Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2362618 C 2006/07/11

(11)(21) 2 362 618

(12) BREVET CANADIEN
CANADIAN PATENT

(13) C

(86) Date de dépôt PCT/PCT Filing Date: 2000/03/15
(87) Date publication PCT/PCT Publication Date: 2000/09/21
(45) Date de délivrance/Issue Date: 2006/07/11
(85) Entrée phase nationale/National Entry: 2001/09/04
(86) N° demande PCT/PCT Application No.: EP 2000/002310
(87) N° publication PCT/PCT Publication No.: 2000/055760
(30) Priorité/Priority: 1999/03/15 (DE199 11 461.7)

(51) Cl.Int./Int.Cl. *G06F 13/00* (2006.01), *G06K 15/02* (2006.01), *G06F 17/21* (2006.01), *G06F 13/10* (2006.01), *G06F 17/24* (2006.01), *G06K 15/00* (2006.01), *G06F 3/12* (2006.01)

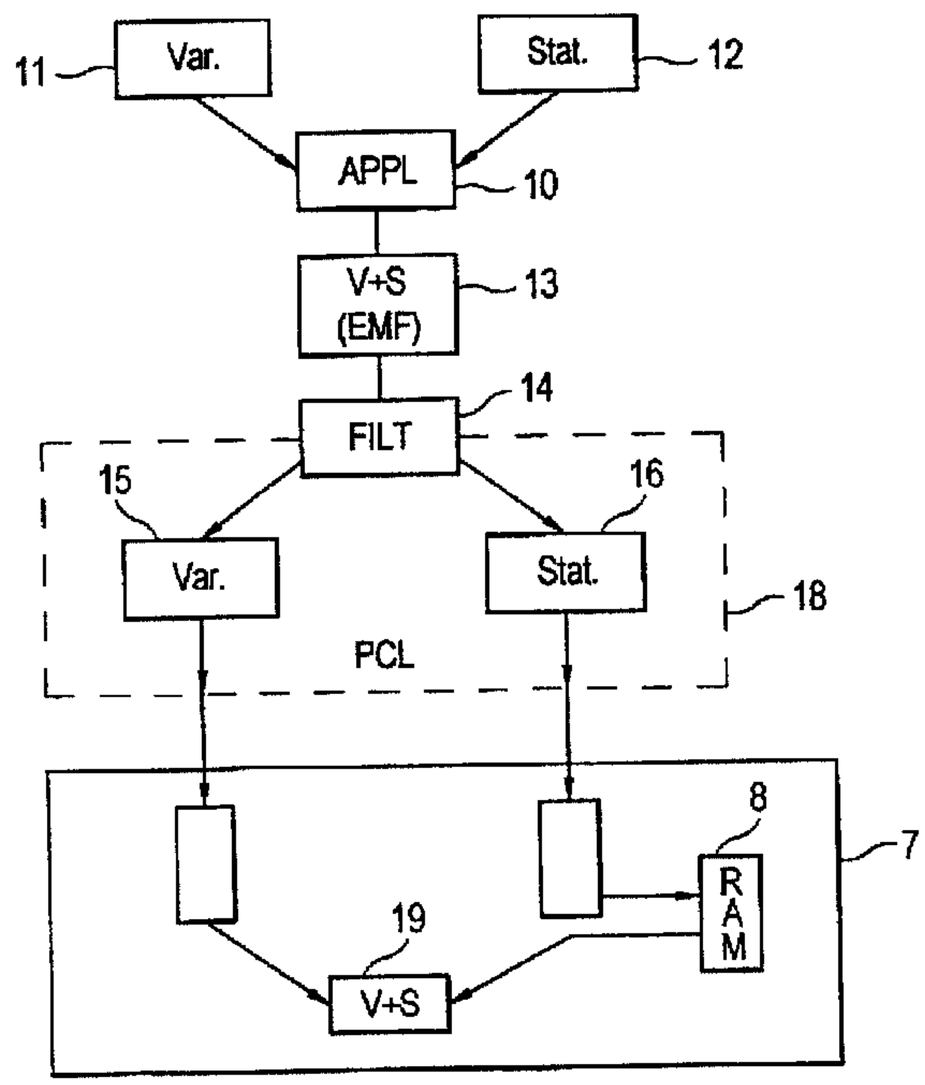
(72) Inventeurs/Inventors:
SCHWIER, HARTWIG, DE;
GREBNER, JURGEN, DE

(73) Propriétaire/Owner:
OCE PRINTING SYSTEMS GMBH, DE

(74) Agent: FETHERSTONHAUGH & CO.

