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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
16 June 2005 (16.06.2005)

PCT

(10) International Publication Number
WO 2005/055495 A2

(51) International Patent Classification[7]: H04L

(21) International Application Number:
PCT/US2004/039740

(22) International Filing Date:
26 November 2004 (26.11.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/525,301 26 November 2003 (26.11.2003) US

(71) Applicant *(for all designated States except US)*: **SAFFI & JONES, LLC** [US/US]; 2 Grange Street, Greenwich, CT 06830 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **SAFFI, Furio** [SE/SE]; Lomv. 605, S-19257 Sollentuna (SE). **JONES, Steven, P.** [US/US]; 2 Grange Street, Greenwich, CT 06830 (US).

(74) Agent: **OLSON, Timothy, J.**; Wiggin & Dana LLP, One Century Tower, New Haven, CT 06508-1832 (US).

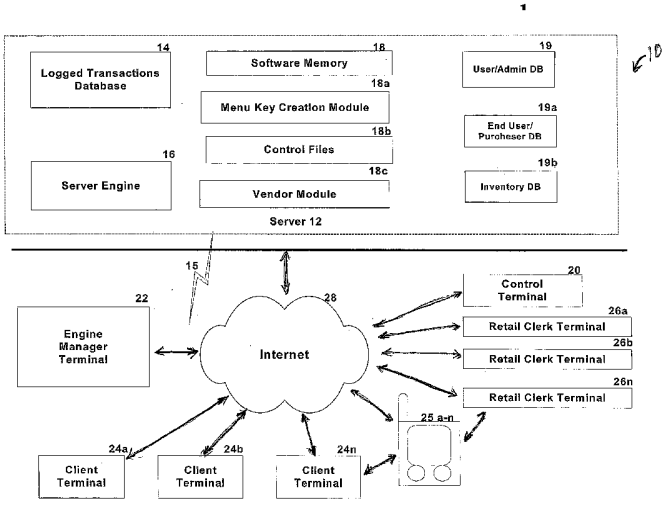
(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR,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

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

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

(54) **Title:** SYSTEM AND METHOD FOR PREPARING MOBILE DIGITAL DEVICES



(57) **Abstract:** A System (10) for configuring one or more mobile digital devices (25) (e.g., pocket personal computers, cellular telephones, and the like) provides the ability to prototype and configure vertically focused or otherwise custom packages of media that can be installed in the mobile digital devices (25) to increase the usefulness and simplify the use of the mobile digital devices (25). The customized packages of media may be installed in the mobile digital devices (25) by way of: a retail clerk operating a retail clerk terminal (26); business personnel operating a client terminal (24); or by the user of the mobile digital device (25) via a connection through network (28) to any of the terminale (22, 24, or 26). A main server system (12), control terminal (20), and engine manager terminal (22) manage the creation and distribution of the packages of media as well as the creation of new client and retail clerk terminals (24, 26).

WO 2005/055495 A2

SYSTEM AND METHOD FOR PREPARING MOBILE DIGITAL DEVICES

Cross Reference to Related Applications

This application claims the benefit of US Provisional Patent Application No. 60/525,301 (Attorney Docket No. 25461.001) filed November 26, 2003 and entitled "Methods and Apparati for Preparing Digital Devices", which is incorporated by reference herein in its entirety.

Background of the Invention

The present invention relates generally to mobile digital devices and, more specifically, to a system and method for configuring mobile digital devices.

A mobile digital device is a portable computer that is small enough to be carried in one's hand and which communicates digital data wirelessly to another computer or data network. Mobile digital devices include, handheld computers, palmtop computers, personal digital assistants (PDAs), pocket computers, pocket personal computers (pocket PCs), cellular telephones (e.g., smart phones), BlackBerry mobile e-mail devices, and the like. As with larger portable computers such as notebook computers, mobile digital devices include one or more microprocessors, digital memory, one or more input/output (I/O) devices, a display, and a battery or other mobile power source. Indeed, modern mobile digital devices have a computing power and flexibility that rival many early notebook computers.

The promise of powerful, flexible, always on, always connected mobile digital devices is becoming a reality as the bottlenecks to usage such as battery life and connectivity (speed + access) and total cost of ownership are moved aside. Individual and corporate buyers are seeking the thought leadership on how to make such devices work for them. They know that, when configured to their personal needs, mobile digital devices will provide a number of business and personal advances such as, for example, increasing worker productivity and adding flexibility to the work environment. Yet with all this motivation, many companies and individuals have yet to implement mobile digital devices in their businesses and lives. The hesitancy of many companies and individuals to adopt this technology is due, at least in part, to certain drawbacks found in mobile digital devices.

Because of their small size, mobile digital devices typically have smaller displays than those found in notebook computers and do not include a full size

keyboard as found in notebook computers. To compensate for the lack of a full-size keyboard, manufacturers have devised a number of different input technologies. For example, some manufacturers have implemented the use of touchscreen displays, which allow the user to input information into the mobile digital device by touching (using a finger or stylus) virtual buttons appearing on the display. Other manufactures have implemented handwriting recognition technologies, which allow the user to input information by printing or writing on a touchscreen or other sensor. Other methods include: the use of very small keyboards upon which the user types with his or her thumbs, the use of pointing devices (e.g., touchpads, multi-directional buttons, joysticks, and the like) which are used to control the movement of a pointer to select items on the display screen, and the use of voice recognition technology where the user inputs data verbally.

While input technologies for mobile digital devices have advanced, the entry of data using such technologies is slow relative to the speed at which information can be entered into larger computers. This drawback is compounded by increases in the computing power and flexibility found in modern mobile digital devices, which require the user to select from and operate a myriad applications, features, and options available on the mobile digital device. In addition, the complex applications, features, and options found in mobile digital devices lends itself to user confusion and error, and often requires the user to undergo costly training to learn how to properly use the device to meet his or her needs.

Moreover, configuring a mobile digital device (e.g., loading and updating applications, inputting and adjusting settings, inputting and adjusting security provisions, and the like) is typically a manual process, which requires data entry using either the inefficient input technologies or by physically coupling the device to a larger computer (e.g., a desktop computer) via a serial or USB port. This manual programming is time consuming and must be performed by trained personnel. The human and capital resource cost of configuring mobile digital devices is particularly high in cases where a salesperson is required to program a number of mobile digital devices or where a business assigns mobile digital devices to many employees. Because of the time and resources required, many businesses are hesitant to adopt mobile digital devices on an enterprise-wide level.

Thus, there remains a need for a system and method for configuring mobile digital devices that allows mobile digital devices to be configured to the needs of the

user while reducing configuration time and while allowing configuration changes to be made on an enterprise level.

Summary of the Invention

The above-described and other needs are met by a method and apparatus for configuring mobile digital devices, wherein a first plurality of software applications from different vendors and a first plurality of user interface features indicative of the first plurality of applications are selected. The first plurality of applications are associated with corresponding user interface features in the first plurality of user interface features to create a first media package, which includes the first plurality of applications and the first plurality of user interface features. The first media package is loaded onto the mobile digital device to configure the mobile digital device. The configured mobile digital device executes the applications in the first plurality of applications in response to selection of the corresponding user interface features in the first plurality of user interface features by a user of the mobile digital device. The first plurality of applications and the first plurality of user interface features may be associated with a vertical market.

In one embodiment, at least one additional media package is selected from a group of additional media packages. The at least one additional media package includes a second plurality of software applications and a second plurality of user interface features indicative of the second plurality of applications. The at lest one additional media package is added to the first media package before loading the first media package on the mobile digital device. One or more optional software applications may be selected from a plurality of optional software applications for addition to the first media package, and a user's manual file may be added to the first media package.

In one embodiment, the mobile digital device may be set to prevent the user's access to user interface features other than those included in the first media package. The mobile digital device may also be set to display the user interface features upon start up of the mobile digital device.

In one aspect of the invention, a key is created for the first media package along with an end-user account associated with the key. The end-user account identifies the user, the mobile digital device, and the applications in the first plurality

of applications. The key is loaded on the mobile digital device to allow the user to use the first plurality of applications.

In one embodiment, the key activates the first plurality of user interface features to allow the user to use the first plurality of applications. Access to the first plurality of applications is prevented in mobile digital devices where the key is not present. The key on the mobile digital device may be changed to prevent access by the user to the first plurality of applications. Changing the key on the mobile digital device may deactivate the first plurality of user interface features to prevent the user to use the first plurality of applications. The key may be changed through a wireless connection to the mobile digital device. The key may be used to identify transactions from the mobile digital device.

In one embodiment, the end user account is used to identify an application in the first plurality of applications in need of upgrading. An upgrade application is then downloaded to the mobile digital device to replace the application in need of upgrading. The applications indicated in the end-user account may be reloaded onto the mobile digital device in response to a request from the user. The end-user account may further include account information associated with the applications in the first plurality of applications.

In one aspect, the first media package is loaded on the mobile digital device by a terminal connected by a network to a main server system. The main server system enables the terminal by providing software to the terminal. In response to a request from the terminal for a key to the first media package, the main server system identifies and authenticates the terminal before sending the key to the terminal. The terminal may include one of: a retail clerk terminal, a client terminal, and an engine manager terminal.

In any of the embodiments, the first media package may be loaded on the mobile digital device via the Internet.

Brief Description of the Drawings

Referring now to the drawing wherein like items are numbered alike in the various Figures, where:

Fig. 1 is a schematic diagram of a system for configuring one or more mobile digital devices;

Fig. 2 is a schematic diagram of a key creation process employed by the system of Fig. 1;

Fig. 3 is a schematic diagram of a portion of the key creation process of Fig. 2;

Fig. 4 is a schematic diagram depicting registration and transaction control steps employed by the system of Fig. 1;

Fig. 5 is a schematic diagram depicting various messages transmitted and received by each of the terminals and the main server system in the system of Fig. 1;

Fig. 6 is a schematic diagram depicting the functionality and creation of various terminals (benches) in the system of Fig. 1;

Fig. 7 is a schematic diagram depicting the interaction of a smart menu with various applications in a mobile digital device;

Fig. 8 is a schematic diagram depicting a configured media package prototyping process employed by the system of Fig. 1;

Fig. 9 depicts various examples of user interfaces of a mobile digital device as may be configured using the process of Fig. 8;

Fig. 10 is a flow diagram of a process for the prototyping process employed by the system of Fig. 1;

Fig. 11 is a schematic diagram depicting various components of a configured media package for a mobile digital device;

Fig. 12 is a schematic diagram depicting various components of an engine manager terminal of the system of Fig. 1;

Fig. 13 is a schematic diagram depicting a process to prepare client and retail clerk terminals to receive files to enable them to provide the service of distributing configured media packages;

Fig. 14 is a flow diagram of a process of activating an engine manager terminal for a transaction with the main server system employed by the system of Fig. 1;

Fig. 15 is a flow diagram of a process of activating client and retail clerk terminals for a transaction with the main server system employed by the system of Fig. 1;

Fig. 16 depicts a window displayed to the operator of the engine manager terminal for inputting information to create a client or retail clerk terminal in the system of Fig. 1;

Fig. 17 is a flow diagram depicting the process of installing and configuring the media package onto the mobile digital device;

Fig. 18 is a schematic diagram depicting the creation of a media package from one or more standard media packages, additional media packages, and optional applications for the mobile digital device; and

Fig. 19 depicts a window displayed to the operator of the engine manager terminal, client terminal, or retail clerk terminal to allow the configuration of a media package for the mobile digital device.

Detailed Description

Referring to Fig 1., a system for configuring one or more mobile digital devices is depicted generally at 10. The system 10 includes seven basic components: a main server system 12, a control terminal 20, an engine manager terminal 22, client terminals 24, retail clerk terminals 26, and one or more mobile digital devices 25, each of which is configured to communicate data via a network 28 such as the Internet. Each of the components may communicate with the network 28 via wired communication, wireless communication, or a combination of wired and wireless communication. As will be described in further detail hereinafter, system 10 provides the ability to prototype and configure vertically focused or otherwise custom packages of media (e.g., applications, settings, business rules, data files, documents, graphics, user interfaces, and the like) that can be installed in the mobile digital devices 25 to increase the usefulness and simplify the use of the mobile digital devices 25 for the end-user/purchaser. The customized packages of media may be installed in the mobile digital devices 25 by way of: a retail clerk operating the retail clerk terminal 26; business personnel operating the client terminal 24; or by the user of the mobile digital device 25 via a connection through network 28 to any of the retail, client, or engine manager terminals 22, 24, 26. The main server system 12, control terminal 20, and engine manager terminal 22 manage the creation and distribution of the packages of media as well as the creation of new client and retail clerk terminals (24, 26).

As also discussed hereinafter, a smart menu key 50 (Fig. 2) is created for the media package. The smart menu key 50 is loaded on the mobile digital device 25 to allow the user to use the applications in the media package. The main server system 12 creates an end-user account associated with the key 50, which identifies the user, the mobile digital device 25, and the applications in the media package to allow for

various functions, such as upgrading or reloading the applications in the mobile digital device 25.

As used herein, a mobile digital device 25 is a portable computer that is small enough to be carried in one's hand and which communicates digital data wirelessly to another computer or data network. Mobile digital devices include, handheld computers, palmtop computers, personal digital assistants (PDAs), pocket computers, pocket personal computers (pocket PCs), cellular telephones (e.g., smart phones), BlackBerry mobile e-mail devices, and the like. The mobile digital device includes one or more microprocessors, digital memory, one or more input/output (I/O) devices, a display, and a battery or other mobile power source.

While network 28 is depicted in Fig. 1 as the Internet, it will be appreciated that distribution of media can be accomplished in various environments other than the Internet. For example, distribution of customized media packages can also be electronic mail environment in which an item as described in electronic mail message along with an indication of the custom configuration that is to be performed to affect the distribution of the item (s). Also, various communication channels may be used such as local area network, wide area network, wireless, cellular, radio or point to point dialup connection.

As shown in Fig. 1, the main server system 12, includes a server engine 16, a logged transactions database 14, software memory 18, menu key creation module 18a, control files 18b, a vendor module 18c, a user/administrator database 19, end-user/purchaser database 19a and inventory database 19b. The components of the main server system 12 may be resident on a single server computer, or may be distributed across different server computers (e.g., a web server, database server, etc.) and memory devices. The server system 12 may comprise any combination of hardware and software that can generate custom configuration of media and distribute media in response to the action being performed. As used herein, a database may comprise any combination of hardware and software that enables one to store, modify, and extract information (e.g., a database management system and associated data storage). As used herein, a module is a set of instructions that, when executed, causes a computer to behave in a predetermined manner, and an engine is a main set of instructions that, when executed, controls various modules in addition to performing other functions.

The main server system 12 performs many functions that can be grouped into three basic functions. First it centralizes the collection and storage of information

using various databases. The user/administrator database 19 retains the name of each user/administrator, their system 10 access information, and their registration key for accessing one or more of the terminal(s) 22, 24, 26. As used herein, terminal may comprise any hardware or software that functions to interact with the main server system 12. As used herein, a key is a unique set of characters or other unique data that may be used to prove authority and provide license to use associated media. The user/administrator database 19 also contains information to allow for security, access, administration and fraud/theft detection. The user/administrator database 19 also retains the identity/registration information for the terminal(s) 22, 24, 26 and access information for the terminal(s) 22, 24, 26. The inventory database 19b retains the information regarding inventory (e.g., media) downloaded from the main server system 12 during each transaction with a terminal 22, 24, or 26. The inventory database 19b contains individual application registration key(s) that activate the usage of the media on the mobile digital devices 25. The inventory database 19b also includes vendor information, access rights and access gateway through the vendor control module 18c for vendors to replenish keys as inventory is depleted. The end user/purchaser database 19a retains end-user/purchaser registration information from transactions that are conducted by plurality of terminals 22, 24, 26. The end-user/purchaser database 19a also contains information such as identity, contact information (email/name) for license and registration purposes of the end-user/purchaser and notification of transactions and keys used by the vendor module 18c.

Second, the main server system 12 manufactures from components of the menu key creation module 18a, useful items such as registration menu key (license) packages, as depicted in Fig. 2, that enables single step registration of software applications included in the custom media distributed by the terminals 22, 24, 26 to be configured on the mobile digital devices 25. The single step end-user registration process is depicted in Fig. 3.

Third, the main server system 12 communicates with the control terminal 20 which acts as a gateway between the main server system 12 and all other terminals 22, 24, and 26. The purpose of this communication is to acquire access authorization to exchange instructions/information to and from all of the other terminals 22, 24, 26, as depicted in Fig. 4. These instructions/information files are transferred using HTTP

requests and other messaging protocols over a communication link such as the Internet 28 using the World Wide Web.

The control terminal 20 performs two basic functions. First, it acts as a gateway providing initial activation and ongoing use of the various terminals 22, 24, and 26 by collecting and exchanging registration information to and with the main server system 12 via messaging and other communications/instructions. These instructions/information files are transferred using HTTP requests and other messaging protocols over a communication link 15 such as the Internet 28 using the World Wide Web.

Second, the control terminal 20 provides the administrator of the control terminal 20 the ability to establish access rules and resets/enforces these rules with time controlled license and registration information allowing a transaction to take place (as depicted in Fig. 4 step 12) or disallowing a transaction to take place (as depicted in Fig. 4 at step 21) between the main server system 12 and the various terminals 22, 24, 26.

