

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
13 February 2003 (13.02.2003)

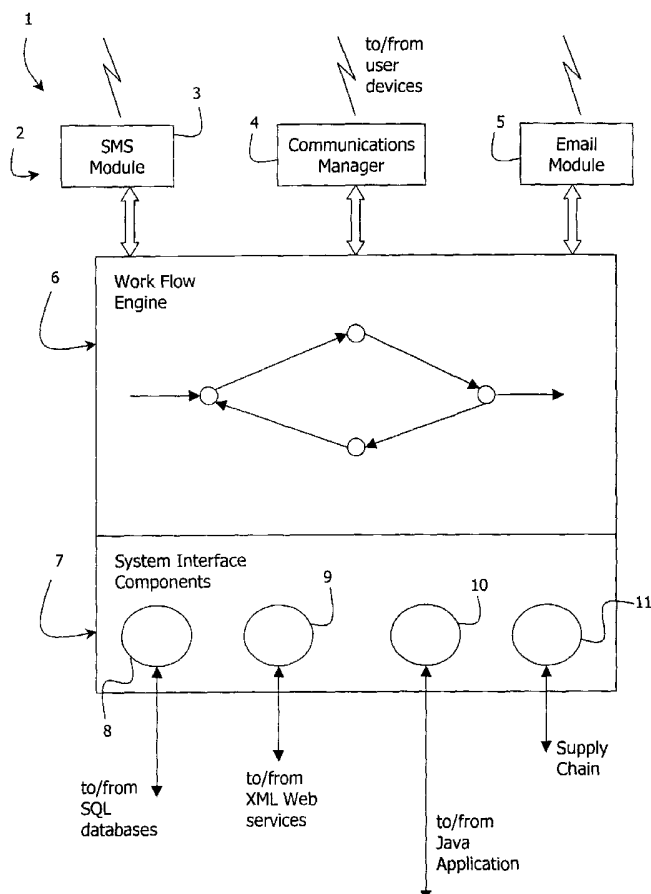
PCT

(10) International Publication Number
WO 03/012706 A2

- (51) International Patent Classification⁷: **G06F 17/60**
- (21) International Application Number: PCT/IE02/00116
- (22) International Filing Date: 2 August 2002 (02.08.2002)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
2001/0737 2 August 2001 (02.08.2001) IE
- (71) Applicant (for all designated States except US): **JEEDCOM INC.** [JP/JP]; 3-12-20 Shimoochiai, Shinjuku, Tokyo 161-0033 (JP).
- (72) Inventor; and
- (75) Inventor/Applicant (for US only): **O'CONNELL, Marcus** [IE/IE]; 40 Whitehall Road, Terenture, Dublin 12 (IE).
- (74) Agents: **O'BRIEN, John** et al.; John A. O'Brien & Associates, 3rd floor, Duncairn House, 14 Carysfort Avenue, Blackrock, County Dublin (IE).
- (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, US, UZ, VN, YU, ZA, ZM, ZW.
- (84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SK,

[Continued on next page]

(54) Title: A PROCESS INTEGRATION PLATFORM AND METHOD



(57) Abstract: A platform (1) performs process integration, in which a process receives inputs from users at user devices and involves interactions with remote data processing systems. The interactions are represented by nodes in a work flow engine (6), in which a workflow represents a complete business process. User inputs are kept to a minimum. A client application is downloaded by a manager (4) to the user device and it queries data capture and upload for an associated workflow. The manager (4) and the client framework (19) allow continuity even when a user device is offline.

WO 03/012706 A2



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:

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

“A process integration platform and method”INTRODUCTION5 Field of the Invention

The invention relates to integration of user devices with business systems in terms of processes which are carried out involving their integration.

10 Prior Art Discussion

It is increasingly the case that development of many medium to large scale businesses require operation of sophisticated systems such as supply chain management or customer relationship management (CRM) systems. It is just as important that all
15 relevant people such as employees or suppliers have appropriate and optimum access to those systems for participation in business processes. This is important so that businesses can adapt and respond promptly to changes and new information.

Heretofore, such access has not been achieved in a completely satisfactory manner.
20 Even where users use a GPRS or a 3G device and network for “always on” access, (a) communication and/or (b) system interfacing problems can arise. Regarding (a), wireless bandwidth is expensive, and mobile applications that use the wireless airwaves to connect to a server extensively are generally expensive to use. Secondly, the wireless networks are significantly less reliable than the physically wired networks. This is mainly
25 due to presence of geographical areas where there is minimal or no network coverage and ‘black spots’ – for example in buildings – where there is no coverage. Another aspect is the limited battery life of wireless devices. A typical mobile phone has a standby time in the hundreds of hours, whereas it may have a much shorter use time. Supporting a continually connected service would rapidly drain the battery in a mobile
30 device. Also, typical browser-based applications are rendered inoperable in environments when there is a breakdown in network connections.

Regarding the system-interfacing problems (b) above, even where the connection is good it is often very time-consuming and awkward to gain access to the full range of systems/applications because of their diverse nature. The business systems may be

5 Web-based applications having good access mechanisms, or on the other hand legacy systems having very poor access.

The invention is therefore directed towards providing a platform for process integration which provides both improved reliability and effectiveness of user communication with

10 systems, and also simpler user access to a range of different systems.

