルである。電話プロトコルハンドラ112gによってデコードされる。これらの埋め込みオブジェクトとしては、上述した電話帳オブジェクト、最近の通話リストオブジェクト等が含まれる。各埋め込みオブジェクトは、パラメータ幅および高さを有する。これらのパラメータは、埋め込みオブジェクトについてURLにおいて指定され、その出力を表示するために、埋め込みオブジェクトによって提供されるウィンドウのピクセル幅および高さを定義する。

【0288】

各埋め込みオブジェクトはまた、URL仕様に従って実行するメソッドのセットも有する。

40

【0289】

(1) 埋め込みオブジェクト

各埋め込みオブジェクト毎に、それが許容するパラメータならびにそれに対して行うことができるメソッドが記述されている。これらのメソッドは、キー128、ハイパーリンク、またはソフトキーメニューエントリにバインディングされたストリングである。

【0290】

phone:dialing?width=number&height=number&storekey=n,storelabel/editlabel

電話をダイアリングするための埋め込みオブジェクトを含むストリームを返す。該オブ

50

ジェクトは、番号により、または氏名により格納された電話帳データ構造におけるエントリを索引でき、かつその入力フィールドの下に、`<INPUT>` タグの `phonename` および `phonenum` 属性に関して、上述したようにマッチしたものを表示する。

パラメータ

[storekey] ユーザがマッチリストにおけるエントリを選択した場合には、ユーザがエントリを編集できるようにするために、またはユーザが入力した番号または氏名を格納できるようにするために、いずれのソフトキー 130 を使用すべきかを指定する。storekey パラメータについて指定される storelabel および editlabel 値は、ソフトキーが以下の 2 つの機能のいずれかを行うようセットされる場合に、ソフトキーに与えられるラベルを指定する。

10

メソッド

[edit] マッチリストで選択された番号を編集するよう要求する。

[store] ユーザがタイプしたものから適切なフィールドを記入した新しい電話帳エントリシーケンスに入ること、ユーザがタイプしたものを電話帳に格納するよう要求する。現行テキストエントリモードが数字である場合、入力されたデータは電話番号としてとられ、現行テキストエントリモードが非数字である場合には、氏名ととられる。

【0291】

また、「@」キャラクタを含むアクションはいずれも、このオブジェクトのメソッドとして考慮される。phone:list オブジェクトについて記述された @ エスケープは、ここでもまた機能する。

20

【0292】

phone:list?width=number&height=number&editkey=n,label&service=string&data=string&showstatus

電話帳データ構造からのレコードを表示するために、電話帳埋め込みオブジェクトを含むストリームを返す。

パラメータ

[Editkey] 何が選択されるかに応じて（電話番号のみを編集できる）、ラベルおよびアクションを変更すべきソフトキー 130 を指定する。

[Service,data] サービス (URL) またはデータ (格納されたコンテンツ) エントリが選択されるときに、ステータスメッセージエリア 212 に表示するストリングを指定する一方、showstatus は、任意のステータスメッセージ 212 が表示されているか否か（電話番号エントリの場合、電話番号が表示される）を決定する。

30

メソッド

[New] リストが検索モードであり、かつマッチリストにおけるエントリが選択される場合、氏名、番号等を収集し、結果を電話帳に格納するユーザインタフェース定義ファイル 104 のシーケンスである、新しい電話帳エントリシーケンスが、選択されたエントリの氏名を新しいエントリの氏名として使用して入力される。マッチリストにおいてエントリが選択されない場合、新しい電話番号エントリシーケンスは、ユーザが初期データとしてタイプしたものと共に入力される。データは、現行テキストエントリモードが数字である場合には電話番号のために、現行テキストエントリモードが非数字である場合には氏名として使用される。

40

[Edit] 現行電話帳エントリを編集するために URL が生成され、URL がフェッチされる。

[Delete] ユーザに要求を確認した後、現行電話帳エントリを削除する。

【0293】

また、「@」キャラクタを含む URL はいずれも、このオブジェクトのメソッドとして考慮され、かつ続くエスケープについて検索される。エスケープが見つかったら、エスケープが現行選択からの関連データにより置換される。すべてのエスケープが一旦置換されると、結果得られる URL がフェッチされる。

【0294】

50

【表 2】

エスケープ値

エスケープ	置換するもの
@i	選択されたエントリのレコードID
@o	エントリ（0～7）において選択された電話番号のオフセット
@n	選択されたエントリにおける氏名
@N	選択された電話番号（選択されたアイコンが電話番号ではない場合、エラーを生じさせる）
@I	選択された電話番号のアイコン（選択されたアイコンが電話番号ではない場合、エラーを生じさせる）
@S	選択された電話番号の短縮ダイヤル番号（選択されたアイコンが電話番号ではない場合、エラーを生じさせる）
@r	選択されたエントリにおけるすべての番号の呼び出し音
@R	選択されたエントリにおけるすべての番号の優先呼び出し音
@c	選択されたエントリのカテゴリ
@g	選択されたエントリのカテゴリ内のグループ
@@	1つの「@」キャラクタ

10

【0295】

phone:recentcall?filter=number&editkey=n,editlabel/storelabel&callkey=n,label&width=n&height=n&noempty

最近受信した、または最近かけた電話のリストを表示することができる埋め込みオブジェクトを含むストリームを戻す。呼のリストにおける各アイテムは、それに関連するフラグのセットを有する。オブジェクトに、あるフラグセットを有するアイテムのみを表示するよう指示することができる。現在定義された1つのフラグはビット0であり、ユーザが電話にできることができなかった番号から電話をかける。

パラメータ

`filter`が存在する場合、それは特定タイプの呼のみを表示すべきであることを示す（1つのみの現行フィルタの値は1であり、これは、出ることのなかった着信を意味する）。`editkey`および`callkey`は、選択された呼に基づいて、ラベルおよびアクションを変更すべきソフトキーを指定する。`noempty`が存在する場合、それは、呼のリストが、フィルタリング後に空である場合には、オブジェクトを含む画面を表示すべきではないことを示す。

メソッド

[edit] 現行選択が電話帳にある番号である場合、番号についての電話帳編集画面を提示する。

[store] 現行選択が電話帳にある番号ではない場合、電話番号を現行選択にセットした、新しい電話帳エントリシーケンスを提示する。

[call] 現行選択されている番号への呼を示す。

[dismiss] リストにおけるすべてのアイテムについて、「missed」フラグ（ビット0）がクリアされた後に、画面を閉じる。

【0296】

phone:ringtone?width=number&height=number&ptrAddr=hex

着信に対して装置が生成することのできる呼び出し音のリストを表示できる埋め込みオブジェクトを含むストリームを戻す。

パラメータ

`ptrAddr`が与えられる場合、オブジェクトは、ユーザにシステムデフォルト呼び出し音を使用するというオプションを与える。ユーザが呼び出し音を選択すると、次にその番号が`hex`で与えられるメモリアドレスに配置される。`ptrAddr`が与えられない場合、オブジェクトはシステムデフォルト呼び出し音をセットする。

メソッド

[test] 選択された呼び出し音を一度通して再生する。

20

30

40

50

[ok] 呼び出し音を選択されたものにセットし、画面を閉じる。

【0297】

phone:speeddial?width=number&height=number&ptrAddr=hexnumber&Name=string&icon=n
短縮ダイヤルロケーションのリストを表示するための埋め込みオブジェクトを含むストリームを戻す。

パラメータ

ptrAddrは、ユーザがエントリを選択したとき、オブジェクトがどこに選択された短縮ダイヤル番号を格納するかを指定する。Nameおよびicon引数は、phone:storeのものと同一であり、短縮ダイヤル番号が割り当てられた番号の氏名およびアイコンを指定する。

メソッド

[Ok] 現行選択が現在別の番号に割り当てられていない場合には、短縮ダイヤル番号を現行選択にセットし、画面を閉じる。Nameおよびicon引数は、ユーザに対する確認画面を構築する際に使用される。

[Clear] 選択された短縮ダイヤルエントリの割り当てをクリアする。

【0298】

(2) コンテンツ／コマンドURL

phone:active

アクティブコール画面を表示させて、ユーザがアクティブであり得るいずれの呼も操作できるようにする「CallManager」型のストリームを戻す。いずれの呼もアクティブではない場合には、何も行わない。

【0299】

phone:answer

任意の着信に応答させ、アクティブコール画面を表示させるストリームを戻す。着信がない場合には何も行わない。着信があった場合には、参照するURLがURLスタックからポップする。

【0300】

phone:conf

任意の着信に応答させ、かつ現行アクティブコールと合わせる。アクティブコール画面を表示させるストリームを戻す。着信がない場合には何も行わない。着信があった場合には、参照するURLがURL履歴スタック108からポップする。

【0301】

phone:dial?num=string&name=string&icon=number&hidden

指示された番号に、ボイスコールを生成させる。nameおよびiconが提供される場合、これらは呼進行を表示するページ、およびユーザが番号を電話帳に追加したいか否かについて訊ねられる任意の追従ページにおいて使用される。hiddenが指定される場合、ユーザは、呼が一旦接続されると、呼を見ることがない（アクティブコール画面が表示されない）。URLは、呼進行画面を表示させるストリームを戻す。

【0302】

引数が与えられない場合、URLは、ユーザが電話する電話番号を入力できるようにする、ダイヤラー画面を表示させるストリームを戻す。

【0303】

要求を発しているページが十分な特権を欠く場合、ユーザは、電話番号のダイヤルを許可するか否かについて訊ねられる。例えば、受信したテキストメッセージは、ユーザが気付かない900番や長距離の番号への電話を含みうるため、十分な特権を欠く。

【0304】

phone:display?id=number

指定された電話帳レコードのデータフィールドを表示するストリームを戻す。

【0305】

phone:edit?name=string&num=string&offset=number&id=number

10

20

30

40

50

渡されたパラメータと共に、電話帳の新規／編集画面を表示させるストリームを戻す。
offsetおよびidは、既存のレコードを編集するために、内部でのみ使用される（offsetはいずれの電話番号を編集するかを示す一方、idは編集すべき記録の識別子である）。しかし、nameおよびnumは、新規電話帳レコードの作成に使用することができ、ユーザに適切なアイコンを選択する機会や、別の場合には、電話帳に格納する前にエントリを編集する機会を与える。

【0306】

phone:firstopen?password=string

与えられたパスワードstringを構成設定に格納されたものに対してチェックする。マッチすれば、参照画面がポップされ、phone:mainで置換される。マッチしなければ、電話をオフにする。これにより、電源オンセキュリティ画面をHTMLページにすることができる。

10

【0307】

phone:ignore

任意の着信を拒絶させる。着信がある場合、参照しているURLがURL履歴スタック108からポップされる。着信がない場合には、何も行わない。いずれのストリームも戻されず、そのため、参照しているURLをフェッチする前にアクティブであったURLが再度表示される。

【0308】

phone:indir?url=string&pop=action

最後にフェッチしたものからストリームを戻す前に、1つまたは複数のURL（各自の副作用を、それが何であれ、起動する）をフェッチする。pop引数を与えられる場合、それは、戻されたデータを表示する前に、1つまたは複数のURLをURL履歴スタック108から削除すべきであることを示す。actionは、1つのURL、現行の対話にあるすべてのURL、または履歴スタック108からのすべてのURLを削除するために、pop、abort、またはclearのうちの1つであることができる。

20

【0309】

phone:look?cat=number&sort&form=format

番号がcat引数によって与えられるカテゴリにある電話帳から氏名を検索する。sort引数を与えられる場合、結果はアルファベット順にソートされる。form引数は、データを提供すべきフォーマットを指定する。これらは常に、ある形式のHTML（このURLの結果はHTMLストリームである）であるが、使用されるタグは以下のように様々である。