Referring to Fig. 5, the various messages transmitted and received by each of the terminals 20, 22, 24, 26 and the main server system 12 are identified. The logged transaction database 14 records these actions performed throughout the networked system. Actions such as registration of created terminals 24 (block 24), applications purchased/used (block 28), and administrator/purchasers information and transaction results (block 41) are all logged for future use.

Software memory 18 holds and retains the parent software/media applications for each of the applications that are offered for use in developing and delivering the final software package to the mobile digital device 25 for use by the end user/purchaser, as shown at step 5 of Fig. 2. A child copy of this software/media application is passed from the main server system 12 to the engine manager terminal 22 to be used when the engine manager terminal creates additional terminals 24 and 26.

As shown in Fig. 2, the menu key creation module 18a creates a smart menu key 50, as depicted at step 3a. The smart menu key 50 provides authority to use a smart menu interface, as discussed hereinafter with reference to Fig. 9. Additionally, the smart key creation module 18a creates registration menu key packages 52, as depicted at step 3b. A registration key package 52 is a combination of various keys that provide authority to use the various media selected to be included in the final

package 54 to be provided to the mobile digital device 25 at step 5. The smart menu key 50 is attached to the registration menu key package 52 and is used to manage the registration menu key package 52 to ensure that the applicable plurality of applications that make up the final media package 54 are registered and fully configured on the mobile digital device 25 as specified by the terminals 22, 24, 26 making the requests to perform distribution of configured media onto the mobile digital device 25. The registration information associated with the smart menu key 50 includes an end-user identification, device identification, short cut settings, identification of the associated applications, and any information for sub-accounts associated with the applications (e.g., where an application enables voice over IP phone communication, the sub-account information may include phone number, account number, SIP identification, and the like). This registration information is then stored as an end-user account in the end-user database 19a. The end-user account can be accessed either through the terminal 22, 24, 26 or through network 28 by the device 25 so that the device 25 can be refreshed back to its most recently configured condition should the device 25 become corrupt, lost, or should the end-user obtain a new mobile digital device 25. For example, the applications indicated in the end-user account may be reloaded onto the mobile digital device25 in response to a request from the user to the main server system 12. The end user account may also be queried to identify an application in need of upgrading. An upgrade application is then downloaded to the mobile digital device 25 to replace the application in need of upgrading. Such upgrading may occur periodically, or when upgraded applications become available. Also, if transactions from the mobile digital device 25 are stored in the end-user database 19a or another database in main server system 12, the smart key 50 associated with the end-user account can be used to identify transactions from the mobile digital device.

Fig. 3 depicts the registration window for use in initiating the registration process, which includes fields for data to be entered by the operator of the terminal 22, 24, or 26 for requesting keys 50 and 52. The data may include the an owner name, company name, and e-mail address associated with the mobile digital device 25.

As shown in Fig. 4, the control files 18b provide control instructions for the control terminal 20. These instructions insure that only authorized users are able to access and operate the terminals 22, 24, 26, as depicted in steps 12b and 14, and that

only authorized terminals 22, 24, 26 are able to access the main server system 12, as depicted in steps 23 an 14, to complete the requests for distribution of custom configured media requested by a terminal 22, 24, 26 operator, for a specific end-user/purchaser.

Referring again to Fig. 1, the vendor module 18c is the interface to the system 10 for the vendors of software applications that have been authorized/selected to be incorporated in any of the packages that are distributed as configured media by any of the terminals 22, 24, 26. The vendor module 18c enables the vendors to have control over their applications by providing an interface to update their applications and query inventory related issues pertinent to their applications. This interface provides the vendors the ability to access information in the various databases to check on key usage, update versions of their application(s), refresh inventory of keys (licenses) and receive information on end-users/purchasers so that the vendors can remain in control of the support for their application(s).

Referring to Fig. 6, the engine manager terminal 22 is created by the main server system 12, as shown at step 57. The engine manager terminal 22 has the ability to: 1) create and manage other media configuration and distribution benches such as plurality of client terminals 24 and retail clerk terminals 26 as shown at steps 58 and 59; 2) prototype new custom configurations for creating media packages as shown in Fig. 8; and 3) create and distribute configured media packages 54 on mobile digital devices 25 as shown in Fig. 2 at step 5. The engine manager terminal 22 may be accessed by the mobile digital device 25 via network 28 so that the mobile digital device can obtain configured media packages 54 via the network 28 (e.g., via the Internet). This allows the user of the mobile digital device 25 to obtain new packages 54 or to upgrade existing packages 54 to include more recent versions of the media in the packages 54. Access of the engine manager terminal 22 may be initiated automatically by the mobile digital device 25 or the engine manager terminal 22 so that upgrades can be made periodically.

The client terminals 24 are created by the engine manager terminal 22 as shown in step 58 of Fig. 6. The client terminals 24 retain all functionality of the engine manager terminal 22 with the exception of being able to create additional terminals/engines and manage them.

The retail clerk terminals 26 are created by the engine manager terminal as shown in step 59 of Fig. 6. The retail clerk terminals 26 retain only the functionality

of distributing custom configured media packages 54 to the mobile digital devices 25 as shown in Fig. 2 at step 5.

Referring again to Fig. 5, with the exception of the control terminal 20, each of the terminals 22, 24, and 26 contain globally unique engine identifiers, as indicated at block 36. Among other functions, these identifiers uniquely identify the user/administrator of the terminals 22, 24, 26, as indicated at block 24, to the main server system 12 in the registration and control process.

The engine identifier, indicated at block 36 of Fig. 5, and the user/administrator unique identifiers, indicated at block 24, are combined and stored in a file (e.g., a "cookie") on the terminals 22, 24, and 26. In one embodiment, the main server system 12 assigns and sends both the administrator/user identifier, block 24, to the user/administrator database 19 (Fig. 1) and the engine identity, block 36, to the control file 18b (Fig. 1) only once when the terminal 22, 24, or 26 first interacts with the main server system 12. From there on, the terminal(s) 22, 24, 26 include their user/administrator identifiers, block 24, and engine identifiers, block 24, with all messages sent to the main server system 12 so that the main server system 12 can identify the source of the messages and the terminal systems 22, 24, 26 can interact by exchanging information via network 28.

Prototyping and Creation of Packages 54

Fig. 7 is a schematic diagram showing the interaction of a smart menu interface (smart menu) 170 with various icons 163 (e.g., pictures, symbols, images, and the like) and applications 174 within a mobile digital device 25. The icons 163 appear as part of a user interface 161 on a touchscreen display 160 of the mobile digital device 25. The applications 174 are stored in the mobile digital device 25 and provide functionality to the mobile digital device 25. While icons 163 are shown, other features of the user interface 161 (e.g., fields, scroll bars, voice recognition commands, handwriting recognition commands, and the like) may also be used.

The smart menu 170 is a middleware application operating on the mobile digital device 25 that associates the icons 163 and other portions of the user interface 161 with the applications 174. For example, when a user selects (touches) one of the icons 163, the smart menu 170 activates the associated application 174. The smart menu 170 also provides configuration specifications to the operating system of the mobile digital device.

In one aspect of the invention, the smart menu 170 activates the user interface 161 features (e.g., icons 163) on the device 25 to allow the user to use the applications associated with the user interface features in response to receiving a valid smart key 50 (Fig. 2). If a valid smart key 50 is not present, the smart menu 170 deactivates the user interface 161 features to prevent the user from using the applications on the mobile digital device 25. This feature is particularly useful in preventing the unauthorized use of the applications. For example, the smart key 50 on the mobile digital device may be changed (e.g., altered or deleted) to prevent access to the applications on the device by the user The smart key 50 may be changed through a wireless connection to the mobile digital device, allowing remote deactivation of the user interface 161 features and, thus, preventing the user from using the associated applications. Other functions of the smart menu 170 will be apparent from further discussion herein.

In Figs. 9 and 12, a flow diagram discloses in detail how the smart menu interface 170 is created for use. The interface construction process of Figs. 9 and 12 are used for four basic purposes: 1) prototyping configured media packages for distribution to mobile digital devices, as depicted in Fig. 8; 2) configuring applications with icons (e.g., pictures, symbols, images, and the like); and 3) creating vertically focused or otherwise custom packages that can be used to configure mobile digital devices 25 (e.g. loaded as packages 54 of Fig. 2). Each of these functions increases and simplifies the usefulness of the mobile digital devices 25 for the end-user/purchaser.

One of the many important functions that can be performed with this aspect of the invention is the ability to prototype and configure vertically focused or otherwise custom mobile digital devices 25. This is possible with consolidation of applications and icons, as depicted in Fig. 8, that are of interest and meaning to different business verticals.

Referring to Fig. 7 and Fig. 9, for example, three examples of a touchscreen display 160 of a mobile digital device 25 are shown, each depicting a user interface 161 comprising various menus 162, icons 163, and the like, each of which may be selected (touched) by the user of the mobile digital device 25 to initiate one or more associated application 174. The smart menu 170 allows a designer to configure and build prototype packages that are vertically focused for the various industries by combining different applications 174 with different icons 163 and other features of the

user interface 161 (e.g., fields, scroll bars, voice recognition commands, handwriting recognition commands, and the like). For example, the user interface 161 and associated applications 174 may be configured for the medical market, with icons 163 identifying the applications 174 that are useful to that market (e.g., access to the Physician's Desk Reference, Lab Data, etc.). In other examples, the user interface 161 may be configured for the legal market or for the real estate market, with icons 163 being associated with applications 174 useful to those respective markets

The ability to configure and build vertically focused or otherwise custom packages simplifies the use of the mobile digital device 25. With the use of icons 163 that are familiar to the end user and meaningful in their practice of business, what is normally complex and hard to use is simplified for use and business process automation.

Fig. 10 is a flow diagram depicting the design, configuration and building of a prototype package for use by the terminals 22, 24, 26 to distribute as a media package (e.g., package 54 of Fig. 2) onto a mobile digital device 25. While Fig. 10 depicts one such method, it is contemplated that other such methods may be used.

In step 901 of Fig. 10, software applications 174 that are useful for a specific group (vertical market) are selected. One such method of selection could be performed for a vertically focused group such as, for example, medical, legal, realtor, finance and sales. For example, as depicted in Fig. 8, industry experts 101 and or other special interests could select the applications 174 to be included in the package, as shown at 102 of Fig. 8.

Upon completion of the selection of applications 174 to be included in packages, the prototype user advances to Step 903. In Step 903, selected application 174 functionalities are matched to meaningful icons 163 (e.g., pictures, symbols, images and the like), as depicted at 105 of Fig. 8. The icons 163 are designed and selected to create an attractive user-friendly interface that will reduce learning time for the mobile digital device 25, reduce user anxiety, reduce complexity of the mobile digital device 25, and incorporate and potentially automate business vertical knowledge. Industry experts, psychiatrists, psychologists and or other special interests could perform selection of icons 163.

In step 905 of Fig. 10, the selected applications 174 are uploaded to a prototype mobile digital device 25 for eventual binding with the smart menu 170, as shown at 106 of Fig. 8. In step 907, the prototype mobile digital device 25 is used to

create the user interface 161 (e.g., a skin) that connects the smart menu 170 with the selected applications 174. In step 909, the prototype smart menu 170 is completed with the creation of main menus, sub menus, and shortcuts, as depicted at 60, 62 and 64, respectively, of Fig. 7, finishing the smart menu configuration and then binding it with the icons 163 and other user interface features that were selected in step 903 to complete the user interface 161 (Fig. 9).

Step 911 is the completion of the prototype by connecting the smart menu 170 with the selected applications 174, setting configuration specifications and binding them all as one package and transferring it to the engine manager terminal 22 or client terminal 24, as shown at 106 of Fig. 8. In step 913, the completed package is then combined with user's manuals in the terminal, as depicted in Fig. 11, and a production ready package 54 is created. Step 915 terminates the process with the completed production ready package 54 ready to be incorporated by any of the terminals 22, 24, 24 of Fig. 1 to perform distribution of configured media onto a mobile digital device 25, as described in Fig. 2, for use by the end user/purchaser.

Prototyping Terminals and Building of Terminals 24, 26

Fig. 6 depicts the abilities of each of the terminals 20, 22, 24, and 26. As previously noted, the engine manager terminal 22 is created by the main server system 12, as shown at step 57. The engine manager terminal 22 has the ability to: 1) create and manage other media configuration and distribution benches such as plurality of client terminals 24 and retail clerk terminals 26 as shown at steps 58 and 59; 2) prototype new custom configurations for creating media packages as shown in Fig. 8; and 3) create and distribute configured media packages 54 on mobile digital devices 25 as shown in Fig. 2 at step 5. The Engine Manager Terminal 22 includes of a directory of files that are supplied to it from the main server system 12 and is configured to perform the smart menu prototyping process previously discussed with reference to Fig. 10.

As detailed in Fig. 12, the engine manager terminal 22 as shown in Fig. 1 comprises an installation bench directory 82 that includes all the components needed to perform the distribution of configured media packages 54 (Fig. 2) to mobile digital devices 25 (Fig. 2). Illustratively, these components include:

1) Packages.cfg file 69, which retains information on all the different packages 54. It lists information such as, the name of the package 54, the list of the

various applications (media) 174 (Fig. 8) that make up the package 54, the vendor and the version of the package 54.

2) Messages.txt file 74, which contains all of the on-line screen instructions and prompts to simplify use of the engine by the engine administrator.

3) ManualHeader.txt file 72, which is the template file for all the user manuals (Fig. 11). File 72 is automatically generated when a new application 174 is added to be used in any of the packages 54.

4) Applications.cfg file 73, which retains information on all the different applications 174. File 73 lists information such as, the name of the application 174, the vendor of the application 174 and the version of the applications 174 that will be used by the engine to perform the distribution of configured media packages 54 to the mobile digital devices 25.

5) Engine_Manager_Terminal_Engine.exe executable program 75, which enables the use of the engine manager terminal 22 to make others terminals such as client terminals 24 and retail clerk terminals 26 (Fig. 1).

6) InstallBench.exe executable program 76, which provides the engine manager terminal 22 with the ability to perform the distribution of configured media packages 54 to mobile digital devices 25.

As shown in Fig. 6, the engine manager terminal 22 has the unique ability to create additional terminals 24, 26 for commercial purposes of being able to perform the distribution of configured media packages 54 (Fig. 2) to mobile digital devices 25. The components that are needed to build the other engines are also shown in Fig. 12, and they include:

1) Menu Key file 77, which comprises the main smart menu key definition file that enables the user-interface and the key definition files for the other applications that are to be included for use in the building of configured media packages 54 for distribution to mobile digital devices 25.

2) Plurality of packages 54 that have been built by the prototyping functionality described with reference to Fig. 10 and Fig. 8.

3) Applications file 79, which includes a compressed file (e.g., a cabinet (".cab") file) and a manual for each of the various applications 174 that can be added to a package 54 for distribution to mobile digital devices 25.

4) Bench Factory file 80, which includes the components needed to manufacture and prepare for distribution and activation new client and retail clerk terminals 24, 26. The various configurations that are allowed to be used for the terminals 24, 26 and the holding place for terminals 24, 26 that have been built and are awaiting delivery.

5) Help Files folder 81, which contains all the help files for the various applications 174 that may be configured in a package 54 for distribution to mobile digital devices 25.

Distribution and Activation of Terminals Created by the Engine Manager Terminal 22

Fig. 13 depicts the process used to prepare client and retail clerk terminals 24 and 26 of Fig. 1 to receive files to enable them to provide the service of distributing configured media packages 54 (Fig. 2) to mobile digital devices 25. The process starts with the engine manager terminal 22 creating the necessary files 83 for the client terminal 24 and the necessary files 84 for the retail clerk terminal 26. Once the necessary files 83 and 84 have been properly configured, all or part of the files may be compressed and encrypted. The files 83 and 84 are then delivered to the customers 86 (e.g., the operators of the retail clerk terminals 26 and/or the client terminals 24) via a connection through network 28 (e.g., by the customer's access to a web site 85 through the Internet). After delivery of the files 83, 84 to the customers 86, the client and/or clerk terminals 24, 26 must then be activated, as described with reference to Fig. 14 and Fig. 15, so that the terminals can perform the distribution of configured media packages 54 to mobile digital devices 25.

Activating a Terminal for Use to Enable Distribution of Configured Media Packages

Fig. 14 and Fig. 15 are flow diagrams depicting the process of activating an engine manager terminal 22, a client terminal 24, or a retail clerk terminal 26 for a transaction with the main server system 12 such as performing the distribution of configured media packages 54 (Fig. 2). In order to allow these transactions to begin in an orderly fashion, the main server system 12, needs to identify the terminal 22, 24, or 26 it is activating.

The difference between Fig. 14 and Fig. 15 is centered around how the terminal's identity is created. The engine manager terminal 22 gets its identity directly from the main server system 12 that created it. The other terminals, the client terminals 24 and the retail clerk terminals 26, get their identity from the engine manager terminal 22 that creates them at the time they are created.