SUMMARY OF THE INVENTION

According to the invention, there is provided a process integration platform comprising a

15 user device interface comprising means for communicating with user devices, a system interface comprising means for communicating with data processing systems, and a processing means residing between the device and system interfaces, characterised in that,

20 the processing means comprises a work flow engine comprising means for executing a business process as a workflow having nodes representing interactions with data processing systems, and for executing a workflow with user inputs made transparently of the interactions.

25 In one embodiment, the device interface comprises means for downloading a client application to a user device, said application being for capturing user inputs for an associated workflow.

In another embodiment, the workflow engine comprises means for activating a workflow

30 in response to reception of a dataset from an associated client application.

In a further embodiment, the device interface comprises means for downloading a client application comprising means for continuing user data capture after disconnection of the user device from the platform.

- 5 In one embodiment, the device interface comprises an agent for capturing responses from the work flow engine as a proxy for a user device while the user device is connected.

- 10 In another embodiment, the device interface comprises means for maintaining a register of client applications, and links between each client application and user devices and workflows.

- 15 In a further embodiment, the device interface comprises a plurality of modular transcoders, each being associated with a type of user device and comprising means for transforming content for transmission to a user device.

- 20 In one embodiment, the work flow engine comprises a dispatcher function comprising means for invoking a workflow activity and for updating a work flow management database.

In another embodiment, the work flow engine comprises a repository storing a work flow model representing a business process, and the dispatcher comprises means for invoking an activity from the repository.

- 25 In a further embodiment, the work flow engine comprises a scheduler comprising means for controlling timing of activities.

In one embodiment, timing in terms of sequence of accesses to data processing systems.

In another embodiment, the system interface comprises a plurality of modular components, each being a middleware channel for accessing an associated data processing data.

- 5 In a further embodiment, a component comprises database access function code strings, and means for inserting data received from a user device via the work flow engine in real time.

- 10 In one embodiment, a component comprises means for parsing a WSDL to generate client stubs for a data processing system service, and for choosing a method of a Web service according to a workflow.

- 15 In another embodiment, a component comprises means for receiving an object-oriented class name, method name, and parameter values from the work flow engine 6, and for instantiating an object in a data processing system accordingly.

In a further embodiment, a component comprises means for communicating directly with a remote data processing system by XML file transfer.

- 20 In one embodiment, the invention further comprises a messaging module comprising means for activating a workflow in response to a user message.

- 25 According to another aspect, the invention provides a process integration system comprising a platform as defined above, and user devices comprising means for receiving a client application from the platform and for executing the client application while offline.

- 30 In one embodiment, the user device comprises an offline cache for locally storing user data while offline.

DETAILED DESCRIPTION OF THE INVENTION

Brief Description of the Drawings

5 The invention will be more clearly understood from the following description of some embodiments thereof, given by way of example only with reference to the accompanying drawings in which:-

10 Fig. 1 is a block diagram illustrating the structure of a process integration platform of the invention;

Fig. 2 is a block diagram of a client framework;

15 Fig. 3 is a block diagram of a work flow engine of the platform;

Fig. 4 is a diagram illustrating a user device screen, a work flow engine, and a set of business systems; and

20 Fig. 5 is a flow diagram for a workflow process.

Description of the Embodiments

Referring to Fig. 1 a process integration platform 1 comprises:-

25 a user-interfacing layer 2 comprising an SMS module 3, a communications manager 4, and an Email module 5;

a work flow engine 6; and

30 a system interface layer 7 comprising components 8, 9, 10, and 11.

- 6 -

In overview, the platform 1 allows users to access a range of diverse business systems in a simple, reliable, and effective manner. The communications manager 4 ensures optimum use of communication links from the user devices to the platform 1 in which there is very effective access without a requirement for an “always on” connection. In parallel, the modules 3 and 5 (aswell as possibly other modules) allow “fast-track” alerting and responding for certain events. The mainstream access is via the manager 4 whereas the specific alerting and responding links are via the modules 3 and 5. The work flow engine 6 performs work flow processing between the various business systems so that, importantly, a single view is presented to the user. Thus, the user does not need to navigate across the several systems. In the embodiment illustrated the components 8, 9, 10, and 11 allow access to SQL databases, XML Web services, supply chain systems, and a Java Message Service. This range of systems and applications may be expanded in a modular manner by adding components to the layer 7.

Thus, the platform 1 provides effective and reliable communication for routine access without need for an “always on” connection, an effective alerting/responding mechanism, and presentation of a single view across all business systems to the user. This is achieved without the need to invest in very sophisticated user devices and network subscriptions or to significantly modify legacy business systems.

The interface layer 7 components allow effective integration of the platform 1 (and hence the user community) with a wide range of business systems. The component 8 comprises an SQL string for executing on each of a number of SQL databases. The component 8 receives and inserts information from the work flow engine 6, and this is used to change the SQL strings dynamically during execution of business processes. The value of each variable is determined by a particular result of a preceding activity. The information is received from the user device via the work flow engine 6, acting as an aggregator of information. In the work flow engine 6 a node represents an interaction with one of the business systems. An example is a query to a purchasing SQL database. Thus the nodes are used in various sequences, each sequence being a business process of interactions. The full process is presented to the user in a single composite manner. A

simple example is submission of a loan request. The user sees only a single screen of information containing a "Submit" operation, whereas the work flow engine 6 actually implements several interactions according to flow through the nodes. The components of the layer 7 are simply used as middleware channels for these interactions.

