

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

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
1 May 2003 (01.05.2003)

PCT

(10) International Publication Number
WO 03/036489 A1

(51) International Patent Classification⁷: **G06F 15/00**

San Jose, CA 95133 (US). **FLETCHER, Adrian**; 4001
Discovery Drive #340, Boulder, CO 80303 (US).

(21) International Application Number: PCT/US02/34006

(22) International Filing Date: 24 October 2002 (24.10.2002)

(74) Agents: **MEYER, Sheldon**, et al.; Fliesler, Dubb, Meyer
and Lovejoy LLP, Suite 400, Four Embarcadero Center,
San Francisco, CA 94111-4156 (US).

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/386,487 24 October 2001 (24.10.2001) US

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

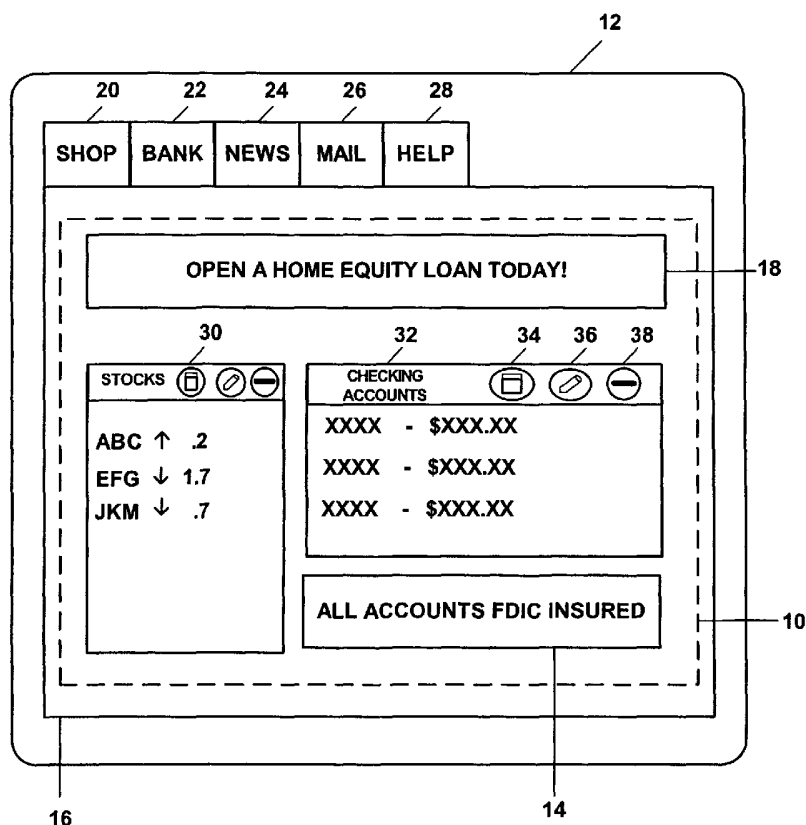
(71) Applicant: **BEA SYSTEMS, INC.** [US/US]; 2315 North
First Street, San Jose, CA 95131 (US).

(84) Designated States (*regional*): ARIPO patent (GH, GM,
KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW),
Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),

(72) Inventors: **QIAN, Shelly**; 10340 Leola Court #2, Cuper-
tino, CA 95014 (US). **LOR, Robert**; 1936 Cape Ann Place,

[Continued on next page]

(54) Title: SYSTEM AND METHOD FOR PORTAL RENDERING



(57) Abstract: A system and method for rendering a portal, comprising rendering a portal containing a portal page (10) with a first Java Server Page (JSP); rendering the portal page (10) containing a portlet (18) with a second JSP; rendering the portlet (30) containing a portlet element with a third JSP; and rendering the portlet (32) containing the portlet element with a fourth JSP.

WO 03/036489 A1



European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

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.

Published:

— *with international search report*

SYSTEM AND METHOD FOR PORTAL RENDERING

5

Inventors:

Shelly Qian
Robert Lor
Adrian Fletcher

10

CLAIM OF PRIORITY

This application claims priority from ENHANCED PORTALS [FLAGSTAFF RELEASE], U.S. Provisional Application No. 60/386,487, Inventors: Phil Griffin, et al., filed on October 24, 2001, and which is
15 incorporated herein by reference.

CROSS REFERENCES

This application is related to the following co-pending applications which are each hereby incorporated by reference in their entirety: SYSTEM
20 AND METHOD FOR PORTAL PAGE LAYOUT, Inventors: Phil Griffin, et al., filed on _____; SYSTEM AND METHOD FOR DELEGATED SYSTEM ADMINISTRATION, U.S. Application Serial No. _____, Inventors: Phil Griffin, et al., filed on _____; SYSTEM AND METHOD FOR RULE-BASED ENTITLEMENTS, U.S. Application Serial No.
25 _____, Inventors: Phil Griffin, et al., filed on _____.

FIELD OF THE DISCLOSURE

The present invention disclosure relates to portal rendering.

30

BACKGROUND

A portal is a point of access to data and applications that provides a unified and personalized view of information and resources. Typically, a portal

is implemented as one or more pages on a website. Portal pages can integrate many elements, such as live data feeds, static information and multimedia presentations. However, if the software required for rendering each element on a portal page is complex, maintenance problems can arise. For example, a web
5 page that contains text and graphics is coded in at least two programming languages, often intermingled within the same file used to describe the web page. Implementing even minor changes requires a great deal of skill and knowledge. What is needed is a more modular approach to portal rendering that permits users of average skill to create and modify portal pages with minimal
10 effort.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is an illustration of an exemplary portal in accordance to one embodiment of the invention.