30

- ・link:各エントリ名が、適切な場所へのリンクとしてフォーマットされる。サービスフィールドを有するエントリの場合、HREFがサービスフィールドのコンテンツであり、一方、データフィールドを有するエントリの場合、HREFがそのデータを示す。他のすべてのエントリは、エントリの最初の番号をダイアルする。

- ・form:nameが<SELECT>リストの<OPTION>要素（これは、このデータをフェッチする<INC>タグを取り囲まなければならない）としてフォーマットされる。各オプションの値は、リンクについて上述したように、URLである。

40

- ・menu:各エントリは、リンクの場合と同様にして<KEYMENU>タグとして提供される。使用するキーは、form=menu=xで指定されるが、但しxはキーである。

- ・count:所与のカテゴリにあるレコードの数を生成する。

【0310】

phone:look URLは、新しい<INC>タグと組み合わせられると、HTMLページがある種のブランド化したグラフィカルコンテキストで電話帳のサブセットを表示できるようにする。より重要なこととして、それは、サービス運営者およびユーザの双方に対して、いずれのサービスがユーザにとって利用可能であるかを管理する単純な方法を提供する。サービスのグループは、特定のカテゴリで電話帳に記録される。そして、装置は、このURLを使用して、これらのエントリを表示すると共に、ユーザが1つのサー

50

ビスを選択できるようにするページを有する。サービスの追加および削除は、単にエントリを電話帳に追加またはそこから削除するだけの問題であり、サービスのリストを表示するページを変更する必要はない。

【0311】

phone:main

所定のメイン画面を表示させるストリームを戻す。

【0312】

phone:release?id=number

i d 引数が与えられる場合には、アクティブコールまたは指定したコールを解除する。何のデータも戻さない。

【0313】

phone:shortcut?num=n

ショートカット機能を起動する。n は、0～9 の範囲である。

【0314】

ショートカットは、無線通信装置 100 の製造業者によって定義され、通常、ソフトキー 130 の 1 つを押下したまま、次に数字キー 134 の 1 つを押下することで、起動される。これらは、一般に、電源オンパスワード画面を除き、すべての画面から利用することができる。

【0315】

例えば、ショートカット 1 は、キーパッドをロックまたはロック解除しうる一方で、ショートカット 2 は、電話の呼び出し音をミュートにし、ショートカット 3 は、無線通信装置 100 がオンされたときに、パスワード保護を起動または非起動としうる。あるショートカットはまた、電源オンパスワード画面以外のある画面に限定されることもある。

【0316】

phone:store?Name=string&Phone=speed=icon=string&Owner=string&Service=url&Category=number&Group=number&Ring=number&Data=type%0adata&speed=speed&icon=icon&id=id&offset=number&prio=n&pop=string

電話帳におけるレコードを作成、増補、または編集する。i d および o f f s e t が指定される場合、レコードにおける選択された電話番号が編集される。他の場合、同じ N a m e のレコードがすでに存在していれば、レコードが増補される一方、マッチする N a m e がなければ、新しいレコードが作成されることになる。

パラメータ

[Phone] このパラメータのコンテンツは、「s p e e d」および「i c o n」パラメータが指定されているか否かによってかなり異なる。これらが指定されていなければ、このパラメータは、「speed=icon=number/speed=icon=number/…」の形式のストリングを含む。換言すれば、それぞれに短縮ダイアルロケーションおよびアイコンを関連させた、1 つまたは複数の電話番号を指定するストリングである。「s p e e d」および「i c o n」が指定されていれば、このパラメータは、電話帳に格納する単一の電話番号のみを含む。

[Owner] エントリのコンテンツを電波を介して更新可能とするパスワードを指定する。

[Service] エントリに格納すべき U R L を指定する。この U R L は、電話帳では電話番号のように現れるが、S e n d キーが押下されると、U R L のデータがフェッチされる。

[Category] エントリを通常の電話帳画面から隠すが、H T M L ページに包含する場合には、p h o n e : l o o k によって見つけられるようにする。C a t e g o r y 0、1、および 2 は、通常の電話帳により表示される。

[Ring] この電話帳レコードにおける任意の番号からの着信時に使用する呼び出し音（0～127、0 はシステムデフォルトの使用を意味する）を指定する。高ビット（32768）がセットされる場合、それは、装置が静止モード（quiet mode）である場合であっ

10

20

30

40

50

ても、この電話帳レコードにおける任意の番号からの着信があれば、呼び出し音を鳴らすべきであることを示す。高ビットはまた、`p r i o`引数を用いてセットすることもできる。これが1である場合、高ビットがセットされることになる。

[Data] 任意のデータを電話帳レコードに格納できるようにする。データは、電話帳では電話番号のように現れるが、`S e n d`キーの押下時には、データが表示される。引数値の最初の部分は、データ型である。通常これは、テキストあるいはHTML pのいずれかである。2番目の部分は、16進数のストリングとしてのデータ自身であり、オクテット当たり2つを格納する。

[Pop] (オプション) `p h o n e : s t o r e U R L`を要求したURLを、string値に従ってURL履歴スタック108から削除させる (pop⇒そのURLだけを削除する、`d o n e`⇒最も最近のもので、一番上のURL以降のすべてのURLを削除する、`clear`⇒メイン画面まで遡るすべてのURLを削除する)。

【0317】

この特定のコマンドは、ユーザにかなりの柔軟性を提供する。第1に、DATA引数は、電話番号だけではなくいずれのデータも電話帳に格納できるようにする。特に、URL、イメージ、オーディオデータ、および任意の他のコンテンツを、無線通信装置100に汎用データベースを作成することで、格納することができる。例えば、ユーザは、ウェブコンテンツを閲覧し、表示されているURLを選択し、かつそれを後に再度呼び出すために、電話帳に即座に格納することができる。

【0318】

第2に、RING引数は、電話帳エントリそれぞれに異なる呼び出し音を指定できるようにする。このRINGトーンは、着信が電話帳エントリ内の任意の番号から受信したときに使用される。これにより、ユーザが、各種電話番号に特定の異なった呼び出し音を指定することができる。例えば、ユーザは、異なる家族の面々、同僚、医者のおフィス毎に特定の呼び出し音を指定することができる。

【0319】

第3に、RING引数はまた、ringトーン値に高ビットをセットするか、またはPRIO引数に非ゼロを指定することで、任意の電話帳エントリおよびその呼び出し音について優先呼び出しを可能にする。従来の無線通信装置は、通常、電話を無音にして、通常着信音が鳴らないようにする静止モードを含む。本発明では、このRING引数は、そのようにブロックせずに、音が鳴るようにさせる電話帳エントリからの呼を指定することができる。したがって、ユーザは、この優先呼び出しを家族の面々、および他の大事な人についてセットすることで、静止モードであっても、かかる人々からの電話を鳴らすことができる。

【0320】

コールマネージャコンテンツハンドラ114bは、各着信の電話番号を、格納された電話番号と比較し、かつ指定した呼び出し音（もしあれば）を使用して、電話の呼び出しを選択かつ制御することで、この特徴を実施する。

【0321】

b) メッセージプロトコル

テキストメッセージは、英数字ページと同様に、本発明を用いて受信し、閲覧することが可能である。メッセージは、ファイルに格納され、一意の識別子により識別される。このプロトコルは、メッセージプロトコルハンドラ112fによって処理される。

【0322】

(1) 埋め込みオブジェクト

`message:list?width=number&height=number&type=type&num=string&lockkey=n,locklabel/unlocklabel`

所与のタイプのメッセージリストを表示する埋め込みオブジェクトを含むストリームを戻す。

パラメータ

10

20

30

40

50

[num] (オプション) 所与のソースからのメッセージのみを表示するよう指示する。

[lockkey] (オプション) 選択されたメッセージが現在ロックされているか否かに基づいて、いずれのソフトキー130を更新すべきかを指定する。そのソフトキーを押下すると、メッセージのロック状態がトグルされる。パラメータのlocklabelおよびunlocklabel部分は、実施する機能に基づいて、ソフトキー130に与えるべきラベルを指定する。

[type] (オプション) どんなタイプのメッセージを表示すべきかを指定する。可能な値は、「TEXT」、「FAX」、または「VOICE」である。

メソッド

[lock] 選択したメッセージを自動削除からロックする。

10

[unlock] 選択したメッセージをロック解除して、それを自動的に削除できるようにする。

[new] 新しい送信メッセージの作成を開始する。

[open] 選択したメッセージを表示し、それを読んだものとしてマークする。

【0323】

@を含む任意のURLもまた、このオブジェクトのメソッドであると考えられる。これは、以下のエスケープについて検索される。エスケープが見つかる場合、エスケープは現行選択からの関連データにより置換される。すべてのエスケープが一旦置換されると、結果得られるURLがフェッチされる。

【0324】

20

【表3】

メッセージについてのエスケープ

エスケープ	置換するもの
@i	選択されたメッセージのレコードID
@b	選択されたメッセージの本文 (URLエンコード)
@n	音声によりメッセージに返信したい場合の、電話すべき電話番号。数字ページ (電話番号だけのメッセージ) の場合、これはページにおける番号である。他のすべてのメッセージの場合、これはメッセージ送信者の電話番号である。
@s	メッセージ送信者の電話番号。
@@	1つの「@」キャラクタ

30

【0325】

(2) コンテンツ/コマンドURL

message:count?type=type

テキストとして、待機しているメッセージの数を含むストリームを戻す。可能なtype値は、VOICE、TEXT、またはFAXである。通常、これは、すべてのタイプのメッセージについてインボックスを実施する際に<INC>タグと共に使用され、装置がサポートするメッセージタイプのリストを提供し、このリストにより、ユーザがそのタイプのメッセージについて個々の画面を得ることが可能になる。リストは、このURLおよび<INC>タグを使用して、この画面上で使用可能なメッセージの数を、リストを構成するテキストの一部として表示する。URLはまた、そのタイプの任意のメッセージが使用可能であるか否かに基づいて、異なるタグを選択するために、IFタグのTEXT属性において使用することもできる。例えば、静止画像は、メッセージがないときに使用できるが、動画は、そのタイプの使用可能なメッセージがあるときに使用される。

40

【0326】

message:operate?reply=&delete=&lock=&unlock=&adjust=&id=number

idが指定されているメッセージに対して操作を行う。可能な操作は (URLでは、これらのうちの1つのみを指定することができる)、

・reply:宛先アドレスをidが指定されたメッセージの送信者にセットして、メッセー

50

ジ作成画面を提示させる。

- ・delete：メッセージを削除させる。
- ・delconfirm：ユーザに削除の確認を求めた後に、メッセージを削除させる。
- ・lock：メッセージをロックされているとマークさせ、メッセージ格納が満杯のときに、そのメッセージが自動的に削除されないようにする。
- ・unlock：メッセージをロック解除とマークさせ、メッセージ格納が満杯のときに、そのメッセージを自動的に削除できるようにする。

【0327】

adjust 引数が与えられる場合、それはURL履歴スタック108を以下のように調整させる。

- ・reply：調整は行われない。
- ・delete：参照しているURLがURLスタックからポップされる。
- ・lock、unlock：message：readURLからのストリームを、メッセージについて戻し、参照しているURLを置換する。

このURLは、操作するのに十分な特権を必要とする。参照しているURLが特権を欠く場合には、操作はユーザに確認される。

【0328】

message:read?id=number

idが渡されたメッセージを表示するストリームを戻す。ストリームは、テンプレートHTMLファイルのためのものである。そのテンプレートを埋めるパラメータは、メッセージのコンテンツに基づいて戻される。