5

The component 9 allows internal or external web services to be integrated as part of a business process controlled by the platform. The component 9 is given the URL of the WSDL (Web Services Description Language) by the work flow engine 6. The component 9 parses the WSDL and generates the necessary client stubs for that service.

10 Thereafter, the work flow engine 6 chooses which method in the web service to be called to implement a business process, and it provides the relevant parameter values.

The component 10 allows access to a Java application. The component 10 allows the user, via the work flow engine 2, to call functions in a versatile manner from existing
15 Java or Enterprise Java business systems. The work flow engine 6 is programmed to specify the class name and method name and any required parameters. The component 10 uses this data to call the class at runtime. Also, Enterprise Java Beans (EJBs) can also be called by specifying the appropriate method in the remote system interface. Thus there is versatility at the level of modularity of the components in the layer 7, and also in
20 the range of functions which can be called by a single component.

Regarding the component, 11 this gains access to a third party's supply chain system using XML.

25 The layer 7 is versatile because any desired number and type of component may be added, depending on the nature of the business systems.

The work flow engine 6 allows users to access the platform 1 and have a controlled single view to one or several of the business systems according to the relevant business
30 process. The connection manager 4 manages the user connection in an optimum manner. Through wireless networks are proliferating quickly and there is movement

towards widespread high-speed wireless connectivity, this is not always the case today. Furthermore, data intensive applications can be prevented from being practical on mobile devices by bottlenecks in the wireless network. The platform 1 and user device clients allow users to view reports, or start complex applications on a mobile device without always requiring a network connection. When a wireless connection is not available, users can continue to use applications, synchronising again either when a wireless connection becomes available, or via a cradle when the user returns to the office.

Referring to Fig. 2, a client framework 19 comprises a mobile portal 20, an application manager 21, and application cache 22, and a connection manager 23. The applications which run on the user devices are small. However, because of the power of the platform 1, they can capture and display data for a range of business systems in a composite view. They are therefore used to ultimately update business systems, while the user only needs to be aware of (and comfortable with) a simple interface. This is achieved because the platform 1 essentially integrates business process interactions, so that only those interactions necessary to complete the business process are performed. Because only the necessary interactions are presented to the user they can typically fit on a single user device screen. It is thus optimisation of interactions required for a business process which achieves the user-level compactness, not data consolidation.

The connection manager 23 monitors network status between the user device and the platform 1. If a network connection exists, requests are routed directly to the platform 1 through the live connection. When a connection to the server is established, synchronisation is automatically triggered, and cached requests are immediately uploaded to the platform 1, and any updates downloaded to the user device. If there is no network request available, the connection manager 23 caches requests locally on the device in an offline cache 24 for upload at a later point in time. This ensures that user requests are always guaranteed, even when there is no network connection available.

A user device can also be synchronised via a cradle connected to any PC. Cradle synchronisation software automatically detects when there are cached requests on the user device in the cradle and forwards them to the platform 1. Any updates from the platform 1, including new mobile applications or new versions of mobile applications are also automatically downloaded to the mobile device. Cradle synchronisation has the advantage that it does not require devices to have some form of wireless connection. Users can take their mobile device with them when they are out of the office, and re-synchronise it when they return.

10 The connection manager 23 includes full 128bit SSL encryption technology, so all communication between the mobile device and the server is completely secure. The connection manager 23 utilises standard SSL port numbers (port 443), so there are no corporate firewall configuration problems.

15 The client framework 19 makes mobile application provisioning and management simple through an intuitive GUI-based environment for creating, managing, and deploying wireless applications. Complex mobile applications can be quickly created using a RAD (Rapid Application Development) tool. These applications provide users a way to interact with complex business processes.

20

Once the client application has been created, it is registered on the platform 1 as being "downloadable" to a client device. Using application descriptor files, a client application cache on the user device verifies in real time that the latest version of the application is already on the mobile device. If the application had previously been downloaded, the application manager 21 can immediately launch the application. If the application has not been previously downloaded, or a new version of the application had been created on the server, the application manager 21 automatically downloads a copy of the application to the mobile device, even over a wireless connection. Because the client applications represent consolidated interactions for a business process, optimised by the work flow engine 6, they are typically quite small (of the order of 10 Kbytes). They can therefore be easily downloaded over the air.

25
30

In the communications manager 4 client applications are registered and stored in a local repository. The manager 4 also comprises client agents which represent clients (users) on the platform 1 when they are disconnected. The client agents receive a request from the relevant user device and subsequently ensures that the request is implemented when the user device is disconnected. Each client agent also collects responses from the work flow engine and other messages bound for the user, and subsequently downloads them to the user device when a connection is established.

Also, the manager 4 also comprises device profiling and content transcoders for handling access by mobile Internet user devices. The transcoders transform and optimise content for the connecting device. The transcoders interrogate a request from a user device to determine the characteristics and capabilities of the device. Using this information, the transcoders transform and optimise the content for display on the user device. This is done in real time as each request is made. As a new type of user device becomes available, it is only necessary to add a new corresponding transcoder to the manager 4. Transcoders for user devices with limited display capabilities are programmed to segment display pages and downloaded section-by-section for browser-based interaction.