15 **Figure 2** is diagram of a system in accordance to one embodiment of the invention.

Figure 3 is diagram of portal rendering in accordance to one embodiment of the invention.

Figure 4 is diagram of portlet rendering in accordance to one
20 embodiment of the invention.

DETAILED DESCRIPTION

The invention is illustrated by way of example and not by way of limitation in the figures of the accompanying drawings in which like references
25 indicate similar elements. It should be noted that references to “an” or “one” embodiment in this disclosure are not necessarily to the same embodiment, and such references mean at least one.

Figure 1 is an illustration of an exemplary portal in accordance to one embodiment of the invention. In one embodiment, by way of example, portal
30 **16** is rendered upon display area **12**, which can be a region in the display area of

a device (e.g., a computer monitor). In a further embodiment, a portal can be rendered as one or more pages available on the World Wide Web. Such pages can be implemented using Hypertext Markup Language (HTML), or other programming language, so as to be render-able by a commercial web browser. One such browser is Microsoft Internet Explorer, available from Microsoft, Corp. of Redmond, Washington.

Portal 16 contains several elements. Portal page selection tabs 20-28 can each be selected by a user to render a different page within the portal. Portal pages can be thought of as panels or panes that are swapped into and out of a display region of the available portal real estate. If tab 20 were selected, for example, the portal page corresponding to that tab would be rendered. By way of a non limiting example, selection of a portal element can be accomplished with an input device such as a mouse, a motion detector, a voice command, a hand or eye gesture, etc. Although tabs 20-28 in Fig. 1 are displayed horizontally, in another embodiment the tabs could be displayed vertically or using some other scheme. In yet another embodiment, tabs 20-28 could be rendered as buttons or as other kinds of interactive controls.

Portal 16 is also contains portal page 10, which includes portlets and other elements. A portlet is an application that can manage its own graphical user interface (GUI). Portlets can communicate with each other and with other software and hardware components (e.g., Enterprise Java Beans™, Java™ Beans, servlets, applets, etc.). The Java™ programming language, its libraries, environment and toolkits are available from Sun Microsystems, Inc. of Santa Clara, California. The other software and hardware components may be part of the same execution environment as the portlet or may be in a different execution environment. In one embodiment, a portlet is implemented as a JavaServer Page™ (JSP). JSP's are part of the Java™ standard available from Sun Microsystems, Inc. For example, portlet 30 displays real-time stock ticker information. A user might be able to configure such a portlet to display certain stocks of interest. In another embodiment, the user can select a given stock

displayed in portlet 30 and receive more detailed information, such as the price history, price to earnings ratio, etc. The portlet would be responsible for handling user input and responding accordingly. Portlet GUI 32, for example, displays up-to-date information pertaining to a user's checking accounts. Likewise, portlet 32 could provide detailed information on transactions if the user were to select an account. Advertisement portlet 18 displays an advertisement that could be directed specifically to the current user based on demographics or other information. For instance, if a user had an outstanding home loan in good standing, the advertisement could be for a home equity loan. Likewise, if the user had an appropriate amount in a savings account, the advertisement could be for a new car loan. Static area 14 contains non-interactive content, such as text, an image or any combination thereof.

In one embodiment, a portal page such as 10 can be described with a layout (hereinafter referred to as a "template"). A template can specify the location of elements on a portal page. A template can include one or more placeholders which define regions of the template wherein a portlet can render itself, or wherein other static or dynamic content can be rendered. A given placeholder can host more than one portlet. In such a case, the template area occupied by the placeholder could be shared amongst the portlets in a number of ways, including but not limited to stacking portlets on top of one another (not shown). In another embodiment, a template can be implemented as a Hypertext Markup Language (HTML) table and created with conventional HTML editors. Each cell in a such a table can represent a placeholder. Placeholders can be dynamically bound to portlet GUI's during portal page rendering. In one embodiment, this can be accomplished by inserting directives to initiate portlet execution, such as JSP tag library calls, in the HTML table cells representing placeholders.

Figure 2 is diagram of a system in accordance to one embodiment of the invention. Although this diagram depicts objects as functionally separate, such depiction is merely for illustrative purposes. It will be apparent to those skilled

in the art that the objects portrayed in **Fig. 2** can be arbitrarily combined or divided into separate software, firmware or hardware components. Furthermore, it will also be apparent to those skilled in the art that such objects, irregardless of how they are combined or divided, can execute on the same computer or can be arbitrarily distributed among different computers connected by a network.

In one embodiment, by way of illustration, client **200** can render a portal by delegating the work to a hierarchy of JSP's wherein one JSP renders parts of its GUI by invoking other JSP's. Client **200** can be a web browser, for example. However, client **200** may be any software, firmware or hardware capable of communicating with server **202**. Server **202** can be any web server having a Java™ run-time environment and support for JSP, such as the BEA WebLogic Server™, available from BEA Systems, Inc. of San Jose, California.

