



(19) **United States**

(12) **Patent Application Publication**

**Omar et al.**

(10) **Pub. No.: US 2004/0199665 A1**

(43) **Pub. Date: Oct. 7, 2004**

(54) **SYSTEM AND METHOD FOR PUSHING DATA FROM AN INFORMATION SOURCE TO A MOBILE COMMUNICATION DEVICE INCLUDING TRANSCODING OF THE DATA**

filed on Oct. 9, 2001. Provisional application No. 60/330,604, filed on Oct. 25, 2001. Provisional application No. 60/340,839, filed on Dec. 19, 2001.

**Publication Classification**

(76) Inventors: **Salim H. Omar**, Waterloo (CA);  
**Russell N. Owen**, Waterloo (CA);  
**Herbert A. Little**, Waterloo (CA);  
**Tomasz K. Rybak**, Waterloo (CA);  
**Michael S. Brown**, Waterloo (CA);  
**David P. Yach**, Waterloo (CA)

(51) **Int. Cl.<sup>7</sup>** ..... **G06F 15/173**  
(52) **U.S. Cl.** ..... **709/238; 709/224**

(57) **ABSTRACT**

Correspondence Address:

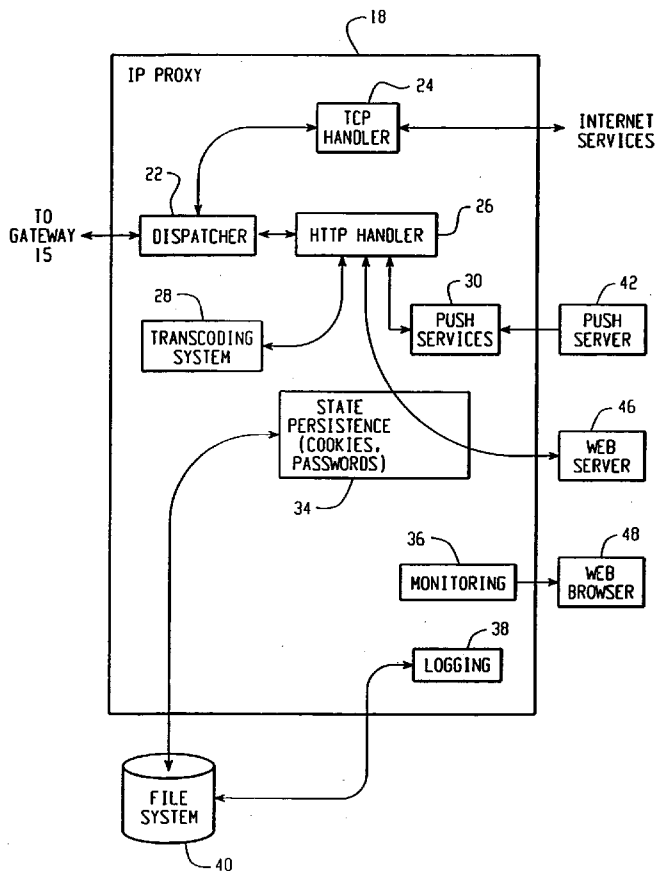
**Jones Day**  
**North Point**  
**901 Lakeside Avenue**  
**Cleveland, OH 44114 (US)**

A system for pushing information content from an information source to a mobile communication device over a network includes a transcoding system and a first network device. The transcoding system includes a plurality of transcoders, each transcoder operable to transcode the information content from a respective input content type into a respective output content type. The first network device is in communication with the transcoding system, and includes a push module. The push module is operable to receive a connection request from the information source. The connection request includes an identifier associated with the mobile communication device. The push module is further operable to select a corresponding connection handler that is operable to select one or more transcoders from the plurality of transcoders to transcode the information content.

- (21) Appl. No.: **10/483,116**
- (22) PCT Filed: **Jul. 12, 2002**
- (86) PCT No.: **PCT/CA02/01074**

**Related U.S. Application Data**

- (60) Provisional application No. 60/305,044, filed on Jul. 12, 2001. Provisional application No. 60/327,752,