When a user selects to run the application manager 21 on the user device they are firstly presented with a logon screen. This performs role-based authorisation following authentication of the user with the platform. This allows the application manager 21 to provide the user with a list of client applications that the user is allowed to access. When the user selects a particular application to run, the application manager 21 performs the following sequence:

- If a connection to the platform 1 is available, it requests the latest version information from the platform for that particular application. The communications manager 4 on the platform 1 returns a Java Application Descriptor (JAD) file. This file describes all of the properties of the application, including its version number.

- The application manager 21 checks its local cache of applications to see if that version of the application has previously been downloaded. If it has then that application instance is launched.
- 5 • If the application manager 21 has not previously downloaded that version of the application it sends a request to the platform to download the application over-the-air (OTA). When the download is complete, the application is added to the local cache and then executed.
- 10 • If no connection to the server exists, the application manager 21 is unable to verify the version number of the application. In this instance, the JAM executes the latest version that exists in the cache, should one exist.

From the perspective of the platform 1, to achieve the above user authentication and role
15 authorisation, the communications manager 4 assigns an access group to each registered user. When a user logs in the manager 4 identifies the relevant access group and retrieves a list of the accessible client applications. The list is downloaded to the application manager 21 in the wireless device, and the user is presented with the list of available client applications. The selected client application is then downloaded and is
20 stored in the device's application cache 22.

The application manager 21 launches the client application with or without continued
existence of the communications link to the platform 1. The client application prompts
input of user data and caches it locally in the cache 24.

25

At the next connection to the platform 1 the application manager 21 uploads the data,
addressed to the business process as represented by the work flow engine 6. Referring to
Fig. 3, the work flow manager 6 is now described in more detail.

- 30 Workflow is the automation of a business process, in whole or part, during which documents, information or activities are passed from one participant to another for

action, according to a set of procedural rules. Participants perform the work represented by a workflow activity instance and can be human or machine resources. Each activity is a piece of work that forms one logical step within a business process and requires a combination of human and machine resources to support process execution.

5 Participants often perform these activities from a list of single units of work.

A workflow is used to represent a business process flow. In general:

- A business process flow consists of discrete *activities* performed by *actors* in some pre-defined *sequence* according to predefined *constraints*.
- 10 • The flow may be *initiated* by multiple different actors and may have multiple different *outcomes*. This implies that there are *decision points* along the way that control the exact sequencing of activities.
- A flow may be broken down into *sub-flows*.
- A group of activities may have *transactional* semantics – that is if one fails they all fail.
- 15 • Actors may be *human* or they may be *software* services.
- Activities may run in *parallel*.
- Actors fulfill *roles*.

The workflow engine 6 comprises a management database 30, a dispatcher 31, a
20 scheduler 32, and a workflow model repository 33.

The workflow engine controls the flow of work through the platform; by interpreting process rules to determine the scheduling of activities, invoking the relevant application and recording the results in the database 30. The dispatcher 31 interacts with the
25 activities which form the business process. It is responsible for invoking the workflow activity, and being informed of when the activity has finished executing, upon which time it updates the workflow management DB 30.

A workflow is a business process, namely a set of interactions with business systems. It is defined in a visual business process definition tool.

Each activity in a workflow operates on an item of data called the *work item*. This item of data may consist of *multiple parts*, each part being handled separately by each activity. For example, a purchase order work item may consist of the item to be purchased, the purchaser, the provider, the billing details and the shipping details. The workflow itself may consist of placing the order, checking it is in stock, verifying the credit details, packing, shipping, delivery and return. The work item cannot be added to or modified arbitrarily. Furthermore, decisions made about the sequencing of the workflow are made solely on the contents of the work item.

Each activity may also contain *activity-specific data*.

A new workflow *instance* may be started in a number of different ways, including:

- submission of a cached mobile Java application.
- receiving an SMS message to the server with an appropriate keyword,
- changes being made to a database table, and
- schedule to start automatically at pre-defined times.

20

Referring to Fig. 4 a user device displays a screen 40 arising from interaction with the platform 1. More specifically, the work flow engine 6 is the core for integrating the user device with a range of business systems. Fig. 4 is a simple example for illustrative purposes, in which the business systems are:

- 25 an enterprise resource planning (ERP) system 45;
 a supplier database 46;
 a financial application 47; and
 a shipping extranet 48.

- 14 -

The work flow engine 6 has a workflow representing a business process, in which nodes of the workflow represent particular interactions with the systems 45-48.

While updating of the systems 45-48 to complete the process may involve a large number of update operations, the work flow engine 6 only requires the user to input the minimum number of inputs required of the user. A single piece of data such as "supplier" inputted by the user is taken by the relevant node and used to perform a number of operations transparently to the user. In the example illustrated, there may be approximately fifty business system updates and fetches to complete a requisition order process, but only four core user inputs are needed. Each work flow node is programmed to:

- request, via the manger 4, and a client application, the minimum number of inputs; and
- perform, via the layer 7, the required business system operations to complete an interaction.

Referring to Fig. 5 in step 61 of a method 60 a client application executing on a user device submits data fields inputted by the user. In step 62, the WFE6 activates a business process. It uses the supplier ID inputted by the user to query the supplier database 46. It then, in step 64, queries the shipping extranet 48 for shipping details, using the supplier details as an index. If the product is not available in step 66 the WFE6 causes the module 3 to transmit an SMS message indicating that the product is not available. If the product is available, in step 67, the WFE 6 creates a purchase order (P.O.) in the ERP system 45, and transmits the P.O. details to the user via the manager 4 in step 68.

