ROCHFORD, SUZANNE L., CA  
WILLE, ALLAN, CA  
NORTEL NETWORKS LIMITED, CA  
Int.Cl. G06F 17/60  
1999/06/30  
INTERFAC GRAPHIQUE, SYSTEME ET PROCEDE DE GESTION DE PROCESSUS  
PROCESS MANAGEMENT GRAPHICAL USER INTERFACE, SYSTEM AND METHOD
(57) L'invention concerne un procédé, une interface graphique (GUI) et un système permettant la gestion d'un processus défini par une pluralité de tâches qui peuvent être accomplies par différents utilisateurs et qui doivent être exécutées dans un ordre particulier. Le procédé comprend l'affichage des éléments de tâche correspondant aux tâches du processus, l'identification graphique des éléments de tâche correspondant aux tâches déjà exécutées et, si l'utilisateur de ce procédé est capable d'exécuter la tâche suivante non accomplie, l'identification spécifique de l'élément de tâche correspondant. Ces étapes sont répétées après l'exécution de la tâche suivante non exécutée. Ce procédé permet de suivre l'avancement d'un processus et permet d'alerter l'employé de télécommunications responsable de l'exécution de la tâche suivante du processus lorsque la tâche en cours est achevée, ce qui permet d'améliorer l'efficacité du processus, la fiabilité de l'exécution des tâches et le rendement.

(57) A method, GUI and system for managing a process defined by a plurality of tasks capable of being completed by different users and intended to be executed in a particular order. The method includes displaying task elements corresponding to the tasks in the process, graphically identifying the task elements corresponding to tasks which have been completed and, if the user of the method is capable of completing the next uncompleted task, specially identifying the corresponding task element. These steps are repeated upon completion of the next uncompleted task. In this way, the progress of a process is tracked and the telco employee responsible for completing the next task in the process is alerted when the current task is completed, which achieves greater process efficiency, more reliable task execution and superior cost effectiveness.
PROCESS MANAGEMENT GRAPHICAL USER INTERFACE, SYSTEM AND METHOD
PROCESS MANAGEMENT GRAPHICAL USER
INTERFACE, SYSTEM AND METHOD

FIELD OF THE INVENTION

The present invention relates to process management and, more particularly, to methods and systems for managing processes in telecommunications networks.

BACKGROUND OF THE INVENTION

Many business processes, particularly those pertaining to the telecommunications industry, require the execution of a number of tasks in a particular order. Each task in the sequence of tasks corresponding to a process can be viewed as belonging to one of a plurality of functional categories. In the telecommunications network management domain, an industry-wide model has been developed in which tasks corresponding to a network management process are grouped into five main functional categories, namely Fault, Configuration, Accounting, Performance and Security categories. This categorization is commonly referred to as the "FCAPS" functional model, which is formally described in ITU-T Recommendation M.3400, "TMN Management Functions", published in April, 1997.

Due to the diversity of the multiple tasks which together define a telecommunications process, individual tasks are usually assigned to specialized personnel in departments such as engineering, accounting, and so on. In addition to having the responsibility of completing their individual assigned tasks when prompted to do so, telco craftspersons must also have knowledge of the employee to which the subsequent task is assigned, so that this next employee may be given the go-ahead to complete the subsequent task. This eventually leads to completion of the last task in the process, which then
commonly results in the delivery, assurance or billing of a service to a customer.

As the same telco craftsperson is seldom involved in performing more than a few consecutive tasks, it becomes imperative to ensure good communication among telco personnel. This requires that individuals responsible for performing given tasks have adequate knowledge of the overall process flow so that they are at least made aware of the identity of the telco employee assigned to the next task. This, in turn, requires that the telco invest extensively in human resources training, especially in the case of new hires who begin by knowing a very limited number of people within the organization.

Moreover, the consequences of exchanging incorrect information among personnel obviously include a reduction in the probability of successfully completing the process as a whole and delivering, assuring or billing the service.

To achieve greater process efficiency, more reliable task execution and superior cost effectiveness, it would be desirable to provide a process manager which informs users of the progress of a process as individual tasks are completed. Furthermore, it would be desirable if the users were permitted to launch task management utilities from within the process manager so as to enable them to complete certain more complex tasks.

Unfortunately, current process management software is concerned with the management of process flow from a static point of view and is devoid of the capability to automatically monitor a process as it progresses.