(54) Titre : PROCEDE, PROGRAMME INFORMATIQUE ET SYSTEME PERMETTANT LE TRANSFERT DE DONNEES INFORMATIQUES A UN APPAREIL DE SORTIE

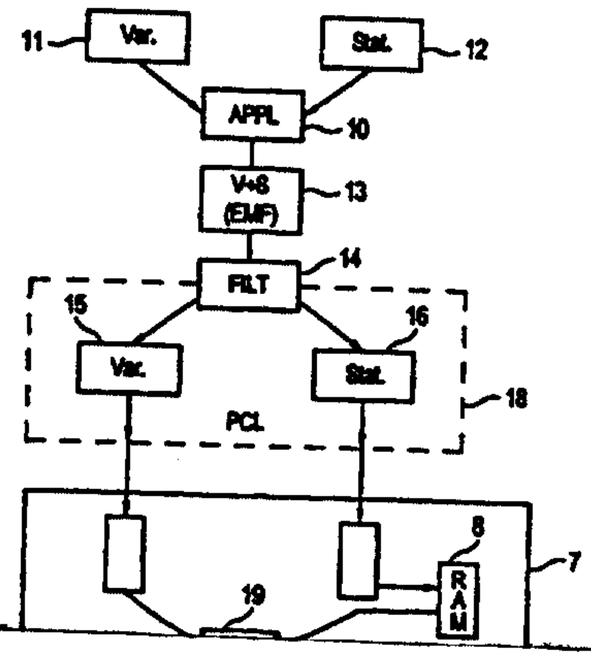
(54) Title: METHOD, COMPUTER PROGRAM PRODUCT AND SYSTEM FOR THE TRANSMISSION OF COMPUTER DATA TO AN OUTPUT DEVICE



(57) Abrégé/Abstract:
The invention relates to a method for outputting data from a computer system (1) to an outputting device (7). According to said method, a master document (25) with a variable data area (27, 28, 29) and a static data area (12, 26) is generated. The variable data area (27, 28, 29) is marked and variable data (11) are placed in said variable data area (27, 28, 29). This produces a serial data flow (31) with individual documents (32, 34). In said serial data flow, the variable data and the static data are permanently linked according to individual documents. The variable data (11, 30) are then separated from the static data (12, 33) from this serial data flow (31), using the marking. The variable data (11, 30) from the first individual document (32) are transmitted to the outputting device (7) separately from the static data (12, 33) and stored there. The static data (12, 33) of the succeeding individual documents are not transmitted to the outputting device (7). In the outputting device (7), the variable data (11, 30) are then re-linked with the stored static data (12, 33) according to the individual documents.

(57) Abstract

The invention relates to a method for outputting data from a computer system (1) to an outputting device (7). According to said method, a master document (25) with a variable data area (27, 28, 29) and a static data area (12, 26) is generated. The variable data area (27, 28, 29) is marked and variable data (11) are placed in said variable data area (27, 28, 29). This produces a serial data flow (31) with individual documents (32, 34). In said serial data flow, the variable data and the static data are permanently linked according to individual documents. The variable data (11, 30) are then separated from the static data (12, 33) from this serial data flow (31), using the marking. The variable data (11, 30) from the first individual document (32) are transmitted to the outputting device (7) separately from the static data (12, 33) and stored there. The static data (12, 33) of the succeeding individual documents are not transmitted to the outputting device (7). In the outputting device (7), the variable data (11, 30) are then re-linked with the stored static data (12, 33) according to the individual documents.



METHOD, COMPUTER PROGRAM PRODUCT AND SYSTEM FOR THE TRANSMISSION OF COMPUTER DATA TO AN OUTPUT DEVICE

The invention is directed to a method, a computer program product and a system or, respectively, an apparatus for the transmission of data from a computer system to an output device, particularly print data to a printer device.

A printer device is connected to many computer systems. Different print data format have thereby established themselves dependent on the system environment. For example, the PCL and postscript print data formats are standard in the Windows® environment. Given these print data formats, the data (documents) to be output are sent to the printer completely packed.

In contrast to the PCL and postscript formats that have just been described, print data can be separated into resources (scripts, forms, layouts, etc.) and variable data given other print data formats such as AFPDS (Advanced Function Presentation Data Stream) or IPDS (Intelligent Print Data Stream). The resources and the variable data are thereby merged in relatively late processing steps, i.e. only shortly before the printing. EP-A-0 131 966 discloses a corresponding printing system that receives a print job containing a plurality of documents from a host computer, whereby form data and variable data of a document are transmitted separately from one another. The form data, as well as data that occur multiply in a plurality of documents, are transmitted only once per print job, are stored in the printing system and are employed for printing a plurality of individual documents. The variable data, in contrast, are transmitted once per document.

Computers in a typical office environment (office domain) are frequently equipped with a Windows® operating system or similar operating system such as Linux® and Macintosh®. The PCL and postscript print data languages are standard in this office domain, i.e. given relatively low printing outputs up to approximately 40 pages per minute. In contrast, the AFPDS (Advanced Function Presentation Data Stream) and IPDS (Intelligent Printer Data Stream) data streams are standard in the high-performance printing field above 40 pages per minute.

Specific properties of the data output given windows-like operating systems as well as their operating systems such as, for example, the program Word for Windows®, the print languages such as PCL® or PostScript® allocated to them, become especially clear given printout of series documents, i.e. given the use of what is referred to as the mail-merge function. Such series documents are usually composed of individual documents that are composed of static, repeatedly reoccurring data (master, master document) and modifiable/variable data that are introduced into the master or, respectively, into the master document. With reference to the overall document, the variable data in a series document generally only amount to a fraction of the data quantity. The static part thus defines the necessary performance parameters of the system (RAM memory, disk storage, transmission capacity, etc.) in order to achieve a performance-suited printing speed.

