(19)日本国特許庁(JP)　(12)公表特許公報(A)　(11)特許出願公表番号
特表2004-532478
(P2004-532478A)
(43)公表日　平成16年10月21日(2004.10.21)

(51)Int.Cl.[7]	F I	テーマコード（参考）
G06F 13/00	G06F 13/00 650B	

審査請求 有　予備審査請求 有　（全 157 頁）

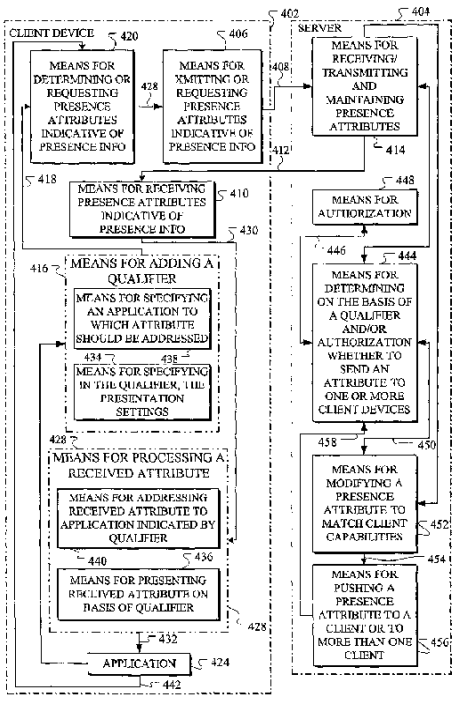
(21)出願番号	特願2002-590697(P2002-590697)
(86)(22)出願日	平成14年5月10日(2002.5.10)
(85)翻訳文提出日	平成15年11月10日(2003.11.10)
(86)国際出願番号	PCT/FI2002/000403
(87)国際公開番号	WO2002/093959
(87)国際公開日	平成14年11月21日(2002.11.21)
(31)優先権主張番号	60/290,123
(32)優先日	平成13年5月11日(2001.5.11)
(33)優先権主張国	米国(US)
(31)優先権主張番号	20012158
(32)優先日	平成13年11月7日(2001.11.7)
(33)優先権主張国	フィンランド(FI)

(71)出願人　398012616
ノキア　コーポレイション
フィンランド　エフイーエンー０２１５０
エスプー　ケイララーデンティエ　４
(74)代理人　100099759
弁理士　青木　篤
(74)代理人　100092624
弁理士　鶴田　準一
(74)代理人　100102819
弁理士　島田　哲郎
(74)代理人　100108383
弁理士　下道　晶久
(74)代理人　100082898
弁理士　西山　雅也

最終頁に続く

(54)【発明の名称】移動インスタント・メッセージング及びプレゼンス・サービス

(57)【要約】
本発明は移動メッセージング及びプレゼンス・サービスに関する。本発明の1つの態様によれば、移動メッセージング・システムのクライアント装置は、属性の使用を指定する1つかそれ以上のパラメータを備える修飾子をプレゼンス属性に追加する。プレゼンス属性を受信するクライアント装置は、受信した属性中の修飾子パラメータに応じて前記受信したプレゼンス属性を処理する。本発明の別の態様は、同じ役割のプレゼンス集合の全体または一部に加入する権利を有するメンバの関連する許可グループを有する役割内で単一のプレゼンス集合中の名称、属性及び値を有するプレゼンス項目をどのように組み立て格納するかを示すことである。



【特許請求の範囲】
【請求項１】
少なくとも1つのクライアント装置とサーバとを備える移動メッセージング・システムであって、
前記クライアント装置が、プレゼンス情報をプレゼンス属性として前記サーバに送信する手段と、プレゼンス属性を前記サーバから受信する手段とを備え、前記プレゼンス情報が、属性名称によって識別される複数のプレゼンス属性種類によって分類され、
前記サーバが、前記受信したプレゼンス属性に基づいてプレゼンス情報を維持する手段を備える移動メッセージング・システムにおいて、
前記クライアント装置が、修飾子をプレゼンス属性に追加する手段であって、前記修飾子が前記の属性の使用を指定する1つかそれ以上のパラメータを含む手段を備え、
前記クライアント装置が、前記の受信した属性中の前記修飾子パラメータに応じて前記受信したプレゼンス属性を処理する手段を備えることを特徴とする移動メッセージング・システム。
【請求項２】
前記クライアント装置が、前記修飾子中で前記属性の表示設定を指定する手段を備え、
前記クライアント装置が、前記修飾子に基づいて前記受信した属性を提示する手段を備えることを特徴とする、請求項1に記載のシステム。
【請求項３】
前記クライアント装置が、前記修飾子中で前記属性のアドレス指定先となるアプリケーションを指定する手段を備え、
前記クライアント装置が、前記受信した属性を前記修飾子によって指示されるアプリケーションにアドレス指定する手段を備えることを特徴とする、請求項1に記載のシステム。
【請求項４】
前記サーバが、前記修飾子に基づいて前記属性を1つかそれ以上のクライアント装置に送信すべきかどうかを決定する手段を備えることを特徴とする、請求項1～3の何れか一項に記載のシステム。
【請求項５】
少なくとも1つのクライアント装置とサーバとを備える移動メッセージング・システムであって、
前記クライアント装置が、プレゼンス情報をプレゼンス属性として前記サーバに送信する手段と、プレゼンス属性を前記サーバから受信する手段とを備え、前記プレゼンス情報が、属性名称によって識別される複数のプレゼンス属性種類によって分類され、
前記サーバが、受信したプレゼンス属性に基づいてプレゼンス情報を維持する手段を備える移動メッセージング・システムにおいて、
前記クライアント装置が、オーソライザと、属性名称と、修飾子との組合せによって識別されるプレゼンス情報属性を構成する手段であって、前記オーソライザが前記属性を維持する責任を負う主体を指定し前記修飾子が前記属性の使用を指定する手段を備え、
前記サーバが、受信した属性と同じ識別子を含むすでに格納された属性を探索する手段と、前記受信した属性の識別子の組合せが前記すでに格納された属性のものと同一であれば前記すでに格納された属性を前記受信した属性によって置換し、そうでなければ前記受信した属性を追加する手段を備え、
前記クライアント装置が、受信した属性と同じ識別子を含むすでに格納された属性を探索する手段と、前記受信した属性の識別子の組合せが前記すでに格納された属性のものと同一であれば前記すでに格納された属性を前記受信した属性によって置換し、そうでなければ前記受信した属性を追加する手段を備えることを特徴とする移動メッセージング・システム。
【請求項６】
前記クライアント装置から前記サーバによって受信した前記プレゼンス属性が、プレゼンス・グループに関連する発行者ユーザに応じてデータベース中に格納されることを特徴と

する、請求項1～5の何れか一項に記載の移動メッセージング・システム。
【請求項７】
各プレゼンス属性が、属性名称要素及び属性値を含む項目の一部であることを特徴とする、請求項1～6の何れか一項に記載の移動メッセージング・システム。
【請求項８】
前記属性名称要素が、前記属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含むことを特徴とする、請求項7に記載の移動メッセージング・システム。
【請求項９】
プレゼンス集合が、単一のプレゼンス・グループに関連する発行者ユーザの単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする、請求項1～8の何れか一項に記載の移動メッセージング・システム。
【請求項１０】
前記クライアント装置のユーザが発行者として前記クライアント装置または1つより多い発行者役割で1つより多いクライアント装置を使用できることを特徴とする、請求項9に記載の移動メッセージング・システム。
【請求項１１】
移動メッセージング・システムのための移動クライアント装置であって、
属性名称によって識別される複数のプレゼンス属性種類によって分類されるプレゼンス情報をプレゼンス属性としてサーバに送信する手段を備える移動クライアント装置において、
前記属性の使用を指定する1つかそれ以上のパラメータを備える修飾子をプレゼンス属性に追加する手段をさらに備えることを特徴とする移動クライアント装置。
【請求項１２】
移動メッセージング・システムのための移動クライアント装置であって、プレゼンス属性をサーバから受信する手段を備え、前記のプレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類される移動クライアント装置において、
修飾子をプレゼンス属性に追加する手段であって、前記修飾子が前記属性の使用を指定する1つかそれ以上のパラメータを備える手段と、
前記受信した属性中の前記の修飾子パラメータに応じて前記の受信したプレゼンス属性を処理する手段とをさらに備えることを特徴とする移動クライアント装置。
【請求項１３】
移動メッセージング・システムのための移動クライアント装置であって、プレゼンス情報をプレゼンス属性としてサーバに送信する手段と、
プレゼンス属性を前記サーバから受信する手段であって、前記プレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類される手段とを備える移動クライアント装置において、
オーソライザと、属性名称と、修飾子との組合せによって識別されるプレゼンス情報属性を構成する手段であって、前記オーソライザが前記属性を維持する責任を負う主体を指定し前記修飾子が前記属性の使用を指定する手段と、
受信した属性と同じ識別子を含むすでに格納された属性を探索する手段と、
前記受信した属性の識別子の組合せがすでに格納された属性のものと同一であれば前記すでに格納された属性を前記受信した属性によって置換し、そうでなければ前記受信した属性を追加する手段とをさらに備えることを特徴とする移動クライアント装置。
【請求項１４】
各プレゼンス属性が属性名称要素及び属性値を含む項目の一部であることを特徴とする、請求項11～13の何れか一項に記載の移動クライアント装置。
【請求項１５】
前記属性名称要素が、前記属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含むことを特徴とする、請求項14に記載の移動クライアント装置。

【請求項１６】
プレゼンス集合が、単一のプレゼンス・グループに関連する発行者ユーザの単一の発行者役割に属する1つかそれ以上のプレゼンス属性を備えることを特徴とする、請求項11～15の何れか一項に記載の移動クライアント装置。
【請求項１７】
前記移動クライアント装置のユーザが発行者として前記クライアント装置または1つより多い発行者役割で1つより多いクライアント装置を使用できることを特徴とする、請求項16に記載の移動クライアント装置。
【請求項１８】
移動メッセージング・システムのためのサーバであって、前記サーバが、受信したプレゼンス属性に基づいてプレゼンス情報を維持する手段を備え、前記プレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類されるサーバにおいて、
受信した属性と同じ識別子を含むすでに格納された属性を探索する手段と、
前記受信した属性の識別子の組合せが前記すでに格納された属性のものと同一であれば前記すでに格納された属性を前記受信した属性によって置換し、そうでなければ前記受信した属性を追加する手段とを備えることを特徴とするサーバ。
【請求項１９】
クライアント装置から前記サーバによって受信した前記プレゼンス属性が、プレゼンス・グループに関連する発行者ユーザに応じてデータベース中に格納されることを特徴とする、請求項18に記載のサーバ。
【請求項２０】
各プレゼンス属性が、属性名称要素及び属性値を含む項目の一部であることを特徴とする、請求項18または19に記載のサーバ。
【請求項２１】
前記属性名称要素が、該属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含むことを特徴とする、請求項20に記載のサーバ。
【請求項２２】
プレゼンス集合が、単一のプレゼンス・グループに関連する発行者ユーザの単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする、請求項18～21の何れか一項に記載のサーバ。
【請求項２３】
前記サーバと通信し発行者の役目を果たすクライアント装置のユーザが、該クライアント装置または1つより多い発行者役割で1つより多いクライアント装置を使用できることを特徴とする、請求項22に記載のサーバ。
【請求項２４】
プレゼンス・システムであって、
プレゼンス・ユーザが発行者または加入者として前記のシステムと対話できるようにする少なくとも1つのプレゼンス・クライアントを有する少なくとも1つの物理装置と、
関連するプレゼンス・グループに応じて加入者によってアクセスするため発行者の属性のプレゼンス集合の有効な値を維持するサーバとを備えるプレゼンス・システム。
【請求項２５】
各属性が、属性名称要素及び属性値を含む項目の一部であることを特徴とする、請求項24に記載のプレゼンス・システム。
【請求項２６】
前記属性名称要素が、属性の使用法に関連する情報を有する修飾子を含むことを特徴とする、請求項25に記載のプレゼンス・システム。
【請求項２７】
前記属性名称要素が、該属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含むことを特徴とする、請求項26に記載のプレゼンス・システム。
【請求項２８】

プレゼンス集合が、単一のプレゼンス・グループに関連する単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする、請求項24～27の何れか一項に記載のプレゼンス・システム。

【請求項２９】

前記システムと対話する前記ユーザが発行者として前記プレゼンス・クライアントまたは1つより多い発行者役割で1つより多いプレゼンス・クライアントを使用できることを特徴とする、請求項28に記載のプレゼンス・システム。

【請求項３０】

各属性が、属性名称要素及び属性値を含む項目の一部であることを特徴とする、請求項28に記載のプレゼンス・システム。

【請求項３１】

前記少なくとも1つの物理装置が移動物理装置であることを特徴とする、請求項24～30の何れか一項に記載のプレゼンス・システム。

【請求項３２】

物理装置中に格納するためコンピュータ可読媒体中に組み込まれたコンピュータ・プログラムにおいて、

前記のプログラムが、関連する少なくとも1つのプレゼンス・グループに応じて1つかそれ以上の加入者によってアクセスするための属性の少なくとも1つのプレゼンス集合の発行者としてプレゼンス・ユーザがプレゼンス・システムと対話できるようにするプレゼンス・クライアント・プログラムであることを特徴とするコンピュータ・プログラム。

【請求項３３】

前記プログラムが、加入者がメンバであるプレゼンス・グループに関連する属性の少なくとも1つの集合への前記加入者としてプレゼンス・ユーザが前記プレゼンス・システムと対話できるようにすることを特徴とする、請求項32に記載のプレゼンス・クライアント・プログラム。

【請求項３４】

各属性が、属性名称要素及び属性値を含む項目の一部であることを特徴とする、請求項32または33に記載のプレゼンス・クライアント・プログラム。

【請求項３５】

前記属性名称要素が、属性の使用法に関連する情報を有する修飾子を含むことを特徴とする、請求項34に記載のプレゼンス・クライアント・プログラム。

【請求項３６】

前記属性名称要素が、該属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含むことを特徴とする、請求項35に記載のプレゼンス・クライアント・プログラム。

【請求項３７】

プレゼンス集合が、単一のプレゼンス・グループに関連する単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする、請求項32～36の何れか一項に記載のプレゼンス・クライアント・プログラム。

【請求項３８】

前記のシステムと対話する前記のユーザが発行者として前記プレゼンス・クライアント・プログラムまたは1つより多い発行者役割で1つより多いプレゼンス・クライアント・プログラムを使用できることを特徴とする、請求項37に記載のプレゼンス・クライアント・プログラム。

【請求項３９】

各属性が、属性名称要素及び属性値を含む項目の一部であることを特徴とする、請求項37に記載のプレゼンス・クライアント・プログラム。

【請求項４０】

前記物理装置が移動物理装置であることを特徴とする、請求項32～39の何れか一項に記載のプレゼンス・クライアント・プログラム。

【請求項４１】
内部に格納されたコンピュータ可読媒体中に組み込まれたコンピュータ・プログラムを有する物理装置において、前記のプログラムが、請求項32～40の何れか一項に記載のプレゼンス・クライアント・プログラムであることを特徴とする物理装置。
【請求項４２】
物理装置に格納されるためにコンピュータ可読媒体中に組み込まれたデータ構造において、
該データ構造が、プレゼンス集合に関連するプレゼンス・グループに応じて加入者によってアクセスするため1つかそれ以上の発行者の属性の前記プレゼンス集合の有効な値を格納するプレゼンス・データベースであることを特徴とするデータ構造。
【請求項４３】
各属性が、属性名称要素及び属性値を含む項目の一部であることを特徴とする、請求項42に記載のデータ構造。
【請求項４４】
前記属性名称要素が、属性の使用法に関連する情報を有する修飾子を含むことを特徴とする、請求項43に記載のデータ構造。
【請求項４５】
前記属性名称要素が、該属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含むことを特徴とする、請求項44に記載のデータ構造。
【請求項４６】
プレゼンス集合が、単一のプレゼンス・グループに関連する単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする、請求項42～45の何れか一項に記載のデータ構造。
【請求項４７】
前記物理装置と対話するユーザが発行者として、1つより多い発行者役割を発行するために前記データ構造を使用でき、各役割が別個のプレゼンス・グループに関連する別個のプレゼンス集合を有することを特徴とする、請求項42～46の何れか一項に記載のデータ構造。
【請求項４８】
前記物理装置が移動装置であることを特徴とする、請求項42～47の何れか一項に記載のデータ構造。
【請求項４９】
内部に格納されたコンピュータ可読媒体中に組み込まれたデータ構造を有する物理装置において、前記データ構造が、請求項42～48の何れか一項に記載のプレゼンス・データ構造であることを特徴とする物理装置。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、移動通信システムにおけるメッセージングに関し、特に、移動インスタント・メッセージング及びプレゼンス・サービスにおけるプレゼンス属性に関する。
【０００２】
インスタント・メッセージング・サービスは、エンドユーザに、高速で対話形の、主としてテキストによる通信を行う手段を提供する。インスタント・メッセージングの有用性は、ユーザのチャット・パートナーまたは「友人」のオンライン状態及び利用可能性を追跡すると共にその状態または利用可能性の変化をユーザに通知するサービスの追加によって大きく向上する。この種のサービスは「プレゼンス・サービス」と呼ばれる。一般に、プレゼンスは様々な手段を介してインスタント・メッセージング・サービスに接続されたユーザまたはクライアントに関する様々な動的情報を含むものと考えられる。この情報の例は、通信するユーザの到達可能性、利用可能性及び位置である。インスタント・メッセージングとプレゼンス・サービスの組合せはインスタント・メッセージング及びプレゼンス

・サービス(IMPS)と呼ばれる。この種のサービスは有線インターネット・ユーザには利用可能であったが、有線ユーザと移動ユーザとの間の相互接続性は達成されていなかった。

【０００３】

ワイヤレス・ビレッジ(Wireless Village)・イニシアティブは移動インスタント・メッセージング及びプレゼンス・サービスについての仕様を規定するため設立された。ワイヤレス・ビレッジのインスタント・メッセージング及びプレゼンス・サービス(IMPS)には、プレゼンス、インスタント・メッセージング、グループ及び共有内容という4つの主要な機能が含まれる。共有内容によってユーザとオペレータとは、自分たちが画像、音楽及び他のマルチメディア内容を掲示することのできる固有の記憶領域を設定し、IMまたはチャット・セッションで他の個人及びグループと共有することができる。ワイヤレス・ビレッジ・イニシアティブによって、オペレータとエンドユーザとの両方がグループを生成し管理できるようになる。プレゼンスはワイヤレス・ビレッジ・イニシアティブにとって主要な実現技術である。インターネットによる既存のインスタント・メッセージング・サービスでは、プレゼンスの値は普通、ユーザがアクセスできる状態、不在、通信したくない、等といった非常に簡単なものである。こうした値が所定の値の集合から選択される。ワイヤレス・ビレッジの移動IMPSソリューションに関する白書「ワイヤレス・ビレッジ、移動IMPSイニシアティブ:白書(「Wireless Village,The Mobile IMPS Initiative:White Paper」)」2001年4月26日付が出版されている。既存の移動端末は、個人の状態をデスクトップ・コンピュータより正確に反映する個人ツールと考えられる。ユーザ及び移動端末から得られる情報の広範さを考慮すると、プレゼンス情報のドメインを予想することは非常に困難である。従って、新しい種類のプレゼンス情報を容易に使用及び追加することのできる機構を開発する必要がある。

【発明の開示】

【発明が解決しようとする課題】

【０００４】

本発明の目的は、すでに決定されたプレゼンス属性の他に新しいプレゼンス属性を生成するソリューションを提供することである。

【０００５】

本発明の別の目的は、クライアントが使用するプレゼンス属性をどのように編成し格納するかを示すことである。

【課題を解決するための手段】

【０００６】

本発明の第1の態様によれば、移動メッセージング・システムのクライアント装置は、属性の使用を指定する1つかそれ以上のパラメータを備える修飾子をプレゼンス属性に追加する。プレゼンス属性を受信するクライアント装置は、受信した属性中の修飾子パラメータに応じて受信したプレゼンス属性を処理する。プレゼンス属性はあるユーザ及び/またはクライアント装置に関するプレゼンス情報を記述するデータの集まりであり、そのプレゼンス情報は他のユーザを対象としたものである。また、プレゼンス情報はクライアント装置間のマシン同士の通信に関する情報を含んでもよい。

【０００７】

さらに、本発明の第1の態様によれば、少なくとも1つのクライアント装置とサーバとを備える移動メッセージング・システムであって、クライアント装置がプレゼンス情報をプレゼンス属性としてサーバに送信する手段とプレゼンス属性をサーバから受信する手段とを備え、そのプレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類され、サーバが受信したプレゼンス属性に基づいてプレゼンス情報を維持する手段を備える移動メッセージング・システムは、クライアント装置が属性の使用を指定する1つかそれ以上のパラメータを備える修飾子をプレゼンス属性に追加する手段を備え、クライアント装置が受信した属性中の修飾子パラメータに応じて受信したプレゼンス属性を処理する手段を備えることを特徴とする。

【０００８】

本発明の第2の態様によれば、クライアント装置は、オーソライザ(authorizer)、属性名称及び修飾子の組合せによって識別されるプレゼンス情報属性を構成し、その際オーソライザが属性の維持に責任を負う主体を指定し、修飾子が属性の使用を指定する。プレゼンス属性を受信すると、サーバ及びクライアント装置は、受信した属性と同じ識別子を含むすでに格納された属性を探索する。受信した属性の識別子の組合せがすでに格納された属性のものと同一であれば、すでに格納された属性を受信した属性によって置換する。そうでなければ、以前の属性を置換せずに受信した属性を追加する。

【０００９】

さらに、本発明の第2の態様によれば、少なくとも1つのクライアント装置とサーバとを備える移動メッセージング・システムであって、クライアント装置がプレゼンス情報をプレゼンス属性としてサーバに送信する手段とプレゼンス属性をサーバから受信する手段とを備え、プレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類され、サーバが受信したプレゼンス属性に基づいてプレゼンス情報を維持する手段を備える移動メッセージング・システムは、クライアント装置が、オーソライザ、属性名称及び修飾子の組合せによって識別されるプレゼンス情報属性を構成し、オーソライザが属性の維持を担当する主体を指定し修飾子が属性の使用を指定し、サーバが、受信した属性と同じ識別子を含むすでに格納された属性を探索する手段と、受信した属性の識別子の組合せがすでに格納された属性のものと同一であればすでに格納された属性を受信した属性によって置換しそうでなければ受信した属性を追加する手段とを備え、クライアント装置が、受信した属性と同じ識別子を含むすでに格納された属性を探索する手段と、受信した属性の識別子の組合せがすでに格納された属性のものと同一であればすでに格納された属性を受信した属性によって置換しそうでなければ受信した属性を追加する手段とを備えることを特徴とする。

【００１０】

さらに、本発明の第1及び第2の態様によれば、本システムは、クライアント装置が、属性の提示設定を修飾子中で指定する手段を備え、クライアント装置が、修飾子に基づいて受信した属性を提示する手段を備えることを特徴とする。

【００１１】

またさらに、本発明の第1及び第2の態様によれば、本システムは、クライアント装置が、属性のアドレス指定先となるべきアプリケーションを修飾子中で指定する手段を備え、クライアント装置が、受信した属性を修飾子によって示されるアプリケーションにアドレス指定する手段を備えることを特徴とする。

【００１２】

またさらに、本発明の第1及び第2の態様によれば、本システムは、サーバが、修飾子に基づいて属性を1つかそれ以上のクライアント装置に送信すべきかを決定する手段を備えることを特徴とする。

【００１３】

さらに、本発明の第1及び第2の態様によれば、本システムは、クライアント装置からサーバによって受信されたプレゼンス属性が、プレゼンス・グループに関連する発行者ユーザに応じてデータベースに格納されることを特徴とする。

【００１４】

またさらに、本発明の第1及び第2の態様によれば、本システムは、各プレゼンス属性が属性名称要素及び属性値を含む項目の一部であることを特徴とする。属性名称要素は、属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含んでもよい。

【００１５】

またさらに、本発明の第1及び第2の態様によれば、本システムは、プレゼンス集合が、単一のプレゼンス・グループに関連する発行者ユーザの単一の発行者役割(publisher role)に属する1つかそれ以上の属性を備えることを特徴とする。

【００１６】

またさらに、本発明の第1及び第2の態様によれば、移動メッセージング・システムは、ク

ライアント装置のユーザが発行者としてそのクライアント装置または1つより多い発行者役割における1つより多いクライアント装置を使用できることを特徴とする。

【００１７】

本発明の第3の態様によれば、移動メッセージング・システムのための移動クライアント装置であって、そのクライアント装置がプレゼンス情報をプレゼンス属性としてサーバに送信する手段を備え、そのプレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類される移動クライアント装置は、そのクライアント装置がさらに、属性の使用を指定する1つかそれ以上のパラメータを備える修飾子をプレゼンス属性に追加する手段を備えることを特徴とする。

【００１８】

本発明の第4の態様によれば、移動メッセージング・システムのための移動クライアント装置であって、そのクライアント装置がプレゼンス属性をサーバから受信する手段を備え、そのプレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類される移動クライアント装置は、そのクライアント装置がさらに、属性の使用を指定する1つかそれ以上のパラメータを備える修飾子をプレゼンス属性に追加する手段と、受信した属性中の修飾子パラメータに応じて受信したプレゼンス属性を処理する手段とを備えることを特徴とする。

【００１９】

本発明の第5の態様によれば、移動メッセージング・システムのための移動クライアント装置であって、そのクライアント装置が、プレゼンス情報をプレゼンス属性としてサーバに送信する手段と、プレゼンス属性をサーバから受信する手段とを備え、そのプレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類される移動クライアント装置は、そのクライアント装置がさらに、オーソライザ、属性名称及び修飾子の組合せによって識別されるプレゼンス情報属性を構成する手段であって、オーソライザが属性の維持に責任を負う主体を指定し、修飾子が属性の使用を指定する手段と、受信した属性と同じ識別子を含むすでに格納された属性を探索する手段と、受信した属性の識別子の組合せがすでに格納された属性のものと同一であればすでに格納された属性を受信した属性によって置換しそうでなければ受信した属性を追加する手段とを備えることを特徴とする。

【００２０】

さらに、本発明の第3、第4及び第5の態様によれば、本移動クライアント装置は、各プレゼンス属性が属性名称要素及び属性値を含む項目の一部であることを特徴とする。属性名称要素は、属性名称要素及び属性値を一意に保持する責任を負う権限を示する権限文字列を含んでもよい。

【００２１】

さらに、本発明の第3、第4及び第5の態様によれば、本移動クライアント装置は、プレゼンス集合が、単一のプレゼンス・グループに関連する発行者ユーザの単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする。

【００２２】

またさらに、本発明の第3、第4及び第5の態様によれば、本移動クライアント装置は、クライアント装置のユーザが発行者としてそのクライアント装置または1つより多い発行者役割における1つより多いクライアント装置を使用できることを特徴とする。

【００２３】

本発明の第6の態様によれば、移動メッセージング・システムのためのサーバであって、そのサーバが受信したプレゼンス属性に基づいてプレゼンス情報を維持する手段を備え、そのプレゼンス情報が属性名称によって識別される複数のプレゼンス属性種類によって分類されるサーバは、そのサーバがさらに、受信した属性と同じ識別子を含むすでに格納された属性を探索する手段と、受信した属性の識別子の組合せがすでに格納された属性のものと同一であればすでに格納された属性を受信した属性によって置換しそうでなければ受信した属性を追加する手段とを備えることを特徴とする。

【００２４】
さらに、本発明の第6の態様によれば、本サーバは、クライアント装置からサーバによって受信されたプレゼンス属性が、プレゼンス・グループに関連する発行者ユーザに応じてデータベース中に格納されることを特徴とする。
【００２５】
またさらに、本発明の第6の態様によれば、本サーバは、各プレゼンス属性が属性名称要素及び属性値を含む項目の一部であることを特徴とする。属性名称要素は、属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含んでもよい。
【００２６】
またさらに、本発明の第6の態様によれば、本サーバは、プレゼンス集合が、単一のプレゼンス・グループに関連する発行者ユーザの単一の発行者役割に属する1つかそれ以上のプレゼンス属性を備えることを特徴とする。サーバと通信し発行者の役目を果たすクライアント装置のユーザはそのクライアント装置または1つより多い発行者役割における1つより多いクライアント装置を使用できる。
【００２７】
本発明の第7の態様によれば、プレゼンス・システムは、プレゼンス・ユーザが発行者または加入者としてシステムと対話できるようにする少なくとも1つのプレゼンス・クライアントを有する少なくとも1つの物理装置と、関連するプレゼンス・グループに応じて加入者がアクセスする発行者の属性のプレゼンス集合の有効な値を維持するサーバとを備える。
【００２８】
さらに、本発明の第7の態様によれば、プレゼンス・システムは、各属性が属性名称要素及び属性値を含む項目の一部であることを特徴とする。属性名称要素は、属性の使用に関する情報を有する修飾子を含んでもよい。属性名称要素は、属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含んでもよい。
【００２９】
またさらに、本発明の第7の態様によれば、本プレゼンス・システムは、プレゼンス集合が、単一のプレゼンス・グループに関連する単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする。発行者としてシステムと対話するユーザは、プレゼンス・クライアントまたは1つより多い発行者役割における1つより多いプレゼンス・クライアントを使用できる。各属性は、属性名称要素及び属性値を含む項目の一部であってもよい。
【００３０】
またさらに、本発明の第7の態様によれば、少なくとも1つの物理装置は移動物理装置である。
【００３１】
本発明の第8の態様によれば、物理装置に格納するためコンピュータ可読媒体内に組み込まれたコンピュータ・プログラムは、そのプログラムが、関連する少なくとも1つのプレゼンス・グループに応じて1つかそれ以上の加入者がアクセスする属性の少なくとも1つのプレゼンス集合の発行者としてプレゼンス・ユーザがプレゼンス・システムと対話できるようにするプレゼンス・クライアント・プログラムであることを特徴とする。
【００３２】
さらに、本発明の第8の態様によれば、本プレゼンス・クライアント・プログラムは、このプログラムが、加入者がメンバであるプレゼンス・グループに関連する属性の少なくとも1つの集合への加入者としてプレゼンス・ユーザがプレゼンス・システムと対話できるようにすることを特徴とする。
【００３３】
またさらに、本発明の第8の態様によれば、本プレゼンス・クライアント・プログラムは、各属性が属性名称要素及び属性値を含む項目の一部であることを特徴とする。属性名称要素は属性の使用に関する情報を有する修飾子を含んでもよい。

【００３４】
さらに、本発明の第8の態様によれば、本プレゼンス・クライアント・プログラムは、属性名称要素が属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含むことを特徴とする。
【００３５】
またさらに、本発明の第8の態様によれば、本プレゼンス・クライアント・プログラムは、プレゼンス集合が、単一のプレゼンス・グループに関連する単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする。システムと対話するユーザは発行者としてプレゼンス・クライアント・プログラムまたは1つより多い発行者役割における1つより多いプレゼンス・クライアント・プログラムを使用できる。各属性は、属性名称要素及び属性値を含む項目の一部であってもよい。
【００３６】
またさらに、本発明の第8の態様によれば、本プレゼンス・クライアント・プログラムは、物理装置が移動物理装置であることを特徴とする。
【００３７】
本発明の第9の態様によれば、内部に格納されたコンピュータ可読媒体中に組み込まれたコンピュータ・プログラムを有する物理装置は、そのプログラムが本発明の第8の態様に従うプレゼンス・クライアント・プログラムであることを特徴とする。
【００３８】
本発明の第10の態様によれば、物理装置中に格納するためコンピュータ可読媒体に組み込まれたデータ構造は、そのデータ構造が、プレゼンス集合に関連するプレゼンス・グループに応じて加入者がアクセスするため1つかそれ以上の発行者の属性のプレゼンス集合の有効な値を格納するプレゼンス・データベースであることを特徴とする。
【００３９】
さらに、本発明の第10の態様によれば、データ構造は、各属性が属性名称要素及び属性値を含む項目の一部であることを特徴とする。属性名称要素は、属性の使用に関する情報を有する修飾子を含んでもよい。属性名称要素は、属性名称要素及び属性値を一意に保持する責任を負う権限を示す権限文字列を含んでもよい。
【００４０】
またさらに、本発明の第10の態様によれば、プレゼンス集合は、単一のプレゼンス・グループに関連する単一の発行者役割に属する1つかそれ以上の属性を備えることを特徴とする。
【００４１】
またさらに、本発明の第10の態様によれば、データ構造は、物理装置と対話するユーザが発行者として1つより多い発行者役割を発行するためデータ構造を使用することができ、各役割は別個のプレゼンス・グループに関連する別個のプレゼンス集合を有することを特徴とする。
【００４２】
またさらに、本発明の第10の態様によれば、データ構造は、物理装置が移動装置であることを特徴とする。
【００４３】
本発明の第11の態様によれば、内部に格納されたコンピュータ可読媒体中に組み込まれたデータ構造を有する物理装置は、そのデータ構造が本発明の第10の態様に従うプレゼンス・データベースであることを特徴とする。
【発明の効果】
【００４４】
本発明の利点は、修飾子を追加することによって所定のプレゼンス属性の調整が可能なことである。修飾子を使用して新しい属性を追加してもよい(修飾子を伴う属性は一意に識別できる、すなわち属性名称が同じだが修飾子のない属性と機能的に分離できるため)。すなわち、例えばユーザまたはアプリケーション開発者は、所定の種類に制限されること