【0329】

message:send?addr=string&body=string&newaddr=&newbody=&pop=&reply=id

テキストメッセージを指示したアドレスに送信するよう要求する。addrがない、またはnewaddrが存在する場合、これは、与えられた本文をいずれも維持しながら、ユーザがメッセージのアドレスをセット可能とするストリームおよびパラメータを戻す。

bodyがない、またはnewbodyが存在する場合、これは、与えられたいずれのaddrおよびreplyも維持しながら、ユーザがメッセージの本文を追加または編集できるようにするストリームおよびパラメータを戻す。

addrおよびbodyの双方が与えられ、かつnewaddrおよびnewbodyの双方が存在しない場合には、メッセージが送信される。replyが与えられる場合には、対応するメッセージが、返信済みとしてマークされる。

メッセージが送信され、かつpopが存在する場合、参照しているURLがURL履歴スタック108からポップされる。

このURLはまた、POST動作(PutURL)と共に動作し、その場合、addrがURLにおいて指定され、かつURLに配置すべきデータストリームがメッセージの本文としてとられる。

このURLは、操作するのに十分な特権を必要とする。参照しているURLが特権を欠く場合には、操作はユーザに確認される。

【0330】

c) 構成プロトコル

装置機能との対話のための第3のプロトコルは、構成プロトコルであり、これは装置の構成情報をセットすると共に、得るために使用される。このプロトコルは、構成プロトコルハンドラ112bによって処理される。

【0331】

このプロトコルで形成されるURLは、装置設定自身である。URLがフェッチされると、URLの現行設定が、テキストに変換され、ストリームとして戻される。データが構成プロトコルURLにポストされると、データが必要に応じて変換され、装置設定がセットされる。構成URLは、取得および設定の双方の場合に、装置設定内のビットをアドレス指定することができる。そして、URLは以下のように見える。

10

20

30

40

50

config:setting.bitnumber:bitsize

:bitsizeが存在しない場合、1と仮定される。

bitnumberは、0～31であり、31が最上位ビットである。

【0332】

大部分の設定は、任意のモジュールでフェッチすることができるが、（無線通信装置のPIN番号のように）いくつかは十分な特権を有するページ（通常、ROMメモリ126にあるHTMLファイル）のみによって取得することができる。十分な特権（ここでもまた、通常は、ROMメモリ126にあるHTMLファイルによる）なしでは、装置設定をセットすることはできない。

複数の設定は、「config:set」にポストすることで、セットすることができる。そして、ストリームは、「setting.bitnumber:bitsize/value」という形式の行を含む。

【0333】

【表4】

構成URL

URL	型	機能
config: keypadlock	boolean	セットされているとき、キーパッドをロックするか、またはロック解除する。
config:smspriority	int16	着信メッセージの重要度をランク付ける。0⇒すべてのメッセージを表示、1⇒smsキーワードを有するメッセージを表示、2⇒すべてのメッセージをインボックスに入れる。
config.smskeyword	char[16]	着信メッセージが表示するのに十分重要であるか否かを決定するために、着信メッセージにおいて探す言葉を指定する。
config:ringtone	int16	デフォルトにより、使用する呼び出し音シーケンス（1～127）を指定する。
config:ringmode	int16	電話をどのように鳴らすかを指定する。0⇒周期的に鳴らす、1⇒周期的に鳴らす+バイブレーション、2⇒一度だけ鳴らす、3⇒まったく鳴らさない。
config:setquietmode	boolean	セットされているとき、優先度の高い番号からの呼のみについて、電話を通常通りに鳴らすようにする。
config:quietmode	int16	静粛モードにあるとき、優先度の高くない番号について電話をどう鳴らすかを指定する。0⇒バイブレーション、1⇒音を出す、2⇒何もしない。
config:setpassword	boolean	セットされているとき、電話を電源オンにしたときは常に、ユーザにパスワードを訊ねる。
config:password	char[4]	入力されなければならないパスワード
config:backlight	boolean	真であれば、電話の使用時にバックライトが点灯する。
config:keyclicks	boolean	真であれば、キーボードが、押下されたキーに適したクリック音を生成する。
config:autosearch	boolean	真であれば、ユーザが電話をダイヤルする際に、マッチングについて電話帳を検索する。
config:followup	boolean	真であり、かつ電話帳になく、最近かけていない番号に電話をかけた、または電話を受けた場合、ユーザにその番号を電話帳に入れるか否かを訊ねる。
config:lasttime	int16	最後の呼が接続されていた秒数。
config:totaltime	int32	接続されて費やされた秒数の合計。

【0334】

d) 「extra」プロトコル

「extra」プロトコルは、主に<INC>、<IF>、および<TEMPLATE>タグと共に使用され、HTML p テンプレートを本発明の大半のユーザインタフェースとして機能できるようにする。このプロトコルは、エクストラプロトコルハンドラ112cによって処理される。

【0335】

一般に、エクストラパラメータは、以下のいずれかからHTML p テンプレートに渡される。

1) MMI 102のCコード

2) arguments スtringとして、ファイルURLに。これは、「file://filename?variablename=extra#data」の形式をとる。エクストラデータは、その変数名とともに格納され、かつ何のページであれ、ページがfilenameをフェッチするためのHTMLを完成させるために使用される。

3) 形式データを次のURLに渡すために、新しいMETHOD=NEXT属性を使用した先の形式からのデータとして。

【0336】

さらに、HTML p ページにあるURLがフェッチされ、引数を持たない場合、フェッチされているURLを含むページに渡されたパラメータはいずれも、フェッチされているURLにも渡される。

【0337】

エクストラプロトコルハンドラ112cは、URLにマッチする引数を探して、該引数をストリームに変換する。例えば、「<INC src=extra:body>」は、HTML p ストリームへの「body」引数を含む。別の例として、標準のグラフィカル要素を有する標準フォーマットでメッセージを表示するために使用されるHTML p ページがあると仮定する。表示するメッセージは、HTML ページをロードするURLについての引数として与えられる。

【0338】

例えば、URL「file://message.html?message=Please+enter+a+valid+phone+number」がエクストラプロトコルハンドラ112に与えられる場合、これは、テキスト「Please enter a valid phone number」を変数名「message」を有するテキストStringとして格納する。このエクストラデータは、Stringデータを出力する「extra:message」URLの使用により、他のどんなページにも表示される。図22aから図22cは、この使用を図示したものである。この例では、エクストラデータは<INC>タグによって検索され、テキストStringを直接的にページに組み込むことになる。

【0339】

一般に、エクストラプロトコルハンドラ112cは、HTML p コンテンツハンドラ114cがHTML p ページにおける「extra」URLを構文解析する結果、呼び出される。そのように識別すると、URLは、エクストラプロトコルハンドラ112cに渡されて、エクストラデータをデコードすると共に検索し、これをHTML p コンテンツハンドラ114cに戻して、ページにレンダリングする。

【0340】

e) 内蔵プロトコル

内蔵プロトコルは、IMGタグのSRC属性で使用するため、内蔵アイコンおよびイメージへのアクセスを提供する。これらのアイコンおよびイメージは、メモリ126のROMに格納される。「内蔵」テキストは、URLのプロトコルコンポーネントを形成し、所望のアイコンの名称は、URLのデータコンポーネントを構成する。図23は、アイコンの好ましいセットと、それらを指定するための完全なURLとを示す。

【0341】

一般に、内蔵プロトコルハンドラ112aは、HTML p コンテンツハンドラ114cがHTML p ページにおける「builtin」URLを構文解析する結果、呼び出される。そ

10

20

30

40

50

のように識別すると、URLは内蔵プロトコルハンドラ112aに渡されて、アイコンまたはイメージデータをデコードすると共に、メモリ126から検索し、これをHTMLpコンテンツハンドラ114cに戻して、ページにレンダリングする。

【0342】

C. ポータブルコンポーネント

再度図3を参照して、ポータブルコンポーネント116は、ユーザインタフェースエンティティとユーザインタフェースの実施に使用される他の機能的コンポーネントのセット、および無線通信装置100に必要とされる格納装置である。コンポーネントは、可搬性レイヤ118によって提供されるAPIに書き込むと共に、MMI102の基本的な実施要素としての役割を果たしながら、依然として異なる無線通信装置100との使用のためポータブルである。

【0343】

(1) グラフィックス

グラフィックシステム224は、画面表示136をウィンドウと呼ばれるセクションに分割する。ウィンドウは、子ウィンドウが完全に親ウィンドウ内に含まれる階層に構成される。しかし、本発明のウィンドウシステムは、他のウィンドウシステムとは異なり、2つのタイプのウィンドウ、「ダル(dull)」と「スプライト(sprite)」とを区別する。従来のシステムのように、あらゆるユーザインタフェースコンポーネントおよびウィンドウに各自のビットマップを持たせる(これは、より複雑なビットマップ処理を必要とする)のではなく、グラフィックシステム224は、ユーザインタフェースコンポーネントが通常重複しないという事実を利用する。グラフィックシステム224は代わりに、いくつかのウィンドウ(例えば、ダイアログボックス、または他のウィンドウと重複し、それらを隠す必要のある他のウィンドウ)をビットマップを有する(「スプライト」にする)ものとして定義する一方で、その他(「ダル」ウィンドウ)を、ビットマップを有する最も近くにある先祖のビットマップに描画する。この区別により、ユーザインタフェースコンポーネントの格納に必要なメモリの量が低減し、かつ画面表示136の更新プロセスが単純化される。

【0344】

ユーザインタフェース要素がウィンドウに引き出されるとき、描画は実際に直接画面表示136にではなくメモリ126におけるビットマップに起こる。ある時点で(実際に、コールバックキュー110が処理されるトップループにおいて)、これら任意かつすべてのビットマップへのすべての変更が画面表示136に転送される。この動作により、画面表示136の正確かつクリーンな更新が保証されると共に、基礎をなすビデオドライバを単純化する。該ビデオドライバは、メモリ126から画面表示136にビットマップを転送する必要があるだけである。

【0345】

グラフィックシステム224は、以下の基本的なグラフィックプリミティブを提供する。

- ・ライン(太いまたは細い、任意または特別な場合の単一ピクセル厚の垂直または水平、水平ライン、垂直ライン、および長方形もまたフィルパターンを有することができ、それらをドットまたは波線にする)
- ・長方形(外郭線または中実)
- ・ビットマップ(ピクセル当たり1つのビット、ステンシル(セットビットはカラーで描画されるが、クリアビットは何もしない)として、またはイメージ(セットビットは一色で描画されるが、クリアビットは別の一色で描画される)として)
- ・テキスト

【0346】

画面表示136の座標系は、座標が画面表示136上のピクセル間にあるようなものである。その結果、長方形が(a, b)から(c, d)に、もう一つを(c, d)から(e, d)に描画する場合、重複しない。

【0347】

テキストは、Text Stateと呼ばれるデータ構造を用いて、描画かつ処理される。テキストは、各種の構成可能な属性を有する。

- ・ポイントサイズ（小、中、大に予めセットされる）
- ・スタイル（標準、太字、斜体、下線、固定幅、および取り消し線）
- ・カラー

【0348】

Text Stateはまた描画位置を格納し、これは、描画されたストリングのすぐ後ろになるよう更新されるため、複数のストリングを、幅を計算する必要なく、次々と描画することができる。

【0349】

2. ユーザインタフェースガジェット

無線通信装置100のユーザインタフェースの大部分は、ソフトキー130およびソフトキーメニューの形で設けることができる。コンテンツエリア214は、通常、テキスト、アイコン、およびHTMLフォーム等を表示する。これらの特徴をサポートするために、多くのユーザインタフェースガジェット226が設けられる。

・Checkbox：チェックボックスおよびラジオボタンの双方を実施するために使用される（ラジオボタンは、ユーザが選択したときに、選択解除する必要のあるその他のラジオボタンを知るために、外部コールバックに頼る）。HTMLにおける<INPUT TYPE=checkbox>および<INPUT TYPE=radio>に応答して、インスタンス生成される。