JSP's separate GUI functionality executing on a client from application logic executing on a server, thereby enabling users to easily change a GUI without having to alter underlying application logic. Generally speaking, a JSP includes a page file that contains traditional HTML and Java™, but has a ".jsp" file name extension rather than ".html". The ".jsp" extension indicates to server **202** that the file contains embedded directives to invoke special JSP processing within server **202**. For example, consider the following JSP page:

```

<%@page import="java.util.*" %>
<HTML>
<HEAD>
25 <TITLE>JSP Date Example</TITLE>
</HEAD>
<CENTER>
<H2>Today's Date is:</H2>
30 <%= new Date() %>
</CENTER>
</BODY>
</HTML>

```

Embedded in the HTML code above are statements expressly for processing by a JSP engine (not shown) in server **202**. For example, the scriptlet code

fragment "new Date()" is delimited by the special symbols <%= and %>. When the JSP page is invoked by client 200, it will be processed and executed on server 202 by the JSP engine, whereas the HTML code will be processed and executed on client 200. The above scriptlet creates a new instance of the Date class on server 202 and returns the current date as a string to client 200, which is then displayed by client 200:

Today's Date is:
Tues July 05 13:34:33 PDT 2002

In one embodiment, when client 200 invokes a JSP page, web server 202 compiles the page into a servlet and spawns it as a running process. A servlet is a process deployed in a web server. Any dynamic output of the servlet is sent back to client 200 as a response. Processing of JSP code within the web server may also result in the inclusion (and invocation) of other JSP files via the <jsp:include> or similar directive.

Figure 3 is diagram of portal rendering in accordance to one embodiment of the invention. Each box in the diagram having a file name therein represents a JSP page. The arrows beneath the boxes represent execution flow from one JSP page to another via JSP include directives. index.jsp 300 is the entry point for the portal web page. It forwards the web page request to the main portal JSP page, Portal.jsp 302. Portal.jsp 302 is responsible for rendering the portal and its contents. As such, it must acquire various information needed for this endeavor. In one embodiment, Portal.jsp 302 retrieves such information via calls to a JSP tag library, esp.tld 204. esp.tld 204 includes JSP tags for communicating with portal manager 206. Portal manager 206 is responsible for retrieving and persisting information pertaining to portal "skins", portal templates and portlets. In one embodiment, the portal manager can be implemented as a stateless session Enterprise Java Bean™ (EJB). Persistence manager 208 is responsible for persisting this information and providing it to portal manager 206. Persistence manager 208 may utilize a

relational database management system (RDMBS) **210**.

In one embodiment, a skin can be a collection of files that includes a cascading style sheet and a directory of images that define the look and feel of a portal. Every button, banner, portlet header, background color, and font characteristic can be determined by the skin. In one embodiment, a portal can be associated with more than one user or portal group. A user group can be any arbitrary collection of users that is determined statically, or dynamically by evaluating rules that take into account information about a user and other information. A portal has a default skin that can be customized for a group. In this way, a group can define its own look and feel for a particular portal page. The group skin can be further customized to suit the needs of individual users. Skin information can be cached in server **202** so that it may be used by other JSP's invoked by Portal.jsp during rendering of the portal page.

Another piece of information retrieved by Portal.jsp **302** via esp.tld **204** is a template corresponding to each portal page. As with skins, a template can be customized. In one embodiment, customization of a template involves the layout or placement of portal elements (e.g., portlets, static data, dynamic data, header, footer, content, etc.) on the template. A template can be afforded an arbitrary number of levels of customization. In another embodiment, a template can have three levels of customization: global, group and user. An initial template is created for a portal page at the global level. This is the default template used for describing the portal page if there are no further customizations. Customizations at the group level supersede the global level template. Likewise, customizations at the user level take precedence over the group and global levels.

As part of retrieving the template, portal manager **206** binds portlets to template placeholders. In doing so, portal manager **206** utilizes access controller **212** to determine capabilities or permissions (e.g., whether the given portal user is entitled to execute, view or edit the portlet in question) based on one or more entitlements. An entitlement is a rule that grants or denies access

to a resource capability. In one embodiment, a resource can be any resource available on a computer network, including but not limited to a portlet, a portlet GUI component, a portal page, portlet content, etc. In one embodiment, evaluation of an entitlement consists of dynamically associating roles with a user based on role rules that take into account information about the user, information about the user's communication session, or the current state of the system. For example:

- (1) When all of these conditions apply, the user is a GoldMember:
Checking account balance > \$5,000
Combined account balance > \$50,000

In rule (1) above, a user is deemed a GoldMember if there is more than \$5,000 in their checking account and they have more than \$50,000 in their combined accounts. Certain portlets, for example, may only be for GoldMember users. Thus, if only GoldMember users were entitled to execute the portlet in question, it would not be bound to a template for non-GoldMember users. Similarly, if the entitlement was for viewing, rather than execution, the portlet would be bound to the template but only GoldMember users could view it. If the entitlement was for editing, then the portlet would be bound to the template and would be visible to a user, however only GoldMember users would be allowed to edit the portlet's contents. Access controller **212** directs role mapper **214** to determine which roles correspond to client **200**. Access controller **212** then directs decision module **216** to determine whether or not access to a resource should be allowed based on entitlements corresponding to the roles.

The final piece of information Portal.jsp **302** needs in order to render the portal is the currently selected portal page tab **20-28**. This determines which portal page template to use initially, since each portal page is described by a template and each tab corresponds to a unique portal page. After obtaining this preliminary information, as shown in Fig. 3, Portal.jsp **302** uses a JSP include

directive to invoke header.jsp 304. header.jsp 304 renders a standard header which is the same for each page in the portal. The header could be a navigation pane, for example, allowing a user to jump to different areas of the portal. It could also provide a standard graphic design element to unify pages within the portal. Next, Portal.jsp 302 invokes PortalPageHeader.jsp 306. PortalPageHeader.jsp 306 renders tabs 20-28 based on the number of portal page templates retrieved. Next, Portal.jsp 302 invokes PortalPageContent.jsp 308 to render the selected portal page. A placeholder defines regions of the template wherein a portlet can display itself, or wherein other static or dynamic content can be rendered. In one embodiment, PortalPageContent.jsp 308 is responsible for rendering the page elements corresponding to each placeholder. If the placeholder hosts a portlet, PortalPageContent.jsp 308 invokes Portlet.jsp 310.

