(51) International Patent Classification 6 :		(11) International Publication Number:	**WO 00/33173**
G06F 3/44	**A1**	(43) International Publication Date:	8 June 2000 (08.06.00)

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: TERNULF, Yngve [SE/SE]; Maskinkajen 9, S–417 64 Göteborg (SE). POUR-MAKHDOMI, Shahram [SE/SE]; Hasselvägen 25, S–435 38 Mölnlycke (SE). OTREUS, Finn [SE/SE]; Flädervägen 11, S–435 38 Mölnlycke (SE).

(54) Title: METHOD FOR INSERTING OBJECTS INTO A WORKING AREA IN A COMPUTER APPLICATION

(57) Abstract

The present invention relates to a method for facilitating the insertion of an object in a working area on a computer display, and is implemented in a computer application software. The method comprises the steps of indicating at least one subarea of the working area where an object is insertable, indicating an object type in association with each subarea, an object of said object type being insertable in said subarea, receiving input from the user selecting one of the at least one subarea, and inserting in the selected subarea an object of the type that is indicated at the selected subarea. Through this method, the user only needs to provide one single piece of input in order to insert an object in a valid location in a working area.

1

METHOD FOR INSERTING OBJECTS INTO A WORKING AREA
IN A COMPUTER APPLICATION

Technical field

The present invention relates to a method for fascilitating the insertion of an object into a working area on a computer display, said method being implemented

5   in a computer application software and comprising the step of receiving input from the user selecting where on the screen an object of a previously specified object type is to be inserted.

10   ### Background art

In numerous computer applications, objects are selected by a user and inserted into an area of a computer display. This is especially frequent in different kinds of programming and designing, where

15   objects belonging to several different object types are available to the user for insertion into a working area, such as a worksheet, a document, or any other limited area of a screen, used by the active software application to display objects. The user first clicks on a particular

20   type of object (e.g. a certain type of programming item) and then clicks on the location where an object of the particular type is desired.

In some applications a drag-and-drop procedure is utilized, meaning that the user clicks on the object type

25   and without releasing the mouse button moves the cursor to the desired location and then releases the button. An object of the chosen type is hereby inserted at this location. This functionality is implemented in e.g. Viso from Visio Corp.

30   In other applications the cursor changes its appearance when an object type is clicked on, to indicate that an object will be inserted at the location of the

2

next click. This functionality is implemented in e.g.
Corel Draw.

     In some cases, objects of certain object types
should only be inserted into specific parts of the
5   working area, or in connection with objects of other
predetermined object types. Particularly in certain types
of programming, e.g. network PLC-programming, where the
overall structure has to follow a predetermined pattern,
this is a common situation. Inserting an object into a
10 location where it is not ment to be located may result in
a non-working object, or even in a non-working program,
with tedious error searching as a result.

     It is known in the art to implement test routines
for each insertion, in order to avoid erroneous placement
15 of objects. This can either be accomplished by making
certain object types unavailable to the user unless
certain predetermined conditions are met, or by refusing
the insertion of a selected object into certain areas.
However, the user then finds him/herself in a situation
20 where the intended operation is impossible to perform,
e.g. the object he/she intends to insert into a certain
location is refused by the application. This is
frustrating, especially as the user does not obtain
information regarding why the object cannot be inserted,
25 nor regarding what object can be inserted instead.

     An even larger problem with known methods for
inserting objects of different types into a working area
is the fact that the cursor always needs to be moved
between the icons or menus from which the object type is
30 chosen, and the desired location.

     Another problem is the icons or menus themselves. If
menus are used, it can be difficult to quickly find the
desired object type. Sometimes several sub-menus need to
be opened in order to find a specific object type. If
35 icons are used they normally provide a better overview,
but instead they take up valuable space that is needed
for other purposes.

Summary of the invention

A first object of the present invention is therefore to provide a method for inserting an object into a working area on a computer screen, that simplifies the procedure.

A second object of the present invention is to provide a method for inserting an object into a working area on a computer screen, wherein the need for icons or menus is eliminated.

A third object of the present invention is to provide a method for inserting an object into a working area on a computer screen, wherein the problem of attempting to make invalid insertions is avoided.

These objects are accomplished by using a method according to the preamble of claim 1, further characterized by the steps of indicating at least one subarea of the working area where an object is insertable, indicating an object type in association with each subarea, an object of said object type being insertable into said subarea, receiving input from the user selecting one of said at least one subarea, and inserting into the selected subarea an object of the type that is indicated at the selected subarea.