Relatively small documents, i.e. individual pages, reports or books (up to approximately 300 pages) can be printed without further ado as packed documents. In contrast thereto, printing series documents in this way can lead to substantial time delays because the static data must be continuously transmitted from the generating computer system to the printer device, i.e. with every individual document.

Another problem given this way of printing is the design of documents with auxiliary information, for example linking images in or the introduction of forms into the documents. These auxiliary information are frequently not produced together with the document to be printed out but often derive from a different data source and are sometimes designed in an involved way in order to be able to be employed for a plurality of applications.

When printing out documents from user programs, for example from Word for Windows®, the auxiliary information are previously generally inserted into the document via an editor (for example, via the Word program). When this document is to be multiply output, particularly as a series document, then there is again the problem that the auxiliary information must be processed given each document, i.e. the data stream repeatedly contains the same ii and, thus, redundancy.

There is the same problem when individual forms, banner, header or trailer pages are attached to the document via a Windows printer driver.

An object of embodiments of the invention is to achieve a high throughput of documents given the output of document data from a computer to an output device.

Accordingly, there is provided a method for the output of data from a computer system to an output device, whereby (a) a master document having a variable data area and having a static data area is generated; (b) the variable data area is marked by a marking selectable by means of a selection window; (c) variable data are inserted into the variable data area in the computer system, as a result whereof a serial data stream with individual documents arises, said individual documents respectively containing both variable data as well as static data; (d) the variable data from the serial data stream are separated from the static data on the basis of the marking in the computer system; (e) the variable data separated from the static data from the first individual document are transmitted to the output device; (f) the static data of the first individual document are stored in the output device; (g) the static data of the following individual documents are not transmitted to the output device; and (h) the variable data are in turn joined with the stored, static data individual document by individual document in the output device.

There is also provided a computer readable medium having computer readable instruction stored thereon, that when executed by one or more computers implement the method described above.

There is also provided a system for the implementation of a method described above that comprises at least one computer.

The invention is based on the perception that a

serial data stream wherein static and variable data are inseparably connected to one another individual document by individual document arises given the output of series documents from Windows® or windows-like applications. The invention has recognized that a separation of these data into static and variable parts - even when these parts were already previously joined (packed) - allows the output on a printer device or some other output device to become significantly more efficient and high-performance.

According to a first embodiment of the invention, a separation of static and variable data ensues during the output of the data. For example, the variable data are filtered out of the serial data stream and further-processed differently than the static data. Whereas the variable data are completely transmitted from the computer to the output device, the static data - insofar as they repeat - are transmitted only once from the computer to the output device. As a result thereof, the quantity of data can be considerably reduced and, thus, the performance capability of the transmission can be significantly enhanced.

In a system wherein variable and static data are initially connected to one another for each individual document, a renewed, individual document by individual document separation of the variable data from the static data ensues in an intermediate step according to this first aspect of the invention before the data are transmitted to the output device, particularly to a printer device. In order to be able to implement this separation, the variable data are previously provided with a suitable marking; in particular, they are identified chromatically. In particular, the marking already ensues in the production of the document by marking the wildcards at which the variable

data are inserted. Preferably, the marking is no longer visible given the output of the data in the printer device.

The inventive separation between variable data and static data can already ensue in the source format of the application (for example, directly in the editor) or in

an operation following the application, for example during the course of the conversion of a Windows document into the EMF format or within the EMF format. In particular, the separation can ensue in a following filtering for the EMF conversion and, in particular, given the conversion of the data stream into a print language such as PCL or postscript. The invention makes an intelligent printer driver possible with which extensive print data streams can be transmitted to a printer device at high speed.

According to the first aspect of the invention, a limitation of the static data particularly ensues, i.e. a particular about the scope that the data occupy. As a result of this limitation, a document-exact and location-exact allocation (referencing with corresponding referencing data) of the variable data to the static data can ensue in a simple way in the following filter operation.

According to a second aspect of the invention, a supplemental document, for example one or more PCL macros, is optionally attached to one or more arbitrary areas (pages) of an existing but arbitrary document at the beginning of the output, particularly of a print out. Particularly given series documents, it is thereby adequate to make the allocation only once for a master document.

The allocation can then be valid, i.e. activated, for all individual documents. The allocation ensues by means of a logical linking of the two documents, particularly by means of a referencing wherein corresponding referencing data are formed. Additional parameters can thereby preferably be recited, for example the position of the second document within the first document. The second document is particularly characterized in that it has a given, non-variable data content (for example, graphics areas or non-variable text constituents). Within a typical Windows® system environment, the referencing is thereby particularly controlled via data that are input via a user interface. The referencing then ensues within a converter unit that converts a windows-specific Enhanced Metafile data stream (EMF data stream) into a print data stream such as, for example, PCL® or PostScript®.

In advantageous embodiments of the second aspect of the invention, the page area of the master document wherein the second document is joined to the

master document is indicated. The incorporation of the second document can ensue in various modes, for example as overlay or as watermark.