**SUMMARY OF THE INVENTION**

According to a first broad aspect, the invention may be summarized as a method of managing a process defined by a
plurality of tasks capable of being completed by different users and intended to be completed in a particular order. The method includes the steps of displaying a plurality of task elements corresponding to the tasks in the process, graphically identifying the task elements corresponding to tasks which have been completed and, if the user of the method is capable of completing the next uncompleted task, specially identifying the corresponding task element. These steps are repeated upon completion of the next uncompleted task.

Preferably, the displayed task elements are arranged in a manner indicative of the order in which the tasks are intended to be completed. A graphical indicator, e.g., an arrow, may be displayed proximate the task element corresponding to the next uncompleted task. Preferably, each task element has a visual attribute indicative of whether or not the corresponding task is capable of being completed by the user.

In response to user selection of a task, the method preferably includes the additional step of displaying information associated with the selected task. The information may includes a graphical indication of whether the selected task is capable of being completed by the user. It may also include a graphical indication of the functional category to which the selected task belongs. The information may also consist of statistical data pertaining to the selected task.

In other cases, the information includes a plurality of fields, wherein the fields contain data if the selected task has already been completed and wherein the fields require completion by the user if the selected task has not been completed. Selected ones of the fields may be completed by launching a task manager. If the user is indeed capable of completing the selected task, then the method preferably includes the additional steps of displaying a graphical entry
mechanism for confirming the data occupying the fields and, in response to the current user accessing the entry mechanism, registering the task as having been completed and displaying updated information related to the selected task.

According to another broad aspect, the invention may be summarized as including the steps of graphically informing the user assigned to complete the next uncompleted task that the next uncompleted task requires execution and, upon completion of the next uncompleted task, graphically informing the user assigned to complete the task following the next uncompleted task that the following task requires execution.

The invention also relates to a computer system and graphical user interface adapted to carry out any of the above methods and to a computer readable medium having software stored thereon for instructing a computer to carry out any of the above methods. Preferably, the methods of the invention are implemented in a network management system.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects and features of the present invention will become apparent to those of ordinary skill in the art upon review of the following description of specific embodiments of the invention in conjunction with the accompanying drawings, in which:

Fig. 1 is a block diagram of a network connected to a network management device featuring a graphical user interface according to an embodiment of the invention;

Fig. 2 shows a workflow icon corresponding to a process;

Fig. 3 shows a display appearing on the screen of the network management terminal of Fig. 1 as a result of launching
a process manager in accordance with the preferred embodiment of the invention;

Fig. 4 shows the display resulting from the selection of a process and a task in the display of Fig. 3;

Fig. 5 shows the display resulting from invoking a task manager from the display of Fig. 4;

Fig. 6 shows the display resulting from having completed the selected task;

Fig. 7 shows the display resulting from having advanced to a next task;

Fig. 8 shows the display resulting from having moved back to a previous task; and

Fig. 9 is a flowchart illustrating operation of the network management terminal.

**DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS**

Referring to Fig. 1, a NMS (network management system) 10 is shown connected to a database 12 and to a data collector 14. The data collector 14 is connected to a network 16 which might for example be an IP (Internet protocol) network. The data collector 14 is capable of collecting information from the network 16.

The NMS 10 has a local database 29 and has a network management processor 19 which is connected to a display 21 and to various user input devices such as a keyboard 22 and a mouse 23. (If the display 21 is touch sensitive, then the display itself provides a user input device.) The local database 29 contains a list of all processes, a list of all tasks in each process, the FCAPS category to which each task belongs, a list of fields associated with each task, the identity of the user(s) or user group(s) responsible for the execution of each task, an indication of which tasks were completed by which users and the
level of authority associated with each potential user of the NMS 10.

The NMS 10 is adapted to perform two main functions. Firstly, it presents a GUI 24 of the display 21 to enable the presentation of information to a user in a coherent manner and to enable the receipt of instructions from the user through the user input devices 22, 23. Secondly, it performs processing on the data stored in the database 12 in accordance with instructions received from users through the GUI 24 to put it into a form suitable for display.