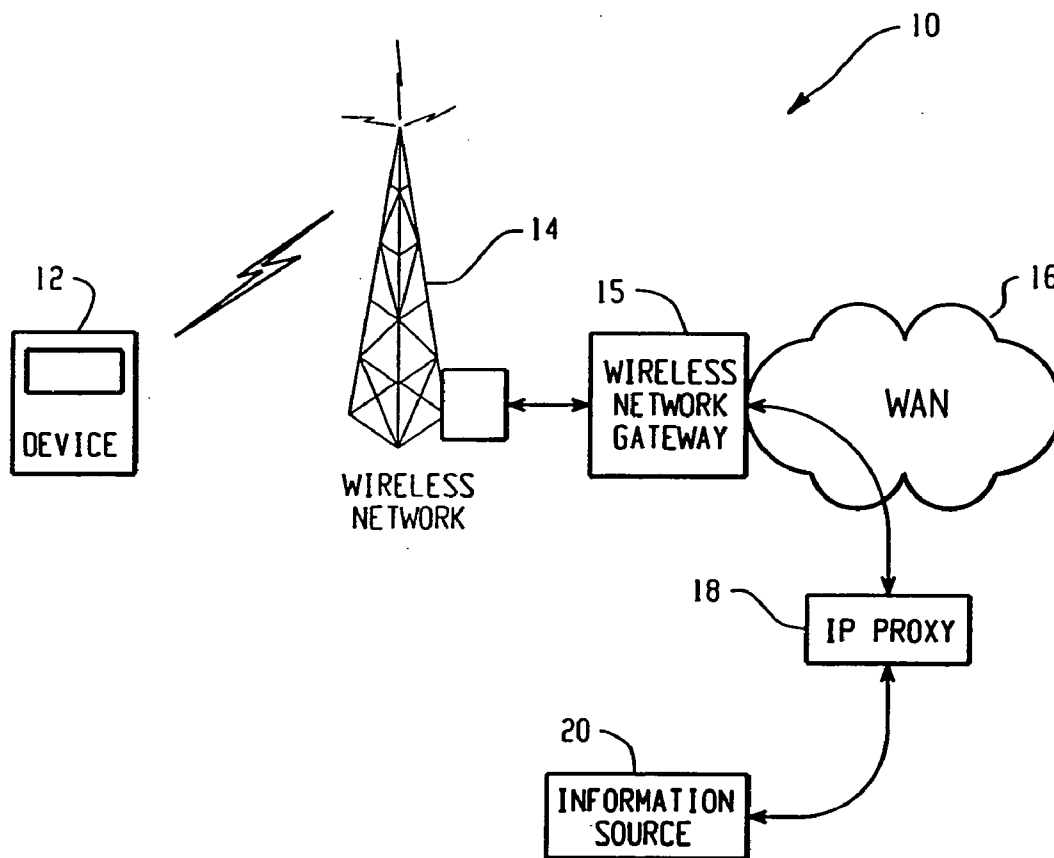


Fig. 1

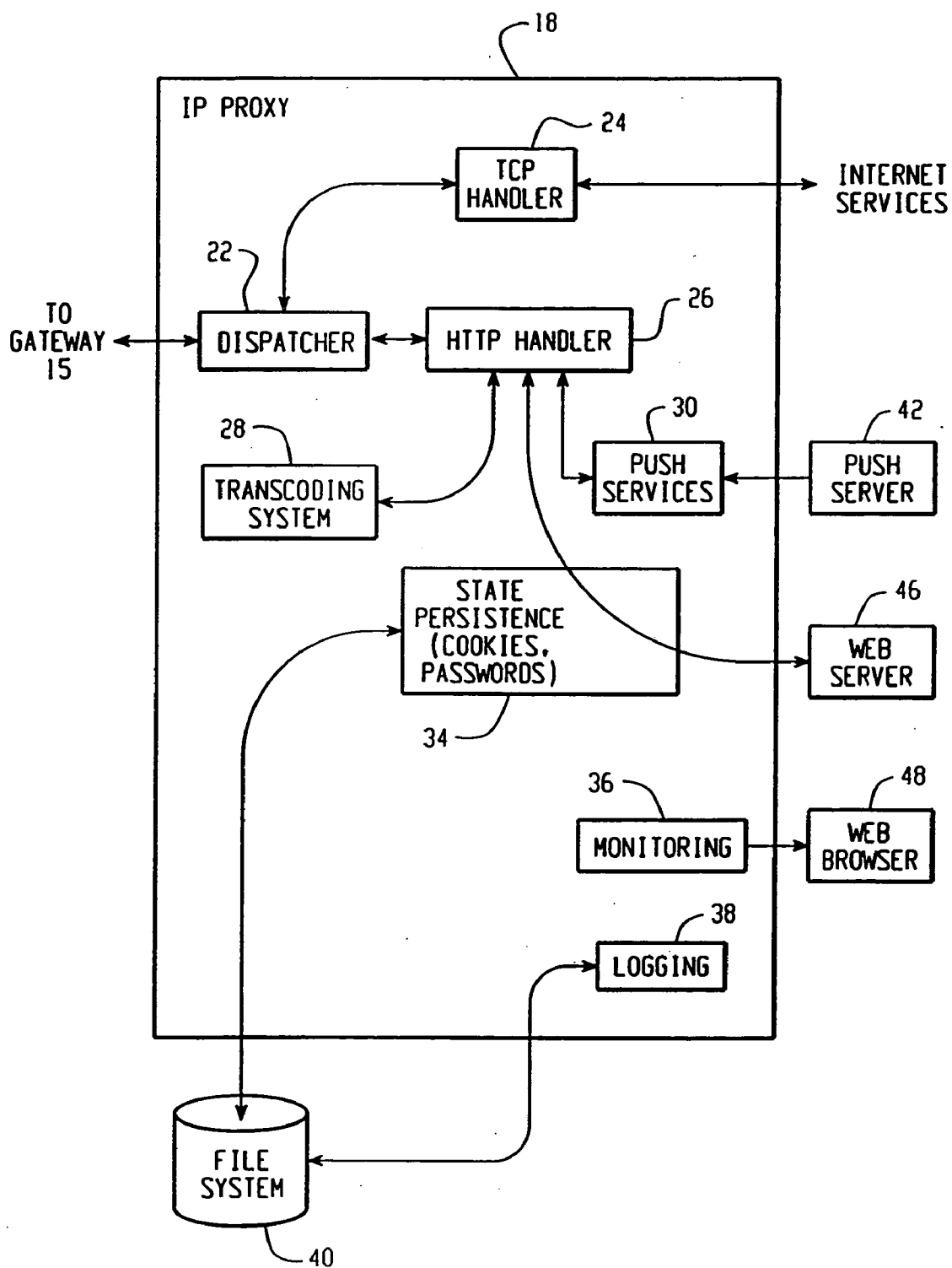


Fig. 2

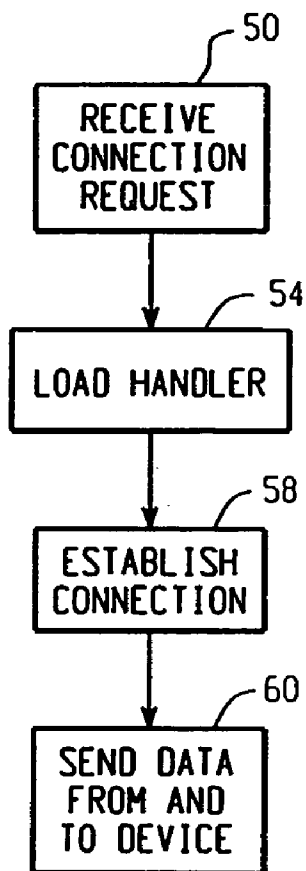


Fig. 3

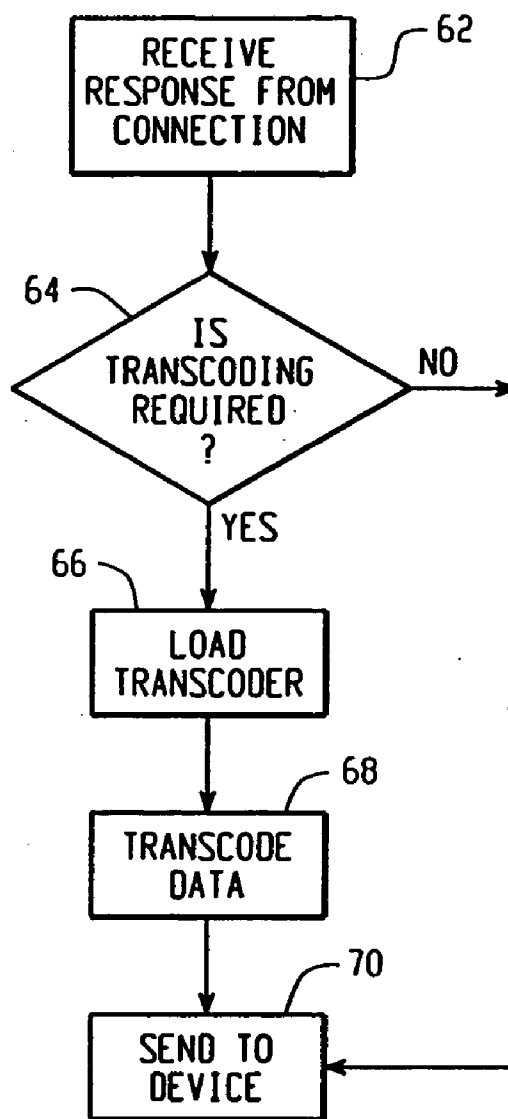


Fig. 4

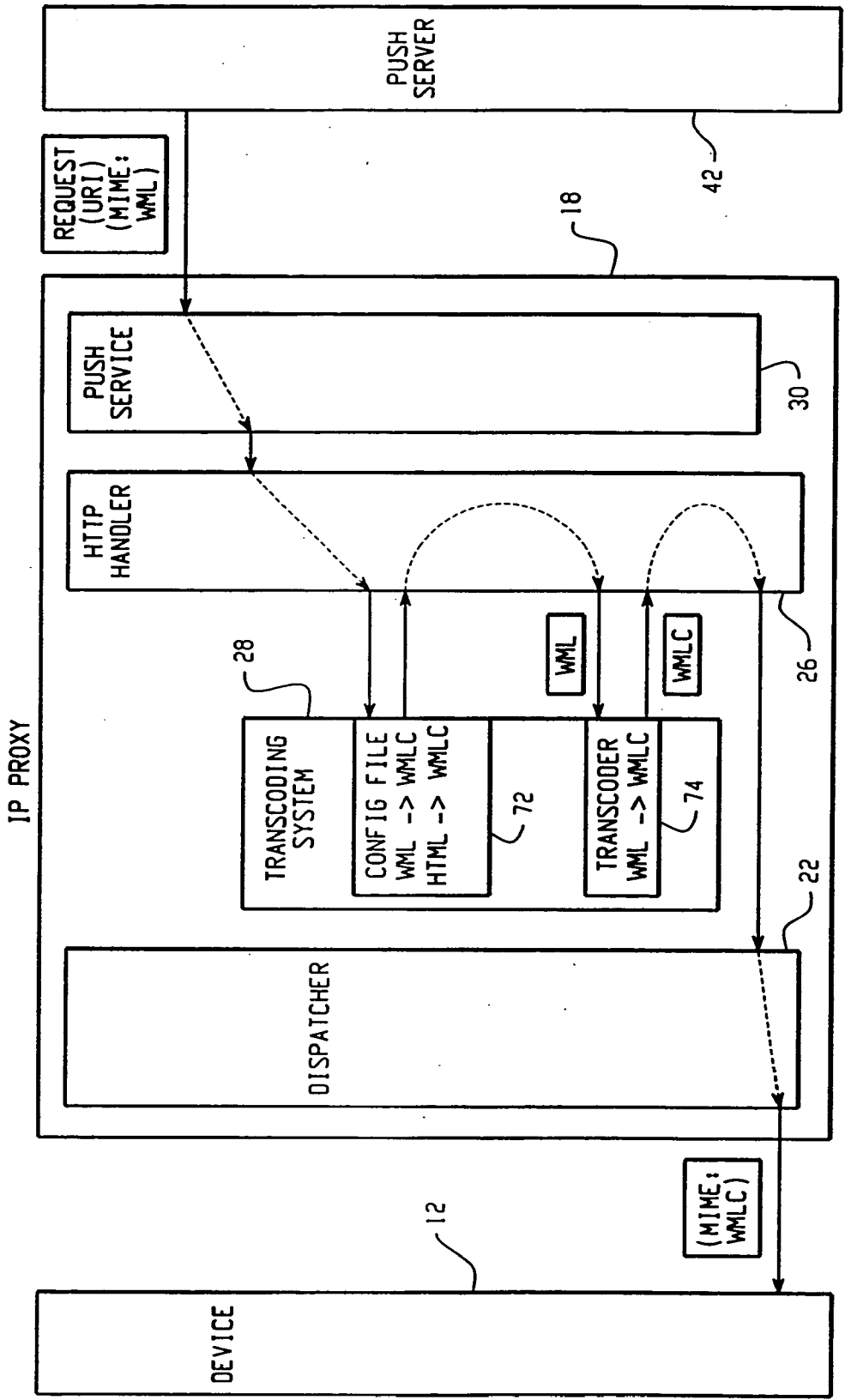


Fig. 5

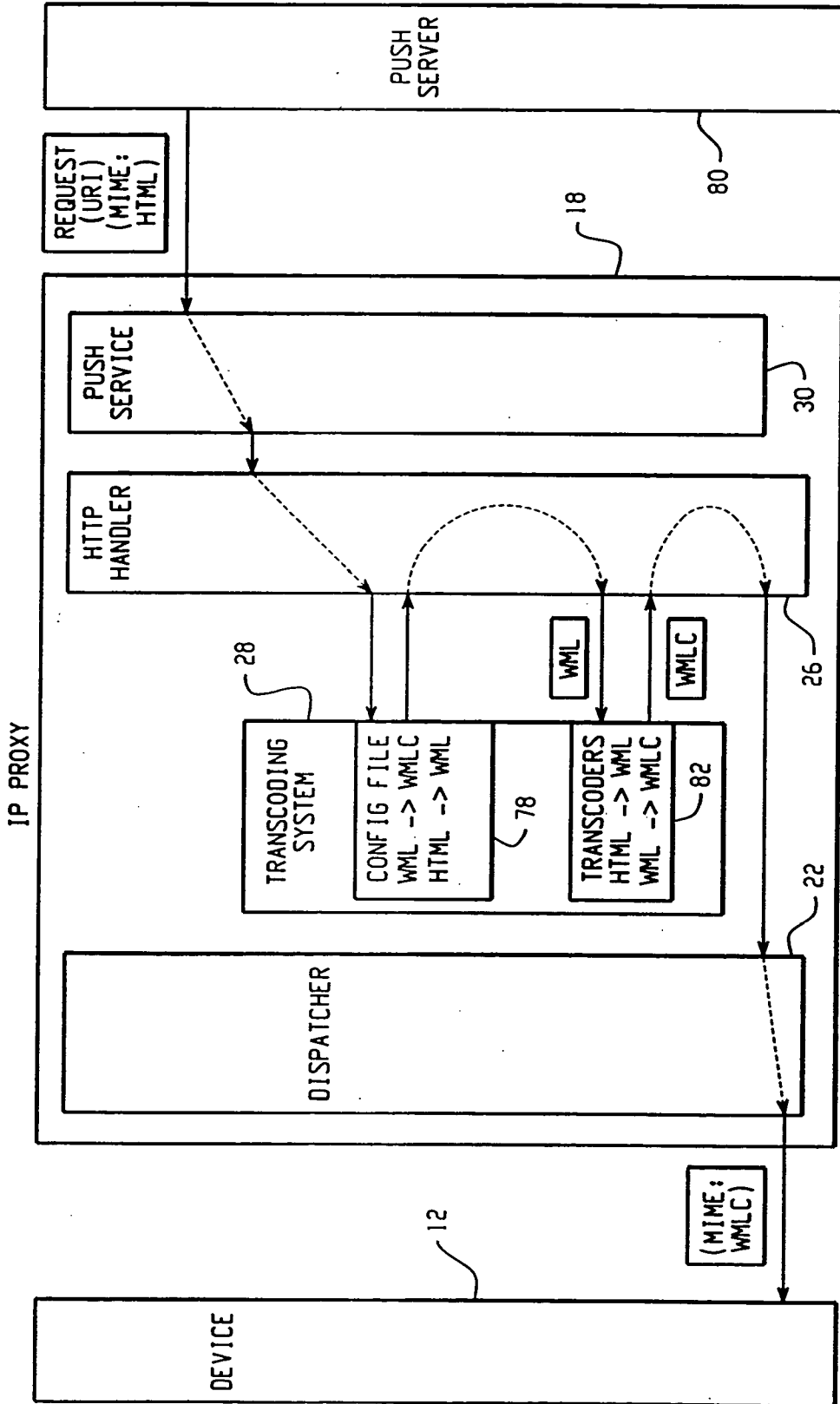


Fig. 6

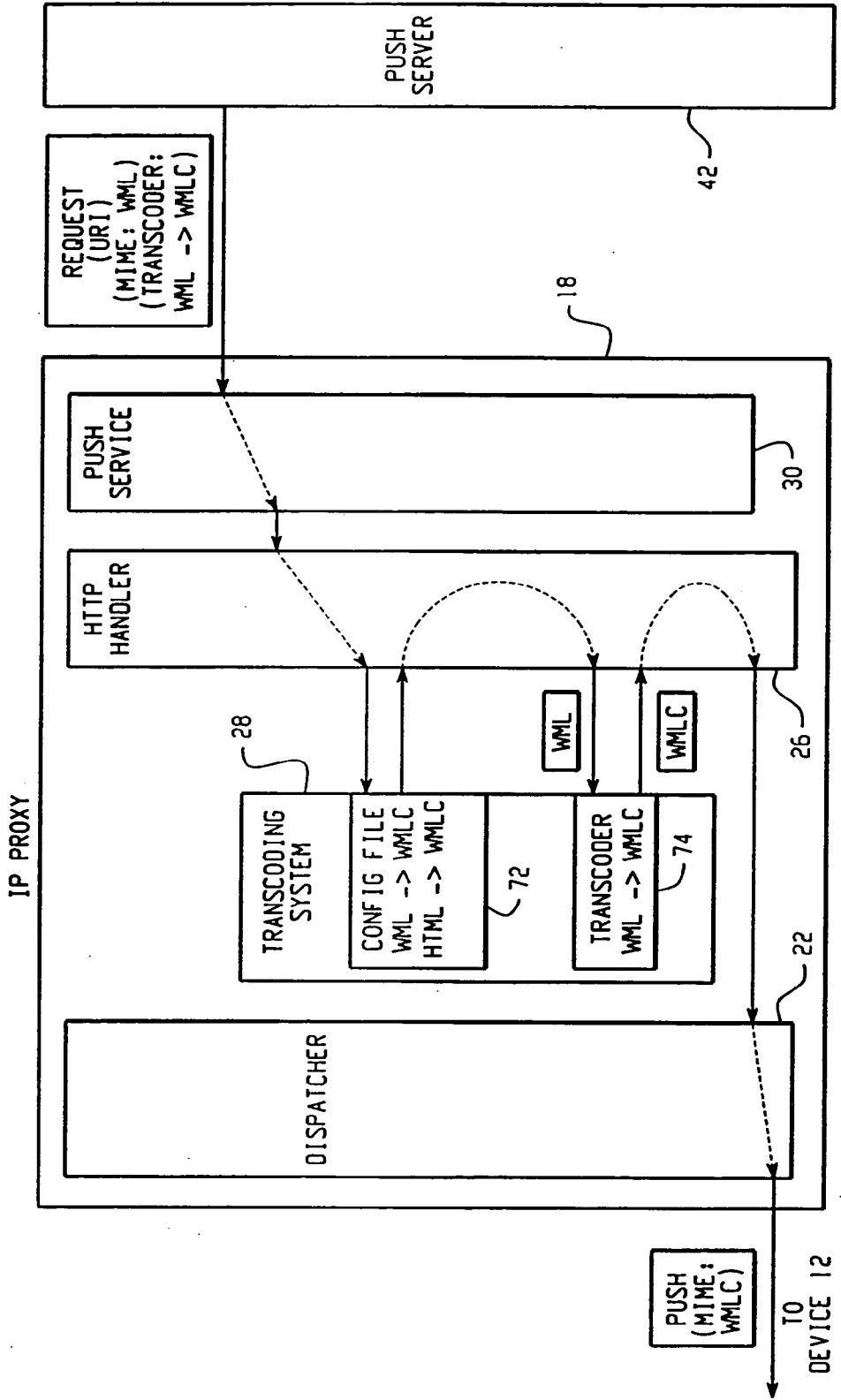


Fig. 7

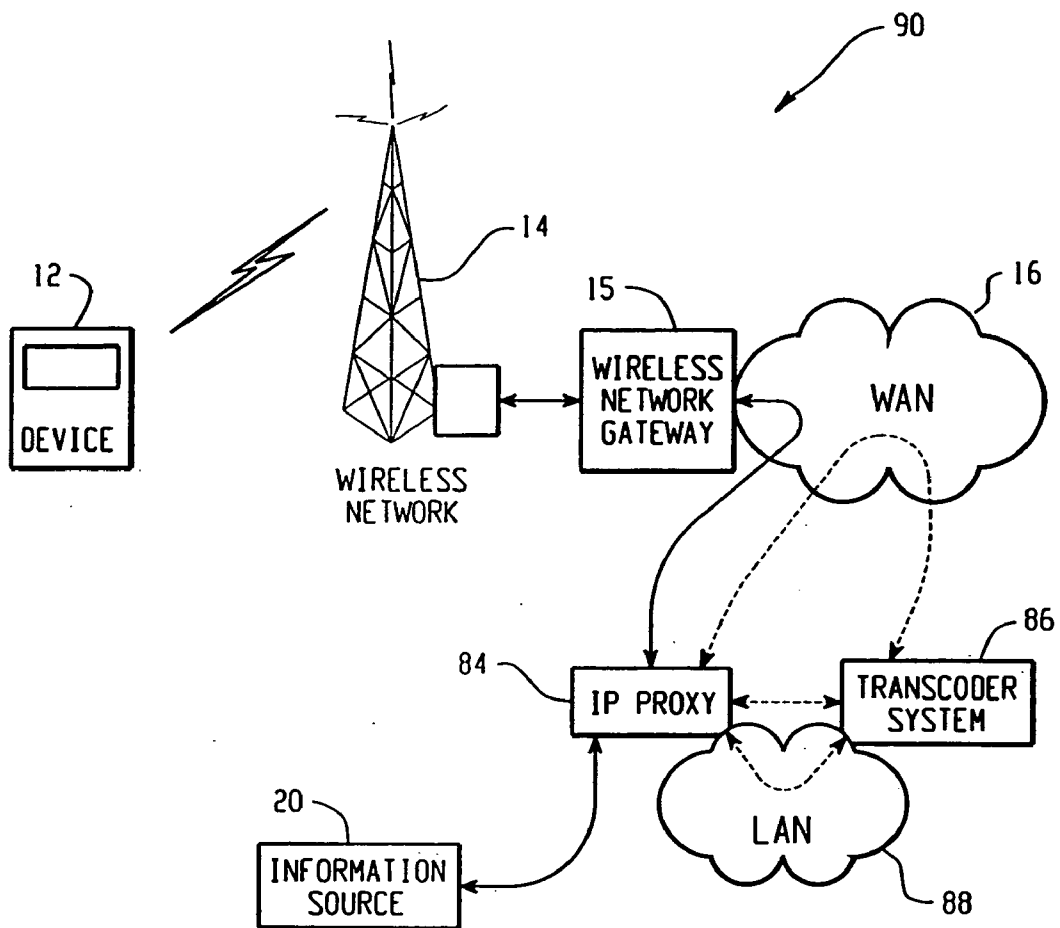


Fig. 8



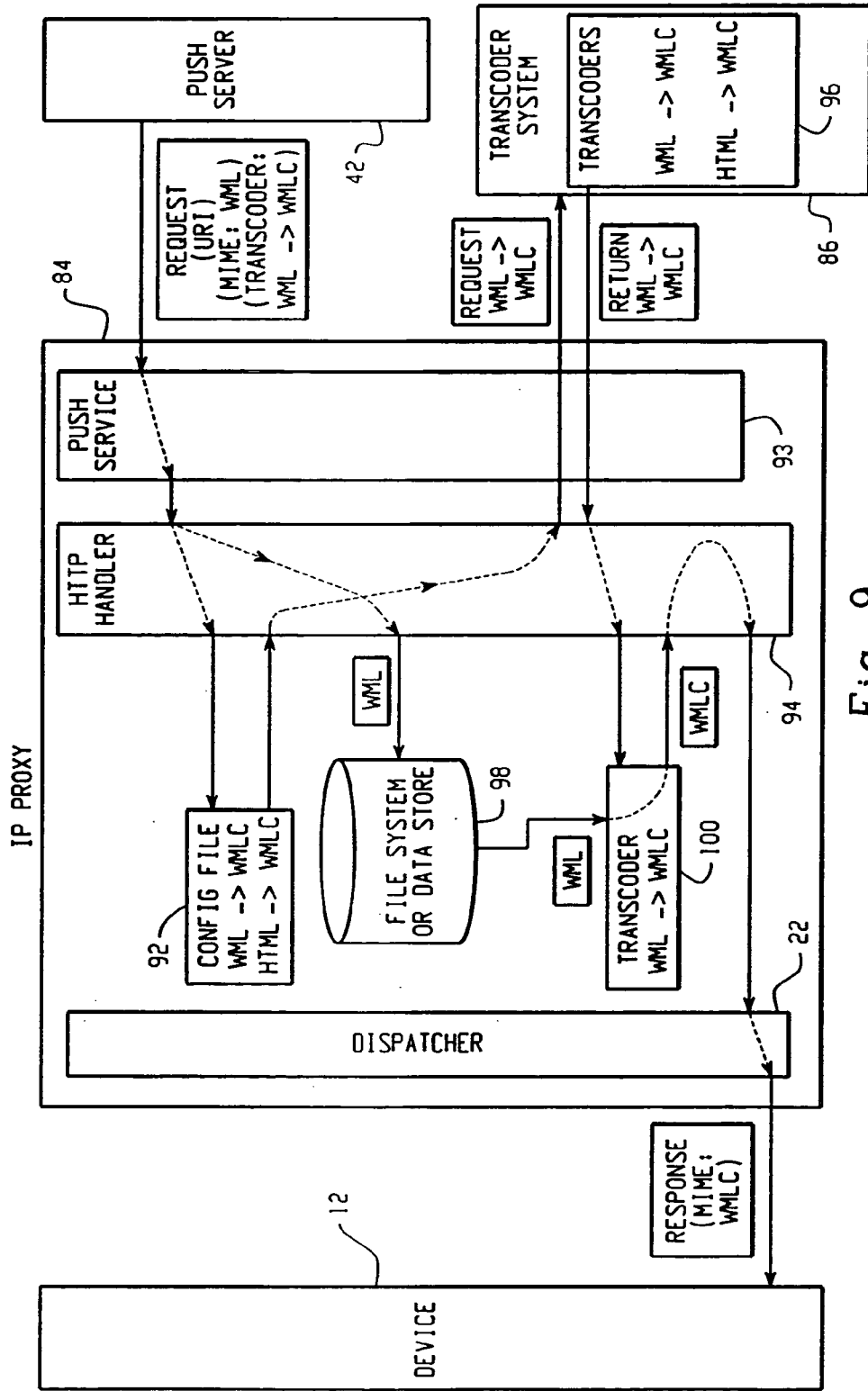


Fig. 9

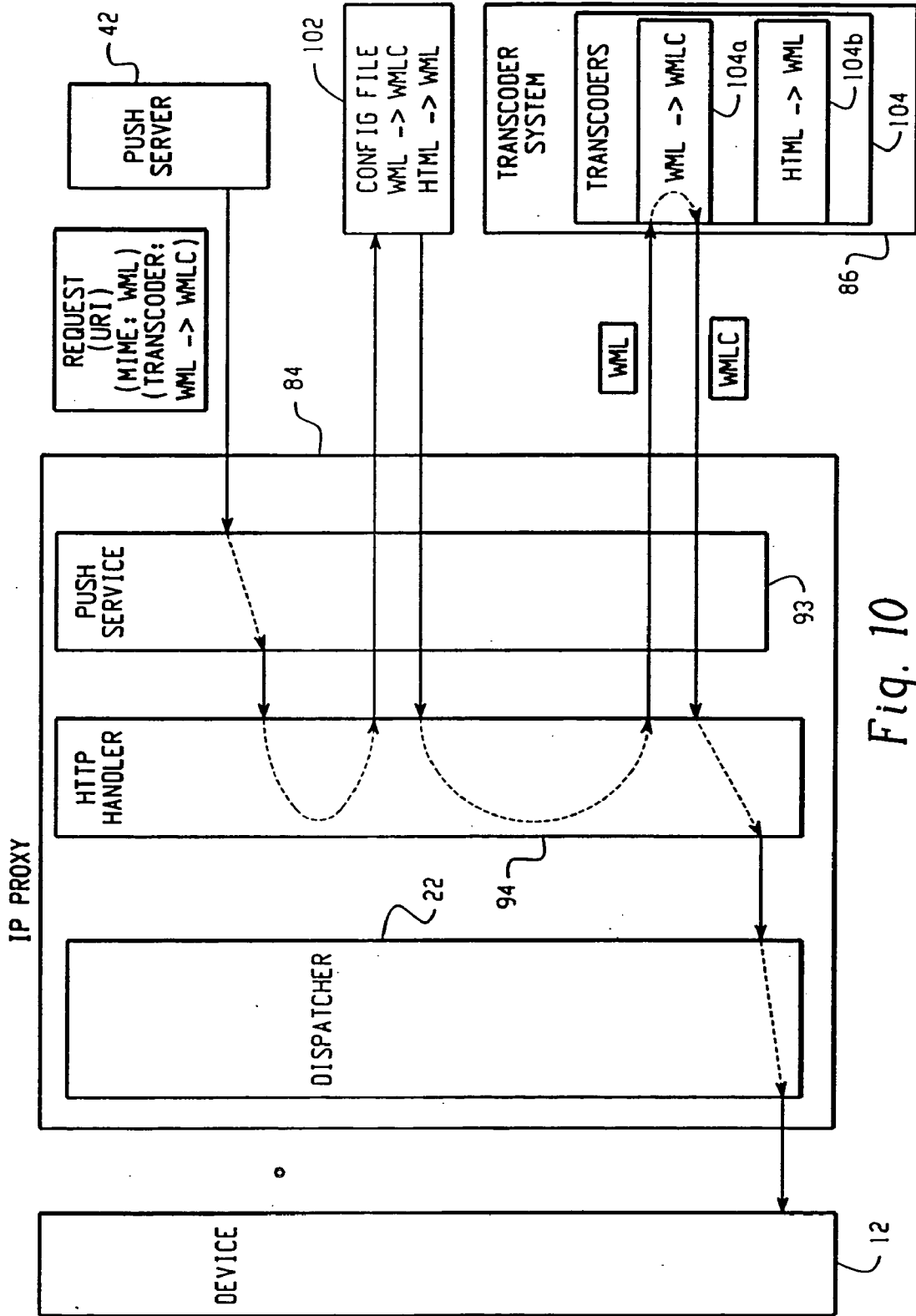


Fig. 10

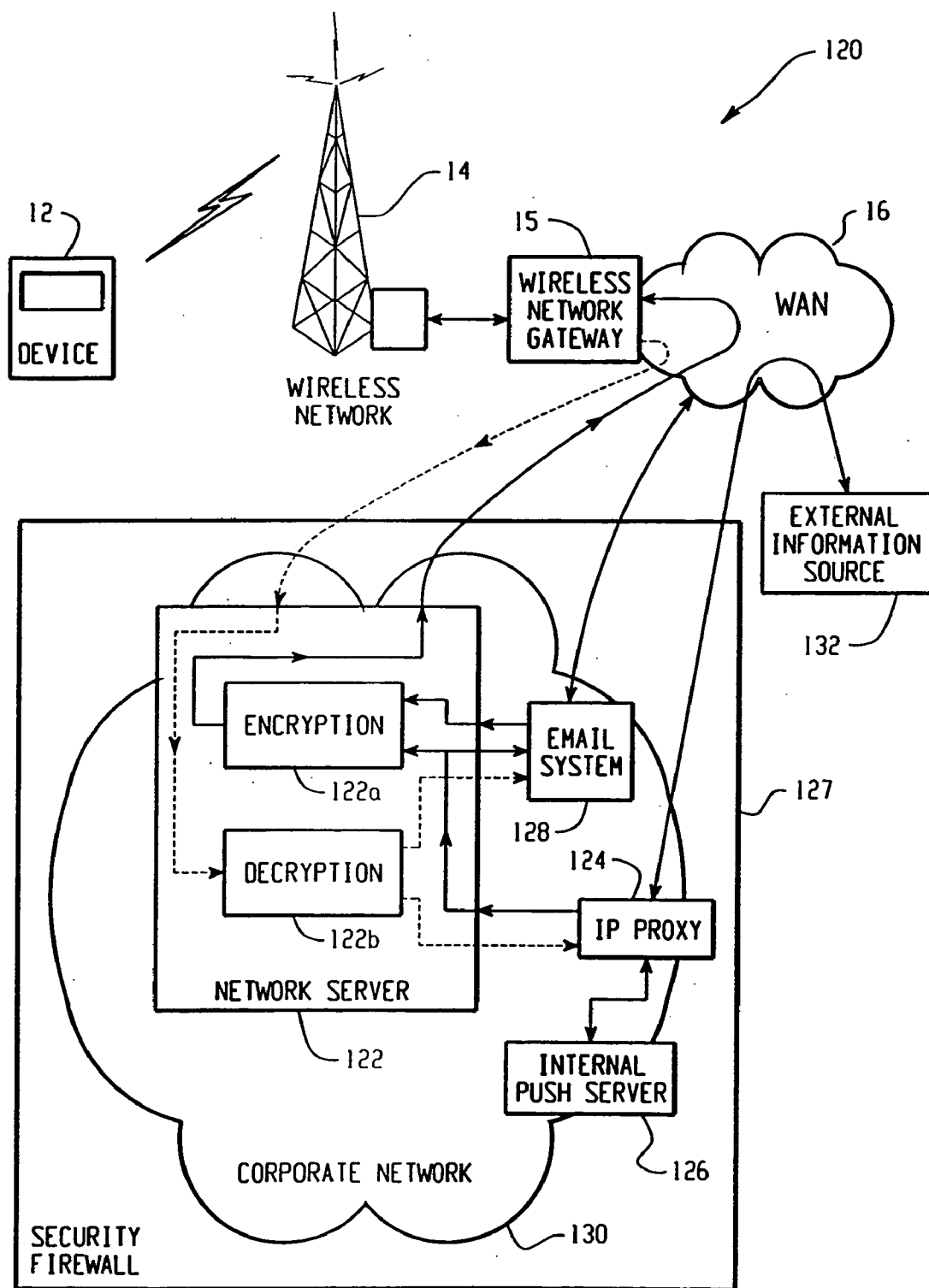


Fig. 11

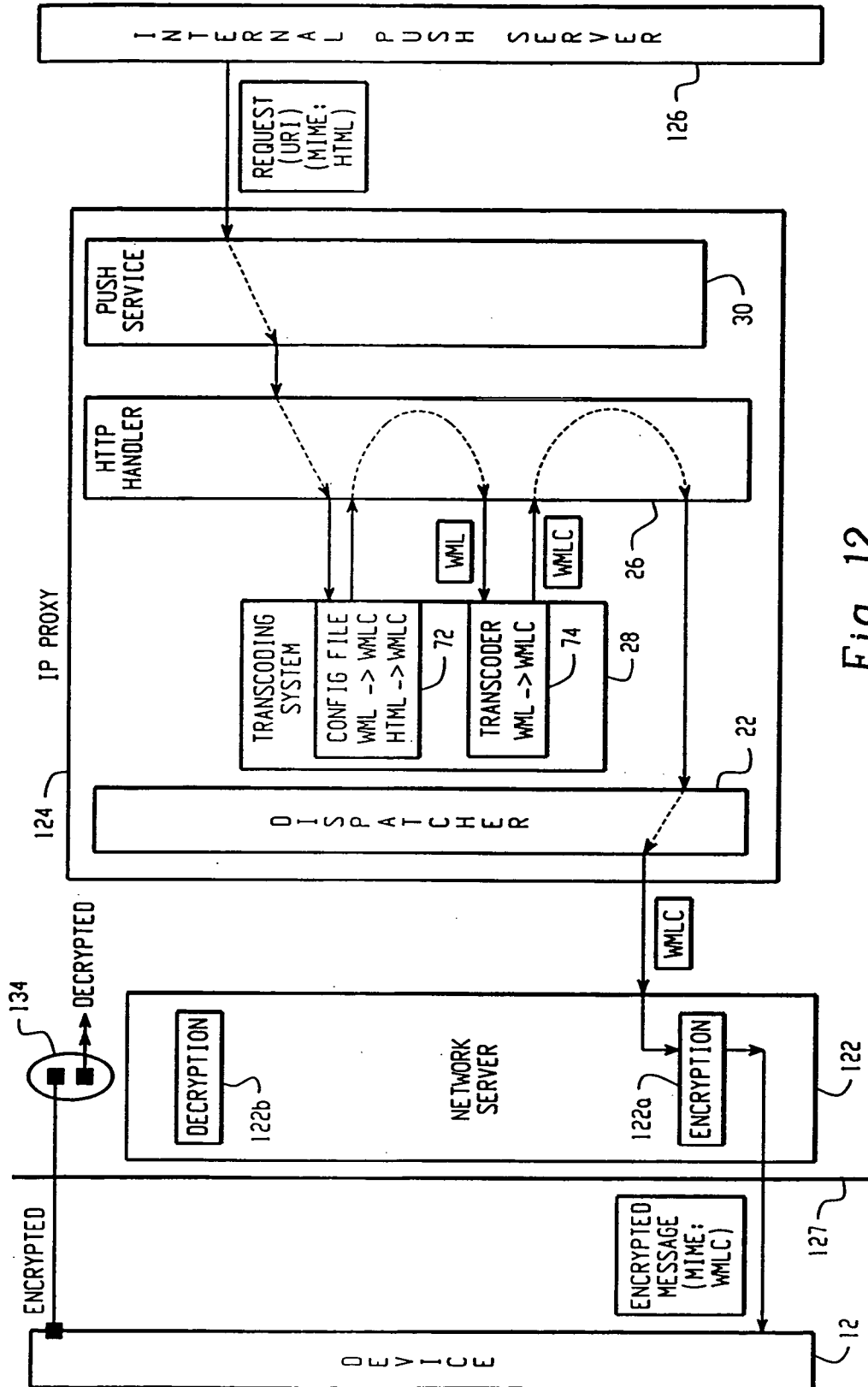


Fig. 12

**SYSTEM AND METHOD FOR PUSHING DATA FROM AN INFORMATION SOURCE TO A MOBILE COMMUNICATION DEVICE INCLUDING TRANSCODING OF THE DATA**

**SUMMARY**

[0001] This application claims benefit of the following U.S. Provisional Applications: Serial No. 60/305,044, entitled "System And Method For Providing Remote Data Access For A Mobile Communication Device" and filed on Jul. 12, 2001; Serial No. 60/327,752, entitled "System and Method For Providing Remote Data Access To A Mobile Communication Device" and filed Oct. 9, 2001; Serial No. 60/330,604, entitled "System And Method For Providing Remote Data Access And Transcoding For A Mobile Communication Device" and filed Oct. 25, 2001; and Serial No. 60/340,839, entitled "System And Method For Pushing Data From An Information Source To A Mobile Communication Device" and filed Dec. 19, 2001. The complete disclosures of all of the above-identified provisional applications are hereby incorporated into this application by reference.

[0009] The instant application describes a system and method for pushing information from an information source to mobile communication devices.

[0010] The systems and methods described herein provide for pushing of any of a plurality of types and formats of information to mobile communication devices. Particular information translation operations may be selected by a mobile communication device, an information source or an intermediate data server system and performed on an information source side of a mobile communication system. This not only reduces the complexity of device processing operations and any device hardware and software components associated with such operations, but also provides for customized device information formats.

**CROSS REFERENCE TO RELATED APPLICATIONS**

[0011] In one embodiment, a system for pushing information content from an information source to a mobile communication device over a network includes a transcoding system and a first network device. The transcoding system includes a plurality of transcoders, each transcoder operable to transcode the information content from a respective input content type into a respective output content type. The first network device is in communication with the transcoding system, and includes a push module. The push module is operable to receive a connection request from the information source. The connection request includes an identifier associated with the mobile communication device. The push module is further operable to select a corresponding connection handler that is operable to select one or more transcoders from the plurality of transcoders to transcode the information content.

[0002] This application is also related to the following co-pending Non-Provisional Applications: Ser. No. \_\_\_\_\_, entitled "System And Method For Providing Remote Data Access For A Mobile Communication Device" and filed on \_\_\_\_\_, 2002; and Ser. No. \_\_\_\_\_, entitled "System And Method For Providing Remote Data Access And Transcoding For A Mobile Communication Device" and filed on \_\_\_\_\_, 2002, the complete disclosures of which are hereby incorporated into this application by reference.

**BACKGROUND**

**BRIEF DESCRIPTION OF THE DRAWINGS**

[0003] 1. Field of the Invention

[0012] FIG. 1 is a general block diagram of a communication system that provides for pushing data from an information source to a mobile communication device.

[0004] This invention relates generally to mobile communications, and in particular to pushing information to mobile communication devices.

[0013] FIG. 2 is a more detailed block diagram of the system shown in FIG. 1.

[0005] 2. Description of the State of the Art

[0014] FIG. 3 is a flow chart representing general connection handler-related operations in an IP system.

[0006] Known solutions for providing information to mobile communication devices tend to be relatively limited. For example, Wireless Application Protocol (WAP) browsers for mobile devices typically provide access only to information associated with WAP-compliant sources and when such information is requested by a user. Although other known and similar products may allow a mobile device user to access further information sources, such products generally do not make efficient use of mobile communication network resources, particularly wireless communication links, since some sort of information request must be must generally be made before every transfer of information.

[0015] FIG. 4 is a flow chart of connection handler data processing operations.

[0016] FIG. 5 is a signal flow diagram of an example information push operation.

[0007] Furthermore, most known data access systems and methods are not suited to provide truly secure access to confidential information stored on private networks, such as corporate information located on a data store behind a security firewall.

[0017] FIG. 6 is a signal flow diagram showing multiple or "chained" transcoding operations for an HTTP-based push operation.

[0018] FIG. 7 is a signal flow diagram of an example of push server-controlled transcoder selection for an HTTP-based push operation.

[0008] Therefore, there remains a need for a system and method for pushing information from an information source to a mobile communication device.