Packed datafiles multiply containing one and the same static data occur not only in PCL and postscript printing data streams but also definitely occur in AFPDS, IPDS and in LCDS data streams. Further, documents/data streams that are generated in a first printer language (for example, in PostScript) can also have data (documents, macro datafiles) that have been generated in other languages (for example, IPDS overlays, IPDS page segments or PCL macros) embedded in them. The invention is therefore suited for the optimum output of any data streams.

What is particularly understood as output device for the invention is a printer device. Nonetheless, the invention can also be employed in conjunction with other output devices by themselves or in combination with a printer device. For example, a print server, a CD-ROM writer device or a print data archive can also serve as output device, as disclosed in the PCT Patent Application PCT/EP98/05460.

Exemplary embodiments of the invention are explained in greater detail below on the basis of some Figures.

Shown are:

Figure 1 a printing system

Figure 2 a data stream in the printing system

Figure 3 a selection mask

Figure 4 a second selection mask

Figure 5 a series document

Figure 6 a second data stream

Figure 7 a further selection mask

Figure 8 a standard Windows system environment and

Figure 9 a modified Windows system environment.

Figure 1 shows a personal computer 1 with a connected printer 7. In a known way, the personal computer 1 contains a central processor unit (CPU) 2, a monitor 3 connected thereto, as well as an input device 4 (keyboard, mouse, touch

screen or the like), a main memory 5 and a hard disk 6. The computer 1 is connected to the printer 7 via the interface 9. The personal computer 1 is operated with an operating system, for example with Microsoft Windows 95® or Windows NT. Various application programs in turn run under this operating system, for example the application 10 Winword 97® from the Microsoft Office 97® package. How variable and static data of a series letter from the Winword application are printed is explained with reference to the example of Figure 2. Variable data 11 and static data 12 that are stored in the main memory 5 and/or on the hard disk 6 are thereby incorporated into the application. For this purpose, the user can produce a master document in the Winword program 10, static data areas and variable data areas being provided in said master document. The variable data areas are intended to be filled with variable data that are stored in a separate datafile (Word document, data bank, Excel document, etc.). Further details about this series letter production are set forth, for example, on pages 75 through 93 in the book by Rainer-Walter Schwabe, Word 97: leicht, klar, sofort, Markt- und Technik-Verlag, Haar (1997), ISBN 3-8272-5267-3.

Wildcards for the variable data are thereby created in the variable data areas of the Winword master document, for example with the particular <<name>>. When the series letter is called, the variable datum in these wildcards is then taken record-by-record from the field "Name" of the datafile that contains the variable data.

In order to print out the series letter, i.e. the individual documents with the respectively inserted, variable data, the wildcards for the variable data, for example <<name>>, are distinguished from the static data of the master document with a suitable marking. This occurs, for example, by formatting the wildcards in a specific color. The color should thereby be selected such that the printer is not in the position of printing out data in this color. A color is thus employed that lies outside the reproducible color spectrum of the printer. This type of marking can assure that the visual appearance of the document output on the printer 7 is not affected by the marking. The marking can ensue in a known way within the application 10, i.e. with the command "Format|Character" of the application Word for Windows in the present example.

Before a document is printed out from the application, two particulars that control the rest of the printout must be produced. First, what properties the marking of the wildcards for variable data has must be specified. In the present example, this is the marking of the wildcards with the color red (also see Figure 4). Second, the scope of the mater document must be indicated. The individual documents of the series letter can be distinguished or, respectively, separated from one another with this particular in the later filter operation.

For printing out the series letter, the wildcards in the variable data areas of the master document in replaced record-by-record in the application 10 by the corresponding variable data 11 of the variable data memory, and a data stream or, respectively, a datafile of the entire document, i.e. of the static and of the variable data, is generated in the Enhanced Metafile Format (EMF) 13. In this EMF spooling, which is implemented via a Windows printer driver given output of the series letter, each individual document of the series letter or, respectively, of the series document is respectively newly constructed from the master document. In order to avoid a redundant data repetition of the static data, the static part 16 and the variable part 15 of the individual documents from the series letter data stream are respectively separated from one another in the filter operation 14. The variable data are thereby recognized on the basis of their marking that was previously undertaken as described above. In addition to this marking information, the scope of the master document, for example the number of pages it fills, is also needed in this filter operation. As a result thereof, the individual documents can be distinguished from one another in the filter event.

In the filter event 14, the static data are separated from the variable data on the basis of the marking of the variable data and on the basis of the indicated limitation (number of pages of the master document). The static data are transmitted to the printer device 7 and are stored as form or macro thereat in the main memory 8. stored as form or macro in the main memory 8. [sic] The capacity of the main memory is thereby fashioned of such a size that a plurality of documents (macros, forms) as well as their appertaining referencing data can be simultaneously stored therein. The main memory, as RAM (Random Access Memory), can thereby

typically amount, for example, to 4, 8, 64, 128, 512 or even more megabytes (MB) or can also be fashioned as hard disk with memory capacities of a number of gigabytes that are typical thereof. A combination of the two memory types (RAM, hard disk) can also be meaningful, whereby documents (macros, etc.) of a print job that are called more often are stored in the RAM and documents that are called less often, for example documents (macros) of the print job that are called only once, are stored on the hard disk.