Figure 4 is diagram of portlet rendering in accordance to one embodiment of the invention. Portlet.jsp 310 invokes Titlebar.jsp 402 which, in turn, renders the portlet's window title bar. The title bar might display icons that, when selected, allow a user to edit the portlet's contents 36, detach the portlet such that it occupies a window independent from the web browser's main window 34, or remove the portlet 38 (i.e., close its window). Titlebar.jsp 402 can access entitlement information via esp.tld 204 in deciding whether or not these icons are present or active. Portlet.jsp 310 invokes Banner.jsp 404 to render a banner in a like manner to that of the portal page. Portlet.jsp 310 invokes aheader.jsp 406 and afooter.jsp 410 render the header and footer portions of the portlet window. Portlet content generation (e.g., stock quotes, account balances) begins when Portlet.jsp 310 invokes acontent.jsp 408 to initiate portlet execution (e.g., by making a JSP tag library call).

One embodiment may be implemented using a conventional general purpose or a specialized digital computer or microprocessor(s) programmed according to the teachings of the present disclosure, as will be apparent to those skilled in the computer art. Appropriate software coding can readily be

prepared by skilled programmers based on the teachings of the present disclosure, as will be apparent to those skilled in the software art. The invention may also be implemented by the preparation of integrated circuits or by interconnecting an appropriate network of conventional component circuits, as
5 will be readily apparent to those skilled in the art.

One embodiment includes a computer program product which is a storage medium (media) having instructions stored thereon/in which can be used to program a computer to perform any of the features presented herein. The storage medium can include, but is not limited to, any type of disk
10 including floppy disks, optical discs, DVD, CD-ROMs, microdrive, and magneto-optical disks, ROMs, RAMs, EPROMs, EEPROMs, DRAMs, VRAMs, flash memory devices, magnetic or optical cards, nanosystems (including molecular memory ICs), or any type of media or device suitable for storing instructions and/or data.

15 Stored on any one of the computer readable medium (media), the present invention includes software for controlling both the hardware of the general purpose/specialized computer or microprocessor, and for enabling the computer or microprocessor to interact with a human user or other mechanism utilizing the results of the present invention. Such software may include, but is not
20 limited to, device drivers, operating systems, execution environments/containers, and user applications.

The foregoing description of the preferred embodiments of the present invention has been provided for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise forms
25 disclosed. Many modifications and variations will be apparent to the practitioner skilled in the art. Embodiments were chosen and described in order to best describe the principles of the invention and its practical application, thereby enabling others skilled in the art to understand the invention, the various embodiments and with various modifications that are suited to the particular use
30 contemplated. It is intended that the scope of the invention be defined by the

following claims and their equivalents.

CLAIMS

What is claimed is:

- 5
1. A method for rendering a portal, comprising:
rendering a portal containing a portal page with a first Java Server Page (JSP);
rendering the portal page containing a portlet with a second JSP;
10 rendering the portlet containing a portlet element with a third JSP; and
rendering the portlet element with a fourth JSP.
 2. The method of claim 1 wherein:
wherein rendering utilizes a skin.
 - 15 3. The method of claim 2 wherein:
the skin can be customized.
 4. The method of claim 1 wherein:
20 the portal page is a Hypertext Markup Language (HTML) table.
 5. The method of claim 1 to be used by a portal group which contains at least one user wherein:
the portal page is described by a template wherein the template can be
25 customized by the group and/or by the at least one user.
 6. The method of claim 5 wherein:
a placeholder is associated with the template.
 - 30 7. The method of claim 6 wherein:
the portlet is dynamically bound to the placeholder.

8. The method of claim 1 wherein:
the portlet has at least one capability which is determined by evaluation
of at least one entitlement.
- 5
9. The method of claim 1 wherein:
at least one entitlement acts as security for the portlet.
10. The method of claim 1 wherein:
access to the portlet is guarded by at least one entitlement.
- 10
11. The method of claim 1 wherein:
rendering the portal includes rendering a header and a footer.
12. The method of claim 1 wherein:
rendering the portlet element includes rendering at least one of a title
bar, a banner, a header, a content and a footer.
- 15
13. A method for rendering a portal, comprising:
rendering a portal containing a customizable portal page with a first Java
Server Page (JSP);
rendering the portal page containing a portlet with a second JSP;
rendering the portlet containing a portlet element with a third JSP;
rendering the portlet element with a fourth JSP; and
wherein the portlet corresponds to a placeholder on the portal page.
- 20
- 25
14. The method of claim 13 wherein:
wherein rendering utilizes a skin.
15. The method of claim 14 wherein:
- 30

the skin can be customized.

16. The method of claim 13 wherein:
the portal page is a Hypertext Markup Language (HTML) table.

5

17. The method of claim 13 to be used by a portal group which contains at least one user wherein:

the portal page is described by a template wherein the template can be customized by the group and/or by the at least one user.

10

18. The method of claim 17 wherein:
the placeholder is associated with the template.

19. The method of claim 13 wherein:
15 the portlet is dynamically bound to the placeholder.

20. The method of claim 13 wherein:
the portlet has at least one capability which is determined by evaluation of at least one entitlement.

20

21. The method of claim 13 wherein:
at least one entitlement acts as security for the portlet.

22. The method of claim 13 wherein:
25 access to the portlet is guarded by at least one entitlement.

23. The method of claim 13 wherein:
rendering the portal includes rendering a header and a footer.

- 30 24. The method of claim 13 wherein:

rendering the portlet element includes rendering at least one of a title bar, a banner, a header, a content and a footer.