・Icon：内蔵アイコンを表示する。

・LabelLine：標準の場所に標準フォントのオプションのテキストラベルを有する水平ライン。HTMLにおける<HR>タグによってインスタンス生成される。

・List：ユーザインタフェースにおける各種アイテムリストすべての基礎。特殊なリストのサブクラスは、個々のアイテムを描画し（Listによって配置される）、Listの選択をいつキーストロークまたは他の手段で調整すべきであるかを決定する。

・Popup：ユーザが1つまたは複数のアイテムを選択することができるストリングのポップアップリストを実施する。各アイテムは、それにバインディングされたストリング値を有する。SIZEパラメータが1であるとき、HTMLにおける<SELECT>タグによってインスタンス生成される。

・ScrollBanner：指定した速度で、右から左に、または左から右にスクロールすることができる単一行テキストバーを実施する。HTMLにおける<MARQUEE>タグによってインスタンス生成される。

・StringList：Popupのリスト部分を実施するが、また、スクローリングリストとして独立することもできる。ソフトキーメニューは、StringListによって実施される。SIZEパラメータが1ではないとき、HTMLにおける<SELECT>タグによってインスタンス生成される。

・TextEdit：単一行または複数行テキスト編集エリア。HTMLにおける<INPUT TYPE=text>および<INPUT TYPE=password>タグによってインスタンス生成される。

これらのエンティティは、各種グラフィック要素を表示するために、必要に応じて各種モジュールによって作成される。

【0350】

これら各種タイプのユーザインタフェース要素は、ページが構文解析されるとき、かつ対応するHTMLタグに応答して、HTML pコンテンツハンドラ114cによって作成される。例えば、ページにおけるINPUT TYPE=TEXTタグは、画面表示136の適切な場所にあるTextEditオブジェクトという結果になる。ユーザがアップまたはダウンキーで該オブジェクトを選択すると、かかるキーストロークがシェル106によりTextEditオブジェクトに渡されるため、ユーザによるキーストローク入力を受け取るよう、入力フォーカスが与えられる。

【0351】

ユーザインタフェースガジェットと関連するものは、テキストを入力かつ表示するための他の2つのモジュールである。TextEntryモジュール228は、キーストロークを許容して、それらをテキスト入力のためにコマンドにマッピングする。該コマンドは、仮の（後続するキーストロークに基づくさらなる変更の対象となる）キャラクタおよびワードの表示、仮のキャラクタおよびワードの置換を含み、最後の仮のキャラクタまたはワードを最終とし、カーソルまたは挿入シンボルキャラクタを移動する。

【0352】

テキスト入力を要求するエントリはいずれも、TextEntryモジュール228により登録されてから、キーストローク自身を挿入するのではなく、略すべてのキーストロークをそれに渡す。各種絵文字言語のフロントエンド処理も同様に、このモジュールで処理される。

【0353】

もう一方のモジュールは、TextWrapモジュール230であり、これは、任意の領域を処理すると共に、これら領域内のテキストおよびオブジェクトをラップする。このモジュールは、主にHTMLコンテンツの表示に使用されるが、複数行をラップすることができるリストエントリについても使用可能である。

【0354】

3. データ格納

実施のもう一つの要素は、データ格納232である。データ格納232は、以下の特性を有する単純な「flat-file」データベースである。

- ・レコード当たり255フィールドまで。フィールドは、名称および番号の双方を有するが、実際にデータにアクセスするには番号のみが用いられる。
- ・すべてのレコードは、論理的にすべてのフィールドを持つように定義される。格納最適化として、特定のレコードの特定のフィールドにデータが与えられていない場合、そのレコードの該フィールドは、スペースをとらない。
- ・各レコードは、レコードへのアクセスを得るために使用される一意の識別子である（今のところ16ビット）。
- ・データベースレコードは、直接操作されない。代わりに、レコードのフィールドを取得、セットする関数を使用される。
- ・データベースは、それ用に操作された8つまでのインデックスを持つことができる。各インデックスは、選択ルーチンと比較ルーチンとを有する。選択ルーチンは、いずれのレコードがインデックスの一部であるかを決定し、比較ルーチンは、インデックスにおけるレコードをソートするために使用される。インデックスが定義されると、これは、いずれのデータベースフィールドが選択および比較ルーチンによって使用されるかを指定する。レコードが変更されると、それらフィールドの1つが変更される場合のみ、レコードが再配置されるか、追加されるか、またはインデックスから削除される。
- ・レコードにアクセスするためには、2つの機能のうちの1つが、DataStoreRecordトークンを得るために呼び出される。エンティティがレコードの検査または操作を行うとき、DataStoreRecordDoneを呼び出す。

【0355】

4. ファイルシステム

無線通信装置100は、2つの基礎をなすファイルシステムと通信するファイルインタフェースを備える。

- ・コードにコンパイルされたデータ構造である読み出し専用ファイルシステム
- ・無線通信装置100のフラッシュメモリチップの中で分散されるフラッシュファイルシステム

ファイルアクセスは、関数の（最小）類似したセットを介する。

- ・FileOpen
- ・FileCreate
- ・FileRead
- ・FileWrite

10

20

30

40

50

・FileSeek ・FileTruncate ・FileClose ・FileDelete

【0356】

各ファイルシステムは、すべての基本的な動作を呼び出すルーチンを含む構造によって定義される。オープンファイルのリファレンスは、ファイルシステム特定データ、およびファイルがあるファイルシステムについてのルーチン表へのポインタを含む。ファイルが開かれると、上位レイヤが影響を受けるファイル名を検査し、適切なルーチン表を選択してから、そのテーブルにおけるOpenルーチンを使用して、ファイルを開く。その後、ファイルリファレンスを通して、ファイルにアクセスする。

【0357】

10

D. 可搬性レイヤ

再度図3を参照して、可搬性レイヤ118の各種モジュールが示される。可搬性レイヤ118は、任意の電話制御モジュール120およびリアルタイムオペレーティングシステム122でMMI102を比較的単純に実施するように設計される。可搬性レイヤは、以下のモジュールを提供する。

【0358】

1. 呼制御

呼制御モジュール140は、上位レイヤが呼を作成、操作し、DTMFトーンを生成すると共に、状態変更の通知を受け取れるようにする。インタフェースは、動作が呼に対して行われるという意味では非同期であるが、各動作の成功または失敗は、動作が要求された後そのうち報告される。

20

【0359】

【表 5】

呼制御関数

関数	目的
CallCreate	ダイヤルする番号を与えられると、電話をかけるよう試みる。呼は、音声、データ、またはファックスである。
CallCreateHidden	CallCreateと同様であるが、呼がユーザに見えないよう通知する。
CallRelease	電話を切るよう試みる。呼がコールグループ（カンファレンス）の一部である場合、グループ全体が切られる。
CallHold	呼（またはコールグループ）を保留するよう試みる。
CallResume	呼（またはコールグループ）を保留から外すよう試みる。別の呼またはコールグループが現在アクティブである場合、それがまず保留される。
CallCombine	保留呼を現行アクティブコールまたはコールグループに合わせて、新しいまたはより大きなコールグループを形成するよう試みる。
CallSeparate	コールグループから1つの呼を抽出して、それを再び別個の呼とするよう試みる。
CallGetInfo	どんなアクションを行うことができるか、その現在の状態、接続されている番号、接続した時間、保留した時間、およびユーザから隠すべきであるか否か等、アクティブコールについての各種情報を検索する。
CallPickup	着信に出るよう試みる。
CallStartSequence	互いに依存した呼操作ステップシーケンスの開始をマークする。ステップのうちの1つが失敗すると、シーケンスの残りのステップは試みられない。
CallEndSequence	呼操作ステップシリーズの終了をマークする。
CallStartDTMF	特定のキーに対応するDTMFトーンを生成し始める。
CallEndDTMF	予め開始されていたDTMFトーンの生成を止める。
CallEnumerate	システムにおける各アクティブコールに関数呼び出す。
CallDataAvailable	データコールの場合、読み出しに利用可能なバイト数を返す。
CallDataRead	データコールの場合、使用可能なデータを読み出す。
CallDataWrite	データコールの場合、データをネットワークに書き込む。
CallSetNotify	各呼動作の終了時、および着信の到来等、各種非同期イベントの発生時に呼び出すルーチンを指定する。

10

20

30

【0360】

2. メッセージ制御

メッセージ制御モジュール142は、上位レイヤがショートテキストメッセージを送受信できるようにする。このレイヤは、より大きなメッセージのセグメント化および再組立をサポートしても、しなくてもよい。

【0361】

【表 6】

メッセージ制御関数

関数	目的
SMSCreate	テキストメッセージの各種パラメータが与えられる場合（宛先番号、本文データ、本文フォーマット、およびオプション）、送信することができるメッセージのSMSトークンを戻す。
SMSCreateFromURL	URLエンコード引数のストリングを取り、それらを使用して、メッセージについてのSMSトークンを作成する。
SMSSend	SMSトークンを取り、関連するメッセージを送信する。メッセージを送信する試みにおいて、成功したか、失敗したときに、Notification 関数が呼び出される。
SMSDestroy	SMSトークンによって使用されるメモリを解放する。

10

【0362】

3. プラットフォーム

プラットフォームモジュール144は、プラットフォームに特定の関数を用意する。

【0363】

【表 7】

プラットフォーム関数

関数	目的
PlatMutexInitialize	PlatMutexGrab に渡されるとき、排他的アクセスを保証することができる変数を初期化する。MMI 102 は、シングルスレッドであるが、異なるスレッドで実行しうる可搬性レイヤ 118 がある。異なるスレッド（割り込み処理コードも別個のスレッドと見なされる）から呼び出されうる上位レイヤの部分は、この機構を使用して、データ構造への排他的なアクセスを保証する。
PlatMutexGrab	PlatMutexInitialize によって初期化される相互排他的変数により支配されるどんなものへの排他的アクセスも得る。
PlatMutexRelease	PlatMutexInitialize によって初期化される相互排他的変数により支配されるどんなものへの排他的アクセスも解放する。
PlatMutexDestroy	PlatMutexInitialize によって割り当てられたいかなる資源も解放する。
PlatWaitForSomething	PlatHaveSomething が呼び出されるまで、一時停止される。これは、上位レイヤが不必要にプロセッサの時間を無駄にしないように、可搬性レイヤ 118 がプロセッサの制御をシャットダウンするか、または譲るためのフックを提供する。
PlatHaveSomething	PlatWaitForSomething で待機している場合、上位レイヤを解放する。
PlatRingPlay	システムのスピーカを通して呼び出し音または他のサウンドを再生する。サウンドは、インデックス、音量レベル、およびバイブレーションを発生させるべきか否かによって定義される。127 のサウンドがシステムサウンド（例えば、キークリックおよびエラーサウンド）として定義される一方、127 のサウンドが電話呼び出し音用の階調として定義される。すぐに戻し、サウンドをバックグラウンドで再生できるようにする。
PlatRingStop	PlatRingPlay で開始されたアクティブサウンドを中断する。
PlatRingGetNumberOfRings	サポート／定義される電話呼び出し音の階調数を戻す。
PlatRingGetName	ユーザに表示して、デフォルトの呼び出し音か、または特定の人用の呼び出し音を選択できるようにするために、電話の呼び出し音の階調の 1 つの名称を戻す。
PlatGetPowerStatus	バッテリーおよび AC アダプタの現在の状態を検索する。