This results in a simple procedure for inserting objects into the working area. Instead of having to first select an object type, and then to select a location, the user only has to make one selection. A software implementing the method according to the invention indicates to the user locations where insertions are possible, and what type of object is insertable at each location. The input needed from the user to perform an insertion is thus reduced from two to one. According to the invention, one single piece of input is enough to insert an object at a valid location in the working area.

Further, the need for menus and icons is eliminated, as the user is given clear indications of what objects

are insertable into the working area. There is no need to hide, or make unavailable, contents in a menu, as it is only available objects that are displayed.

The method according to the invention is preferably used when restrictions are imposed on what objects can be inserted into the working area, and where they can be inserted. These restrictions can be caused for example by relationships between programming steps constituting a computer program, said steps being represented by the objects, or by physical relationships between processes or physical entities, said processes or entities being represented by the objects.

The inventive method avoids the above mentioned problem of making invalid insertions. These insertions can not even be attempted, as the user, even before the action to insert an object is performed, is made aware of what type of an object is insertable at a certain location.

A software implementing the method performs an identification of all possible, valid insertions of objects into the working area and indicates them to the user. The user only has to decide which of the insertions to perform.

According to one embodiment of the invention, the indication of said at least one subarea of the working area where an object is insertable includes graphically outlining said subarea.

According to another embodiment of the invention, the indication of said at least one subarea is activatable and deactivatable by the user. This makes it possible for the user to examine the results of his /her work without the indications of subareas visible. When the user intends to insert a new object, the indication is simply activated by giving a certain input to the software.

Input from the user is preferably received using a pointing device, preferably in electronic contact with

5

the software application and controlling a cursor on the
display.

This makes it very easy for the user to provide
input to the software application regarding where in the
5   working area the user wishes to insert an object.

According to a further embodiment of the invention,
the step of indicating at least one subarea of the
working area where an object of a type that is insertable
can be performed in response to the action of moving the
10   cursor into said subarea.

In this way, the step of indicating a subarea is
closely associated with the user activity of moving the
mouse. No indications of subareas are displayed in the
working area unless the user actually moves the cursor in
15   the working area.

The step of indicating an object type in association
with each subarea preferably comprises displaying a
symbol representing said object type in connection to
said subarea. According to one embodiment of the
20   invention, this symbol is displayed by changing the
appearance of the cursor, in response to the action of
moving the cursor into said subarea.

One application of the method is in the field of
network design. Hereinafter, the term "network" is used
25   for any collection of objects, operations, events,
processes, options, conditions etc. that logically relate
to each other and form part of a system. Networks can
appear in many contexts, for example organisation charts,
circuit block diagrams or computer programs, and can be
30   designed, or programmed, with the help of a computer in
the above mentioned way by inserting objects of different
types into a working area. Network programming is
particularly useful when designing systems for automatic
control, for example PLC-systems.

35   Normally, several different object types are
available, where each type has specific physical
properties. One object type can for example represent

physical events, while another might represent a
conditional step, with several possible outcomes.

    Brief description of the drawings
5   Preferred embodiments of the present invention,
given by way of example only, will now be described in
more detail, with reference to the accompanying drawings.
        Fig 1 illustrates an environment in which the method
according to the invention may be used.
10   Fig 2 is a schematic block diagram of a software
application according to prior art.
        Fig 3 is a schematic block diagram of a software
application according to the present invention.
        Fig 4 is an example of a working area on a display.
15   Fig 5 is a flowchart of a first embodiment of the
method according to the invention.
        Fig 6 is a flowchart of a second embodiment of the
method according to the invention.
        Fig 7 gives some examples of how indication of
20   subareas and object types can appear in the working area
7 of the display.

    Detailed description of preferred embodiments of the
invention
25   Fig 1 illustrates an environment in which the method
according to the invention can be utilized. A software
application 1, e.g. a programming tool, is installed and
running on a computer 2. A user 3 interacts with the
application 1, giving instructions 9 to the software
30   application through a keyboard 4 and with a pointing
means, preferably a mouse 5. The application presents the
results of the user's actions on a display unit,
preferably a monitor 6, displaying to the user a working
area 7, where operations corresponding to said actions
35   are performed. Said operations may include inserting
objects 8 into the working area 7, where they represent
for example program actions or physical events.

Fig 2 shows a software application 1' according to
prior art, suited for the insertion of objects into a
working area. Said application 1' typically includes
means 11 for receiving input from the user, means 12 for
5   deciding how to respond to the input from the user, and
means 13 to insert objects into the working area 7.