- 5 25. A method for rendering a portal, comprising:
 spawning a first process to render a portal containing a portal page;
 spawning a second process to render the portal page containing a portlet;
 spawning a third process to render the portlet containing a portlet
 element; and
 spawning a fourth process to render the portlet element.
- 10 26. The method of claim 25 wherein:
 wherein rendering utilizes a skin.
- 15 27. The method of claim 26 wherein:
 the skin can be customized.
28. The method of claim 25 wherein:
 the portal page is a Hypertext Markup Language (HTML) table.
- 20 29. The method of claim 25 to be used by a portal group which contains at
 least one user wherein:
 the portal page is described by a template wherein the template can be
 customized by the group and/or by the at least one user.
- 25 30. The method of claim 29 wherein:
 a placeholder is associated with the template.
31. The method of claim 30 wherein:
 the portlet is dynamically bound to the placeholder.
- 30

32. The method of claim 25 wherein:
the portlet has at least one capability which is determined by evaluation
of at least one entitlement.
- 5 33. The method of claim 25 wherein:
at least one entitlement acts as security for the portlet.
34. The method of claim 25 wherein:
access to the portlet is guarded by at least one entitlement.
- 10 35. The method of claim 25 wherein:
rendering the portal includes rendering a header and a footer.
36. The method of claim 25 wherein:
15 rendering the portlet element includes rendering at least one of a title
bar, a banner, a header, a content, and a footer.
37. The method of claim 25 wherein:
a process is a servlet.
- 20 38. A machine readable medium having instructions stored thereon that
when executed by a processor cause a system to:
render a portal containing a portal page with a first Java Server Page
(JSP);
25 render the portal page containing a portlet with a second JSP;
render the portlet containing a portlet element with a third JSP; and
render the portlet element with a fourth JSP.
39. The machine readable medium of claim 38 wherein:
30 wherein rendering utilizes a skin.

40. The machine readable medium of claim 39 wherein:
the skin can be customized.
- 5 41. The machine readable medium of claim 38 wherein:
the portal page is a Hypertext Markup Language (HTML) table.
42. The method of claim 38 to be used by a portal group which contains at
least one user wherein:
- 10 the portal page is described by a template wherein the template can be
customized by the group and/or by the at least one user.
43. The machine readable medium of claim 42 wherein:
a placeholder is associated with the template.
- 15 44. The machine readable medium of claim 43 wherein:
the portlet is dynamically bound to the placeholder.
45. The machine readable medium of claim 38 wherein:
- 20 the portlet has at least one capability which is determined by evaluation
of at least one entitlement.
46. The method of claim 38 wherein:
at least one entitlement acts as security for the portlet.
- 25 47. The method of claim 38 wherein:
access to the portlet is guarded by at least one entitlement.
48. The machine readable medium of claim 38 wherein:
- 30 rendering the portal includes rendering a header and a footer.

49. The machine readable medium of claim 38 wherein:
rendering the portlet element includes rendering at least one of a title
bar, a banner, a header, a content and a footer.

5

50. A system for rendering a portal and adapted to communicate with a
client, comprising:

a server to perform a hierarchy of rendering processes in response to
communication from the client; and

10 wherein the server can evaluate at least one entitlement to determine a
permission associated with an element to be rendered.

51. The system of claim 50 wherein:

the server can evaluate at least one entitlement which associates a role
15 with a user based on a role rule.

52. The system of claim 50 wherein:

a rendering process can be a servlet.

20 53. The system of claim 50 wherein:

the server communicates to the client a portal skin and a portal page
template.

54. The system of claim 53 wherein:

25 the portal page template can be customized for a group and/or by at least
one user belonging to the group.