なく、自分たちの要求に最も適合するかまたは現在の実際の状態を記述する新しいプレゼンス属性を容易に決定できる。修飾子によってプレゼンス情報の送信者(所有者)は受信クライアント装置でプレゼンス情報をどのように使用するかを指定できるようになるので、修飾子を使用することによって別の利点がもたらされる。

【００４５】

本発明の1つの実施例では、属性のアドレス指定先となるアプリケーションは修飾子で指定される。受信された属性は修飾子で示されたアプリケーションにアドレス指定される。本実施例のさらなる利点は、送信側クライアント装置は使用されるアプリケーションを規定してもよく、それによってあるプレゼンス情報をあるアプリケーションに送るため修飾子を使用してもよいということである。

【００４６】

本発明の別の利点は、同じ役割のプレゼンス集合の全体または一部に加入する権利を有するメンバの関連する許可グループを有する役割内で単一のプレゼンス集合中の名前、属性及び値によってどのようにプレゼンス項目を組み立てるかを示すことである。

【００４７】

以下、本発明は好適実施例により添付の図面を参照してさらに詳細に説明される。

【発明を実施するための最良の形態】

【００４８】

図1は、移動IMPSシステムを示す。いくつかの移動クライアント(MC)は移動ネットワークMNWと場合によっては1つかそれ以上の中間ネットワークONWとを介してIMPSサーバSに接続される。典型的には中間ネットワークとしてインターネットが使用されるので、非移動クライアントCがIMPSシステムによるサービスを提供されることもある。IMPSサーバSは、プレゼンス・サービスに関して、プレゼンス情報を所有する発行側クライアントのためのホーム・サービス要素である発行者サーバPSと、加入側または要求側クライアントのためのホームである加入者サーバSSというサーバ要素に機能的に分割できる。従って、クライアントMCは両方のサーバからサービスを提供される。すなわち、MCはPSに対してそのプレゼンス情報を更新して発行側クライアントの役目を果たし、その一方で、加入者サーバSSから、他のクライアントに関するプレゼンス情報をプレゼンス属性として要求し受信する。サーバPSは、プレゼンス情報に関するユーザの発行上の選択に基づいてプレゼンス・データを維持しその分配を管理する。SS及びPSの機能は1つの物理サーバ装置上で実行してもよくまた複数のサーバ装置に分配してもよい。

【００４９】

この説明はプレゼンス関連サービス機能に主眼を置いていることに注意されたい。他の重要な移動IMPSサービス機能としてメッセージング機能、ユーザ・グループ管理機能、内容管理機能、加入者管理機能、及びクライアント機能がある。移動IMPSサービスはこうしたサービス機能を使用することによってもたらされる。例えば、クライアントMCはいくつかのユーザ・グループに属してもよく、サーバSはグループのメンバ構成を管理し、インスタント・メッセージングを処理し、グループのメンバ間でプレゼンス情報を配信する。また、サーバの1つの重要な役割は情報フローの制御である。すなわち、サーバは、例えばプレゼンスまたは他の情報のうち、「友人」のグループのメンバに配信してよいものと、「仕事の同僚」のグループのメンバに配信してよいものと、また任意のクライアントに公開してよいものとを規定する、ユーザの選択に基づくフィルタを有してもよい。

【００５０】

様々なトランスポート層プロトコルが使用可能であり、典型的にはIPプロトコルを使用してネットワーク層サービスを提供する。様々な下位層伝送プロトコルを使用してもよい。移動ネットワークMNWは、GSMサービスをサポートするセルラ・ネットワーク、GPRSサービス(汎用パケット無線サービス)をサポートするネットワーク、UMTSネットワーク(ユニバーサル移動通信システム)のような第3世代移動通信ネットワーク、無線ローカルエリア・ネットワークWLANまたは私設ネットワークといった任意の無線ネットワークでよい。また、ブルートゥース(Bluetooth™)通信のような短距離赤外線または無線接続を、MCとサー

バSとの間の通信経路の一部として使用してもよい。移動クライアント装置MCは、例えば移動局、PDA装置、または無線モデムを備えているかまたはそれに接続されたラップトップ・コンピュータでもよい。移動IMPSメッセージは、例えば回線交換データ呼、パケット交換データ伝送コンテキスト、SMSまたはMMS(マルチメディア・メッセージング・サービス)といったメッセージング・サービスを使用して転送すればよい。図2は、プレゼンス属性の使用法を示す。移動クライアントMCは、発行すべき1つかそれ以上のプレゼンス属性を決定すると(201)、発行者サーバPSに対してプレゼンス属性の更新を行う(202)、すなわち、1つかそれ以上のプレゼンス属性を発行する。プレゼンス属性の決定(201)は、何らかのプレゼンス情報が変更された時、自動的にかまたはユーザの意志によって、クライアントが移動IMPSサーバとの論理移動IMPSセッションを確立する時なされる。例えば、段階201は、所定の時間、データの時点で自動的に開始してもよく、移動クライアントMCでのユーザ・プロファイルの変更によって開始してもよい。クライアントMC(A)は、サーバSSの移動IMPSサービスにアクセスする時、別のクライアント(B)のプレゼンス情報を要求してもよい(203)。加入者サーバSSは発行者サーバPS(クライアントAの)からのこの情報を要求する(204)。これが発行の選択(クライアントBの)によって許可されると、PSは1つかそれ以上のプレゼンス属性をSSに送信する(205)。クライアントBが発行の選択を設定することで、要求された情報の一部の(クライアントA宛、または一般的な)送信を阻止することが可能である。また、クライアントがSSのサービスとの論理接続を確立する時、SSはユーザの選択(Aの)に基づいてプレゼンス情報を自動的に要求してもよい(204)。SSはプレゼンス属性を受信側クライアントMC(A)に転送する(206)。

【００５１】

加入者サーバSS(及び発行サーバPS)は典型的には変更されていないクライアントに対してクライアントに起源を有するプレゼンス属性を送信する。しかし、サーバPSに実装された内容適応機構が存在することもある。内容適応は、受信側クライアントのクライアント機能と調和するような形でプレゼンス属性を変更するという課題を扱う。クライアントからの要求に対する応答としてプレゼンス情報を送信することに加えて、発行の選択に応じてプレゼンス情報を利用可能なクライアントMC(サービスに論理的に接続された)に送りつけることも可能である。この送りつけるやり方のプレゼンス通知は、発行者サーバが発行者クライアントから更新を受信する場合、発行者サーバが属性値の変化を検出する場合、または値を更新する実装に特有の内部トリガによりといった3つの機構によってトリガすればよい。

【００５２】

従って、クライアントMCは、PSに対して1つかそれ以上のプレゼンス属性を更新し、SSからプレゼンス属性を受信及び処理し(207)、少なくとも1つのプレゼンス属性から得たプレゼンス情報をユーザに提示するよう構成されている。MCは好適には、プレゼンス属性を伝えるために使用される更新メッセージ中の新しいプレゼンス属性値が受信される(またはクライアントが移動IMPSセッションを終了する)までプレゼンス情報(プレゼンス属性値)を格納する。さらに、後でより詳細に示されるように、クライアント装置は自動的に受信したプレゼンス情報を利用してその機能をしかるべく調整してもよい。図2に示されるシグナリングに加えて、プレゼンス属性を要求側クライアントに送信する前に、プレゼンス情報の発行者から許可を要求してもよい。図2は、例えばメッセージ202の後でサーバが応答しうる状態メッセージを示していない。

クライアント起源のプレゼンス属性とは、値フィールドが発行側クライアントによって満たされたものである。サーバ起源のプレゼンス属性とは、値フィールドが発行者サーバによって満たされたものである。プレゼンス属性のある部分がクライアントによって満たされ残りの部分が発行者サーバによって満たされている場合、そのプレゼンス属性はクライアント-サーバ起源である。本発明の好適実施例によれば、ユーザまたは組織は所定の集合の属性の他に新しいプレゼンス属性を規定してもよい。プレゼンス属性は以下の種類に分類できる。

-クライアント状態:通信するためのクライアント装置の利用可能性を記述するプレゼンス

属性。例えば、ネットワークの到達可能性、GPRS接続、オン/オフ状態、オペレータ。従って、移動IMPSサービスにおけるこれらの属性は、非移動クライアント装置のために使用されるIMPSのものとは大きく異なっている。
-ユーザ状態:通信するためのユーザの利用可能性を記述するプレゼンス属性。例えば、準備完了、会議中、手が離せない、不在、電話中、チャット中、就寝中。
-地域情報:ユーザのいる地域の環境を記述するプレゼンス属性。例えば、現地時間、騒々しい/静かな環境、室内、室外、例えば、地理的位置、訪問先PLMN、都市/通り、施設といったユーザの所在地。例えば、移動クライアントの正確な位置は地域情報属性について直接得られ、利用可能性状態(会議中、夏季別荘に滞在中、等)は移動クライアントMCのユーザ・プロファイル設定を介して容易に得られる。
-個人状態:個人ユーザ状態を記述する様々な個人属性。例えば、気分、個人の関心及び意図。
-クライアント機能:様々な通信手段、様々な媒体種類、及び様々な特徴をサポートするクライアント装置の機能を記述するプレゼンス属性。
-ユーザ属性:クライアント装置またはユーザが自己のテキスト・プレゼンス値及び外部値に対する参照を規定できるようにするプレゼンス属性。
-拡張プレゼンス情報:ベンダ特定であるかまたはサービス・プロバイダが動的に規定する非標準プレゼンス属性であるが、それにもかかわらず標準プレゼンス・サーバを通じて伝える必要があるもの。
すなわち、移動クライアント装置及び実際のユーザについて様々なプレゼンス属性が存在しうる。例えば、ユーザをメッセージを受信できるものとして規定せず、ユーザのクライアント装置をオンラインになっているものとして規定してもよい。また、SMSがベアラとして使用されている場合、ユーザはメッセージを受信することはできるがオンラインではないものとして規定してもよい。

【００５３】

プレゼンス属性の一般構造及び識別

表1はプレゼンス属性の一般構造を記述する。プレゼンス属性は一般に識別子部分と複数の値フィールドとを備えている。Reqフィールドはその要素が必須Mか、随意選択Oか、または条件付きCかを決定する。属性情報はXMLフォーマット(拡張可能マークアップ言語)である。

【００５４】

【表１】

情報要素	Req	種類	内容
NameSpec	M	XML	属性識別情報
Value	M	XML	属性についての値

表１　プレゼンス属性構造

【００５５】

NameSpec要素の下位要素を表2に記述する。

【００５６】

【表２】

NameSpec	Req	種類	内容
Name	M	文字列	属性の名称
Qualifier	O	文字列	属性使用の範囲に関する情報
Authority	C	文字列	属性のNameSpec及び値フィールドの名称が一意であることの責任を負う権限

表２　NameSpec要素

【００５７】
属性の名称は「Name」という情報要素によって与えられる文字列である。Name情報要素は表3で規定されるフォーマット中の全てのプレゼンス属性について規定される。
【００５８】
【表３】

情報要素	Name
データ種類	文字列
フォーマット	自由テキスト・フォーマット
内容	プレゼンス属性の名称

表３　Name要素

【００５９】
Qualifier要素のフォーマットを表4に示す。
【００６０】
【表４】

情報要素	Qualifier
データ種類	文字列
フォーマット	自由テキスト・フォーマット
内容	属性使用の範囲を変更する

表４　Qualifier要素

【００６１】
Qualifier要素を使用して、属性使用の範囲を指定する。修飾子は特に、新しい属性を追加するか、またはプレゼンス情報の送信者(発行者)が受信側クライアント装置でプレゼンス情報をどのように使用すべきかを指定できるようにする、という2つの目的のために使

用できる。すなわち、修飾子文字列は、受信側クライアント装置における1つかそれ以上のアプリケーションについてのパラメータとして使用してもよい。

【００６２】

例えば、発行者が自分の正確な所在地(例えば通りの番地)を知るのは一部のユーザだけ(グループAと呼ぶ)に制限し、他のユーザ(グループBと呼ぶ)にはより大まかな住所(例えば都市名だけ)を知らせるようにしたい場合、発行者は都市名を含めた位置属性をグループBに対して発行すればよい。グループAに対しては、発行者はQualifier(例えば「私の親友」といった)を位置属性に添付する。これによって事実上、Qualifier「私の親友」を持った新しい属性が生成される。次に発行者はこの新しい属性に通りの名称を含め、それをグループAに対して発行する。属性が異なっているので、サーバPSは両者の値を区別することができる。また、ある人物がグループAとBとの両方に属する場合、この人物のクライアント装置はこれらの2つの属性を区別するよう設定できる。クライアント装置は、ユーザが起動したグループに応じて属性を提示(及び場合によっては利用)するよう設定してもよい。ありうるQualifierの値はクライアント及びサービス・プロバイダが事前割当してもよく、またユーザ(発行者)が動的に割当てもよい。また、サービス・プロバイダは動的に割当てられるQualifierの値の数を制限してもよい。

【００６３】

権限要素は属性とその内容を一意に保持する責任を負う主体を決定する。これは属性拡張機構に関する。

【００６４】

【表５】

情報要素	Authority
データ種類	文字列
フォーマット	URL
内容	属性の一意性に責任を負う主体を識別する

表５　Authority要素

所定のプレゼンス属性が変更されずに使用されている場合、Authority文字列は省略できる。Qualifier文字列を使用してもAuthority要素の使用を必要とする属性は変更されない。Qualifier文字列は、新しい属性(新しいName)を導入する時か、または新しい値フィールドをすでに指定された属性に追加する時使用しなければならない。どちらの場合でも、属性は新しいものとみなされAuthorityはこの新しい属性を維持する責任を負う。

一般に、値フィールド名称は属性内で一意でなければならない。すなわち、新しい値フィールドを既存の属性に導入することは、その属性を規定した主体によって設定された規則によって処理されなければならない。新しい値フィールドを追加することによって古い属性は新しいものに変換される。新旧の属性を共存させるため、Authorityフィールドのプレゼンスはこのことをシグナリングしなければならない。また、出資者が公開維持のため属性を開放することもありうる。この種の属性はIANA(インターネット・アサインド・ナンバー・オーソリティ)のような適切な公共機関によって登録され、やはりAuthorityフィールドでシグナリングされる。この場合、任意の出資者は、Authorityフィールドを変更する必要なしに属性についての追加値フィールドを登録してもよい。サーバ(PS、SS)は、値フィールドの意味が理解できない時でも、属性から値フィールドを除去しない。クライアントMCは属性中の理解できない値フィールドは全て無視する。

【００６５】

属性は要素NameSpecによって識別され一意とされる。図3は、NameSpec要素及び修飾子の使用の例を示す。クライアント装置MC1のある属性について修飾子が決定される(301)。ユーザが修飾子を決定することも、クライアント装置MCがそれを決定することもありうる。望ましいユーザ・グループを指定し、受信側クライアント装置(MC2)でのプレゼンス情報の提示方法を決定し、あるいはそうでなければ受信側クライアント装置MC2による属性の使用方法を指定するために修飾子を使用してもよい。修飾子を決定する際移動クライアントMCのユーザ・プロファイルを利用できる。例えば、MC1は現在のプロファイル(例えば会議中)、カレンダー・エントリ(会議は12:00に終了)及び現地時間に基づいてプレゼンス属性を構成する。修飾子を使用することによって、プレゼンス属性は受信側クライアントMC2にとって有用な情報を含むよう容易に変更できる。属性はサーバPSに送信される(302)。

【００６６】

PSは受信した属性のNameSpec要素を格納された属性と比較する(303)。PSはまずAuthority文字列を互いに比較する。Authority文字列を含まない属性はAuthority文字列を含むいかなる属性とも異なっている。次に、属性名称を比較する。最後にQualifier文字列を比較する。Qualifier文字列を含まない属性はQualifier文字列を有するいかなる属性とも異なっている。2つの属性が同じなのは、これら3つの比較が全て同じ結果を出した時だけである。すなわち、修飾子を有する受信した属性を、同じ属性名称を有するが異なった修飾子を有する属性と機能的に分離することが可能である。

【００６７】

発行サーバPSはこの種の比較を行い、受信した属性の値フィールドは何れかの既存のプレゼンス情報を置換するものか、またその属性はMC2のプレゼンス情報記憶装置に追加すべき新しいものかを決定する。この比較に基づいて、PSは受信した属性中の情報を格納する(303)。前記受信した属性の全ての識別子がすでに格納された属性のものと同じであれば、PSはすでに格納された属性の前のプレゼンス情報を前記受信した属性のプレゼンス情報によって置換する。そうでなければ、PSは、前の情報を置換せず、前記受信した属性のプレゼンス情報を追加する。PSはその属性を少なくとも1つのクライアント装置MC2に送信する(304)(自動的に、またはMC2からの要求に応じて送りつける)。1つの実施例によれば、修飾子は属性の送り先となるグループを決定する。さらに、例えば電話帳の私用連絡先と公用連絡先でプレゼンス情報の提示方法を変えるため、修飾子を使用してもよい。すなわち、PSは修飾子に基づいて受信側クライアント装置を決定してもよい。動的電話帳サービスのサービス機能は、まずクライアント状態属性、ユーザ状態属性及び個人状態属性をさらに十分に以下で説明した後に説明する。

【００６８】

また、受信側クライアントは、PSの場合と同様の比較を行った(305)後で、受信したプレゼンス属性を格納する必要の有無、またその方法を決定する。プレゼンス属性のこの種の３つから成る識別によって、プレゼンス属性の非常に柔軟な使用、管理、及び生成が可能になる。

【００６９】

修飾子を使用してクライアント装置MC2における属性の使用を指定する多くの方法が存在する。典型的には、クライアント装置MCは複数のアプリケーションをサポートできる。第1の実施例によれば、クライアント装置MC1は使用すべきアプリケーションを指定する修飾子を追加する。アプリケーションとは一般に、例えばポート番号によって識別できる任意のアプリケーション・エンティティを指す。このアプリケーションはクライアント装置MC1で属性のプレゼンス情報を処理したものと同じものでもよく、別のアプリケーションでもよい。受信側クライアント装置MC2は受信した属性を修飾子が指示するアプリケーションにアドレス指定する(305)。例えば、修飾子を使用して、同じプレゼンス情報を電話帳アプリケーションとゲーム・アプリケーションとに送信してもよい。これらのアプリケーションはプレゼンス情報を異なった形で使用しうるので、アプリケーションの必要に応じて属性を正確に調整することも可能である。

【００７０】

第2の実施例によれば、クライアント装置MC1は属性の提示を指定する修飾子を追加する。受信側クライアント装置MC2は、その修飾子に基づいて受信した属性を提示する(305)。修飾子は、例えば、情報全てをユーザに示すべきか否か、又は情報のどの部分を示すべきかを決定してもよい。また、修飾子は、色、フォント等の様々なユーザ・インタフェース設定を決定してもよい。すなわち、MC2のUIは修飾子中の設定に基づいて設定される(305)。

【００７１】

クライアント状態属性

好適実施例によれば、移動クライアントMCは、移動クライアントMCの現在の伝送機能を記述するプレゼンス属性を利用する。この種の属性の構造を表6に示す。この属性は、モデム属性と呼ばれ、移動ベアラを取り扱うユーザ端末部分または機能に関するプレゼンス情報を提供する。

【００７２】

【表６】

情報要素	Req	種類	値	内容
NameSpec Name	M	文字列	モデム	属性の名称
Value	M	XML	下記の表7参照	属性の値 型は構造である

表６　モデム属性構造

【００７３】

【表７】

モデム値	Req	単一(S)/多重(M)	内容
Status	M	S	値フィールドの名称
CommAddr	O	S	値フィールドの名称
CS_Status	O	S	値フィールドの名称
PS_Status	O	S	値フィールドの名称
RoamingStatus	O	S	値フィールドの名称
CS_CallStatus	O	M	値フィールドの名称
PDP_ContexStatus	O	M	値フィールドの名称

表７　モデム属性の値フィールド

【００７４】

表8に示されるStatus値フィールドは移動モデムの状態を示す。

【００７５】

【表８】

情報要素	Status
データ種類	列挙文字列
フォーマット	以下の値: ON-端末のモデム部分は電源が入っている OFF-端末のモデム部分は電源が切られているかまたは到達範囲外である DIS-この属性によって示された値フィールド(もしあれば)は無効である。以前の更新から得られた全ての値フィールドは無効である
内容	モデム属性のStatusフィールド
範囲	ON|OFF|DIS

表８　モデムのStatus値フィールド

【００７６】
Status値フィールドはモデムの電源が入っているか否かを示す。好適実施例によれば、プレゼンス属性には、好適には属性の必須Status値フィールド中にあるDIS指示が含まれる。属性中にDISが設定されると、その属性中で与えられた全ての値は無効になる。従って、受信側移動クライアントMCはその属性の値フィールドを無視することができる。また、そのプレゼンス属性の前の値も削除される(及びゼロにされる)。すなわち、DIS指示を有するが他の値フィールドのない属性を送信するのが効果的である。この種の属性が必要とする空間は非常に小さいので、無線インタフェース上の必要な帯域幅を節約できる。特にユーザ属性を記述する属性の場合、このことは特に有用である。
【００７７】
再び表7を参照すると、モデム属性のCommAddr値フィールドにはモデム(MC)の通信アドレスが含まれる。これは、通信手段と連絡先アドレスという2つの部分を含む。手段部分は、サポートされる通信方法、特に、モデムがパケット交換(PS)データ、回線交換(CS)データまたは音声、SMSまたはMMSのどれをサポートするかに関する情報を伝える。連絡先部分にはアドレス、例えばMSISDN番号が含まれる。
【００７８】
CS_Status値フィールドは、モデムの回線交換状態(登録または未登録)を示す。PS_Status値フィールドは、モデムのパケット交換状態(接続または未接続)を示す。RoamingStatus値フィールドはホームPLMN(公衆陸上移動ネットワーク)と、場合によってはモデムが現在ローミングしているPLMNとを示す。CS_CallStatus値フィールドはCSベアラの着信呼状態(データまたは音声、アクティブまたは非アクティブ)を提供する。モデムが多重呼機能をサポートする場合、モデム属性はこうした着信呼状態のリストを有してもよい。PDP_Context Status値フィールドには、QoS(サービス品質)情報のようなPDP(パケット・データ・プロトコル)コンテキストに関する情報が含まれる。
【００７９】
上記の例に加えて、移動クライアントの伝送機能に関する他の情報を伝えるためにモデム値フィールドを使用してもよい。第1の例では、移動クライアントの最大ビットレートがMode(モード)属性で配信される。従って、受信側クライアント装置は最大ビット速度を越えないように伝送速度を設定できる。第2の例では、クライアント装置は、データ・ファイルをクライアント装置に送信する時パケット交換伝送モードだけを使用するべきかを決定する。第3の例は、ローミング装置がある種の通信だけを可能にする(例えば、音声呼だ

けを許可し、データ・ファイルは送信しない)よう命令するものである。

【００８０】

ユーザ状態属性

1つの好適実施例によれば、ある活動に参加しようというユーザの意欲に関する属性が規定される。活動はこのAvailability属性に属する値フィールドによって指定される。表9は、Availability属性に関する構造を示す。

【００８１】

【表９】

情報要素	Req	種類	値	内容
NameSpec Name	M	文字列	Availability	属性の名称
Value	M	XML	下記の表10参照	属性の値

表９　Availability属性構造

表10は、Availability属性の値フィールドを記述する。

【００８２】

【表１０】

Availability拡張値	Req	単一(S)/多重(M)	内容
Status	M	S	値フィールドの名称
CommAvail	O	S	値フィールドの名称
PhoneAvail	O	S	値フィールドの名称
SMSAvail	O	S	値フィールドの名称
MMSAvail	O	S	値フィールドの名称
IMAvail	O	S	値フィールドの名称
EmailAvail	O	S	値フィールドの名称
Image	O	M	値フィールドの名称
Text	O	M	値フィールドの名称

表１０　Availability属性値

【００８３】

表11に示される状態値フィールドは利用可能性情報の状態を示す。

【００８４】

【表１１】

情報要素	Status
データ種類	列挙文字列
フォーマット	以下の値のうちの1つ: ENA-この属性に含まれる値フィールドは最新の情報を含む。以前に更新されこの属性に含まれない値フィールドはまだ最新である DIS-この属性に含まれる値フィールド(もしあれば)は無効な情報を含む。以前に更新された値フィールドは無効である
内容	利用可能性属性の発行状態を規定する
範囲	ENA|DIS

表１１　Statusフィールド

【００８５】
Status値フィールドは利用可能性情報の発行が可能か否かを示す。すでに説明したように、DIS指示を使用してAvailability属性の値を無効化してもよい。例えば、サーバPSは、移動クライアントMCが移動IMPSセッションを終了した後で、DIS指示を伴う利用可能性属性を送信してもよい。また、クライアントへの接続が突然途絶えた時この種のメッセージを送信してもよい。従って、受信側移動クライアントは、もはや移動IMPSシステム中に存在しないユーザ及びクライアント装置に関する全ての利用可能性情報を削除できる。
【００８６】
表10のCommAvail値フィールドは、ユーザが何らかの形態の遠隔通信に参加する意向であるかを示す。PhoneAvail値フィールドは、ユーザが電話呼に参加する意向であるかを示す。SMSAvail値フィールドは、ユーザがSMS交換に参加する意向であるかを示す。MMSAvail値フィールドは、ユーザがMMS(マルチメディア・メッセージング・サービス)交換に参加する意向であるかを示す。IMAvail値フィールドは、ユーザがIM(インスタント・メッセージング)交換に参加する意向であるかを示す。EmailAvail値フィールドは、ユーザが電子メール交換に参加する意向であるかを示す。
【００８７】
Image値フィールドに関する構造を表12に示す。
【００８８】
【表１２】

Image	Req	内容
ContaindImage	C	転送符号化形式で属性中に含まれる画像
ReferredImage	C	画像へのURL
ValueField	M	Status、ImageまたはText以外のこの属性中の任意の値フィールドの名称

表１２　Image

【００８９】
この値フィールドは、Status、TextまたはImageフィールド以外のAvailability属性中の値フィールドの何れにも画像を関連付ける。ContainedImage値フィールドには画像が含まれるが画像のサイズまたはフォーマットは制限されることがある。ReferredImage値フィールドには関連する画像を有するリソースへのURLが含まれる。ValueField値フィールドは、画像が関連付けられる値フィールドを規定する。例えば、発行者は、DISCの意味に関する画像意味情報を伝えるため、現在「DISC」という値を有する値フィールドPhoneAvailに画像を関連付けてもよい(例えば、ユーザが電話を利用できるだけに制限されていることを意味する)。画像フィールドは、この属性中で多数のインスタンスを有してもよい。この値フィールドが属性中に含まれる時はいつでも、その目標値も同じ属性中に含まれなければならない。この関連付けは目標値フィールドが有効である限りにおいてのみ有効である。目標値フィールドが更新または無効化されると、この属性への古い関連付けがもしあれば、受信側クライアントによって破棄されなければならない。
【００９０】
Text値フィールドに関する構造を表13に示す。
【００９１】
【表１３】

Text	Req	内容
ContainedText	M	テキスト文字列
ValueField	M	Status、ImageまたはText以外のこの属性中の任意の値フィールドの名称

表１３　Text

【００９２】
Text値フィールドは、Status、ImageまたはTextフィールド以外のAvailability属性中の値フィールドの何れにもテキスト文字列を関連付ける。Text値フィールドには、ContainedText中のテキスト文字列とValueField中の関連する値フィールドの名称とが含まれる。ContainedText要素中のテキストのサイズは制限されることがある。例えば、発行者は、NA

VLの意味に関する追加意味情報を伝えるため、現在「NAVL」という値を有する値フィールドPhoneAvailにテキストを関連付けてもよい(例えば、「14:00まで会議中」)。テキスト値フィールドはこの属性中で多数のインスタンスを有してもよく、すなわち、同じテキストが多数の値フィールドに関連付けられることもある。この値フィールドが属性中に含まれる時はいつでも、その目標値フィールドも同じ属性中に含まれなければならない。この関連付けは目標値フィールドが有効である限りに置いてのみ有効である。目標値フィールドが更新または無効化されると、この値フィールドへの古い関連付けがもしあれば、この値フィールドは受信側クライアントによって破棄されなければならない。画像及びテキストは自動的にプレゼンス属性に追加されることもある。

【００９３】

個人状態属性

PersonalStatus属性は、発行者の個人状態を示す。選択項目と詳細はこの属性に属する値フィールドによって指定される。表14はPersonalStatusに関する属性構造を示す。

【００９４】

【表１４】

情報要素	Req	種類	値	内容
NameSpec Name	M	文字列	PersonalStatus	属性の名称
Value	M	XML	下記の表15参照	属性の値

表１４　PersonalStatus属性構造

【００９５】

表15は、PersonalStatus属性の値フィールドを示す。

【００９６】

【表１５】

PersonalStatus 値	Req	単一(S)/多重(M)	内容
Status	M	S	値フィールドの名称
Text	O	S	値フィールドの名称
Mood	O	S	値フィールドの名称
Time	O	S	値フィールドの名称
Image	O	M	値フィールドの名称

表１５　PersonalStatus属性値

【００９７】
表16に示される状態値フィールドはPersonalStatus情報の状態を示す。
【００９８】
【表１６】

情報要素	Status
データ種類	列挙文字列
フォーマット	以下の値のうちの1つ: ENA-この属性に含まれる値フィールドは最新の情報を含む。以前に更新されこの属性に含まれない値フィールドはまだ最新である DIS-この属性に含まれる値フィールド(もしあれば)は無効な情報を含む。以前に更新された値フィールドは無効である
内容	位置属性の発行状態を規定する
範囲	ENA|DIS

表１６　PersonalStatus属性の状態

【００９９】
このフィールドは、この情報の発行が可能か否かを示す。PersonalStatus属性と共にDIS指示を使用してもよい。
【０１００】
Text値フィールドは自由なテキスト形式で発行者の状態を示す。Mood値フィールドは発行者の気分を示す。Time値フィールドは発行者の現地時間を提供する。
【０１０１】
好適実施例によれば、PersonalStatus属性中で画像を利用してもよい。表12ですでに示さ

れたように、Image値フィールドはある画像をStatusまたはImageフィールド以外のこの属性中の値フィールドの何れにも関連付ける。例えば、IN_LOVEの意味に関する画像意味情報を伝えるため、現在「IN_LOVE」という値を有する値フィールドMoodに画像を関連付けてもよい。このImage値フィールドは、Availability属性についてすでに示されたのと同様の方法で使用すればよい。

【０１０２】

本発明は、既存のクライアント装置及びサーバ中で実現できる。それらは全て、上記で説明した本発明の機能を実現可能なプロセッサ及びメモリを有している。コンピュータ・プログラムは内部または外部メモリからサーバまたはクライアント装置のプロセッサにロードすればよく、それをプロセッサ中で実行すれば、本発明の機能を実現する手段を与えることになる。また、ハードウェア実装またはソフトウェアとハードウェア実装との組合せを使用してもよい。