This simple example clarifies that a workflow is a set of interactions to complete a process. The workflow is activated upon receipt of data inputted by the user under control of an associated client application. In this example, all user data is received initially to activate the workflow. However, in other examples individual workflow nodes may interactively request further user inputs.

It will be appreciated that the invention allows users using a diverse number of devices to the fully integrated with business processes. This is achieved despite the fact that a range of diverse business systems may be involved in completing the processes. The invention
5 also allows continuity even where communication links to the platform are broken.

The invention is not limited to the embodiments described but may be varied in construction and detail.

Claims

1. A process integration platform comprising a user device interface (4) comprising means for communicating with user devices, a system interface (7) comprising
5 means for communicating with data processing systems, and a processing means residing between the device and system interfaces, characterised in that,

the processing means comprises a work flow engine (6) comprising means for
executing a business process as a workflow having nodes representing
10 interactions with data processing systems, and for executing a workflow with user inputs made transparently of the interactions.
2. A process integration platform as claimed in claim 1, wherein the device interface
15 (4) comprises means for downloading a client application to a user device, said application being for capturing user inputs for an associated workflow.
3. A process integration platform as claimed in claim 2, wherein the workflow
20 engine (6) comprises means for activating a workflow in response to reception of a dataset from an associated client application.
4. A process integration platform as claimed in any preceding claim, wherein the
25 device interface (4) comprises means for downloading a client application comprising means for continuing user data capture after disconnection of the user device from the platform (1).
5. A process integration platform as claimed in claim 4, wherein the device interface
(4) comprises an agent for capturing responses from the work flow engine (6) as a proxy for a user device while the user device is connected.
- 30 6. A process integration platform as claimed in any of claims 2 to 3, wherein the device (4) interface comprises means for maintaining a register of client

applications, and links between each client application and user devices and workflows.

- 5 7. A process integration platform as claimed in any preceding claim, wherein the device interface (4) comprises a plurality of modular transcoders, each being associated with a type of user device and comprising means for transforming content for transmission to a user device.
- 10 8. A process integration platform as claimed in any preceding claim, wherein the work flow engine (6) comprises a dispatcher function (31) comprising means for invoking a workflow activity and for updating a work flow management database (30).
- 15 9. A process integration platform as claimed in claim 8, wherein the work flow engine (6) comprises a repository (33) storing a work flow model representing a business process, and the dispatcher (31) comprises means for invoking an activity from the repository (33).
- 20 10. A process integration platform as claimed in claims 8 or 9, wherein the work flow engine (6) comprises a scheduler (32) comprising means for controlling timing of activities.
- 25 11. A process integration platform as claimed in claim 10, wherein timing in terms of sequence of accesses to data processing systems.
12. A process integration platform as claimed in any preceding claim, wherein the system interface (7) comprises a plurality of modular components (8-11), each being a middleware channel for accessing an associated data processing data.

13. A process integration platform as claimed in claim 12, wherein a component (8) comprises database access function code strings, and means for inserting data received from a user device via the work flow engine (6) in real time.
- 5 14. A process integration platform as claimed in claim 12 or 13, wherein a component (9) comprises means for parsing a WSDL to generate client stubs for a data processing system service, and for choosing a method of a Web service according to a workflow.
- 10 15. A process integration platform as claimed in any of claims 12 to 14, wherein a component (10) comprises means for receiving an object-oriented class name, method name, and parameter values from the work flow engine 6, and for instantiating an object in a data processing system accordingly.
- 15 16. A process integration platform as claimed in any of claims 2 to 15, wherein a component (11) comprises means for communicating directly with a remote data processing system by XML file transfer.
- 20 17. A process integration platform as claimed in any preceding claim, further comprising a messaging module (2, 5) comprising means for activating a workflow in response to a user message.
- 25 18. A process integration system comprising a platform as claimed in any preceding claim, and user devices comprising means for receiving a client application from the platform and for executing the client application while offline.
19. A process integration system as claimed in claim 18, wherein the user device comprises an offline cache for locally storing user data while offline.