Fig 3 shows a software application 1 according to
the present invention. The software comprises
identification means 15 for identifying valid insertions
10   of objects 8 into the working area 7. This means 15 is
similar in structure to the means 12 in prior art, but
with the important difference that it operates without
waiting for user input. Instead of, as in prior art,
controlling whether an insertion of an object requested
15   by the user is in fact a valid insertion, the means 15
are operative to constantly identify locations in the
working area where objects 8 are insertable, and what
type of object is insertable in every location. In short,
the means 15 anticipate all possible insertions that can
20   be requested by the user, and identifies the ones that
are valid.

Further, the application 1 is provided with
indication means 16, for indicating information to the
user on the monitor. Again, the structure of means 16 is
25   not in itself unique, and similar indication techniques
can be found in many software applications. According to
the invention, however, the indication means 16 use
identification data 17, supplied by the identification
means 15, to indicate all available, valid insertions to
30   the user in the present situation.

The software application 1 also comprises reception
means 18 and insertion means 19, similar to the reception
means 11 and the insertion means 13 of the prior art,
respectively.

35   In the present context, the means 15, 16, 18, 19 are
software routines included in the application 1. Of
course, and as is well understood by a person skilled in

8

the art, in some cases one or several of these means 15, 16, 18, 19 can be comprised of separate software, or software associated with e.g. the monitor or mouse, said software operating together with the software application
5   1.

In the following description of the method according to the invention, said software application 1 is a program network design tool, a working area 7 of which is shown in fig 4. This type of programming is used for
10   example in PLC-programming.

In the illustrated example, the program network 20 comprises two types of items, namely step items 21, which represent one or several physical actions, and transition items 22, which define conditions for moving the program
15   control from one step item to another. Each item is represented by an object 21, 22 in the working area 7, and the relationship between the two types of items impose restrictions on how items can be inserted into the working area. For example, two step items cannot be
20   connected to each other without an intermediate transition item.

Reference numerals 23a, 23b, 23c, 23d indicate four of several subareas in the working area 7, where objects are insertable. The subareas are naturally located in
25   connection to the existing network, as this is where new objects can be inserted.

In the illustrated example, the subareas 23a, 23b, 23c, 23d are outlined by rectangles 25, which identify and limit each subarea. Of course, any indication using
30   graphics, colour or text is possible. Further, in the vicinity of each subarea 23a-d, an object type 26a-d is indicated. Again, in the illustrated example, the object type 26a-d associated with each subarea 23a-d respectively, is indicated by a symbol which represents
35   the associated object type 26a-d.

In the illustarted example, an inserted object normally includes an step item 21 and a transition item

9

22. Depending on where the object is inserted, however, the order of the two items vary. In subarea 23a, an object consisting of a transition item followed by an step item is insertable. This object type is represented
5   by the symbol 26a. In subarea 23b, the order is reversed, and this object type is represented by the symbol 26b.

Another possible object is a fork 24a, 24b, which indicates a division of the network path. A fork can be inserted before (23c) or after (23d) a transition, and
10 different symbols, 26c and 26d respectively, are associated with the two types of forks.

Fig 5 illustrates schematically a first embodiment of the method according to the invention. In step 30, the identification means 15 identifies subareas 23a, 23b of
15 the working area where objects are insertable. Further, in step 31, the identification means 15 identifies what type of object can be inserted in each subarea. As mentioned, following a step item there must be a transition item and vice versa.

20   When the identification of subareas and insertable object types is performed, the program control waits (in step 32) for the user to activate a grid 25, 26a-d. This activation can be accomplished by hitting a key on the keyboard 4 or by using the mouse. Reception means 18 are
25 arranged to receive and respond to such input by instructing the indication means 16 to activate the grid. The grid 25, 26a-d can take on any number of forms, but has the purpose of indicating a plurality of subareas in the working area 7 (step 33 in fig 5), and in association
30 with each subarea an object type (step 34 in fig 5). Both steps 33 and 34 are performed by the indication means 16, which have received identification data 17 from the identification means.

The program control now waits for the user to click
35 in one of the subareas (step 35), preferably using the mouse 5. Once a click in a subarea is detected by the reception means 18, the insertion means 19 inserts an

object of the indicated object type into the subarea. For
example, if the user clicks in subarea 23a, an object of
type 26a, consisting of a transition item and an step
item is inserted into the subarea. Note that the user
5   accomplishes the entire insertion with just one click. As
an insertable, valid object type is associated with each
subarea, a selection of one of the subareas is sufficient
for inserting an object.

        This insertion affects the network, typically
10   extending it to include more items. The result of the
insertion described above is the network 20' shown in fig
4b. The program control now instructs the identification
means 15 to reidentify subareas and insertable object
types (step 37 and 38). This results in an update of the
15   grid in step 39 and 40, again using the indication means
16.

        The program control then returns to step 35, and
again waits for the user to click in a subarea, for the
insertion of a new object. Of course, when the program
20   control waits at for example step 35, the user can choose
to deactivate the grid or to peform other actions in the
application, such as provide an item with additional
data, define parameters etc.