10

20

30

【0364】

4. タイマ

タイマモジュール 146 は、上位レイヤが指定した時間量の経過後に、関数呼び出しを受け取れるようにする基本的なタイミングサービスを提供する。

40

【0365】

【表 8】

タイマ関数

関数	目的
TimerAlloc	繰り返し使用できるタイマを割り当てる。タイマが切れたときに、ルーチン呼び出すようセットする。ルーチンは、コールバックキューを介して同期して呼び出され、他の動作を邪魔しない。
TimerStart	ミリ秒単位で、タイマの次のタイムアウト間隔を指定する。おおよそその時間量後、TimerAlloc において指定されたルーチンが呼び出される。
TimerStop	タイマが開始されないようにする。タイマがすでに開始されており、タイマルーチンへの呼び出しがコールバックキューにある場合には、該呼び出しをキューから削除して、タイマを止めた後に呼び出しを得るという処理の必要がないようにする。
TimerFree	割り当てられたタイマを解放する。タイマが作動中であつた場合には、また停止される。
TimerGetSeconds	秒の32ビットカウンタを戻す。カウンタは、秒毎に規則的に一度増分されるものである限り、特別な何かに関連する必要はない。

10

【0366】

5. 表示

20

表示モジュール148は、画面表示136を管理する。

【0367】

【表 9】

表示関数

関数	目的
DisplayDriverBltIn	ビットマップの部分をディスプレイ上の一点にコピーする。
DisplaySetBacklight	ディスプレイのバックライトを点灯または消灯する。
DisplayDriverSetContrast	ディスプレイのコントラストを指定された値にセットする。
DisplayDriverGetContrast	表示のために、現在のコントラストを検索する。

30

【0368】

6. フラッシュドライバ

フラッシュドライバモジュール150は、上位レイヤファイルシステムモジュール234が無線通信装置100のフラッシュメモリチップに対する読み出しおよび書き込みをできるようにする。

【0369】

【表 10】

フラッシュドライブ関数

関数	目的
FlashDriverInitialize	フラッシュメモリへのアクセスを初期化する。
FlashDriverGetInfo	フラッシュドライバおよびそれが駆動しているデバイスについての情報を戻す。これは、 ・デバイスにおける消去ユニットの数 ・各消去ユニットがどれくらい大きい ・デバイスが、任意のバイト境界ではなく、ページ単位で読み出し、かつ書き込む場合、ドライバはページサイズを指定することができる。 ・デバイスまたはドライバのいずれかが、書き込まれたページについてのエラー修正をサポートする場合、ドライバはこのことをフラッシュファイルシステムに示すことができる。
FlashDriverErase	デバイスの消去ユニットを消去する。これは、同期または非同期で発生しうるが、いずれの場合でも、消去が完了すると、コールバック関数が呼び出され、消去が成功したか否かを示す。
FlashDriverWrite	多数のブロックのバイトをデバイスの消去ユニットに書き込む。デバイスがページ単位で読み出し、かつ書き込む場合、ブロックのサイズは常にページサイズとなるが、ドライバは、ページを構成する1つまたは複数のブロックに、害のないバイトを提供する必要がある場合がある。
FlashDriverRead	デバイスの消去ユニットから多数のバイトを読み出す。

10

20

【0370】

7. 構成プロトコルハンドラ

このプロトコルハンドラ 112b は、CコードおよびHTMLが無線通信装置 100 の構成設定を取得し、セットできるようにする。これらの設定は、上位レイヤで実施されるものであり、モジュールは、必要に応じて下位レベルコードへの他の設定と通信する。構成設定にアクセスする必要があるため、構成プロトコルハンドラ 112b は、可搬性レイ

30

【0371】

構成URLは、取得およびセットの双方のために、装置設定内のビットをアドレス指定しうる。構成URLは、以下の構文を有する。

config.setting.bitnumber:bitsize

:bitsizeが存在しない場合には、1と仮定される。bitnumberは、0～15であり、15が最上位ビットである。

【0372】

大部分の設定は、それらを必要とする任意のモジュールまたはページによってフェッチされうるが、電話のPIN番号等のいくつかの構成設定は、十分な特権を有するURL（通常、装置のROMにあるHTMLファイル）によってのみ取得可能なものもある。十分な特権（ここでも、通常装置のROMにあるHTMLファイル）なしでは、いずれの装置設定もセットすることはできない。

40

【0373】

以下は、構成プロトコルハンドラ 112b の各関数の呼び出しに好ましい構成URLのセットである。

【0374】

【表 1 1】

構成プロトコルハンドラ関数

関数	目的
ConfigGetIntValue	統合構成設定を検索する。
ConfigSetIntValue	統合構成設定をセットする。
ConfigGetStringValue	ストリング構成設定を検索する。
ConfigSetStringValue	ストリング構成設定をセットする。
ConfigParseSetting	構成設定へのストリングリファレンスを構文解析し、設定のデータ型と、統合設定について、値のいずれのビットに影響する、またはフェッチするかをもたらす。
ConfigCheckPassword	パスワードについてURLからの引数を検査し、それを構成設定に格納されたパスワードと比較する。
ConfigConfirmPassword	ユーザからのパスワード取得と、パスワードが添付されたURLの再提出を処理する (ConfigCheckPasswordによって検査される)。

10

【0375】

要約すると、本発明は、システム、各種メソッド、および無線通信装置の柔軟性および機能性を実質的に強化するソフトウェア製品を提供する。本発明は、プロトコルハンドラおよびコンテンツハンドラの使用を含め、無線通信装置のすべての特徴および機能性に、マークアップ言語ベースのユーザインタフェースを介してアクセスかつ操作することができるシステムおよびソフトウェアアーキテクチャを提供する。HTMLpの拡張特徴およびHTMLpコンテンツハンドラは、特に、ウェブおよび他のコンテンツを無線通信装置の小さな画面表示に容易に表示できるようにすると共に、メニュー化、ナビゲーション特徴、およびHTMLのフォーム処理特徴を強化する。無線通信装置の電話または他の機能性へのアクセスを、マークアップ言語ベースのユーザインタフェースを通して提供することで、本発明は、サービス運営者が、熟練またはソフトウェア開発環境を必要とせず、または従来のMMIのソフトウェア管理問題なしで、新しいユーザインタフェースおよび異なる無線通信装置用のカスタム特徴セットを容易かつ迅速に生成できるようにする。さらに、本発明は、サービス運営者および製造業者が、個々でもまた各サービス運営者毎に異なるMMIソフトウェアを作成する必要なく、望むように無線通信装置を迅速かつ効率的にブランド化できるようにする。

20

30

付録 A

シェル関数

シェル106は、キー処理、URL履歴スタック108管理、他のデータ処理に使用するために、本発明のその他の部分に多くの関数を提供する。これらについて、以下に手短かに説明する。

40

【表 1 2】

キー処理関数	
関数	目的
ShellProcessKey	可搬性レイヤからのキーストロークを許容し、これをキーボードターゲットリストの最初のエンティティに渡す。
ShellGrabInput	エンティティをキーストローク受け取りに関心のあるものとして登録する。キーボードターゲットリストは、LIFO順に処理される。エンティティは、2つのルーチン プラスボイド*の表として定義され、これらのルーチンは任意の目的のために使用できる。2つのルーチンは、キーストロークを実際に受け取る ProcessKey と、エンティティがキーボードターゲットリストの最上部にあるとき、またはないとき、通知される GrabChange である。
ShellReleaseInput	キーボードターゲットリストからエンティティを登録解除する。
ShellPreviousInput	キーストロークをこの関数を呼び出す前に登録されたエンティティに渡す。コンテンツハンドラの ContentProcessKey 関数から呼び出されると、シェルがシステム全体のデフォルトに従ってキーストロークを処理する。
ShellMarkTop	現行のURLを複雑な対話の始まりとしてマークし、そのすべてのコンポーネントは、適切な引数を ShellGoBack に渡すことで、一度にスタックからポップすることができる。
ShellMarkTemp	最上部のURLの「temporary」属性をオンオフする。URLが一時的とマークされると、別のURLをフェッチするというあらゆる試みにより、新しいURLが表示される前に、一時的なURLをスタックからポップさせる。
ShellPutURL	渡されたURLについてのデータを格納する。URLスタックには影響を与えない。
ShellSetState	URLスタックにおける最上部のURLに関連する状態ブロックを記録する。URLについてのコンテンツハンドラは、ブロックのコンテンツを解放して変更する責任を負う。シェルは、単にアドレスを追跡するだけである。意図は、ユーザがこのURLにポップバックするときに、同じ視覚的場所に戻れるようにすることである。シェルは、プロトコルハンドラによって戻されたURLデータをキャッシュするよう試みない。
ShellGetState	ShellSetState を呼び出すことによりセットされた、最上部のURLについての状態ブロックを検索する。
ShellGetExtraData	最上部のURLと共に渡されたエクストラデータへのポインタを戻す。

10

20

30

【表 1 3】

URLスタック管理	
関数	目的
ShellGetURLWithDataAndFlags	URLに関連するデータをフェッチして、コンテンツエリアの制御を、プロトコルハンドラによって戻されたコンテンツのタイプのハンドラに渡し、URLをURLスタックの最上部に配置する。プロトコルハンドラによって戻された ContentStream に加えて、コンテンツハンドラは、このルーチンに渡される ShellExtraData を受け取る。 呼び出し側 (caller) はまた、URLの優先度 (そのデータの表示がどれ程重要か: URLスタックが、優先度がより高いURLを含む場合、渡されたURLに関連するデータのフェッチおよび表示が、ユーザが優先度がより高いURLをスタックから削除されるまで、表示される) を含むフラグのセットと、渡されたURLをフェッチする特権レベルとを渡す。
ShellGetURL	エクストラデータを渡さず、URLスタックの最上部にあるURLの優先度および特権を用いて、URLに関連するデータをフェッチして、それを表示する。
ShellGetURLFlags	URLスタックの最上部にあるURLについてフラグ (優先度および特権) を取得する。
ShellDisplayContent	これは、本発明におけるダイアログボックスに道義的に等しい。これは、プロトコルハンドラおよびURLに関連させる必要なく、コンテンツハンドラを直接提示できるようにする。コンテンツハンドラは、それを提示するURLが享受するのと同じフラグおよび特権を受け取る。
ShellGetURLStream	コンテンツエリアに表示されているものを変えずに、URLに関連するデータをフェッチする。
ShellRerun	URLスタックの最上部のURLについてのデータを再度フェッチし、それを再度表示させる。
ShellGoBack	URLスタックから1つまたは複数のURLをポップする。呼び出し側は、最上部のURLか、現在の対話に関連するすべてのURLか (Error! Reference source not found. (エラー! リファレンスソースが見つかりません。)) の項参照)、またはまさに最初のすべてのURLを単に削除することを示すパラメータを渡す。
ShellGoBackString	ShellGoBack と同様だが、動的に割り当てられたストリングを許容する。
ShellPopURL	URLスタックから特定のURLをポップする。

10

20

30

【表 1 4】

URL 生成および分解	
関数	目的
ShellParseURLArgs	URL においてエンコードされた引数を構文解析し始める。引数は、「name=value」の形式であり、複数の引数を「&」キャラクタで分離している。引数は通常、URL における最初の「?」に続く。
ShellGetNextURLArg	ShellParseURLArgs によって戻されたトークンを使用して、引数リストから次の引数をフェッチする。
ShellDoneWithURLArgs	URL からの引数の構文解析を終了する。
ShellFindURLArgs	特定の引数について URL 引数ストリングを検索して、それらの値を戻すショートハンド関数。引数によっては必須としてマークされるものがあり、すべての必須引数が存在しない場合には、値を戻さないようにする。
ShellBeginURLArgs	URL 引数リストの構築を開始する。name/value 対の初期セットは、この呼び出しにおいて与えることができる。
ShellAppendURLArg	別の name/value 対を構築されている引数リストに追加する。
ShellFinishURLArgs	すべての name/value 対が適切にフォーマットされ、エンコードされた新しく完成された引数ストリングを戻す。
ShellPrependURLToArgs	これらが引数である URL を指定する。ShellFinishURLArgs は、引数を添付した URL を戻す。
ShellEncodeURLArg	URL 引数リストに包含するためのストリングをエンコードする。これは、ShellAppendURLArg および ShellBeginURLArgs によって用いられるが、また汎用的な使用にも利用可能とされる。