【０１０３】

図4は、本発明による、サーバ404と通信するクライアント装置402を示す。クライアント装置は、図1の1つかそれ以上の移動クライアントまたは非移動クライアント、または図2及び図3に示された任意の他の移動クライアントと同様のものでよい。同様に、サーバ404は図1に示されたサーバまたは図2及び図3に示された任意の他のサーバと同様のものでよい。クライアント装置402には、プレゼンス情報をプレゼンス属性として信号線路408上のサーバに送信または要求する手段406が含まれる。プレゼンス情報の送信は、例えば、プレゼンス属性がMC1からPSに送信される図3のステップ302か、または、プレゼンス属性がMCによって更新されPSに送信される図2のステップ202に対応する。線路408上でプレゼンス情報を要求することは、MCがSSからプレゼンス属性を要求する図2のステップ203に相当する。線路408上のクライアント装置402からサーバ404への送信は、図1に示された、移動クライアント(MC)から移動ネットワーク(MNW)を通じ中間ネットワークONWを介してIMPSサーバに至る伝送経路と同様のものでもよい。又は、これは非移動クライアントCから中間ネットワークONWを介してサーバSに至る経路と同様のものでもよい。もちろん、他の経路も考えられ、本発明は使用される物理媒体または物理媒体の組合せの経路に依存しない。また、クライアント装置402には、信号線路412上のサーバ404からプレゼンス属性を受信する手段が含まれる。このプレゼンス情報は、属性名称によって識別される複数のプレゼンス属性種類によって分類される。

【０１０４】

サーバ404には、属性を受信/送信し、受信したプレゼンス属性に基づいてプレゼンス情報を維持する手段414が含まれる。

【０１０５】

本発明によれば、クライアント装置402にはさらに、修飾子をプレゼンス属性に追加する手段416が含まれ、その際修飾子は属性の使用を指定する1つかそれ以上のパラメータを備えている。追加された修飾子は信号線路418上で提供され、手段406によって線路408上でサーバ404に送信される。これは、線路422上でアプリケーション424から受信された命令に応じてプレゼンス属性を決定する手段420を介して行ってもよい。また、アプリケーションは、プレゼンス属性を要求するための手段420を利用してもよい。どちらの場合でも、信号は線路426上で手段420から、線路408上でプレゼンス属性を送信または要求する手段406に提供すればよい。例えば、図2のステップ202によりプレゼンス属性を更新する際に手段416が修飾子を追加した場合、線路408上の信号には、属性の使用を指定する1つかそれ以上のパラメータを有する修飾子を伴うプレゼンス属性が含まれる。

【０１０６】

線路412上でサーバ404から着信したプレゼンス属性に対処するため、クライアント装置402にはまた、受信した属性中の修飾子パラメータに応じて線路412上でサーバ404から受信したプレゼンス属性を処理する手段428が含まれる。線路412上の受信プレゼンス属性は手段410によって受信され、線路430上で受信した属性を処理する手段428に対して提供されればよい。処理の後、手段428は線路432上でアプリケーション424に対して信号を提供す

ればよく、信号はアプリケーション424によってさらに使用される。

【０１０７】

修飾子を追加する手段416は、サーバ404から属性を受信したクライアントが提示設定に基づいて属性を提示できるように、属性の修飾子に提示設定を指定する手段434を含んでもよい。その結果、図4のクライアント装置402のようなクライアント装置には、他のクライアント装置によって指定され線路412上でサーバ404から受信された修飾子に基づいて受信した属性を提示する手段436が含まれる。

【０１０８】

同様に、手段416には、サーバ404に送信される修飾子中で、受信側クライアントにおいて属性のアドレス指定先となるアプリケーションを指定する手段438が含まれる。クライアント装置402が属性のアドレス指定先となるアプリケーションを指定する修飾子を受信する場合、線路430上で受信した属性を解釈し受信した属性を修飾子が指示するアプリケーションにアドレス指定する手段440を有する。

【０１０９】

ここでサーバ404をさらに詳細に検討すると、サーバ中のプレゼンス・データベースのプレゼンス・グループによって指定された1つかそれ以上のクライアント装置に属性を送信するかどうかを修飾子に基づいて決定する手段444を含んでもよい。このような決定は、線路446上で許可を提供する手段448から与えられた許可に応じて行ってもよい。修飾子及び許可によって属性を1つかそれ以上のクライアント装置に送信するよう指示されたならば、サーバ404は、例えばクライアント402または同様の装置に対して適切な場合それを行う。

【０１１０】

特定のクライアント向けのプレゼンス属性が、サーバにとって既知の情報によればクライアントの機能に適合しないこともありうる。このような情報は例えば、線路450上の手段444によって、クライアントの機能に合うようプレゼンス属性を変更する手段452に提供すればよい。変更された属性は線路450上の手段444に戻せばよい。他方、プレゼンス属性が「一方的に送りつける」方式で提供される場合、変更されたプレゼンス属性を線路454上で手段456に提供すればよく、手段456は変更されたプレゼンス属性を対象クライアントまたは1つより多いクライアントに必要に応じて送りつける適切なステップを取ることができる。これは例えば線路458上で手段444に対して信号してもよい。

【０１１１】

認識されるように、図4に示される機能ブロックは、当業者に周知の個別ハードウェア、専用集積回路、マイクロコントローラ、ソフトウェア、ファームウェア等を使用して実行すればよい。さらに、図面中で別個の機能ブロックに属する機能は分離されている必要はなく、機能を他の機能ブロックに自由に追加またはそこから削除することによって他のブロックに組み込むことができる。同様に、機能ブロック間の協調関係をその順序や相互関係について変更し、同時に上記で言及された同じ最終結果を実行してもよい。また、やはり認識されるように、本発明によれば、図4に示されたクライアント装置及びサーバ装置の細部は図示されたものと同様の他の形態を取ってもよい。本発明の他の態様は、類似しているが同一ではない形で示してもよい。

【０１１２】

例えば、図5はサーバ装置504と通信するクライアント装置502を示すが、そこでは図4の手段420と同様の手段506は、プレゼンス情報をプレゼンス属性として線路508上で、図4のサーバ404内の手段414と同様の、サーバ504内のプレゼンス属性を受信し維持する手段510に送信する。同様に、クライアント装置502は、サーバ504の手段510から線路514上でプレゼンス情報を示すプレゼンス属性を受信する手段512を含んでもよい。図4のクライアント装置402の手段420及び416と同様、図5のクライアント装置502は、オーソライザ、属性名称及び修飾子の組合せによって識別されるプレゼンス情報属性を構成する手段516を含んでもよく、その際オーソライザは属性の維持に責任を負う主体を指定し、修飾子は属性の使用を指定する。上記のように構成された属性は、線路508上でサーバ504に送信するために

図示されるように線路518上で手段506に提供されてもよい。サーバ504は、線路522上で手段510から受信した上記のように構成された属性に応答して、許可、属性名称及び修飾子の同じ組合せを含む、すでに格納された属性を探索する手段520を含んでもよい。受信した属性の識別子の組合せが、すでに格納された属性、例えば記憶手段524中に格納されたものと同一であれば、受信した属性を使用して、信号線路526上のすでに格納された属性を置換する。この方法で、属性パラメータが変化した場合、更新されたパラメータが記憶手段524中に格納されるようになる。そうでなければ、受信した属性は新しい属性として記憶手段に追加される。この機能は上記で説明したように手段520で実行してもよく、また完全に独立した手段528で実行してもよいが、この手段528は線路530上で手段520から受信属性を受信し、識別子の組合せが同じであればすでに格納された属性を受信した属性によって置換し、そうでなければそれ自体と記憶手段524との間の接続線路532上で受信した属性を追加するものである。

【０１１３】

クライアント装置502には上記で説明したのと同様の機能が含まれ、そのことは例えば、線路542上で手段512から着信属性を受信し、受信した属性と同じ識別子を含む、すでに格納された属性を探索する手段540と、受信した属性の識別子の組合せがすでに格納された属性のものと同一であればすでに格納された属性を受信した属性によって置換する手段542とによって示される。そうでなければ、受信した属性は記憶手段544に追加される。探索手段540は手段542を内部に含んでもよく、またそれらは図示されるように独立したものでもよい。後者の場合、信号線路546は手段540から手段542に対して受信した属性に関する属性情報を提供する。記憶手段544は、線路548上で属性を置換または追加する手段542と双方向通信し、線路550上ですでに格納された属性を探索する手段540と双方向通信してもよい。

【０１１４】

上記で言及したように、ここで本発明の動的電話帳サービス実施例のためのサービス機能を説明する。動的電話帳サービスはリッチコール(rich call)・サービスと考えてもよい。これは「電話をかける前に」B当事者のプレゼンス情報をA当事者に示す機能を充実させるため有用である。この場合、B当事者は1つかそれ以上のユーザ電話帳エントリである。プレゼンス情報は、(1)クライアントの利用可能性、(2)ユーザの利用可能性、(3)現地の条件、(4)個人の状態、(5)クライアントの機能、(6)ユーザの属性、及び(7)拡張プレゼンス・サービスといった、上記で言及したのと同じカテゴリに分割できる。

【０１１５】

概念上、プレゼンス・システムは、図6に示されるように、プレゼンス・クライアント602、604、606と、プレゼンス・ユーザ608、610、612と、プレゼンス・ユーザ役割614、616、618、620、622、624と、プレゼンス・プロキシ626と、プレゼンス・サーバ628、630からなる。プレゼンス・クライアントはユーザがプレゼンス・システムと対話できるようにするソフトウェアまたはプログラムである。ユーザはプレゼンス・クライアントを使用してプレゼンス・システムと対話する人物である。例えば移動送受話器またはPCのような物理装置632、634は、1つのプレゼンス・クライアント・インスタンス606を有してもよいが、特殊な場合には、多数のプレゼンス・クライアント・インスタンス602、604を有してもよい。プレゼンス・クライアントは単一のユーザによって所有される。ユーザは1つより多いクライアントを有してもよいが、こうしたクライアントは通常異なった装置に置かれる。

【０１１６】

ユーザ608、610、612は概念上発行者と加入者とに分類される。発行者はプレゼンス情報の発信者である。加入者はプレゼンス情報の受信者である。ユーザは、自分自身のプレゼンス情報の発行者であると同時にある他の発行者のプレゼンス情報の加入者であってもよい。ユーザは1つかそれ以上の役割を有してもよい。発行者役割は、プレゼンス集合と呼ばれるプレゼンス値の集合に関連付けられる。同じユーザの2つの異なるプレゼンス集合のプレゼンス値は互いに独立しており異なる役割に関連付けられる。加入者役割は、同一

の発行者役割のプレゼンス情報、すなわち同じプレゼンス集合の論理受信者である。

【０１１７】

プレゼンス・プロキシ626は随意選択のネットワーク要素でありプレゼンス・サービスの拡張性を改善する。プロキシは、発行者からサーバに向かうアップリンク方向か、またはサーバから加入者に向かうダウンリンク方向に伝わる異なったプレゼンス集合のプレゼンス値を一時的に格納する。クライアントがオンラインになる時、プロキシは現在のプレゼンス情報によってクライアントを更新すればよい。また、発行者が新しいプレゼンス値をサーバに送信する時、プロキシはプロキシによって登録された全ての加入者クライアントを更新すればよい。プロキシは現在の値を一時的にだけキャッシュできる。プレゼンス情報が発行者から到来しても、プロキシは、このプレゼンス情報の全ての更新が同じプロキシ経由で行われていると想定することはできない。プロキシがプレゼンス集合に関連する加入者グループを知らない場合、プロキシはこの情報をサーバから要求すればよい。

【０１１８】

プレゼンス・サーバ628、630は、各プレゼンス集合に関連するグループに関する有効なプレゼンス値と情報を維持するネットワーク要素である。サーバは直接またはプロキシを通じてプレゼンス・クライアントと通信する。サーバは、プロキシによってキャッシュされたプレゼンス値の有効期間についてプロキシに通知する。有効期間が切れると、プロキシは値を破棄するか、またはサーバからリフレッシュしなければならない。サーバは、プレゼンス値が変化する頻度を監視することによって、プレゼンスの項目毎に有効期間を割当てる。有効期間は動的であり、すなわちプレゼンス項目の有効期間中に変化してもよい。

【０１１９】

プレゼンス・サーバは、図6に示されるように、他のプレゼンス・サーバともプレゼンス情報を交換する。例えば、発行者と加入者とが管理上異なったプレゼンス・サーバに属する場合、プレゼンス情報は両方のサーバを通らなければならない。サーバの互換性がない場合、一方または両方のサーバにゲートウェイ機能が存在する必要がある。

【０１２０】

図7は、本発明によるプレゼンス・データベース702の構造を示す。1つのプレゼンス項目704は、名称708、属性710及び値712といった3つのプロパティを有する。プレゼンス集合714は1つまたはそれ以上のプレゼンス項目からなる。プレゼンス集合714はユーザの1つの役割716に属する。1つの役割に対して1つより多いプレゼンス集合は存在できない。さらに、1つの役割716に属する1つの許可グループ718が存在する。許可グループは、同じ役割のプレゼンス集合の全体または一部に加入する権利を有するメンバからなる。

【０１２１】

プレゼンス集合中の項目704、720、…、722は一意であり、すなわち互いに区別できる。項目は主として名称によって互いに区別される。プレゼンス集合が同じ名称を有する2つかそれ以上の項目を含む場合、各項目にははそれらの項目のIDを伝える属性がなければならない。グループ718のメンバ724、726、…、728は一意である。同じ発行者734の異なった役割716、730、…、732における異なったプレゼンス集合は同じ名称またはIDを有する項目を含んでもよい。異なったグループも同様に同じメンバを含んでもよい。

【０１２２】

役割716は、役割ID、グループIDまたはプレゼンス集合IDによって識別すればよい。例えば、グループIDはサービス・プロバイダによって割当ててもよく、サービス・プロバイダ・ドメイン内で一意である。従って、プレゼンス・データベース中の個々の要素をアドレス指定するため以下のIDが必要になる。

GroupID;ItemName

【０１２３】

プレゼンス集合が同じ名称を有する1つより多い項目を含む場合、ItemNameにはItemID属性を割当てなければならない。

【０１２４】

UserID、DeviceID及びClientIDといったIDは必要ないことに注意されたい。

【０１２５】
A. 典型的プレゼンス・プロトコル
A.1.0 未加入のプレゼンス(図8)
ユーザのプレゼンス情報は、例えば図2のステップ203、204、205、206で提示されたメッセージの流れによって示されるように問い合わせを発行するか、図9の場合のように加入してプレゼンス項目を受信するか、または図8に示されるようにプレゼンス項目を獲得するかによってメッセージ・サービスとは別個に獲得しなければならない。
【０１２６】
プレゼンス・サービスのユーザは、任意の適切な時点で、図8に示されるようにプレゼンスの更新メッセージを送信することによって、プレゼンス・サーバ中の自分のプレゼンス情報を更新すればよい。すでに言及したように、ユーザは、図8に示されるように、プレゼンスの獲得メッセージを発行してある他のユーザのプレゼンス情報を要求してもよい。プレゼンス情報は要求したユーザに返送される。
【０１２７】
発行者は自分のプレゼンス情報を部分的にだけ更新してもよい。同様に、ユーザは部分的なプレゼンス情報だけを要求してもよい。
【０１２８】
ユーザが要求されたプレゼンス情報に対して許可を有さない場合、空の内容がユーザに送信される。
【０１２９】
ある所定のユーザに対するプレゼンス情報の許可は、プレゼンス集合に対応するグループにそのユーザを含めることによってなされる。このことは以下、「プレゼンス・データベースの管理」と題された部分で説明される。
【０１３０】
A.1.1 UpdatePresence
方向:プレゼンス・クライアント→プレゼンス・サーバ
内容モデル:(TransactionID, GroupID, Presence)
属性:なし
使用法:このプリミティブ(primitive)は、プレゼンス・サーバ上の1つかそれ以上のプレゼンス項目の値を更新する。更新すべきプレゼンス項目はPresenceという情報要素で伝送される。所定のGroupIDについてサーバ中に存在しない項目がPresence中に存在する場合、サーバはこの新しい項目のために記憶装置を割当て、このメッセージの内容をコピーする。そうでなければ、サーバは古い値を新しい値によって置換する。
【０１３１】
内容にはUserIDまたはPublisherIDは必要ない。発行者とUpdatePresenceプリミティブを含むXML文書との間の関連付けは認証プロトコルで行われる。
【０１３２】
A.1.2 TransactionID
内容モデル:(#PCDATA)
属性:なし
使用法:これはクライアントからサーバへの要求をサーバからクライアントへの対応する応答に関連付ける一意の識別子である。クライアントは、第1の応答を得る前に1つより多い要求をサーバに送信していてもよい。第1の応答は必ずしも第1の要求に関連するものではない。従って、要求と応答とを互いに関連付ける機構が必要である。典型的には、TransactionIDはシーケンス番号である。これは要求においてクライアントによって割当てられ、応答でサーバによって使用される。
【０１３３】
A.1.3 GroupID
内容モデル:(#PCDATA)
属性:なし

使用法:このIDはサービス・プロバイダ・ドメイン内で許可グループとプレゼンス集合とを含む発行者役割を一意に識別する。これはプレゼンス・サービス・プロバイダによって割当てられる。

【０１３４】

A.1.4 UpdateStatus

方向:プレゼンス・サーバ→プレゼンス・クライアント

内容モデル(TransactionID,StatusCode)

属性:なし

使用法:このプリミティブは、クライアントのUpdatePresence要求に対するプレゼンス・サーバからの応答である。StatusCodeは、サポートしない、新しい割当て、成功、失敗といった値を有してもよい。

【０１３５】

A.1.5 GetPresence

方向:プレゼンス・クライアント→プレゼンス・サーバ

内容モデル:(TransactionID,((GroupID,PresenceNames?)*)|PresenceNames?)

属性:なし

使用法:このプリミティブは、プレゼンス・サーバからプレゼンス情報を要求するために使用される。ゼロまたはそれ以上のGroupIDが存在しうる。各GroupIDについて、随意選択のPresenceNamesが結合されうる。GroupIDが使用されない場合、要求範囲はユーザがメンバである全てのグループとなる。サーバは、認証プロトコルによって要求を出したユーザのIDを要求に関連付ける。PresenceNamesは後でこの文書中で規定される。これは、値の要求先である名称によってプレゼンス項目のリストを作成する。これは随意選択である。これはGroupIDと結合されてもよく、GroupIDのリストが作成されていない場合単独でもよい。GroupIDと結合されると、サーバは所定のグループに対するプレゼンス項目の探索を制限する。存在しない場合、全てのプレゼンス項目の値が要求される。

【０１３６】

A.1.6 PresenceItems

方向:プレゼンス・サーバ→プレゼンス・クライアント

内容モデル(TransactionID,StatusCode,(GroupID,Presence)*)

属性:なし

使用法:このプリミティブは、要求されたプレゼンス項目と要求の結果状態とのリストを供給する。要素Presenceは所定のGroupIDに対応するプレゼンス集合であり、後で本明細書で規定される。StatusCodeは、許可されない、項目が利用できない、成功、失敗といった値を有してもよい。

【０１３７】

A.2.0 加入済のプレゼンス(図9)

プレゼンス情報を配信するもう1つの機構は誰かのプレゼンス情報に加入することである。メッセージの流れを図9に提示する。

要求を出すユーザはプレゼンス加入メッセージA.2.1をプレゼンス・サーバに送信し、誰かのプレゼンス情報に加入する。

プレゼンス情報の加入が完了すると、要求を出したユーザは、最初に、また他の当事者がプレゼンス情報を更新した時は常に新しいプレゼンス情報A.2.2を受信する。

要求を出したユーザが、それ以上プレゼンス情報を受信したくない場合、プレゼンス情報の加入の取消しを行えばよい(A.2)。代りに、プレゼンス情報はある期間だけ加入してもよく、その場合加入取消しメッセージは必要ない。

要求を出したユーザは、プレゼンス情報のうち、自分が獲得する許可を得ている部分だけに加入できる。

【０１３８】

A.2.1 SubsPresence

方向:プレゼンス・クライアント→プレゼンス・サーバ

内容モデル:(TransactionID,SubsPeriod?,((GroupID,PresenceNames?)*)|PresenceNames?)
属性:なし
使用法:このプリミティブは、加入者が許可されている発行者役割からのプレゼンス情報に加入するために使用される。随意選択のGroupIDが加入する発行者役割を指定する。GroupIDがない場合、加入者が許可されている全ての役割に加入する。各GroupIDを、加入した項目をPresenceNamesに列挙したグループ内に制限するPressenceNamesに関連付けてもよい。随意選択のSubsPeriodは加入期間の長さを指定する。これがない場合、加入は対応するUnsubsPresenceプリミティブが呼び出されるか、またはサービス・プロバイダが加入を終了するまで継続する。PresenceNamesは許可された集合から加入したプレゼンス項目を指定する。これがない場合、プレゼンス集合(GroupID)内のユーザが許可を有する全てのプレゼンス項目に加入する。GroupIDなしで単独で使用される場合、許可された全てのグループから指名された項目に加入する。PresenceNamesは後で本明細書で規定される。加入者の身元は認証プロトコルで発見される。

【０１３９】

A.2.2 SubsPeriod
内容モデル:(#PCDATA)
属性:なし
使用法:これは加入期間の長さを規定する。時間単位はTBDである。代りに、この内容は加入が終了する日付を示してもよい。

【０１４０】

A.2.3 PushPresence
方向:プレゼンス・サーバ→プレゼンス・クライアント
内容モデル:(GroupID,Presence)+
属性:なし
使用法:このプリミティブは、発行者によって変更された加入したプレゼンス項目をユーザに送りつける。GroupIDは役割を識別する。

【０１４１】

A.2.4 UnsubsPresence
方向:プレゼンス・クライアント→プレゼンス・サーバ
内容モデル:(TransactionID,((GroupID,PresenceNames?)*)|PresenceNames?)
属性:なし
使用法:このプリミティブは、プレゼンス情報の加入を取消すために使用される。加入の終了は所定のグループだけに制限してもよく、さらにあるプレゼンス項目だけに制限してもよい。GroupIDが与えられない場合、加入の終了を全てのグループ中の指名されたプレゼンス項目だけに制限することが可能である。GroupIDとPresenceNamesが与えられない場合、ユーザの加入全体が終了する。

【０１４２】

A.2.5 SubsStatus
方向:プレゼンス・サーバ→プレゼンス・クライアント
内容モデル:(TransactionID,StatusCode,(GroupID,PresenceNames)*)
属性:なし
使用法:このプリミティブは、加入の取消し動作に対してStatusCodeを返送し、まだ加入しているプレゼンス項目とグループのリストを作成する。StatusCodeは、サポートしない、存在しない項目、未加入の項目、成功、失敗といった値を有してもよい。

【０１４３】

A.3.0 プレゼンス・データベースの管理
プレゼンス・ユーザは、プレゼンス・サーバで自分のユーザ・グループとプレゼンス集合とを管理してもよい。
ユーザ・グループとプレゼンス集合とを含む役割をCreatePresGroup(プレゼンス・グルー

プの生成)メッセージA.3.1を使用して生成する。プレゼンス・サーバは、要求された動作の状態と生成されたグループのIDを示すPresGroupStatusメッセージA.3.6によって応答する。PresenceInfoメッセージA.3.7がメンバが獲得または加入することを許可されるプレゼンス項目を示すグループのメンバに送信される。
プレゼンス・ユーザは、GetPresGroupStatusメッセージA.3.8によってグループ情報を要求してもよい。グループ情報の要求はグループの所有者に制限される。
ユーザ・グループを所有するプレゼンス・ユーザはグループのメンバまたはプレゼンス集合のプレゼンス項目を追加または削除してもよい。
発行者はDeletePresGroupメッセージによってグループの削除を送信してもよい。

【０１４４】

A.3.1 CreatePresGroup
方向:プレゼンス・クライアント→プレゼンス・サーバ
内容モデル:(TransactionID,MemberList, Presence)
属性:なし
使用法:このプリミティブは、プレゼンス・サーバで役割を生成するようサーバに要求する。許可グループはmemberlist中のメンバを含み、プレゼンス集合はPresence中に含まれる。

【０１４５】

A.3.2 MemberList
内容モデル:(MemberDescription)+
属性:なし
使用法:メンバのリスト。記述はユーザに起源を有する。

【０１４６】

A.3.3 MemberDescription
内容モデル:(UserOriginatedMD?, ServerOriginatedMD?)
属性:なし
使用法:ユーザ起源部分はユーザが見たメンバの記述である。サーバ起源部分はサーバ・データベース中に格納されたメンバの記述である。

【０１４７】

A.3.4 UserOriginatedMD
内容モデル:(#PCDATA)
属性:なし
使用法:これは人物を識別するユーザ起源の手段である。これは名称、対象人物の電話の1つのMSISDN、電子メール・アドレス、SIPアドレス等でよい。また、それらの組合せでもよい。この記述のデータ・フォーマットは本明細書の範囲内ではないが、vCardのような周知のフォーマットを使用すればよい。

【０１４８】

A.3.5 ServerOriginatedMD
内容モデル:(#PCDATA)
属性:なし
使用法:これは人物を識別するサーバ起源の手段である。これは名称、対象人物の電話の1つのMSISDN、電子メール・アドレス、SIPアドレス等でよい。また、サーバ・データベースに格納されたそれらの組合せでもよい。この記述のデータ・フォーマットは本明細書の範囲内ではないが、vCardのような周知のフォーマットを使用すればよい。

【０１４９】

A.3.6 PresGroupStatus
方向:プレゼンス・サーバ→プレゼンス・クライアント
内容モデル:(TransactionID,StatusCode,(GroupID,MemberList,PresenceNames)*)
属性:なし
使用法:このプリミティブは、現在存在する発行者役割のGroupIDをその役割に関するグル

ープ・メンバ及びプレゼンス項目名と共に返送する。サーバは、その動作が成功した場合、新しい役割に関する新しいGroupIDを割当てている。サーバはmemberlistのサーバ起源部分を満たしている。これによって発行者は、グループが正しいメンバを含んでいることを検証できる。サーバは、ユーザ起源部分中にすでに存在するレベル、または人物を一意に識別するために必要なレベルを越えてmemberlist中にどのような機密または個人情報も提供すべきではない。これはTBDである。StatusCodeは、サポートしない、未知のメンバ、未知のプレゼンス項目、成功、失敗といった値を獲得しうる。

【０１５０】

A.3.7 PresenceInfo

方向:プレゼンス・サーバ→プレゼンス・クライアント

内容モデル:(GroupID,Presence?)+

属性:なし

使用法:このプリミティブは、新しいプレゼンス項目を許可グループに定期的に配送する。このメッセージは、加入及び未加入両方の、所定の許可グループの全てのプレゼンス項目を含む。このメッセージは、グループのメンバ構成が変更された場合新しいメンバに送信され、新しいプレゼンス項目が追加された場合グループの全てのメンバに送信される。このメッセージは、GroupIDに関連するプレゼンス集合(Presence)が存在しないようにして削除されたグループ・メンバに送信される。

【０１５１】

A.3.8 GetPresGroupStatus

方向:プレゼンス・クライアント→プレゼンス・サーバ

内容モデル:(TransactionID,GroupID*)

属性:なし

使用法：このプリミティブは、サーバからのPresGroupStatusメッセージを要求する。ユーザは自分自身のグループからだけ情報を要求してよい。GroupIDは情報を所定のグループに制限する。GroupIDが存在しない場合、状態は要求者が所有する全てのグループから戻される。

【０１５２】

A.3.9 AddMembers

方向:プレゼンス・クライアント→プレゼンス・サーバ

内容モデル:(TransactionID,(GroupID,MemberList)+)

属性:なし

使用法:このプリミティブは、指名されたメンバを所定のグループに追加するようサーバに要求する。

【０１５３】

A.3.10 RemoveMembers

方向:プレゼンス・クライアント→プレゼンス・サーバ

内容モデル:(TransactionID,(GroupID,MemberList)+)

属性:なし

使用法:このプリミティブは、指名されたメンバを所定のグループから削除するようサーバに要求する。MemberListにはサーバ起源情報が満たされる。

【０１５４】

A.3.11 DeletePresGroup

方向:プレゼンス・クライアント→プレゼンス・サーバ

内容モデル:(TransactionID,GroupID*)

属性:なし

使用法:このプリミティブは、1つかそれ以上の発行者役割を削除するようサーバに要求する。GroupIDは削除すべき役割を識別する。GroupIDが与えられない場合、全ての発行者役割を削除する。

【０１５５】

A.3.12 AddPresence
方向:プレゼンス・クライアント→プレゼンス・サーバ
内容モデル:(TransactionID,(GroupID,Presence)+)
属性:なし
このプリミティブは、所定のプレゼンス項目を所定の役割に追加するようサーバに要求する。
【０１５６】
A.3.13 RemovePresence
方向:プレゼンス・クライアント→プレゼンス・サーバ
内容モデル:(TransactionID,(GroupID,Presence?)+)
属性:なし
使用法:このプリミティブは、所定のプレゼンス項目を所定の役割から削除するようサーバに要求する。GroupIDに関連するPresence要素がない場合、全てのプレゼンス項目が役割から削除される。
【０１５７】
A.3.14 動的クライアント変更
発行者または加入者は自分のクライアントをいつでも変更してよい。典型的にはクライアント装置も変更される。まれには、ユーザは同じ装置でクライアントのインスタンスだけを変更することもある。新しい装置がTEで古い装置がMTであってもよく、またその逆であってもよい。また新しいMTに変更することも可能である。クライアントの変更は、ユーザに自分のプレゼンス・データを古いクライアントから新しいクライアントにコピーまたは同期することを要求する。発行者の場合、所定のクライアントについて静的で同期集合中に含まれないあるクライアント特有のプレゼンス・データがあることもある。同期は局所的に行ってもよく、ネットワーク支援型であってもよい。何れの場合でも、既存の同期プロトコルを使用すればよい。これはプレゼンスプロトコルの範囲外である。
【０１５８】
加入者クライアント変更によってもたらされるもう1つの問題は、新しいクライアントのレジストリを、プレゼンス関連送りつけ型メッセージを受信するアクティブ・クライアントにすることである。レジストリは、
-ユーザが同じ装置で新しいクライアントを起動する(自動)か、
-ユーザのSIMが新しい装置に変更される(自動)か、
-ユーザは新しい装置中に別のSIMを有しているが、新しい装置だけの電源が入っている(半自動)か、または
-ユーザは新しい装置中に別のSIMを有しており、新しい装置と古い装置との両方の電源が入っている(手動)か、
に応じて、自動(ユーザの確認は必要ない)、半自動(ユーザの確認が必要)または手動(ユーザはレジストリを起動しなければならない)で行えばよい。
全ての場合で、レジストリは下位レベル・プロトコルによって処理すればよい。例えば、SIPプロトコルがプレゼンス文書用に選択されたトランスポート・プロトコルである場合、レジストリについてもこのプロトコルを使用すればよい。
【０１５９】
A.4.0 プレゼンス・プロトコル用DTD
【０１６０】
【表１７】

<!-- Root element -->

<!ELEMENT PresProtocol (UpdatePresence | UpdateStatus | Get-Presence | PresenceItems | SubsPresence | PushPresence | UnsubsPrecence | SubsStatus | CreatePresGroup | PresGroupStatus | PresenceInfo | GetPres-GroupStatus | AddMembers | RemoveMembers | DeletePresGroup | AddPres-ence | RemovePresence) >

<!ATTLIST PresProtocol Version NMTOKENS #REQUIRED >

<!ELEMENT UpdatePresence (TransactionID, GroupID, Presence) >

<!ELEMENT TransactionID (#PCDATA) >

<!ELEMENT GroupID (#PCDATA) >

<!ELEMENT Presence (#PCDATA) >

<!ELEMENT UpdateStatus (TransactionID, StatusCode) >

<!ELEMENT StatusCode (#PCDATA) >

<!ELEMENT GetPresence (TransactionID, ((GroupID, PresenceNames?)*) | PresenceNames?) >