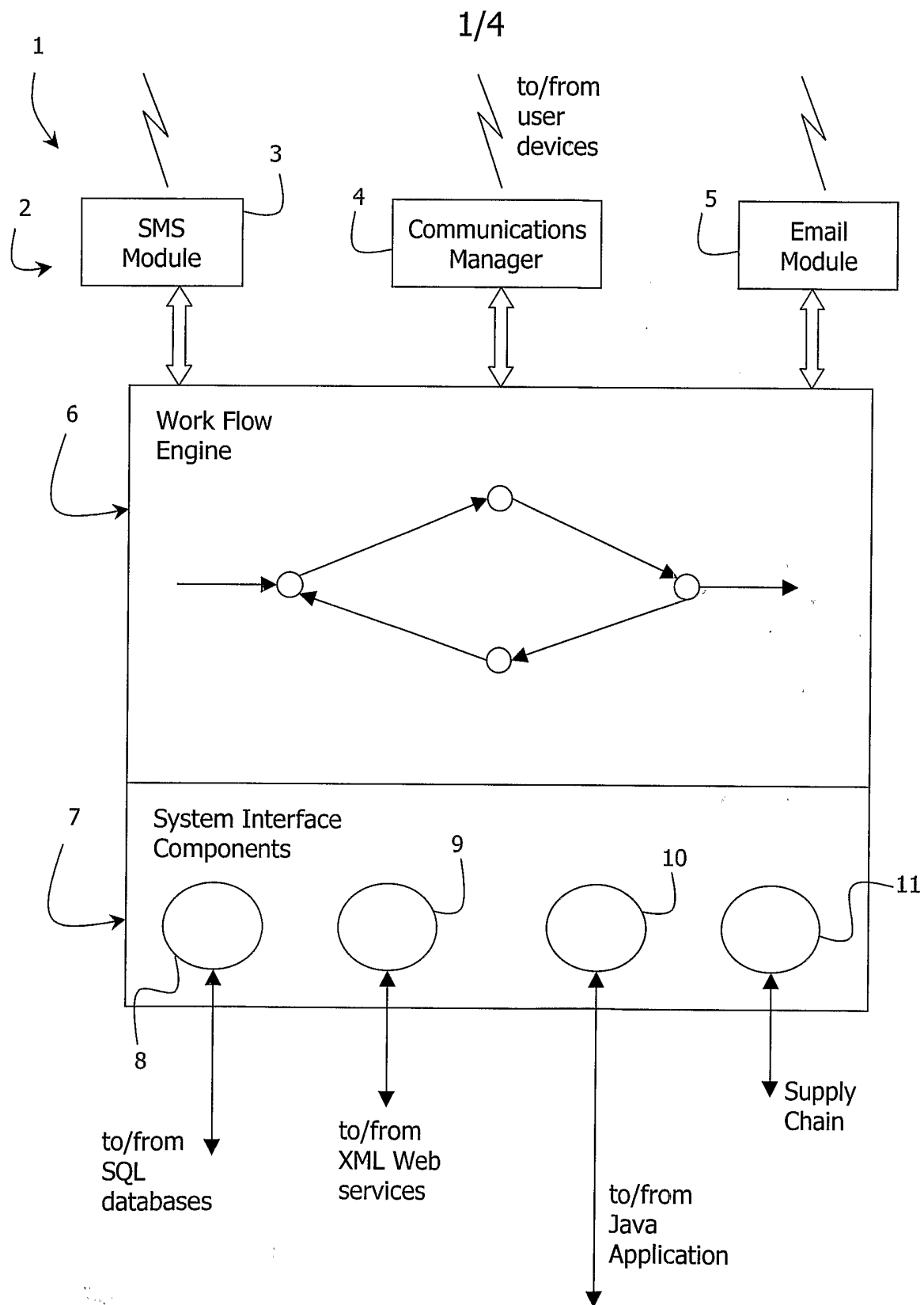


Fig. 1

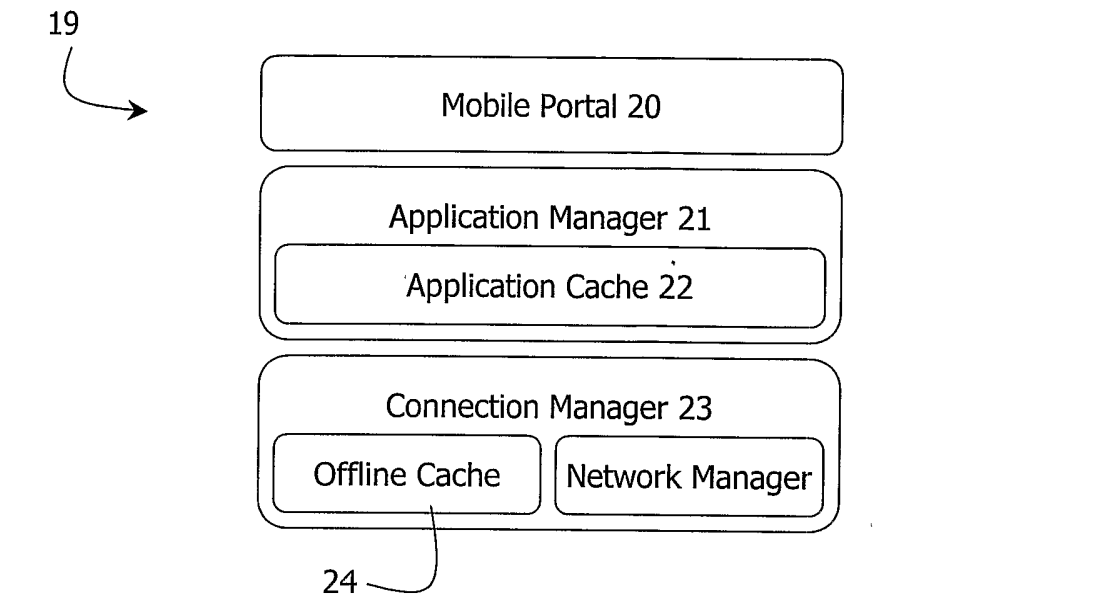