[0019] FIG. 8 is a general block diagram of a communication system with an external transcoder system.

[0020] FIG. 9 is a signal flow diagram illustrating an HTTP-based push operation with an external transcoder system such as shown in FIG. 8.

[0021] FIG. 10 shows a further signal flow diagram for an external transcoder system.

[0022] FIG. 11 is a block diagram showing an IP Proxy system implemented in a secure network.

[0023] FIG. 12 is a signal flow diagram illustrating a corporate data push operation.

#### DETAILED DESCRIPTION

##### [0024] General System Description

[0025] FIG. 1 is a general block diagram of a communication system that provides for pushing of information from a remote information source 20 to a wireless mobile communication device 12. In FIG. 1, the system 10 includes the mobile device 12, a wireless network 14, a wireless network gateway 15, a wide area network (WAN) 16, an Internet Protocol (IP) Proxy system 18, and an information source 20. Although an IP Proxy system 18 is shown in the illustrative example system of FIG. 1, proxy systems for protocols other than IP may be implemented in accordance with the present invention. Protocols at other levels within the Open Systems Interconnection (OSI) model can also be proxied using this system. Such other protocols include but are not limited to HTTP and TCP.

[0026] The mobile device 12 may be any mobile communication device adapted to operate within a wireless communication network 14, and is preferably a two-way communication device. The mobile device 12 may also have voice and data communication capabilities. Depending on the functionality provided by the mobile device 12, the mobile device 12 may be referred to as a data messaging device, a two-way pager, a cellular telephone with data messaging capabilities, a wireless Internet appliance or a data communication device (with or without telephony capabilities), but is referred to herein primarily as a "mobile device". As will be apparent to those skilled in the field of communications, the particular design of a communication subsystem within the mobile device 12 will be dependent upon the communication network 14 in which the mobile device 12 is intended to operate. For example, a mobile device 12 destined for a North American market may include a communication subsystem designed to operate within the Mobitex™ mobile communication system or DataTAC™ mobile communication system, whereas a mobile device 12 intended for use in Europe may incorporate a General Packet Radio Service (GPRS) communication subsystem. Those skilled in the art will also appreciate that other types of mobile devices and networks are also contemplated. The inventive systems and methods described herein may be implemented in conjunction with virtually any wireless network 14.

[0027] The gateway 15 shown in FIG. 1 provides an interface between the wireless network 14 and a WAN 16, which may, for example, be the Internet. Such functions as mobile device addressing, conversion of data between WAN protocols and wireless network protocols, storing and forwarding data to and from the mobile device 12, and other interface functions may be performed by the gateway 15.

[0028] It is also possible that an IP Proxy system 18 could be hosted by a network carrier/operator associated with the wireless network 14. In this case, the connection between the IP Proxy system 18 and the gateway 15 would use a private network of the carrier instead of the WAN 16. The WAN 16 could then be used to communicate between the IP Proxy system 18 and the information source 20.

[0029] The IP Proxy system 18 is a system that effectively provides the information source 20 with access to the mobile device 12, and is described in further detail below. Through the IP Proxy system 18, any information source 20, such as an Internet or web server, that can communicate with the IP Proxy system 18 may push information to a mobile device 12. The information source 20 therefore requires no special applications or protocol support for wireless network communications, since it communicates with the IP Proxy system 18 and not directly with the mobile device 12. Although shown in FIG. 1 as a direct connection, the IP Proxy system 18 and information source 20 may possibly communicate through a network such as a local area network (LAN) or WAN, including the Internet.

[0030] Wireless networks and the Internet use similar addressing schemes, in which recipients, such as mobile devices in a wireless network or Internet-connected computers, are identified by numerical addresses. For example, mobile devices are identified in the Mobitex network using a Mobitex Access Number (MAN), and public Internet nodes are identified using an IP address scheme. However, differences between wireless network and Internet transport mechanisms prevent direct communication between information sources 20, the vast majority of which are Internet-based, and mobile devices such as 12. Furthermore, information source content is largely targeted to desktop or other computer systems with relatively powerful processors and may require that processor-intensive operations such as information parsing be performed by a recipient. Since mobile devices tend to have less powerful processors, these operations take more time on such mobile devices than on computer systems and can consume significant amounts of power from normally limited-power sources. The IP Proxy system 18 bridges the gap between Internet-based and possibly other information sources 20 and a wireless network 14 with associated mobile devices 12. These services may include address mapping, content transformation and verification, and protocol mapping and optimization, for example.