<!ELEMENT PresenceNames (#PCDATA) >

<!ELEMENT PresenceItems (TransactionID, StatusCode, (GroupID,

【０１６１】
【表１８】

```
Presence)*) >

<!ELEMENT SubsPresence (TransactionID, SubsPeriod?, ((GroupID,
PresenceNames?)*) | PresenceNames?) >

<!ELEMENT SubsPeriod (#PCDATA) >

<!ELEMENT PushPresence ((GroupID, Presence)+) >

<!ELEMENT UnsubsPresence (TransactionID, ((GroupID,
PresenceNames?)*) | PresenceNames?) >

<!ELEMENT SubsStatus (TransactionID, StatusCode, (GroupID,
PresenceNames)*) >

<!ELEMENT CreatePresGroup (TransactionID, MemberList,
Presence) >

<!ELEMENT MemberList ((MemberDescription)+) >

<!ELEMENT MemberDescription (UserOriginatedMD?,
ServerOriginatedMD?) >

<!ELEMENT UserOriginatedMD (#PCDATA) >

<!ELEMENT ServerOriginatedMD (#PCDATA) >

<!ELEMENT PresGroupStatus (TransactionID, StatusCode,
(GroupID, MemberList, PresenceNames)*) >

<!ELEMENT PresenceInfo ((GroupID, Presence?)+) >

<!ELEMENT GetPresGroupStatus (TransactionID, GroupID*) >

<!ELEMENT AddMembers (TransactionID, (GroupID, MemberList)+)
```

【０１６２】
【表１９】

```
<!ELEMENT RemoveMembers (TransactionID, (GroupID,
MemberList)+) >

<!ELEMENT DeletePresGroup (TransactionID, GroupID*) >

<!ELEMENT AddPresence (TransactionID, (GroupID, Presence)+) >

<!ELEMENT RemovePresence (TransactionID, (GroupID,
Presence?)+) >
<!-- End of DTD -->
```

【０１６３】

B.　　典型的プレゼンス内容フォーマット

上記で述べたように、プレゼンス内容は以下のクラスに分割できる。

クライアントの利用可能性:通信に対するクライアントの利用可能性を記述するプレゼンス属性、例えば、ネットワークの到達可能性、GPRS接続、オン/オフ状態。

ユーザの利用可能性:通信に対するユーザの利用可能性を記述するプレゼンス属性、例えば、準備完了、会議中、手が離せない、不在、電話中、チャット中、就寝中、等。

地域条件:ユーザの地域環境を記述するプレゼンス属性、例えば、現地時間、騒騒しい/静かな環境、室内、室外、例えば、地理的位置、訪問先PLMN、都市/通り、施設といったユーザの所在地。

個人状態:個人ユーザ状態を記述する様々な個人属性。例えば、気分、個人の関心及び目的。

クライアント機能:例えば、様々な通信手段、様々な媒体種類、及び様々な機能をサポートするクライアントの機能を記述するプレゼンス属性。

ユーザ属性:クライアントまたはユーザが自分自身のテキスト・プレゼンス値及び外部値に対する参照を規定できるようにするプレゼンス属性。

拡張プレゼンス・サービス:サービス・プロバイダが動的に規定する非標準プレゼンス属性であるが、それにもかかわらず標準プレゼンス・サーバ及びプロキシを通じて伝える必要があるもの。

【０１６４】

B.1.　　設計の目標

プレゼンス通信の大部分は、プレゼンス項目の値部分の変更を加入者に送信することである。1つの値だけが変更されることも多い。転送されるデータの量を最小化するため、設計の目標は、プレゼンス項目の階層表示をできる限り平坦に保持することである。このため、プレゼンス項目は、上記で説明したプレゼンス・クラスに従って詳しく調べられる。その代りに、クラスはプレゼンス項目中で随意選択属性として与えられてもよい。

【０１６５】

D.2　　共通属性

プレゼンス要素は多様な属性を含むが、その大部分は任意のプレゼンス項目中で使用してよい。

【０１６６】

D.2.1　　Version

内容構文(content syntax):NMTOKENS

必須(mandatory):はい
使用法:これはプレゼンス内容フォーマットのバージョンを示す。
【０１６７】
D.2.2 Class
内容構文:NMTOKEN
必須:いいえ
使用法:これは属性が属するクラスを示す。
とりうる値は、CLIENT_AVAILABILITY、USER_AVAILABILITY、LOCAL_CONDITIONS、PERSONAL_STATUS、CLIENT_CAPABILITIES、USER_ATTRIBUTES、EXT_PRES_SERVICEである。
【０１６８】
D.2.3 ID
内容構文:PUBIDLITERAL
必須:いいえ
使用法:これは指名されたプレゼンス項目についての固有のIDである。これはクライアントによって割当てられる。
【０１６９】
D.2.4 Cacheability
内容構文:NMTOKEN
必須:いいえ
使用法:これはプレゼンス項目がプロキシによってキャッシュできるか否かを示す。とりうる値はYES、NOである。これはクライアントによって割当てられる。この属性がない場合、NOの値が想定される。
【０１７０】
D.2.5 ValidityPeriod
内容構文:NMTOKEN
必須:いいえ
使用法:これはキャッシュされたプレゼンス項目の有効期間である。この値は時間(単位:秒)である。これはプレゼンス・サーバによって割当てられる。
【０１７１】
D.2.6 DeviceName
内容構文:NMTOKEN
必須:いいえ
使用法:これはクライアント装置の名前である。これはクライアントによって割当てられる。
【０１７２】
D.2.7 Accuracy
内容構文:NMTOKEN
必須:いいえ
使用法:これは位置決め装置の精度である。これはメートル単位で示される。
【０１７３】
D.2.8 ImageType
内容構文:NMTOKEN
必須:いいえ
使用法:これは画像の内容符号化である。とりうる値はJPEG、GIF、BMPである。
【０１７４】
D.2.9 SoundType
内容構文:NMTOKEN
必須:いいえ
使用法:これは音声を符号化するために使用される音声コーデックの種類である。とりうる値はAMR、EFR、MP3、AAC、MIDIである。

【０１７５】
D.2.10 ExtRef
内容構文:PUBIDLITERAL
必須:いいえ
使用法:これは外部参照を示すURLである。
【０１７６】
D.2.11 ExtRefChange
内容構文:NMTOKEN
必須:いいえ
使用法:これは、外部参照が変更されたことを示す。とりうる値はYES、NOである。
【０１７７】
D.2.12 ContentChange
内容構文:NMTOKEN
必須:いいえ
使用法:これは内容値の変更を示す0～255のカウンタである。サーバまたはプロキシは最新の32の値の内容を格納してもよい(但ししなくてもよい)。最新の32の値の中の2つの内容について同じ値があればそれは同じ内容に対応する。
【０１７８】
D.3 クライアントの利用可能性
D.3.1 DeviceOn
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod、DeviceName
使用法:これはユーザ端末のオン/オフ状態を示す。発行者は、グループ管理メッセージを使用することによって、この項目をサーバ中の役割に追加する。その後、値部分がネットワークによって維持される。ユーザは自分のプレゼンス情報中に1つより多い端末を有してもよい。この場合、ID属性の使用は必須である。この内容は「ON」または「OFF」の値を有しうる。ユーザの端末がオンであるがネットワーク到達範囲外である場合、ネットワークはこの項目について「OFF」の値を割当てる。
【０１７９】
D.3.2 DeviceRoaming
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod、DeviceName
使用法:これは、発行者が訪問先のネットワークでローミング中であることを加入者に示す。ユーザは自分のプレゼンス情報中に1つより多い端末を有してもよい。この場合、ID属性の使用は必須である。この内容は「ON」または「OFF」の値を有しうる。
【０１８０】
D.3.3 NetworkType
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod、DeviceName
使用法:これは発行者が現在接続している移動ネットワークの種類を示す。この内容は「2G」、「3G-99」、「3G-R4」または「3G-R5」といった値をとりうる。ユーザは自分のプレゼンス情報中に1つより多い端末を有してもよい。この場合、ID属性の使用は必須である。
【０１８１】
D.4 ユーザの利用可能性
D.4.1 UserStatus
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:これは、発行者が容認するのに同意する邪魔の程度について発行者の現在の状態を示す。「availabie」、「silent」、「in-car」、「busy」といった値が規定される。

【０１８２】
D.4.2 PreferredContact
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:これは発行者の現在の好適な連絡方法が何かを示す。「PHONE」、「VOICE_MESSAGE」「MESSAGE」、「MAIL」、「NO_CONTACT」といった値が規定される。
【０１８３】
D.4.3 PreferredDefaults
内容モデル:((TimePeriod, PrefCont)*)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:これは発行者がプレゼンス・サーバに送信しうるプレゼンス・メタ情報である。この情報は他のプレゼンス情報同様役割に関連するが、この項目は加入者には利用できない。その代わり、これは「PreferredContact」要素の値を制御する。これは、所定期間のデフォルトの好適連絡方法を指定する。ユーザが指定する期間は重なってもよく、またある期間が他の期間を完全に含んでもよい。サーバは、所定の期間が満了するとPreferredDefaultsに応じてPreferredContact要素の値を変更する。但しこれは、ユーザがPreferredContactの値を直接変更することを阻止するものではない。また、ユーザがPreferredContactの値を直接変更することは、期間が満了した時にサーバがそれを再度変更することを阻止するものでもない。ユーザは、内容が空のこの要素をサーバに送信することによって、デフォルト機構を削除してもよい。
【０１８４】
D.4.4 TimePeriod
内容モデル:(PeriodStart, PeriodEnd, RepetitionPeriod?, PeriodPrecedence)
属性:なし
使用法:これは、期間の開始、例えば開始時間、期間の終了、例えば終了時間、反復期間、例えば「DAY」及び期間の優先順位を記述する。
【０１８５】
D.4.5 PeriodStart
内容モデル:(#PCDATA)
属性:なし
使用法:これは開始期間である。これはTime、Time-DayofWeek、Time-DayofMonthまたは完全な日付でよい。
【０１８６】
D.4.6 PeriodEnd
内容モデル:(#PCDATA)
属性:なし
使用法:これは開始期間である。これはTime、Time-DayofWeek、Time-DayofMonthまたは完全な日付でよい。分離の度合いはPeriodStartと同じでなければならない。
【０１８７】
D.4.7 RepetitionPeriod
内容モデル:(#PCDATA)
属性:なし
使用法:これは開始-終了の反復期間を示す。これは「DAY」、「WEEK」、「MONTH」の値のうちの1つでなければならない。この要素が期間の記述中に含まれない場合、反復は適用されないものと想定される。
【０１８８】
D.4.8 PeriodPrecedence
内容モデル:(#PCDATA)
属性:なし
使用法:この要素は重なり合う期間の間の衝突を解決する。これは0と9の間の数値を有し

うる。数字0が最高の優先順位である。2つかそれ以上の期間が重なり合っている場合、サーバは最高の優先順位値に関連付けられた希望連絡方法を取る。

【０１８９】

D.4.9 PrefCont

内容モデル:(#PCDATA)

属性:なし

使用法:これは、発行者にとっての期間に関連付けられたデフォルトの好適な連絡方法が何かを示す。これは、「PreferredContact」要素と同じ値を有しうる。

【０１９０】

D.5 地域条件

D.5.1 LocalTime

内容モデル:(#PCDATA)

属性:Class、ID、Cacheability、ValidityPeriod

使用法:これは発行者の現地時間を示す。

【０１９１】

D.5.2 MeasuredLocation

内容モデル:(#PCDATA)

属性:Class、ID、Cacheability、ValidityPeriod、Accuracy

使用法:これはクライアント装置の測定位置を示す。測定はセンサによるもの(例えばGPS)またはネットワークによるものまたは両者の組合せでよい。Accuracyは、その方法によって達成される平均位置決め精度を示す。この内容には少なくとも横方向位置(x及びy座標)が含まれるが、垂直位置を含んでもよい。サービス・プロバイダが以下説明する「ConvertedLocation」をサポートする場合、このMeasuredLocationはサービス・プロバイダの変換処理、例えば地図照合のための入力情報となりうる。

【０１９２】

D.5.3 ConvertedLocation

内容モデル:(#PCDATA)

属性:Class、ID、Cacheability、ValidityPeriod

使用法:この情報は典型的にはサービス・プロバイダ起源であるが、発行者起源であってもよい。これは、通りの名前のような人間に理解可能なテキスト形式で示されたユーザの位置を示す。この情報は、測定された位置を、例えば地図照合によってこの形式に変換することによって得られる。

【０１９３】

D.5.4 StatedLocation

内容モデル:(#PCDATA)

属性:Class、ID、Cacheability、ValidityPeriod

使用法:これは、発行者自身が表明した発行者の位置である。この内容は短いテキスト文字列である。

【０１９４】

D.5.5 UserEnvironment

内容モデル:(EnvAttributes+)

属性:Class、ID、Cacheability、ValidityPeriod

使用法:これはユーザに関するいくつかの環境属性を示す。

【０１９５】

D.5.6 EnvAttributes

内容モデル:(#PCDATA)

属性:なし

使用法:規定される属性は、「INDOOR」、「OUTDOOR」、「QUIET」、「NOISY」、「ALONE」、「IN_GROUP」である。

【０１９６】

D.6 個人状態
D.6.1 StatusText
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:これはユーザが書いてもよい短いテキスト(約30文字)である。
【０１９７】
D.6.2 StatusImage
内容モデル:(#CDATA)
属性:Class、ID、Cacheability、ValidityPeriod、ImageType
使用法:これはユーザが自分の状態情報に添付してもよい画像である。これは、例えばbase64である転送符号化形式のXML内容で伝送される。ImageType属性は、画像の内容符号化、例えばjpegを記述する。
【０１９８】
D.6.3 StatusSound
内容モデル:(#CDATA)
属性:Class、ID、Cacheability、ValidityPeriod、SoundType
使用法:これはユーザが自分の状態情報に添付してもよい短い(約5～30秒の)音声クリップである。これは、例えばbase64である転送符号化形式のXML内容で伝送される。SoundType属性は、音声の内容符号化、例えばAMR、MP3、AAC、MIDI等を記述する。
【０１９９】
D.7 クライアント機能
プレゼンスのコンテキストでのクライアント機能とは、様々な種類の人間対人間の通信のためのクライアントのホストとなる装置の機能を意味する。これは、アプリケーション・ソフトウェアがクライアント装置の既存のハードウェア及びソフトウェア機能を最大限に利用できるようにする場合とは大きく異なるさらに簡単な問題である。
人間対人間の通信のクラスは、メッセージング、電子メール、音声呼及びマルチメディア呼である。特に、プレゼンスは通信クラスから除外される。プレゼンスの目的は他者に通信手段及び他のユーザ属性を知らせることであって双方向通信自体ではない。このため、プレゼンス情報を分類するために「Client Capability」という用語を使用することはやや紛らわしい面がある。
クライアント機能はネットワーク機能で調べられる。これは例えば、発行者がビデオ呼機能を有しているのは、発行者のクライアント装置がその機能を有し、かつ発行者が現在ローミングしているネットワークがこの機能をサポートする時だけである、という意味である。このため、クライアント機能情報は動的である。
【０２００】
D.7.1 MessagingCapabilities
内容モデル:(MessType*)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:これはクライアント装置の動的メッセージング機能リストを示す。リストが空であるのは、メッセージング機能が現在存在しないことを示す。
【０２０１】
D.7.2 MessType
内容モデル:(#PCDATA)
属性:なし
使用法:これはクライアント装置のメッセージング機能の種類を示す。内容は、「SMS」、「MMS」または「X-メッセージ・アプリケーション名」のうちの1つでよい。メッセージ・アプリケーション名のフィールドは、装置特有のメッセージング方法、例えばスマート・メッセージングである。
【０２０２】
D.7.3 EmailClient

内容モデル:(EmailClientType*)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:これはクライアント装置の動的電子メール機能を示す。リストが空であるのは、電子メール機能が現在存在しないことを示す。
【０２０３】
D.7.4 EmailClientType
内容モデル:(#PCDATA)
属性:なし
使用法:これはクライアント装置の電子メール・クライアントの種類を示す。内容は、「SMTP」、「POP3」、「IMAP4」または「X-mail application name」のうちの1つでよい。クライアント・ソフトウェアはメール・プロトコル名自体をユーザに提供すべきではない。その代わり、クライアント・ソフトウェアはユーザ装置の電子メール環境を理解し、この電子メール環境の範囲内でプロトコル名を解釈すべきである。また、ユーザに提示される情報はユーザにとって有意であるべきである。
【０２０４】
D.7.5 VoiceCallCapability
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:これはクライアント装置の動的音声呼機能を示す。内容は、「NONE」、「VOICE_CALL」、「RICH_VOICE_CALL」のうちの1つである。
【０２０５】
D.7.6 MultimediaCallCapabilities
内容モデル:(MMCap*)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:これはユーザの動的単方向及び双方向通信機能を示す。リストが空であるのは機能が現在存在しないことを示す。
【０２０６】
D.7.7 MMCap
内容モデル:(#PCDATA)
属性:なし
使用法:これはマルチメディア呼機能の種類を示す。これは、「UPLINK_VIDEO_STREAMING」、「DOWNLINK_VIDEO_STREAMING」、「VIDEO_CALL」、「RICH_VIDEO_CALL」のうちの1つでよい。
【０２０７】
D.8 ユーザ属性
このクラスはユーザ規定及びクライアント特有両方のプレゼンス・データを含む。
【０２０８】
D.8.1 UserPresenceItem
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod、ExtRef、ExtRefChange
使用法:これはユーザが固有のプレゼンス項目を規定する手段を示す。同じ役割について1つより多いユーザ指定プレゼンス項目が規定される場合、ID属性の使用が必須である。内容は短いテキストでよい。随意選択の属性ExtRefはこのプレゼンス項目が参照する外部オブジェクトのURLである。このURLは、プレゼンス項目のXML文書を伝える多くの部分に分かれたMIMEの別の部分を参照してもよく、また外部オブジェクトへの参照でもよい。随意選択の属性ExtRefChangeを使用して、外部オブジェクトが変更されたことを加入者に示してもよい。
【０２０９】
D.8.2 ClientTypeRequest
内容モデル:(EMPTY)

属性:Class、ID、Cacheability、ValidityPeriod
使用法:これは発行者が加入者のクライアント種類を要求するために使用する専用の要素である。発行者がこの要素をある役割に関するプレゼンス・データに含める場合、サーバは、PushPresenceメッセージと共にプレゼンス情報の全ての加入者にこれを送信する。これは、随意選択で、非加入者が特定のプレゼンス項目を要求するために使用するPresence Itemsメッセージ中に含まれてもよい。この要素はPresenceInfoメッセージでは定期的に配送されない。これは加入できるプレゼンス要素ではない。プレゼンス・サーバは、最初にこの項目を加入者に送信した後で、周期的にこの項目をその後のPushPresenceメッセージに含め、全ての加入者クライアントが確実にこの項目を有するようにしてもよい。
クライアントは、この項目を受信すると、自分のクライアント種類をプレゼンス・サーバに送信する。さらに、この項目がクライアントに格納されたプレゼンス情報である場合、クライアントは起動する都度その種類をサーバに送信すべきである。
発行者が自分の加入者のクライアント種類を知る別の方法は、発行者が加入者となってクライアント種類を申し込むことであろう。しかしこれは異なったビジネスモデルである。発行者は必ずしも自分自身の加入者のうちのひとりになりたいとは限らない。この要素によって発行者は、発行者とプレゼンス・サービス・プロバイダとの間の既存の契約上の枠組の中でクライアント種類を得ることができる。さらに、この方法によって、加入者クライアントは自分の種類をサーバに送信するよう義務付けられる。発行者が加入モデルを使用してクライアント種類を獲得している場合、この情報を発行者に対して許可するかどうかは加入者の自由である。

【０２１０】

D.8.3 ClientType
内容モデル:(ClientName、ClientManufacturer、ClientVersion)
属性:Class、ID、Cacheability、ValidityPeriod
使用法:加入者クライアント種類の知識は、プレゼンス集合に対するクライアント特有の拡張を可能にするために発行者が使用する。加入者クライアントが拡張項目を復号できない場合、発行者がこうした拡張項目を使用する意味はない。
加入者クライアントは、UpdatePresenceメッセージを使用してクライアント種類情報をプレゼンス・サーバに送信する。メッセージで使用されるGroupIDは発行者の役割に属し、加入者自身の役割には属さない。これは、ユーザがプレゼンス情報の更新を許可されているのは自分自身の役割だけであるという一般原則の例外である。プレゼンス・サーバは、PushPresenceメッセージを使用してこの要素の情報を発行者に送信する。使用されるGroupIDは発行者役割に属する。発行者の視点から言うと、発行者は自分自身のプレゼンス集合からプレゼンス情報を得る。
発行者は自分自身のクライアント種類を自分のプレゼンス集合に含めてもよい。その場合この情報は他のプレゼンス情報と同様に処理され、プレゼンス・サーバまたは加入者クライアントでの特殊な挙動を含まない。

【０２１１】

D.8.4 ClientName
内容モデル:(#PCDATA)
属性:なし
使用法:この要素はプレゼンス・クライアント・アプリケーションの名称を示す。

【０２１２】

D.8.5 ClientManufacture
内容モデル:(#PCDATA)
属性:なし
使用法：この要素はプレゼンス・クライアント製造業者の名称を示す。

【０２１３】

D.8.6 ClientVersion
内容モデル:(#PCDATA)

属性:なし
使用法:この要素はクライアント・アプリケーションのバージョンを示す。
【０２１４】
D.8.7 ClientPresenceItem
内容モデル:(ClientType、CliPresItem)
属性:Class、ID、Cacheability、ValidityPeriod、ContentChange
使用法:この要素は発行者のクライアント・アプリケーション特有のプレゼンス項目である。ClientTypeは発行者のプレゼンス・クライアントのクライアント種類である。この要素の一部として使用される場合、ClientType項目中では属性は使用されない。発行者が1つより多いクライアント特有のプレゼンス項目を規定する場合、ID属性の使用が必須である。随意選択の属性ContentChangeは0から255までのカウンタである。これは、使用される場合、プレゼンス項目の内容が変更される都度1つ増える。これは、プレゼンス・サーバに、非標準プレゼンス項目の内容の変更を検出する代替手段を提供する。プレゼンス・サーバにとって別の代替案は、この項目が同じID属性と共にサーバに送信される都度内容が変更されると想定することであろう。サーバは、ContentChange属性がクライアントによって使用されていない場合このような挙動を取る。同じID属性を有する2つの項目がContentChange中で同じカウンタ値を有する場合、これら2つの項目の内容は同一である。
【０２１５】
D.8.8 CliPresItem
内容モデル:(#PCDATA)
属性:なし
使用法:この要素はクライアント特有のプレゼンス拡張である。その構文及び構造はクライアントには既知だがネットワーク要素には知られていないと想定される。
【０２１６】
D.9 拡張プレゼンス・サービス
拡張プレゼンス・サービス・クラスは、サービス・プロバイダ特有の拡張をプレゼンス・サービスに提供するようになっている。クライアント・アプリケーション特有の拡張と同様、拡張を復号できる必要があるのはサービスの端点(すなわち、発行者と加入者クライアント)だけである。プレゼンス・サーバが拡張機能を理解することもありうるが、それはこの明細書によって義務付けられるわけではない。クライアント特有の拡張の方法とこの方法との間の相違は、サービス・プロバイダが制御するクライアントは標準プレゼンス項目を実装する必要がないことである。プレゼンス項目はサービス・プロバイダによって完全に規定されており、ユーザ装置をホストとするクライアントを新しい機能によって更新する手段が存在すると想定する。1つの可能な更新方法は無線によるソフトウェア・ダウンロードである。新しい種類のプレゼンス項目を生成する場合、サービス・プロバイダはこの新しい種類のプレゼンス項目をサポートするよう既存のクライアントを更新する必要がある。サービス・プロバイダが制御するクライアントは、標準プレゼンス・クライアント経由ではなく直接プレゼンス・サーバと通信する。標準プレゼンス・クライアントはこのクラスの全ての要素を無視する。
拡張プレゼンス・サービスは主としてこのクラスの要素を使用する。標準プレゼンス要素の規定を使用することも可能である。これは標準プレゼンス・プロトコルを使用する。
【０２１７】
D.9.1 ExtPresence
内容モデル:(#PCDATA)
属性:Class、ID、Cacheability、ValidityPeriod、ContentChange
使用法:この機構によって1つより多いプレゼンス項目が規定される場合ID属性の使用は必須である。随意選択のContentChange属性もこの要素について利用可能である。コンテンツの構文及び表示は本明細書の範囲外である。プレゼンス・サーバとプロキシによって、コンテンツはこの要素を含むXML文書と同じ多くの部分に分れたMIME中で伝えられるオブジェクトを参照できるようになる。

【０２１８】
D.10　　プレゼンス内容フォーマット用DTD
【０２１９】
【表２０】

```
<!-- Root element -->

<!ELEMENT Presence ( DeviceOn | DeviceRoaming | NetworkType |
UserStatus | PreferredContact | PreferredDefaults | LocalTime | MeasuredLoca-
tion | ConvertedLocation | StatedLocation | UserEnvironment | StatusText |
StatusImage | StatusSound | MessagingCapabilities | EmailClient | VoiceCall-
Capability | MultimediaCallCapabilities | UserPresenceItem | ClientTypeRequest
| ClientType | ClientPresenceItem | ExtPresence) >

<!ATTLIST Presence Version NMTOKENS #REQUIRED >

<!ELEMENT DeviceOn (#PCDATA) >

<!ATTLIST DeviceOn
```

【０２２０】
【表２１】

```
                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                DeviceName NMTOKEN #IMPLIED >

        <!ELEMENT DeviceRoaming (#PCDATA) >

        <!ATTLIST DeviceRoaming
                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                DeviceName NMTOKEN #IMPLIED >

        <!ELEMENT NetworkType (#PCDATA) >

        <!ATTLIST NetworkType

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                DeviceName NMTOKEN #IMPLIED >

        <!ELEMENT UserStatus (#PCDATA) >

        <!ATTLIST UserStatus

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT PreferredContact (#PCDATA) >
```

【０２２１】
【表２２】

```
<!ATTLIST PreferredContact

            Class NMTOKEN #IMPLIED
            ID PUBIDLITERAL #IMPLIED
            Cacheability NMTOKEN #IMPLIED
            ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT PreferredDefaults ((TimePeriod, PrefCont)*) >

<!ATTLIST PreferredDefaults

            Class NMTOKEN #IMPLIED
            ID PUBIDLITERAL #IMPLIED
            Cacheability NMTOKEN #IMPLIED
            ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT TimePeriod (PeriodStart, PeriodEnd,
            RepetitionPeriod?, PeriodPrecedence) >

<!ELEMENT PeriodStart (#PCDATA) >

<!ELEMENT PeriodEnd (#PCDATA) >

<!ELEMENT RepetitionPeriod (#PCDATA) >

<!ELEMENT PeriodPrecedence (#PCDATA) >

<!ELEMENT PrefCont (#PCDATA) >

<!ELEMENT LocalTime (#PCDATA) >

<!ATTLIST LocalTime

            Class NMTOKEN #IMPLIED
            ID PUBIDLITERAL #IMPLIED
            Cacheability NMTOKEN #IMPLIED
```

【０２２２】
【表２３】

```
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT MeasuredLocation (#PCDATA) >

        <!ATTLIST MeasuredLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                Accuracy NMTOKEN #IMPLIED >

        <!ELEMENT ConvertedLocation (#PCDATA) >

        <!ATTLIST ConvertedLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT StatedLocation (#PCDATA) >

        <!ATTLIST StatedLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT UserEnvironment (EnvAttributes+) >

        <!ATTLIST UserEnvironment

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
```

【０２２３】
【表２４】

```
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT EnvAttributes (#PCDATA) >

<!ELEMENT StatusText (#PCDATA) >

<!ATTLIST StatusText

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT StatusImage (#CDATA) >

<!ATTLIST StatusImage

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ImageType NMTOKEN #IMPLIED >

<!ELEMENT StatusSound (#CDATA) >

<!ATTLIST StatusSound

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                SoundType NMTOKEN #IMPLIED >

<!ELEMENT MessagingCapabilities (MessType*) >
```

【０２２４】
【表２５】

```
<!ATTLIST MessagingCapabilities

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT MessType (#PCDATA) >

<!ELEMENT EmailClient (EmailClientType*) >

<!ATTLIST EmailClientType

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT EmailClientType (#PCDATA) >

<!ELEMENT VoiceCallCapability (#PCDATA) >

<!ATTLIST VoiceCallCapability

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT MultimediaCallCapabilities (MMCap*) >

<!ATTLIST MultimediaCallCapabilities

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
```

【０２２５】
【表２６】

ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT MMCap (#PCDATA) >

<!ELEMENT UserPresenceItem (#PCDATA) >

<!ATTLIST UserPresenceItem

Class NMTOKEN #IMPLIED
ID PUBIDLITERAL #IMPLIED
Cacheability NMTOKEN #IMPLIED
ValidityPeriod NMTOKEN #IMPLIED
ExtRef PUBIDLITERAL #IMPLIED
ExtRefChange NMTOKEN #IMPLIED >

<!ELEMENT ClientTypeRequest (EMPTY) >

<!ATTLIST ClientTypeRequest

Class NMTOKEN #IMPLIED
ID PUBIDLITERAL #IMPLIED
Cacheability NMTOKEN #IMPLIED
ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT ClientType (ClientName, ClientManufacturer, ClientVersion) >

<!ATTLIST ClientType

Class NMTOKEN #IMPLIED
ID PUBIDLITERAL #IMPLIED
Cacheability NMTOKEN #IMPLIED
ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT ClientName (#PCDATA) >

【０２２６】
【表２７】

```
<!ELEMENT ClientManufacture (#PCDATA) >

<!ELEMENT ClientVersion (#PCDATA) >

<!ELEMENT ClientPresenceItem (ClientType, CliPresItem) >

<!ATTLIST ClientPresenceItem

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED
        ContentChange NMTOKEN #IMPLIED >

<!ELEMENT CliPresItem (#PCDATA) >

<!ELEMENT ExtPresence (#PCDATA) >

<!ATTLIST ExtPresence

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED
        ContentChange NMTOKEN #IMPLIED >

<!-- End of DTD -->
```

【０２２７】

D.11 PresenceNames用DTD

PresenceNames用DTDは、プレゼンス項目名と属性だけが含まれていること以外はプレゼンス用のものと同じである。プレゼンス値は使用されない。

【０２２８】

【表２８】

```
<!-- Root element -->
```

【０２２９】

【表２９】

<!ELEMENT PresenceNames (DeviceOn | DeviceRoaming | NetworkType | UserStatus | PreferredContact | PreferredDefaults | LocalTime | MeasuredLocation | ConvertedLocation | StatedLocation | UserEnvironment | StatusText | StatusImage | StatusSound | MessagingCapabilities | EmailClient | VoiceCallCapability | MultimediaCallCapabilities | UserPresenceItem | ClientTypeRequest | ClientType | ClientPresenceItem | ExtPresence) >

<!ATTLIST PresenceNames Version NMTOKENS #REQUIRED >

<!ELEMENT DeviceOn (EMPTY) >

<!ATTLIST DeviceOn
Class NMTOKEN #IMPLIED
ID PUBIDLITERAL #IMPLIED
Cacheability NMTOKEN #IMPLIED
ValidityPeriod NMTOKEN #IMPLIED
DeviceName NMTOKEN #IMPLIED >

<!ELEMENT DeviceRoaming (EMPTY) >

<!ATTLIST DeviceRoaming

Class NMTOKEN #IMPLIED
ID PUBIDLITERAL #IMPLIED
Cacheability NMTOKEN #IMPLIED
ValidityPeriod NMTOKEN #IMPLIED
DeviceName NMTOKEN #IMPLIED >

<!ELEMENT NetworkType (EMPTY) >

<!ATTLIST NetworkType

Class NMTOKEN #IMPLIED
ID PUBIDLITERAL #IMPLIED
Cacheability NMTOKEN #IMPLIED
ValidityPeriod NMTOKEN #IMPLIED

【０２３０】
【表３０】

```
                DeviceName NMTOKEN #IMPLIED >

        <!ELEMENT UserStatus (EMPTY) >

        <!ATTLIST UserStatus

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT PreferredContact (EMPTY) >

        <!ATTLIST PreferredContact

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT PreferredDefaults (EMPTY) >

        <!ATTLIST PreferredDefaults

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT LocalTime (EMPTY) >

        <!ATTLIST LocalTime

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
```

【０２３１】
【表３１】

```
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT MeasuredLocation (EMPTY) >

<!ATTLIST MeasuredLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                Accuracy NMTOKEN #IMPLIED >

<!ELEMENT ConvertedLocation (EMPTY) >

<!ATTLIST ConvertedLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT StatedLocation (EMPTY) >

<!ATTLIST StatedLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT UserEnvironment (EMPTY) >

<!ATTLIST UserEnvironment

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
```