        Another embodiment of the method according to the
25   invention is showed in fig 6. Just as in fig 5, the first
two steps 42, 43 identify subareas and insertable object
types. Then, however, the program control waits for the
user to move the mouse cursor into one of the subareas
before (step 44). When the reception means 18 receive
30   information of this situation, the indication means 16
are instructed to indicate this particular subarea (step
45) and the object type that is insertible into this
subarea (step 46).

        This indication technique is illustrated in fig 4b
35   and 4c, where the cursor 28 is located in a subarea 23',
23'', resulting in the indication of the subarea and

object type with a rectangle 25′, 25′′ and a symbol 26′,
26′′.

The program control then waits for a user action
(step 47). If the user moves the mouse cursor out of the
5   subarea (move), the indications of subarea and insertable
object type are removed (step 48) and program control
returns to step 44.

If the user does not move the cursor, but instead
clicks in said subarea (click) an object is inserted into
10   the subarea (step 49). This insertion affects the
network, typically extending it to include more items.
Therefore, the program control now instructs the
identification means 15 to reidentify subareas and
insertable object types (step 50 and 51), before
15   returning to step 44.

Of course, it is possible, and even advantageous, to
implement a combination of the two embodiments described
above. For example, a grid including only the rectangles
25 can be activated and deactivated, and at the same time
20   the mouse cursor appearance changes to indicate what type
of object is insertable into a subarea.

In some cases, for example when a fork 24a, 24b is
inserted into the working area 7, it can be necessary to
indicate a second location 27a, 27b, nemely where the
25   fork rejoins with the main path of the network. When this
is the case, the insertion of an object 36, 49 is
followed by a program step 52, 53 waiting for additional
user input.

If, in one subarea, objects of different object
30   types are insertable, this can be indicated by dividing
the rectangle 25 into several smaller fields, each with
an object type indicated in its vicinity. Alternatively,
a list or table can be displayed beside the subarea,
indicating different insertable objects.

35   Another solution is to indicate one object type when
the cursor is moved into the subarea. Then, when the user
presses the mouse button, alternative object types are

12

indicated. The user can then move the cursor to the
desired object type, and release the button in this
location, resulting in an insertion of an object of this
type. Note that the user still only makes one click with
5   the mouse, although a cursor move is possibly made
between the press and release actions.

Another solution, preferred by the applicant, is to
indicate several subareas of the standard size, next to
each other, each one with one associated object type.
10   This is illustrated in fig 4a, where objects 26a/26c and
26b/26d are insertable at the same location. When the
user selects one of the subareas 23a/23c, 23b/23d, an
object of the type indicated in that particular subarea
is inserted, and the surrounding subareas are hidden or
15   moved in the update following the insertion.

Many variations of the method according to the
present invention are possible to a person skilled in the
art. The appearance of the indications can, for example,
be changed without loosing the functionality intended in
20   the claims.

13

CLAIMS

1. Method for fascilitating the insertion of an
object into a working area (7) on a computer display (6)
5   said method being implemented in a computer application
software (1) and comprising the step of receiving input
(9) from the user (3) selecting where on the screen an
object of a previously specified object type is to be
inserted,
10   c h a r a c t e r i z e d by the steps of
indicating (33; 45) at least one subarea (23a-d;
23'; 23'') of the working area (7) where an object is
insertable,
indicating (34; 46) an object type (26a-d; 26';
15 26'') in association with each subarea (23a-d; 23';
23''), an object of said object type being insertable
into said subarea,
receiving input (35; 47) from the user selecting one
of said at least one subarea, and
20   inserting (36; 49) into the selected subarea an
object of the type that is indicated in association with
the selected subarea.

2. Method according to claim 1, wherein the step of
25 indicating (33; 45) at least one subarea of the working
area where an object is insertable comprises graphically
outlining said at least one subarea.

3. Method according to claim 1 or 2, wherein the
30 indication (33) of said at least one subarea is
activatable and deactivatable by the user (3).

4. Method according to claim 1-3, wherein input (9)
from the user is received using a pointing device (5).
35

5. Method according to claim 4, wherein the pointing
device (5) is in electronic contact with the computer

14

application (1) and controls a cursor (28) on the display
(6).

6. Method according to claim 4 or 5, wherein the
step of indicating (45) at least one subarea (23'; 23'')
of the working area where an object of a type that is
insertable comprises graphically outlining said subarea
when the cursor (28) is moved into said subarea.

7. Method according to claim 4 - 6, wherein the step
of indicating (34; 46) an object type in association with
each subarea comprises displaying a symbol (26a-d)
representing said object type in connection to said
subarea.