Fig. 2

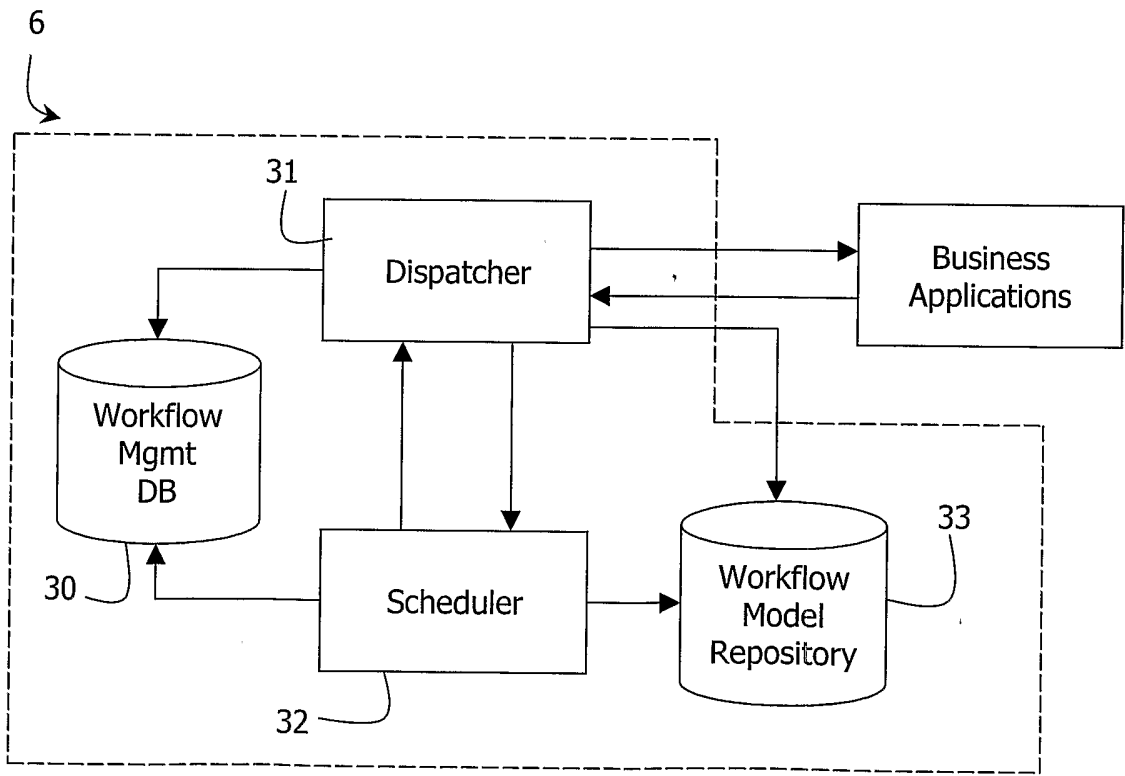


Fig. 3