##### [0031] Detailed Description of the IP Proxy System

[0032] FIG. 2 is a more detailed block diagram of the IP Proxy system 18 shown in FIG. 1. The IP Proxy system 18 may include a dispatcher 22, a Transmission Control Protocol (TCP) handler 24, a Hypertext Transfer Protocol (HTTP) handler 26, a transcoding system 28, one or more push services generally designated 30, a state persistence element 34, a monitoring system 36, and a logging system 38. FIG. 2 also shows a push server 42, a web server 46, a web browser 48, and a file system 40, with which the IP Proxy system 18 may interact from time to time. Many of the components shown in FIG. 2 may be implemented primarily as computer software modules. Elements within the IP Proxy system 18 will typically be running on the same computer, whereas components external to the IP Proxy system 18 are normally resident on separate computers. In an alternative embodiment, the elements of an IP Proxy system 18 may instead be distributed among a group of computers distributed over a network.

[0033] Dispatcher 22 manages data flows and the connection to the gateway 15. Depending on the type of connection or the type of data being transferred or data transaction being performed for example, the dispatcher 22 interacts with the

TCP handler **24** or the HTTP handler **26**. The transcoding system **28** comprises one or more data filters, each of which converts data or other information from one format into a format that can be processed by a mobile device **12**.

[0034] Push services **30** provide for pushing to a mobile device, or transfer of “unsolicited” information from an information source such as push server **42**, which may for example be a web server or a software application, to a mobile device **12** through the IP Proxy system **18**. The push services component **30** allows the push server **42** to address the mobile device using for example the email address of the mobile device owner or some other convenient label. Accordingly, the push server **42** need not know the address of the mobile device **12** in the wireless network **14**.

[0035] The state persistence element **34**, in conjunction with a data file system **40** or a database, enables management of cookies, passwords and possibly other state information associated with web servers **46** to which the IP Proxy system **18** may connect. It preferably stores state information about a connection that persists between discrete network packets, such as an HTTP request/response pair.

[0036] The monitoring system **36** allows remote monitoring of the performance, efficiency, usage and health of an IP Proxy system **18** by an administrator. This monitoring may be accomplished for example through a local interface with the IP Proxy system **18** or possibly remotely through an interface such as a web browser **48**. As its name implies, the logging system **38** may be configured to store usage, connection, user statistics and the like to the file system **40** or some other secondary storage.

[0037] Connections and Handlers

[0038] The IP Proxy system **18** can preferably handle and process content from various information sources **20**, including Internet-based sources. This functionality is provided by connection handlers, which are intermediate objects that have the ability to process content from inbound and outbound connections to an IP Proxy system **18**. In the IP Proxy system **18** shown in FIG. 2, two such handlers, the TCP handler **24** and the HTTP handler **26**, are shown. These handlers can preferably be replaced and customized or additional handlers can preferably be added to an IP Proxy system as needed. The connection handlers can optimise not just the content but also the protocol. For example, some requests that would normally be sent to the mobile device **12** (such as a request for a password) may be resolved by the connection handler, where required information is available through the state persistence element **34** and the file system **40**, for example. This instance of a protocol optimisation can adapt so-called “chatty” protocols to be more wireless friendly by reducing the amount of traffic sent over a wireless network to a mobile device **12**, thereby reducing the effects of wireless network bandwidth constraints and latency.

[0039] Outbound connections would be made from mobile devices **12** in order to send data to and receive data from Internet nodes for example. The IP Proxy system **18** preferably receives connection requests from mobile devices **12** using a particular protocol, such as a proprietary protocol called IP Proxy Protocol or IPPP developed by the assignee of the present application, although other protocols might also be used. The IP Proxy system **18** then establishes an

Internet connection, according to routing information provided by the mobile device **12**, and translates and maps that connection to start forwarding data in both directions. A filtration or transcoding process is invoked whenever necessary, based for example on the type of content being passed over the connection, or on a particular transcoding process specified in the connection request from the mobile device.

[0040] Inbound connections are used, for example, to implement a data push model in accordance with one embodiment. In this embodiment, a mobile device **12** may be sent information without having issued requests to fetch the information, as is the case with outbound connections. As described briefly above, mobile devices **12** may exist on a different network domain than Internet nodes. The IP Proxy system **18** is responsible for bridging the Internet and wireless network domains. Thus, the IP Proxy system **18** requires certain routing information to route the traffic to a particular mobile device **12**. In a push operation, at least some of this routing information must be provided by the Internet node, such as the push server **42**, that issues the request to establish an inbound connection. The IP Proxy system **18** may convert commonly known addressing schemes such as email or IP numbers into the appropriate wireless network address of an intended recipient mobile device **12**. A transcoding process for pushed content may also be selected and specified by a push server **42** or information source **20**.

[0041] Connection handlers in an IP Proxy system **18** are stream-based objects. When an outbound or inbound connection is requested, a virtual piped stream is established between a mobile device **12** and the appropriate connection handler. The connection handler will be instantiated and started to process the content for the established connection. Loading the connection handler is based on a connection request, which preferably contains a reference to appropriate handler name that normally implies the type of the traffic that would go through the virtual piped stream and the location of the handler that must be loaded if it is not already loaded. The functions of connection handlers include mapping Internet or other information source-side connections and mobile device **12** connections, forwarding traffic between these connections, and loading and invoking the appropriate transcoders on information destined for a mobile device **12**.

[0042] Every connection is preferably associated with an instance of a connection handler. This is true even for a connection that does not require that content be processed by the IP Proxy system **18**, such as a pure TCP connection between a mobile device and a server. This type of connection handler forwards content back and forward without making any sort of modification to the content, although it may make modifications to the protocol. For clarity, those skilled in the art will appreciate the distinction between the data or content (what the mobile device requested or is being sent) and the protocol (the “wrappers” and conversions required to deliver the data).

[0043] Connection handlers are also responsible for loading the appropriate content filters or transcoders. A connection handler such as the HTTP connection handler **26** may use a particular transcoder in the transcoder system **28**

selected by the IP Proxy system 18 or specified by either the mobile device 12 or an information source such as push server 42 or web server 46.

[0044] FIG. 3 is a flow chart representing general connection handler-related operations in an IP Proxy system 18. At step 50, the IP Proxy system 18 receives a connection request, which as described above may relate to an inbound connection or an outbound connection. When the connection is associated with a particular handler, such as an HTTP connection that requires HTTP connection handler 26, the appropriate handler is loaded and executed at step 54 and the connection is established, as indicated at step 58. If the request is outbound (from the mobile device 12), then the dispatcher 22 examines the protocol type associated with the request and delegates the connection to the appropriate handler. Data may then be exchanged between a mobile device and an Internet service, push server 42, web server 46 or other information source 20.

[0045] If certain connection handlers are used for a connection, such as for a pure TCP connection as described above, then the data may pass through the IP Proxy system 18 unchanged. In some IP Proxy systems however, content sent over a TCP handler may be modified. When other connection handlers are used however, data destined for a mobile device 12 may need to be converted into a suitable format. FIG. 4 is a flow chart of connection handler data processing operations. At step 62, data destined for a mobile device 12 is received. Although labelled as a response from a connection, following an information request from a mobile device 12 for example, it should be understood that data received by the connection handler may instead be information to be pushed to the mobile device 12 from a push server such as 42 via a push service 30. The connection handler determines at step 64 if transcoding is required. If not, then the information is sent to the mobile device 12 at step 70. Otherwise, the appropriate transcoder is loaded and executed in step 66. The data is transcoded into an acceptable format in step 68 before being sent to the mobile device 12 at step 70. The entity that initiates the communication, the mobile device 12 for fetched data or the push server 42 for pushed data, can preferably specify a particular transcoder to do the transcoding of the fetched or pushed data. Transcoder selection may also be made by a connection handler or IP Proxy system 18 dependent on destination mobile device 12 information available to the IP Proxy system 18 or possibly inferable by an IP Proxy system 18 or components thereof based on previous information transfer operations involving the destination mobile device 12. For example, a transcoder may be invoked to transcode information into the same format that was previously transferred to a destination mobile device 12 the last time information was sent to the mobile device 12.

[0046] A connection handler may be implemented in computer software as a Java™ class file, placed in a certain directory in a file system such that an IP Proxy system Java Virtual Machine (VM) may locate and load the file when needed or requested. As those skilled in the art will appreciate, Java uses CLASSPATH environment variable as a guide to where it should perform a lookup for user defined classes. In one embodiment, paths to connection handlers are to be among the first listed paths in the CLASSPATH so that they would be loaded relatively quickly when requested. The connection direction (inbound or outbound) and the

name associated with a connection handler may also play a role in defining the full class name of a handler. Those skilled in the art will appreciate that the same schemes could be implemented using dynamic linked libraries (DLLs) or dynamic shared objects (DSOs) depending on the target operating system.

[0047] Connection handlers can be associated with a name that represents a protocol at the application layer. For example, if a mobile device 12 is enabled with a web browser and may therefore request to open connection to an Internet server such as 46, it would be appropriate to have HTTP as a name for that connection handler, as shown with connection handler 26. The handler name may adhere to the known rules of naming packages in Java language. Preferably, the handler name is in lower case; however, from an IP Proxy system point of view, it does not matter as long as the Java VM can load that connection handler. Any Connection Handler may also have its class name as Handler.class. An example of a valid full class name that represents a connection handler is as follows:

```
[0048] net.rim.protocol.iplayer.connection.handler.  
<connection direction>.<connection handler  
name>.Handler.class
```

[0049] where connection direction can be either device, which implies outbound connection, or server, which implies inbound connection. Connection handler name is the name associated with the handler, for instance, http, ftp, etc.

[0050] There are at least two ways that an information source such as an Internet node can establish a connection to a mobile device 12 through the example IP Proxy system 18 shown in FIG. 2: (1) using a transportation layer protocol directly, such as TCP, to open a direct connection to the IP Proxy system 18, or (2) using a datagram protocol at the application layer, such as HTTP. The IP Proxy system 18 includes two corresponding connection handlers, which may, for example, represent a basic IP Proxy system 18 which can process two of the most common types of connection. The first is the TCP connection handler 24, associated for example with the name tcp. The second is the HTTP connection handler 26, which may similarly be associated with the name http, as described above. In addition to supporting common connection types, these connection handlers also satisfy requirements for Mobile Information Device Profile (MIDP) implementation at the mobile device. However, the IP Proxy system 18 and the mobile device 12 can be extended to support any other types of connections. In the IP Proxy system 18, connection handlers may possibly be added by providing an application programming interface (API) in the IP Proxy system 18 and developing new connection handlers that adhere to the API, for example.

[0051] In one embodiment, connection handlers in the IP Proxy 18 are loaded from a local storage medium, for example a disk drive associated with a computer on which IP Proxy system software is running. However, in another embodiment, connection handler storage may also or instead be remote from the IP Proxy system 18, such as in a storage medium accessible by the IP Proxy system 18 through a local area network (LAN) connection or even a WAN like the Internet. This embodiment allows sharing of a single directory of connection handlers among all IP Proxy systems 18 that can communicate with the connection handler store. It would also be possible to have third parties extend the



connection handler set by embedding the URL where the connection handler java class can be found.

[0052] If connected to the Internet, a connection handler directory could potentially be accessed and thus shared by all Internet-connected IP Proxy systems 18. Public Internet-connected connection handler directories would preferably receive connection handler requests from IP Proxy systems 18 and in response transmit any requested connection handlers to the requesting IP Proxy system 18. A new connection handler may be required by an IP Proxy system 18 when a mobile device 12 which communicates with the IP Proxy system 18 downloads a new software application or invokes a new mobile device feature which uses a new connection scheme or a connection method that was not previously used by the mobile device 12. A mobile device user or the new application or feature may then send a control message to the IP Proxy system 18, indicating, for example, the name of the required connection handler, perhaps the mobile device application that requires the new connection handler, and an address associated with a connection handler directory from which the new connection handler may be requested. The IP Proxy system 18 would then preferably request the new connection handler from the directory. A connection handler directory could be implemented for example as a web server accessible to an IP Proxy system 18 using HTTP requests.

[0053] When a connection handler is loaded from a remote source, the IP Proxy system 18 preferably stores the handler in a local store in order to provide for faster loading of the handler for subsequent operations involving the corresponding type of connection for either the mobile device 12 for which the connection handler was initially loaded from the directory or a different mobile device 12 supported by the IP Proxy system 18. Depending upon the memory resources available to an IP Proxy system, downloaded connection handlers may be stored indefinitely or for a particular period of time. Alternatively, a least recently used or LRU replacement scheme could be used to provide for more efficient use of available memory by overwriting relatively less frequently used connection handlers when new handlers are downloaded. Other memory management techniques could also be used to optimize local IP Proxy system connection handler storage arrangements.

[0054] Transcoding

[0055] Relative to computer networks such as the Internet, wireless communication networks are slow. Any program that bridges the two, as an IP Proxy system does, may have to transform Internet data so that it is formatted appropriately for a wireless network and mobile device. This process is referred to herein as filtering or transcoding, and usually involves such operations as compressing data from the Internet into a more compact format appropriate for wireless transmission and display on a relatively small mobile device display screen.

[0056] In the following description, transcoding operations are illustrated primarily in the context of the above example of an HTTP handler 26 and HTTP connection. The HTTP connection and handler example is particularly useful in that HTTP allows content tags in the form of Multipurpose Internet Mail Extension (MIME) types, which may be used in some embodiments to determine an appropriate transcoder for received information.

[0057] In an IP Proxy system 18, there may be a single configuration file for each type of connection handler. In the

IP Proxy system 18 for example, a single configuration file associated with the HTTP connection handler 26 may include information for all HTTP content transcoders. This configuration file is used to map transcoders to certain keys. The IP Proxy system 18 may consult this file to determine which content transcoders are available to manipulate any received content destined for a mobile device.

[0058] In the configuration file, general rules are preferably specified for how to define the mapping between content types and transcoders. One example of a possible configuration file entry is as follows:

```
[0059] Entry ={{[default] : {RSV | <Transcoder
name>}}}{[[ InputType]] < ->OutputType> ] : [
Transcoder name]}
```

[0060] where

[0061] default indicates to the IP Proxy system which transcoder should be loaded in case there is no one transcoder associated with a received content type or connection request;

[0062] RSV is a set of reserved keywords that is used in configuration file, such as pass (i.e. forward data to the mobile device without transcoding) or discard (i.e. do not transcode or forward data to the mobile device);

[0063] Transcoder name is the name of the mapped transcoder;

[0064] InputType indicates the input content type that the mapped transcoder can accept, which for an HTTP transcoder configuration file may be a MIME type; and

[0065] OutputType indicates the output type, such as a MIME type for an HTTP transcoder that the transcoder can produce.

[0066] By using a content transcoder configuration file, new transcoders may be added for use by an IP Proxy system 18. Therefore, as new transcoders are developed and become available, they can be added to the configuration file for any appropriate connection handlers and can thereafter be loaded by a connection handler when required, and without affecting other components of the IP Proxy system 18. For example, configuration file entries may be added without shutting down the entire IP Proxy system 18, thus allowing dynamic expansion of data that can be converted for transmission to mobile devices 12.

[0067] In another embodiment, a common configuration file format for all connection handlers is used, and thus a only single configuration file entry need be prepared and can be added to the configuration file for any connection handler. The concept of a common configuration file format for all connection handlers can be further extended to providing a single configuration file for an IP Proxy system 18. Such a configuration file could then be used by all connection handlers in the IP Proxy system 18 to determine which content transcoders are available and to select a particular transcoder for received content. However, it should be understood that a common configuration file format is in no way required. Some connection handlers may share a configuration file entry format or even a single configuration

file, whereas others supported by the same IP Proxy system **18** may have different configuration files and entry formats.

[0068] The IP Proxy system preferably loads and executes a transcoder based on either the type of information to be sent to a mobile device **12**, or a transcoder name specified by a mobile device **12** or an information source **20** to transcode data being sent to a mobile device.

[0069] A transcoder may instead be selected based upon information other than content types, including information in a header portion or other portion of a connection request from a mobile device, a response to an information request, or a communication from an information source including information to be pushed to a mobile device. For example, an IP Proxy system **18** may be configured to determine a type of the mobile device **12** to which data is to be sent. Transcoder selection by the IP Proxy system **18** could similarly be based on a network address or other identifier of the mobile device **12**. Mobile device- or device type-dependent transcoder selection schemes may be supported by providing a device or device type mapping table accessible to the IP Proxy system **18**, which maps devices or device types to transcoders. Alternatively, a configuration file may be adapted to include device or device type identifiers to thereby associate particular transcoders with devices or device types.