The NMS 10 is used by telco staff who participate in the end-to-end completion of a process affecting the network 14. Since each such process consists of a multiplicity of tasks in various FCAPS categories, multiple users (often in different departments) are involved in completing the process. An individual task may be completed independently by launching a specialized task manager which guides the user through completion of the task. These task managers are part of the software running on the processor in the NMS 10.

Still, some users (e.g., novice users with little knowledge of the overall flow of a process) will require guidance as to the order in which certain tasks must be executed. Such guidance is provided by a process manager in accordance with the invention. The process manager is preferably a piece of software running on the processor in the NMS 10. The process manager can be launched by the user selecting a “process manager” palette or icon from an ordinary desktop, a specialized network management desktop or a toolbar located on the screen of the display 21. The process manager may be invoked separately by different users of the NMS 10 and multiple instances of the process manager may run contemporaneously.
Operation of the process manager is now described with reference to the flowchart in Fig. 9. (Details of the preferred graphical user interface will be described later on.) At step 902, the process manager which is launched by a "current" user of the NMS 10 begins by obtaining the identity of the current user from the NMS 10. (Alternatively, if several users in a common user group are capable of performing certain tasks, then the process manager may require only the identity of the current user group.) After obtaining the identity of the user, the process manager can, by consulting the internal database 29, obtain a list of tasks which are the responsibility of the current user (or user group). In addition, also based on the identity of the current user, the process manager can determine the authority of the current user.

At step 904, the tasks (and their associated processes) which are the responsibility of the current user are graphically presented in an "in-box" window. At step 906, the process manager waits for the current user to select one of the displayed processes. Once a process is selected, more details about the selected process are displayed at step 908. In addition to graphically presenting the individual tasks in the process, the display may indicate, for example, the process due date, service level agreement, etc.

At step 910, the process manager then waits for the current user to select one of the displayed tasks. Once a task is selected, more details about the selected task are displayed at step 912. For example, such details may include an indication of whether or not (and by whom) the task was completed and an indication of the functional category to which the task belongs. A plurality of fields may also be displayed in a specific region of the screen.
At step 914, if the selected task is the next task to be completed and is the responsibility of the current user, or if the current user is authorized to do so, the current user may fill the fields with data. If the field is simple to fill with data, then the filled field is displayed on the screen (steps 916A and 916B). If the field requires the intervention of a task manager, this task manager can be launched by the current user (steps 918A and 918B). The current user then interacts with the task manager and the data which has filled the field as a result of this interaction is displayed on the screen (step 916B).

At step 920, once the current user is satisfied with the data filling the various fields associated with the selected task, he or she may decide to "enter" the data. At step 922, the process manager passes the data occupying in the various fields to the NMS 10, which then ensures that the appropriate steps for completing the task are taken.

At step 924, the completion of a task by the NMS results in updates to the information which are graphically presented to the current user on the display 21. For example, the just completed task may disappear from the current user's in-box. In addition, at step 926, the display with which other users of the process manager are interacting is also affected by the completion of a task by the current user. For instance, the in-box of the user responsible for the next task in the process will now specify this next task.

In summary, for each process manager launched by a respective user, the process manager graphically interfaces with the user and performs mainly the following functions:

(I) Displays an "in-box" for the user which graphically identifies the tasks in each process that are the responsibility of that user (step 904).
(ii) Accepts task data from the user and forwards the data to the NMS resulting in the completion of individual tasks (step 922).

(iii) As a result of a task being completed, updates the inboxes of the current user and possibly other users having instantiated their own process managers (steps 924, 926).

The process manager of the invention and its associated GUI are now described in the context of a specific example. With reference to Fig. 3, there is shown a process management window 300 displayed within the screen of the display 21. The window 300 appears subsequent to a user launching the process manager from within the NMS 10. The identity of the current user is known to the process manager by virtue of the fact that the user has already logged in to the NMS 10. Alternatively, if the process manager is implemented as a stand-alone application, a login screen may appear prior to the manifestation of window 300, prompting potential users of the process manager for some form of identification. From the identity of the current user, the process manager consults the internal database 29 and is able to determine the authority of the current user and the tasks for which the current user is responsible.

The window 300 includes an in-box 305 containing a graphical depiction of the processes having at least one task which is the responsibility of the current user. Specifically, in-box 305 shows a workflow icon 310 corresponding to a first process entitled "Network Traffic Control" and a workflow icon 315 corresponding to a second process entitled "Request New
Equipment Installation. The priority associated with each process may be indicated next to the title of the process or next to the respective workflow icon.