【０２３２】
【表３２】

```
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT StatusText (EMPTY) >

<!ATTLIST StatusText

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT StatusImage (EMPTY) >

<!ATTLIST StatusImage

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ImageType NMTOKEN #IMPLIED >

<!ELEMENT StatusSound (EMPTY) >

<!ATTLIST StatusSound

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                SoundType NMTOKEN #IMPLIED >

<!ELEMENT MessagingCapabilities (EMPTY) >

<!ATTLIST MessagingCapabilities
```

【０２３３】
【表３３】

```
                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT EmailClient (EMPTY) >

<!ATTLIST EmailClientType

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT VoiceCallCapability (EMPTY) >

<!ATTLIST VoiceCallCapability

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT MultimediaCallCapabilities (EMPTY) >

<!ATTLIST MultimediaCallCapabilities

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT UserPresenceItem (EMPTY) >

<!ATTLIST UserPresenceItem
```

【０２３４】
【表３４】

```
                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ExtRef PUBIDLITERAL #IMPLIED
                ExtRefChange NMTOKEN #IMPLIED >

<!ELEMENT ClientType (EMPTY) >

<!ATTLIST ClientType

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT ClientPresenceItem (EMPTY) >

<!ATTLIST ClientPresenceItem

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ContentChange NMTOKEN #IMPLIED >

<!ELEMENT ExtPresence (EMPTY) >

<!ATTLIST ExtPresence

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ContentChange NMTOKEN #IMPLIED >
```

【０２３５】
【表３５】

<!-- End of DTD -->

【０２３６】
当業者に明らかなように、技術の進歩に伴って、本発明の概念は多くの異なった方法で実現できる。従って、本発明とその実施例は上記の例に制限されず、添付の請求項の範囲と精神において変化しうる。

【図面の簡単な説明】
【０２３７】
【図１】図1は、移動IMPSシステムを説明するブロック図を示す図である。
【図２】図2は、プレゼンス属性の送信を説明するシグナリング図を示す図である。
【図３】図3は、修飾子の使用法を説明するシグナリング図を示す図である。
【図４】図4は、本発明による、少なくとも1つのクライアント装置とサーバとを備える移動メッセージング・システムの1つの実施例を示す図である。
【図５】図5は、本発明による、少なくとも1つのクライアント装置とサーバとを備える移動メッセージング・システムの別の実施例を示す図である。
【図６】図6は、本発明によるプレゼンスの構成を示す図である。
【図７】図7は、本発明によるプレゼンス・データベースを示す図である。
【図８】図8は、本発明による未加入プレゼンスの流れ図を示す図である。
【図９】図9は、本発明によるプレゼンス情報の加入済配信を示す図である。
【図１０】図10は、本発明によるユーザ・グループとプレゼンス集合との管理を示す図である。

【図２】

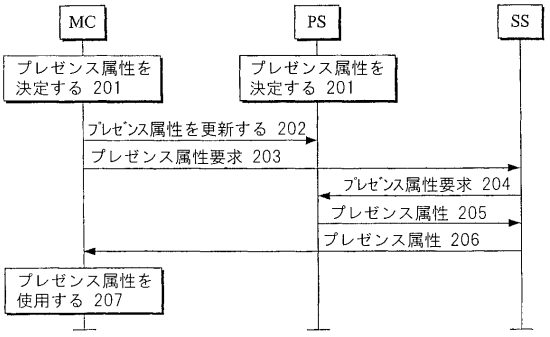


FIG. 2

【図３】

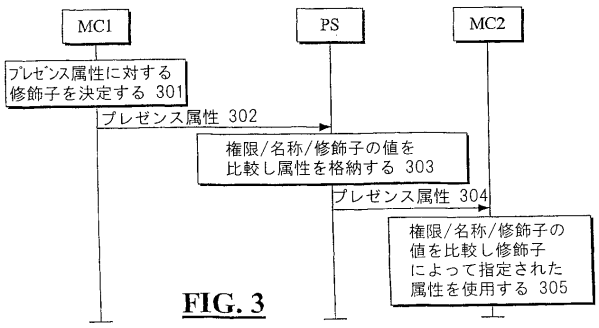


FIG. 3

【図４】

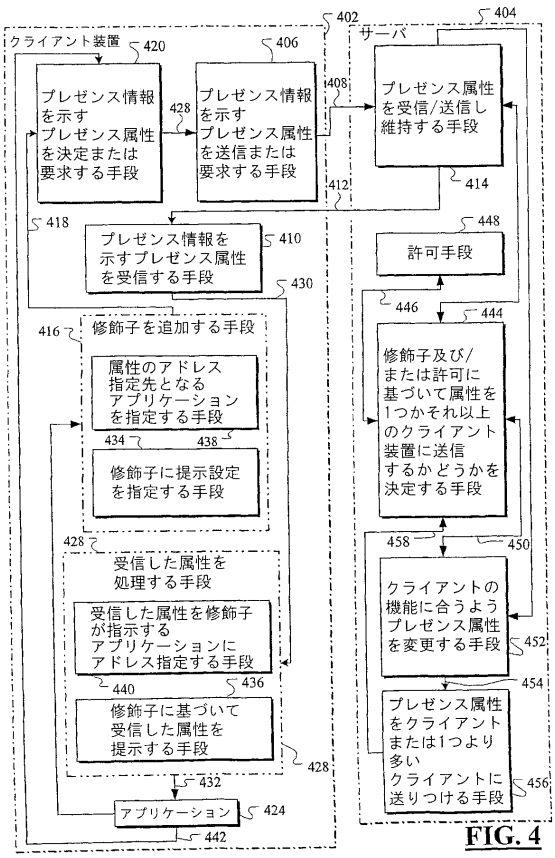


FIG. 4

【図５】

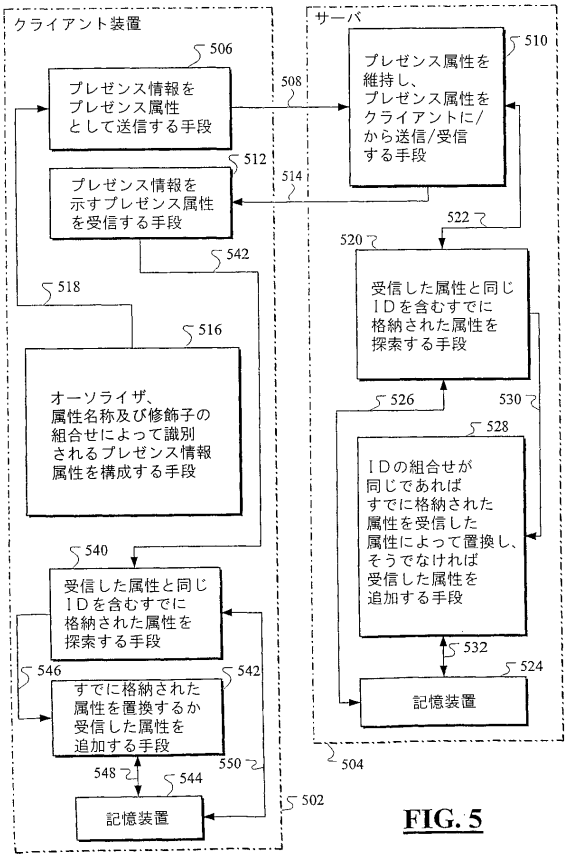


FIG. 5

【図６】

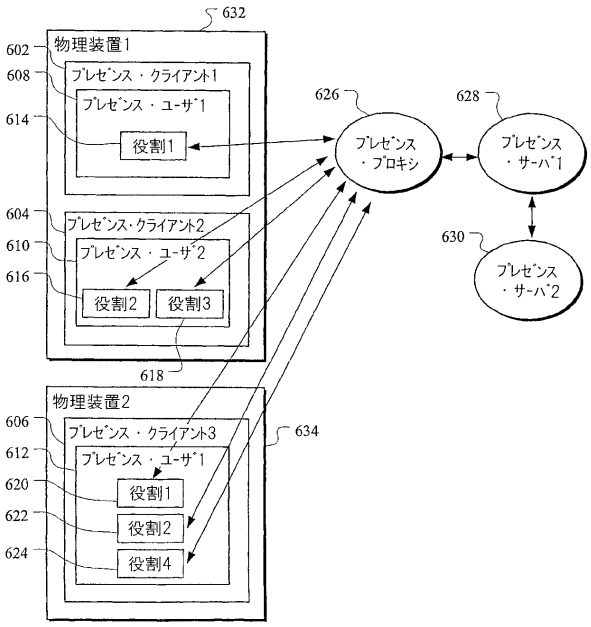


FIG. 6

【図７】

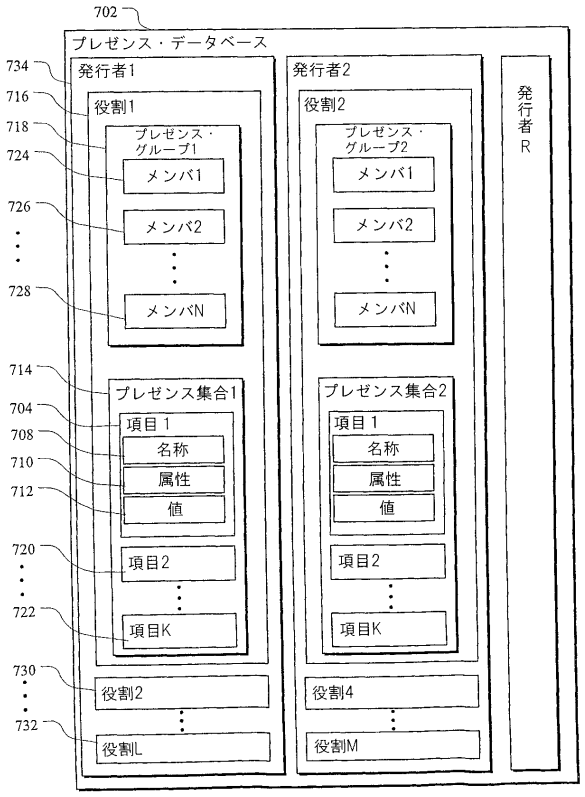


FIG. 7

【図８】

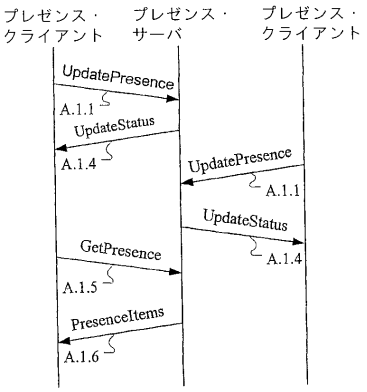


FIG. 8

【図９】

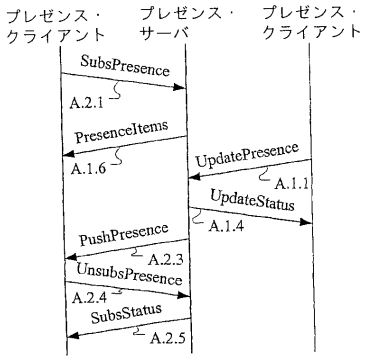


FIG. 9

【図１０】

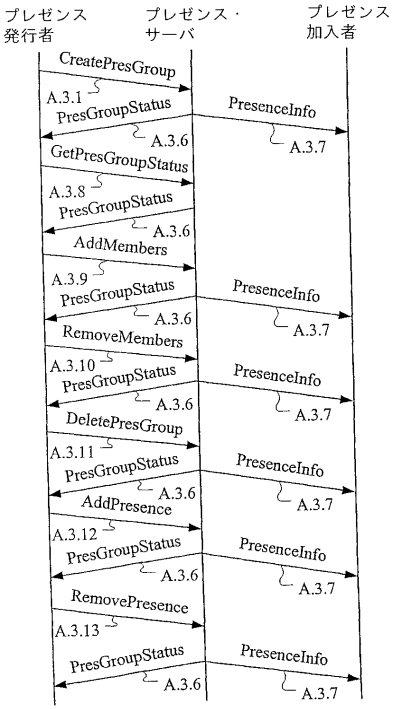


FIG. 10

【国際公開パンフレット】

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
21 November 2002 (21.11.2002)

PCT

(10) International Publication Number
WO 02/093959 A1

(51) International Patent Classification[7]: H04Q 7/22, 7/38

(21) International Application Number: PCT/FI02/00403

(22) International Filing Date: 10 May 2002 (10.05.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/290,123 11 May 2001 (11.05.2001) US
20012158 7 November 2001 (07.11.2001) FI

(71) Applicant *(for all designated States except US)*: NOKIA CORPORATION [FI/FI]; Keilalahdentie 4, FIN-02150 Espoo (FI).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: SALOMÄKI, Ari [FI/FI]; Ojavalnionkatu 10 D 28, FIN-33710 Tampere (FI).

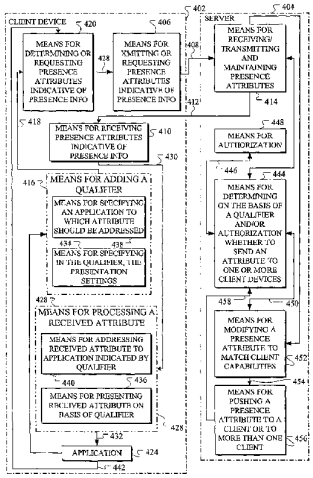
(74) Agent: KOLSTER OY AB; Iso Roobertinkatu 23, P.O.Box 148, FIN-00121 Helsinki (FI).

(81) Designated States *(national)*: AE, AG, AL, AM, AT (utility model), AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ (utility model), CZ, DE (utility model), DE, DK (utility model), DK, DM, DZ, EC, EE (utility model), EE, ES, FI (utility model), FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK (utility model), SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent

[Continued on next page]

(54) Title: MOBILE INSTANT MESSAGING AND PRESENCE SERVICE



(57) Abstract: The invention relates to mobile messaging and presence services. According to one aspect of the invention, a client device of the mobile messaging system adds a qualifier to a presence attribute, the qualifier comprising one or more parameters specifying the use of the attribute. A client device receiving a presence attribute processes the received presence attribute according to the qualifier parameters in said received attribute. Another aspect of the invention is the showing of how to assemble and store presence items with names, attributes and values in a single presence set within a role having an associated authorization group of members that have the right to subscribe to the whole or part of the presence set of the same role.

WO 02/093959 A1

WO 02/093959 A1

(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declaration under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZM, ZW, ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)*

Published:

— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Mobile Instant Messaging and Presence Service

Background of the Invention

The present invention relates to messaging in mobile telecommunication systems and more particularly to presence attributes in a mobile instant messaging and presence service.

An instant messaging service provides the end users with a means for fast, interactive, mainly text-based communication. The utility of instant messaging is greatly enhanced by the addition of a service that will keep track of the online status and availability of your chat partners or "friends"; as well as notify you of changes to their status or availability. This type of service is called a "presence service". In general, *presence* can be considered containing various dynamic information on a user or client connected to the instant messaging service via various means. Examples of this information is reachability, availability and location of the user for communication. The combination of instant messaging and presence services is called an instant messaging and presence service (IMPS). This kind of service has been available for wireline Internet users but the interconnectivity between wireline users and mobile users has been missing.

Wireless Village initiative has been established to define specifications for mobile instant messaging and presence service. The Wireless Village Instant Messaging and Presence Service (IMPS) includes four primary features: presence, instant messaging, groups and shared content. Shared content allows users and operators to setup their own storage area where they can post pictures, music and other multimedia content while enabling the sharing with other individuals and groups in an IM or chat session. The Wireless Village initiative enables both operators and end-users to create and manage groups. Presence is the key enabling technology for the Wireless Village initiative. In the existing Internet-based instant messaging service, the presence values are usually very simple, such as user is active, absent, not willing to communicate etc. These values are selected from the predefined sets of values. A white paper has been published on the Wireless Village mobile IMPS solution: "*Wireless Village, The Mobile IMPS Initiative:White Paper*", dated on 26th April 2001. The existing mobile terminal can be considered a personal tool which reflects the personal status more accurately than a desktop computer. Considering the wide range of information that may be obtained from the user and the mobile terminal, the anticipation of the presence information domain is very difficult. Thus a mechanism

should be developed to enable easy usage and addition of new types of presence information.

Disclosure of Invention

An object of the invention is to provide a solution for creation of new presence attributes besides already determined presence attributes.

Another object of the invention is to show how to organize and store presence attributes for use by clients.

According to a first aspect of the invention, a client device of a mobile messaging system adds a qualifier to a presence attribute, the qualifier comprising one or more parameters specifying the use of the attribute. A client device receiving a presence attribute processes the received presence attribute according to the qualifier parameters in the received attribute. A presence attribute is a collection of data describing presence information on a certain user and/or a client device, the presence information intended for other users. A presence attribute may also contain information for machine-to-machine communication between the client devices.

In further accord with the first aspect of the present invention, a mobile messaging system comprising at least one client device and a server, wherein the client device comprises means for transmitting presence information as presence attributes to the server and means for receiving presence attributes from the server, the presence information being categorized by a plurality of presence attribute types identified by attribute name, and the server comprises means for maintaining presence information based on the received presence attributes, is characterized in that the client device comprises means for adding a qualifier to a presence attribute, the qualifier comprising one or more parameters specifying the use of the attribute, and the client device comprises means for processing a received presence attribute according to the qualifier parameters in the received attribute.

According to a second aspect of the invention, the client device composes presence information attributes that are identified by a combination of an authorizer, an attribute name and a qualifier, wherein the authorizer specifies the body responsible for maintaining the attribute and the qualifier specifies the use of the attribute. When receiving a presence attribute, the server and the client device search for already stored attributes containing same identifiers as the received attribute. An attribute already stored is replaced with the received attribute if the combination of identifiers of the received attribute is identical to that

of the already stored attribute. Otherwise the received attribute is added without replacing any previous attribute.

In further accord with the second aspect of the present invention, a mobile messaging system comprising at least one client device and a server, wherein the client device comprises means for transmitting presence information as presence attributes to the server and means for receiving presence attributes from the server, the presence information being categorized by a plurality of presence attribute types identified by attribute name, and the server comprises means for maintaining presence information based on received presence attributes, is characterized in that the client device comprises means for composing a presence information attribute identified by a combination of an authorizer, an attribute name and a qualifier, the authorizer specifying the body responsible for maintaining the attribute and the qualifier specifying the use of the attribute, the server comprises means for searching for an already stored attribute containing same identifiers as a received attribute and means for replacing the already stored attribute with the received attribute if the combination of identifiers of the received attribute is identical to that of the already stored attribute or otherwise adding the received attribute, and the client device comprises means searching for an already stored attribute containing same identifiers as a received attribute and means for replacing the already stored attribute with the received attribute if the combination of identifiers of the received attribute is identical to that of the already stored attribute or otherwise adding the received attribute.

In further accord with the first and second aspects of the present invention, the system is characterized in that the client device comprises means for specifying in the qualifier the presentation settings of the attribute, and the client device comprises means for presenting the received attribute on the basis of the qualifier.

In still further accord with the first and second aspects of the present invention, the system is characterized in that the client device comprises means for specifying in the qualifier the application to which the attribute should be addressed, and the client device comprises means for addressing the received attribute to the application indicated by the qualifier.

Still further in accord with the first and second aspects of the invention, the system is characterized in that the server comprises means for determining on the basis of the qualifier whether to send the attribute to one or more client devices.

According further to the first and second aspects of the invention, the system is characterised in that the presence attributes received from the client device by the server are stored in a database according to a publisher user in association with a presence group.

In accordance still further to the first and second aspects of the invention, the system is characterized in that each presence attribute is part of an item including an attribute name element and an attribute value. The name element may include an authority string indicative of an authority responsible for keeping the name element and attribute value unique.

According still further to the first and second aspects of the present invention, the system is characterized in that a presence set comprises one or more presence attributes belonging to a single publisher role of a publisher user in association with a single presence group.

In still further accord with the first and second aspects of present invention, a mobile messaging system is characterized in that a user of the client device as a publisher is able to use the client device or more than one client device in more than one publisher role.

According to a third aspect of the present invention, a mobile client device for mobile messaging system, the client device comprising means for transmitting presence information as presence attributes to a server, the presence information being categorized by a plurality of presence attribute types identified by attribute name, is characterized in that the client device further comprises means for adding a qualifier to a presence attribute, the qualifier comprising one or more parameters specifying the use of the attribute.

According to a fourth aspect of the present invention, a mobile client device for mobile messaging system, the client device comprising means for receiving presence attributes from a server, the presence information being categorized by a plurality of presence attribute types identified by attribute name is characterized in that the client device further comprises means for adding a qualifier to a presence attribute, the qualifier comprising one or more parameters specifying the use of the attribute, and means for processing a received presence attribute according to the qualifier parameters in the received attribute.

According to a fifth aspect of the present invention, a mobile client device for mobile messaging system, the client device comprising means for transmitting presence information as presence attributes to the server, and means for receiving presence attributes from the server, the presence informa-

tion being categorized by a plurality of presence attribute types identified by attribute name, is characterized in that the client device further comprises means for composing a presence information attribute identified by a combination of an authorizer, an attribute name and a qualifier, the authorizer specifying the body responsible for maintaining the attribute and the qualifier specifying the use of the attribute, means for searching for an already stored attribute containing same identifiers as a received attribute, and means for replacing the already stored attribute with the received attribute if the combination of identifiers of the received attribute is identical to that of the already stored attribute or otherwise adding the received attribute.

According further to the third, fourth and fifth aspects of the present invention, a mobile client device is characterized in that each presence attribute is part of an item including an attribute name element and an attribute value. The name element may include an authority string indicative of an authority responsible for keeping the name element and attribute value unique.

In further accord with the third, fourth and fifth aspects of the present invention, the mobile client device is characterized in that a presence set comprises one or more presence attributes belonging to a single publisher role of a publisher user in association with a single presence group.

In still further accord with the third, fourth and fifth aspects of the present invention, the mobile client device is characterized in that a user of the mobile client device as a publisher is able to use the client device or more than one client device in more than one publisher role.

According to a sixth aspect of the present invention, a server for a mobile messaging system, the server comprising means for maintaining presence information based on received presence attributes, the presence information being categorized by a plurality of presence attribute types identified by attribute name, is characterized in that the server further comprises means for searching for an already stored attribute containing same identifiers as a received attribute, and means for replacing the already stored attribute with the received attribute if the combination of identifiers of the received attribute is identical to that of the already stored attribute or otherwise adding the received attribute.

In further accord with the sixth aspect of the present invention, the server is characterized in that the presence attributes received from a client de-

vice by the server is stored in a database according to a publisher user in association with a presence group.

In still further accord with the sixth aspect of the present invention, the server is characterized in that each presence attribute is part of an item including an attribute name element and an attribute value. The name element may include an authority string indicative of an authority responsible for keeping the name element and attribute value unique.

Still further in accord with the sixth aspect of the present invention, the server is characterized in that a presence set comprises one or more presence attributes belonging to a single publisher role of a publisher user in association with a single presence group. A user of a client device in communication with the server acting as a publisher is able to use the client device or more than one client device in more than one publisher role.

According to a seventh aspect of the present invention, a presence system, comprises at least one physical device having at least one presence client for enabling a presence user to interact with the system as a publisher or a subscriber, and a server for maintaining valid values of presence sets of attributes of a publisher for access by subscribers according to associated presence groups.

In further accord with the seventh aspect of the present invention, a presence system is characterized in that each attribute is part of an item including an attribute name element and an attribute value. The name element may include a qualifier having information related to attribute usage. The name element may include an authority string indicative of an authority responsible for keeping the name element and attribute value unique.

In still further accord with the seventh aspect of the present invention, the presence system is characterized in that a presence set comprises one or more attributes belonging to a single publisher role in association with a single presence group. The user interacting with the system as a publisher is able to use the presence client or more than one presence client in more than one publisher role. Each attribute may be part of an item including an attribute name element an attribute value.

Still further in accord with the seventh aspect of the present invention, the at least one physical device is a mobile physical device.

According to an eighth aspect of the present invention, a computer program embodied in a computer-readable medium for storage in a physical de-

vice, is characterized in that the program is a presence client program for enabling a presence user to interact with a presence system as a publisher of at least one presence set of attributes for access by one or more subscribers according to an associated at least one presence group.

In further accord with the eighth aspect of the present invention, the presence client program is characterized in that the program enables a presence user to interact with the presence system as a subscriber to at least one set of attributes associated with a presence group in which the subscriber is a member.

In still further accord with the eighth aspect of the present invention, the presence client program is characterized in that each attribute is part of an item including an attribute name element and an attribute value. The name element may include a qualifier having information related to attribute usage.

According further to the eighth aspect of the present invention, the presence client program is characterized in that the name element includes an authority string indicative of an authority responsible for keeping the name element and attribute value unique.

In accordance still further with the eighth aspect of the present invention, the presence client program is characterized in that a presence set comprises one or more attributes belonging to a single publisher role in association with a single presence group. A user interacting with the system as a publisher is able to use the presence client program or more than one presence client program in more than one publisher role. Each attribute may part of an item including an attribute name element and an attribute value.

Still further in accord with the eighth aspect of the present invention, the presence client program is characterized in that the physical device is a mobile physical device.

According to a ninth aspect of the present invention, a physical device having a computer program embodied in a computer-readable medium stored therein, is characterized in that the program is a presence client program according to the eighth aspect of the present invention.

According to a tenth aspect of the present invention, a data structure embodied in a computer-readable medium for storage in a physical device, is characterized in that the data structure is a presence database for storing valid values of presence sets of attributes of one or more publishers for access by subscribers according to presence groups associated with the presence sets.

In further accord with the tenth aspect of the present invention, a data structure is characterized in that each attribute is part of an item including an attribute name element and an attribute value. The name element may include a qualifier having information related to attribute usage. The name element may include an authority string indicative of an authority responsible for keeping the name element and attribute value unique.

In still further accord with the tenth aspect of the present invention, a presence set comprises one or more attributes belonging to a single publisher role in association with a single presence group.

Further still in accord with the tenth aspect of the present invention, the data structure is characterized in that a user interacting with the physical device as a publisher is able to use the data structure to publish more than one publisher role, each role having distinct presence sets in association with distinct presence groups.

In accordance still further with the tenth aspect of the present invention, a data structure is characterized in that the physical device is a mobile device.

According to an eleventh aspect of the present invention, a physical device having a data structure embodied in a computer-readable medium stored therein, is characterized in that the data structure is a presence database according to the tenth aspect of the present invention.

Advantages of the invention are that it is possible to adjust predetermined presence attributes by adding a qualifier. The qualifier may be used to add a new attribute (as an attribute with a qualifier can be uniquely identified, i.e. functionally separated from an attribute with same attribute name but without qualifier). Thus e.g. users or application developers can easily determine new presence attributes best fitting their needs or describing their current actual status without being limited to the predetermined types. The use of the qualifier brings another advantage as it enables the sender of presence information (the owner) to further specify how the presence information is to be used in the receiving client device.

In one embodiment of the invention the application to which the attribute should be addressed is specified in the qualifier. The received attribute is addressed to the application indicated in the qualifier. The further advantage of this embodiment is that the sending client device may define the used applica-

tion and thereby use qualifier to direct certain presence information to a certain application.

Another advantage of the invention is the showing of how to assemble presence items with names, attributes and values in a single presence set within a role having an associated authorization group of members that have the right to subscribe to the whole or part of the presence set of the same role.

Brief Description of the Drawings

In the following, the invention will be described in further detail by means of preferred embodiments and with reference to the accompanying drawings, in which

Figure 1 is a block diagram illustrating a mobile IMPS system;

Figure 2 is a signalling diagram illustrating the transmission of presence attributes; and

Figure 3 is a signalling diagram illustrating the usage of a qualifier.

Figure 4 shows one embodiment of a mobile messaging system comprising at least one client device and a server, according to the present invention.

Figure 5 shows another embodiment of a mobile messaging system comprising at least one client device and a server, according to the present invention.

Figure 6 shows a presence framework, according to the present invention.

Figure 7 shows a presence database, according to the present invention.

Figure 8 shows flow diagrams of unsubscribed presence, according to the present invention.

Figure 9 shows subscribed delivery of presence information, according to the present invention.

Figure 10 shows management of user groups and presence sets, according to the present invention.

Best Mode for Carrying out the Invention

Figure 1 illustrates a mobile IMPS system. A number of mobile clients (MC) can be connected via a mobile network MNW and possibly one or more intermediary networks ONW to an IMPS server S. Typically the Internet is used as

the intermediary network, also non-mobile clients C may be served by the IMPS system. The IMPS server S can be, as regards presence services, functionally divided into server elements: A publisher server PS that is the home service element for a publishing client owning the presence information and a subscriber server SS that is the home for a subscribing or requesting client. Thus the client MC is served by both servers; the MC updates its presence information to the PS and acts as a publishing client and on the other hand requests and receives presence information relating to other clients as presence attributes from the subscriber server SS. The server PS maintains presence data and manages its distribution based on the users' publishing preferences relating to presence information. The functions of SS and PS may be carried out on one physical server device or distributed to a plurality of server devices.

It should be noted that the present description focuses on presence related service capabilities. Other important mobile IMPS service capabilities are messaging capabilities, user group management capabilities, content management capabilities, subscriber management capabilities, and client capabilities. Mobile IMPS services are created by using these service capabilities. For instance, a client MC may belong to several user groups and the server S manages the group memberships, handles instant messaging and delivers presence information between the members of the group. One important role of the server is also to control information flow; the server may have filters on the basis of user preferences defining e.g. what presence or other information can be delivered to members of a group 'Friends', to members of a group 'Work colleagues' or publicly to any client.

Various transport layer protocols may be used and the IP protocol is typically used to provide a network layer service. Different lower layer transmission protocols may be used. The mobile network MNW can be any wireless network, such as a cellular network supporting the GSM service, a network supporting also the GPRS service (General Packet Radio Service), a third-generation mobile communication network, such as a UMTS network (Universal Mobile Telecommunications System), a wireless local area network WLAN or a private network. Also short range infrared or radio connections, such as a Bluetooth™ communication, may be used as a part of the communications path between the MC and the server S. The mobile client device MC may be e.g. a mobile station, a PDA device or a laptop computer comprising or being connected to a wireless modem. The mobile IMPS messages may be transferred using a

circuit-switched data call, a packet-switched data transmission context, messaging services such as SMS or MMS (Multimedia Messaging Service), for-instance. Figure 2 illustrates the usage of presence attributes. As the mobile client MC has determined 201 one or more presence attributes to be published, it updates 202 presence attributes to the publisher server PS, i.e. publishes one or more presence attributes. The determination 201 of presence attributes can be done when the client is establishing a logical mobile IMPS session with the mobile IMPS server or automatically or by a user initiative when some presence information has changed. For instance, phase 201 may be initiated automatically at a pre-determined time, data, or by a change of user profile at the mobile client MC. When a client MC (A) accesses mobile IMPS services of a server SS, it may request 203 presence information on another client (B). The subscriber server SS requests 204 this information from the publisher server PS (of client A). The PS sends 205 one or more presence attributes to SS, if this is allowed by the publication preferences (of client B). It is possible that the publication preferences set by client B prevent some part of the requested information from being sent (to client A or in general). The SS may also automatically request 204 presence information on the basis of the user preferences (of A) when the client establishes a logical connection with the service of the SS. The SS forwards 206 the presence attributes to the receiving client MC (A).

The subscriber server SS (and the publication server PS) typically send the client-originated presence attributes towards the client unmodified. However, there may exist a content adaptation mechanism implemented in the server PS. Content adaptation addresses the issue of modifying a presence attribute in such a way that it matches the client capabilities of the receiving client. In addition to the transmission of presence information as a response to a request from a client, it is also possible to push presence information to available clients MC (that are logically connected to the service) according to the publication preferences. The push-type presence notification can be triggered by three mechanisms: when the publisher server receives an update from the publisher client, when the publisher server detects a change in the attribute value or by implementation-specific internal triggers updating the value.

The client MC is thus configured to update one or more presence attributes to the PS, to receive and handle 207 the presence attributes received from the SS and to present presence information obtained from at least one presence attribute to the user. The MC preferably stores presence information

(presence attribute values) until new presence attribute values are received in an update message used to carry presence attributes (or the client finishes the mobile IMPS session). Further, as will be illustrated in more detail later, the client device may automatically utilize received presence information to adjust its function accordingly. In addition to the signallings shown in Figure 2, an authorization may be requested from the publisher of the presence information before sending presence attributes to a requesting client. Figure 2 does not show any status messages by which the server may respond, e.g. after the message 202.