[0070] In a similar manner, transcoders may be selected based on an address (such as a URL) or other identifier of an information source, to enable information source-specific transcoding. A mapping table or a configuration file accessible to an IP Proxy system such as **18**, may be used to enable transcoder selection based on information source. This type of transcoder selection may be useful, for example, when a particular transcoder is to be used to transcode any content that originates from a specific website and is destined for a mobile device.

[0071] Although content type-based and specified transcoder selection are the primary types of transcoder selection scheme described below, any of these alternative schemes may be used instead of content type-based transcoder selection. The alternative schemes may also be used to select a transcoder, for example, when a transcoder indicated by a primary transcoder selection scheme is not available, such as when a transcoder system does not include a transcoder configured to transcode a received content type into a content type that the mobile device is configured to accept.

[0072] Pushing Information From an Information Source to a Mobile Device

[0073] As described above, an IP Proxy system **18** may support both outbound and inbound connections. However, this application relates primarily relates to pushing information to a mobile device via inbound connections.

[0074] A server or information push operation differs from information request/response operations, such as those normally associated with web browsing for example, in that an information source **20** sends content to a recipient without receiving a request to do so. A mobile device **12** may register for service with a particular push service by establishing such settings as the particular information that should be pushed to the mobile device **12**, a push period or frequency with which information should be pushed to the mobile

device **12**, a content type or transcoder that should be used for information destined for the mobile device **12**, and possibly other settings related to information push operations. These settings may be established using the mobile device **12** itself or some other interface to a push server **42**, such as a web page for example. It should also be appreciated that an IP Proxy system **18** preferably exercises some level of access control. Each push server **42** may be required to register with an IP Proxy system **18** in order to communicate with mobile devices **12**. Control settings could be established at an IP Proxy system **18** by an IP Proxy system owner or operator or possibly remotely by a mobile device user to restrict push operations to particular registered IP Proxy systems **18**. Access controls may be customized on a per-device, device group or IP Proxy system-wide basis.

[0075] FIG. 5 is a signal flow diagram of an example information push operation. FIG. 5 shows only those components of the IP Proxy system **18** directly involved in an HTTP-based push operation, in order to avoid congestion in the drawing.

[0076] In the example of FIG. 5, content is sent from the push server **42** to the IP Proxy system **18** in a connection request. For an HTTP-based operation, the push may be an HTTP post operation, in which the push server **42** submits an HTTP post request to the IP Proxy system **18**. The post request encloses header fields which specify a resource associated with the IP Proxy system **18**, as a Uniform Resource Identifier (URI) for example, and preferably include an indication of the type of content, such as a MIME type of Wireless Markup Language (WML) in FIG. 5. In an HTTP connection request, the MIME type of WML may be specified in a Content-Type field of an HTTP request header.

[0077] The URI in the connection request from the push server **42** preferably specifies a resource that the IP Proxy system **18** associates with a particular destination mobile device **12** or group of mobile devices **12**. For example, the IP Proxy system **18** may establish a resource for each mobile device **12** that has been configured for operation with the particular IP Proxy system **18**. Such device-specific resources may for example be identified using a mobile device identification number that the IP Proxy system **18** can map to an address of the mobile device **12** in the wireless network **14**. Any information posted to a resource by a push server **42** is then forwarded to the corresponding mobile device **12**, as will be described in further detail below. Alternatively, an IP Proxy system **18** may manage a single resource to which information to be pushed to any mobile devices **12** configured for operation with the IP Proxy system **18** may be posted. In such embodiments, a post request would provide additional information to identify any mobile device(s) **12** to which the posted information is to be sent.

[0078] The connection request from the push server **42** is received by the push service module **30**. In the example of FIG. 5, the push operation is HTTP-based, and the push services module **30** therefore invokes the HTTP handler **26**. It should be appreciated that different push services may be associated with respective handlers in an IP Proxy system **18**, and that a single IP Proxy system **18** may provide several different push services. It is also contemplated that multiple push service modules may be associated with a single connection handler. Alternatively, a single push services

module may be functionally similar to the dispatcher 22 and provide an interface between a push server 42 and any handler in an IP Proxy system 18. For the purposes of clarity however, only a single push service module 30 associated with the HTTP handler 26 is shown in FIG. 5.

[0079] Although the connection request from the push server 42 in FIG. 5 is described as an HTTP request, it should also be appreciated that the connection request may possibly conform to some other protocol used for communications between the IP Proxy system 18 and a push server 42. A connection request may conform to a first protocol, possibly a proprietary protocol, for example, but could specify that a particular connection handler for a second protocol should be used to handle the connection, such that the connection request is interpreted as a connection request according to the second protocol. Therefore, references herein to HTTP connection requests include connection requests that conform to other protocols but are interpreted as HTTP connection requests.

[0080] The HTTP handler 26 determines if the information in the post request from the push server 42 should be transcoded before it is sent to the mobile device 12. This may be accomplished, for example, by establishing a preferred content type for information destined for a mobile device 12. In FIG. 5, this content type is shown as a tokenized, compressed version of WML which is generally referred to as Compiled WML or simply WMLC. The HTTP handler 26 then uses the received content type (WML) to perform a lookup in the configuration file 72, shown in the transcoding system 28 in FIG. 5. It will be appreciated by those skilled in the art however, that the configuration file 72 might instead be external to the transcoding system 28, part of the HTTP handler 26, or even external to the IP Proxy system 18, provided that the HTTP handler 26 can access the file. In one embodiment, the configuration file will be stored in a data store accessible by the IP Proxy system 18, typically on the same computer system on which the IP Proxy 18 is running. In another embodiment, the transcoder selection may instead be controlled by the push server 42 by specifying in the request a content type or transcoder to be used for transfer to the mobile device 12, as described in further detail below.

[0081] The HTTP handler 26 searches the configuration file 72 to determine which if any of its associated transcoders can transcode the received content type, WML, into WMLC for transmission to the mobile device 12. In one embodiment, a lookup table which maps input content types to output content types for all configured transcoders is constructed when transcoders are first loaded to the IP Proxy system 18. In FIG. 5, the configuration file 72 or alternatively a lookup table, includes entries for two transcoders, one for converting from WML to WMLC and the other for converting from Hypertext Markup Language (HTML) to WMLC. The HTTP handler 26, having found the configuration file entry for the WML->WMLC transcoder, then loads the WML->WMLC transcoder 74 from a local store for example, and executes the transcoder to convert the received WML content in the post request into WMLC. The WMLC content is then forwarded to the mobile device 12, through the dispatcher 22. Although FIG. 5 shows the dispatcher 22 handling the communication of the WMLC content to the mobile device 12, similar protocol translation or conversion between HTTP used by the handler 26 and a

communication protocol used by the mobile device 12 may instead be performed by the HTTP handler 26 or another IP Proxy protocol translation/conversion module.

[0082] If the information in a connection request from a push server 42 is already in the preferred content type, then no transcoding may be required. In FIG. 5, if the HTTP post request from the push server 42 included WMLC content, then the HTTP handler 26 would preferably forward the WMLC content to the mobile device 12 without transcoding.

[0083] Transcoding of pushed information is in no way restricted to single-transcoder operations. In the example of FIG. 5, each transcoder converts directly from one format into WMLC. However, it is contemplated that multiple transcoders may be used to convert received content into a format or type that the mobile device 12 is configured to accept.

[0084] FIG. 6 is a signal flow diagram showing multiple or "chained" transcoding operations for an HTTP-based push operation. As in FIG. 5, FIG. 6 shows only those components of the IP Proxy system 18 directly involved in an HTTP-based push operation in order to avoid congestion in the drawings. The components shown in FIG. 6 are substantially the same as those shown in FIG. 5 and operate similarly thereto. The push server, configuration file 78 and transcoders shown in FIG. 6 are labelled differently than in FIG. 5 to indicate that information or content types that these components generate or process may be different. The components themselves may otherwise be the same. For example, push server 80 may be similar to push server 42 except that push server 80 generates HTML content. It should also be appreciated that push server 80 could actually be the same server as push server 42 if push server 42 is configured to generate both WML and HTML content. Similarly, the configuration file 78 may store entries having the same format as those in configuration file 72, but is labelled differently since different entries are shown. Transcoders 82 may also be implemented in the same manner as transcoder 74, but the example transcoders 82 process different content types than the transcoder 74.

[0085] An HTTP post request is sent from the push server 80 to the IP Proxy system 18, possibly through one or more intervening networks and interface components. In FIG. 6, the post request from the push server 80 includes information of HTML content type, specified in a request header field for example as a MIME type of HTML. As described above, the push service module 30 recognizes the request as an HTTP request and loads the HTTP handler 26. Although FIG. 6 shows the same push service module 30 as FIG. 5, a connection request for the push server 80 could be handled by a different push service. The HTTP handler 26 then consults the configuration file 78, searching not only for transcoders that output WMLC, but also for transcoders that output content types that may be input to any transcoder that outputs WMLC. In FIG. 6, the HTTP handler 26, perhaps in a first search pass through the configuration file 78, finds the WML->WMLC transcoder entry. The HTTP handler 26 may then repeat the configuration file search for any transcoders such as the HTML->WML transcoder that convert content into WML, which it can convert into WMLC content type. If a content type other than WML and HTML were provided in the post request from the push server 80, then the

configuration file search may be further repeated by the HTTP handler 26, depending for example on acceptable delays in post request processing.

[0086] In order to avoid the delays and demand on processing resources associated with such multiple search passes through a configuration file, a transcoder content type lookup table may be used. When transcoders are first installed in an IP Proxy system 18, a comprehensive mapping table is preferably constructed to map received content types to possible output content types. For example, in FIG. 6, a lookup table entry for WMLC content would indicate that either WML or HTML can be converted into WMLC. Such a table would also preferably indicate that HTML to WMLC transcoding involves two stages of transcoding. The table might instead be organized into single- and chained-transcoding sections, whereby if only a single transcoding operation is preferred, the single-transcoder part of the table including an entry for the WML->WMLC transcoder would be accessed. If further transcoding operations and the associated processing operations and time delays are acceptable, then the HTTP handler 26 may perform a lookup of a received content type or possibly an input type for a previously identified transcoder in a chained-transcoder section of the table. Preferably, the format of the transcoding configuration file may be changed to represent just such a lookup table in order to speed up the search. This may be accomplished, for example, by specifying a path between content types involving multiple transcoders.

[0087] It is also feasible for a chain of transcoders to include both local and remote transcoding services. These remote transcoding services could be transcoder files that an IP Proxy system 18 discovers, downloads and executes or they could be web based transcoding services which receive data in one format and return it in another, as described in further detail below.

[0088] The determination regarding whether or not multiple transcoding operations will be permitted may be made by the HTTP handler 26 either before or after the table or configuration file lookup operation is performed. In the example of FIG. 6, it should be apparent that multiple transcoders may be invoked to convert received content into WMLC.

[0089] Once the configuration file entries for the HTML->WML and WML->WMLC transcoders are found in the configuration file 78 by the HTTP handler 26, the HTTP handler 26 first loads and executes the HTML->WML transcoder to transcode the received HTML content into WML. The HTTP handler then loads and executes the WML->WMLC transcoder on the WML result of the first transcoding operation. The resultant WMLC content is then forwarded to the dispatcher 22 and then to the mobile device 12. When WMLC content is returned by the push server 80, the HTTP handler 26 forwards the content to the dispatcher 22 without transcoding, whereas if WML content is returned, the WML->WMLC transcoder would be invoked, as described above.

[0090] The determination as to whether or not multiple transcoding operations are allowed may also be made dependent upon predetermined criteria such as maximum HTTP request processing time or maximum content transcoding time or processor time for example. This determination might also take mobile device user- or push server-specified

priority into account. If high time priority (low time delay) is assigned by a mobile device 12 user for information destined for the user's mobile device 12, then single transcoder operations may be selected. Alternatively, if a high data priority is associated with information to be sent to a mobile device 12, then any number of chained transcoder operations may be allowed in order to get the information to the mobile device 12 in an acceptable format. User settings could be applicable to all pushed information, certain types of pushed information, or information originating from certain specific push servers. Transcoding could also or instead be controlled by a push server, as described in further detail below.

[0091] Other criteria which may be applied by a connection handler include but are in no way limited to allowing chained transcoders only for relatively small amounts of received content, only at certain times of day, under specific current traffic conditions, or only when the configuration file or lookup table is stored in a local file system. Further criteria will be apparent to those skilled in the art and as such remain within the scope of the present application.

[0092] It is also possible that more than one multiple-transcoder chain may be available to convert between any two content types. In such situations, there may be some priority, based for example on transcoding cost or fidelity, that an IP Proxy system 18 uses to select between several available chains.

[0093] In the above examples of push operations, the push server 42 or 80 indicates the content type of information in the connection request to the IP Proxy system 18. However, if a push server pushes data content but does not specify a content type, then the default transcoder is preferably used. If the default transcoder discards received content or outputs a content type that cannot be accepted by the mobile device, 12 an error message is preferably returned to the push server, which may then re-send the data to the mobile device 12. The error message further preferably indicates to the server a reason for any delivery failure, such that the push server may attempt to remedy the delivery problem if possible before the data is re-sent. Where the data could not be delivered to the mobile device 12 because no content type was specified and the default transcoder could not transcode the data into an acceptable content type, for example, then the push server may re-send the data with an appropriate content type.

[0094] The above illustrative examples also assume that the IP Proxy system 18 knows that the mobile device 12 can accept WMLC content, or at least that WMLC is a preferred content type for mobile device-destined information. If the IP Proxy system 18 does not know which content type(s) that the mobile device 12 can accept, then the default transcoder is preferably used. Alternately, the active connection handler, the HTTP handler 26 in FIGS. 5 and 6, may instead consult the transcoder configuration file 72, 78 or lookup table to determine if a transcoder that accepts the returned content type as input is available. If an available transcoder is found, then it is loaded and used to transcode the received content. If more than one such transcoder is found, then one of them, for example the transcoder having the first entry in the configuration file or the transcoder that was used most recently to transcode data for the particular mobile device 12 to which the content is destined, may be

loaded and executed. In **FIG. 6** for example, if no preferred content type is known by the IP Proxy system **18**, then the HTML->WML transcoder would be loaded and executed and the resultant WML content could then be returned to the mobile device **12**.

**[0095]** Specifying a Content Transcoder From a Push Server

**[0096]** A connection request from a push server may also specify that a particular transcoder be used to transcode any content to be pushed to a mobile device **12**. For an HTTP connection for example, an IP Proxy system **18** may be configured to expect a Content-Transcoder field in an HTTP request header to indicate that a push server **12**, which may, for example, be associated with a mobile device software application or feature, is specifying a particular transcoder. The IP Proxy system **18** will load and execute the specified transcoder to transcode the pushed content. The Content-Transcoder header field should have a value that is valid in the context of the HTTP configuration file, or where another connection handler is used, its corresponding configuration file.

**[0097]** If a requested transcoder is not available, then an error message will preferably be sent back to the push server **42**, for example in the form of an IOException indicating that the requested transcoder is not available. The push server **42** may then have the option to retry the request with a different transcoder. When the pushed information is intended for a mobile device software application or component that requires information in a particular format available only from the specified transcoder however, the request may instead be retried at a later time when the specified transcoder may possibly be available.

**[0098]** Transcoder selection in a connection request from a push server **42** will now be described in further detail by way of an illustrative example of an HTTP-based push operation. **FIG. 7** is a signal flow diagram of an example of push server-controlled transcoder selection for an HTTP-based push operation. As above, **FIG. 7** shows only those components of the IP Proxy system **18** directly involved in an HTTP-based server push operation.

**[0099]** In **FIG. 7**, content is pushed from the push server **42** to the IP Proxy system **18**. For an HTTP-based operation, the push may be an HTTP post operation, as described above. The post request encloses header fields in which at least a transcoder name (WML->WMLC in this example) and possibly an indication of the type of content, such as a MIME type of WML in **FIG. 7**, may be specified. Since the content is provided by the same entity that selects the particular transcoder, the content type will normally be compatible with the specified transcoder and therefore need not necessarily be specified in the post request.

**[0100]** The post request from the push server **42** is received by the push service module **30**. In the example of **FIG. 7**, the push operation is HTTP-based, and the push service module **30** therefore invokes the HTTP handler **26**. As in **FIGS. 5 and 6**, although only a single push service module **30** associated with the HTTP handler **26** is shown in **FIG. 7**, an IP Proxy system **18** may include multiple push service modules, or the module **30** may be associated with multiple connection handlers.

**[0101]** The example connection request shown in **FIG. 7** specifies the particular transcoder in terms of its input

content type (WML) and output content type (WMLC). However, other transcoder naming conventions are also possible. When a configuration file has entries in a format as described above, part of the file entry for each transcoder indicates its respective input and output content types. The "Transcoder Name" field in such a configuration file entry therefore need not necessarily also include the input and output content types. Although many different transcoder naming schemes are possible, a particular transcoder is preferably specified in any mobile device requests and configuration files using the same name.