Workflow icon 315 is shown in more detail in Fig. 2, where it is seen to comprise an arrangement of task elements 205A-H in the form of circles, although other shapes such as polygons are possible. Each task element 205A-H corresponds to a respective task in the process. A task ID number may be located within or next to each task element for numerically identifying the respective task.

The task elements (circles 205A-H) in the workflow icon 315 are aligned in a sequence corresponding to the order in which the respective tasks are to be completed so as to lead to eventual completion of the process. In Fig. 2, for example, the circles 205A-H are arranged from left to right along an imaginary horizontal axis (shown in dotted outline at 210). The leftmost task element 205A corresponds to the task to be completed first and the rightmost task element 205H corresponds to the task to be completed last. Of course, other arrangements, such as top to bottom, are possible, the only requirement being the conveyance of chronological information through the chosen arrangement.

In order to reflect flexibility regarding the sequence in which two or more tasks can be completed, the task elements corresponding to such "flexible" tasks may be aligned differently (e.g., orthogonally) with respect to the axis created by the task elements corresponding to the other "inflexible" tasks (which need to be completed in succession). For example, in Fig. 2, task elements 205B and 205C are stacked vertically as opposed to horizontally.

A task element can also be imparted with an attribute which indicates whether or not the corresponding task has
already been completed (either by the current user or by another user). In Fig. 2, for example, the fill level of circles 205A-H is used to convey whether or not the corresponding task has already been completed. Filled circles 205A-D are used to indicate completed tasks and unfilled circles 205E-H are used to indicate the opposite, i.e., that the corresponding tasks remain uncompleted.

In addition, a task element can, if applicable, be furnished with a visual indication that the corresponding task is the responsibility of the current user. For instance, coloured regions surrounding certain task elements can be used to indicate that the respective tasks are the responsibility of the current user. When the region is darkly coloured (e.g., region 215 surrounding circles 205E and 205F), then the corresponding tasks are to be completed during the current instantiation of the process manager, while a lightly coloured region (such as region 220 surrounding circle 205H) indicates that the corresponding task is indeed executable by the current user, although following the execution by other telco staff of an intervening task corresponding to task element 205G.

By virtue of the presence (or absence) of darkly coloured regions in certain workflow icons in the in-box 305, the current user is instantly made aware of which tasks in which processes he or she is responsible for. This is especially advantageous in assisting new hires who may not yet be familiar with the end-to-end flow of a process.

The current user may then wish to execute one of the tasks for which he or she is responsible and which are surrounded by darkly coloured regions in the in-box 305. Alternatively, the current user may simply wish to obtain information about one of the processes without intending to execute a specific task. In either case, the current user can
access one of the processes displayed in the in-box 305 by selecting the desired workflow icon, e.g., with a mouse.

Alternatively, the current user can initiate a process by clicking on one of several process model tabs or menus in the window 300. The process model tabs or menus can be labelled FULFILMENT, ASSURANCE and BILLING to reflect adherence of each process to the widely accepted “FAB” process model. The FAB process model classifies each telecommunications management process as a fulfilment, assurance or billing process. Individual steps in a process each belong to an FCAPS category, with the result that the flow of a FAB process usually crosses FCAPS boundaries. The FAB process model is more fully described in a document entitled “Telecom Operations Map, GB910, Evaluation Version 1.1”, produced by the Telemanagement Forum in April, 1999 and incorporated by reference herein.

Without loss of generality, it can be assumed that the current user has selected the “Request New Equipment Installation” process represented by workflow icon 315. This action is registered by the process manager which then instructs the display on the screen to change in accordance with the selection. The new display with which the current user now interfaces may resemble that shown in Fig. 4. In-box 305 now contains an indication of which process was selected, in this case represented by a change of color in a region surrounding the title of the “Request New Equipment Installation” process.

Preferably, the window 300 also presents a working area 320 in which general information corresponding to the selected process is displayed. For example, this information may include the name of the process, the customer, the location(s) where the process is taking place, the service level agreement (SLA), the date on which the process was originated and the date by which the process is due to be completed.
In addition, the window 300 preferably displays an enlarged workflow icon 325 corresponding to the selected process. The enlarged workflow icon 325 may contain features in addition to those portrayed by the corresponding workflow icon 315 in the in-box 305. For example, in addition to displaying the task elements, the enlarged workflow icon 325 also contains an indicator (e.g., an arrow) which points to the first unfilled circle in the sequence of circles corresponding to the selected process. In other words, the arrow allows the current user to identify the most urgent uncompleted task in the selected process.