Fig. 16 depicts a window displayed to the operator of the engine manager terminal 22 for inputting information to create a client terminal 24 or the retail clerk terminal 26. The operator of the engine manager terminal 22 inputs known information into various fields in the window (e.g., the name of the terminal owner (field 87) and the company that owns the terminal (field 88)). The information entered by the operator is registered by the main server system 12 (as shown at 24 in Fig. 5). This information is referred to herein as patch-data because it is patched into (attached to) the instructions of the new client or retail clerk terminal 24, 26 for additional use such as allowing the main server system 12 to identify the terminal that is requesting interactions.

Referring to Fig. 1 and Fig. 14, the process to create this identity and provide the main server system 12 the information to identify the engine manager terminal 22 that is requesting a transaction can be described. In Fig. 14, the control terminal 20, in step 304, retrieves necessary information (the terminal identifier and the user/administrator unique identifiers, which are combined and stored in a file (cookie) on the terminal 22) and this information is used to verify legal ownership of the terminal 22.

If legal ownership at step 304 is not established (e.g., invalid or out of date information in the cookie) the terminal 22 displays an error message and proceeds to step 309 protecting any further access to usage from unknown parties and or unauthorized engine(s) by displaying an error message and exiting.

If legal ownership of the terminal 22 is established by finding the correct registry information in the cookie at step 304, the control terminal 20 proceeds to step 306, verifying registry information. If registry information is found the terminal 22 is allowed to pass to the next step 310.

If at step 306 there is no registry information (e.g., cookie including user/administration identification and engine identification) found, the control terminal 20 proceeds to step 306b and begins the process for a first time run of terminal 22, which upon completion will have gathered the necessary information to

create a the registry information cookie. The first time run of terminal 22 is established at step 306b by feeding information back to the main server system 12. Upon receiving a response from the main server system 12, the control terminal 20 updates log files (step 307) and writes information (e.g., user/administration identification and engine identification) into the registry creating a 'cookie' (step 308), thus preparing the new terminal 22 for activation.

Step 310 represents a last action before manager terminal 22 is allowed to access the main server system 12 and begin to distribute configured media packages 54 (Fig. 2) onto a mobile digital device 25 for use by an end user/purchaser. If flag (BenchBlocked) is true as determined in step 310, an indication 23 is sent as shown in Fig. 4 that for some reason the terminal 22 has been intentionally blocked by the main server system 12, and the server system 12 is directed to proceed to step 309 and display an error message on engine manager terminal 22 and exit. If, in step 310, flag (BenchBlocked) is false, new log flags 18 (Fig. 4) are sent to the central server 12 and the process proceeds to step 311, which prepares the engine manager terminal 22 to perform distribution of configured media packages 54 onto a mobile digital device 25 for use by the end user/purchaser.

Figure 25a uses logic similar to that of Figure 25 with one significant exception. Figure 25a is a description of both a client terminal 24 and a retail clerk terminal 26, which are both created by the engine manager terminal 22 instead of the main server system 12, as depicted in Fig. 6. Step 320 in Figure 25a represents the first additional step that is caused by the terminals 24, 26 being created by the engine manager terminal 22. Step 320 checks for valid terminals 24, 26. Since the terminal has been created by a source other than the main server system 12, this is the first opportunity for the terminals 24, 26 to become registered and known to the main server system 12 by creating a 'cookie' which comprises the identifiers of the terminals 24, 26, the engine and the user/administrator of terminals 24, 26. These unique identifiers are combined and stored in a file on the terminals 22, 24, and 26. This cookie is used to verify legal ownership of the Terminals 24, 26 to the main server system 12 and the control terminal 20 in future transactions.

As previously described with reference to Fig. 16, this information is manually input into the data input fields 87, 88 of the interface screen shown in Figure 13 by the administrator of the engine manager terminal 22. In turn, the information will be read by the control terminal 20 for registration (Fig. 5) by the main server system 12. This

information is known as patch-data because it is patched into the instructions of the new terminal 24, 26 for additional uses such as allowing the main server system 12 to identify the terminal 24, 26 that is requesting interactions. This patch-data is then registered with the main server system 12 for logging and validation use when the terminals 24, 26 attempt to be activated for the first time.

In Figure 25a, step 320 is the interaction between the newly created terminals 24, 26 and the control terminal 20 to determine validity of the terminals 24, 26. If patch-data is found in step 320, the control terminal 20 continues to gather information in steps 322-331 to initiate and validate the terminals 24, 26 by combining the patch-data with the other information. The engine identifier and the user/administrator unique identifiers are combined to create a cookie, which is then stored in a file on the terminals 24, 26 at the time the terminals are verified and activated for use in step 331. This new 'cookie' is then used to verify legal ownership of the terminals 24, 26 to the main server system 12 and the control terminal 20 in future transactions.

If the patch-data flag is negative as determined in step 320, the control terminal 20 indicates by displaying an error message on the terminal 26, 26 and immediately proceeds to step 332, which prevents any further access to usage from unknown parties and/or unauthorized engine(s) by displaying an error message and exiting.

The following describes the step-by-step activity of the validation process for terminals 24, 26. If a patch-data flag is set true in step 320, the Control Terminal (20) proceeds to the next step 322 to verify the validity of the patch data. If a patch-data-validity flag is negative at step 322 (i.e., the patch data is invalid), the control terminal 20 displays an error message on the terminal 24, 26 and immediately proceeds to step 332, which protects any further access to usage from unknown parties and/or unauthorized engine(s) by displaying an error message and exiting.

If, at step 322, the patch-data-validity flag is true (i.e., the patch data is valid), the control terminal 20 moves to the next step 324 to determine if user information is required. If user information is required at step 324, the control terminal 20 proceeds to steps 325 to compare the user information to the patch data. If the user information does not match the patch data in step 326, the control terminal 20 displays an error message on the terminal 24, 26 and immediately proceeds to step 332, which protects any further access to usage from unknown parties and/or unauthorized engine(s) by

displaying an error message and exiting. If, at step 326, the user information matches the patch data or if, at step 327, user information is not required, then the control terminal 20 advances to step 327, and reads the registry information (cookie). If the registry information is found in step 328, the terminal is allowed to pass to the next step 330.

If no registry information is found in step 330, the terminal 20 proceeds to step 328a, which processes data for the first time run of the terminal 24, 26. First time run of terminal 24, 26 is established at step 328b, wherein terminal 20 feeds information back to the main server engine 16 to update log files and write information into a registry (cookie) in step 328c, thus preparing the new terminal 24, 26 for enablement. The terminal 20 then moves the process to step 330.

Step 330 represents a last action before terminal 22, 24 is allowed to access the Main Server System 12 and begin to perform distribution of a configured media package 54 (Fig. 2) to the mobile digital device 25 for use by the end user/purchaser. If a BenchBlocked flag 23 (as shown at 23 of Fig. 4), this indicates that for some reason the terminals 24, 26 have been intentionally blocked by the main server system 12. In that case, the control terminal 20 is directed to proceed to step 332, which displays error messages on the terminals 24, 26 and exits. If the BenchBlocked flag is false (as indicated at 18 of Fig. 4), then the process proceeds to step 331, where the terminals 24, 26 are prepared to distribute configured media packages 54 (Fig. 2) to the mobile digital device 25 for use by the end user/purchaser.

Installation and Configuration of Media Packages to a Mobile Digital Device

Fig. 17 is a flow chart depicting the process of installing and configuring the media package 54 onto the mobile digital device 25 by after the engine manager terminal 22, client terminal 24, and/or retail clerk terminal 26 have been activated by the main server system 12, as described above with reference to Fig. 14 and Fig. 15. Referring to Fig. 12 and Fig. 17, this process will be described. The first step 510 is to pick up the media package 54 from the package definition file 78 and proceed to step 514, which determines the presence of a compressed file in the engine manager terminal 22. The compressed file may be a cabinet (cab) file, which is a file that contains all of the drivers and window files needed for an application to run, and is a form of an executable from Microsoft Corporation that is common for use in the applications in the packages installed on the mobile digital devices 25. If no

compressed file is detected, the process is directed in step 516 to display an error message on the terminal 22, 24, or 26 and, thereafter, to move to step 540 to determine if there are additional files to install. If a cab file is detected at step 514, the terminal 22, 24, or 26 moves to step 518, and performs a function of loading the cab file to the mobile digital device 25 and, thereafter, proceeds to call a boot loader program (e.g., Windows CE Boot Loader available from Microsoft Corporation) at step 520 for preparation of the installation of the selected media/software application to the mobile digital device 25 and, then, proceeds to step 524.

In step 524 the terminal 22, 24, or 26 determines whether or not the smart menu 170 (Fig. 7) has been installed. If no installation has been detected, the terminal 22, 24, or 26 moves to step 540. If an installation of the smart menu 170 has been detected at step 524, a flag is set to true before the terminal 22, 24, or 26 proceeds to step 525 to include and store in the PDA registry the appropriate flags. Thereafter, the process proceeds to step 526, wherein the terminal 22, 24, or 26 determines if this is a demonstration installation. If the flag is true at step 526, indicating it is to be a demonstration installation, the terminal 22, 24, or 26 proceeds to step 530 to move the menu configuration and settings to the mobile digital device 25. The process then proceeds to step 531, where any additional menu files are moved to the mobile digital device 25, and proceeds to step 532. If at step 526 the flag is set false, indicating this is not a demonstration installation, then the terminal moves to step 527 and asks the operator of the terminal 22, 24, or 26 to download the key 50 for menu activation using the processes depicted in Fig. 2 or Fig. 3.

If at step 527 the operator of the terminal 22, 24, or 26 wishes to download the product key, the terminal moves to step 528, which downloads product key information for the menu, and then moves to step 529 to creates a product key file on the mobile digital device 25. The process proceeds to step 530, where the terminal 22, 24, or 26 moves the menu configuration and setting information to the mobile digital device, then to step 531 to move any additional menu files to the mobile digital device 25, and then to step 532.

If at step 527 the flag is set to false, indicating that the operator of the terminal 22, 24, or 26 does not wish to download the product key, the terminal 22, 24, or 26 proceeds to step 530 and thereafter to step 531 to move the menu configuration setting to the mobile digital device 25 before moving additional menu files to the device 25. The process continues to step 532 to determine if the menu auto start requested flag is

selected for the installation of this media package, which function can be enabled by an administrator accessing the terminal 22, 24, or 26. This function displaces the normal startup screen with the smart menu screen (e.g., any user interface 161 of Fig. 9).

If the menu start request flag is set to true in step 532, then the terminal 22, 24, or 26 proceeds to step 533 to install shortcuts to the menu startup folder and, thereafter, proceeds to step 534, which binds the menu shortcut to the home key on the mobile digital device 25. If the menu auto start step flag is set to false in step 532, then the terminal 22, 24, or 26 proceeds to step 540, where it determines whether there are more applications to install and, if so, sets its flag to true, causing the terminal 22, 24, or 26 to advance back to step 510. The processes continues until such time as flag 540 is set to false indicating that there are no more applications to add.

If, at step 540, a flag is set indicating there are no more applications to install, then the terminal 22, 24, or 26 proceeds to step 550 to determine if a user's manual is to be included with the media package 54. If the manual is to be included, a flag is set to true at step 550 and the process proceeds to step 551, where the terminal 22, 24, or 26 moves the user's manual file to the device 25. Thereafter, the process proceeds to step 552, where the terminal 22, 24, or 26 updates the package information on the device 25, which comprises the package name, version and included applications. If, at step 550, the flag is set to false indicating that no user's manuals are to be included in the media package, the process proceeds to step 552, where the terminal 22, 24, or 26 updates the media package information (e.g., the package name, version and included applications) on the mobile digital device 25.

The process proceeds to step 553 where the terminal 22, 24, or 26 retrieves complete installation information from the device 25 and update the engine window before proceeding to step 560 to complete the distribution of configured media package 54.

Handling of Optional Packages

System 10 (Fig. 1) allows an administrator or operator of the terminal 22, 24, or 26 to add individual applications and/or optional packages of applications to any of the media packages as they are about to be delivered to the end-user/purchaser. Figure 14 shows the final package 54 configuration at installation time, which may

comprise a standard package 54 that has been selected for this installation, and any additional packages 54' and optional applications 174' to be added to the final package 54 for installation to the mobile digital device 25.

An example of the use of an additional package would be where the end-user/purchaser is a doctor, who has chosen a standard medical package 54. The standard package 54 would be downloaded as selected, but the Doctor, being an ophthalmologist, would like to have the additional package 54' made just for ophthalmologists added to the standard package 54 at time of download by the administrator of the terminal 22, 24, or 26.

Fig. 18 further shows the addition of optional applications 174'. A description of optional applications would be the doctor who wants to add a security application and a global positioning system (GPS) navigation application to the mobile digital device 25 at time of download. The present embodiment allows for the addition of such optional applications 174' to supplement the other applications 174 (see Fig. 8) included in the package 54 at time of download by the administrator of the terminal 22, 24, or 26.

Referring to Fig. 19 a graphical user interface window 200 is depicted, as would be displayed to the operator of the engine manager terminal 22, client terminal 24, or retail clerk terminal 26 to allow the configuration of a package 54 for downloading onto the mobile digital device 25. The left side of window includes a pair of buttons 201, 203 that allow the operator to add or remove, respectively, standard packages 54 (e.g., those packages created as described in Fig. 8) to the final package 54 to be delivered to the device 25. Upon selection of the add button 201, the operator is presented with a window (not shown) from which the operator can select and browse (search) for available standard packages. Upon selecting a standard package, the name of the standard package appears in a selectable field 202. Packages can be removed from this field 202 (and from the final package 54) by highlighting the name of the unwanted package in field 202 and selecting the delete button 203.

Upon selecting a standard package name in field 202, the name and version of the standard package appears in fields 204, and the applications 174 included in the standard package are listed in a field 206. Below field 206 are a plurality of selection boxes and radio buttons 208 that allow the operator to set flags applicable to the smart menu 170 (Fig. 7) associated with the package being configured. These flags are

applied in the installation process described with reference to Fig. 17, and include: an "include menu configuration flag", which directs the installation process (at step 540 of Fig. 17) whether the smart menu 170 is to be included for the package; a "lock menu configuration" flag that directs the installation process (at step 525 of Fig. 17) to apply settings in the mobile digital device 25 that lock the smart menu 170 as the user interface for the mobile digital device 25 (e.g., to allow the user of the device 25 to access only those features associated with the smart menu 170); a "menu autostart" flag, which directs the installation process (at step 532 of Fig. 17) to apply settings in the mobile digital device 25 to initiate the smart menu 170 upon start-up of the mobile digital device 25; and a "mask start menu" flag that directs the installation process (at step 525 of Fig. 17) to apply settings in the mobile digital device 25 to mask the start menu.

The window 200 also includes selection items 210 that allow the operator to select whether the final package 54 is to be a standard installation (i.e., including those applications in the standard package), or whether the final package 54 is to include optional packages 54' and additional applications 174'.

To the right of field 202 in window 200 is a selection field 212, which lists the names of applications that are to be installed in the final package 54. If the operator chooses to include additional applications 174', these additional applications 174' may be added or removed using buttons 214.

Upon highlighting the name of an application in field 212, fields 214 are populated with the name, version number, and a description of the highlighted application. Also, the vendor of the highlighted application appears in a field 216, and a button 218 associated with field 216 allows the vendor information to be updated.

If the operator chooses to include documentation (e.g., a user's manual) associated with the application highlighted in field 212, the file path pointing to the documentation file can be entered into field 220, or a button 222 can be selected to browse for the file. Similarly, the file path pointing to the cabinet file including the highlighted application can be entered in a field 224, or a button 226 can be selected to browse for the application.

To the right of field 212 is a selection field 228, which lists the vendors associated with the various applications in the final package 54. Various buttons and

fields associated with field 228 allow information for the various vendors listed in the field 228 to be entered, updated and deleted.

The lower right hand portion of window 200 includes a button 230, which initiates the key creation routines of Fig. 2 or Fig. 3 for the final package 54, and a button 232, which allows the operator to view the keys associated with the final package 54.

Examples

The following examples highlight the features and advantages of the system 10 for various applications and by no means are intended to be limiting.

Retail Application

In a first example, the retail clerk terminal 26 makes it easy for a retail clerk to provide users and myriad business verticals with mobile digital devices 25 configured for their individual profession (or category of users) including applications that meet their specific needs with user interfaces that are meaningful to the users and their professions. The retail clerk does not need to have any technical training, does not need to understand the business vertical they are selling to, nor do they need to know how to use the portable digital devices 25.

The clerk simply establishes a communications link between a mobile digital device 25 and the retail clerk terminal 26 (e.g., by a wired or wireless connection either directly to the terminal 26 or by way of the network 28) and enters the customer name, email address and selects the media that relates to the business vertical requested by the customer from a list on a computer screen (business verticals such as Lawyer, Doctor, Sales, Financial, etc.). The clerk then follows the onscreen instructions to offer additional applications to further personalize the device 25. For example, as previously described with reference to Fig. 18, if the customer is a doctor and his/her practice is ophthalmology, the standard media package 54 may be supplemented to include an additional media package 54' for eye doctors.

Once downloading of the media package 54 is completed, the user experiences immediate gratification. The mobile digital device 25 is configured with applications 174 (Fig. 8) that are useful to the user, and a user interface 161, as shown in Figure 15, that is meaningful to the user and removes the necessity of a user manual/training material.