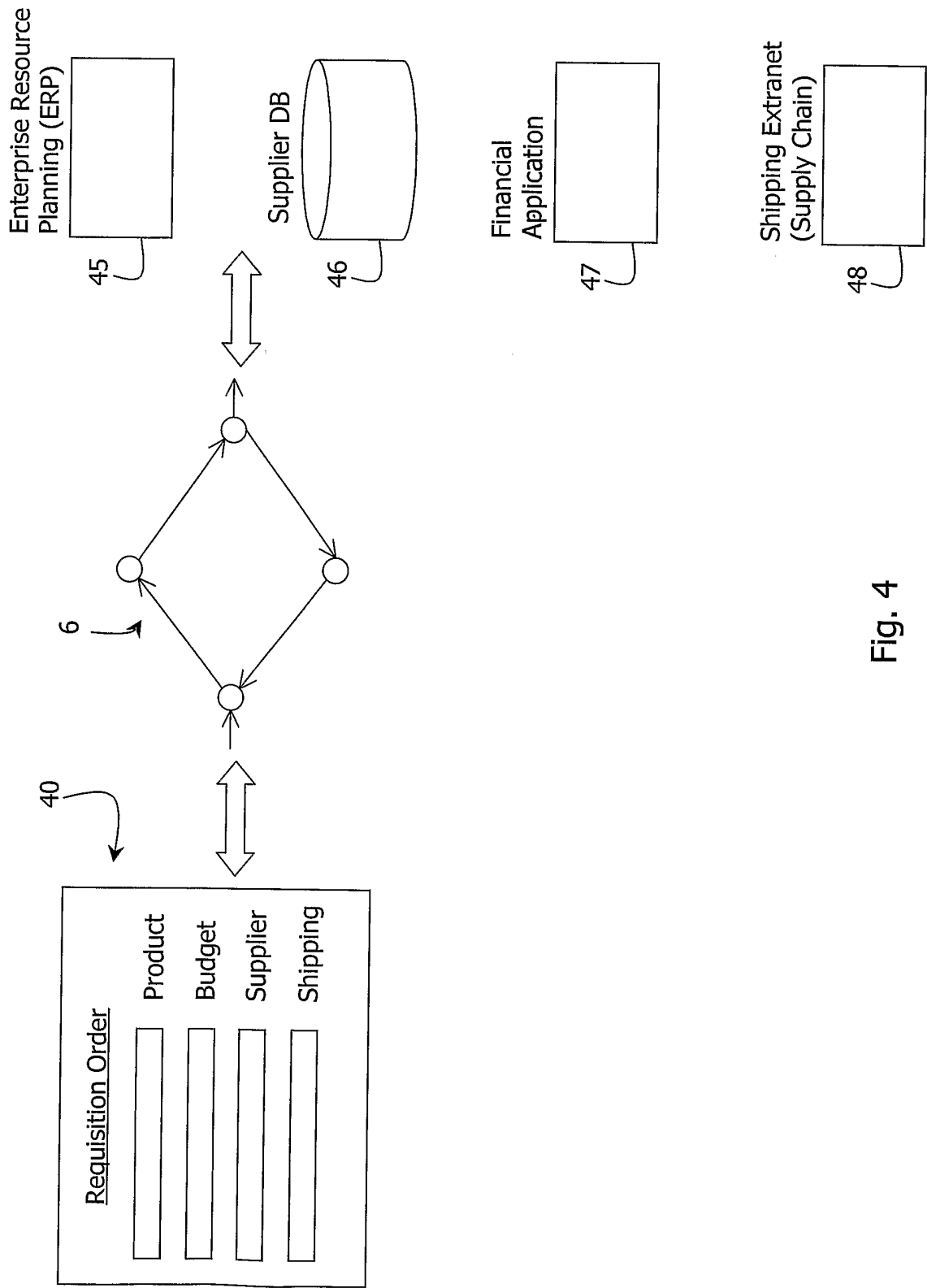


Fig. 4

4/4

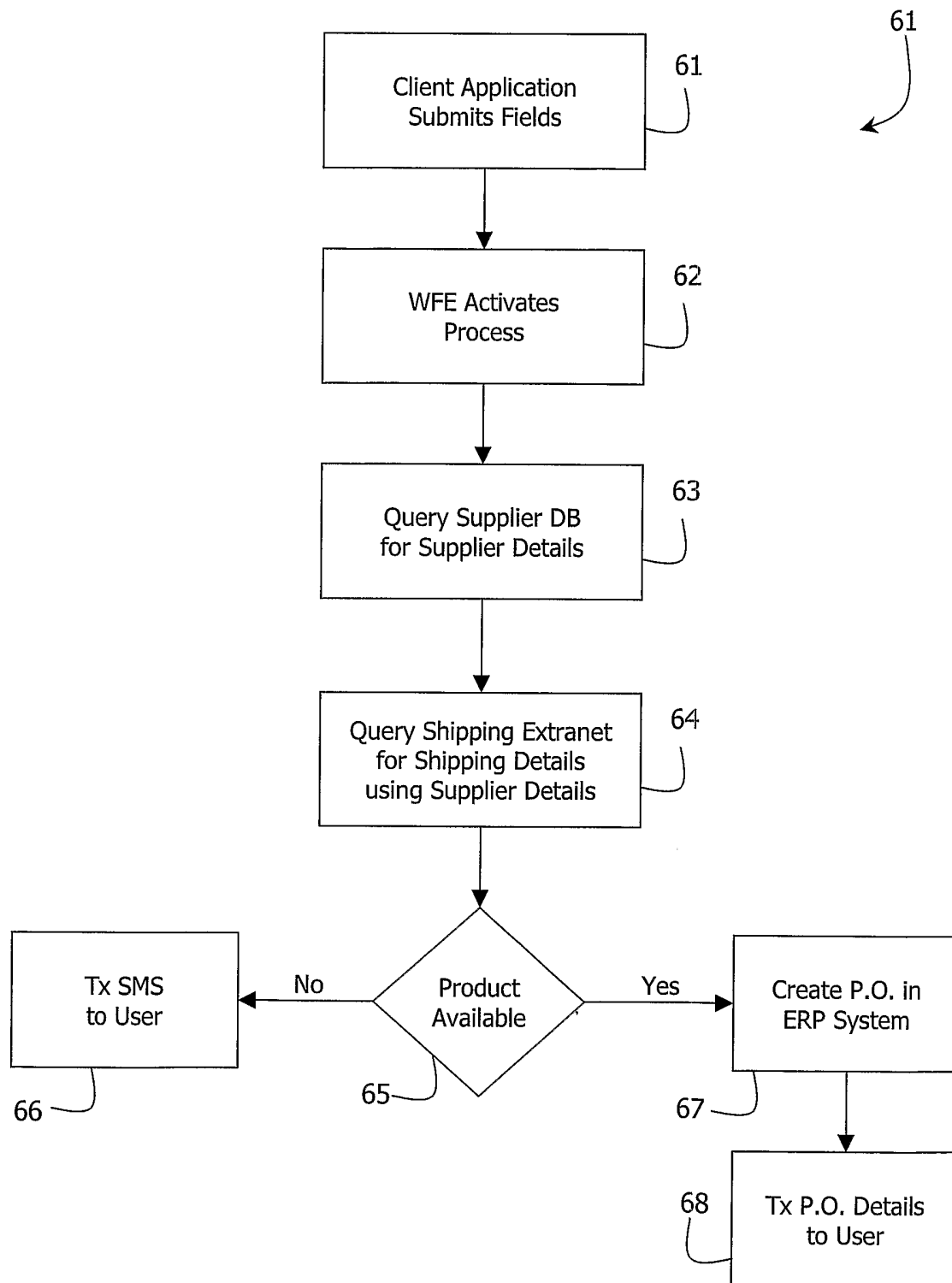


Fig. 5