The variable data, in contrast, together with all needed characteristics (for example, indications of position on the individual document, color particulars, font particulars) are separately transmitted, likewise to the printer device. The transmission of the variable data and of the static data from the computer system 1 to the printer device 7 can ensue via the same data line, whereby, however, a logical discrimination (separability) between the data must be retained.

Beginning with the second individual document, only the filtered, variable data are then transmitted to the printer device 7, as a result whereof a considerable reduction of the data stream between computer system 1 and printer system 7 is achieved.

Within the printer device 7, the received, variable data are mixed again with the static data and printed in common on a recording medium (paper, labels, films, etc.).

In the second and in all further individual documents, the static data 16 are discarded in the computer 1 or, respectively, within the PCL converter 18 and are not transmitted to the printer device 7. In contrast, the variable data 15 together with their characteristic particulars are transmitted to the printer device individual document by individual document. In the printer device 7, these variable data 15 are then merged by an OR-operation with the static data stored in the memory 8.

The filter event 14 precedes the conversion of the data into a PCL data stream. However, it occurs within the PCL converter 18.

A postscript converter or some arbitrary other converter that is standard in the respective system environment can also be employed instead of a PCL converter. The filter event 14 can also be directly applied onto the EMF intermediate datafile or,

respectively, the EMF intermediate data stream 13 or can be applied to the source text of the editor (for example, in Word). The filtered, variable data 15 or, respectively, static data 16 then already enter in the PCL converter.

Figure 3 shows a selection window that is selected before the printing event from the application ensues into the EMF intermediate datafile (event 13). The input window 20 contains a first selection window 21 in which two print modes can be selected. In the first print mode (standard), print data from the windows application are printed out in a standard way, i.e. the filter procedure (event 14) does not occur. Series letters are then transmitted to the printer device 7 individual document by individual document. The above-described filter procedure (event 14) isn activated by selecting the option "extended mail processing" in the mode window 21.

The scope of the master document can be specified in the selection window 22. As a result thereof, the separation of the individual document pages is enabled in the filter event.

The color property with which the wildcards of variable data were marked in the master document is indicated in the selection window 23. Figure [...] shows various selection fields (black/gray, red, green, blue) for these markings.

Whether the dynamic texts are to be printed differently in the printer device 7, for example in a highlight color, can be optionally indicated in the selection window. This assumes that the printer device is in the position to print in two colors, whereby the standard texts are printed in the first color and the dynamic texts are printed in the second color. The first color, for example, is thereby black and the second color is blue (highlight color) or vice versa.

In a further embodiment of the invention, the dynamic texts can be marked either [sic] in a first color that cannot be presented by the printer. The variable data marked in this way are printed out in the same color as the static data. Other variable data, in contrast, are marked in a second color and printed out in the second ink (highlight color) of the printer device.

Figure 5 shows a master document 25. It is composed of static data 26 and of the three variable wildcards 27, 28 and 29 (title, name, competency). The

length of the master document amounts to one page. Variable data are stored in the datafile 30 in the fields name, competency and title. The master document 25 and the variable data 30 are combined into the series document 31, whereby the static text part 33 that corresponds to the static data 16 (Figure 2) is extracted from the first series document 31. These data are employed for generating the second individual document 33 (see Figure 2).

Figure 6 shows how an auxiliary information can be linked into an existing document. As indicated in the preceding examples or, respectively, Figures, let the application here also be the program Winword. A macro 36 that contains an external data source 37 is linked into the Winword document 35. For example, the macro 36a contains an image datafile 37a. The macro 36b contains a line diagram 37b and the macro 37c contains a bar diagram 37c. In the event 38, the macros 36a, 36b and 36c can be optionally selected either individually or a plurality of them at once for linking into the Winword document 35. On which pages and at which position within the individual pages of the Winword document 35 the macro or the respective macros are to be placed is also indicated in this event. A print data stream 39 is generated therefrom, whereby the individual pages 39a, 39b and 39c are provided with the respective reference index macro data M1, M2, M3.

These information (reference index data M1, M2, M3) are converted into the PCL language and are sent to the printer device 7. Simultaneously, the macro information (particularly graphics data) are converted (insofar as they are not already in PCL format) and are transmitted into the printer device 7 separated from the series letter information, i.e. separated from the series print data stream in terms of time and/or in data-oriented fashion, and are deposited in the main memory 8 thereat. A plurality of and, in particular, all complete macro data (graphics information, etc.) required for the print job are thereby stored in the main memory 8. Within the printer device 7, the series letter data (i.e. the series print data stream) are then reunited with the corresponding, complete macro data upon employment of the reference index data M1, M2, M3, i.e. the page 39a is output upon employment of the reference index M1 with the macro 36a (i.e. with the complete print data of the macro 36a), the page 39b is output upon employment of the reference index M2 with the complete data of the

macro 36b, and the page 39c is output upon employment of the reference index M3 with the data of the macro 36c. The referencing, i.e. the logical linking between specific document pages (39a, 39b, 39c) with the allocated macros 36a, 36b, 36c, occurs in the unit 38 in that the reference indices or, respectively, reference index data M1, M2 and M3 are formed.