Corporate Application

Corporations seeking a solution adapted to its unique corporate needs can use a client terminal 24 to provide a consistent mobile environment for the corporation's employees and customers. The client terminal 24 augments the ability to distribute configured media packages 54 with the ability to prototype packages 54 and build out specific configurations within their own environment.

Using the same method detailed in Fig. 8, a package designer would be able to select from pre-configured media packages 54 and add to them or begin from scratch and build user interfaces 161 (Fig. 7) with a selection of icons 163 and applications (media) 174 (Fig. 8) adapted to particular corporate clients and supplied by a main server system 12. The only function that the client terminal (24) cannot perform is the ability to encode other terminals with programmed functionality.

For example, given a case where a company that provides document imaging services to a diverse market is looking to incorporate mobility and recognizes that there are many areas they can use mobile digital devices 25. For example, the company would like to have its sales force be able to access their corporate email as well as have secure access to the rest of the enterprise information to enable them to function in real time. The company also would like to provide the sales force with additional tools to improve their sales performance such as a personality profiler that provides a psychological view of the customer and how to best motivate them. Additionally, they want to be able to secure their mobile digital devices 25 and control their access remotely in case of theft or loss. Finally, because they have an on-line business with the need to display data in real time, the company wants to provide a secure gateway for viewing and demonstration of the site to customers anytime and anywhere they have a sales opportunity.

By selecting a media package 54 as a foundation and loading it to a prototype mobile digital device 25, as shown in Figure 12, the package designer is able to add an additional button on a smart menu 170 as shown in Figure 7 for the secure gateway to allow the sales professionals the ability to demonstrate their site anytime they have the sales opportunity. Further, the smart menu 170 may use existing buttons to connect the company's sales force to the company's systems. Because the package designer is simply modifying an existing package, the time to design is minimized and the outcome is more assured.

In another example, a customer would like to provide the real time tracking of assets and reporting of locations and deliveries using barcode scanning devices configured on a mobile digital device 25 but are concerned that their truck driver will be able to change the programmed functionality and access the internet or otherwise improperly use their mobile digital device 25. In addition to creating an standardized interface that limits the driver's use of the mobile digital device 25, the customer also wants to use a menu locking option to lock the operation of the mobile digital device 25 to their specifications, thus assuring that unauthorized users cannot alter it or access any other portion of the mobile digital device 25 for other uses until authorized.

This problem is solved by selecting a basic package 54 as a foundation and loading it to a prototype mobile digital device 25. The package designer is then able to remove/add additional buttons on the smart menu 170 as shown in Figure 7 to provide access to the desired functionality downloaded into the mobile digital device 25. Next, the package designer may choose to change or add the pictures/icons 163 to simplify their use and better match the internal symbols that the users will relate to and, in doing so, will eliminate training needs. Finally, once the company is satisfied with the design, the package designer uploads the package back to the client terminal 24 as shown in Figure 12 and selects the functions on the client terminal 110 to lock the smart menu 170 and the mobile digital device 25, as shown in Fig. 19, so that it cannot be used or modified by anyone other than the authorized administrator/designer.

Aspects of the present invention can be embodied in the form of computer-implemented processes and apparatuses for practicing those processes. The present invention can also be embodied in the form of computer program code containing instructions embodied in tangible media, such as floppy diskettes, CD-ROMs, hard drives, or any other computer-readable storage medium, wherein, when the computer program code is loaded into and executed by a computer, the computer becomes an apparatus for practicing the invention. The present invention can also be embodied in the form of computer program code, for example, whether stored in a storage medium, loaded into and/or executed by a computer, or transmitted over some transmission medium, such as over electrical wiring or cabling, through fiber optics, or via electromagnetic radiation, wherein, when the computer program code is loaded into and executed by a computer, the computer becomes an apparatus for practicing

the invention. When implemented on a general-purpose microprocessor, the computer program code segments configure the microprocessor to create specific logic circuits.

It should be understood that any of the features, characteristics, alternatives or modifications described regarding a particular embodiment herein may also be applied, used, or incorporated with any other embodiment described herein.

Although the invention has been described and illustrated with respect to exemplary embodiments thereof, the foregoing and various other additions and omissions may be made therein and thereto without departing from the spirit and scope of the present invention.

Claims

What is claimed is:

1. A method of configuring a mobile digital device, the method comprising:

selecting a first plurality of software applications from different vendors;

selecting a first plurality of user interface features indicative of the first plurality of applications;

associating applications in the first plurality of applications with corresponding user interface features in the first plurality of user interface features to create a first media package including the first plurality of applications and the first plurality of user interface features; and

loading the first media package on the mobile digital device to configure the mobile digital device, wherein the configured mobile digital device executes the applications in the first plurality of applications in response to selection of the corresponding user interface features in the first plurality of user interface features by a user of the mobile digital device.

2. The method of claim 1, wherein the first plurality of applications and the first plurality of user interface features are associated with a vertical market.

3. The method of claim 1, further comprising:

selecting at least one additional media package from a group of media packages, the at least one additional media package including a second plurality of software applications and a second plurality of user interface features indicative of the second plurality of applications; and

adding the at least one additional media package to the first media package before loading the first media package on the mobile digital device.

4. The method of claim 3, further comprising:

selecting at least one optional software application from a plurality of optional software applications;

adding the at least one optional software application to the first media package before loading the first media package on the mobile digital device.

5. The method of claim 1, further comprising:

associating a user's manual file with at least one application in the plurality of applications and adding the user's manual file to the first media package before loading the first media package on the mobile digital device..

6. The method of claim 1, further comprising:

setting the mobile digital device to prevent the user's access to user interface features other than those included in the first media package.

7. The method of claim 1, further comprising:

setting the mobile digital device to display the user interface features upon start up of the mobile digital device.

8. The method of claim 1, further comprising:

creating a key for the first media package;

creating an end-user account associated with the key, the end-user account identifying the user, the mobile digital device, and the applications in the first plurality of applications; and

loading the key on the mobile digital device to allow the user to use the first plurality of applications.

9. The method of claim 8, wherein, when loaded on the mobile digital device, the key activates the first plurality of user interface features to allow the user to use the first plurality of applications.

10. The method of claim 8, further comprising:

preventing access to the first plurality of applications on another mobile digital device not including the key.

11. The method of claim 8, further comprising:

changing the key on the mobile digital device to prevent access by the user to the first plurality of applications.

12. The method of claim 11, wherein, changing the key on the mobile digital device deactivates the first plurality of user interface features to prevent the user from using the first plurality of applications.

13. The method of claim 11, wherein the key is changed through a wireless connection to the mobile digital device.

14. The method of claim 8, further comprising:

using the end user account to identify an application in the first plurality of applications in need of upgrading; and

downloading an upgrade application to the mobile digital device to replace the application in need of upgrading.

15. The method of claim 8, further comprising:

reloading the applications indicated in the end-user account onto the mobile digital device in response to a request from the user.

16. The method of claim 8, wherein the end-user account further includes account information associated with the applications in the first plurality of applications.

17. The method of claim 8, further comprising:

using the key to identify transactions from the mobile digital device.

18. The method of claim 1, wherein the first media package is loaded on the mobile digital device by a terminal connected by a network to a main server system, the method further comprising:

enabling the terminal by providing software to the terminal from the main server system; and

in response to a request from the terminal for a key to the first media package, identifying and authenticating the terminal at the main server system before sending the key to the terminal.

19. The method of claim 10, wherein the terminal includes one of: a retail clerk terminal, a client terminal, and an engine manager terminal.

20. The method of claim 1, wherein loading the first media package on the mobile digital device is performed via the Internet.

21. A method of configuring a mobile digital device, the method comprising:

providing a media package including a plurality of software applications from different vendors, and a plurality of user interface features associated with the plurality of applications;

creating a key for the media package;

loading the media package on the mobile digital device;

loading the key on the mobile digital device to activate the plurality of user interface features to allow the user to use the plurality of applications.

22. The method of claim 21, further comprising:

preventing access to the plurality of applications on another mobile digital device not including the key.

23. The method of claim 21, further comprising:

changing the key on the mobile digital device to deactivate the plurality of user interface features to prevent the user from using the plurality of applications.

24. The method of claim 23, wherein the key is changed through a wireless connection to the mobile digital device.

25. The method of claim 21, further comprising:

creating an end-user account associated with the key, the end-user account identifying the user, the mobile digital device, and the applications in the plurality of applications;

using the end user account to identify an application in the plurality of applications in need of upgrading; and

downloading an upgrade application to the mobile digital device to replace the application in need of upgrading.

26. The method of claim 25, further comprising:

reloading the applications indicated in the end-user account onto the mobile digital device in response to a request from the user.

27. The method of claim 25, wherein the end-user account further includes account information associated with the applications in the plurality of applications.

28. The method of claim 25, further comprising:

using the key to identify transactions from the mobile digital device.

29. The method of claim 21, wherein loading the media package and the key on the mobile digital device are performed via the Internet.

30. A storage medium encoded with machine-readable computer program code for configuring a mobile digital device, the storage medium including instructions for causing a computer to implement a method comprising:

selecting a plurality of software applications from different vendors;

selecting a plurality of user interface features indicative of the plurality of applications;

associating applications in the plurality of applications with corresponding user interface features in the plurality of user interface features to create a media package including the plurality of applications and the plurality of user interface features; and

loading the media package on the mobile digital device to configure the mobile digital device, wherein the configured mobile digital device executes the applications in the plurality of applications in response to selection of the corresponding user interface features in the plurality of user interface features by a user of the mobile digital device.

31. A storage medium encoded with machine-readable computer program code for configuring a mobile digital device, the storage medium including instructions for causing a computer to implement a method comprising:

providing a media package including a plurality of software applications from different vendors, and a plurality of user interface features associated with the plurality of applications;

creating a key for the media package;

loading the media package on the mobile digital device;

loading the key on the mobile digital device to activate the plurality of user interface features to allow the user to use the plurality of applications.

32 A system for configuring mobile digital devices, the system comprising at least one terminal configured to:

select a plurality of software applications from different vendors;

select a plurality of user interface features indicative of the plurality of applications;

associate applications in the plurality of applications with corresponding user interface features in the plurality of user interface features to create a media package including the plurality of applications and the plurality of user interface features; and

load the first media package on the mobile digital device to configure the mobile digital device, wherein the configured mobile digital device executes the applications in the plurality of applications in response to selection of the corresponding user interface features in the plurality of user interface features by a user of the mobile digital device.

33. A system for configuring mobile digital devices, the system comprising at least one terminal configured to:

select a media package from a plurality of different media packages, the media package including a plurality of software applications from different vendors, and a plurality of user interface features associated with the plurality of applications;

create a key for the media package;

load the media package on the mobile digital device; and

load the key on the mobile digital device to activate the plurality of user interface features to allow the user to use the plurality of applications.

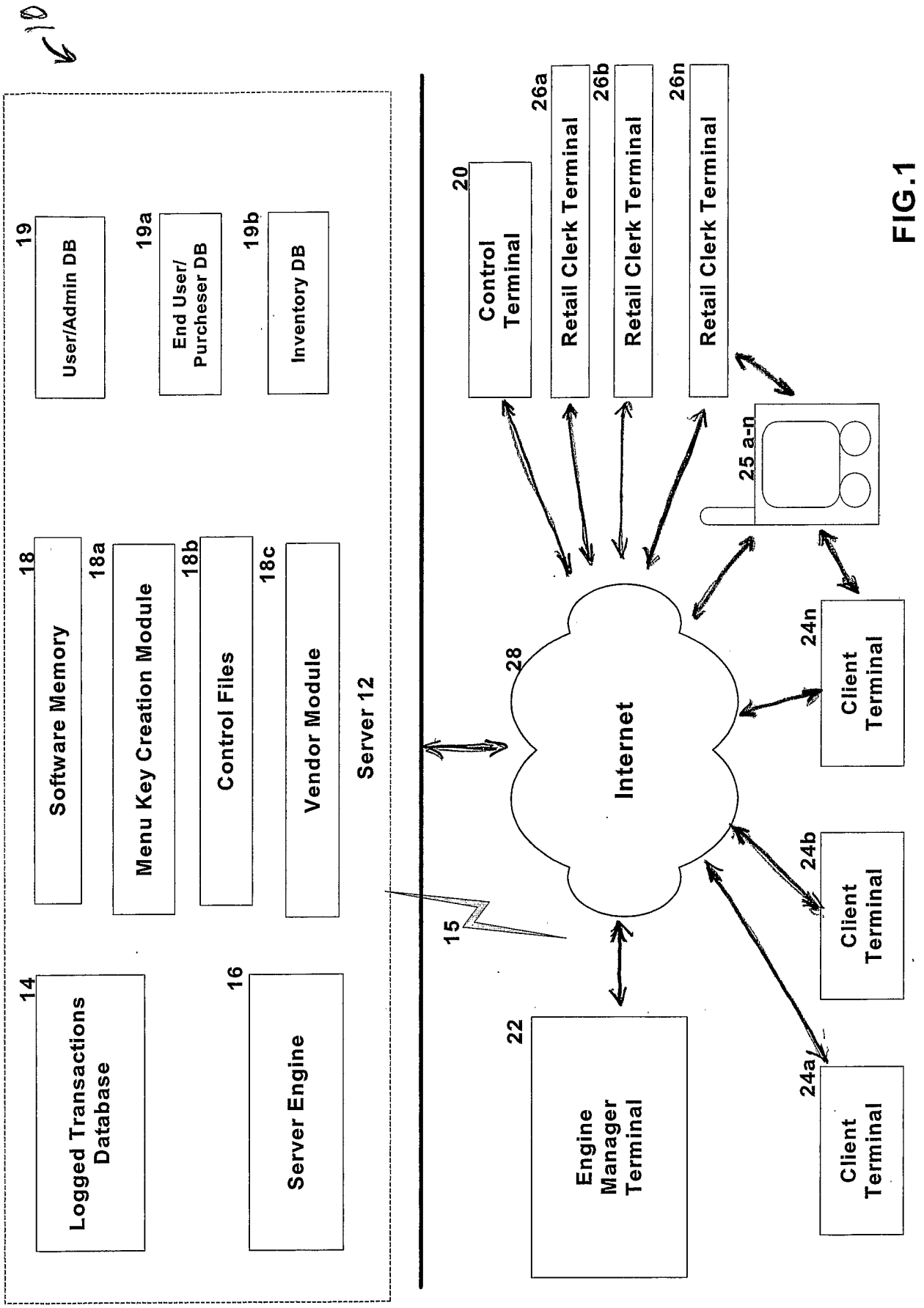

10
19
User/Admin DB
19a
End User/
Purcheser DB
19b
Inventory DB
18
Software Memory
18a
Menu Key Creation Module
18b
Control Files
18c
Vendor Module
Server 12
14
Logged Transactions
Database
16
Server Engine
15
20
Control
Terminal
26a
Retail Clerk Terminal
26b
Retail Clerk Terminal
26n
Retail Clerk Terminal
25 a-n
28
Internet
24n
Client
Terminal
24b
Client
Terminal
24a
Client
Terminal
22
Engine
Manager
Terminal

FIG.1

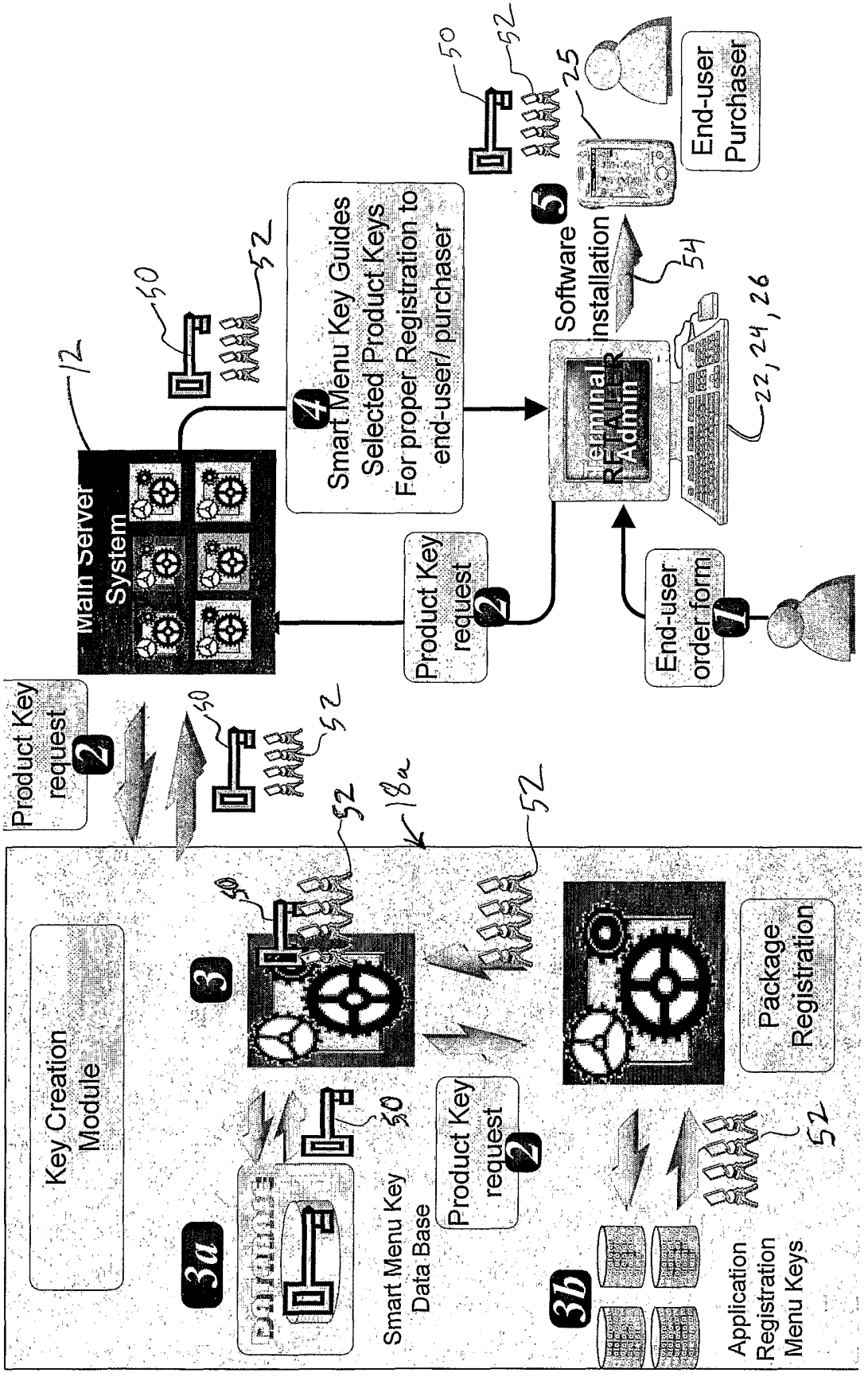

Key Creation Module
3a
Smart Menu Key Data Base
Product Key request
2
3
3b
Application Registration Menu Keys
Package Registration
Product Key request
2
Main Server System
12
4
Smart Menu Key Guides Selected Product Keys For proper Registration to end-user/ purchaser
5
Software installation
End-user order form
1
End-user Purchaser
50
52
54
25
22, 24, 26
18a