In the particular example of Fig. 4, enlarged workflow icon 325 resembles workflow icon 315 which corresponds to the "Request New Equipment Installation" process. It is recalled that workflow icon 315 comprises circles 205A-H, among which circles 205E and 205F are surrounded by darkly coloured region 215 and circle 205H is surrounded by light coloured region 220. In addition, an arrow 330 points to unfilled circle 205E, indicating the task most urgently requiring execution.

It should be understood that if the arrow 330 were to point to a set of vertically arranged task elements (e.g., circles 205B and 205C), this would indicate that the corresponding tasks are next in line to be completed, although in no particular order amongst the tasks themselves.

In general, the task element pointed to by the arrow 330 could be the responsibility of any user, whether or not it is the current user who actually instantiated the current session of the process manager. In the illustrated example or Fig. 4, however, it so happens that unfilled circle 205E pointed to by the arrow 330 is the first circle (viewed from left to right) surrounded by a darkly shaded region, which signifies
that the corresponding task has indeed to be completed by the
same user who has launched the process manager.

The user may select (e.g., by clicking on) any task
element in the enlarged workflow icon 325. The identity of the
selected task element is fed to the underlying process manager,
which then causes details pertaining to the corresponding task
(known as the "selected task") to be displayed in another
working area 335 in the window 300. Unique to the present
invention are the following features: the selected task is
allowed to be the task to be completed, a previously completed
task, or a task not yet completed and, furthermore, the selected
task need not be one that is the responsibility of the current
user.

It should be understood that without user selection
of a particular task, the task details appearing in working area
335 could be blank or could be those of a default task. The
default task can be the next task to be completed by a member
of the telco staff or the next task to be completed by the
current user, for example. In the present case, these tasks are
one and the same and are represented by circle 205E in the
enlarged workflow icon 325.

The task element corresponding to the selected (or
default) task is identified by graphically enhancing the
corresponding circle on the enlarged workflow icon. In Fig. 4,
the selected task is the one corresponding to task element 205E.

Thus, task element 205E is pointed to by another arrow 340,
which is shown as being larger than arrow 330. An indication
of the identity of the selected task may appear in the vicinity
of arrow 340. Although both arrows 330 and 340 point to the
same task element 205E, it is noted that the position of arrow
330 (corresponding to the next task requiring completion) is
determined by the process manager and NMS, while the position
of the arrow 340 (corresponding to the selected task) is controllable by the current user.

Within working area 335 is displayed another working area 345 which contains generic information pertaining to the selected task. The generic information displayed in working area 345 has the same format for any task and may include a brief description of the selected task, the FCAPS category associated with the selected task and the date by which completion of the selected task is due. If the current user does not have the authority to modify the data in the fields relating to the selected task, this fact may be graphically indicated by displaying a lock icon, for example.

In the specific example of Fig. 4, the generic information displayed in working area 345 specifies that the selected task is the "set parameters" task and indicates by displaying the letter "C" that the selected task belongs to the "CONFIGURATION" functional category. Furthermore, the absence of a lock icon in working area 345 indicates that the user is indeed authorized to modify the data in the fields relating to the "set parameters" task.

Working area 335 also comprises a task details tab 380 and a log tab 385. When task details tab 380 is selected, a task details window 350 appears in working area 335; when log tab 385 is selected, a log window is displayed in working area 335. Preferably, only one of the two windows may be displayed in its entirety, depending on which tab is selected by the user. (In Fig. 4, the task details tab 380 has been selected and therefore a task details window 350 is displayed in working area 335.) Of course, those skilled in the art will appreciate that the task details window and the log window may occupy different portions of the same working area 335.
Although not shown in Fig. 4, the log window can be used for recording information (e.g., statistics) pertaining to the selected task being completed by the current user. The log window tracks task statistics over time, records any user notes and shows the escalation path to be followed in case of difficulty. The escalation path refers to a list of personnel which should be notified and actions which should be taken if the selected task takes longer than a predetermined amount of time. A working area in the log window can be filled by the current user with information, such as messages and warnings, through use of the keyboard or other input device.