**[0102]** The HTTP handler **26** preferably uses the transcoder name in the post request, WML->WMLC in **FIG. 7**, to perform a lookup in the configuration file **72** to determine if the specified transcoder is available in the IP Proxy system **18**. It should be appreciated that the configuration file **72** might be part of the transcoding system **28** as shown in **FIG. 7**, external to the transcoding system **28**, part of the HTTP handler **26**, or external to the IP Proxy system **18**.

**[0103]** In **FIG. 7**, an entry for the transcoder specified in the post request exists in the configuration file **72**. The WML->WMLC transcoder **74** is therefore available to the IP Proxy system **18**, and the transcoder **74** is loaded and executed to transcode the WML content enclosed in the post request into WMLC content. The WMLC content is forwarded to the mobile device **12**, through the dispatcher **22**. When content is provided by a push server **42** in a mobile device-acceptable format, WMLC in the example of **FIG. 7**, the post request may specify a null or other predetermined value in an appropriate request header field to specify that the content should be forwarded to the dispatcher **22** without transcoding. It is also contemplated that a push service module **30** may be configured to directly manage the transcoding of pushed content, instead of invoking a separate connection handler.

**[0104]** If the particular transcoder specified in the post request from the push server **42** is not available to the IP Proxy system **18**, then the push operation may be aborted. Alternatively, a different transcoder having an input content type and output content type respectively compatible with the content from the post request and a content type accepted by the mobile device **12** (if known to the IP Proxy system **18**) may be used. Any time the requested transcoder could not be used to transcode pushed content, a push operation failure or error message may be returned to the push server **42**, particularly if the push server **42** is configured to retry undelivered content. Since pushed content was not requested by the mobile device **12**, no such error or failure message would typically be sent to the mobile device **12**. When the default or any other transcoder is used instead of the specified transcoder, then the push server **42** may be informed of the particular transcoder that was used.

**[0105]** Any such alternate transcoding operations may instead be controlled by the push server **42**. For example, when the transcoder configuration file **72** does not include an entry for the specified WML->WMLC transcoder, the IP Proxy system **18** may send a failure or error message to the push server **42** indicating that the specified transcoder is not available or cannot be used, as described above. The push server **42**, a server software application associated with the connection request, or an operator or administrator of push

server 42 may then respond to the message indicating the action to be taken. This action may include, for example, forwarding the content to the mobile device 12 without transcoding, invoking the default transcoder, invoking a different particular transcoder specified by the push server 42, or discarding the content. The push server 42 may also set a transcoder substitution policy, such as no transcoder substitutions allowed, chained transcoders allowed, etc., in the original connection request sent to the IP Proxy system 18.

[0106] The IP Proxy system 18 may also determine which if any of the transcoders with corresponding entries in the configuration file 72 may transcode the pushed content into either the output content type of the transcoder specified in the connection request or other content types, and identify such available transcoders in the failure or error message sent to the push server 42. The push server 42, software application or operator may then use this information to determine if any of the available transcoders should be used to transcode the pushed content. For instance, if the content cannot be transcoded by the specified transcoder into a format required for particular processing operations at the mobile device 12, but a second transcoder is available to transcode the returned content into a content type that can be viewed on the mobile device 12, then the push server 42 may re-submit the content and/or specify the second transcoder. Although the originally intended processing operations might not be possible using content that was transcoded using the second transcoder, the user is able to at least view the content.

[0107] In order to avoid sending connection requests that specify unavailable transcoders, it may be desirable for the push server 42 to query the IP Proxy system 18 for a list of available transcoders prior to issuing a connection request. A connection request can then be prepared using one of the transcoders known to be available to the IP Proxy system 18. If a required transcoder is not available at an IP Proxy system 18, then the push server 42 may query other IP Proxy systems in an attempt to find the required transcoder, prepare a connection request specifying an alternate but available transcoder or abort an information request operation involving the required transcoder.

[0108] The signal flow diagram of FIG. 7 shows a single content transcoder in a server data push via an HTTP post operation. It should be apparent that a server may specify more than one content transcoder, to be used for example in a chained transcoding operation.

#### [0109] External Transcoder Systems

[0110] As described briefly above, transcoders may be loaded as needed from a local store on a computer system on which an IP Proxy system 18 has been implemented. Transcoders may also be loaded from an external store. FIG. 8 is a general block diagram of a communication system with an external transcoder system.

[0111] The system 90 shown in FIG. 8 is similar to system 10 of FIG. 1 except for the external transcoder system 86. Elements common to both systems 10 and 90 have been described above. As shown by the dashed lines in FIG. 8, the IP Proxy system 84 may communicate with the transcoder system 86 through some sort of direct connection such as a serial port or connection, through a WAN 16 such as the

Internet, or through a LAN 88 within which the IP Proxy system 84 and the transcoder system 86 are configured to operate. Other communication links between the IP Proxy 84 and the transcoder system 86 will be apparent to those skilled in the art.

[0112] FIG. 9 is a signal flow diagram illustrating an HTTP-based push operation with an external transcoder system such as shown in FIG. 8. As in the preceding examples, an HTTP post request is sent from the push server 42 to the IP Proxy system 84, specifying a particular transcoder (WML->WMLC) and possibly indicating the content type, WML in this example. The connection request shown in FIG. 9 is for illustrative purposes only, and need not necessarily include a content type indication or specify a particular transcoder.

[0113] The request is received by the push service module 93 in the IP Proxy system 84, which determines that the request is an HTTP request and thus loads and invokes the HTTP connection handler 94. The HTTP handler 94 may be substantially similar to the HTTP handler 26, although it operates somewhat differently than handler 26 to load content transcoders. The HTTP handler 94 receives the request from the push service module 93 and may then refer to a transcoder configuration file 92 or a lookup table as described above to determine whether or not the specified WML->WMLC transcoder is available to convert content received in response to the request. If no transcoder is specified in the post request, then a transcoder may be selected based on a content type, substantially as described above.

[0114] The WML content in the HTTP post request from the push server 42 is preferably stored in a file system or other data store 98, which may be the resource identified by the URI in the request, while the appropriate transcoder is loaded. In the example of FIG. 9, the HTTP handler 94 requests the specified WML->WMLC transcoder from the transcoder system 86. Although this request is shown in FIG. 9 as an HTTP request from the HTTP handler 94, it should be apparent that other transfer mechanisms might instead be used by an IP Proxy system 84 to retrieve a transcoder from a remote transcoder system. For example, if the IP Proxy system 84 communicates with the transcoder system 86 via a LAN 88 (FIG. 8), then a LAN protocol or data access and transfer scheme could be invoked by the HTTP handler 94 in order to retrieve any required transcoders. The push service module 93 in the IP Proxy system 84 may instead be configured to retrieve the specified transcoder from the transcoder system 86, possibly through a connection handler.

[0115] In FIG. 9, the transcoder system 86 locates the requested WML->WMLC transcoder among its available transcoders 96 and returns the requested transcoder to the IP Proxy system 84. Regardless of the particular transcoder transfer mechanism implemented, the IP Proxy system 84, or in the example of FIG. 9 the HTTP handler 94, receives and executes the returned WML->WMLC transcoder, as indicated at 100. The previously received and possibly stored WML content may then be processed by the transcoder 100, and the transcoded content is returned to the mobile device 12 by the dispatcher 22.

[0116] If chained transcoder operations are specified in the connection request from the push server 42, then more than

one transcoder request may be made by the IP Proxy system **84** to the transcoder system **86**. Multiple transcoders may instead be requested in a single request to the transcoder system **86**. Processing of previously received content for chained transcoder operations may proceed either as each 1.5 required transcoder is loaded by the IP Proxy system **84**, with intermediate transcoded content possibly being stored in a file system or data store such as **98**, or only when all required transcoders have been loaded.

[0117] When a transcoding operation is complete, a transcoder loaded from the external system **86** is preferably stored locally by the IP Proxy system **84** in order to avoid subsequent requests to the external transcoder system **86** for the same transcoder. Retrieval and loading of a transcoder from a local or internal store in the IP Proxy system **84** will typically be completed much faster than a request to a remote system and reduces traffic on the communication link between the IP Proxy system **84** and the transcoder system **86**. In such IP Proxy systems, the active connection handler, which is the HTTP handler **94** in **FIG. 9**, preferably determines if a required transcoder is stored in a local data store before requesting the transcoder from the external transcoder system **86**. Depending upon the amount of available storage, transcoders may be stored indefinitely or for a certain predetermined period of time. Other memory management schemes, such as over-writing stored transcoders on an LRU basis, for example, may also be used when memory resources are limited.

[0118] The configuration file **92** or transcoder lookup table may be adapted for external transcoder loading by including an indication of the location of a transcoder in the configuration file or table entry for the transcoder. The file **92** or table is preferably updated if a transcoder is stored to, or overwritten in, a local memory, such that the active handler can determine from the initial lookup operation whether or not the transcoder must be loaded from the external transcoder system **86**. When a transcoder has not been or is no longer stored locally, then the file **92** or lookup table preferably indicates from where the transcoder may be retrieved. For a transcoder that may be retrieved through an HTTP connection, the corresponding file or table entry may indicate the IP address of the transcoder system **86**, whereas a network address may be specified in the configuration file or lookup table when a LAN connection is used. If the location of a transcoder system from which a specified transcoder is available is known to the push server **42**, then the location may also or instead be included in the connection request from the push server **42**.

[0119] It is also contemplated that more than one external transcoder system may be implemented in a communication system such as **90**. In such an arrangement, the configuration file **92** or lookup table would preferably include entries for all transcoders that are available to an IP Proxy system **84** through all of the external transcoder systems with which it can communicate. An IP Proxy system **84** may thereby download transcoders from any of a number of transcoder systems via direct or network connections. Overall operation of an IP Proxy system **84** with multiple transcoder systems would be substantially as described above, except that different transcoder systems may be accessed, possibly using different transfer mechanisms and communication protocols, for each data transcoding operation. Chained

transcoding operations may also potentially involve communication with different transcoder systems.

[0120] The configuration file **92** or lookup table is preferably arranged to facilitate a simple resolution scheme when a particular type of transcoder is available from more than one transcoder system. Although an IP Proxy system **84** may be able to access multiple transcoder systems, an owner or administrator of an IP Proxy system **84** may designate one of these transcoder systems as a preferred or default system from which the IP Proxy system **84** first attempts to download a transcoder. The order of preference of transcoder systems for any transcoder available from more than one transcoder system may for example be reflected in the order of configuration file or lookup table entries. If the file or table is arranged by transcoder type, then entries corresponding to the most preferred sources for a particular transcoder are preferably listed before entries associated with other transcoder systems. The configuration file or lookup table may instead be arranged according to transcoder system, with all entries for the default or preferred transcoder system occurring first. A preferred transcoder system might also be specified in a connection request from the mobile device **12**. In these example arrangements, an IP Proxy system **84** will preferably attempt to load a particular transcoder from a preferred source before accessing any other sources.

[0121] If the specified transcoder could not be loaded by an IP Proxy system **84**, then an error message may be returned to the push server **42**. Any of the error or failure operations described above may be performed by the IP Proxy system **84** and push server **42** if the specified transcoder could not be used to transcode received content.

[0122] **FIG. 10** shows a further signal flow diagram for an external transcoder system. In **FIG. 10**, not only the transcoder system **86**, but also the configuration file **102** is external to the IP Proxy system **84** and therefore may be shared among multiple IP Proxy systems. Communications between an IP Proxy system **84** and the configuration file **102** may be via a direct connection or a network connection, and may be different for different IP Proxy systems. For example, the configuration file **102** may be maintained by an owner or operator of a particular IP Proxy system **84** which is linked to the configuration file by a direct communication link, whereas other IP Proxy systems may communicate with the configuration file **102** through local or wide area network connections. The configuration file **102** might also be maintained at the transcoder system **86**. As above, the configuration file **102** may be implemented as a lookup table. The configuration file **102** may thus be considered a registry, with which one or more external transcoder systems such as **86** register available transcoders.

[0123] When an inbound connection request specifying a particular transcoder is received by the push service module **93** in the IP Proxy system **84**, it is recognized as an HTTP request and the HTTP handler **94** is loaded and invoked by the push service module **93**. As described above, the HTTP handler **94** determines if the specified transcoder is available in the IP Proxy system **84** by consulting a configuration file. In the example of **FIG. 10** however, the configuration file **102** is remote from the IP Proxy system **84**. If the configuration file **102** is accessible via HTTP, then the HTTP handler **94** manages the transcoder lookup function with the configuration file **102**. If the configuration file **102** is not

adapted for HTTP, then a different connection handler may be invoked to facilitate the transcoder lookup or configuration file search. Alternatively, the push service module 93 may perform the transcoder lookup/search function. In the example of FIG. 10, the configuration file 102 includes an entry for the specified WML->WMLC transcoder.

[0124] As above, it is assumed that the push server 42 pushes WML content to a mobile device 12. The transcoding system 86 in the example shown in FIG. 10 includes a set of remotely executable transcoders 104, comprising a WML->WMLC transcoder 104a and an HTML->WML transcoder 104b, and thereby enables remote transcoding of content. Instead of requesting and loading the WML->WMLC content transcoder 104a from the transcoder system 86, the HTTP handler 94, another connection handler, depending on the particular transcoder system and the transfer schemes it supports, or possibly the push service module 93, transfers the WML content to the transcoding system 86. Within the transcoding system 86, the appropriate WML->WMLC transcoder 104a is executed and the WML content is transcoded into WMLC format. The WMLC content is then returned to the HTTP handler 94, or to another connection handler if IP Proxy system 84 to transcoder system 86 communications do not use HTTP. When the WMLC content is returned by the transcoding system 86 and received by the HTTP handler 94, possibly through another connection handler and/or the push service module 93, it is forwarded to the dispatcher 22. The dispatcher 22 then prepares a message including the WMLC content and sends the message to the mobile device 12. The HTTP handler 94 may instead prepare a message for transmission to the mobile device 12, which would then be translated (if necessary) by the dispatcher 22 to conform to a communication protocol or scheme used by the mobile device 12.

[0125] Illustratively, the WML content from the push server 42 may be stored by the HTTP handler 94 in case a data transfer or transcoding error occurs. Local storage of the WML content allows an IP Proxy system 84 to re-submit the content, to either the same transcoder system 86 or a different transcoder system. When a push operation is accomplished via an HTTP post request as shown in FIG. 10, the pushed content may be available to the IP Proxy system 84 from the resource to which the content is posted.

[0126] If the content in the connection request from the push server 42 is HTML content, then the HTTP handler 94 or push service module 93, through another handler if required, would submit the HTML content to the transcoder system 86 for chained transcoding using both the HTML->WML transcoder 104b and then the WML->WMLC transcoder 104a. Such chained transcoding operations may also be specified by the push server 42 in the connection request. Chained transcoders may either be part of the same transcoding system 86 as shown in FIG. 10, or implemented in different transcoder systems. When a chained transcoding operation involves different transcoder systems, content from an information source may first be transmitted to one transcoder system for transcoding into an intermediate content type which is returned to the IP Proxy system 84, and the intermediate content type may then be sent to another transcoder system, for transcoding using the specified transcoder or another intermediate transcoder in a transcoder chain. Content is preferably forwarded between different

transcoding systems via the IP Proxy system 84 which is processing the connection request, but may instead be directly transmitted from one transcoder system to another if compatible data transfer mechanisms have been implemented in each transcoding system.

[0127] Data request errors or failures, such as transcoder errors or other situations in which a specified transcoder is unavailable, may be managed according to any of the schemes described above, possibly including such further operations as using a different transcoder to transcode content, returning an error message to the push server 42, and controlling any subsequent processing of a request or content from the push server 42.

[0128] In addition, a push server such as 42 may consult an external configuration file to determine which transcoders are available to an IP Proxy system 84 before a push request is submitted. If a required type of transcoder is not available, then the push server 42 may determine if any other transcoder operation, including chained transcoder operations, may be suitable for the push request and an intended recipient mobile device 12 and format the push request accordingly, thereby possibly avoiding failures or errors at the IP Proxy system 84. As described above, the configuration file 102 may be a registry including entries for transcoders available from one or more transcoder systems. When entries in the configuration file 102 include an address, such as an IP address, or other identifier of a transcoder system from which a particular transcoder is available, then the address may be supplied to an IP Proxy system 84 by a push server 42 in a push request. At least some transcoder searching operations may thereby be off-loaded from IP Proxy systems 84 to push servers 42.

[0129] In the system of FIG. 10, it is contemplated that the transcoder system 86 and configuration file 102 may communicate with each other to ensure that the configuration file 102 accurately indicates which transcoders are available. A configuration file may be associated with a particular type of connection such as HTTP connections and thus HTTP connection handlers. If a configuration file 102 is associated with a particular transcoder system 86, then the configuration file may be resident within the transcoding system 86.

[0130] If multiple transcoding systems are implemented, a shared configuration file storing transcoder entries for the transcoders available in all transcoder systems may simplify the transcoder lookup performed by a connection handler. An IP Proxy system 84 or push server 42 need then only consult a single configuration file to determine if appropriate transcoders are available from any transcoder systems with which it can communicate. This single configuration file/server could also support protocols to allow external transcoding servers to register. A registration process could add a list of available transcoders to the single configuration file for example.