A *client originated* presence attribute is one which has its value field filled in by the publishing client. A *server originated* presence attribute is one which has its value field filled in by the publisher server. A presence attribute is *client-server originated* when one part of the value field is filled in by the client and the rest by the publisher server. According to a preferred embodiment of the invention, users or organizations may define new presence attributes besides the predetermined set of attributes. The presence attributes can be divided into the following classes:

- Client Status: Presence attributes describing the availability of the client device for communication; network reachability, GPRS attached, on/off status, operator, for instance. Thus the attributes in mobile IMPS service are very different than in IMPS used for non-mobile client devices.
- User Status: Presence attributes describing the availability of the user for communication; ready, meeting, busy, away, on the phone, chatting, do not disturb, for instance.
- Local Information: Presence attributes describing the local environment at the user; local time, noisy/silent environment, in-door, outdoor, location of the user, in terms of, for instance, geographical location, visited PLMN, city/street, premises, for instance. For instance, the exact location of the mobile client may be obtained for the local information attribute directly and the availability status (in a meeting, in a summer cottage, etc) may be readily available via user profile settings of the mobile client MC.
- Personal Status: Various personal attributes describing personal user status; mood, personal interests and intentions, for instance.

- Client Capabilities: Presence attributes describing the capabilities of the client device to support different means of communication, different media types and different features.
- User Attributes: Presence attributes allowing the client device or the user to define their own textual presence values and references to external values.
- Extended Presence Information: Vendor specific or service provider dynamically defined non-standard presence attributes which however need to be passed through standard presence servers.

Thus there may be different presence attributes for mobile client devices and actual users. For instance, the user may be defined as not being available for receiving messages but the user's client device is defined as being on-line. The user may also be defined to be able to receive messages and not being online if SMS is used as a bearer.

GENERAL STRUCTURE AND IDENTIFICATION OF PRESENCE ATTRIBUTES

Table 1 describes the general structure of presence attributes. A presence attribute generally comprises an identifier part and a plurality of value fields. The Req field determines whether the element is mandatory M, optional O or conditional C. Attribute information is in XML format (Extensible Markup Language).

Information Element	Req	Type	Description
NameSpec	M	XML	Attribute identity information
Value	M	XML	Value for the attribute

Table 1. Presence Attribute Structure

The sub-elements of the Namespec element are described in Table 2.

NameSpec	Req	Type	Description
Name	M	String	Name of the attribute
Qualifier	O	String	Information related to the scope of attribute usage
Authority	C	String	The authority responsible for the fact that attribute NameSpec and value field names are unique

Table 2. The NameSpec element

The name of the attribute is a string given by the information element 'Name'. The Name information element is defined for all presence attributes in the format defined in Table 3.

Information element	Name
Data type	String
Format	Free text format
Description	Name of a presence attribute

Table 3. The Name element

The format of the Qualifier element is illustrated in Table 4.

Information element	Qualifier
Data type	String
Format	Free text format
Description	Modifies the scope of attribute usage

Table 4. The Qualifier element

The Qualifier element is used to specify the scope of the attribute usage. The qualifier can be used especially for two purposes: to add a new attribute or to enable the sender of presence information (the publisher) to specify how the presence information is to be used in the receiving client device. Thus the qualifier string may be used as a parameter for one or more applications in the receiving client device.

For instance, if the publisher wants to limit the knowledge of his exact location (e.g. the street address) to only some of the users (call it group A) and give a more approximate location (e.g. only a city name) to others (call it group B), he may publish a location attribute with the city name to group B. For group A he attaches a Qualifier (say 'My best friends') to the location attribute. This effectively creates a new attribute with Qualifier 'My best friends'. He then includes the street address in this new attribute and publishes it to group A. As these attributes are different, the server PS is able to keep their values separate. Also if a person belongs to both groups A and B, the client device of this person can be configured to distinguish between these two attributes. The client device may be configured to present (and possibly utilize) the attribute according to the group which the user has activated. The possible Qualifier values may be preassigned by the client and the service provider or they may be assigned dynamically by the user (publisher). A service provider may also limit the number of dynamically assigned Qualifier values.

The authority element determines the body who is responsible for keeping the attribute and its contents unique. This relates to the attribute extension mechanism.

Information element	Authority
Data type	String
Format	URL
Description	Identifies the body responsible for the uniqueness of the attribute

Table 5. The Authority element

When predetermined presence attributes are used unmodified, the Authority string can be omitted. Use of a Qualifier string is not a modification to the attribute that requires the use of the Authority element. It must be used when introducing a new attribute (a new Name) or when adding a new value field to an already specified attribute. In both of these cases the attribute is considered to be a new one and the Authority is responsible for maintaining this new attribute.

Generally a Value field name must be unique within an attribute. Thus, the introduction of a new value field to an existing attribute must be handled by the rules set by the body who has defined the attribute. Adding a new value field transforms the old attribute into a new one. This must be signalled by the presence of the Authority field in order to allow both the new and the old attribute to co-exist. It is also possible that a stakeholder releases an attribute for public maintenance. This kind of attribute is registered by a proper authority, such as IANA (Internet Assigned Numbers Authority), and also signaled in the Authority field. In this case any stakeholder may register additional value fields for the attribute without having to change the Authority field. The server (PS, SS) does not remove a value field from an attribute even though it would not understand the semantics of the value field. The client MC ignores all the value fields in the attribute that it does not understand.

An attribute is identified and made unique by the element NameSpec. Figure 3 shows an example of the use of the NameSpec element and the qualifier. A qualifier is determined 301 for an attribute in client device MC1. It is possible that the user determines the qualifier or that the client device MC deter-

mines it. The qualifier may be defined in order to specify the desired the user group, determine how to present the presence information in a receiving client device (MC2), or otherwise to specify how the receiving client device MC2 should use the attribute. It is possible to utilize the user profiles of the mobile client MC when determining the qualifier. For instance, the MC1 composes presence attribute on the basis of the current profile (e.g. on a meeting), calendar entries (meeting ends at 12.00) and the local time. By using the qualifier, the presence attribute can be easily modified to include a lot of useful information for the receiving client MC2. The attribute is sent 302 to the server PS.

PS compares 303 the NameSpec element of the received attribute to already stored attributes. The PS first compares the Authority strings with each other. An attribute not containing an Authority string is different from any attribute that has an Authority string. Next the attribute names are compared. Finally the Qualifier strings are compared. An attribute that does not contain a Qualifier string is different from any attribute that has a Qualifier string. Two attributes are the same only when all of these three comparisons give the same result. Thus it is possible to functionally separate the received attribute with the qualifier from attributes with same attribute name but with a different qualifier.

The publication server PS does this kind of comparison to determine whether the value fields of the received attribute shall replace some already existing presence information or whether the attribute is a new one to be added to the presence information storage of the MC2. On the basis of the comparison, the PS stores 303 the information in the received attribute. The PS replaces previous presence information of an already stored attribute with the presence information of said received attribute if all identifiers of said received attribute are the same as in already stored attribute. Otherwise the PS adds the presence information of said received attribute without replacing any previous information. PS sends 304 the attribute to at least one client device MC2 (either automatically pushes it or as a response to a request from the MC2). According to an embodiment, the qualifier determines a group to which the attribute is directed. Further, the qualifier may be used to present presence information in different ways for private contacts and public contacts in a phone book, for instance. Thus the PS may determine the receiving client devices on the basis of the qualifier. Service capabilities for a dynamic phone book service will be described later, after first describing the client status attributes, user status attributes and personal status attributes more fully below.

Also the receiving client MC2 decides, after a similar comparison 305 as in PS, whether and how to store the information of the received presence attribute. This kind of ternary identification of presence attributes enables very flexible usage, management and creation of presence attributes.

There are many ways in which the qualifier may be used to specify the usage of the attribute in the client device MC2. Typically the client device MC is capable of supporting a plurality of applications. According to a first embodiment, the client device MC1 adds a qualifier specifying an application to be used. Application refers generally to any application entity which can be identified e.g. by a port number. The application may be the same as used to process the presence information of the attribute in the client device MC1 or another application. The receiving client device MC2 addresses 305 the received attribute to the application indicated by the qualifier. For instance, by using the qualifier, the same presence information may be sent to a phone book application and a game application. These applications may use the presence information differently and thus it is possible also to tailor attributes exactly for application needs.

According to a second embodiment, the client device MC1 adds a qualifier specifying the presentation of the attribute. The receiving client device MC2 presents 305 the received attribute on the basis of the qualifier. The qualifier may determine e.g. whether the information is shown to the user at all or not or which parts of the information are shown. The qualifier may also determine various user interface settings such as colors, fonts etc. Thus the UI of the MC2 is configured 305 on the basis of the settings in the qualifier.

CLIENT STATUS ATTRIBUTES

According to a preferred embodiment, the mobile clients MC utilize a presence attribute which describes the current transmission capabilities of a mobile client MC. A structure for this kind of attribute is illustrated in Table 6. This attribute, referred to as a Modem attribute, gives presence information on those user terminal parts or functions that are dealing with mobile bearers.

Information Element	Req	Type	Value	Description
Name Spec. Name	M	String	Modem	Name of the attribute
Value	M	XML	See Table 7 below	Value for the attribute. Type is a structure

Table 6. Modem Attribute Structure

Modem.Value	Req	Single/Multiple	Description
Status	M	S	Name of the Value field
CommAddr	O	S	Name of the Value field
CS_Status	O	S	Name of the Value field
PS_Status	O	S	Name of the Value field
Roaming-Status	O	S	Name of the Value field
CS_CallStatus	O	M	Name of the Value field
PDP_ContexStatus	O	M	Name of the Value field

Table 7. Value fields of Modem attribute

The Status value field as illustrated in Table 8 indicates the status of the mobile modem.

Information element	Status
Data type	Enumerated String
Format	Following values: ON – The modem part of the terminal is powered on OFF – The modem part of the terminal is powered off or out of coverage DIS – The value fields (if any) given by this attribute are invalid. All the value fields obtained from earlier updates are invalid
Description	Status field of the modem attribute
Range	ON | OFF | DIS

Table 8. Status value field of Modem

The Status value field indicates whether the modem is turned on or off. According to a preferred embodiment, a presence attribute includes a DIS indication preferably in the mandatory Status value field of the attribute. When DIS is set in an attribute, all values given in the attribute are invalid. The receiving mobile client MC is thus able to ignore the value fields of the attribute. Also previous values of the presence attribute are removed (and nulled). Thus it is practical to send an attribute with DIS indication but no other value fields. This kind of attribute requires very little space and thus critical bandwidth over the radio interface can be saved. This is very useful especially with attributes describing user attributes.

Referring again to Table 7, the CommAddr value field of the Modem attribute includes the communication address of the modem (MC). It contains two parts: the communication means and the contact address. The means part carries information about the supported communication methods, especially whether the modem supports packet-switched (PS) data, circuit-switched (CS) data or voice, SMS or MMS. The Contact part includes the address, e.g. an MSISDN number.

The CS_Status value field indicates the circuit switched status of the modem (registered or not registered). The PS_Status value field indicates the

packet switched status of the modem (attached or not attached). The RoamingStatus value field indicates the home PLMN (Public Land Mobile Network) and possibly the PLMN in which the modem is currently roaming. The CS_CallStatus value field gives the in-call status of a CS bearer (data or voice; active or not active). The modem attribute may have a list of these in-call statuses in case if multicall capability is supported by the modem. The PDP_Context Status value field includes information about the PDP (Packet Data Protocol) context, such as QoS (Quality of Service) information.

In addition to above examples, the modem value field may be used to carry other information related to transmission capabilities of the mobile client. In a first example a maximum bit rate of the mobile client is delivered in the Mode attribute. The receiving client device can then configure its transmission rate so that the maximum bit rate is not exceeded. In a second example the client device determines that only the packet-switched transmission mode is to be used when sending data files to the client device. A third example is that a roaming device orders that only certain type of communication is enabled (e.g. only voice calls are allowed and data files shall not be sent).

USER STATUS ATTRIBUTES

According to one preferred embodiment, an attribute is defined for the willingness of the user to engage in an activity. The activity is specified by the value fields belonging to this Availability attribute. Table 9 illustrates the structure for the Availability attribute.

Information Element	Req	Type	Value	Description
Name Spec. Name	M	String	Availability	Name of the attribute
Value	M	XML	See Table 10 below	Value for the attribute.

Table 9. Availability Attribute Structure

Table 10 describes the value fields of the Availability attribute.

Availability-Extension-Value	Req	Single/ Multiple	Description
Status	M	S	Value field name
CommAvail	O	S	Value field name
PhoneAvail	O	S	Value field name
SMSAvail	O	S	Value field name
MMSAvail	O	S	Value field name
IMAvail	O	S	Value field name
EmailAvail	O	S	Value field name
Image	O	M	Value field name
Text	O	M	Value field name

Table 10. Availability attribute values

The status value field as illustrated in Table 11 indicates the status of availability information.

Information element	Status
Data type	An enumerated String
Format	One of the following values: ENA – The value fields included in this attribute contain up-to-date information. The value fields updated earlier and not included in this attribute are still up-to-date DIS – The value fields (if any) included in this attribute contain invalid information. The value fields updated earlier are invalid.
Description	Defines the publishing status of availability attribute
Range	ENA | DIS

Table 11. Status field

The Status value field indicates whether the publishing of availability information is enabled or not. The DIS indication may be used as already described to invalidate the values of the Availability attribute. For instance, the server PS may send an availability attribute with DIS indication after the mobile client MC has closed the mobile IMPS session. This kind of message may also be sent when a connection to client is suddenly lost. Thus the receiving mobile client can remove all availability information relating to the user and the client device that are no longer present in the mobile IMPS system.

The CommAvail value field in Table 10 indicates whether the user is willing to engage into any form of remote communication. The PhoneAvail value field indicates whether the user is willing to engage in a telephone call. The SMSAvail value field indicates whether the user is willing to engage in an SMS exchange. The MMSAvail value field indicates whether the user is willing to engage in an MMS (Multimedia Messaging Service) exchange. IMAvail value field indicates whether the user is willing to engage in an IM (Instant Messaging) exchange. The EmailAvail value field indicates whether the user is willing to engage in an EMAIL exchange.

The structure for Image value field is illustrated in Table 12.

Image	Req	Description
Containedimage	C	An image included in the attribute in transfer encoded form
Referredimage	C	A URL to the image
Value Field	M	The name of any Value Field in this attribute except Status, Image or Text

Table 12. Image

This value field associates an image with any of the value fields in the Availability attribute except in the Status, Text or Image field. The ContainedImage value field includes the image, the size or the format of the image may however be restricted. The ReferredImage value field includes a URL to resource having the associated image. The ValueField value field defines the value field the image is associated with. For instance, the publisher may associ-

ate an image with the value field PhoneAvail which currently has value 'DISC' (meaning that the user is limitedly available for telephony, for instance) in order to convey pictorial semantic information about the meaning of DISC. The Image value field may have multiple instances in this attribute. Whenever this value field is included in the attribute, its target value field must also be included in the same attribute. The association is valid only as long as the target value field is valid. When the target value field is updated or invalidated, any old association with this attribute must be discarded by the receiving client.

The structure for Text value field is illustrated in Table 13.

Text	Req	Description
Contained-Text	M	A text string
ValueField	M	The name of any Value Field in this attribute except Status, Image or Text

Table 13. Text

Text value field associates a text string with any of the value fields in the Availability attribute except in the Status, Image or Text field. The Text value field includes the text string in ContainedText and the name of the associated value field in ValueField. The size of the text may be limited in the ContainedText element. For example, the publisher may associate a text with value field PhoneAvail which currently has value 'NAVL' (e.g. 'in meeting until 14:00') in order to convey additional semantic information about the meaning of NAVL. The Text value field may have multiple instances in this attribute, i.e. the same text may be associated with a multiple value fields. Whenever this value field is included in the attribute, its target value field must also be included in the same attribute. The association is valid only as long as the target value field is valid. When the target value field is updated or invalidated any old associations with this value field must be discarded by the receiving client. Images and text may also be automatically added to a presence attribute.

PERSONAL STATUS ATTRIBUTES

PersonalStatus attribute indicates the personal status of the publisher. The options and details are specified by the value fields belonging to this attribute. Table 14 illustrates attribute structure for the PersonalStatus.

Information Element	Req	Type	Value	Description
Name-Spec.Name	M	String	Personal-Status	Name of the attribute
Value	M	XML	See Table 15 below	Value for the attribute

Table 14. PersonalStatus attribute structure

Table 15 illustrates the value fields of the PersonalStatus attribute.

Personal-Status.Value	Req	Single/Multiple	Description
Status	M	S	Name of the value field
Text	O	S	Name of the value field
Mood	O	S	Name of the value field
Time	O	S	Name of the value field
Image	O	M	Name of the value field

Table 15. PersonalStatus attribute values

The status value field as illustrated in Table 16 indicates the status of PersonalStatus information.

Information element	Status
Data type	An enumerated String
Format	One of the following values: ENA – The value fields included in this attribute contain up-to-date information. The value fields updated earlier and not included in this attribute are still up-to-date DIS – The value fields (if any) included in this attribute contain invalid information. The value fields updated earlier are invalid.
Description	Defines the publishing status of location attribute
Range	ENA | DIS

Table 16. Status of PersonalStatus attribute

This field indicates whether the publishing of this information is enabled or not. The DIS indication may also be used with the PersonalStatus attribute.

The Text value field indicates the status of the publisher in a free text form. The Mood value field indicates the mood of the publisher. The Time value field gives the local time of the publisher.

According to a preferred embodiment, an image may also be utilized in the PersonalStatus attribute. As already illustrated in Table 12 the Image value field associates an image with any of the value fields in this attribute except in the Status or Image field. For instance, the publisher may associate an image with value field Mood which currently hasvalue 'IN_LOVE' in order to convey pictorial semantic info about the meaning of IN_LOVE. The Image value field may be used in a similar way as already illustrated with attribute Availability.

The present invention can be implemented in the existing client devices and servers. They all have processors and memory with which the inventive functionality described above may be implemented. A computer program may be loaded from an internal or external memory to the processor of the server or the client device, causing, when executed in the processor, the means

to implement the inventive functionality. Also hardware implementation or a combination of software and hardware implementation may be used.

Fig. 4 shows a client device 402 in communication with a server 404, according to the present invention. The client device may be similar to one or more of the mobile clients or the non-mobile client of Fig. 1 or any of the other mobile clients shown in Figs. 2 and 3. Likewise, the server 404 may be similar to the server shown in Fig. 1 or any of the other servers shown in Figs. 2 and 3. The client device 402 includes means 406 for transmitting or requesting presence information as presence attributes to the server on a signal line 408. Transmitting presence information would correspond for instance to the step 302 of Fig. 3 in which presence attributes are sent from MC1 to PS or the step 202 of Fig. 2 in which presence attribute(s) are updated by the MC and sent to the PS. Requesting presence information on the line 408 would be comparable to the request step 203 of Fig. 2 wherein the MC requests a presence attribute from the SS. The transmission on the line 408 from the client device 402 to the server 404 could be similar to the transmission path shown in Fig. 1 from a mobile client (MC) through a mobile network (MNW) via an intermediary network ONW to the IMPS server(s). Or, it could be similar to the path from the non-mobile client C through the intermediary network ONW to the server S. Of course other possible paths are contemplated as well and the invention does not depend on the path of the physical media or combination of physical media used. The client device 402 also includes means 410 for receiving presence attributes from the server 404 on a signal line 412. Such presence information is categorized by a plurality of presence attribute types identified by attribute name.

The server 404 includes means 414 for receiving/transmitting attributes and for maintaining presence information based on the received presence attributes.

According to the invention, the client device 402 additionally includes means 416 for adding a qualifier to a presence attribute wherein the qualifier comprises one or more parameters specifying the use of the attribute. The added qualifier is provided on a signal line 418 for transmission by the means 406 on the line 408 to the server 404. This may be done via a means 420 for determining presence attributes in accordance with instructions received on a line 422 from an application 424. The application can also utilize the means 420 for requesting presence attributes. In either event, a signal may be provided on

a line 426 from the means 420 to the means 406 for transmitting or requesting presence attributes on the line 408. In case the means 416 has added a qualifier, for instance in updating a presence attribute as per step 202 of Fig. 2, the signal on the line 408 will include a presence attribute with a qualifier having one or more parameters specifying use of the attribute.

For handling presence attributes incoming on the line 412 from the server 404, the client device 402 will also include means 428 for processing a received presence attribute on the line 412 from the server 404 according to qualifier parameters in the received attribute. The received presence attribute on the line 412 may be received by the means 410 and provided on a line 430 to the means 428 for processing the received attribute. After processing, the means 428 may provide a signal on a line 432 to the application 424 for further use by the application.

The means for adding a qualifier 416 may include means 434 for specifying in the qualifier presentation settings of the attribute so that the client receiving the attribute from the server 404 will be able to present the attribute on the basis of the presentation settings. Consequently, a client device such as the client device 402 of Fig. 4 will include means 436 for presenting received attributes on the basis of such a qualifier specified by another client device and received from the server 404 on the line 412.

Likewise, the means 416 may include a means 438 for specifying in the qualifier to be sent to the server 404 an application to which the attribute should be addressed in the receiving client. For such a client device 402 receiving a qualifier specifying the application to which the attribute should be addressed, it will have means 440 for interpreting such an attribute received on the line 430 for addressing a received attribute to the application indicated by the qualifier.

Turning now to the server 404 in more detail, it may also include means 444 for determining on the basis of a qualifier whether to send an attribute to one or more client devices such as specified by a presence group in a presence database in the server. Such a determination may also depend on an authorization provided on a line 446 from a means 448 for providing such authorization. If the qualifier and authorization indicate that the attribute should be sent to one or more client devices, then the server 404 will do so, for instance to the client device 402 as well as similar devices, if appropriate.

It may be that a presence attribute intended for a particular client may not match that client's capabilities, according to information known to the server. Such information may be provided for instance by the means 444 on a line 450 to a means 452 for modifying the presence attribute to match the client's capabilities. The modified attribute may be provided back to the means 444 on the line 450. On the other hand, in cases where presence attributes are provided according to a "push" technology, the modified presence attribute may be provided on a line 454 to a means 456 which is capable of taking the appropriate steps to push the modified presence attribute to the client or to more than one client, as appropriate. This may be signalled for instance on a line 458 to the means 444.

It should be realized that the functional blocks shown in Fig. 4 may be carried out using discrete hardware, specialized integrated circuits, microcontrollers, software, firmware, etc., as will be apparent to those of skill in the art. Moreover, the functions attributed to distinct functional blocks in the figure need not be separate but can be incorporated in other blocks by free addition or subtraction of functions into or from other functional blocks. Likewise, the cooperative relationships between the functional blocks may be modified in their order and interrelationships while at the same time carrying out the same end results mentioned above. It should also be realized that the details of the client device and server device shown in Fig. 4 can take other forms which are similar to those shown, according to the invention. Other aspects of the invention can be illustrated in a similar but by no means identical way.

For instance, Fig. 5 shows a client device 502 communicating with a server device 504 with means 506 similar to the means 420 of Fig. 4 for transmitting presence information as presence attributes on a line 508 to a means 510 for receiving and maintaining presence attributes within the server 504, similar to the means 414 within the server 404 of Fig. 4. Likewise, the client device 502 may include means 512 for receiving presence attributes indicative of presence information on a line 514 from the means 510 of the server 504. Similar to the means 420 and 416 of the client device 402 of Fig. 4, the client device 502 of Fig. 5 may include means 516 for composing a presence information attribute identified by a combination of an authorizer, an attribute name and a qualifier, the authorizer specifying a body responsible for maintaining the attribute and the qualifier specifying the use of the attribute. The so-composed attribute may be provided on a line 518 to the means 506 for transmittal on the

line 508 to the server 504, as shown. The server 504 may include means 520 that is responsive to the so-composed attribute received on a line 522 from the means 510 for searching for already stored attributes containing the same combination of authorization, attribute name and qualifier. If the combination of identifiers of the received attribute is identical to that of the already stored attribute, for instance stored in a storage means 524, the received attribute is used to replace the already stored attribute over a signal line 526. In this way, if the attribute parameters have changed, the updated parameters will be stored in the storage means 524. Otherwise, the received attribute is added to the storage means as a new attribute. This function may be carried out in the means 520 as described above, or may be carried out in a completely separate means 528 which receives the received attribute on a line 530 from the means 520 and includes functions for replacing the already stored attributes with the received attributes if the combination of identifiers are the same and otherwise adding the received attribute over a connection line 532 between itself and the storage means 524.

The client device 502 will include similar functionalities as just described above as shown for instance by a means 540 which receives incoming attributes on a line 542 from the means 512 and searches for already stored attributes containing the same identifiers as the received attribute and means 542 for replacing the already stored attribute with the received attribute if the combination of identifiers of the received attribute is identical to that of the already stored attribute. Otherwise, the received attribute is added to a storage means 544. The means 540 for searching may include the means 542 within or they may be separate as shown in the figure. In the latter case, a signal line 546 provides attribute information concerning the received attribute from the means 540 to the means 542. The storage means 544 may be in bi-directional communication over line 548 with the means 542 for replacing or adding attributes and the line 550 with the means 540 for searching for already stored attributes.

As mentioned above, the service capabilities for a dynamic phonebook service embodiment of the present invention will now be described. A dynamic phonebook service can be viewed as a rich call service. It is useful "before the call" to enrich cases where B party's presence information is shown to the A party. In this case, the B party is one or more of the user phonebook entries. The presence information can be divided into the same categories as mentioned above, i.e., (1) client availability, (2) user availability, (3) local condi-

tions, (4) personal status, (5) client capabilities, (6) user attributes, and (7) extended presence service:

Conceptually, the Presence System consists of Presence Clients 602, 604, 606, Presence Users 608, 610, 612, Presence User Roles 614, 616, 618, 620, 622, 624, Presence Proxies 626 and Presence Servers 628, 630, as shown in Fig. 6. A Presence Client is the software or program which enables for the user an interaction with the Presence System. The User is a person who interacts with the Presence System using the Presence Client. A physical device 632, 634, e.g., mobile handset or PC, may have one 606, or in special cases, multiple presence client instances 602, 604. A presence client is owned by a single user. A user may own more than one client but then these clients are typically in different devices.

Users 608, 610, 612 are conceptually classified in Publishers and Subscribers. A publisher is the originator of presence information. A subscriber is the receiver of presence information. A User may be both publisher of own presence information and subscriber of some other publisher's presence information at the same time. A user may have one or more Roles. A publisher role is associated with a set of presence values called Presence Set. The presence values of two different presence sets of the same user are independent of each other and are associated with different roles. A subscriber role is the logical receiver of presence information of identical publisher role i.e. of the same presence set.

A Presence Proxy 626 is an optional network element that improves the scalability of the Presence Service. A proxy temporarily stores presence values of different presence sets traveling uplink from the publisher to the server or downlink from the server to a subscriber. When a client comes on-line the proxy may update the client with current presence information. Also when a publisher sends a new presence value to the server the proxy may update all the subscriber clients that are registered with the proxy. A proxy can cache the presence values only temporarily. Even when the presence info is coming from the publisher the proxy cannot assume that all the updates of this presence info is taking place via the same proxy. If the proxy is not aware of the subscriber group associated with a presence set then the proxy may ask for this information from the server.

A presence Server 628, 630 is a network element that maintains valid presence values and information on groups that are associated with each presence set. The server communicates with presence clients either directly or through a proxy. The server informs the proxy about the validity period of presence values cached by the proxy. When the validity period expires the proxy must either discard the values or refresh them from the server. The server assigns the validity periods presence item by item basis by monitoring how frequently the presence values change. The validity period is dynamic i.e. it may change during the lifetime of the presence item.

A presence server exchanges presence information also with other presence servers, as shown in Fig. 6. For instance, if the publisher and subscriber belong admistratively to different presence servers then the presence info must go through both servers. In case the servers are incompatible there needs to be a gateway function on one or both servers.

Figure 7 shows the structure of a presence database 702, according to the invention. A single presence item 704 has three properities: name 708, attributes 710 and value 712. A presence set 714 consists of a single or more presence items. The presence set 714 belongs to a single role 716 of the user. There cannot be more than one presence set for a single role. In addition there is a single authorisation group 718 that belongs to a single role 716. The authorisation group consists of members that have the right to subscribe the the whole or a part of the presence set of the same role.

The items 704, 720, ..., 722 in a presence set are unique i.e they can be distinguished from each other. The items are primarily distinguished from each other by their name. In case the presence set contains two or more items with the same name then there must be an attribute in each item that carries the id's of these items. The members 724, 726, ..., 728 of a group 718 are unique. Different presence sets in different roles 716, 730, ..., 732 of the same publisher 734 may contain items with the same name or id. Different groups may likewise contain the same members.

A role 716 may be identified by a Role ID, Group ID or Presence Set ID. For instance, the Group ID may be assigned by the service provider and is unique within the service providers domain. Therefore, the following ID's would be needed to address individual elements in the presence database:

GroupID; ItemName

In case the presence set contains more than one item with the same name the ItemName must have the ItemID attribute assigned.

Note that ID's such as UserID, DeviceID and ClientID are not needed.

A. EXEMPLARY PRESENCE PROTOCOL

A.1.0 Unsubscribed Presence (Fig. 8)

The presence information of a user can be obtained separately from messaging services by issuing a query to the presence server, as indicated for instance in the message flows presented in steps 203, 204, 205, 206 of Fig. 2, or by subscribing and receiving presence items as in Fig. 9, or by getting presence items as shown in Fig. 8.

The user of a presence service may, at any suitable time, update his presence information in the presence server by sending update presence message as shown in Fig. 8. As mentioned, a user may issue a get presence message to request the presence information of some other user as also shown in Fig. 8. The presence information is delivered back to the requesting user.

The publisher may update his presence information only partially. Similarly, the user may request only partial presence information.

If the user does not have authorisation for a requested presence information an empty content is sent to the user.

The authorisation of presence information for a given user is done by including the user into the group corresponding to the presence set. This is described below in the part labeled 'Management of presence database'.

A.1.1 UpdatePresence

Direction: Presence Client -> Presence Server

Content Model: (TransactionID, GroupID, Presence)

Attributes: None

Usage: This primitive updates the values of one or more presence items on the presence server. The presence items to be updated are carried in the information element Presence. If there are items in the Presence that do not exist in the server for the given GroupID then the server allocates storage for these new items and copies the contents from this message. Otherwise the server replaces the old value with the new value.

No UserID or PublisherID is needed in the content. The association between the publisher and the XML-document containing the UpdatePresence primitive is made in the authentication protocol.

A.1.2 TransactionID

Content Model: (#PCDATA)

Attributes: None

Usage: This is a unique idenfier that associates the request from client to server with the corresponding response from server to client. The client may have sent more than one request to the server before it gets the first response back. The first response does not necessarily refer to the first request. Therefore there needs to be a mechanism that associates requests and responses together. Typically a TransactionID is a sequence number. It is assigned by the client in the request and used by the server in the response.

A.1.3 GroupID

Content Model: (#PCDATA)

Attributes: None

Usage: This id uniquely identifies within the service provider domain the publisher role including the authorisation group and presence set. It is assigned by the presence service provider.

A.1.4 UpdateStatus

Direction: Presence Server -> Presence Client

Content Model: (TransactionID, StatusCode)

Attributes: None

Usage: This primitive is the response from the presence server to the client UpdatePresence request. The StatusCode may have following values: not supported, new allocation, success, failure.

A.1.5 GetPresence

Direction: Presence Client -> Presence Server

Content Model: (TransactionID, ((GroupID, PresenceNames?)*) | PresenceNames?)

Attributes: None

Usage: This primitive is used to request presence information from the presence server. There may be zero or more GroupIDs. For each GroupID there may be coupled an optional PresenceNames. When the GroupID is not used then the request scope is all the groups the user is a member. The server associates the requesting user's identity to the request by the authentication protocol. The PresenceNames is defined later in this document. It lists those presence items by name from which the value is requested. It is optional. It is either coupled with GroupIDs or may be alone when no GroupIDs are listed. When coupled with GroupIDs the server limits the search of the presence items for the given group. When not present the values of all presence items is requested.

A.1.6 PresenceItems

Direction: Presence Server -> Presence Client

Content Model: (TransactionID, StatusCode, (GroupID, Presence)*)

Attributes: None

Usage: This primitive supplies the list of presence items requested and the result status of the request. The element Presence is the presence set corresponding to a given GroupID and is defined later in this document. The StatusCode may have following values: not authorised, item not available, success, failure.