In the example of Figure 6, the individual document comprises the three pages 39a, 39b, 39c. When the document 35 is a series document, then it suffices to make the allocation for an individual document (master document), so that this allocation is valid for all individual documents. The individual documents can thereby be individual pages or -- as shown in Figure 6 -- can comprise a plurality of pages. The advantage of this procedure described for the macro linking is once again that the reoccurring information -- macros here -- need be transmitted only once from the computer 1 to the printer device 7 and can be linked from the main memory in the computer to individual documents as often as desired. Here, too, the quantity of data to be transmitted between computer 1 and printer device 7 is minimal because macros in the individual documents are not transmitted completely but only by indicating their reference index 2 (M1, M2, M3).

Figure 7 shows a selection window that is called in the referencing unit 38 in the computer 1. Presets for macros can be undertaken in the window 41, i.e. standardized macro collections and/or links to specific document pages can be deposited. Additional, new macros can be selected for a macro preset via the selection key 42. The position of the macro on specific document pages can be defined in the selection field, for example on all pages, on even-numbered or odd-numbered pages or on specific page numbers. The placement type as overlay (complete superimposition) or watermark (macro information only in the background) within the document can be selected with the selection field 44.

Figure 8 shows a structure diagram according to which a print event normally sequences within a Windows® operating system running on a computer. What is referred to as a "User-Mode Client" (GDI32.dll) 46 is thereby called proceeding from an application 45, this "User-Mode Client" 46 driving the display device (Graphic Device Interface, GDI). Various printer drivers can thereby be called

and the settings of the appertaining printer supported by the respective driver can be set job-specifically. After these settings have been carried out and the print job has been enabled (pressing "OK"), a standard check is carried out under Windows to see whether the output format that is thereby generated corresponds to the EMF standard (EMF stands for Enhanced Metafile Format). When this is the case, the print data stream is supplied as EMF data stream to a print processor 49 situated in the spooler 50, being supplied thereto via a spool datafile 48. The data are supplied therefrom to a port monitor 51 and are supplied to the destination printer device 52 as what is referred to as a RAW data stream, i.e. as data stream in a printer language such as, for example, PCL that is matched to the destination printer device. The port monitor 51 thereby controls the output of the data to the output channel (LPT1, LPT2) of the computer allocated to the destination printer device.

When the query 47, however, yields that the document generated in the application 45 is not in EMF format, then what is referred to as a kernel mode 53 is activated wherein a GDI rendering engine (GRE), i.e. the program element "win32k.sys" 54 collaborates with the kernel mini control module 55. A RAW data stream is thereby generated that, as RAW spool datafile 56, is in turn supplied to the print processor 49 in the spooler 50. From there, the handover via the port monitor 51 to the output device 52 in turn ensues as RAW data stream.

How the aspects of the invention described in conjunction with Figures 6 and 7 are implemented within a Windows environment becomes clear in Figure 9. As a result of the invention, the referencing of various documents (maser document, macro, overlay) in such a Windows® environment is controlled via data that are input via a user interface or, respectively, via an input module 59. The referencing ensues within a converter unit 58 that converts a windows-specific Enhanced Metafile data stream (EMF data stream) into a PCL print data stream (RAW data stream of the destination printer).

The processing of a data output (printing) proceeding from the application 45 (for example, Word) initially ensues according to Figure 9 exactly as in the standard Windows® environment according to Figure 8. However, an inventively modified driver is employed as printer driver in Figure 9, this generating a data stream

in the EMF format, as a result whereof a spool file 48 is directly generated under Windows or, respectively, via the query 47, and the spool file is supplied to an inventively adapted print processor 56 in the Windows spooler 50. The kernel mode 43 or, respectively, the modules GRE 54 and kernel mini 55 called therein are not supported by the driver, this being indicated by the cross 57 in Figure 9.

Another inventive adaptation is comprised in the print processor 56 that is located in the spooler 50. In contrast to the standard Windows environment of Figure 8, this "Enhanced Print Environment (EPE) Print Processor" 56 does not forward the EMF data directly to the port monitor 51 but calls the converter unit 58, wherein the EMF data stream is converted into a PCL print data stream. The conversion is thereby controlled by the parameters that were previously input via the input module 59 (OPS PCL user interface). Among other things, the input module 59 effects the display of the macro window 40 shown in Figure 7 for this purpose. The output can also ensue into various channels via settings that are either controlled via the input module 59 or, too, directly via the printer driver, which collaborates with the GDI user mode client 46. The output of these PCL-RAW print data can thereby ensue either into an output datafile (channel 1) that, for example, is stored on hard disk or directly to an SCSI-capable printer (channel 2) or back again into the spooler 50 to the port monitor 51 and from the latter via a standard interface (channel 3) to a destination printer device 52.

Exemplary embodiments of the invention have been described. It is thereby clear that, instead of the described Winword application program, other applications, presentation programs, table calculation programs, image processing programs beside text processing programs or other programs standard in the computer field can effectively use the invention. Instead of the PCL printer language that has been described, of course, other printer languages can also be employed, for example PostScript, AFPDS, IPDS, PDF or LCDS, particularly when a completely packed data stream is present that is not resolved into individual constituents such as scripts, forms, etc.