10

20

【表 1 5】

コンテンツストリーム管理	
関数	目的
ShellCreateContentStream	渡されたタイプのストリームに ContentStream 構造を割り当てる。
ShellCloseContentStream	コンテンツストリームを閉じて、ContentStream 構造を解放する。

30

【表 1 6】

ソフトキー管理	
関数	目的
ShellDefineKeyGo	ソフトキーに特定のラベルを与え、キーの押下時には、現行のコンテンツハンドラの ContentActivate 関数に渡されたストリングを与えるよう要求する。
ShellDefineKeyBack	ソフトキーに特定のラベルを与え、キーの押下時には、ShellGoBack を渡された引数と共に呼び出すよう要求する。
ShellDefineKeyMenu	ソフトキーに特定のラベルを与え、キーの押下時には、渡されたエントリを有するメニューを提示するよう要求する。エントリの 1 つが選択されると、現行のコンテンツハンドラの ContentActivate 関数が、選択されたエントリに指定されたストリングと共に呼び出される。
ShellDefineKeyNothing	ソフトキーについて、任意の先行するバインディングを削除するよう要求する。

40

【表 1 7】

その他の画面要素	
関数	目的
ShellDisplayStatus	ステータスメッセージエリアが見えている場合、シェルは渡されたメッセージをそのエリアに表示する。
ShellDisplayTitle	タイトルエリアが見えている場合、シェルは渡されたタイトルをそのエリアに表示する。
ShellSetScrollable	ユーザに、コンテンツがいずれかの方向（上または下）にスクロールできることを知らせる。
ShellSetNewMail	新しいメールアイコンを画面に示すか、または画面から削除する。
ShellEnlargeContent	コンテンツエリアを拡大して、画面の指示された部分を占有するよう要求する。占有は、次のURLが表示されるまで継続され、かつ要求を行ったURLが再度表示されるときに、自動的に再確立されない。

10

【表 1 8】

テキストエントリモード	
関数	目的
ShellSetTextEntryMode	グローバルテキストエントリモードを変更し、ディスプレイ上のモードインジケータを更新する。シェル自体は、モードに基づいて別様に動作しないが、本発明の他の部分が考慮するのは、現行モード設定の管理者である。
ShellSetTextShiftState	本発明の別の部分により使用されるシフト状態をセットする。シフト状態はまた、シェルが表示するモードインジケータを変更することができる。
ShellGetTextEntryMode	現行のテキストエントリモードをフェッチする。
ShellGetTextShiftState	現行のシフト状態をフェッチする。
ShellSetModeSequence	「mode」キーについてのデフォルト処理の実施時に、シェルがサイクル動作すべきモードのシーケンスを指定する。1 から 3 つすべてのモードのいずれも指定することができる。
ShellEnforceModeSequence	グローバルモードが、渡されたパラメータに応じて、現行シーケンスにおける任意のモード、あるいは現行シーケンスにおける最初のモードのいずれかにセットされるよう保証する。

20

30

【表 1 9】

アイドル時処理	
関数	目的
ShellAddIdleHook	指定した秒数が、いずれのユーザ入力もなく経過した後、関数を呼び出すよう要求する。
ShellRemoveIdleHook	ShellAddIdleHook により先に登録された関数を削除する。
ShellDisableIdle	すべてのアイドル時処理をディセーブルする。これは、複数回呼び出すことができるが、各呼び出しは、対応する ShellEnableIdle への呼び出しとマッチしなければならない。
ShellEnableIdle	ShellEnableIdle への未処理の呼び出しがもうない場合、アイドル時処理を再度イネーブルさせる。

40

【表 2 0】

その他の関数	
関数	目的
ShellPowerOff	URLスタックからすべてのURLをポップし、可搬性レイヤに装置の電源を切るよう求める。
ShellStandardDialog	標準フォーマットでフル画面ダイアログを提示する。呼び出し側は、メッセージ、所望のダイアログのスタイル、ソフトキーで何を行うか、デフォルトアクションをとるためにユーザ入力なしでいる時間を指定することができる。

10

【符号の説明】

【0 3 7 6】

- 1 0 0 : 無線通信装置
- 1 0 2 : M M I
- 1 0 4 : ユーザインタフェース定義ファイル
- 1 0 6 : シェル
- 1 0 7 : ブラウザ
- 1 0 8 : U R L 履歴スタック
- 1 1 0 : コールバックキュー
- 1 1 2 : プロトコルハンドラ
- 1 1 4 : コンテンツハンドラ
- 1 1 6 : ポータブルコンポーネント
- 1 1 8 : 可搬性レイヤ
- 1 2 0 : 電話制御モジュール
- 1 2 2 : リアルタイムオペレーティングシステム
- 1 2 4 : プロセッサ
- 1 2 6 : メモリ
- 1 3 0 : ソフトキー
- 1 3 2 : 関数
- 1 3 4 : 数字キー
- 1 3 6 : 画面表示
- 1 4 0 : 呼制御
- 1 4 2 : メッセージ制御
- 1 4 4 : プラットフォーム
- 1 4 6 : タイマ
- 1 4 8 : 表示
- 1 5 0 : フラッシュドライブ
- 2 0 0 : ステータスバー
- 2 1 0 : タイトルバー
- 2 1 2 : ステータスライン
- 2 1 5 : スクロールアロー
- 2 1 6 : ソフトキーラベル
- 2 2 0 : フォーカストライアングルおよび選択
- 2 2 4 : グラフィックス
- 2 2 6 : U I (ユーザインタフェース) ガジェット
- 2 2 8 : テキストエントリ
- 2 3 0 : テクストラップ
- 2 3 2 : データ格納
- 2 3 4 : ファイルシステム
- 2 3 6 : ローカリゼーション

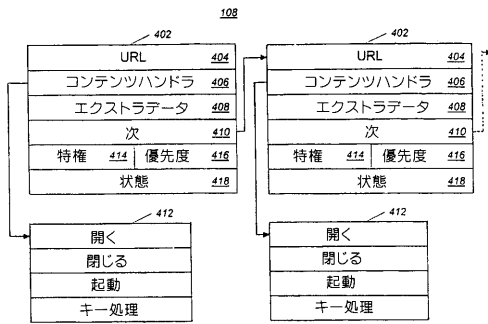
20

30

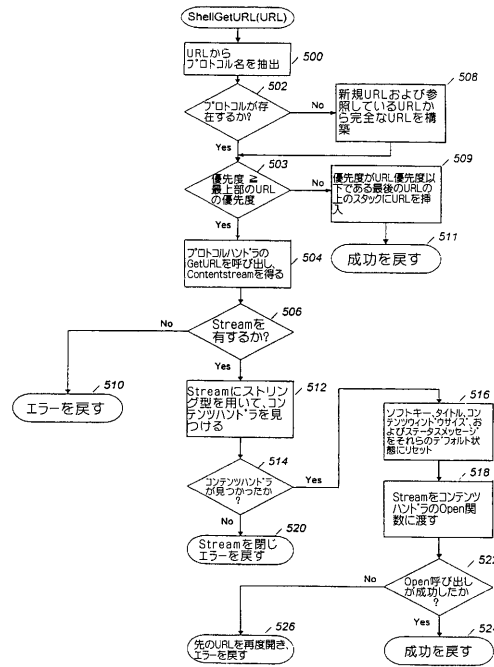
40

50

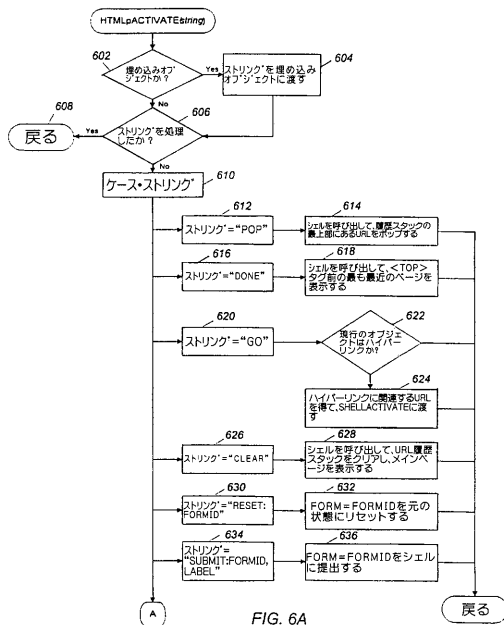
【 図 4 】



【図 5】



【図 6 a】



【 図 6 b 】

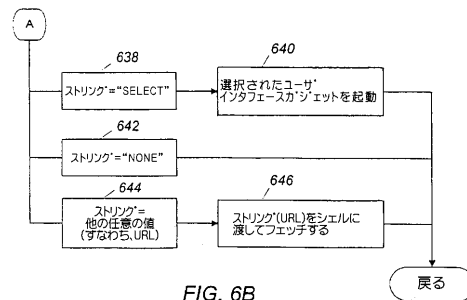


FIG. 6B

【図 7】

HTMLp	Result
<pre> <?PHP?> <HTML> <HEAD> <KEY KEY=2 LABEL="PhoneBook" ACTION=file://phonenumber.html"> <KEY KEY=1 LABEL="Menu" ACTION=MENU> <KEYEND KEY=1 LABEL="messages" ACTION=file://msgs.html"> <KEYEND KEY=1 LABEL="Recent Calls" ACTION=file://recent.html"> <KEY KEY=1 LABEL="Phone Settings" ACTION=file://phonettst.html"> <KEYEND KEY=1 LABEL="PhoneBook" ACTION=file://phonenumber.html"> </HEAD> <BODY> <CENTER> < e l u l a o > < e s l v i c e s </BODY> </HTML> </pre>	

FIG. 7

【图 8】

HTMPl	Result
<pre><HTML> <HEAD> <TITLE>Phone Book</TITLE> <KEY KEY=1 Label=New Actions="new"> <KEY KEY=2 Label=Clear action=delete> <HELP>Press any number key to search for an entry</HELP> <HELP>Press Send to call someone</HELP> </HEAD> <BODY> <OBJECT CODE="phone:list?editkey=2&showstatus"> </BODY> </HTML></pre>	
<pre><HEAD> <TITLE>Phone Book</TITLE> <KEY KEY=1 Label=New Actions="new"> <KEY KEY=2 Label=Clear action=delete> <HELP>Press any number key to search for an entry</HELP> <HELP>Press Send to call someone</HELP> </HEAD> <BODY> <OBJECT CODE="phone:list?editkey=2&showstatus"> </BODY> </HTML></pre>	

FIG. 8

【 図 1 0 】

[illegible]

FIG. 10

【 図 1 1 】

[illegible]

FIG. 11

【图 9】

HTMLp	Result
<pre> <HTML> <HEAD> <KEY key=2 LABEL=Ignore ACTION=phone:ignore> <IF TEST=extra:conf> <KEY key=1 LABEL=Conference ACTION=phone:conference> <ELSE> <KEY key=1 LABEL="Pick Up" ACTION=phone:answer> </IF> <KEY key=send ACTION=phone:answer> <IF TEST=conf:anysep> <KEY key=default ACTION=phone:answer> <KEY key=back ACTION=phone:answer> </IF> <KEY key=send ACTION=phone:ignore> <KEY key=up ACTION=phone:upvolume> <KEY key=down ACTION=phone:downvolume> </HEAD> <BODY> <TEMPLATE> <CENTER> <text:name>