A.2.0 Subscribed Presence (Fig. 9)

Another mechanism to deliver presence information is to subscribe someone's presence information. The message flow is presented in Fig. 9.

The requesting user sends a subscribe presence message A.2.1 to the presence server to subscribe someones presence information.

When the subscription of presence information is complete, the requesting user will receive new presence information A.2.2 initially and always when the other party updates its presence information.

When the requesting user does not any more want to receive the presence information, he may unsubscribe A.2 the presence information. Alternatively, the presence information may be subscribed to a time period and the unsubcribe message is not needed.

The requesting user may subscribe only part of the presence information that he is authorised to get.

A.2.1 SubsPresence

Direction: Presence Client -> Presence Server

Content Model: (TransactionID, SubsPeriod?, ((GroupID, Presence-Names?)*) | PresenceNames?)

Attributes: None

Usage: This primitive is used to subscribe presence information from publisher roles for which the subscriber is authorised. The optional GroupID specifies the publisher role which is subscribed. When the GroupID is missing then all the roles for which the subscriber is authorised are subscribed. Each GroupID may be associated with a PresenceNames which limits the subscribed items within the group to those listed in the PresenceNames. The optional SubsPeriod specifies the length in time for the subscription. When it is missing the subscription lasts until an corresponding UnsubsPresence primitive is invoked or until the service provider terminates subscription. The PresenceNames specifies the presence items that are subscribed from the authorised set. When it is missing then all the presence items that the user is authorised for within the presence set (GroupID) are subscribed. When used alone without a GroupID then the named items from all authorised groups are subscribed. PresenceNames is defined later in this document.

The identity of the subscriber is discovered in the authentication protocol.

A.2.2 SubsPeriod

Content Model: (#PCDATA)

Attributes: None

Usage: This defines the length of the subscription period. The time unit is TBD. Alternatively the content may indicate the date when the subscription ends.

A.2.3 PushPresence

Direction: Presence Server -> Presence Client

Content Model: (GroupID, Presence)+

Attributes: None

Usage: This primitive pushes the subscribed presence items that have been modified by the publisher to the users. The GroupID identifies the role.

A.2.4 UnsubsPresence

Direction: Presence Client -> Presence Server

Content Model: (TransactionID, ((GroupID, PresenceNames?)*) | PresenceNames?)

Attributes: None

Usage: This primitive is used to unsubscribe presence information. The termination of subscription may be limited to given groups and can further be limited to given presence items. When no GroupID is given it is possible to limit the subscription termination for the named presence items in all the groups. When no GroupID and no PresenceNames is given then the whole subscription of the user is terminated.

A.2.5 SubsStatus

Direction: Presence Server -> Presence Client

Content Model: (TransactionID, StatusCode, (GroupID, PresenceNames)*)

Attributes: None

Usage: This primitive returns the StatusCode for the unsubscribe operation and lists the presence items and groups that are still subscribed. The StatusCode may have following values: not supported, non-existent item, non-subscribed item, success, failure.

A.3.0. Management of Presence Database

The presence user may manage his user groups and presence sets in the presence server.

A role containing a user group and presence set is created using CreatePresGroup message A.3.1. The presence server will reply with PresGroupStatus message A.3.6 indicating the status of the requested operation and an id for the created group. A PresenceInfo message A.3.7 is sent to members of the group indicating the presence items they are authorised to get or subscribe.

The presence user may request group info with the GetPresGroupStatus message A.3.8. The group info request is limited to the owner of the group.

The presence user owning the user group may add and delete members of the group or presence items for the presence set.

The publisher may send delete a group by message DeletePresGroup.

A.3.1 CreatePresGroup

Direction: Presence Client -> Presence Server

Content Model: (TransactionID, MemberList, Presence)

Attributes: None

Usage: This primitive requests the server to create a role in the presence server. The authorisation group contains the members in the memberlist and the presence set is contained in Presence.

A.3.2 MemberList

Content Model: (MemberDescription)+

Attributes: None

Usage: The list of members. The description is originated from the user.

A.3.3 MemberDescription

Content Model: (UserOriginatedMD?, ServerOriginatedMD?)

Attributes: None

Usage: The user originated part is a description of a member as seen by the user. The server originated part is a description of a member as stored in the server database.

A.3.4 UserOriginatedMD

Content Model: (#PCDATA)

Attributes: None

Usage: This is a user originated means to identify a person. It may be a name, an MSISDN of one of the phones of the target person, an email address, a SIP address etc. It may also be a combination of these. The dataformat

of the description is not within the scope of this document but known formats such as vCard should be used.

A.3.5 ServerOriginatedMD

Content Model: (#PCDATA)

Attributes: None

Usage: This is a server originated means to idenfy a person. It may be a name, an MSISDN of one of the phones of the target person, an email address, a SIP address etc. It may also be a combination of these as stored in the server database. The dataformat of the description is not within the scope of this document but known formats such as vCard should be used.

A.3.6 PresGroupStatus

Direction: Presence Server -> Presence Client

Content Model: (TransactionID, StatusCode, (GroupID, MemberList, PresenceNames)*)

Attributes: None

Usage: This primitive returns the GroupID's of the currently existing publisher roles together with the group members and presence item names for the role. The server has assigned a new GroupID for the new role if the operation was succesful. The server has filled the server originated part of the memberlist. This allows the publisher to verify that the group contains the right members. The server should not give any confidential or personal information in the memberlist beyond the level already existing in the user originated part or a level that is needed to uniquely identify a person. This is TBD. The StatusCode may get following values: not supported, unknown member, unknown presence item, success, failure.

A.3.7 PresenceInfo

Direction: Presence Server -> Presence Client

Content Model: (GroupID, Presence?)+

Attributes: None

Usage: This primitive advertises the new presence items to the authorisation group. The message contains all the presence items of the given authorisation group – both subscribed and unsubscribed. This message is sent to new members in the group if the group membership has changed or to all the

members of the group if new presence items have been added. The message is sent to removed group members so that there is no presence set (Presence) associated with the GroupID.

A.3.8 GetPresGroupStatus
Direction: Presence Client -> Presence Server
Content Model: (TransactionID, GroupID*)
Attributes: None
Usage: This primitive requests the PresGroupStatus message from the server. The user may only request info from his own groups. The GroupID limits the info to the given groups. When no GroupID exist then the status is returned from all the groups owned by the requestor.

A.3.9 AddMembers
Direction: Presence Client -> Presence Server
Content Model: (TransactionID, (GroupID, MemberList)+)
Attributes: None
Usage: This primitive requests the server to add the named members into given groups.

A.3.10 RemoveMembers
Direction: Presence Client -> Presence Server
Content Model: (TransactionID, (GroupID, MemberList)+)
Attributes: None
Usage: This primitive requests the server to remove the named members from the given groups. The MemberList is filled with the server originated info.

A.3.11 DeletePresGroup
Direction: Presence Client -> Presence Server
Content Model: (TransactionID, GroupID*)
Attributes: None
Usage: This primitive requests the server to remove one or more of the publisher roles. The GroupID identifies the role that is to be removed. When no GroupID is given then all the publisher roles are removed.

A.3.12 AddPresence

Direction: Presence Client -> Presence Server

Content Model: (TransactionID, (GroupID, Presence)+)

Attributes: None

Usage: This primitive requests the server to add given presence items into given roles.

A.3.13 RemovePresence

Direction: Presence Client -> Presence Server

Content Model: (TransactionID, (GroupID, Presence?)+)

Attributes: None

Usage: This primitive requests the server to remove given presence items from the given roles. When there is no Presence element associated with GroupID then all the presence items are removed from the role.

A.3.14 Dynamic Client Change

A publisher or subscriber may change his client at any time. Typically also the client device changes. In a rare case the user may only change the client instance in the same device. The new device may be a TE and the old a MT or vice versa. Also change to a new MT is possible. The client change requires the user to copy or synchronize his presence data from the old client to the new client. In case of a publisher there might be some client specific presence data that is static for a given client and is not included in the synchronisation set. The synchronisation may take place locally or may be network assisted. In any case the existing synchronisation protocols should be used. This is not within the scope of the presence protocol.

Another issue brought by the subscriber client change is the registry of the new client to be the active client that receives the presence related push messages. The registry may take place automatically (no user confirmation needed), semi-automatically (user confirmation required) or manually (user must initiate the registry) depending whether

- User activates a new client in the same device (automatic);
- User SIM changes to the new device (automatic);
- User has a different SIM in the new device but only the new device is turned on (semi-automatic); or

- User has a different SIM in the new device and both the new device and old device are on (manual).

In all the cases the registry may be handled by lower level protocols. For example if the SIP-protocol is the chosen transport protocol for presence documents then this protocol can also be used for the registry.

A.4.0 DTD for the Presence Protocol

```
<!-- Root element -->

<!ELEMENT PresProtocol ( UpdatePresence | UpdateStatus | GetPresence | PresenceItems | SubsPresence | PushPresence | UnsubsPrecence | SubsStatus | CreatePresGroup | PresGroupStatus | PresenceInfo | GetPresGroupStatus | AddMembers | RemoveMembers | DeletePresGroup | AddPresence | RemovePresence) >

<!ATTLIST PresProtocol Version NMTOKENS #REQUIRED >

<!ELEMENT UpdatePresence (TransactionID, GroupID, Presence) >

<!ELEMENT TransactionID (#PCDATA) >

<!ELEMENT GroupID (#PCDATA) >

<!ELEMENT Presence (#PCDATA) >

<!ELEMENT UpdateStatus (TransactionID, StatusCode) >

<!ELEMENT StatusCode (#PCDATA) >

<!ELEMENT GetPresence (TransactionID, ((GroupID,
PresenceNames?)*) | PresenceNames?) >

<!ELEMENT PresenceNames (#PCDATA) >

<!ELEMENT PresenceItems (TransactionID, StatusCode, (GroupID,
```

```
Presence)*) >

<!ELEMENT SubsPresence (TransactionID, SubsPeriod?, ((GroupID,
PresenceNames?)*) | PresenceNames?) >

<!ELEMENT SubsPeriod (#PCDATA) >

<!ELEMENT PushPresence ((GroupID, Presence)+) >

<!ELEMENT UnsubsPresence (TransactionID, ((GroupID,
PresenceNames?)*) | PresenceNames?) >

<!ELEMENT SubsStatus (TransactionID, StatusCode, (GroupID,
PresenceNames)*) >

<!ELEMENT CreatePresGroup (TransactionID, MemberList,
Presence) >

<!ELEMENT MemberList ((MemberDescription)+) >

<!ELEMENT MemberDescription (UserOriginatedMD?,
ServerOriginatedMD?) >

<!ELEMENT UserOriginatedMD (#PCDATA) >

<!ELEMENT ServerOriginatedMD (#PCDATA) >

<!ELEMENT PresGroupStatus (TransactionID, StatusCode,
(GroupID, MemberList, PresenceNames)*) >

<!ELEMENT PresenceInfo ((GroupID, Presence?)+) >

<!ELEMENT GetPresGroupStatus (TransactionID, GroupID*) >

<!ELEMENT AddMembers (TransactionID, (GroupID, MemberList)+)
>
```

<!ELEMENT RemoveMembers (TransactionID, (GroupID, MemberList)+) >

<!ELEMENT DeletePresGroup (TransactionID, GroupID*) >

<!ELEMENT AddPresence (TransactionID, (GroupID, Presence)+) >

<!ELEMENT RemovePresence (TransactionID, (GroupID, Presence?)+) >
<!-- End of DTD -->

B. EXEMPLARY PRESENCE CONTENT FORMAT

As stated above, the presence content can be divided in the following classes:

Client Availability: Presence attributes describing the availability of the client for communication, for instance, network reachability, GPRS attached, on/off status.

User Availability: Presence attributes describing the availability of the user for communication, for instance, ready, meeting, busy, away, in call, chatting, don't disturb, etc.

Local Conditions: Presence attributes describing the local environment at the user, for instance, local time, noisy/silent environment, in-door, out-door, location of the user, in terms of, for instance, geographical location, visited PLMN, city/street, premises.

Personal Status: Various personal attributes describing personal user status, for instance, mood, personal interests and intentions.

Client Capabilities: Presence attributes describing the capabilities of the client, for instance, to support different means of communication, different media types and different features.

User Attributes: Presence attributes allowing the client or the user to define their own textual presence values and references to external values.

Extended Presence Service: Service provider dynamically defined non-standard presence attributes which however need to be passed through standard presence servers and proxies.

B.1. DESIGN GOALS

The bulk of presence communication is sending the changes in the value part of presence items to the subscribers. Often only a single value has changed. In order to minimize the amount of transferred data the design goal is to keep the hierarchical representation of presence items as flat as possible. For this reason the presence items are scoped under the presence classes described above. Instead the classes may be given in the presence items as an optional attribute.

D.2 COMMON ATTRIBUTES

The presence elements contain a variety of attributes most of which can be used in any presence item.

D.2.1 Version

Content Syntax: NMTOKENS
Mandatory: yes
Usage: This gives the version of the presence content format

D.2.2 Class

Content Syntax: NMTOKEN
Mandatory: no
Usage: This indicates the class the attribute belongs to. Possible values are: CLIENT_AVAILABILITY, USER_AVAILABILITY, LOCAL_CONDITIONS, PERSONAL_STATUS, CLIENT_CAPABILITIES, USER_ATTRIBUTES, EXT_PRES_SERVICE.

D.2.3 ID

Content Syntax: PUBIDLITERAL
Mandatory: no
Usage: This is a unique ID for a named presence item. It is assigned by the client.

D.2.4 Cacheability

Content Syntax: NMTOKEN
Mandatory: no

Usage: This indicates whether the presence item can be cached by proxies or not. The possible values are: YES, NO.It is assigned by the client. If this attribute is missing a value of NO is assumed.

D.2.5 ValidityPeriod
Content Syntax: NMTOKEN
Mandatory: no
Usage: This is a validity period for the cached presence item. The value is time in seconds. It is assigned by the presence server.

D.2.6 DeviceName
Content Syntax: NMTOKEN
Mandatory: no
Usage: This is the name of the client device. It is assigned by the client.

D.2.7 Accuracy
Content Syntax: NMTOKEN
Mandatory: no
Usage: This is the accuracy of a positioning device. It is given in meters.

D.2.8 ImageType
Content Syntax: NMTOKEN
Mandatory: no
Usage: This is the content encoding of an image. Some possible values are: JPEG, GIF, BMP.

D.2.9 SoundType
Content Syntax: NMTOKEN
Mandatory: no
Usage: This is the type of sound codec used to encode the sound. Some possible values are: AMR, EFR, MP3, AAC, MIDI

D.2.10 ExtRef
Content Syntax: PUBIDLITERAL
Mandatory: no
Usage: This is a URL giving an external reference.

D.2.11 ExtRefChange
Content Syntax: NMTOKEN
Mandatory: no
Usage: This indicates that the content of an external reference has changed. The possible values are: YES, NO

D.2.12 ContentChange
Content Syntax: NMTOKEN
Mandatory: no
Usage: This is a counter from 0 to 255 which indicates a change in the content value. The server or proxy may store the contents (but is not required to) of the last 32 values. Same values for two contents within the last 32 values should correspond to same content

D.3 Client Availability

D.3.1 DeviceOn
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod, DeviceName
Usage: This gives the on/off status of the user terminal. The publisher adds this item into the role in the server by using the group management messages. Thereafter the value part is maintained by the network. The user may have more than one terminal in his presence information. In this case the use of ID attribute is mandatory. The content can have values "ON" or "OFF". When the user's terminal is on but outside network coverage the network assigns value "OFF" for this item.

D.3.2 DeviceRoaming
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod, DeviceName
Usage: This indicates to the subscriber that the publisher is roaming in a visited network. The user may have more than one terminal in his presence information. In this case the use of ID attribute is mandatory. The content can have values "YES" or "NO"

D.3.3 NetworkType
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod, DeviceName
Usage: This indicates the type of mobile network the publisher is currently attached. The content can have values "2G", "3G-99", "3G-R4" or "3G-R5". The user may have more than one terminal in his presence information. In this case the use of ID attribute is mandatory.

D.4 User Availability

D.4.1 UserStatus
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This indicates the current status of the publisher in terms of amount of distraction he is willing to accept. The following values are defined: "available", "silent", "in-car", "busy".

D.4.2 PreferredContact
Content Model: (#PCDATA)
Attributes: Class, ID, Cachebility, ValidityPeriod
Usage: This indicates what is the current preferred contact method for the publisher. Following values are defined: "PHONE", "VOICE_MESSAGE", "MESSAGE", "MAIL", "NO_CONTACT".

D.4.3 PreferredDefaults

Content Model: ((TimePeriod, PrefCont)*)

Attributes: Class, ID, Cacheability, ValidityPeriod

Usage: This is presence metainformation that the publisher may send to presence server. The information is associated to a role as other presence information but this item is not available to subscribers. Instead it controls the values of the element 'PreferredContact'. It specifies a default preferred contact for given time periods. User specified time periods may overlap or a time period may even completely enclose another. The server changes the value of PreferredContact element according to the PreferredDefaults at the expiration of given periods. It however does not block a user changing the PreferredContact value directly. Also the user changing the PreferredContact value directly does not block the server changing it again when a period expires. The user may remove the default mechanism by sending this element with empty contents to the server.

D.4.4 TimePeriod

Content Model: (PeriodStart, PeriodEnd, RepetitionPeriod?, Period-Precedence)

Attributes: None

Usage: This desribes the start of a period e.g. start time, the end of the period e.g. end time, the repetion period e.g. 'DAY' and the presedence of the period.

D.4.5 PeriodStart

Content Model: (#PCDATA)

Attributes: None

Usage: This is the start period. It may be Time, Time-DayofWeek, Time-DayofMonth or a full date..

D.4.6 PeriodEnd
Content Model: (#PCDATA)
Attributes: None
Usage: This is the start period. It may be Time, Time-DayofWeek, Time-DayofMonth or a full date. The resolution must be the same as in PeriodStart.

D.4.7 RepetitionPeriod
Content Model: (#PCDATA)
Attributes: None
Usage: This gives the repetition period for the start-end. It must be one of the following values: "DAY", "WEEK", "MONTH". When this element is not included in the period description then it is assumed that no repetion is applied.

D.4.8 PeriodPrecedence
Content Model: (#PCDATA)
Attributes: None
Usage: This element resolves the conflict between overlapping periods. It can have a numeric value between 0 and 9. Number 0 is the highest precedence. When two or more periods overlap then the server takes the contact preference associated with the highest precedence value.

D.4.9 PrefCont
Content Model: (#PCDATA)
Attributes: None
Usage: This indicates what is the period associated default preferred contact method for the publisher. It can have the same values as the element 'PreferredContact'.

D.5 Local Conditions

D.5.1 LocalTime
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This gives the local time of the publisher

D.5.2 MeasuredLocation
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod, Accuracy
Usage: This gives the measured position of the client device. The measurements may be either sensor based (e.g. GPS) or network based or combination of both. The attibute Accuracy gives indication of the average positioning accuracy achieved by the method. The content includes at least the lateral position (x- and y-coordinates) but may include also the vertical position. If the service provider supports 'ConvertedLocation' described below then the MeasuredLocation may be input information for the service provider conversion process e.g. map matching.

D.5.3 ConvertedLocation
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This information is typically originated from the service provider but may also originate from the publisher. It gives the location of the user given in a human understandable text form such as a name of a street. The information is derived by converting a measured position into this form by e.g. map matching.

D.5.4 StatedLocation
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This is the location of the publisher as stated by the publisher himself. The content is a short text string.

D.5.5 UserEnvironment
Content Model: (EnvAttributes+)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This gives some environmental attributes about the user.

D.5.6 EnvAttributes
Content Model: (#PCDATA)
Attributes: None
Usage: The attributes defined are: "INDOOR", "OUTDOOR", "QUIET", "NOISY", "ALONE", "IN_GROUP".

D.6 Personal Status

D.6.1 StatusText
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This is a short text (about 30 characters) that the user may write.

D.6.2 StatusImage
Content Model: (#CDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod, ImageType
Usage: This is an image the user may attach to his status information. It is carried in the XML content in a transfer encoded form e.g. base64. The ImageType attribute describes the content encoding of the image e.g. jpeg.

D.6.3 StatusSound
Content Model: (#CDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod, SoundType
Usage: This is a short (about 5 to 30 seconds) sound clip the user may attach to his status information. It is carried in the XML content in a transfer encoded form e.g. base64. The SoundType attribute describes the content encoding of the sound e.g. AMR, MP3, AAC, MIDI etc.

D.7 Client Capabilities

The client capabilities in the context of presence mean the capability of the device hosting the client for various types of human to human communication. This is very much different and simpler issue than the case of allowing an application software to take maximum advantage of the existing hw and sw capabilities of the client device.

The classes of human to human communication are: messaging, e-mail, voice call and multimedia call. In particular presence is excluded from the communication classes. The purpose of presence is to make others aware of the communication means and other user attributes but not to be a two way communication means itself. For this reason the use of the term 'Client Capabilities' to classify presence information is somewhat misleading.

The client capabilities are scoped with the network capabilities. This means for example that the publisher has the video call capability only when his client device has this capability AND the network he is currently roaming supports this capability. This makes the client capability information dynamic.

D.7.1 MessagingCapabilities

Content Model: (MessType*)

Attributes: Class, ID, Cacheability, ValidityPeriod

Usage: This gives the dynamic messaging capability list of the client device. An empty list means that no messaging capabilities currently exist.

D.7.2 MessType

Content Model: (#PCDATA)

Attributes: None

Usage: This gives the messaging capability type of the client device. The content may be one of the following: "SMS", "MMS" or "X-*message application name*". The field *message application name* is a device specific messaging method e.g. smart messaging.

D.7.3 EmailClient
Content Model: (EmailClientType*)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This gives the dynamic e-mail capabilities of the client device. An empty list means that no e-mail capabilities currently exist.

D.7.4 EmailClientType
Content Model: (#PCDATA)
Attributes: None
Usage: This gives the e-mail client type of the client device. The content may be one of the following: "SMTP", "POP3", "IMAP4" or "X-*mail application name*". The client software should not render the mail protocol names as such to the user. Instead the client software should understand the user device email environment and interpret these names in the scope of this email environment. The information presented to the user should be also meaningful to the user.

D.7.5 VoiceCallCapability
Content Model: (#PCDATA)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This gives the dynamic voice call capabilities of the client device. The content is one of the following: "NONE", "VOICE_CALL", "RICH_VOICE_CALL".

D.7.6 MultimediaCallCapabilities
Content Model: (MMCap*)
Attributes: Class, ID, Cacheability, ValidityPeriod
Usage: This gives the dynamic one-way and two-way multimedia communication capabilities of the user. An empty list indicates that no capabilities currently exist.

D.7.7 MMCap
Content Model: (#PCDATA)

Attributes: None

Usage: This gives the multimedia call capability type. It may be one of the following: "UPLINK_VIDEO_STREAMING", "DOWNLINK_VIDEO_STREAMING", "VIDEO_CALL", "RICH_VIDEO_CALL".

D.8 User Attributes

This class contains both user defined and client specific presence data.

D.8.1 UserPresenceItem

Content Model: (#PCDATA)

Attributes: Class, ID, Cacheability, ValidityPeriod, ExtRef, ExtRefChange

Usage: This gives a means for the user to defined their own presence items. If more than one user specified presence item is defined for the same role then the use of ID attribute is mandatory. The content may be a short text. The optional attribute ExtRef is a URL of an external object referenced by this presence item. The URL may refer to another part of the multipart MIME that carries the XML-document of the presence item or it may be a reference to an external object. The optioanal attribute ExtRefChange may be used to indicate the subscribers that the external object has changed.

D.8.2 ClientTypeRequest

Content Model: (EMPTY)

Attributes: Class, ID, Cacheability, ValidityPeriod

Usage: This is a special element used by the publisher to request the client types of his subscribers. When a publisher includes this element into his presence data for a given role then the server sends this to all the subscribers of the presence information with the PushPresence message. Optionally it may also be

included in the PresenceItems message used by non-subscribers to request specific presence items. This element is not advertised in the PresenceInfo message. It is not a presence element that can be subscribed to. The presence server after having first time sent this item to the subscribers may periodically include this item also to further PushPresence messages to make sure that all the subscriber clients have this item.

The clients upon receiving this item send their client types to the presence server. In addition a client should send it's type to the server every time the client is activated when this item is the presence info stored in the client.

An alternative way of the publisher getting to know the client types of his subscribers would have been for the publisher to subscribe their client types. This however is a different business model. A publisher may not always want to be a subscriber of his own subscribers. This element allows the publisher to get the client types within the existing contractual frame between the publisher and the presence service provider. In addition this method mandates the subscriber client to send their types to the server. If the publisher would have used the subscription model to get the client types then the authorisation of this information to the publisher would have been at the discretion of the subscriber.

D.8.3 ClientType

Content Model: (ClientName, ClientManufacturer, ClientVersion)

Attributes: Class, ID, Cacheability, ValidityPeriod

Usage: Knowledge of the subscriber client types is used by publishers to enable client specific extensions to the presence set. It does not make sense for a publisher to use these extension items if his subscriber clients are not able to decode them.

A subscriber client sends the cliend type info to the presence server using the UpdatePresence message. The GroupID used in the message belongs to the publisher's role and not to the subsriber's own role. This is an exception from the general rule that a user is

allowed to update the presence information of his own role only. The presence server sends the information of this element to the publisher using the PushPresence message. The GroupID used belongs to the publisher role. From the publisher point of view then he is getting presence information from his own presence set.

A publisher may include his own client type into his presence set. Then this information is handled like any other presence information and does not imply any special behaviour in presence servers or subsciber clients.

D.8.4 ClientName
Content Model: (#PCDATA)
Attributes: None
Usage: This element gives the name of the presence client application

D.8.5 ClientManufacture
Content Model: (#PCDATA)
Attributes: None
Usage: This element gives the name of the presence client manufacture

D.8.6 ClientVersion
Content Model: (#PCDATA)
Attributes: None
Usage: This element gives the version of the client application.

D.8.7 ClientPresenceItem
Content Model: (ClientType, CliPresItem)
Attributes: Class, ID, Cacheability, ValidityPeriod, ContentChange
Usage: This element is the publisher client application specific presence item. The ClientType is the client type of the publisher's presence client. No attributes are used in the ClientType item when used as part of this element. If the publisher defines more than one client specific presence item then the use of the ID attribute is

mandatory. The optional attribute ContentChange is a counter from 0 to 255. When used it is increased by one every time the content of the presence item changes. This provides an alternative means for the presence server to detect change in the content of a non-standard presence item. The other alternative would have been for the presence server to assume that the content changes every time this item is sent to the server with the same ID attribute. The server behaves in this way when the ContentChange attribute is not used by the client. When two items with the same ID attribute have the same counter value in ContentChange then the content of these two items is the same.

D.8.8 CliPresItem

Content Model: (#PCDATA)

Attributes: None

Usage: This element is the client specific presence extension. It's syntax and structure is assumed to be known to the client but not to the network elements.

D.9 Extended Presence Service

The extended presence service class is meant to provide a service provider specific extension to the presence service. Like the client application specific extension only the service end points (i.e. the publisher and the subscriber clients) need to be able to decode the extensions. It is possible that the presence server also understands the extensions but this is not mandated by this document. The difference between the client specific extension method and this method is that the service provider controlled clients do not need to implement the standard presence items. It is assumed that the presence items are completely defined by the service provider and there is a means to update the user device hosted clients with new features. One possible update method is sw downloading via air. If a service provider creates new types of presence items then he needs

to update the existing clients to support these new types. The service provider controlled client communicates with the presence server directly and not via a standard presence client. A standard presence client ignores all elements of this class.

An extended presence service uses primarily the elements of this class. It is allowed to use also the standard presence element definitions. It uses the standard presence protocol.

D.9.1 ExtPresence

Content Model: (#PCDATA)

Attributes: Class, ID, Cacheability, ValidityPeriod, ContentChange

Usage: The use of ID attribute is mandatory if more than one presence item is defined by this mechanism. The optional ContentChange attribute is available also for this element. The syntax and represantation of the contents are not in the scope of this document. The presence servers and proxies should allow the contents to refer to objects carried in the same multipart MIME as the XML-documents containing this element.