The invention has created an intelligent printer driver or, respectively, an intelligent pre-stage for printing with which the data stream between an application

computer and a printer device can be significantly reduced. As a result thereof, a higher document throughput can be achieved in the output. The invention can be applied both in a single-location system wherein a computer is connected to a printer as well as in a network system wherein a plurality of computers send print data to one or more printer devices. Instead of transmitting the data directly to the printer device, they can be sent from the printing system in which the data are generated to the printer via a print server. Such a system is described, for example, in Druckerbuch, edited by Dr. Gerd Goldmann, Océ Printing Systems GmbH, Poing, Edition 3c (May 1998), ISBN 3-00-001019-X, pages 12-1 through 12-8, having a corresponding English-language description in The World of Printers, edited by Dr. Gerd Goldmann, Océ Printing Systems GmbH, Edition 3a (November 1998), ISBN 3-00-001081-5, pages 12-1 through 12-8.

When the inventive separation between variable data and static data already ensues in the source format of the application (for example, directly in the editor), then it is possible to further automate or, respectively, accelerate the output of data streams in that the static part is further-processed separately from the variable part, for example by conversion of the static part into a macro datafile (for example, in PCL), transmission to the output device, storing thereat and loading the macro in the print generator as needed. The filtering of the data stream can be eliminated by means of this further automation step.

The invention is particularly suited for use in Windows systems and in windows-like systems such as Linux or Macintosh operating systems that at least partly comprise identical or similar systems components as Windows.

The invention can appear in the greatest variety of embodiments, for example as computer datafile, as program module, as program element, as program library or as collection of a plurality of individual datafiles. This embodiment of the invention, which is particularly contained in computer program products such as a memory element, a diskette, a disk storage, a CD-ROM, a programmable electronic

module (ROM, PROM, EPROM), can also be spread via data networks, for example as datafile attachment of an e-mail as well as via the Internet.

List of Reference Characters

1	computer
2	CPU
3	monitor
4	keyboard
5	main memory
6	hard disk
7	printer
8	printer memory
9	interface
10	Winword® application
11	variable data
12	static data
13	enhanced metafile
14	filtering
15	filtered, variable data
16	filtered, static data
17	data printed out
18	PCL converter
19	bitmap memory
20	input window for print output
21	mode selection window
22	window for indicating the scope of the master document
23	window for indicating the marking property (color)
24	optimum window for indicating the print output color
25	master document
26	static data
27	title wildcard
28	name wildcard
29	competency wildcard
30	variable data

31 series letter data stream

32 first series letter document

33 static data of the first series letter document

34 second series letter document

35 Winword document

36 macro

37 external datafile

38 referencing unit

39 referenced page stream

40 macro window

41 macro preset field

42 selection field

43 macro position field

44 placement selection field

45 application program

46 user control module

47 query

48 EMF spool datafile

49 print processor in the spooler

50 spooler

51 port monitor

52 printer device

53 kernel mode

54 GRE control module

55 kernel mini control module

56 enhanced print environment processor

57 blocking of the kernel mode

58 converter unit

59 input module

60 output datafile

61 SCSI printer

29246-22

CLAIMS:

1. A method for the output of data from a computer system to an output device, whereby

(a) a master document having a variable data area and having a static data area is generated;

(b) the variable data area is marked by a marking selectable by means of a selection window;

(c) variable data are inserted into the variable data area in the computer system, as a result whereof a serial data stream with individual documents arises, said individual documents respectively containing both variable data as well as static data;

(d) the variable data from the serial data stream are separated from the static data on the basis of the marking in the computer system;

(e) the variable data separated from the static data from the first individual document are transmitted to the output device;

(f) the static data of the first individual document are stored in the output device;

(g) the static data of the following individual documents are not transmitted to the output device; and

(h) the variable data are in turn joined with the stored, static data individual document by individual document in the output device.

2. The method according to claim 1, whereby a printer device is employed as output device.

3. The method according to claim 1 or 2, whereby the marking of the variable data area ensues by means of a visually perceptible identification, particularly chromatically.

4. The method according to claims 2 and 3, whereby the marking ensues with a color that lies outside the printable color spectrum of the printer device.

5. The method according to any one of claims 1 to 4, whereby the scope of the master document is indicated.

6. The method according to any one of claims 1 to 5, whereby characteristic data, particularly position data, font data and/or color data, are transmitted to the output device with the variable data.

7. The method according to any one of claims 1 to 6, whereby the static data are stored in a macro datafile.

8. The method according to any one of claims 1 to 7, whereby the serial data stream is generated in one of the printer languages PCL, PostScript or LCDS.

9. The method according to any one of claims 1 to 8, whereby the master document or the second document is a document, particularly an overlay or a page segment, of the printer language IPDS.

10. The method according to any one of claims 1 to 9, whereby master document or the second document is a document of the print data language LCDS.

11. The method according to any one of claims 1 to 10, whereby the data output in a Windows system environment or in a windows-like system environment (Linux, Macintosh) is controlled via data that are input via a user interface.