 <img src=file:zingali(extra:pal)
 </CENTER> </TEMPLATE> </BODY> </HTML> </pre>	

FIG. 9

【图 1 2】

HTMLp	Result
<pre> <HTML> <HEAD> <TITLE>Set Backlight</TITLE> </HEAD> <BODY> <TEMPLATE> FORM MENU=POST ACTION=/config;set? Backlight should be:cb? <input >="" <=""
="" <input="" <="" <input="" body>="" charging?="" checked="config:backlight=2" config:backlight="" delay"="" delay:="" form>="" html>="" in="" name="ok" or="" pre="" template>="" type="submit" use="" value="OK>" when=""/> </pre>	

FIG. 12

【图 1 3】

HTMLp	Result
<pre>file info.html <HTML> <center>Information Services/
</center> <FORM ACTION=body.html> <SELECT SIZE=5 NAME=URL SAMPLE> <OPTION VALUE=http://www.africa.html>Africa World Match</OPTION> <OPTION VALUE=http://www/billing.html>Billing Inquiries</OPTION> <OPTION VALUE=http://www/portfolio.html>Stock Portfolio</OPTION> <OPTION VALUE=http://www/weather.html>Local Weather</OPTION> </SELECT> <INPUT TYPE=submit KEY=send> </FORM> </HTML> file body.html <!-- Specifies a body surrounded by a border with the operator logo at the top. The including file can add a subcaption immediately below, but left and right margin are set so text will not impair on the border --> <BODY BACKGROUND=gif-back.gif SGMPROPRIETIES=Fixed> <BLOGQUOTE>
 </BLOGQUOTE> file stdlogo.html <!-- Closes the tags that were opened by body.html --> </BLOGQUOTE> </BODY> file stdlogo.html <!-- Provides the operator logo with the text flow set to receive a subcaption for the logo --> </pre>	

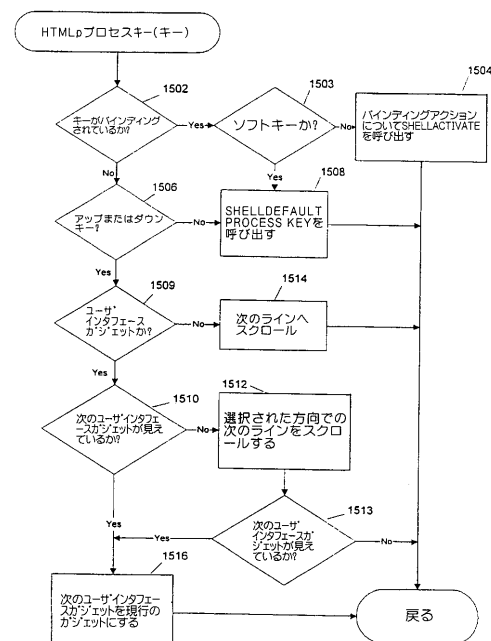
FIG. 13

【图 14】

HTMLp	Result
<pre><HTML> <HEAD><TITLE>Dialing</TITLE> <TEMP> <KEY KEY="Label="Recent" Actions="file://recent.html"> </HEAD> <BODY> <FORM ACTION="phone:dial"> <INPUT NAME="num" TYPE="text"> <INPUT TYPE="submit" KEY="send"> </FORM> </BODY> </HTML></pre>	

FIG. 14

【图 15】



【図 16】

HTML	Result
<pre> <HTML> <BODY> <FORM METHOD=POST ACTION="http://cgi-bin/purchase"> Name: <INPUT TYPE=TEXT NAME=name>
 Address: <INPUT TYPE=TEXT NAME=street>
 City: <INPUT TYPE=TEXT NAME=city> State: <INPUT TYPE=TEXT NAME=state MAXLENGTH=2 SIZE=2> Zip: <INPUT TYPE=TEXT NAME=zip SIZE=5>
 Credit Card: <INPUT TYPE=RADIO NAME=cardtype VALUE=visa>VISA <INPUT TYPE=RADIO NAME=mc>MasterCard <INPUT TYPE=RADIO NAME=amex>American Express
 Number: <INPUT TYPE=TEXT NAME=number>
 <INPUT TYPE=submit name=buy value=Buy></FORM> </BODY> </HTML> </pre>	

FIG. 16
PRIOR ART

【図 17 a】

CLIENT	SERVER
<pre> <HTML> <HEAD> <TITLE>Purchase Skis</TITLE> <TOP> </HEAD> <BODY> Please enter your name: <FORM METHOD=GET ACTION="http://skishop/cgi-bin/addr"> <INPUT TYPE=TEXT NAME=name SIZE=25>
 <INPUT KEY=send TYPE=submit> </FORM> </BODY> </HTML> GET http://skishop/cgi-bin/addr?name=George+Will </pre>	<pre> <HTML> <HEAD> <TITLE>Confirm Order</TITLE> </HEAD> <BODY> <DL> <DT>Name:</DT><DD>George Will</DD> <DT>Address:</DT><DD>2 Anywhere, Madison, WI, 53705</DD> <DT>Credit Card:</DT><DD>VISA</DD> <DT>Card Number:</DT><DD>7732215661000</DD> </DL> <FORM METHOD=POST ACTION="http://skishop/cgi-bin/purchase"> <INPUT TYPE=hidden NAME=name VALUE="George Will"> <INPUT TYPE=hidden NAME=street VALUE="2 Anywhere"> <INPUT TYPE=hidden NAME=city VALUE="Madison"> <INPUT TYPE=hidden NAME=state VALUE="WI"> <INPUT TYPE=hidden NAME=zip VALUE="53705"> <INPUT TYPE=hidden NAME=cardtype VALUE="visa"> <INPUT TYPE=hidden NAME=number VALUE="7732215661000"> <INPUT TYPE=submit name=buy value="Buy"> </FORM> </BODY> </HTML> POST http://skishop/cgi-bin/purchase?name=George+Will&street=2+Anywhere&city=Madison&state=WI&zip=53705&cardtype=visa&number=7732215661000 </pre>
	サーバに維持されるHTMLテンプレートにname)を挿入、クライアントにアップロード
	サーバに維持されるHTMLテンプレートにname)を挿入、クライアントにアップロード

FIG. 17a
PRIOR ART

【図 18 a】

HTMLp	Result
<pre> file: purchaseform.html <HTML> <HEAD> <TITLE>Purchase Skis</TITLE> <TOP> <KEY KEY=2 LABEL="Never Mind" ACTION=done> </HEAD> <BODY> Please enter your name: <FORM METHOD=POST ACTION="file://addr.html"> <INPUT TYPE=TEXT NAME=name cap=name abcmodes=alpha,num SIZE=25>
 <INPUT TYPE=submit> </FORM> </BODY> </HTML> file: addr.html <HTML> <HEAD> <TITLE>Purchase Skis</TITLE> <KEY KEY=2 LABEL="Never Mind" ACTION=done> </HEAD> <BODY> Now we need your address: <FORM METHOD=POST ACTION="file://credit.html"> <INPUT TYPE=TEXT NAME=street cap=name abcmodes=num,alpha SIZE=25>
 <INPUT TYPE=TEXT NAME=city cap=name abcmodes=alpha SIZE=10>
 <INPUT TYPE=TEXT NAME=state cap=name abcmodes=alpha SIZE=2>
 <INPUT TYPE=TEXT NAME=zip cap=name abcmodes=num SIZE=5> <INPUT TYPE=submit> </FORM> </BODY> </HTML> file: credit.html <HTML> <HEAD> <TITLE>Purchase Skis</TITLE> <KEY KEY=2 LABEL="Never Mind" ACTION=done> </HEAD> <BODY> Finally, your credit card info: <FORM METHOD=POST ACTION="file://confirm.html"> <INPUT TYPE=RADIO NAME=cardtype VALUE=visa>VISA <INPUT TYPE=RADIO NAME=cardtype VALUE=mc>MasterCard <INPUT TYPE=RADIO NAME=cardtype VALUE=amex>American Express
 Number: <INPUT TYPE=TEXT NAME=number abcmodes=num> <INPUT TYPE=submit name=buy value="Review"> </FORM> </BODY> </HTML> </pre>	

FIG. 18a

【図 17 b】

CLIENT	SERVER
<pre> <HTML> <HEAD> <TITLE>Purchase Skis</TITLE> </HEAD> <BODY> Finally, your credit card info: <FORM METHOD=GET ACTION="http://skishop/cgi-bin/confirm"> <INPUT TYPE=hidden NAME=name VALUE="George Will"> <INPUT TYPE=hidden NAME=street VALUE="2 Anywhere"> <INPUT TYPE=hidden NAME=city VALUE="Madison"> <INPUT TYPE=hidden NAME=state VALUE="WI"> <INPUT TYPE=hidden NAME=zip VALUE="53705"> <INPUT TYPE=RADIO NAME=cardtype VALUE=visa>VISA <INPUT TYPE=RADIO NAME=cardtype VALUE=mc>MasterCard <INPUT TYPE=RADIO NAME=cardtype VALUE=amex>American Express
 Number: <INPUT TYPE=TEXT NAME=number> <INPUT TYPE=submit name=buy value="Review"> </FORM> </BODY> </HTML> GET http://skishop/cgi-bin/confirm?name=George+Will&street=2+Anywhere&city=Madison&state=WI&zip=53705&cardtype=visa&number=7732215661000 </pre>	<pre> <HTML> <HEAD> <TITLE>Confirm Order</TITLE> </HEAD> <BODY> <DL> <DT>Name:</DT><DD>George Will</DD> <DT>Address:</DT><DD>2 Anywhere, Madison, WI, 53705</DD> <DT>Credit Card:</DT><DD>VISA</DD> <DT>Card Number:</DT><DD>7732215661000</DD> </DL> <FORM METHOD=POST ACTION="http://skishop/cgi-bin/purchase"> <INPUT TYPE=hidden NAME=name VALUE="George Will"> <INPUT TYPE=hidden NAME=street VALUE="2 Anywhere"> <INPUT TYPE=hidden NAME=city VALUE="Madison"> <INPUT TYPE=hidden NAME=state VALUE="WI"> <INPUT TYPE=hidden NAME=zip VALUE="53705"> <INPUT TYPE=hidden NAME=cardtype VALUE="visa"> <INPUT TYPE=hidden NAME=number VALUE="7732215661000"> <INPUT TYPE=submit name=buy value="Buy"> </FORM> </BODY> </HTML> POST http://skishop/cgi-bin/purchase?name=George+Will&street=2+Anywhere&city=Madison&state=WI&zip=53705&cardtype=visa&number=7732215661000 </pre>
	サーバに維持されるHTMLテンプレートにクレジットカード/パラメータを挿入、クライアントにアップロード
	すべてのパラメータによりブラウザクションを処理

FIG. 17b
PRIOR ART

【図 18 b】

HTMLp	Result
<pre> <HTML> <HEAD> <TITLE>Confirm Order</TITLE> <KEY KEY=2 LABEL="Never Mind" ACTION=done> </HEAD> <BODY> <template> <DL> <DT>Name:</DT><DD>(extra:name)</DD> <DT>Address:</DT><DD>(extra:street), (extra:city), (extra:state), (extra:zip)</DD> <DT>Credit Card:</DT><DD>(extra:cardtype)</DD> <DT>Card Number:</DT><DD>(extra:cardtype)</DD> </DL> </template> <FORM METHOD=POST ACTION="http://cgi-bin/purchase"> <INPUT TYPE=submit name=buy value="Buy"> </FORM> </BODY> </HTML> Final HTTP transaction sent to server: POST //cgi-bin/purchase name/George Will street/2 Anywhere city/Madison state/WI zip/53705 cardtype/visa number/7732215661000 buy/Buy! </pre>	