FIG. 2

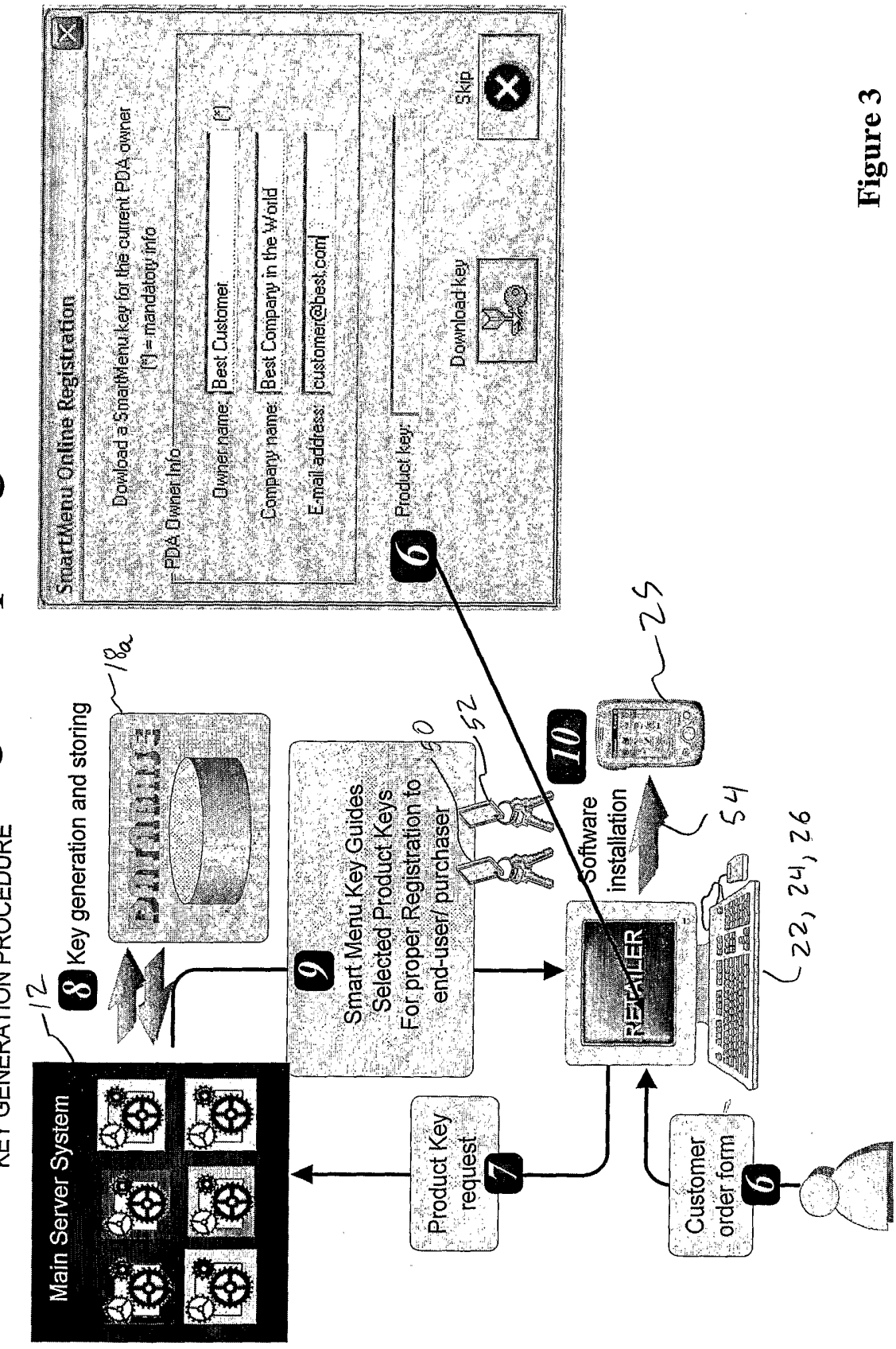

Single Step Registration
KEY GENERATION PROCEDURE
SmartMenu Online Registration
Download a SmartMenu key for the current PDA owner
[*] = mandatory info
PDA Owner Info
Owner name:
Best Customer
[*]
Company name:
Best Company in the World
E-mail address:
customer@best.com
Product key:
Download key
Skip
6
Main Server System
12
8 Key generation and storing
18a
9
Smart Menu Key Guides
Selected Product Keys
For proper Registration to
end-user/ purchaser
50
52
10
Software
installation
25
54
RETAILER
22, 24, 26
Product Key
request
7
Customer
order form
6