The format of the task-specific information displayed in the task details window 350 depends on the nature of the selected task. In Fig. 4, the task details window 350 is shown to comprise a plurality of fields 355A-J which can be filled with data. The way in which data can fill a field varies from one field to another. For some fields (e.g., “time” and “date” fields 355B and 355C, respectively), the field is filled by incrementing or decrementing the value displayed in the field through the use of up and down arrows displayed next to the field in question. For other fields (e.g., fields 355C-J), the field is filled by choosing from a list of options displayed when a button is displayed next to the respective field is pressed.

Still other fields (e.g., “current software” field 355A) are filled through the use of a task manager that is launched by pressing a button displayed next to the respective field. For example, an UPDATE button 360 is displayed next to “current software” field 355A; when the user clicks on the UPDATE button 360, the underlying process manager registers this fact and launches a task manager (and a corresponding graphical user interface) for helping the user complete the selected task.
Fig. 5 shows an example of a task manager window 370 which may appear on top of window 300 when the task manager is launched subsequent to clicking the UPDATE button 360. The task manager launched in this way can be identical to the task manager which would be accessed if the current user were to perform the selected task without the benefit of a process manager. When the task manager is invoked, the process manager passes control of the display to the task manager. The user then fills the fields of the task manager and exits the task manager in a suitable way, e.g., by pressing a DONE button 375 in task manager window 370. Upon exiting the task manager, the process manager regains control of the screen and the user continues to interact with the display of the process manager.

With reference again to Fig. 4, once all data fills all the relevant fields in the task details window 350 and when the log window (not shown, but accessible via tab 385) has been filled with log information, if appropriate, the current user can register the data with the process manager by accessing an entry mechanism. The entry mechanism may consist of the user clicking on a button 390 displayed in working area 355. Of course, the button 390 may have a descriptive title which is more closely associated with the nature of the selected task.

For example, in Fig. 4, the button 390 is entitled “upload parameters”.

It is noted that until the data is entered via the entry mechanism, the data in the various fields only remains stored in local memory. After it is entered, e.g., by pressing the “upload parameters” button 390, the process manager passes the data occupying the various fields 355A-J in the task details window 350 to the underlying network management system (NMS) 10.

The NMS takes the appropriate steps to ensure that the task is completed. After the task has been completed the NMS updates
the local database and the process manager accordingly modifies the current user's display, in particular the in-box 305.

More specifically, as shown in Fig. 6, circle 205E (corresponding to the just completed task) is filled in both workflow icon 315 and enlarged workflow icon 325. In addition, because the task corresponding to task element 205E has been completed, shaded region 210 now surrounds only circle 205F. Moreover, although arrow 340 remains pointed to circle 205E (which is now filled), arrow 330 has advanced to circle 205F corresponding to the next task requiring completion.

Furthermore, changes appear on the display being viewed by some other users who have instantiated the process manager. The affected users are those who have at least one task in their in-box which is part of the process corresponding to workflow icon 315. In general terms, the user responsible for the next task in the process will witness the filling of the circle corresponding to the just completed task and will see a darkly coloured region appear around the task element corresponding to the next task.

(In the case of Figs. 3-6, however, the same user is responsible for executing both the set parameters task represented by task element 205E and the subsequent task, and therefore the region 210 surrounding task element 205F remains darkly coloured.)

With continued reference to Fig. 6, since the current user is also responsible for executing the task corresponding to circle 205F, he or she may wish to proceed with that task after having completed the current task. Alternatively, the current user may simply be interested in browsing through other tasks of the process in order to become familiar with the information displayed for various tasks without necessarily desiring to enter data in any of the fields. To this end, the
current user is allowed to view information associated with different tasks by clicking on a desired task element in the enlarged workflow icon 325. A PREVIOUS STEP button 400 and a NEXT STEP button 405 can be provided in working area 335 for allowing the user to advance or back up on a task-by-task basis. The data occupying the various fields corresponding to the original task can be stored in local fields so that it may be viewed upon return to the original task.

Without loss of generality, it can be supposed that the current user has decided to advance to the next task. To achieve this, the current user can click on circle 205F corresponding to next task in the enlarged workflow icon 325 or, alternatively, on the NEXT TASK button 405 displayed in working area 335. This selection is registered by the process manager and the display is changed to that shown in Fig. 7.