[0131] An external transcoding system 86 preferably supports a query function to allow a push server 42 to determine which transcoders are available before a connection request is prepared and sent to an IP Proxy system 84. Transcoders can also be added to the transcoder system 86 and configuration file 102. A push server 42 may add a transcoder to the transcoding system 86 and push content that relies on the new transcoder to mobile devices such as mobile device 12 through the IP Proxy system 84.



[0132] External transcoder systems **86** include download systems from which transcoders may be downloaded by an IP Proxy system **84** and executed locally, as shown in **FIG. 9**, and remote transcoding systems to which content is sent for transcoding at the transcoding system as shown in **FIG. 10**. In another embodiment, a “hybrid” transcoder system incorporates both of these types of transcoder systems. When a hybrid transcoder system is available to an IP Proxy system **84**, the IP Proxy system **84** may either download a required transcoder from the transcoder system or send content to the transcoder system to be transcoded remotely. Alternatively, if the push server **42** knows the content type or transcoder that should be used for information to be sent to the mobile device **12**, then the push server **42** may itself download a transcoder from or submit content for transcoding to an external transcoding system and include the transcoded content in the connection request. This offloads transcoding from an IP Proxy system **84** to a push server **42** and makes an information push operation independent of transcoders available to an IP Proxy system **84**. This concept of push server transcoding could be further extended to include transcoder downloading from an IP Proxy system **84** and local execution of the transcoder on a push server **42**.

[0133] The selection of transcoder download or remote transcoding may be dependent, for example, upon the amount of data to be transcoded, the complexity of the transcoding (single or chained operations), a type of transcoding specified in a connection request, or other criteria. Similarly, chained transcoding operations may involve download transcoding systems and local transcoder execution as well as remote transcoding systems.

[0134] External transcoding systems may also support such services as transcoder downloading or remote transcoding for a push server such as **42**. A push server **42** may be configured to manage transcoding of information content before the information content is pushed to the mobile device **12**. In **FIG. 10**, for example, the push server **42** may consult the configuration file **102** to determine whether an appropriate transcoder, a WML->WMLC transcoder, is available in a transcoder system. Since the transcoder system **86** includes a WML->WMLC transcoder **104a**, the configuration file **102** would include an entry for the transcoder **104a** and possibly an indication of an address, such as a URL or IP address, for example, from which the transcoder is available. In **FIG. 10**, the transcoder system **86** is a remote transcoding system, such that the push server **42** may submit the information content to be transcoded to the transcoder system **86**. The push server **42** may therefore incorporate a connection handler which enables communication with the transcoder system **86**. Transcoded WMLC content from the transcoder **104a** would then be returned to the push server **42**. The push server **42** preferably caches the transcoded content in a local or remote data store accessible to the push server **42**. The cached transcoded WMLC content may then be retrieved from the data store and pushed to a mobile device **12** through the IP Proxy system **84**. A push request from the push server **42** preferably includes an indication that the information content to be pushed to the mobile device **12** has already been transcoded into a content type that the mobile device is configured to accept. Since the information content in such a push request has been transcoded, it is forwarded to the mobile device **12** by the push services module **93**, through a connection handler such as the HTTP handler **94**, if necessary, and the dispatcher **22**.

[0135] Although “pre-transcoding” by a push server has been described above in the context of a remote transcoding system, it should be appreciated that information content may instead be locally transcoded by a push server **42** using a download transcoding system or a transcoding system provided at the push server **42**.

[0136] Example Implementation

[0137] An example implementation of an IP Proxy system will now be described. **FIG. 11** is a block diagram showing an IP Proxy system **124** implemented in a secure network.

[0138] The system **120** in **FIG. 11** includes a mobile device **12** that operates within a wireless network **14**. Through a gateway **15**, the mobile device can receive and preferably also send data over a WAN **16** such as the Internet. These elements of the system **120** are substantially the same as similarly labelled elements in **FIG. 1**. In the system **120** however, the IP Proxy system **124** is configured within a private network such as a corporate network **130**, behind a security firewall **127**, and communicates with the gateway **15** through a network server computer **122**. In a particular example embodiment, the network server **122** is associated with an email system **128**. Two information sources, an internal push server **126** and an external information source **132**, are also shown in **FIG. 11**.

[0139] The network server **122** preferably enables secure communication to the mobile device **12**, as indicated by the encryption and decryption blocks **122a** and **122b**. The network server **122** encrypts any communications directed to a mobile device **12**. The intended recipient mobile device **12**, using a secret key stored therein, can decrypt encrypted communications from the network server **122**. A mobile device **12** similarly encrypts any information sent to the network server **122**, which can be decrypted by the decryption module **122b**. Those skilled in the art of cryptography will appreciate that the keys and encryption algorithms used at the network server **122** and mobile device **12** are preferably chosen so that it would be computationally infeasible to decrypt encrypted information without the required secret key. One preferred encryption scheme is triple-DES (Data Encryption Standard).

[0140] Key distribution between a network server **122** and a mobile device **12** may be accomplished via a secure connection such as a secure physical connection between the mobile device **12** and the network server **122**, or between the mobile device **12** and another computer within the corporate network. Known public key cryptography techniques may instead be used for key distribution. In a public key scheme, a public key is used to encrypt information in such a way that the encrypted information may be decrypted using a corresponding private key. The public key is stored by, and may be retrieved from, a publicly accessible key repository commonly referred to as a certificate authority or CA, whereas the private key is stored only at a mobile device or system with which the public key is associated. Thus, a network server **122** or any other sender that wishes to send encrypted information to a mobile device **12** may retrieve the mobile device’s public key from a CA and use the public key to encrypt information destined for the mobile device **12**. A mobile device **12** may similarly obtain a network server’s public key from a CA and use the public key to encrypt communication signals to be sent to the server.

[0141] Regardless of the particular key distribution scheme and encryption techniques used, encrypted commu-

nications between a mobile device 12 and network server 122 may be used, for example, where corporate or other private information is to be accessed using a mobile device 12. Consider the example of the internal push server 126 within the security firewall 127, described below with reference to FIG. 12. FIG. 12 is a signal flow diagram illustrating a corporate data push operation. In keeping with the above illustrative example operations, FIG. 12 shows an HTTP-based data push operation.

[0142] In FIG. 12, an HTTP post request from the internal push server 126 is received by the push service module 30 and recognized as an HTTP request. The push service module 30 loads and invokes the HTTP handler 26 in this example, which then consults the configuration file 72 or transcoder lookup table to determine if the a transcoder is available to transcode the received WML content into a device-acceptable format. As described above, an appropriate transcoder may be chosen by the IP Proxy system 124 or specified in the request from the push server 126. In FIG. 12, the WML->WMLC transcoder 74 is loaded and invoked by the HTTP handler 26 and the transcoded content is forwarded to the network server 122 through the dispatcher 22. The network server 122 then encrypts the content received from the IP Proxy system 124 in its encryption module 122a and sends the encrypted content to the mobile device 12.

[0143] In some implementations, the protocol conversion or translation operations associated with the dispatcher 22 may instead be performed by the network server 122. In an alternate embodiment, IP Proxy system functionality may be incorporated into a network server 122 to thereby provide a network server that allows access to network resources using a mobile device 12. In another embodiment, an IP Proxy system 124 may incorporate encryption/decryption and communications functions of the network server 122 in order to communicate with the wireless network gateway (FIG. 11) and thus mobile devices such as 12.

[0144] The internal push server 126 may be associated with a computer system or data store preferably configured for operation on the private network 130, such as a file server or other data store accessible through the network 130. In the example of a corporate network, the information source 126 may include confidential or otherwise sensitive information that an owner of the network 130 strives to keep private. The security firewall 127 is intended to prevent unauthorized access to private network components including the information source 126. In some situations, the very existence of information stored at the information source must remain confidential. The encryption of content sent to the mobile device 12 as shown in FIG. 12 prevents an unauthorized party from determining the contents of the request without breaking the encryption, which as described above is not computationally feasible for strong encryption schemes such as 3DES.

[0145] Encryption of pushed content by the encryption module 122a in the network server 122 before it is sent to the mobile device 12 ensures that the content can only be viewed by the mobile device 12. Confidential corporate information therefore remains encrypted and thus secure until received and decrypted at the mobile device 12, thereby effectively extending the security firewall 127 to the mobile device 12. Information sent by the mobile device 12 to the network server 122 is similarly encrypted by the mobile

device 12 and remains encrypted until decrypted by the decryption module 122b. For example, an HTTP get request may be prepared on the mobile device 12, and then encrypted and sent from the mobile device 12 to the network server 122 in order to request information resident on an information source within the corporate network 130. The request remains encrypted until received by the network server 122 and decrypted, behind the security firewall 127, as indicated at 134 in FIG. 12. The request is therefore virtually as secure as a request sent from a computer system on the network 130.

[0146] Once decrypted, the request is passed to the HTTP handler 26, which requests the information from the appropriate source. Returned information is transcoded if required, passed to the dispatcher 22, encrypted by the encryption module 122a and returned to the mobile device 12. Both the request and the information returned to the mobile device 12 in response thereto are secure.

[0147] In known remote data access schemes such as WAP, gateway systems which provide for data access using mobile devices 12 are normally located outside corporate or private premises, at the location of a service provider for example. Any confidential or sensitive information encrypted at the private premises is decrypted at the gateway system, outside the corporate firewall, and then re-encrypted before being sent to the destination mobile device or devices 12. The information is therefore in the clear at the gateway system and thus accessible by an owner or operator of the gateway system. Furthermore, the owner or operator of a private network from which the information was sent typically has no control over security arrangements at the gateway system, such that the information is vulnerable to attacks on the gateway system.

[0148] The arrangement shown in FIGS. 11 and 12 provides for secure remote access to private, confidential or otherwise sensitive information. Information is encrypted from end-to-end between the network server 122 and any mobile device 12. Any level of security may be implemented at the security firewall 127 to protect confidential information stored at an internal push server such as 126 or other internal information sources, and when encrypted by the network server 122, information is not decrypted at any intermediate point before being received at a mobile device 12. The information is in the clear only "inside" the point 134, behind the security firewall 127, and on the mobile device 12. Security arrangements such as password or passphrase control are also preferably implemented at the mobile device 12 to prevent an unauthorized user from using the mobile device or decrypting received encrypted information. For example, computer workstations may be protected by password-deactivated system locking and access to a corporate network 130 is normally protected by login passwords. Similarly, a password may be required to use a mobile device 12, while a different passphrase may be necessary to decrypt any encrypted information stored on the mobile device. A mobile device 12 and information stored thereon is thereby just as secure as a network workstation and information stored on a network. Such techniques as limited password or passphrase entry retries, mobile device 12 or mobile device memory reset after a predetermined number of failed password/passphrase

entries, dynamic and possibly random password/passphrase updates and the like may be used to further improve mobile device security.

[0149] For an external information source 132 (FIG. 11), a data push operation would be substantially the same as shown in FIG. 12, except that the information source is outside the firewall 127. It should be appreciated that any information source may be configured to provide information in response to a request from an IP Proxy system 124, push information to a mobile device through an IP Proxy system 124, or possibly to perform both functions. Any information exchange between the mobile device 12 and the network server 122 may be encrypted, but information exchanged with the information source 132 may be unsecure. If the information provided by the information source 132 is not private or confidential, then unsecure exchange between the IP Proxy system 124 and the source 132 will be sufficient for most purposes. However, if the external source 132 provides private information, then alternate arrangements are preferably provided.

[0150] One possible measure to improve the security of information being requested from an external source 132 is to secure the communications between the IP Proxy system 124 and the source 132. For example, the IP Proxy system 124 may be adapted to support Secure HTTP (HTTPS), Secure Sockets Layer (SSL) or other secure communication schemes in order to securely access information at the information source 132. Information from the source 132 may thereby be securely transferred to the IP Proxy system 124 and is then protected by the security firewall 127. Encrypted information may be decrypted by the IP Proxy system 124, by the active connection handler for example, and transferred to the network server 122, which then encrypts the information for transmission to the mobile device 12. As above, information is only in the clear behind the firewall 127. Alternatively, a secure communication session may be established between the mobile device 12 and source 132 through the IP Proxy system 124. In the system of FIG. 11, communications between the mobile device 12 and network server 122 would then be double-encrypted.

[0151] As shown in FIG. 11, the network server 122 is also associated with the email system 128. In one embodiment, the network server 122 provides redirection of data items from the email system 128 to mobile device 12. One such system is described in detail in U.S. Pat. No. 6,219,694, entitled "System And Method For Pushing Information From A Host System To A Mobile Data Communication Device Having A Shared Electronic Address", and issued to the assignee of the present application on Apr. 17, 2001. The complete disclosure of this patent is hereby incorporated into this application by reference.

[0152] Since the network server 122 is also associated with the IP Proxy system 124, integrated functionality between the email system 128 and the IP Proxy system 124 may be possible. For example, the IP Proxy system 124 may use encryption functionality of the network server 122 as well as a transport mechanism via which the network server 122 communicates with the mobile device 12. Other functions of the network server 122, such as data compression for example, may similarly be exploited by an IP Proxy system 124 to improve the efficiency of use of wireless communication resources.

[0153] Similarly, content destined for a mobile device 12 may be addressed to the mobile device using an email address in the email system 128 associated with the mobile device user. In this example, content forwarded to the mobile device 12 by the IP Proxy system 124 may also be stored in the user's mailbox on email system 128 by the network server 122, as indicated in FIG. 11, to thereby provide both a record of IP Proxy system operations and a stored copy of any forwarded content. Other integrated functions may include but are in no way limited to email-based content requests from mobile devices and addressing of device-destined information by the IP Proxy system 124 using an email address on the email system 128. Still further integrated functions may be enabled where a network server 122 or the IP Proxy system 124 is associated with any other services.

[0154] It will be appreciated that the above description relates to exemplary embodiments by way of example only. Other variations exist and are within the scope of the invention. For example, embodiments of the invention have been described primarily in the context of an IP-based system. Similar proxy systems for other types of communication systems are also contemplated within the scope of the invention. Other types of connections, connection handlers and transcoders than those described above will also be apparent to those skilled in the art.

[0155] Depending upon the particular implementation of a remote data access system and the features to be supported, not all of the elements shown in FIG. 2 are required.

[0156] The instant invention is also in no way limited to content type indication using MIME types. MIME types are useful in conjunction with the instant invention, but are not required to practice the invention. Other content type indicators may be substituted for MIME type to indicate the type or format of requested or received content.

[0157] Although the transcoders described above convert between well-known information types or formats, custom transcoders could be developed and implemented for virtually any information format, including for example application program file types and proprietary formats. As described above, a proxy system in accordance with the instant invention is preferably configurable and new content transcoders may be added.

[0158] It is also possible that information content from an information source may include multiple different content types, not just a single content type as described above. For such multiple-type content, transcoders may be selected, for example, to transcode the content into a single content type, or into multiple content types accepted at a mobile device. Selection of transcoders may be controlled according to any of the transcoder selection schemes described above. In the case of transcoder selection by a mobile device or information source, a list of transcoders for any or each part of multiple-type information type content may be specified in a connection request, a response to a request, or a push request. A respective transcoder may be selected and used for each part of the information content having a particular content type. A push server may instead transcode any or all parts of multiple-type information content before such content is pushed to a mobile device.

[0159] When any part of multiple-type information content cannot be transcoded as desired or required, where a

suitable transcoder is not available for example, only other parts of the information content might be transcoded and sent to a mobile device. Alternatively, a default transcoding operation as described above may be used to transcode parts of multiple-type content. Non-transcoded parts of multiple-type content, or possibly all of the multiple-type content, could instead be replaced with a link or other information that may be used to subsequently access the information content or parts thereof, and sent to a mobile device. Information indicating the multiple content types and/or required or recommended transcoders could also be sent to the mobile device. The information content or parts thereof may then be retrieved by the mobile device by submitting a connection request or possibly further transcoding instructions or an alternate transcoder selection to an IP Proxy system or push server.

**[0160]** Furthermore, a proxy system may be implemented in any network, not only in a corporate network as shown in **FIG. 11**. Installation of a proxy system in an ISP, ASP, or Virtual Network Operator (VNO) system would provide for secure remote access to network information and secure transfer of information between any network users, including transfers between mobile devices of ISP, ASP or VNO users.

**[0161]** Although the invention has been described in detail with reference to certain illustrative embodiments, variations and modifications exist within the scope and spirit of the invention as described and defined in the following claims.

What is claimed is:

1. A system for pushing information content from an information source to a mobile communication device over a network, comprising:

a transcoding system comprising a plurality of transcoders, each transcoder operable to transcode the information content from a respective input content type into a respective output content type; and

a first network device in communication with the transcoding system, the first network device comprising a push module, wherein the push module is operable to receive a connection request from the information source comprising an identifier associated with the mobile communication device, and further operable to select a corresponding connection handler operable to select one or more transcoders from the plurality of transcoders to transcode the information content.

2. The system of claim 1, wherein the first network device is further operable to transmit the information content transcoded by the one or more selected transcoders to the mobile communication device.