Figure 3

Registration and Transaction Control

REGISTRATION AND REPORTING

4. Control Transactions

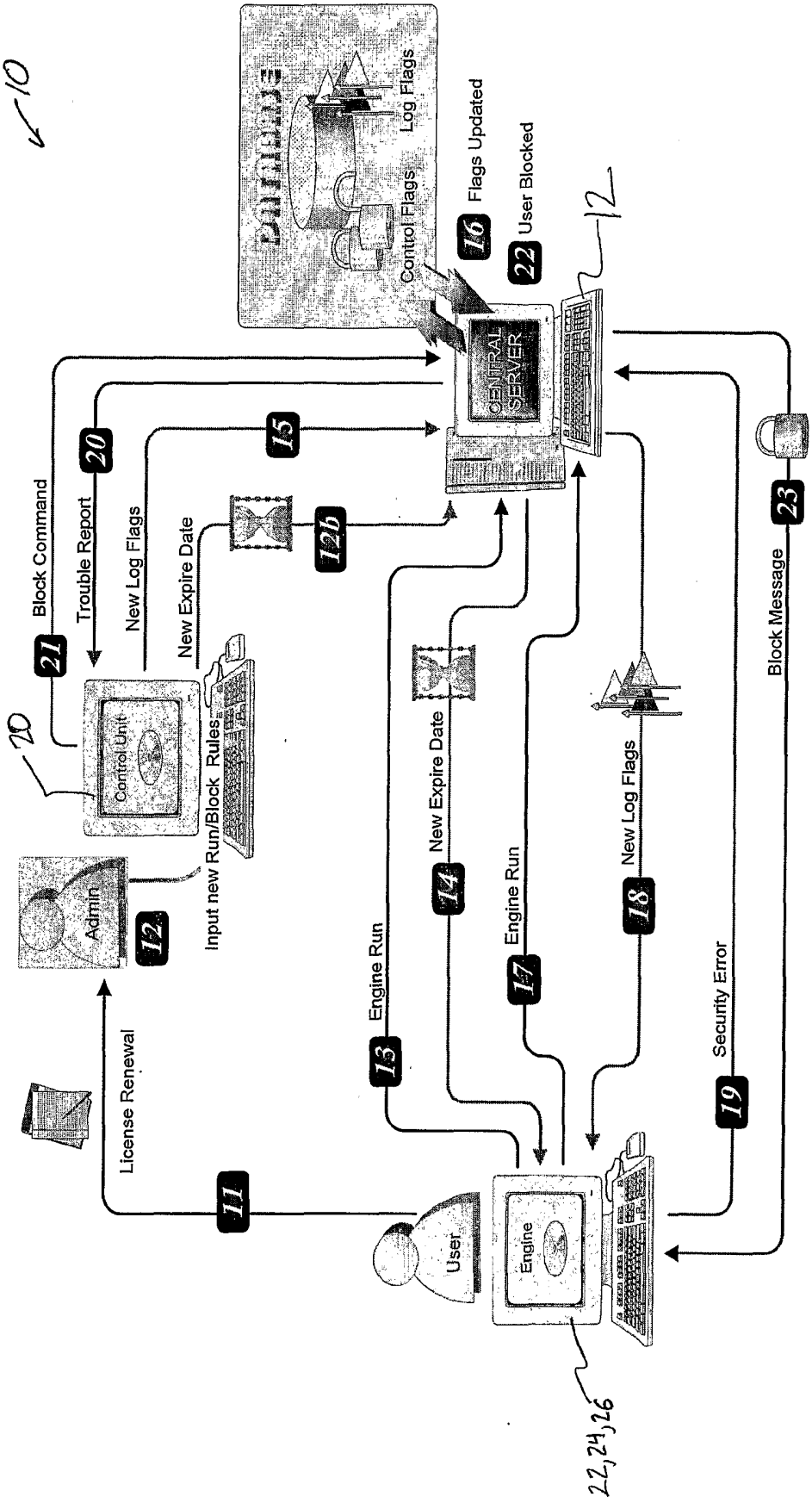


Fig. 4

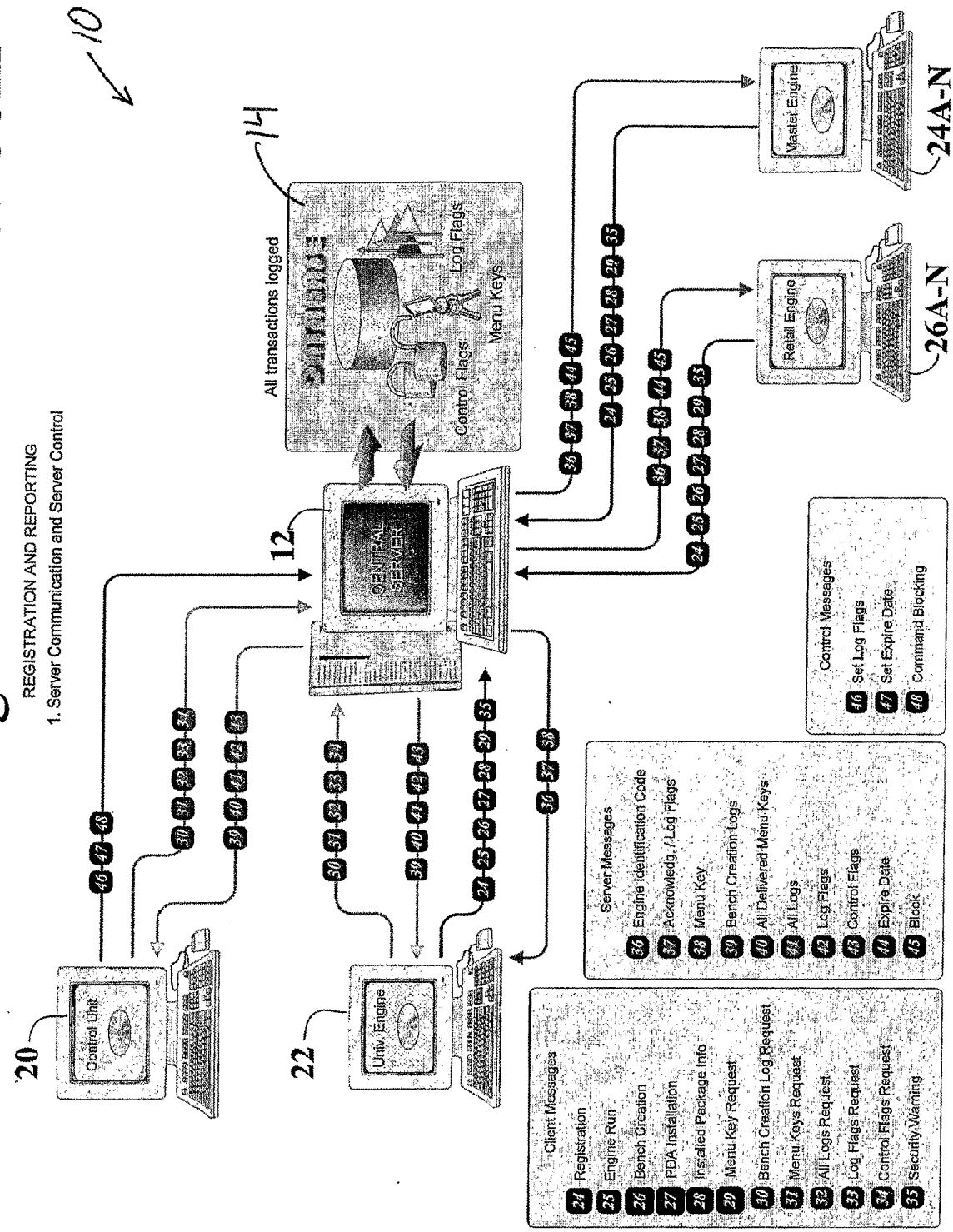


FIGURE 5

Fig. 5

FIG 6 Bench Creation

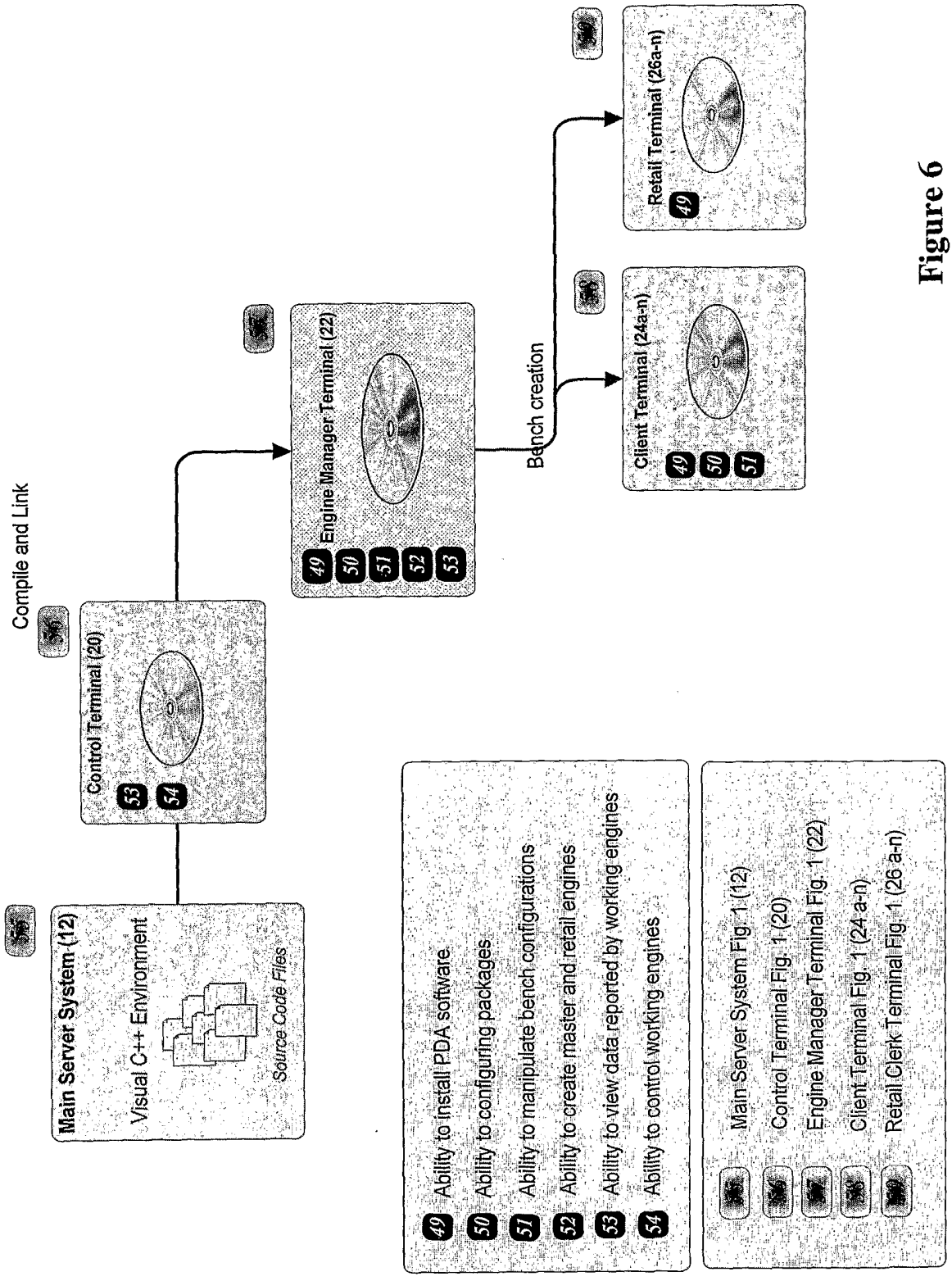


Figure 6

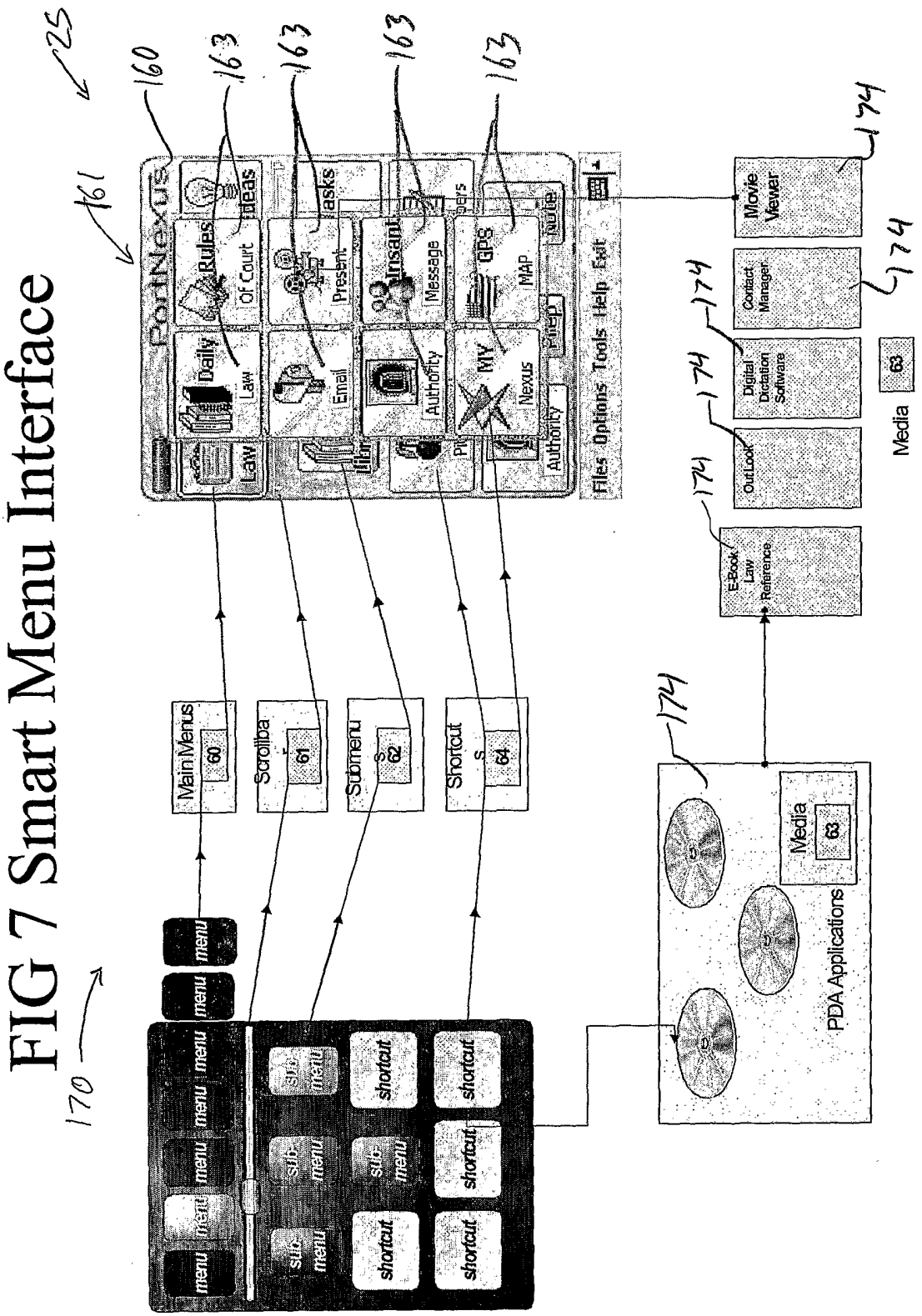


FIG 7 Smart Menu Interface
170
161
160
163
163
163
163
125
PortNexus
Daily
Law
Rules
Of Court
Ideas
Email
Present
Tasks
Authority
Message
Insant
MY
Nexus
GPS
MAP
Files Options Tools Help Exit
Movie Viewer
Contact Manager
Digital Dictation Software
OutLook
E-Book Law Reference
Media
63
174
MainMenus
60
Scrollba
61
Submenu
62
Shortcut
64
menu
shortcut
PDA Applications

Figure 7

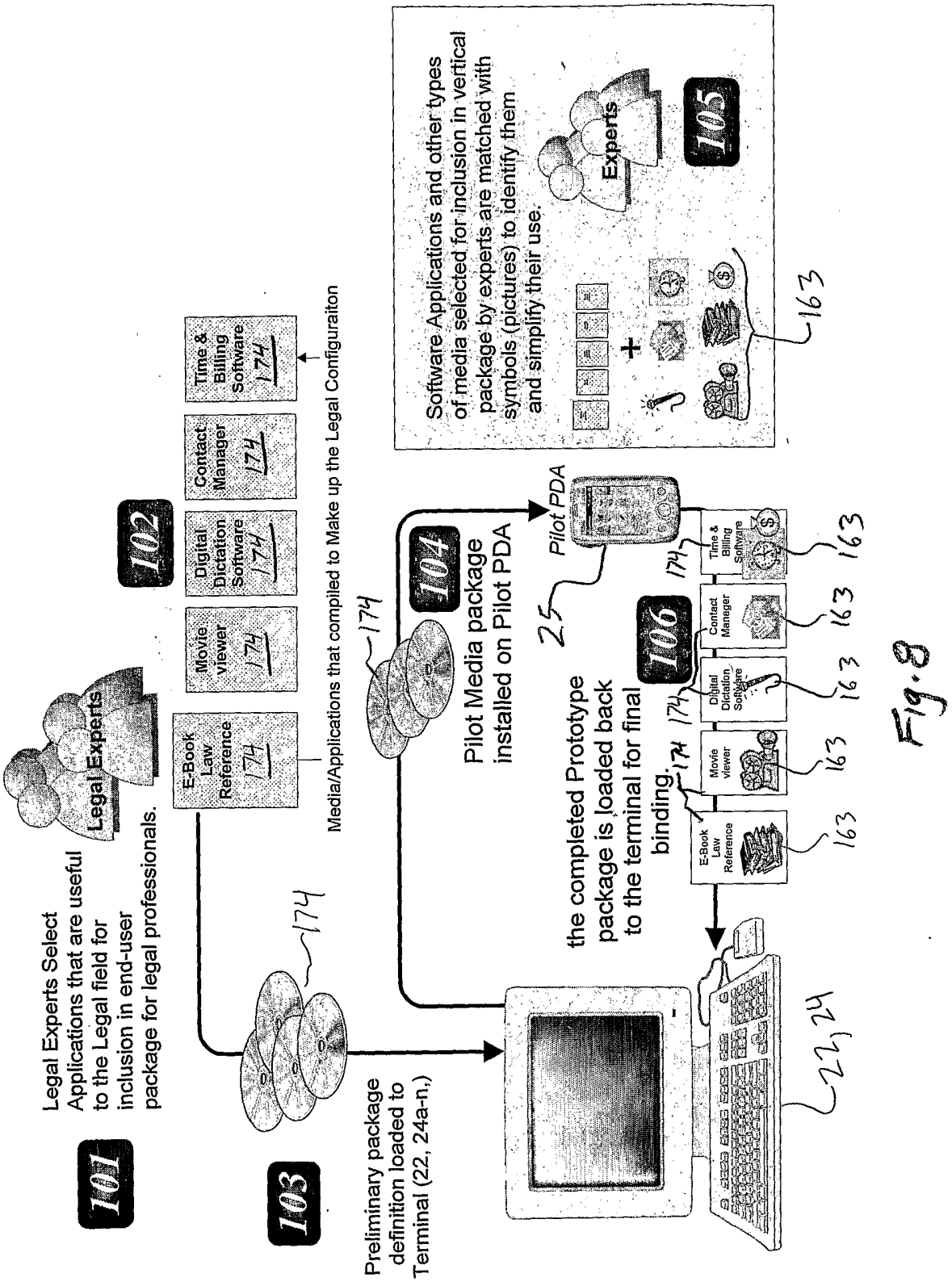
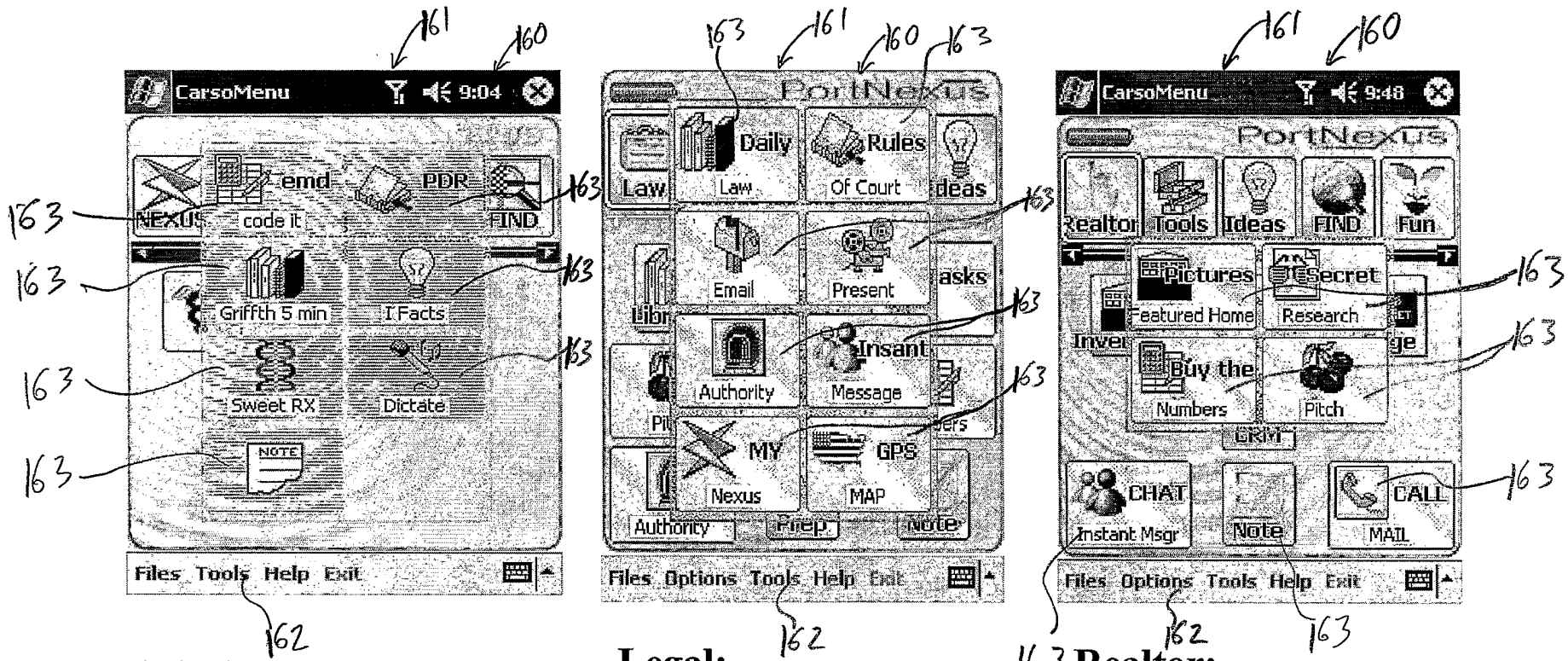

101
Legal Experts Select Applications that are useful to the Legal field for inclusion in end-user package for legal professionals.
Legal Experts
102
E-Book Law Reference 174
Movie viewer 174
Digital Dictation Software 174
Contact Manager 174
Time & Billing Software 174
Media/Applications that compiled to Make up the Legal Configuraiton
103
174
Preliminary package definition loaded to Terminal (22, 24a-n,)
104
Pilot Media package installed on Pilot PDA
Pilot PDA
25
the completed Prototype package is loaded back to the terminal for final binding.
106
174
E-Book Law Reference
Movie viewer
Digital Dictation Software
Contact Manager
Time & Billing Software
163
22, 24
Software Applications and other types of media selected for inclusion in vertical package by experts are matched with symbols (pictures) to identify them and simplify their use.
Experts
105

Fig. 8

Smart Menu Vertical Samples



Medical:

- **Physician Desk Reference**
- **Griffith's 5 Minute CC**
- **I-Facts**
- **ABC Lab Data**

Legal:

- **Daily Law Library**
- **Rules of Court**
- **Time & Billing**
- **Legal Dictionary**

Realtor:

- **Property Showcase**
  - Video, sound and pictures.
- **Know Your customer**
  - Inventory
  - Buyers/sellers

Fig. 9

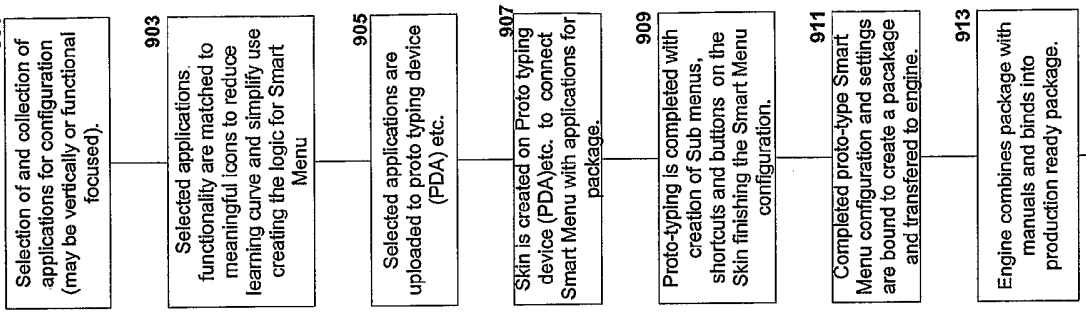

Proto type flow
901
Selection of and collection of applications for configuration (may be vertically or functional focused).
903
Selected applications. functionality are matched to meaningful icons to reduce learning curve and simplify use creating the logic for Smart Menu
905
Selected applications are uploaded to proto typing device (PDA) etc.
907
Skin is created on Proto typing device (PDA)etc. to connect Smart Menu with applications for package.
909
Proto-typing is completed with creation of Sub menus, shortcuts and buttons on the Skin finishing the Smart Menu configuration.
911
Completed proto-type Smart Menu configuration and settings are bound to create a pacakage and transfered to engine.
913
Engine combines package with manuals and binds into production ready package.
915
Terminate