With reference to Fig. 7, therefore, the enlarged workflow icon 325 still shows task element 205E as being filled. Arrow 340 still points to the selected task element, namely circle 205F. Thus, arrows 330 and 340 again point to the same task element. Generic information about the newly selected task is displayed in working area 345 and reveals that the title of the selected task is "check parameter values". The "C" indicates that the "check parameter values" task belongs to the "CONFIGURATION" functional category. Of course, if the newly selected task belongs to a different FCAPS functional category, then a code corresponding to the appropriate functional category will be displayed in working area 345.

Additionally, task-specific information is displayed in a different task details window 350', which now includes a plurality of fields 355' whose contents are to be verified by the current user as part of the "check parameter values" task.

It is noted that task-specific information is displayed
differently in Fig. 7 than in Fig. 5. For example, in the “check parameter values” task at hand in Fig. 7, the user is simply asked to verify the data previously entered as part of the previous “set parameters” task. In the case of the “check parameter values” task, no task manager is required, as verification of parameters is an operation that is usually performed manually. Once these fields are verified by the current user, the “check parameter values” task can be completed by clicking on a “verify parameters” button 390' which acts as the entry mechanism for this task.

After clicking on the verify parameters button 390', the process manager alerts the NMS 10, which then modifies the internal database 29 to indicate that the “check parameter values” task has been completed by the current user. The process manager subsequently modifies workflow icon 315 and modified workflow icon 325 by filling in the corresponding task element, namely task element 205F.

It should be noted that, at any time, the current user is also able to view information pertaining to a previously executed task. In addition, if the current is properly authorized to do so, he or she may even enter data into fields (in the log window, for example) associated with a previously completed task. The current user can access a previously completed task by clicking on the PREVIOUS TASK button 400 as many times as necessary or by clicking on the desired task in the enlarged workflow icon 325. A feature unique to the process manager and graphical user interface of the present invention is that the user is permitted to view the data entered for a previous task (possibly belonging to a different functional category) although he or she may not have been the originator of this information. In addition, the user is permitted to view information about tasks not yet completed, although the insight
gained from such browsing may be limited to knowledge about the
types of fields requiring completion.

Fig. 8 shows the display resulting from the current
user having selected the task corresponding to circle 205D in
the enlarged workflow icon 325 prior to completing the "check
parameter values" task. Arrow 340 therefore points to the
selected task element 205D. The fact that task element 205F is
unfilled indicates that the "check parameter values" task was
not completed prior to accessing the selected task. Thus, arrow
330 remains pointed towards task element 205F.

Working area 345, which displays generic information
relevant to the selected task, shows that the selected task is
entitled "design test sheet". The "F" indicates that the
"design test sheet" task belongs to the "FAULT" functional
category. By virtue of the presence of a lock icon 395 in
working area 345, the current user is alerted to the fact that
he or she is not authorized to modify the data entered the
"design test sheet" task. (For example, the current user may
only have the authority to enter data for tasks in the
"CONFIGURATION" category.) Thus, although a variety of fields
355' are visible in the task details window 350', the current
user is forbidden from entering new data because this task is
outside his or her area of responsibility. To prevent the
current user from entering data, the entry mechanism embodied
by a "download to test manager" button 390' has been disabled
by having been "grayed out".

Should the current user wish to return to the
previously accessed "check parameter values" task represented
by task element 205F and pointed to by arrow 330, he or she can
click on the corresponding task element in the enlarged workflow
icon 325 or twice on the NEXT TASK button 410 in working area 335.
Upon return to the display corresponding to the "check
parameter values” task, the various fields will preferably conserve all the data previously filling the various fields before having navigated to the “design test sheet” task, by virtue of this data having been stored in local memory during browsing.

It is apparent that due to the many differences that exist in the type of information that is displayed for each task and the manner in which it is entered, the number of craftspeople needed to complete a process in its entirety is usually quite large. The present invention allows different users to navigate through tasks in a process using the same graphical user interface and process manager which, through the use of coordinated in-boxes, allows process flow to be tracked across FCAPS boundaries and allows users to be alerted of the completion of tasks as completion occurs. Advantageously, this leads to an increase in the speed and reliability with which data services are delivered to the customer, as well as a considerable reduction in operations and training costs and an improvement in overall process efficiency.