8. Method according to claim 5 - 7, wherein the step
of indicating (46) an object type in association with
each subarea (23'; 23'') comprises changing the
appearance of the cursor (28).

9. Method according to one of the previous claims,
wherein the object types represent various physical items
and are inserted into the working area to create a
logical network.

10. Method according to claim 9, wherein the network
represents a system for automation.

11. Computer-readable medium, on which is stored
instructions for one or several general purpose computers
(2), comprising means (15, 16, 18, 19) for enabling said
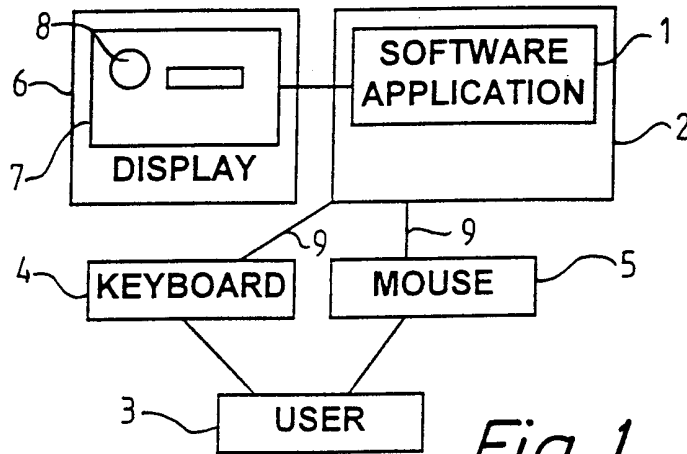one or said several computers (2) to perform the steps of
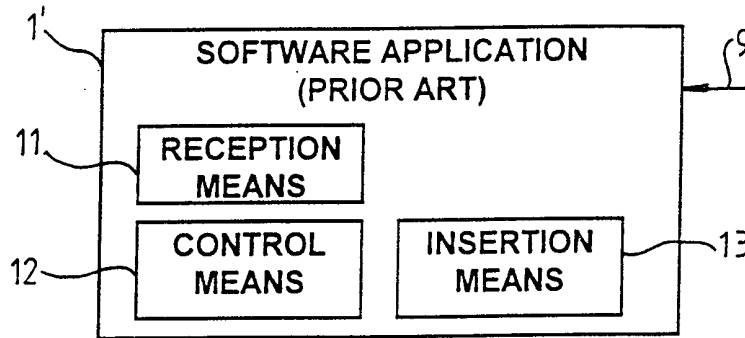the method according to claim 1 - 10.
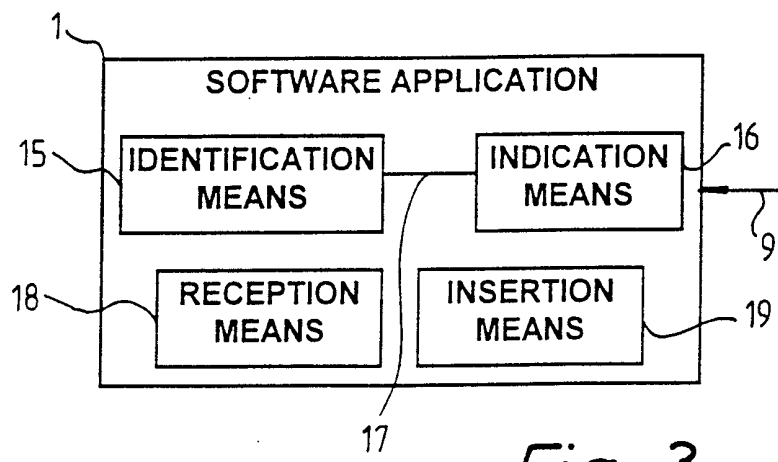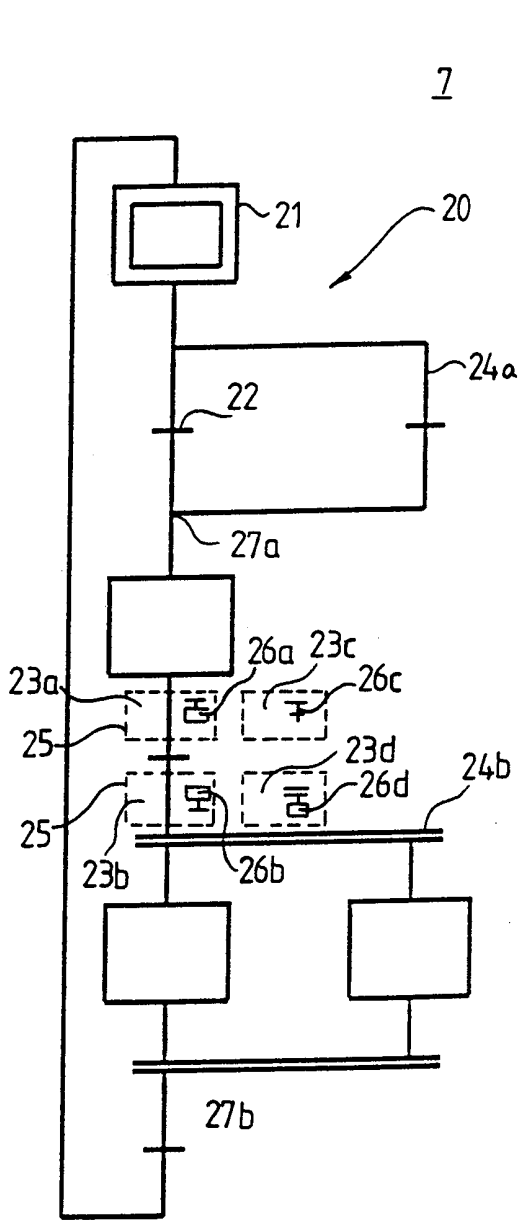
Fig. 1