Fig. 10

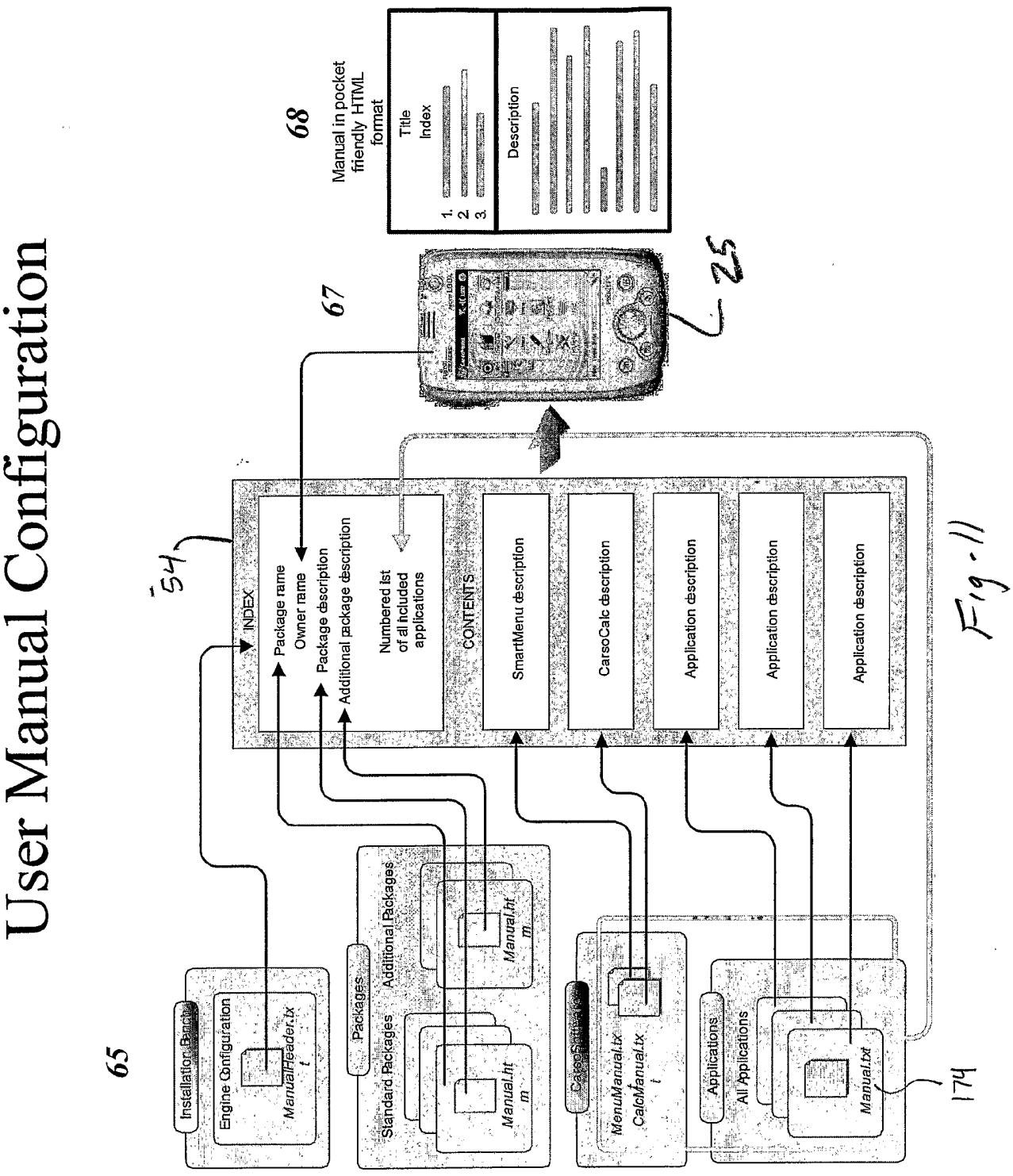

User Manual Configuration
65
Installation Bench
Engine Configuration
ManualHeader.tx t
Packages
Standard Packages
Additional Packages
Manual.ht m
Manual.ht m
CarsoCalc
MenuManual.tx t
CalcManual.tx t
Applications
All Applications
Manual.txt
174
54
INDEX
Package name
Owner name
Package description
Additional package description
Numbered list of all included applications
CONTENTS
SmartMenu description
CarsoCalc description
Application description
Application description
Application description
67
25
68
Manual in pocket friendly HTML format
Title
Index
1.
2
3.
Description

Fig-11

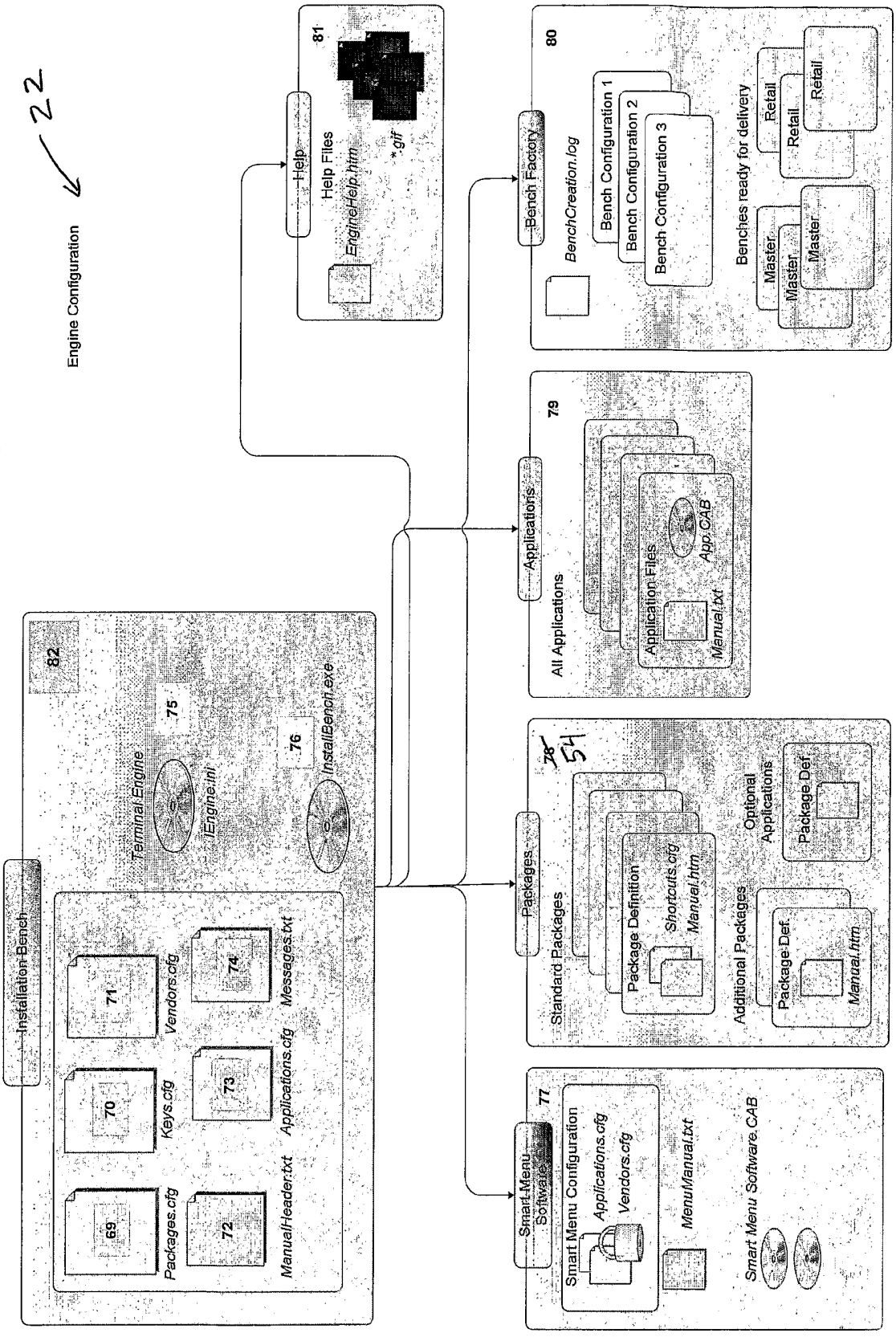

Engine Manager Terminal Directory
22
Engine Configuration
Installation Bench
82
Terminal Engine
75
TEngine.inl
76
InstallBench.exe
69
Packages.cfg
70
Keys.cfg
71
Vendors.cfg
72
ManualHeader.txt
73
Applications.cfg
74
Messages.txt
Help
Help Files
81
EngineHelp.htm
*.gif
Bench Factory
80
BenchCreation.log
Bench Configuration 1
Bench Configuration 2
Bench Configuration 3
Benches ready for delivery
Master
Retail
Applications
79
All Applications
Application Files
App.CAB
Manual.txt
Packages
78
54
Standard Packages
Package Definition
Shortcuts.cfg
Manual.htm
Additional Packages
Optional Applications
Package Def
Smart Menu Software
77
Smart Menu Configuration
Applications.cfg
Vendors.cfg
MenuManual.txt
Smart Menu Software.CAB

Fig. 12

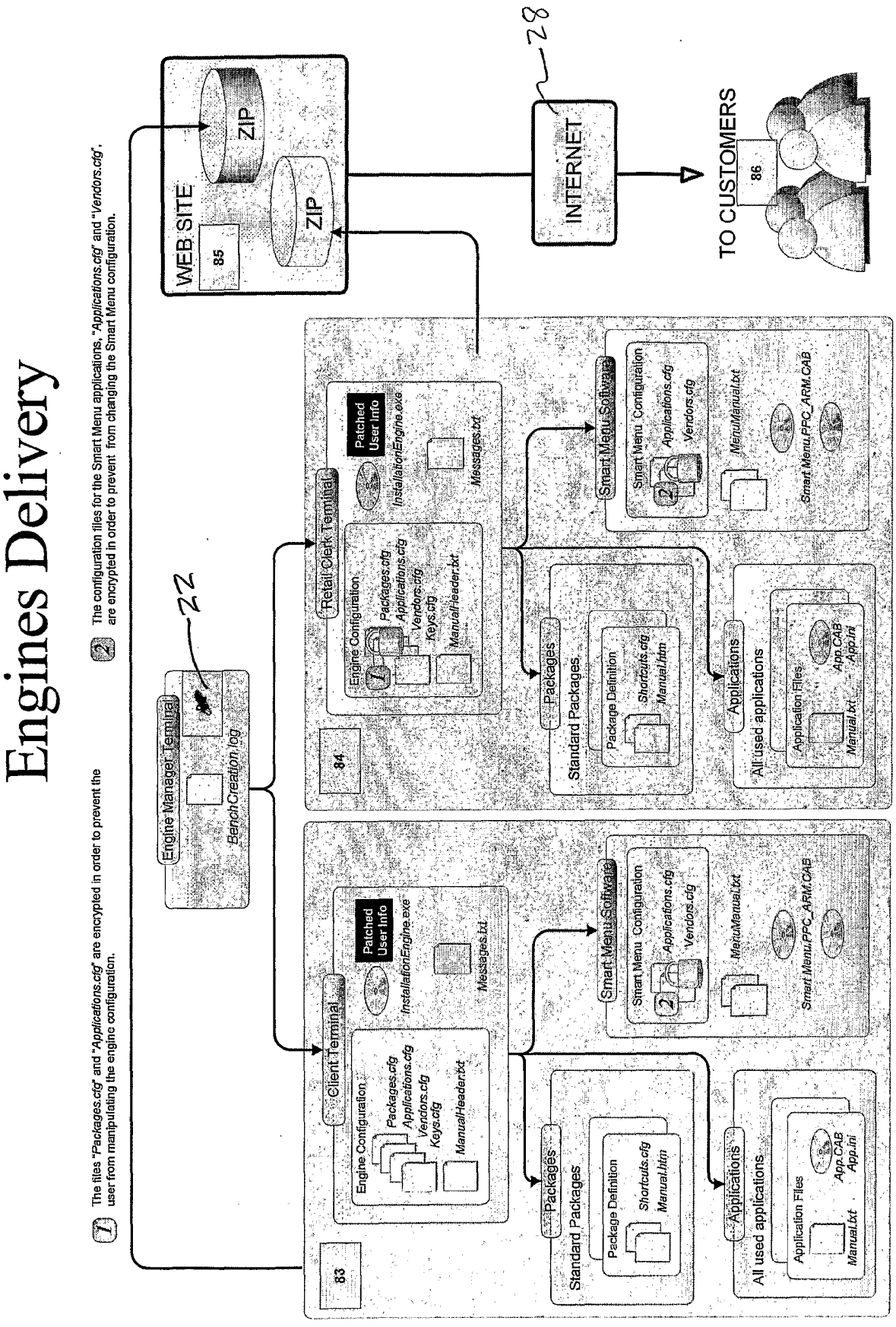

Engines Delivery
1 The files "Packages.cfg" and "Applications.cfg" are encrypted in order to prevent the user from manipulating the engine configuration.
2 The configuration files for the Smart Menu applications, "Applications.cfg" and "Vendors.cfg", are encrypted in order to prevent from changing the Smart Menu configuration.
Engine Manager Terminal
BenchCreation.log
22
WEB SITE
85
ZIP
ZIP
28
INTERNET
TO CUSTOMERS
86
Retail Clerk Terminal
84
Engine Configuration
Packages.cfg
Applications.cfg
Vendors.cfg
Keys.cfg
Manual-Header.txt
Patched User Info
InstallationEngine.exe
Messages.txt
Packages
Standard Packages
Package Definition
Shortcuts.cfg
Manual.htm
Applications
All used applications
Application Files
Manual.txt
App.CAB
App.ini
Smart Menu Software
Smart Menu Configuration
Applications.cfg
Vendors.cfg
MenuManual.txt
Smart Menu.PPC_ARM.CAB
Client Terminal
83

Fig. 13

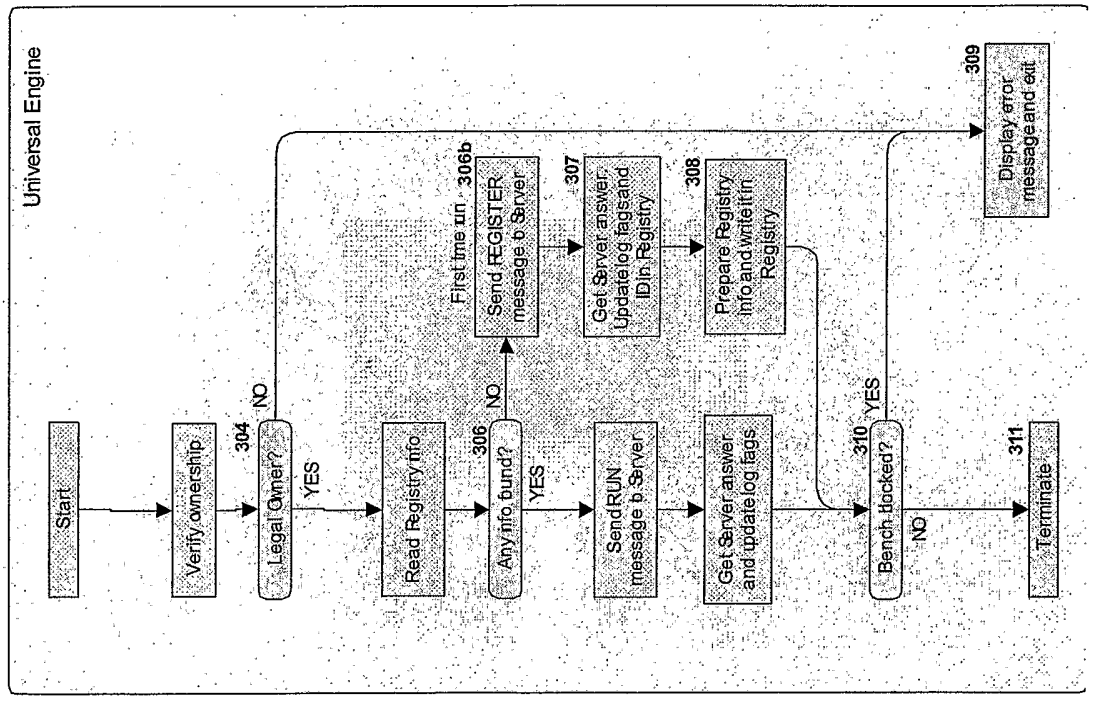

Universal Engine
Start
Verify ownership
304
Legal Owner?
NO
YES
Read Registry info
306
Any info found?
NO
YES
First time run
306b
Send REGISTER message to Server
307
Get Server answer, Update log tags and IDin Registry
Send RUN message to Server
308
Prepare Registry info and write it in Registry
Get Server answer and update log tags
310
Bench locked?
YES
NO
309
Display error message and exit
311
Terminate

Fig 14

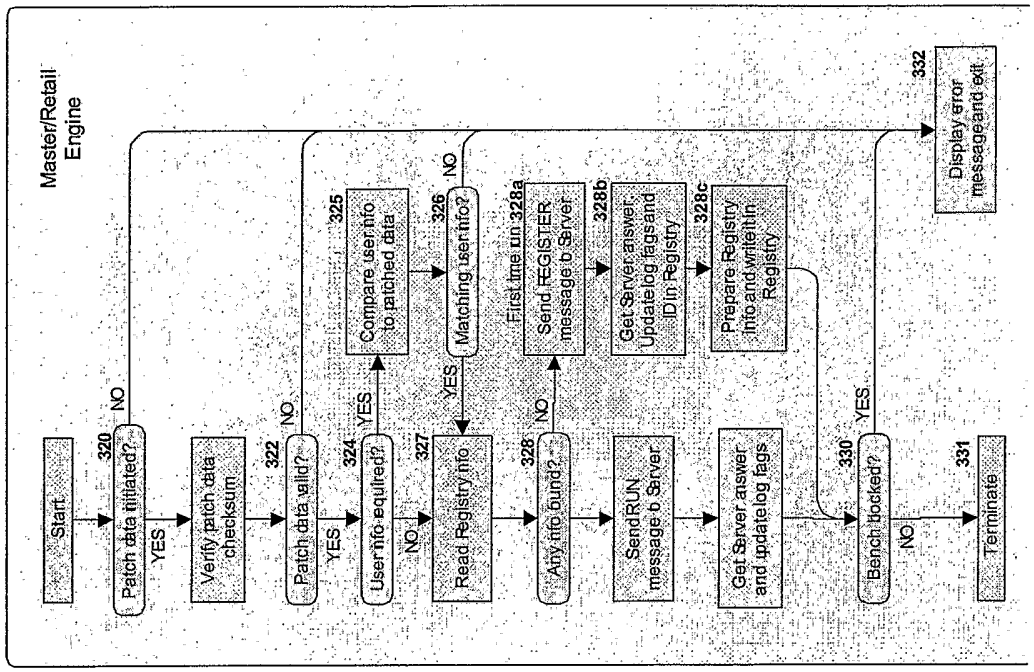

Master/Retail Engine
Start
320
Patch data initiated?
NO
YES
Verify patch data checksum
322
Patch data valid?
NO
YES
324
User info required?
YES
NO
325
Compare user info to patched data
326
Matching user info?
NO
YES
327
Read Registry info
First time run 328a
328
Any info found?
NO
YES
Send REGISTER message to Server
328b
Get Server answer, Update log tags and ID in Registry
328c
Prepare Registry info and write it in Registry
Send RUN message to Server
Get Server answer and update log tags
330
Bench blocked?
NO
YES
331
Terminate
332
Display error message and exit