55. The system of claim 53 wherein:

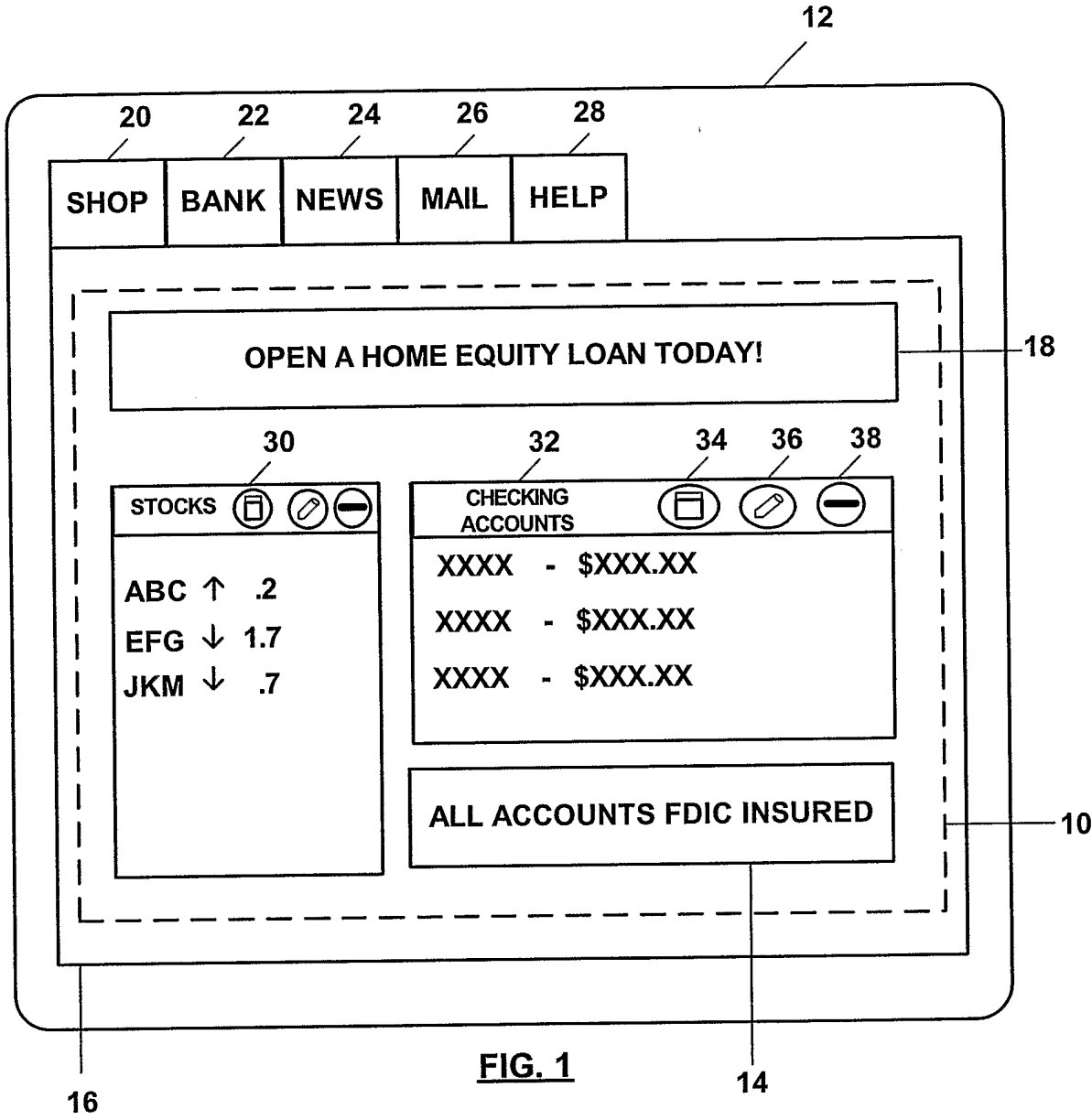
the server selectively binds at least one portlet to the template.

30

56. The system of claim 55 wherein:
binding is based on the evaluation of the at least one entitlement.
57. The system of claim 50, further comprising:
5 a portal manager, adapted to communicate to the client a portal skin and
a portal page template.
58. The system of claim 50, further comprising:
an access controller that can provide access to resources based on
10 evaluation the at least one entitlement.
59. The system of claim 50 wherein:
the hierarchy is determined by the client.
- 15 60. The system of claim 50 wherein:
the element includes at least one of a portlet element and a portlet.
61. The system of claim 50 wherein:
the permission is whether a user is entitled to any one or more of
20 execute, view or edit a portlet.
62. A method for rendering a portlet, comprising:
rendering a portlet wherein at least one entitlement acts as security for
the portlet.
25
63. The method of claim 62 wherein:
rendering utilizes a skin.
64. The method of claim 63 wherein:
30 the skin can be customized.

- 5 65. The method of claim 62 wherein:
the portlet is dynamically bound to a placeholder for a portal page based on
evaluation of the at least one entitlement.
66. The method of claim 62 wherein:
 the portlet has at least one capability which is determined by evaluation
of at least one entitlement.
- 10 67. The method of claim 66 wherein:
the capability is at least one of execute, view and edit.
68. The method of claim 62 wherein:
 rendering includes invoking at least one Java Server Page.
- 15 69. The method of claim 62 wherein:
 rendering includes rendering at least one of a title bar, a banner, a
header, a content and a footer.