3. The system of claim 1, wherein the transcoding system is further operable to transmit the information content transcoded by the one or more selected transcoders to the mobile communication device.

4. The system of claim 1, wherein the connection request further comprises transcoder request data that identifies a requested transcoder.

5. The system of claim 1, wherein the connection handler is operable to determine one or more acceptable content types that the mobile communication device is configured to accept.

6. The system of claim 5, wherein the connection handler is operable to search the plurality of transcoders for transcoders operable to transcode the information content from a received content type of the information content into the one or more acceptable content types.

7. The system of claim 5, wherein the transcoding system is operable generate and store mapping data comprising transcoding chains, each transcoding chain selecting one or more transcoders to transcode the information content from a respective input content type into a respective output content type.

8. The system of claim 7, wherein the connection handler is operable to select a transcoding chain to transcode the information content from a received content type of the information content into one of the accepted content types.

9. The system of claim 5, wherein the transcoding system comprises a configuration file associated with the plurality of transcoders, and the connection handler is operable to search the configuration file to determine whether any of the transcoders are operable to transcode the information content from a received content type of the information content into the one or more acceptable content types, and to select the transcoders where any of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

10. The system of claim 9, wherein the connection handler is further operable to transmit an error message to the information source where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

11. The system of claim 9, wherein the information content includes multiple content types, and the connection handler is further operable to transmit an error message to the information source where none of the transcoders are operable to transcode one or more of the multiple content types into the one or more acceptable content types.

12. The system of claim 9, wherein the connection handler is further operable to determine a type of the mobile communication device and to select one or more transcoders from the plurality of transcoders based on the type of the mobile communication device where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

13. The system of claim 9, wherein the connection handler is further operable to select one or more transcoders from the plurality of transcoders based on the identifier associated with the mobile communication device where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

14. The system of claim 9, wherein the connection handler is further operable to determine an address associated with the information source and to select one or more transcoders from the plurality of transcoders based on the address associated with the information source where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

15. The system of claim 9, wherein the connection handler is further operable to transmit a list of selectable transcoders to the information source where none of the transcoders are

operable to transcode the information content from the received content type into the one or more acceptable content types.

**16.** The system of claim 15, wherein the connection handler is operable to receive selected transcoder data from the information source and to select one of the selectable transcoders from the list of selectable transcoders based on the selected transcoder data.

**17.** The system of claim 9, wherein the connection handler is further operable to discard the information content where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

**18.** The system of claim 9, wherein the connection handler is further operable to pass the information content where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

**19.** The system of claim 9, wherein the transcoding system is further operable to transcode the information content into a content type pushed to the mobile communication device in response to a previous connection request where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

**20.** The system of claim 9, wherein the information content includes multiple content types, and the first network device is further operable to transmit only transcoded content types to the mobile communication device where none of the transcoders are operable to transcode one or more of the multiple content types into the one or more acceptable content types.

**21.** The system of claim 4, wherein the transcoder request data comprises a network address specifying the location of a transcoder.

**22.** The system of claim 21, wherein the transcoding system is operable to access the location specified by the network address and retrieve the transcoder.

**23.** The system of claim 4, wherein the transcoding system comprises a configuration file associated with the plurality of transcoders, and the connection handler is operable to search the configuration file to determine whether the requested transcoder is one of the plurality of transcoders and to select the requested transcoder where the requested transcoder is one of the plurality of transcoders.

**24.** The system of claim 23, wherein the connection handler is further operable to transmit an error message to the information source where the requested transcoder is not one of the plurality of transcoders.

**25.** The system of claim 24, wherein the connection handler is further operable receive alternate transcoder request data in response to the error message, the alternate transcoder request data identifying an alternate transcoder.

**26.** The system of claim 23, wherein the connection handler is further operable to transmit a list of selectable transcoders to the information source where the requested transcoder is not one of the plurality of transcoders, and is further operable to receive selected transcoder data from the information source and to select one of the selectable transcoders based on the selected transcoder data.

**27.** The system of claim 23, wherein the connection handler is further operable to discard the information content where the requested transcoder is not one of the plurality of transcoders.

**28.** The system of claim 23, wherein the connection handler is further operable to pass the information content where the requested transcoder is not one of the plurality of transcoders.

**29.** The system of claim 1, wherein the identifier comprises a network address of the mobile communication device.

**30.** The system of claim 1, wherein the connection handler is further operable to determine a type of the mobile communication device and to select one or more transcoders from the plurality of transcoders based on the type of the mobile communication device.

**31.** The system of claim 1, wherein the connection handler is further operable to select one or more transcoders from the plurality of transcoders based on the identifier associated with the mobile communication device.

**32.** The system of claim 1, wherein the connection handler is further operable to determine an address associated with the information source and to select one or more transcoders from the plurality of transcoders based on the address associated with the information source.

**33.** A method for pushing information content to a mobile communication device, comprising the steps of:

receiving the information content from an information source;

receiving an address of the mobile communication device;

providing a plurality of transcoders, each transcoder operable to transcode information content from a first content type into a second content type;

selecting one of a plurality of connection handlers based on the information content received;

selecting one or more transcoders from the plurality of transcoders, the one or more transcoders selected by the connection handler;

transcoding the information content using the one or more of the plurality of transcoders selected to generate transcoded information content; and

sending the transcoded information content to the mobile communication device.

**34.** The method of claim 33, wherein the step of selecting one or more transcoders from the plurality of transcoders comprises the steps of:

determining whether any of the plurality of transcoders are operable to transcode the information content from a received content type of the information content into any of one or more accepted content types that the mobile communication device is configured to accept; and

selecting a transcoder operable to transcode the information content from the received content type into one of the accepted content types where any of the plurality of transcoders are operable to transcode the information content from the received content type into any of the one or more accepted content types.

**35.** The method of claim 34, further comprising the step of discarding the information content where none of the plurality of transcoders are operable to transcode the information content from the received content type into any of the one or more accepted content types.

**36.** The method of claim 34, further comprising the step of performing a default transcoding operation on the information content where none of the plurality of transcoders are operable to transcode the information content from the received content type into any of the one or more accepted content types.

**37.** The method of claim 36, wherein the default transcoding operation comprises the step of passing the information content.

**38.** The method of claim 36, wherein the default transcoding operation comprises the step of transcoding the information content into a content type previously sent to the mobile communication device.

**39.** The method of claim 34, further comprising the steps of:

transmitting a list of selectable transcoders to the information source where none of the plurality of transcoders are operable to transcode the information content from the received content type into any of the one or more accepted content types;

receiving selected transcoder data from the information source; and

selecting one of the selectable transcoders from the list of selectable transcoders based on the selected transcoder data.

**40.** The method of claim 33, wherein the information source is a web server connected to the Internet.

**41.** The method of claim 33, further comprising the steps of:

receiving a network address specifying the location of a transcoder operable to transcode the information content from the received content type into one of the accepted content types;

accessing the location specified by the network address; and

retrieving the transcoder.

**42.** The method of claim 33, wherein the step of transcoding the information content using one or more of the plurality of transcoders selected comprises the steps of:

sending the information content to a transcoding system; and

receiving transcoded information content from the transcoding system.

**43.** The method of claim 33, wherein the step of sending the transcoded information content to the mobile communication device comprises the step of encrypting the transcoded information content.

**44.** The method of claim 33, wherein the step of selecting one or more transcoders from the plurality of transcoders comprises the steps of:

generating a list of transcoders according to an order of preference; and

selecting one or more of the transcoders in the list of transcoders based on the order of preference.

**45.** The method of claim 33, further comprising the step of mapping the plurality of transcoders to create a plurality of transcoding chains, each transcoding chain associating one or more transcoders to transcode a respective input content type into a respective output content type.

**46.** The method of claim 45, wherein the step of selecting one or more transcoders from the plurality of transcoders comprises the steps of:

identifying transcoding chains having a respective input content matching a received content type of the information content and a respective output content type matching one of one or more accepted content types that the mobile communication device is configured to accept; and

selecting an identified transcoding chain to transcode the information content.

**47.** The method of claim 46, further comprising the steps of:

determining a priority status related to the information content; and

transcoding the information content or passing the information content depending on the priority status.

**48.** The method of claim 33, wherein the mobile communication device is configured to accept one or more content types selected from the group consisting of Wireless Markup Language (WML), Hypertext Markup Language (HTML), compiled WML (WMLC) and Extensible Markup Language (XML).

**49.** The method of claim 33, wherein the step of selecting one or more transcoders from the plurality of transcoders comprises the steps of:

determining a type of the mobile communication device; and

selecting one or more transcoders from the plurality of transcoders based on the type of the mobile communication device.

**50.** The method of claim 33, wherein the step of selecting one or more transcoders from the plurality of transcoders comprises the step of selecting one or more transcoders from the plurality of transcoders based on the address of the mobile communication device.

**51.** The method of claim 33, wherein the step of selecting one or more transcoders from the plurality of transcoders comprises the steps of:

determining an identifier associated with the information source; and

selecting one or more transcoders from the plurality of transcoders based on the identifier.

**52.** The method of claim 33, wherein:

the information content comprises multiple content types; and

selecting respective transcoders from the plurality of transcoders to transcode the multiple content types.

**53.** The method of claim 33, wherein the step of selecting one or more transcoders from the plurality of transcoders comprises the steps of:

determining whether the information content has been pre-transcoded into a content type that the mobile communication device is configured to accept; and

transmitting the information content to the mobile communication device without further transcoding where the information content has been pre-transcoded.

**54.** A system for receiving pushed information content over a network, comprising:

a mobile communication device comprising a communication subsystem operable to transmit push data over the network, the push data comprising an acceptable content type the mobile communication device is configured to receive, the mobile communication device further operable to receive pushed information content in the acceptable content type during a push period.

**55.** The system of claim 54, further comprising an information source operable to receive the push data and to provide the pushed information content for transmission to the mobile communication device during the push period.

**56.** The system of claim 55, wherein the push data further comprises push period data specifying the push period, and the information source is operable to provide the pushed information content for transmission to the mobile communication device once every push period.

**57.** The system of claim 56, wherein the information source is operable to transmit an error message to the mobile communication device where the pushed information content cannot be provided in the acceptable content type.

**58.** The system of claim 55, wherein the information source comprises a transcoding system operable to transcode information content into the acceptable content type.

**59.** The system of claim 55, wherein the information source is operable to transmit the information content to a transcoding system, to receive transcoded information content in the acceptable content type from the transcoding system, and to transmit the transcoded information content to the mobile communication device.

**60.** The system of claim 57, wherein the mobile communication device is further operable to transmit alternate content types to the information source in response to the receiving the error message.

**61.** The system of claim 54, further comprising a proxy server in communication with the information source, the proxy server operable to select a from a plurality of transcoders to transcode the pushed information content into the acceptable content type.

**62.** The system of claim 61, wherein the proxy server is operable to transmit an error message to the information source where the pushed information content cannot be transcoded into the acceptable content type.

**63.** The system of claim 62, wherein the information source is further operable to transmit alternate content types to the proxy server in response to receiving the error message.

**64.** A system for pushing information content to a mobile communication device, comprising:

means for receiving a mobile communication device address from the information source;

means for providing a plurality of transcoders, each transcoder operable to transcode information content from a first content type into a second content type;

means for selecting one or more transcoders from the plurality of transcoders;

means for transcoding the information content using the one or more transcoders selected to generate transcoded information content; and

means for sending the transcoded information content to the mobile communication device.

**65.** The system of claim 64, wherein the means for selecting one or more transcoders from the plurality of transcoders comprises means for determining whether any of the plurality of transcoders are configured to transcode a received content type of the information content into any of one or more accepted content types that the mobile communication device is configured to accept.

**66.** The system of claim 65, wherein the means for determining whether any of the plurality of transcoders are configured to transcode the received content type into any of the one or more accepted content types comprises means for discarding the information content where none of the plurality of transcoders are configured to transcode the received content type into any of the one or more accepted content types.

**67.** The system of claim 65, wherein the means for determining whether any of the plurality of transcoders are configured to transcode the received content type into any of the one or more accepted content types comprises means for performing a default transcoding operation on the information content where none of the plurality of transcoders are configured to transcode the received content type into any of the one or more accepted content types.

**68.** The system of claim 67, wherein the default transcoding operation passes the information content.

**69.** The system of claim 67, wherein the default transcoding operation transcodes the information content into a content type previously sent to the mobile communication device.

**70.** The system of claim 67, wherein the means for determining whether any of the plurality of transcoders are configured to transcode the received content type into any of the one or more accepted content types comprises means for transmitting a list of selectable transcoders to the information source where none of the plurality of transcoders are configured to transcode the received content type into any of the one or more accepted content types.

**71.** The system of claim 65, wherein the means for selecting one or more transcoders from the plurality of transcoders further comprises means for selecting one or more transcoders based on the mobile communication device address where none of the plurality of transcoders are configured to transcode the received content type into any of the one or more accepted content types.

**72.** The system of claim 65, wherein the means for selecting one or more transcoders from the plurality of transcoders further comprises means for determining an address of the information source, and means for selecting one or more transcoders based on the address of the information source where none of the plurality of transcoders are configured to transcode the received content type into any of the one or more accepted content types.

**73.** The system of claim 64, wherein the means for selecting one or more transcoders from the plurality of transcoders comprises means for selecting one or more transcoders based on the mobile communication device address.

**74.** The system of claim 64, wherein the means for selecting one or more transcoders from the plurality of transcoders comprises means for determining a type of the

mobile communication device, and means for selecting one or more transcoders based on the type of the mobile communication device.

75. The system of claim 64, wherein the means for selecting one or more transcoders from the plurality of transcoders comprises means for determining an address of the information source and means for selecting one or more transcoders based on the address of the information source.

76. The system of claim 64, wherein:

the mobile communication device address comprises a network address specifying the location of a transcoder; and

the means for selecting a transcoder from the plurality of transcoders comprises means for accessing the location specified by the network address and retrieving the transcoder.

77. The system of claim 64, further comprising means for encrypting the transcoded information content.

78. The system of claim 64, further comprising means for compressing the transcoded information content.

79. The system of claim 64, wherein the means for selecting one or more transcoders from the plurality of transcoders comprises:

means for searching the plurality of transcoders for a set of transcoders configured to transcode a received content type of the information content into one or more accepted content types that the mobile communication device is configured to accept;

means for generating a list of respective input content types corresponding to the set of transcoders; and

means for sending the list of respective input content types and the one or more accepted content types to the information source.

80. The system of claim 64, wherein the means for providing a plurality of transcoders comprises means for mapping the plurality of transcoders to create a plurality of map entries, each map entry associating one or more transcoders to transcode a respective input content type into a respective output content type.

81. The system of claim 80, wherein the means for mapping the plurality of transcoders to create a plurality of map entries comprises means for determining the input content types for the plurality of transcoders, determining the output content types for the plurality of transcoders, and creating a plurality of map entries, each map entry associating a respective input content type with a respective output content type.

82. A system for pushing information content from an information source to a mobile communication device over a network, comprising:

a transcoding system comprising a plurality of transcoders, each transcoder operable to transcode the information content from a respective input content type into a respective output content type; and

a proxy server in communication with the transcoding system, the proxy server comprising a push module, wherein the push module is operable to receive a connection request from the information source comprising an identifier associated with the mobile communication device, and the push module is further

operable to select one or more transcoders from the plurality of transcoders to transcode the information content.

83. The system of claim 82, wherein the proxy server is further operable to transmit the information content transcoded by the one or more transcoders selected to the mobile communication device.

84. The system of claim 82, wherein the push module is operable to determine one or more acceptable content types that the mobile communication device is configured to accept.

85. The system of claim 84, wherein the push module is operable to search the plurality of transcoders for transcoders operable to transcode the information content from a received content type of the information content into the one or more acceptable content types.

86. The system of claim 84, wherein the transcoding system is operable generate and store mapping data comprising transcoding chains, each transcoding chain selecting one or more transcoders to transcode the information content from a respective input content type into a respective output content type.

87. The system of claim 86, wherein the push module is operable to select a transcoding chain to transcode the information content from a received content type of the information content into one of the accepted content types.

88. The system of claim 84, wherein the transcoding system comprises a configuration file associated with the plurality of transcoders, and the push module is operable to search the configuration file to determine whether any of the transcoders are operable to transcode the information content from a received content type of the information content into the one or more acceptable content types and to select the transcoders where any of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

89. The system of claim 88, wherein the push module is further operable to transmit an error message to the information source where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

90. The system of claim 89, wherein the push module is further operable receive information content in an alternate received content type in response to the error message.

91. The system of claim 88, wherein the push module is further operable to transmit a list of selectable transcoders to the information source where none of the transcoders are operable to transcode the information content from the received content type into the one or more acceptable content types.

92. The system of claim 91, wherein the push module is operable to receive selected transcoder data from the information source and to select one of the selectable transcoders from the list of selectable transcoders based on the selected transcoder data.

93. The system of claim 82, wherein the push module is operable to select one or more transcoders from the plurality of transcoders based on the identifier.

94. The system of claim 82, wherein the push module is further operable to determine an address of the information source, and to select one or more transcoders from the plurality of transcoders based on the address.