Fig. 15

ENGINE PATCH DATA FOR MAKING ENGINES

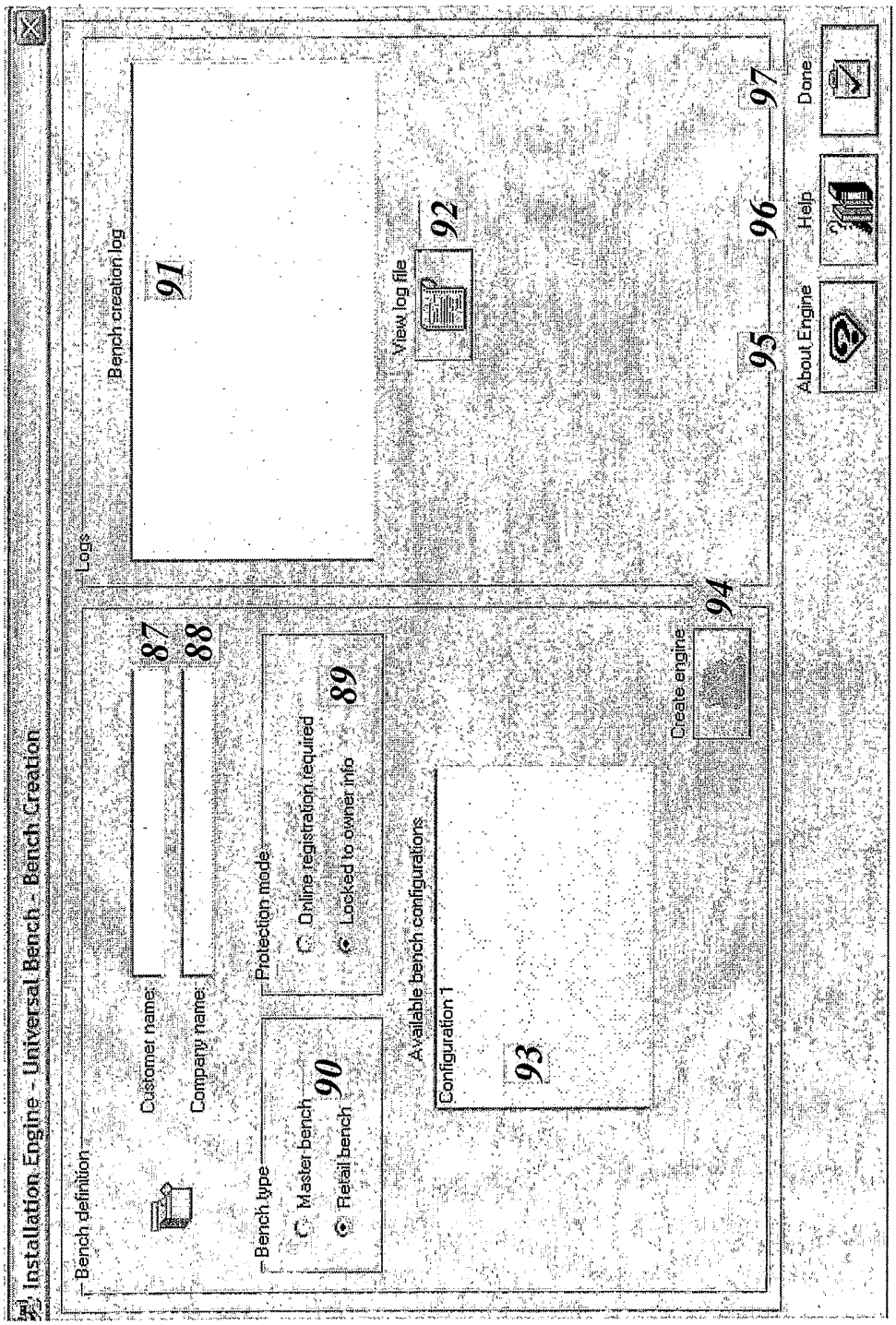


Fig. 16

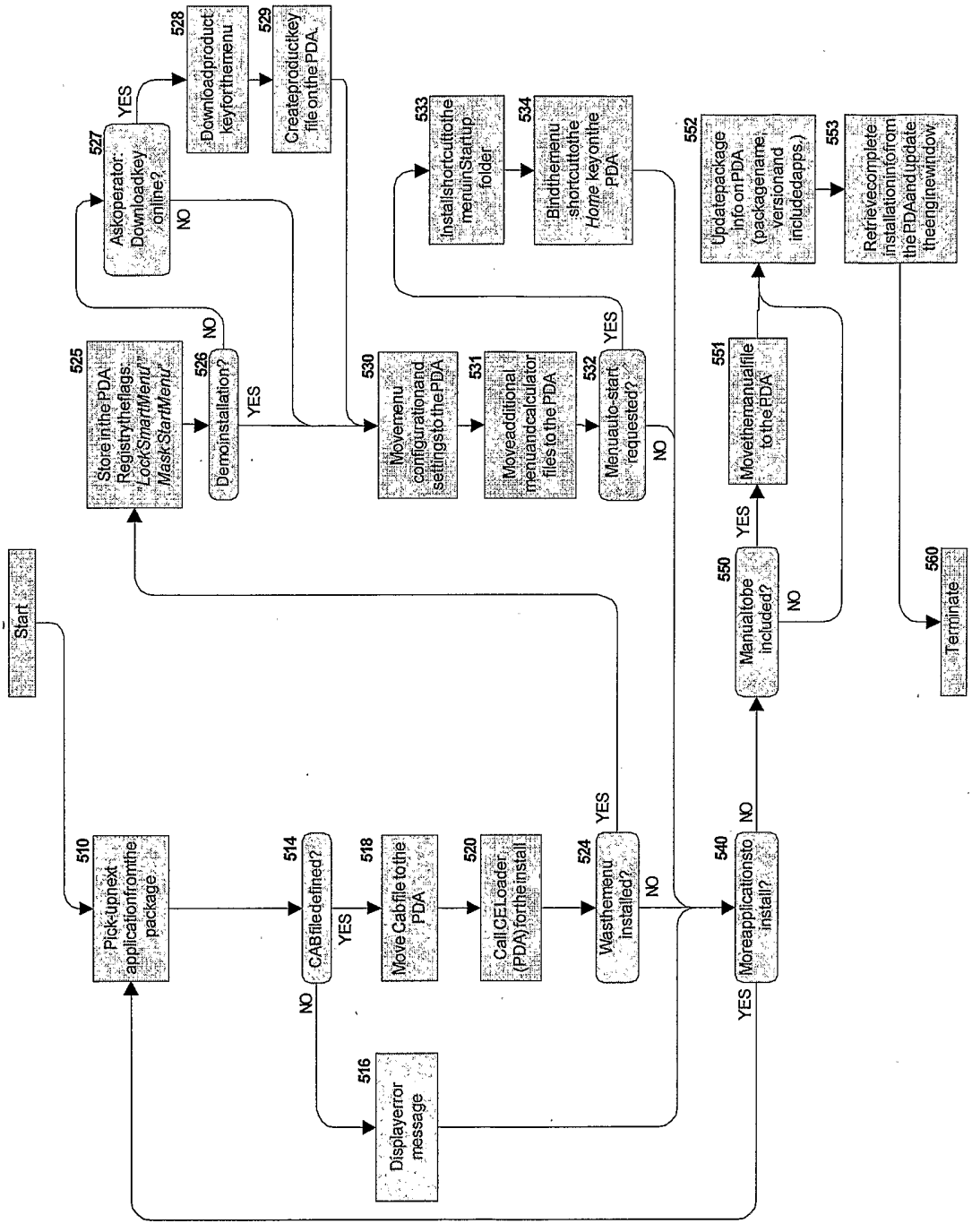

Start
510
Pick-Upnext applicationfromthe package.
514
CABfiledefined?
YES
NO
516
Displayerror message
518
Move Cabfile tothe PDA
520
Call CELoader (PDA) forthe install
524
Wasthemenu installed?
YES
NO
525
Store inthe PDA Registrytheflags *LockStartMenu* *MaskStartMenu*
526
Demoinstallation?
YES
NO
527
Askoperator: Downloadkey online?
YES
NO
528
Downloadproduct keyforthemenu
529
Createproductkey file onthe PDA.
530
Movemenu configurationand settingstothePDA
531
Moveadditional menuandcalculator files to the PDA
532
Menuauto-start requested?
YES
NO
533
Installshortcuttothe menuinStartUp folder
534
Bindthemenu shortcuttothe *Home* keyonthe PDA
540
Moreapplicationsto install?
YES
NO
550
Manualtobe included?
YES
NO
551
Movethemanualfile to thePDA
552
Updatepackage info onPDA (packagename, versionand includedapps.)
553
Retrievecomplete installationinfofrom thePDAandupdate theengine'swindow
560
Terminate

Fig. 17

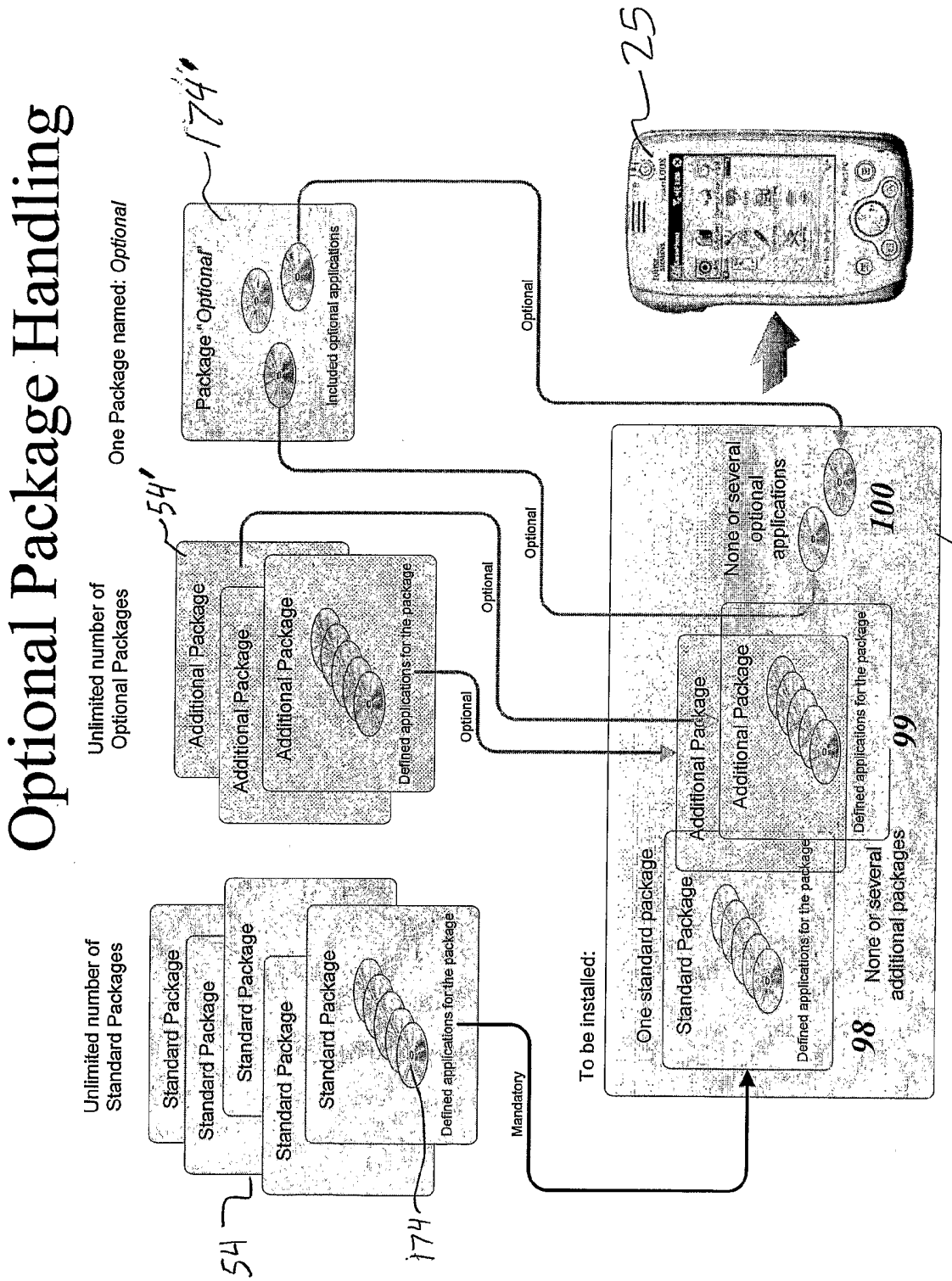

Optional Package Handling
Unlimited number of Standard Packages
Standard Package
Defined applications for the package
Mandatory
54
174
Unlimited number of Optional Packages
One Package named: Optional
Additional Package
Defined applications for the package
Optional
54'
Package "Optional"
Included optional applications
174'
Optional
25
To be installed:
One standard package
Standard Package
Defined applications for the package
Additional Package
Defined applications for the package
98
None or several additional packages
99
None or several optional applications
100
54

Fig. 18

Engine Administrator Prototype and Edit Configuration Screen

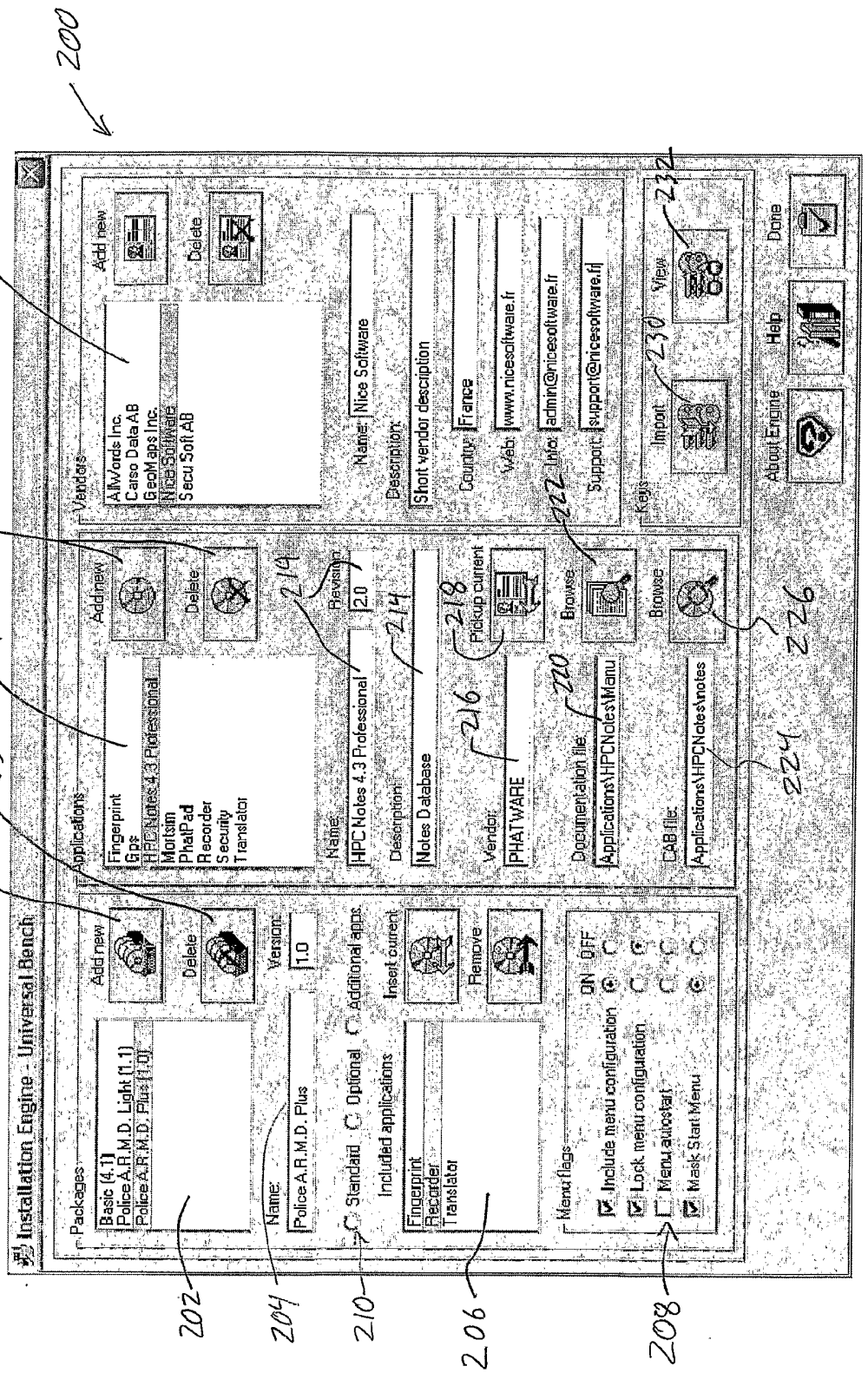


Fig-19