D.10 DTD for the presence content format

```
<!-- Root element -->

<!ELEMENT Presence ( DeviceOn | DeviceRoaming | NetworkType | UserStatus | PreferredContact | PreferredDefaults | LocalTime | MeasuredLocation | ConvertedLocation | StatedLocation | UserEnvironment | StatusText | StatusImage | StatusSound | MessagingCapabilities | EmailClient | VoiceCallCapability | MultimediaCallCapabilities | UserPresenceItem | ClientTypeRequest | ClientType | ClientPresenceItem | ExtPresence) >

<!ATTLIST Presence Version NMTOKENS #REQUIRED >

<!ELEMENT DeviceOn (#PCDATA) >

<!ATTLIST DeviceOn
```

```
                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                DeviceName NMTOKEN #IMPLIED >

        <!ELEMENT DeviceRoaming (#PCDATA) >

        <!ATTLIST DeviceRoaming
                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                DeviceName NMTOKEN #IMPLIED >

        <!ELEMENT NetworkType (#PCDATA) >

        <!ATTLIST NetworkType

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                DeviceName NMTOKEN #IMPLIED >

        <!ELEMENT UserStatus (#PCDATA) >

        <!ATTLIST UserStatus

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT PreferredContact (#PCDATA) >
```

```
<!ATTLIST PreferredContact

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT PreferredDefaults ((TimePeriod, PrefCont)*) >

<!ATTLIST PreferredDefaults

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT TimePeriod (PeriodStart, PeriodEnd,
        RepetitionPeriod?, PeriodPrecedence) >

<!ELEMENT PeriodStart (#PCDATA) >

<!ELEMENT PeriodEnd (#PCDATA) >

<!ELEMENT RepetitionPeriod (#PCDATA) >

<!ELEMENT PeriodPrecedence (#PCDATA) >

<!ELEMENT PrefCont (#PCDATA) >

<!ELEMENT LocalTime (#PCDATA) >

<!ATTLIST LocalTime

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
```

```
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT MeasuredLocation (#PCDATA) >

        <!ATTLIST MeasuredLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                Accuracy NMTOKEN #IMPLIED >

        <!ELEMENT ConvertedLocation (#PCDATA) >

        <!ATTLIST ConvertedLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT StatedLocation (#PCDATA) >

        <!ATTLIST StatedLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT UserEnvironment (EnvAttributes+) >

        <!ATTLIST UserEnvironment

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
```

```
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT EnvAttributes (#PCDATA) >

        <!ELEMENT StatusText (#PCDATA) >

        <!ATTLIST StatusText

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT StatusImage (#CDATA) >

        <!ATTLIST StatusImage

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ImageType NMTOKEN #IMPLIED >

        <!ELEMENT StatusSound (#CDATA) >

        <!ATTLIST StatusSound

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                SoundType NMTOKEN #IMPLIED >

        <!ELEMENT MessagingCapabilities (MessType*) >
```

```
<!ATTLIST MessagingCapabilities

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT MessType (#PCDATA) >

<!ELEMENT EmailClient (EmailClientType*) >

<!ATTLIST EmailClientType

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT EmailClientType (#PCDATA) >

<!ELEMENT VoiceCallCapability (#PCDATA) >

<!ATTLIST VoiceCallCapability

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED >

<!ELEMENT MultimediaCallCapabilities (MMCap*) >

<!ATTLIST MultimediaCallCapabilities

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
```

```
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT MMCap (#PCDATA) >

        <!ELEMENT UserPresenceItem (#PCDATA) >

        <!ATTLIST UserPresenceItem

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ExtRef PUBIDLITERAL #IMPLIED
                ExtRefChange NMTOKEN #IMPLIED >

        <!ELEMENT ClientTypeRequest (EMPTY) >

        <!ATTLIST ClientTypeRequest

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT ClientType (ClientName, ClientManufacturer, ClientVer-
sion) >

        <!ATTLIST ClientType

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT ClientName (#PCDATA) >
```

```
<!ELEMENT ClientManufacture (#PCDATA) >

<!ELEMENT ClientVersion (#PCDATA) >

<!ELEMENT ClientPresenceItem (ClientType, CliPresItem) >

<!ATTLIST ClientPresenceItem

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED
        ContentChange NMTOKEN #IMPLIED >

<!ELEMENT CliPresItem (#PCDATA) >

<!ELEMENT ExtPresence (#PCDATA) >

<!ATTLIST ExtPresence

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED
        ContentChange NMTOKEN #IMPLIED >

<!-- End of DTD -->
```

D.11 DTD for PresenceNames

The DTD for PresenceNames is the same as for Presence except that only the presence item names and attributes are included. No presence values are used.

```
<!-- Root element -->
```

```
<!ELEMENT PresenceNames ( DeviceOn | DeviceRoaming | Net-
workType | UserStatus | PreferredContact | PreferredDefaults | LocalTime |
MeasuredLocation | ConvertedLocation | StatedLocation | UserEnvironment |
StatusText | StatusImage | StatusSound | MessagingCapabilities | EmailClient |
VoiceCallCapability | MultimediaCallCapabilities | UserPresenceItem | Client-
TypeRequest | ClientType | ClientPresenceItem | ExtPresence) >

<!ATTLIST PresenceNames Version NMTOKENS #REQUIRED >

<!ELEMENT DeviceOn (EMPTY) >

<!ATTLIST DeviceOn
        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED
        DeviceName NMTOKEN #IMPLIED >

<!ELEMENT DeviceRoaming (EMPTY) >

<!ATTLIST DeviceRoaming

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED
        DeviceName NMTOKEN #IMPLIED >

<!ELEMENT NetworkType (EMPTY) >

<!ATTLIST NetworkType

        Class NMTOKEN #IMPLIED
        ID PUBIDLITERAL #IMPLIED
        Cacheability NMTOKEN #IMPLIED
        ValidityPeriod NMTOKEN #IMPLIED
```

```
                DeviceName NMTOKEN #IMPLIED >

        <!ELEMENT UserStatus (EMPTY) >

        <!ATTLIST UserStatus

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT PreferredContact (EMPTY) >

        <!ATTLIST PreferredContact

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT PreferredDefaults (EMPTY) >

        <!ATTLIST PreferredDefaults

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT LocalTime (EMPTY) >

        <!ATTLIST LocalTime

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
```

```
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT MeasuredLocation (EMPTY) >

        <!ATTLIST MeasuredLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                Accuracy NMTOKEN #IMPLIED >

        <!ELEMENT ConvertedLocation (EMPTY) >

        <!ATTLIST ConvertedLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT StatedLocation (EMPTY) >

        <!ATTLIST StatedLocation

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT UserEnvironment (EMPTY) >

        <!ATTLIST UserEnvironment

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
```

```
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT StatusText (EMPTY) >

        <!ATTLIST StatusText

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT StatusImage (EMPTY) >

        <!ATTLIST StatusImage

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ImageType NMTOKEN #IMPLIED >

        <!ELEMENT StatusSound (EMPTY) >

        <!ATTLIST StatusSound

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                SoundType NMTOKEN #IMPLIED >

        <!ELEMENT MessagingCapabilities (EMPTY) >

        <!ATTLIST MessagingCapabilities
```

```
                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT EmailClient (EMPTY) >

        <!ATTLIST EmailClientType

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT VoiceCallCapability (EMPTY) >

        <!ATTLIST VoiceCallCapability

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT MultimediaCallCapabilities (EMPTY) >

        <!ATTLIST MultimediaCallCapabilities

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT UserPresenceItem (EMPTY) >

        <!ATTLIST UserPresenceItem
```

```
                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ExtRef PUBIDLITERAL #IMPLIED
                ExtRefChange NMTOKEN #IMPLIED >

        <!ELEMENT ClientType (EMPTY) >

        <!ATTLIST ClientType

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED >

        <!ELEMENT ClientPresenceItem (EMPTY) >

        <!ATTLIST ClientPresenceItem

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ContentChange NMTOKEN #IMPLIED >

        <!ELEMENT ExtPresence (EMPTY) >

        <!ATTLIST ExtPresence

                Class NMTOKEN #IMPLIED
                ID PUBIDLITERAL #IMPLIED
                Cacheability NMTOKEN #IMPLIED
                ValidityPeriod NMTOKEN #IMPLIED
                ContentChange NMTOKEN #IMPLIED >
```

<!-- End of DTD -->

It will be evident to those skilled in the art that as technology advances, the inventive concept can be implemented in many different ways. Therefore the invention and its embodiments are not limited to the above examples but may vary within the scope and spirit of the appended claims.

CLAIMS

1. A mobile messaging system comprising at least one client device and a server, wherein

the client device comprises means for transmitting presence information as presence attributes to the server and means for receiving presence attributes from the server, said presence information being categorized by a plurality of presence attribute types identified by attribute name, and

the server comprises means for maintaining presence information based on the received presence attributes, characterized in that

the client device comprises means for adding a qualifier to a presence attribute, the qualifier comprising one or more parameters specifying the use of the attribute, and

the client device comprises means for processing a received presence attribute according to the qualifier parameters in said received attribute.

2. A system according to claim 1, characterized in that

the client device comprises means for specifying in the qualifier the presentation settings of the attribute, and

the client device comprises means for presenting the received attribute on the basis of the qualifier.

3. A system according to claim 1, characterized in that

the client device comprises means for specifying in said qualifier the application to which the attribute should be addressed, and

the client device comprises means for addressing the received attribute to the application indicated by the qualifier.

4. A system according to any one of the preceding claims, characterized in that

the server comprises means for determining on the basis of the qualifier whether to send the attribute to one or more client devices.

5. A mobile messaging system comprising at least one client device and a server, wherein

the client device comprises means for transmitting presence information as presence attributes to the server and means for receiving presence at-

tributes from the server, said presence information being categorized by a plurality of presence attribute types identified by attribute name, and

the server comprises means for maintaining presence information based on received presence attributes, characterized in that

the client device comprises means for composing a presence information attribute identified by a combination of an authorizer, an attribute name and a qualifier, the authorizer specifying the body responsible for maintaining the attribute and the qualifier specifying the use of the attribute,

the server comprises means for searching for an already stored attribute containing same identifiers as a received attribute and means for replacing said already stored attribute with said received attribute if the combination of identifiers of said received attribute is identical to that of said already stored attribute or otherwise adding said received attribute, and

the client device comprises means searching for an already stored attribute containing same identifiers as a received attribute and means for replacing said already stored attribute with said received attribute if the combination of identifiers of said received attribute is identical to that of said already stored attribute or otherwise adding said received attribute.

6.A mobile messaging system according to any one of the preceding claims, characterised in that

the presence attributes received from the client device by the server are stored in a database according to a publisher user in association with a presence group.

7. A mobile messaging system according to any one of the preceding claims, characterized in that

each presence attribute is part of an item including an attribute name element and an attribute value.

8. A mobile messaging system according to claim 7, characterized in that

said name element includes an authority string indicative of an authority responsible for keeping said name element and attribute value unique.

9. A mobile messaging system according to any one of the preceding claims, characterized in that

a presence set comprises one or more presence attributes belonging to a single publisher role of a publisher user in association with a single presence group.

10. A mobile messaging system according to claim 9, characterized in that

a user of said client device as a publisher is able to use said client device or more than one client device in more than one publisher role.

11. A mobile client device for mobile messaging system, said client device comprising

means for transmitting presence information as presence attributes to a server, said presence information being categorized by a plurality of presence attribute types identified by attribute name, characterized in that said client device further comprises

means for adding a qualifier to a presence attribute, the qualifier comprising one or more parameters specifying the use of the attribute.

12. A mobile client device for mobile messaging system, said client device comprising means for receiving presence attributes from a server, said presence information being categorized by a plurality of presence attribute types identified by attribute name

characterized in that said client device further comprises

means for adding a qualifier to a presence attribute, the qualifier comprising one or more parameters specifying the use of the attribute, and

means for processing a received presence attribute according to the qualifier parameters in said received attribute.

13. A mobile client device for mobile messaging system, said client device comprising means for transmitting presence information as presence attributes to the server, and

means for receiving presence attributes from the server, said presence information being categorized by a plurality of presence attribute types

identified by attribute name, characterized in that said client device further comprises

means for composing a presence information attribute identified by a combination of an authorizer, an attribute name and a qualifier, the authorizer specifying the body responsible for maintaining the attribute and the qualifier specifying the use of the attribute,

means for searching for an already stored attribute containing same identifiers as a received attribute, and

means for replacing said already stored attribute with said received attribute if the combination of identifiers of said received attribute is identical to that of said already stored attribute or otherwise adding said received attribute.

14. A mobile client device according to any one of the claims 11 - 13, characterized in that

each presence attribute is part of an item including an attribute name element and an attribute value.

15. A mobile client device according to claim 14, characterized in that

said name element includes an authority string indicative of an authority responsible for keeping said name element and attribute value unique.

16. A mobile client device according to any one of the claims 11 - 15, characterized in that

a presence set comprises one or more presence attributes belonging to a single publisher role of a publisher user in association with a single presence group.

17. A mobile client device according to claim 16, characterized in that

a user of said mobile client device as a publisher is able to use said client device or more than one client device in more than one publisher role.

18. A server for a mobile messaging system, said server comprising means for maintaining presence information based on received presence attributes, said presence information being categorized by a plurality of presence at-

tribute types identified by attribute name, characterized in that said server further comprises

means for searching for an already stored attribute containing same identifiers as a received attribute, and

means for replacing said already stored attribute with said received attribute if the combination of identifiers of said received attribute is identical to that of said already stored attribute or otherwise adding said received attribute.

19. A server according to claim 18, characterized in that

the presence attributes received from a client device by the server is stored in a database according to a publisher user in association with a presence group.

20. A server according to claim 18 or 19, characterized in that each presence attribute is part of an item including an attribute name element and an attribute value.

21.A server according to claim 20, characterized in that

said name element includes an authority string indicative of an authority responsible for keeping said name element and attribute value unique.

22. A server according to any one of the claims 18 - 21, characterized in that

a presence set comprises one or more presence attributes belonging to a single publisher role of a publisher user in association with a single presence group.

23. A server according to claim 22, characterized in that

a user of a client device in communication with said server acting as a publisher is able to use said client device or more than one client device in more than one publisher role.

24. A presence system, comprising:

at least one physical device having at least one presence client for enabling a presence user to interact with the system as a publisher or a subscriber; and

a server for maintaining valid values of presence sets of attributes of a publisher for access by subscribers according to associated presence groups.

25. A presence system according to claim 24, characterized in that each attribute is part of an item including an attribute name element and an attribute value.

26. A presence system according to claim 25, characterized in that said name element includes a qualifier having information related to attribute usage.

27. A presence system according to claim 26, characterized in that said name element includes an authority string indicative of an authority responsible for keeping said name element and attribute value unique.

28. A presence system according to any of the claims 24 - 27, characterized in that a presence set comprises one or more attributes belonging to a single publisher role in association with a single presence group.

29. A presence system according to claim 28, characterized in that said user interacting with said system as a publisher is able to use said presence client or more than one presence client in more than one publisher role.

30. A presence system according to claim 28, characterized in that each attribute is part of an item including an attribute name element an attribute value.

31. A presence system according to any one of the claims 24 - 30, wherein said at least one physical device is a mobile physical device.

32. A computer program embodied in a computer-readable medium for storage in a physical device, characterized in that

the program is a presence client program for enabling a presence user to interact with a presence system as a publisher of at least one presence

set of attributes for access by one or more subscribers according to an associated at least one presence group.

33. The presence client program according to claim 32, characterized in that the program enables a presence user to interact with the presence system as a subscriber to at least one set of attributes associated with a presence group in which the subscriber is a member.

34. The presence client program according to claim 32 or 33, characterized in that each attribute is part of an item including an attribute name element and an attribute value.

35. The presence client program according to claim 34, characterized in that said name element includes a qualifier having information related to attribute usage.

36. The presence client program according to claim 35, characterized in that said name element includes an authority string indicative of an authority responsible for keeping said name element and attribute value unique.

37. The presence client program according to any one of the claims 32 - 36, characterized in that a presence set comprises one or more attributes belonging to a single publisher role in association with a single presence group.

38. The presence client program according to claim 37, characterized in that

said user interacting with said system as a publisher is able to use said presence client program or more than one presence client program in more than one publisher role.

39. The presence client program according to claim 37, characterized in that

each attribute is part of an item including an attribute name element and an attribute value.

40. The presence client program according to any of the claims 32 - 39, characterized in that the physical device is a mobile physical device.

41. A physical device having a computer program embodied in a computer-readable medium stored therein, characterized in that the program is a presence client program according to any of the claims 32 - 40.

42. A data structure embodied in a computer-readable medium for storage in a physical device, characterized in that

the data structure is a presence database for storing valid values of presence sets of attributes of one or more publishers for access by subscribers according to presence groups associated with said presence sets.

43. A data structure according to claim 42, characterized in that

each attribute is part of an item including an attribute name element and an attribute value.

44. A data structure according to claim 43, characterized in that

said name element includes a qualifier having information related to attribute usage.

45. A data structure according to claim 44, characterized in that

said name element includes an authority string indicative of an authority responsible for keeping said name element and attribute value unique.

46. A data structure according to any of the claims 42 - 45, characterized in that

a presence set comprises one or more attributes belonging to a single publisher role in association with a single presence group.

47. A data structure according to any of the claims 42 - 46, characterized in that

a user interacting with said physical device as a publisher is able to use said data structure to publish more than one publisher role, each role having distinct presence sets in association with distinct presence groups.

48. A data structure according to any of the claims 42 - 47, characterized in that said physical device is a mobile device.

49. A physical device having a data structure embodied in a computer-readable medium stored therein, characterized in that the data structure is a presence data structure according to any of the claims 42 - 48.

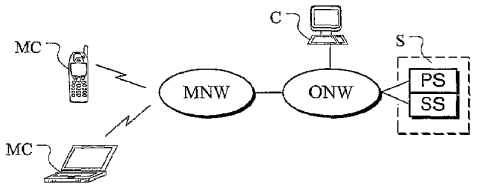


FIG. 1

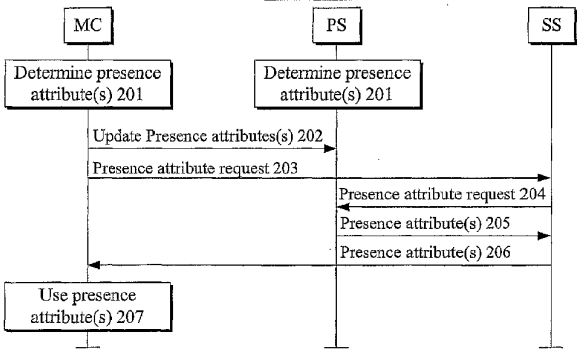


FIG. 2

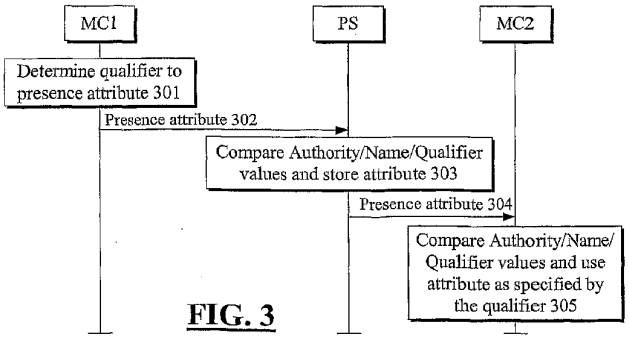


FIG. 3

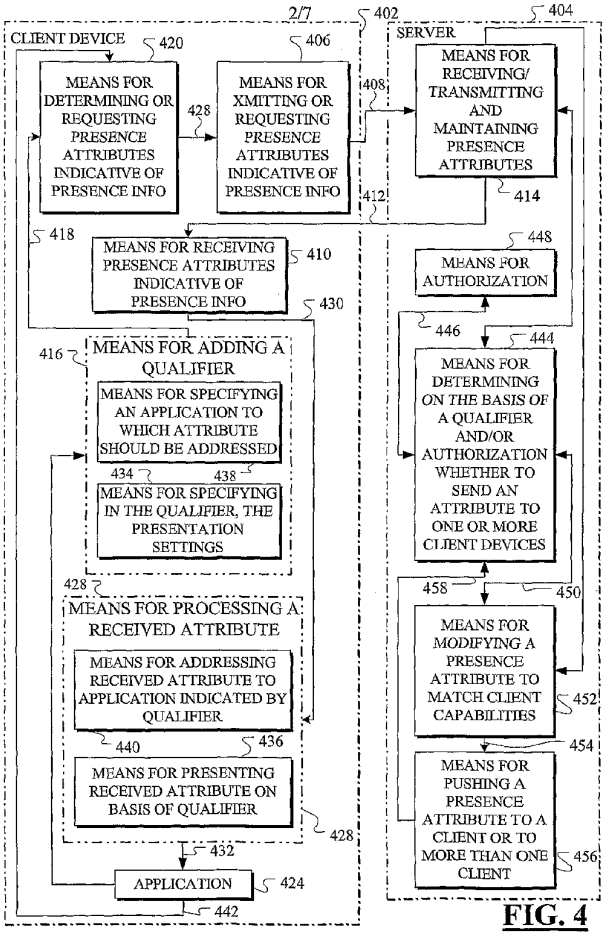

CLIENT DEVICE
420
406
MEANS FOR DETERMINING OR REQUESTING PRESENCE ATTRIBUTES INDICATIVE OF PRESENCE INFO
428
MEANS FOR XMITTING OR REQUESTING PRESENCE ATTRIBUTES INDICATIVE OF PRESENCE INFO
402
404
SERVER
408
MEANS FOR RECEIVING/ TRANSMITTING AND MAINTAINING PRESENCE ATTRIBUTES
414
412
418
MEANS FOR RECEIVING PRESENCE ATTRIBUTES INDICATIVE OF PRESENCE INFO
410
448
MEANS FOR AUTHORIZATION
430
446
444
416
MEANS FOR ADDING A QUALIFIER
MEANS FOR SPECIFYING AN APPLICATION TO WHICH ATTRIBUTE SHOULD BE ADDRESSED
434
438
MEANS FOR SPECIFYING IN THE QUALIFIER, THE PRESENTATION SETTINGS
MEANS FOR DETERMINING ON THE BASIS OF A QUALIFIER AND/OR AUTHORIZATION WHETHER TO SEND AN ATTRIBUTE TO ONE OR MORE CLIENT DEVICES
428
458
450
MEANS FOR PROCESSING A RECEIVED ATTRIBUTE
MEANS FOR ADDRESSING RECEIVED ATTRIBUTE TO APPLICATION INDICATED BY QUALIFIER
MEANS FOR MODIFYING A PRESENCE ATTRIBUTE TO MATCH CLIENT CAPABILITIES
452
440
436
454
MEANS FOR PRESENTING RECEIVED ATTRIBUTE ON BASIS OF QUALIFIER
MEANS FOR PUSHING A PRESENCE ATTRIBUTE TO A CLIENT OR TO MORE THAN ONE CLIENT
456
428
432
APPLICATION
424
442

FIG. 4

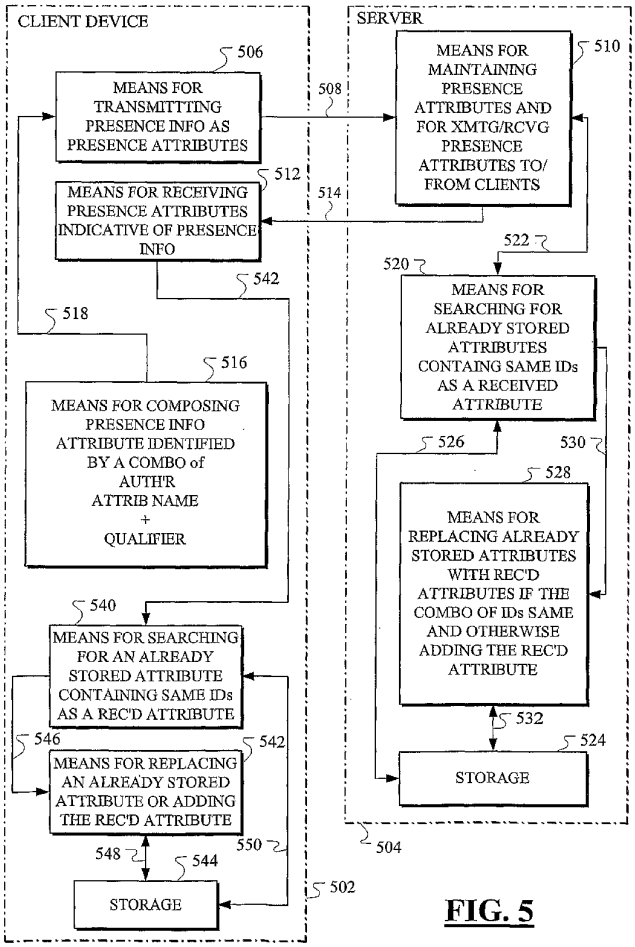

CLIENT DEVICE
SERVER
506
MEANS FOR TRANSMITTING PRESENCE INFO AS PRESENCE ATTRIBUTES
508
MEANS FOR MAINTAINING PRESENCE ATTRIBUTES AND FOR XMTG/RCVG PRESENCE ATTRIBUTES TO/ FROM CLIENTS
510
512
MEANS FOR RECEIVING PRESENCE ATTRIBUTES INDICATIVE OF PRESENCE INFO
514
522
520
MEANS FOR SEARCHING FOR ALREADY STORED ATTRIBUTES CONTAING SAME IDs AS A RECEIVED ATTRIBUTE
542
518
516
MEANS FOR COMPOSING PRESENCE INFO ATTRIBUTE IDENTIFIED BY A COMBO of AUTH'R ATTRIB NAME + QUALIFIER
526
530
528
MEANS FOR REPLACING ALREADY STORED ATTRIBUTES WITH REC'D ATTRIBUTES IF THE COMBO OF IDs SAME AND OTHERWISE ADDING THE REC'D ATTRIBUTE
540
MEANS FOR SEARCHING FOR AN ALREADY STORED ATTRIBUTE CONTAINING SAME IDs AS A REC'D ATTRIBUTE
532
524
546
542
MEANS FOR REPLACING AN ALREADY STORED ATTRIBUTE OR ADDING THE REC'D ATTRIBUTE
STORAGE
548
550
544
504
502
STORAGE

FIG. 5

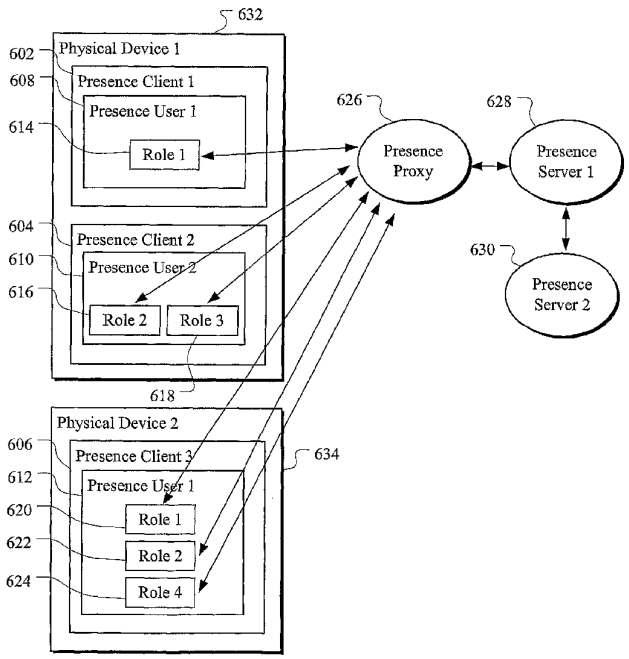

632
Physical Device 1
602
Presence Client 1
608
Presence User 1
614
Role 1
626
Presence Proxy
628
Presence Server 1
604
Presence Client 2
610
Presence User 2
616
Role 2
Role 3
630
Presence Server 2
618
Physical Device 2
606
Presence Client 3
634
612
Presence User 1
620
Role 1
622
Role 2
624
Role 4

FIG. 6

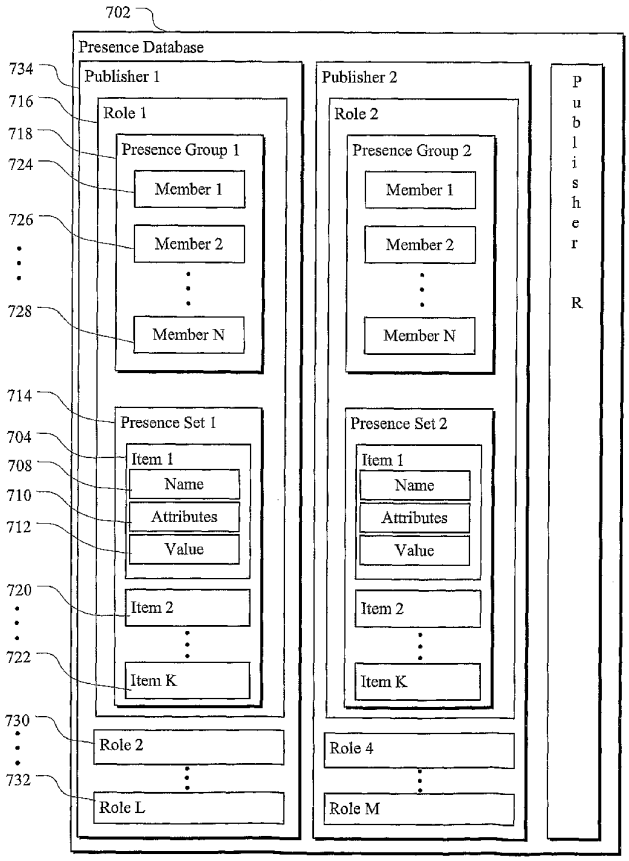

702
Presence Database
734
Publisher 1
716
Role 1
718
Presence Group 1
724
Member 1
726
Member 2
728
Member N
714
Presence Set 1
704
Item 1
708
Name
710
Attributes
712
Value
720
Item 2
722
Item K
730
Role 2
732
Role L
Publisher 2
Role 2
Presence Group 2
Member 1
Member 2
Member N
Presence Set 2
Item 1
Name
Attributes
Value
Item 2
Item K
Role 4
Role M
Publisher R

FIG. 7

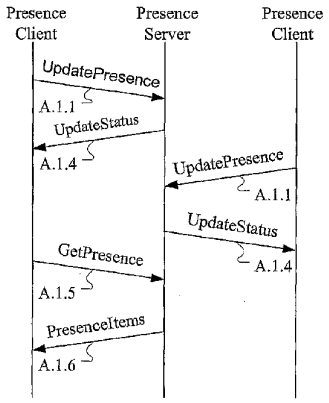

Presence Client
Presence Server
Presence Client
UpdatePresence
A.1.1
UpdateStatus
A.1.4
UpdatePresence
A.1.1
UpdateStatus
A.1.4
GetPresence
A.1.5
PresenceItems
A.1.6

FIG. 8

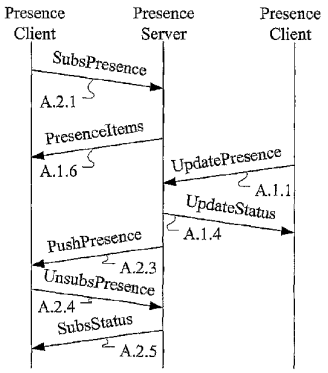

Presence Client
Presence Server
Presence Client
SubsPresence
A.2.1
PresenceItems
A.1.6
UpdatePresence
A.1.1
UpdateStatus
A.1.4
PushPresence
A.2.3
UnsubsPresence
A.2.4
SubsStatus
A.2.5

FIG. 9

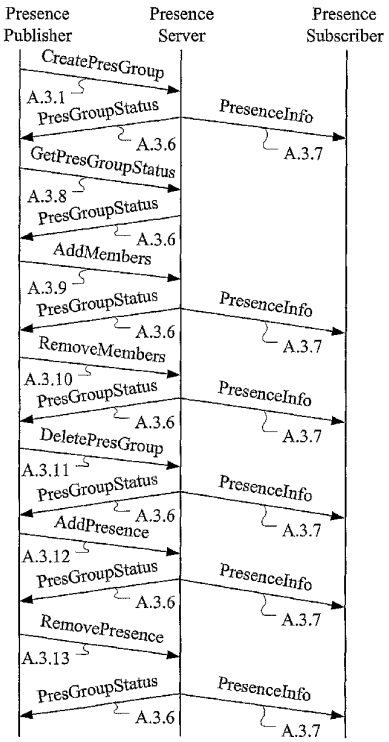

Presence Publisher
Presence Server
Presence Subscriber
CreatePresGroup
A.3.1
PresGroupStatus
A.3.6
PresenceInfo
A.3.7
GetPresGroupStatus
A.3.8
PresGroupStatus
A.3.6
AddMembers
A.3.9
PresGroupStatus
A.3.6
PresenceInfo
A.3.7
RemoveMembers
A.3.10
PresGroupStatus
A.3.6
PresenceInfo
A.3.7
DeletePresGroup
A.3.11
PresGroupStatus
A.3.6
PresenceInfo
A.3.7
AddPresence
A.3.12
PresGroupStatus
A.3.6
PresenceInfo
A.3.7
RemovePresence
A.3.13
PresGroupStatus
A.3.6
PresenceInfo
A.3.7

FIG. 10

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI 02/00403

A. CLASSIFICATION OF SUBJECT MATTER

IPC7: H04Q 7/22, H04Q 7/38
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC7: H04Q

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 0130057 A2 (COMVERSE NETWORK SYSTEMS, LTD.), 26 April 2001 (26.04.01), page 1, line 16 - line 19; page 23, line 11 - line 15, claim 14, abstract --	1-49
P,A	EP 1139608 A2 (LUCENT TECHNOLOGIES INC.), 4 October 2001 (04.10.01), column 2, line 29 - line 49, abstract --	1-49
P,A	EP 1176840 A1 (MICROSOFT CORPORATION), 30 January 2002 (30.01.02), column 3, line 12 - column 4, line 1 --	1-49

[X] Further documents are listed in the continuation of Box C. [X] See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
8 August 2002	15 -08- 2002
Name and mailing address of the ISA/ Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86	Authorized officer Malin Söderman/MN Telephone No. +46 8 782 25 00

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI 02/00403

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
P,A	US 6301609 B1 (MURALI ARAVAMUDAN ET AL), 9 October 2001 (09.10.01), column 2, line 25 - line 48	1-49
	--	
P,A	WO 0191486 A1 (BOLT MEDIA LTD.), 29 November 2001 (29.11.01), page 1, line 46 - page 2, line 12; page 3, line 1 - line 19	1-49
	--	
P,A	US 2001027111 A1 (SHINJI MOTEGI ET AL), 4 October 2001 (04.10.01), claims 1-4, abstract	1-49
	--	

Form PCT/ISA/210 (continuation of second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT
Information on patent family members

06/07/02

International application No.
PCT/FI 02/00403

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 0130057 A2	26/04/01	AU 1049301 A	30/04/01
		GB 0209514 D	00/00/00
EP 1139608 A2	04/10/01	AU 3131701 A	04/10/01
		BR 0101149 A	30/10/01
		CN 1319957 A	31/10/01
		JP 2002010328 A	11/01/02
EP 1176840 A1	30/01/02	NONE	
US 6301609 B1	09/10/01	US 2001051162 A	13/12/01
WO 0191486 A1	29/11/01	AU 5857801 A	03/12/01
		GB 0012351 D	00/00/00
		GB 2367451 A	03/04/02
US 2001027111 A1	04/10/01	JP 2001346267 A	14/12/01

Form PCT/ISA/210 (patent family annex) (July 1998)

フロントページの続き

(81)指定国 AP(GH,GM,KE,LS,MW,MZ,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,CH,CY,DE,DK,ES,FI,FR,GB,GR,IE,IT,LU,MC,NL,PT,SE,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DZ,EC,EE,ES,FI,GB,GD,GE,GH,GM,HR,HU,ID,IL,IN,IS,JP,KE,KG,KP,KR,KZ,LC,LK,LR,LS,LT,LU,LV,MA,MD,MG,MK,MN,MW,MX,MZ,NO,NZ,OM,PH,PL,PT,RO,RU,SD,SE,SG,SI,SK,SL,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VN,YU,ZA,ZM,ZW

(72)発明者 サロマキ アリ
フィンランド国, エフイーエン－３３７１０ タンペレ, オヤバイニオンカトゥ １０ デー ２８