While the preferred embodiment of the present invention has been described and illustrated, it will be apparent to persons skilled in the art that numerous modifications and variations are possible. The scope of the invention, therefore, is only to be limited by the claims appended hereto.
WE CLAIM:

1. A method of managing a process defined by a plurality of tasks capable of being completed by different users and intended to be completed in a particular order, the method comprising:
   a) displaying a plurality of task elements corresponding to the tasks in the process;
   b) graphically identifying the task elements corresponding to tasks which have been completed;
   c) if the user of the method is capable of completing the next uncompleted task, specially identifying the corresponding task element; and
   d) repeating steps b) through d) upon completion of said next uncompleted task.

2. A method as claimed in claim 1, wherein the displayed task elements are arranged in a manner indicative of the order in which the tasks are intended to be completed.

3. A method as claimed in claim 2 or 3, further comprising: displaying a graphical indicator proximate the task element corresponding to the next uncompleted task.

4. A method as claimed in claim 3, wherein said indicator is an arrow.

5. A method as claimed in any one of claims 1 to 4, wherein each task element has a visual attribute indicative of whether or not the corresponding task is capable of being completed by the user.
6. A method as claimed in any one of claims 1 to 5, further comprising:
   in response to user selection of a task, displaying information associated with the selected task.

7. A method as claimed in claim 6, further comprising:
   displaying a graphical indicator proximate the task element corresponding to the selected task.

8. A method as claimed in claim 7, wherein the indicator is an arrow.

9. A method as claimed in claim 7, wherein the information includes a graphical indication of whether the selected task is capable of being completed by the user.

10. A method as claimed in claim 9, wherein the indication is provided by a lock icon.

11. A method as claimed in claim 7, wherein each task belongs to one of a plurality of functional categories and wherein the information includes a graphical indication of the functional category to which the selected task belongs.

12. A method as claimed in claim 7, wherein the information includes statistical data pertaining to the selected task.

13. A method as claimed in claim 7, wherein the information includes a plurality of fields, wherein the fields contain data if the selected task has already been completed and
wherein the fields require completion by the user if the selected task has not been complete.

14. A method as claimed in claim 13, wherein selected ones of the fields are completed by launching a task manager.

15. A method as claimed in claim 13, further comprising:
if the user is capable of completing the selected task, displaying a graphical entry mechanism for confirming the data occupying said fields; and
in response to the current user accessing the entry mechanism, registering the task as having been completed and displaying updated information related to the selected task.

16. A method as claimed in claim 15, further comprising forwarding said data to a network management system.

17. A method as claimed in any one of claims 1 to 16, further comprising:
consulting a database to determine the tasks which the user is capable of completing.

18. A method as claimed in claim 7, wherein selection of a task comprises user selection of one the corresponding displayed task elements.

19. A method as claimed in claim 7, further comprising:
displaying a browsing mechanism for travelling between displayed task elements;
wherein selection of a task comprises using the browsing mechanism until the corresponding task element is reached.
20. The method of any one of claims 1 to 19 implemented in a network management system.

21. A computer system adapted to carry out the method of any one of claims 1 to 19.

22. A graphical user interface adapted to carry out the method of any one of claims 1 to 19.

23. A computer readable medium having software stored thereon for instructing a computer to carry out the method of any one of claims 1 to 19.

24. A method of managing a process defined by a plurality of tasks assigned to be completed by corresponding users in a predetermined order, the method comprising:

a) graphically informing the user assigned to complete the next uncompleted task that said next uncompleted task requires execution; and

b) upon completion of said next uncompleted task, graphically informing the user assigned to complete the task following said next uncompleted task that said following task requires execution.

25. A method as claimed in claim 24, further comprising:

c) redefining said following task to be the next uncompleted task; and

d) repeating steps b) and c) until the last task in the process is completed.

26. The method of claim 24 or 25 implemented in a network management system.
27. A computer system adapted to carry out the method of any one of claims 24 to 26.

28. A graphical user interface adapted to carry out the method of any one of claims 24 to 26.

29. A computer readable memory for storing statements or instructions for use in the execution in a computer of a method according to any one of claims 24 to 26.
FIG. 1

FIG. 2
FIG. 4