20



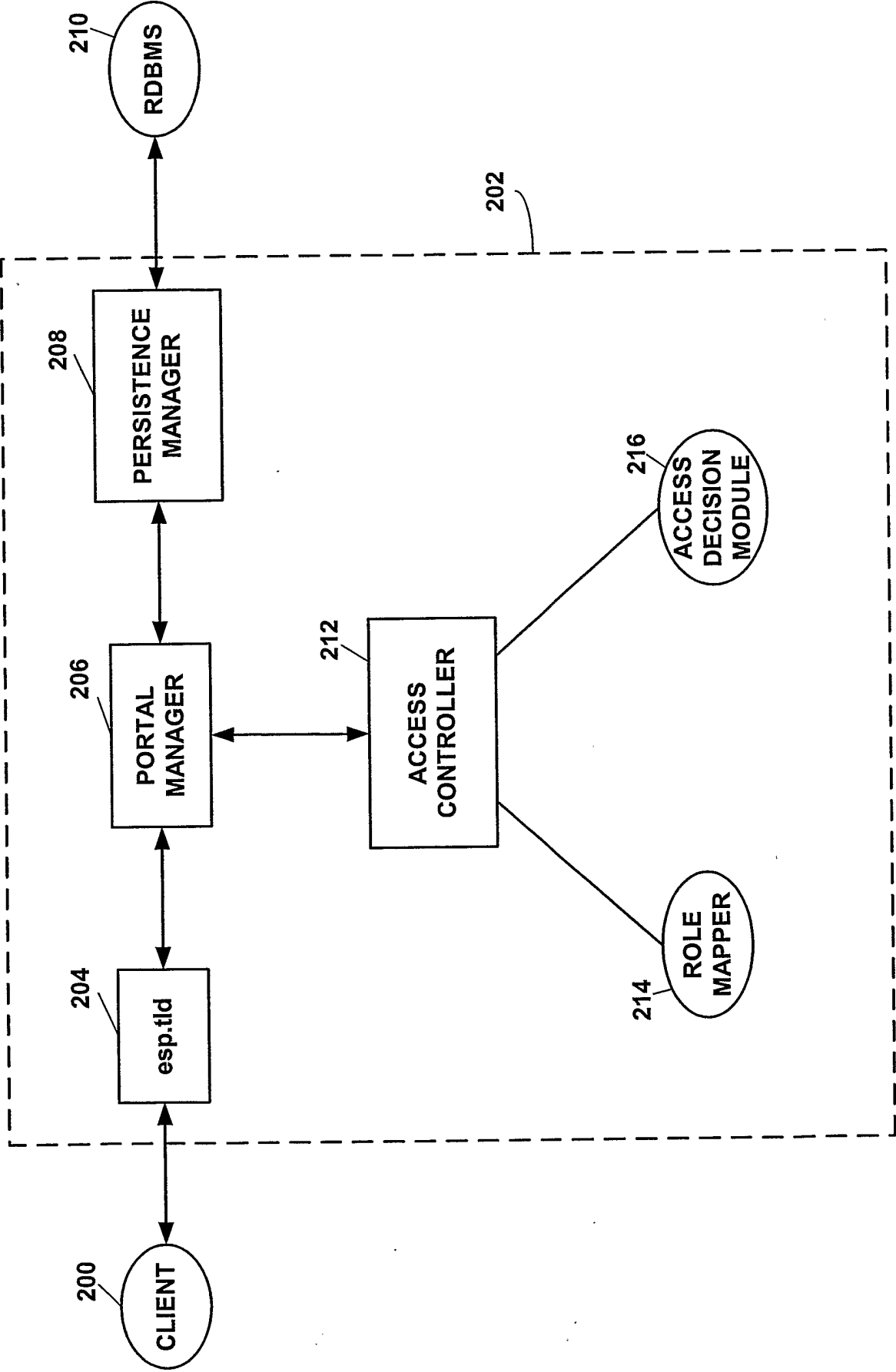


FIG. 2

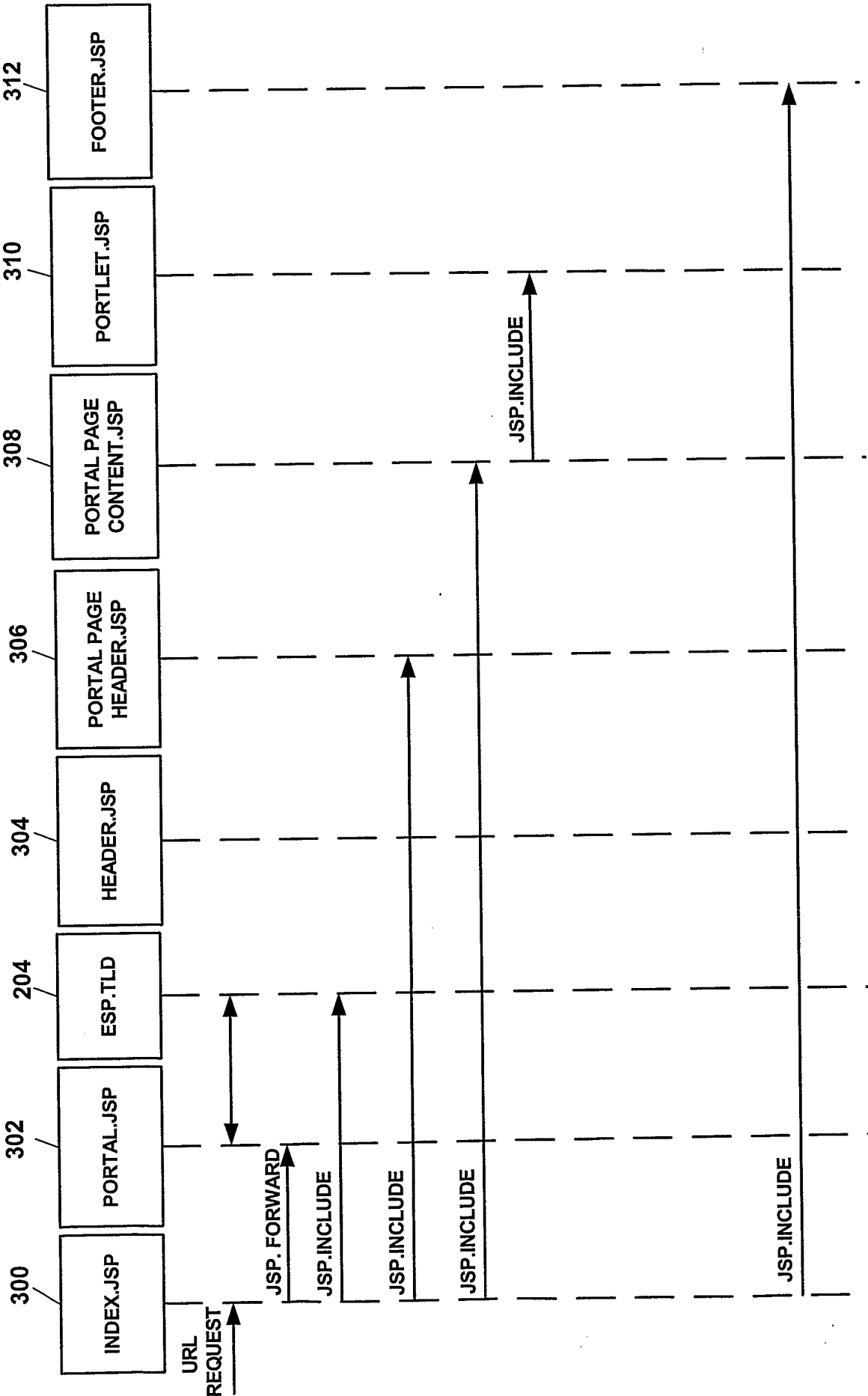


FIG. 3

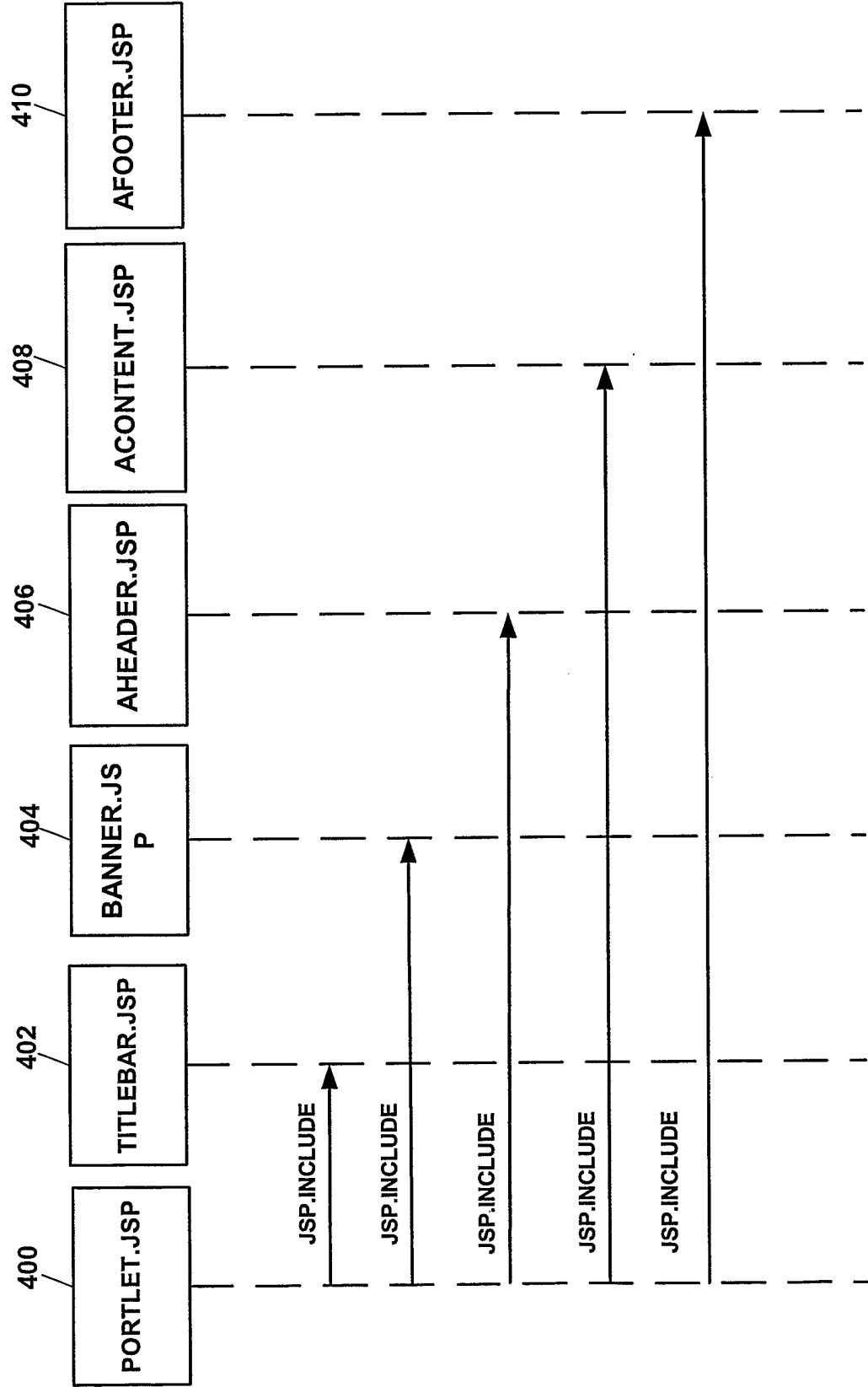


FIG. 4

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US02/34006

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06F 15/00

US CL : 715/501.1

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 715/501.1, 513; 345/700

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

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

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 6,133,915 A (ACURI) 17 October 2000 (17.10.2000), figures 2a-6, col 2, line 40.to col 3, line 40, cols 7-14.	1-65
A	US 6,188,399 B1 (VOAS et al.) 13 February 2001 (13.02.2001), figure 2E, col 3, lines 30-53.	1-65
X, P	US 6,327,628 B1 (ANUFF et al.) 04 December 2001 (04.12.2001), all.	1-65
X, P	WO 02/063496 A2 (MUNIZ) 15 August 2002 (15.08.2002), all.	1-65
A	MORAN et al., Multimodal User Interfaces in the Open Agent Architecture, ACM 1997, all.	1-65
A	ROUFF, Formal Specification of User Interfaces, SIGCHI Bulletin, July 1996, vol. 17, no. 3, all	1-65



Further documents are listed in the continuation of Box C.



See patent family annex.

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

Date of the actual completion of the international search

27 December 2002 (27.12.2002)

Date of mailing of the international search report

13 JAN 2003

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231
Facsimile No. (703)305-3230

Authorized officer

Heather Herndon

Telephone No. 703-308-5186

James R. Matthews