Fig. 2

Fig. 3

Fig. 4a

Fig. 4b

Fig. 4c

30 — IDENTIFY
SUBAREAS

31 — IDENTIFY
INSERTABLE
OBJECT TYPES

32 — WAIT FOR USER
TO ACTIVATE
GRID

33 — INDICATE
SUBAREAS

34 — INDICATE
INSERTABLE
OBJECT TYPES

35 — WAIT FOR USER
TO CLICK IN
SUBAREA

36 — INSERT OBJECT
INTO SUBAREA

37 — REIDENTIFY
SUBAREAS

38 — REIDENTIFY
INSERTABLE
OBJECT TYPES

39 — INDICATE
SUBAREAS

40 — INDICATE
INSERTABLE
OBJECT TYPES
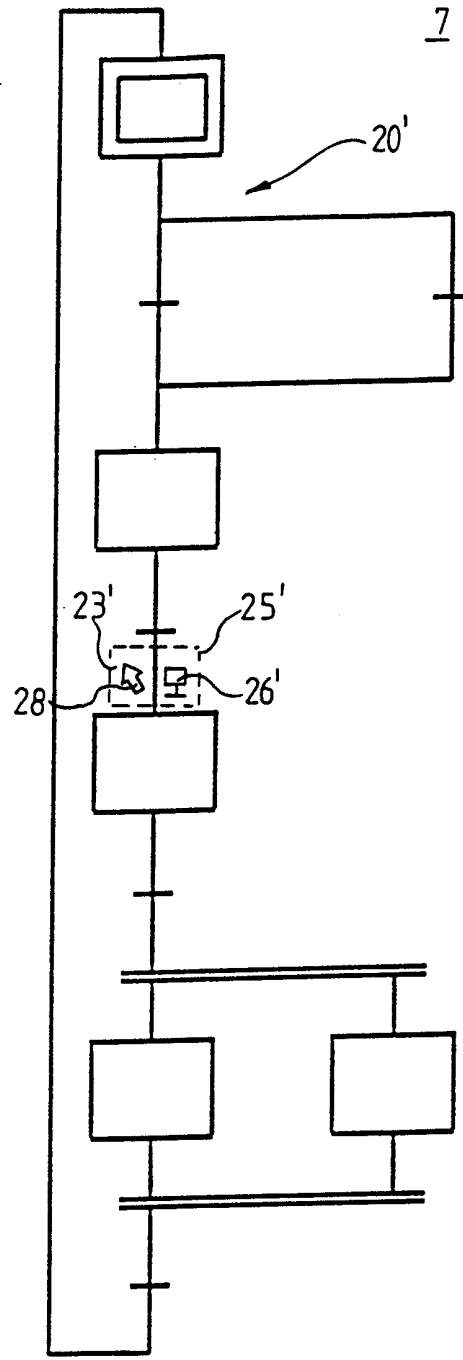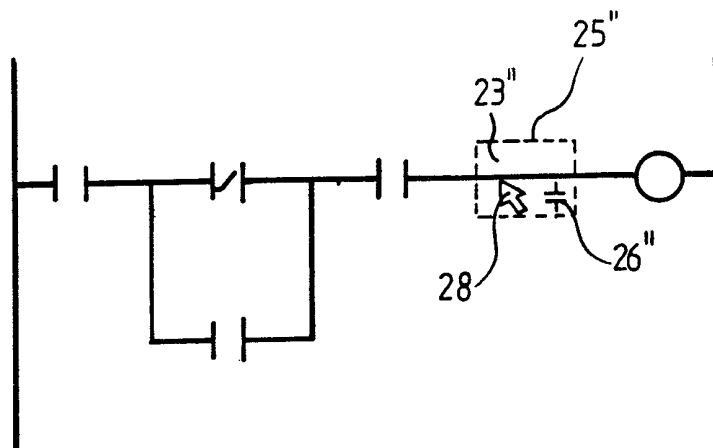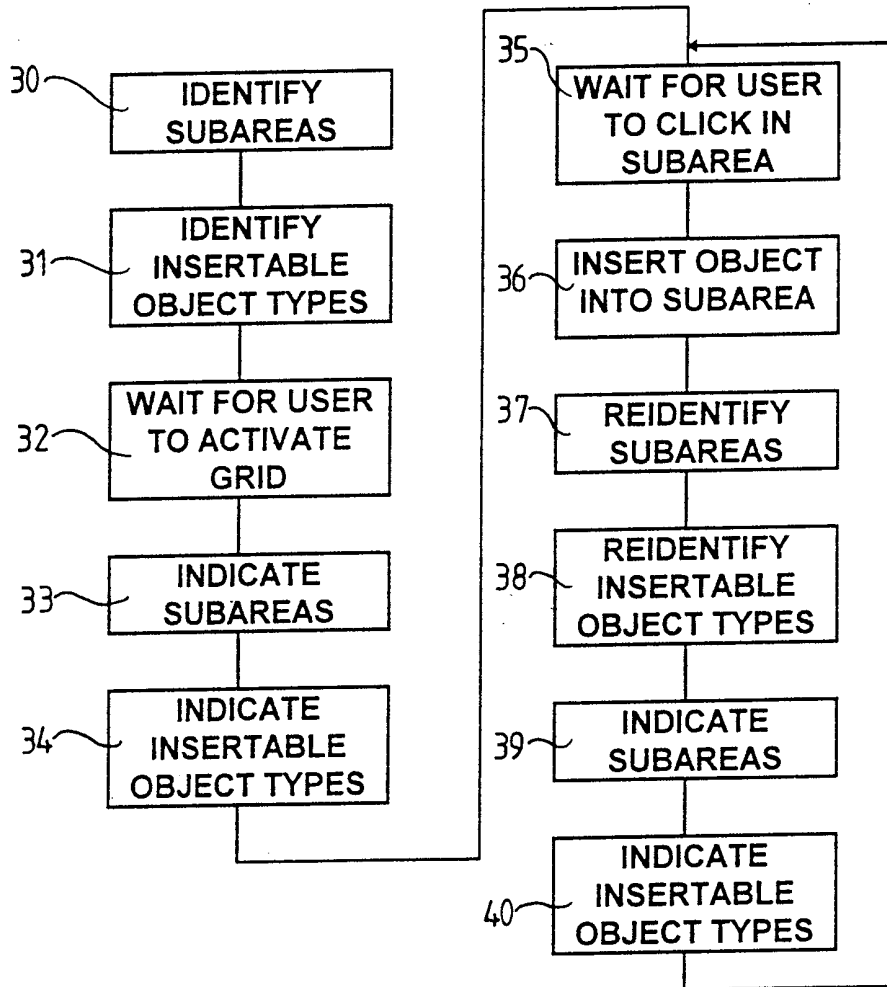
Fig. 5
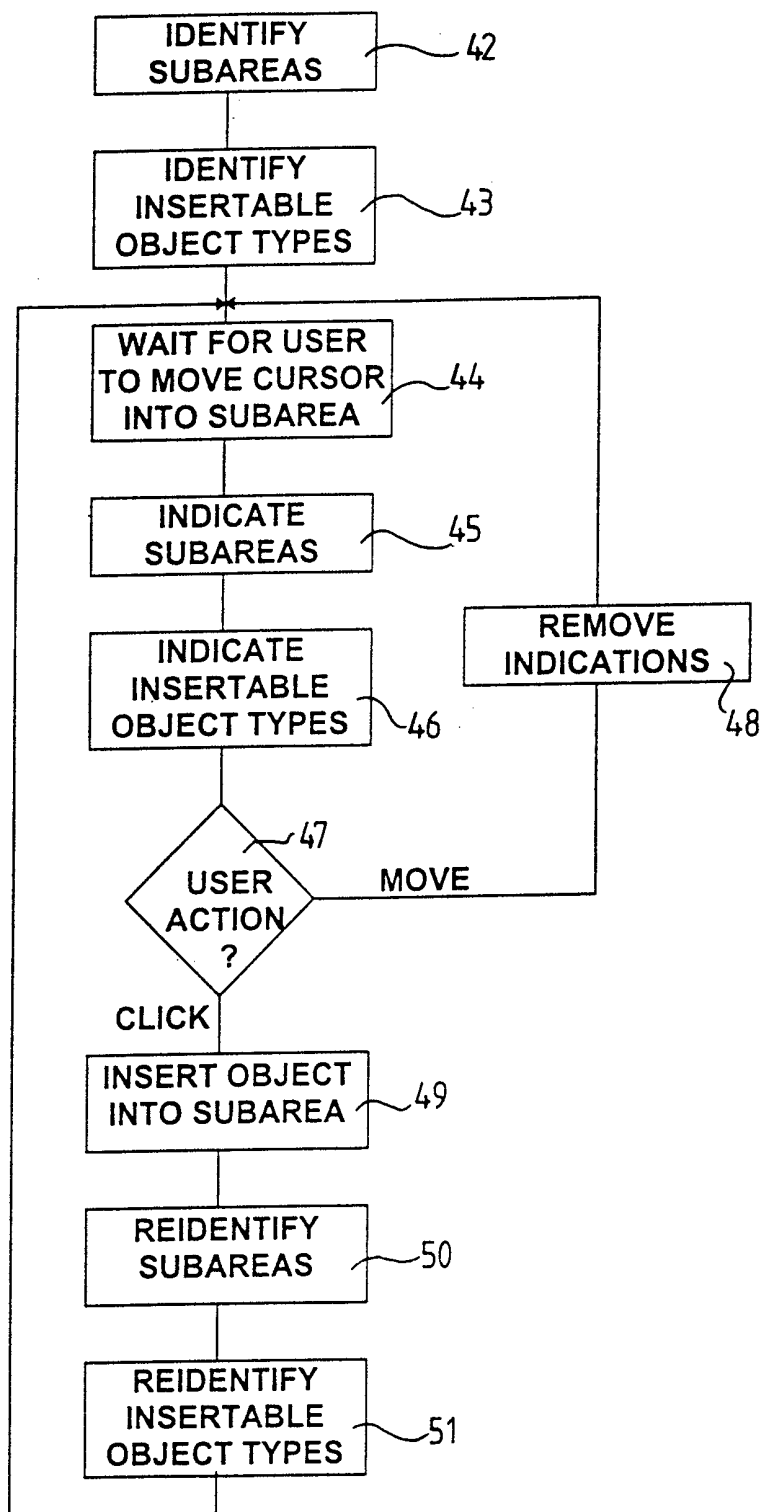
Fig. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/SE 98/02183