FIG. 18b

【図 19】

HTMLp	Result
<pre> <HEAD> <TITLE>Green-Tongued Iguana</TITLE> </HEAD> <BODY> The green-tongued iguana is a native of Adelaide, Australia. It likes to live under space trees, unlike the Blue-Tongued Iguana, its cousin to the north. </BODY> </pre>	

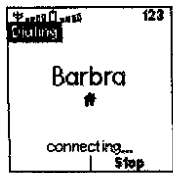
FIG. 19

【図 20】

HTML.p	Result
<pre> <HEAD> <TITLE>OK to SEA 3/24</TITLE> <LINKMENU> </HEAD> <BODY> <TABLE> <TR> <TD><U>Flight</U></TD> <TD ALIGN=CENTER><U>Dep</U></TD> <TD ALIGN=CENTER><U>Arr</U></TD> </TR> <TR> <TD>UA 909</TD> <TD ALIGN=RIGHT>09:33</TD> <TD ALIGN=RIGHT>11:22</TD> </TR> <TR> <TD>AS 63</TD> <TD ALIGN=RIGHT>10:02</TD> <TD ALIGN=RIGHT>11:57</TD> </TR> <TR> <TD>AS 102</TD> <TD ALIGN=RIGHT>12:38</TD> <TD ALIGN=RIGHT>14:08</TD> </TR> <TR> <TD>UA 887</TD> <TD ALIGN=RIGHT>15:10</TD> <TD ALIGN=RIGHT>17:21</TD> </TR> </TABLE> </BODY> </pre>	

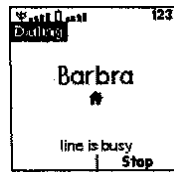
FIG. 20

【図 21 a】



(a)

【図 21 b】



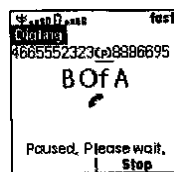
(b)

【図 21 c】



(c)

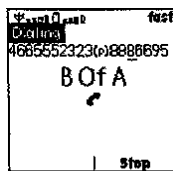
【図 21 d】



(d)

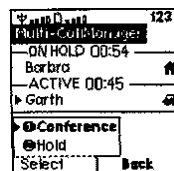
FIG. 21

【図 21 e】



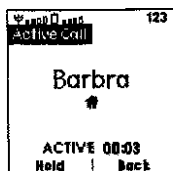
(e)

【図 22 c】



(c)

【図 22 a】



(a)

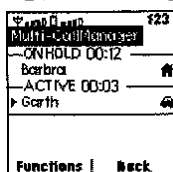
【図 23】

HTML.p	Result
<pre> <HTML> <HEAD> <TITLE>Error!</TITLE> <KEY REVE=1 LABEL="Sorry!" ACTION=pop> </HEAD> <BODY> <CENTER>
 <INC SRC=extra:message> </CENTER> </BODY> </HTML> </pre>	

FIG. 23

FIG. 22

【図 22 b】



(b)

【図 24】

キーキャプス		その他	
builtin:key1	①	builtin:smiley	😊
builtin:key2	②	builtin:signal	📶
builtin:key3	③		
builtin:key4	④		
builtin:key5	⑤		
builtin:key6	⑥		
builtin:key7	⑦		
builtin:key8	⑧		
builtin:key9	⑨		
builtin:key0	⑩		
電話番号タイプ			
builtin:type0	☎	builtin:bigtype0	☎
builtin:type1	📠	builtin:bigtype1	📠
builtin:type2	📞	builtin:bigtype2	📞
builtin:type3	📠	builtin:bigtype3	📠
builtin:type4	📞	builtin:bigtype4	📞
builtin:type5	📠	builtin:bigtype5	📠
builtin:type6	📞	builtin:bigtype6	📞
builtin:type7	📠	builtin:bigtype7	📠

【図 25】

HTMLp	Result
<pre> <HEAD> <DIAL NAME="Voice Mail" NOSCREENCHANGE>8144299x4416722</DIAL> <TITLE>Voice Box</TITLE> <KEY KEY=#5 ACTION=vmListen.http> <KEY KEY=#4 ACTION=vmGreeting.http> <KEY KEY=#6 ACTION=vmNew.http> <KEY KEY=#9 ACTION=vmSignoff.http> <KEY KEY=default ACTION=none> </KEY> </HEAD> <BODY> Greeting Options
 Listen to Messages
 Record a Message
 Sign Off
 </BODY> </pre>	
<pre> <HEAD> <DIAL NAME="Voice Mail" NOSCREENCHANGE>8144299x4416722</DIAL> <TITLE>listen</TITLE> <KEY KEY=#2 ACTION=phone:dial?num=2> <KEY KEY=#3 ACTION=phone:dial?num=3> <KEY KEY=#5 ACTION=phone:dial?num=5> <KEY KEY=#7 ACTION=phone:dial?num=7> <KEY KEY=#9 ACTION=pop> </KEY> </HEAD> <BODY> Back Up a bit
 Erase
 Next Message
 Save Message
 Stop Listening
 </BODY> </pre>	

FIG. 25

【手続補正書】

【提出日】平成24年4月23日(2012.4.23)

【手続補正1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項1】

無線通信装置において使用されるコンピュータプログラムであって、該無線通信装置は、メモリと、画面表示と、コンピュータプログラムを実行し、前記無線通信装置の動作を制御するプロセッサと、を備え、前記コンピュータプログラムは、

プロトコルコンポーネントおよびデータコンポーネントを有するURLを受信するためのシェルであって、前記データコンポーネントは、実行されるコマンドまたは取得されるコンテンツを特定するものであり、前記シェルは、前記プロトコルコンポーネントに従って前記データコンポーネントをプロトコルハンドラに提供し取得したコンテンツを処理のためにコンテンツハンドラに提供する、シェルと、

URLを受信し、前記データコンポーネントによって特定されるコンテンツを取得して該取得したコンテンツを前記シェルに提供する、または前記データコンポーネントによって特定されるコマンドを実行する、各々が前記シェルと通信可能に接続される複数のプロトコルハンドラと、

前記取得されたコンテンツを受信し、前記無線通信装置の画面表示に出力するために該取得したコンテンツを処理する、各々が前記シェルと通信可能に接続される複数のコンテンツハンドラと、

を備える、コンピュータプログラム。

【請求項 2】

前記複数のプロトコルハンドラは、

前記シェルからURLを受信し、前記無線通信装置における電話通信機能を起動するために該URLをデコードする電話プロトコルハンドラと、

前記シェルからURLを受信し、前記無線通信装置のメモリに記憶されるデータにアクセスするために該URLをデコードするファイルプロトコルハンドラと、

前記シェルからURLを受信し、該URLのデータコンポーネントによってアドレス指定されるコンテンツであって、前記無線通信装置から遠隔に格納されるコンテンツを取得する遠隔ファイルプロトコルハンドラと、を含み、

前記複数のコンテンツハンドラは、

URLに対応するマークアップ言語コンテンツを受信し、該コンテンツを前記無線通信装置の画面表示に表示するマークアップ言語コンテンツハンドラを含む、請求項 1 に記載のコンピュータプログラム。

【請求項 3】

前記複数のプロトコルハンドラは、

前記シェルからURLを受信し、英数字メッセージの表示または前記無線通信装置の送信機能を起動するために、前記データコンポーネントによって特定されるコマンドを実行するメッセージプロトコルハンドラと、

前記シェルからURLを受信し、該URLのデータコンポーネントに従って前記無線通信装置の構成設定を行う構成プロトコルハンドラと、

を含む、請求項 2 に記載のコンピュータプログラム。

【請求項 4】

前記無線通信装置と共同し、

前記画面表示は、取得したコンテンツを表示し、

前記メモリは、前記画面表示に接続され、前記シェル、前記プロトコルハンドラ、および前記コンテンツハンドラを記憶するものであり、

前記プロセッサは、前記シェル、前記プロトコルハンドラ、および前記コンテンツハンドラを実行するために前記メモリに接続される、請求項 1 に記載のコンピュータプログラム。

【請求項 5】

メモリ、画面表示、およびブラウザプログラムを実行するためのプロセッサを含む無線通信装置であって、前記ブラウザプログラムは、

プロトコルコンポーネントおよびデータコンポーネントを有するURLを受信するためのシェル手段であって、前記データコンポーネントは、実行されるコマンドまたは取得されるコンテンツを特定するものであり、前記シェル手段は、前記プロトコルコンポーネントに従って前記データコンポーネントをプロトコルハンドラ手段に提供し、前記取得したコンテンツを処理のためにコンテンツハンドラ手段に提供するシェル手段と、

URLを受信し、前記データコンポーネントによって特定されるコンテンツを取得して該取得したコンテンツを前記シェル手段に提供する、または前記データコンポーネントによって特定されるコマンドを実行する、各々が前記シェル手段と通信可能に接続される複数のプロトコルハンドラ手段と、

前記取得されたコンテンツを受信し、前記無線通信装置の画面表示に出力するために該取得したコンテンツを処理する、各々が前記シェル手段と通信可能に接続される複数のコンテンツハンドラ手段と、を備える、無線通信装置。

【請求項 6】

無線通信装置であって、

画面表示と、

複数のキーと、

複数の設定可能な特徴と、

前記画面表示および前記キーに接続されるプロセッサと、

プロトコルコンポーネントおよびデータコンポーネントを有するURLを受信するための前記プロセッサによって実行されるシェルであって、前記データコンポーネントは、実行されるコマンドまたは取得されるコンテンツを特定するものであり、前記シェルは、前記プロトコルコンポーネントに従って前記データコンポーネントをプロトコルハンドラに提供し取得したコンテンツを処理のためにコンテンツハンドラに提供する、シェルと、

各々がURLを受信するために前記シェルと接続され、前記プロセッサによって実行される複数のプロトコルハンドラであって、前記データコンポーネントによって特定されるコンテンツを取得して該取得したコンテンツを前記シェルに提供する、または前記データコンポーネントによって特定されるコマンドを実行する、複数のプロトコルハンドラと、

前記プロセッサにより実行され、前記シェルに接続されるマークアップ言語コンテンツハンドラであって、URLに対応するマークアップ言語コンテンツを受信し、該コンテンツを前記画面表示に表示するマークアップ言語コンテンツハンドラと、を備え、

前記マークアップ言語コンテンツハンドラは、グループからマークアップ言語タグをデコードするものであり、該グループは、

前記複数のキーのうちの一つのアクションを定義するキータグと、

前記画面表示に周期的に表示されるヘルプテキストデータを定義するヘルプタグと、

キーに関連するメニューについてメニューアイテムを定義するキーメニュータグと、

第2のマークアップ言語ページのデータを第1のマークアップ言語ページに含めるために、第1のマークアップ言語ページとは異なる第2のマークアップ言語ページを特定するタグと、

設定に関するユーザ入力を受け付けるための入力フィールド、および設定可能な特徴についてのURLを含む式の値に等しい選択属性を定義する入力タイプタグであって、前記選択属性は、前記入力フィールドが予め選択されているか否かを示す、入力タイプタグと、

URL、および前記式の値に従って条件付きで表示される第1のマークアップ言語データを有する条件付きタグと、からなる、無線通信装置。

フロントページの続き

(72)発明者 エッガーズ, マイケル・ディー

アメリカ合衆国カリフォルニア州 9 4 6 1 1, オークランド, ラ・クエスタ・アベニュー 2 1 6
0

Fターム(参考) 5B084 AA02 AA12 AB04 AB35 BB01 BB12 DC02 DC03 DC17