A. CLASSIFICATION OF SUBJECT MATTER

IPC6: G06F 3/44

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)

IPC6: G06F

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

SE,DK,FI,NO classes as above

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

WPI, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5481712 A (WILLIAM M. SILVER ET AL), 2 January 1996 (02.01.96), column 1, line 35 - column 4, line 44, figures 1-3, claims 1-34, abstract	1-11
A	US 5524238 A (JOHN A. MILLER ET AL), 4 June 1996 (04.06.96), see the whole document	1-11
A	WO 9806219 A1 (STARSIGHT TELECAST, INCORPORATED), 12 February 1998 (12.02.98), page 2, line 31 - page 3, line 17; page 4, line 9 - line 13	1-11

☒	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		
"E"	erlier document but published on or after the international filing date	"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
"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)	"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
"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	"&"	document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
17 Sept 1999	2 2 -09- 1999
Name and mailing address of the ISA/	Authorized officer
Swedish Patent Office Box 5055, S-102 42 STOCKHOLM	Göran Magnusson/mj
Facsimile No. + 46 8 666 02 86	Telephone No. + 46 8 782 25 00

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International application No.

PCT/SE 98/02183

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 4742473 A (JOEL K. SHUGAR ET AL), 3 May 1988 (03.05.88), abstract	1-11
	--	
E,A	US 5862379 A (ROBERT V. RUBIN ET AL), 19 January 1999 (19.01.99), claims 1-24, abstract	1-11
	--	
E,A	WO 9928794 A2 (KLöCKNER-MOELLER GMBH), 10 June 1999 (10.06.99), see the whole document	1-11
	--	

INTERNATIONAL SEARCH REPORT

Information on patent family members

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
US 5481712 A	02/01/96	NONE		
US 5524238 A	04/06/96	NONE		
WO 9806219 A1	12/02/98	AU	3908297 A	25/02/98
		EP	0919097 A	02/06/99
US 4742473 A	03/05/88	NONE		
US 5862379 A	19/01/99	NONE		
WO 9928794 A2	10/06/99	NONE		