12. The method according to any one of claims 1 to 11, whereby

(a) the master document is generated;

(b) at least one second document is logically linked with the master document; and

(c) the data of the second document are sent to the output device separated from the data of the master document.

13. The method according to claim 12, whereby the data of the second document are stored in the output device.

14. The method according to any one of claims 12 to 13, whereby the data of the master document are joined with the data of the second document for the output of an individual document.

15. The method according to any one of claims 12 to 14, whereby the logical linking ensues via a referencing.

16. The method according to any one of claims 12 to 15, whereby the area of the master document, particularly the page region, wherein the second document is to be linked with the master document can be specified.

17. The method according to any one of claims 12 to 16, whereby whether the second document is an overlay or a watermark document is indicated.

18. A computer readable medium having computer readable instruction stored thereon, that when executed by one or more computers implement the method of any one of claims 1 to 17.

29246-22

19. A system for the implementation of a method according to any one of claims 1 to 17 that comprises at least one computer.

FETHERSTONHAUGH & CO.
OTTAWA, CANADA
PATENT AGENTS

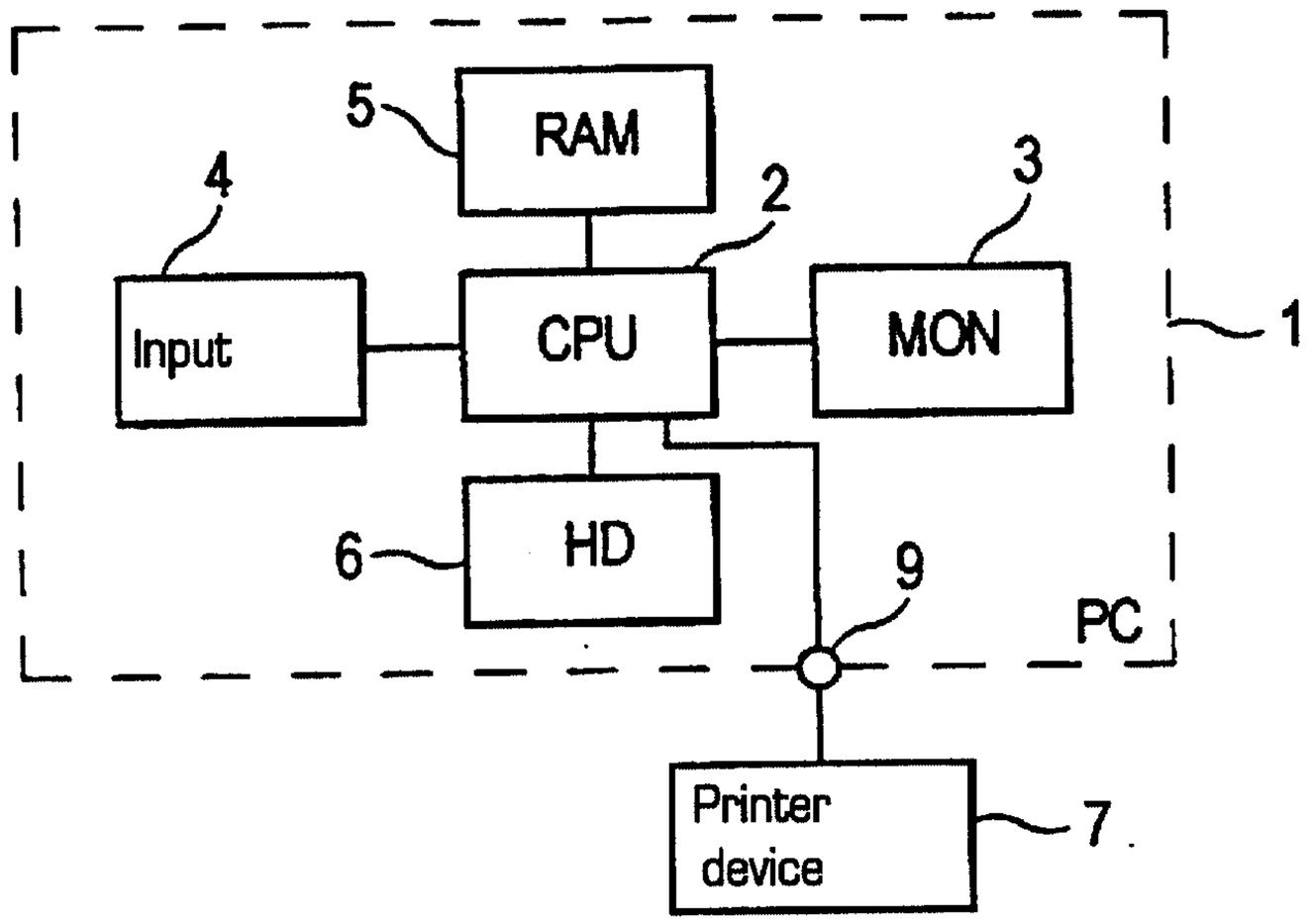

5
RAM
4
2
3
Input
CPU
MON
1
6
HD
9
PC
Printer
device
7

Fig.1

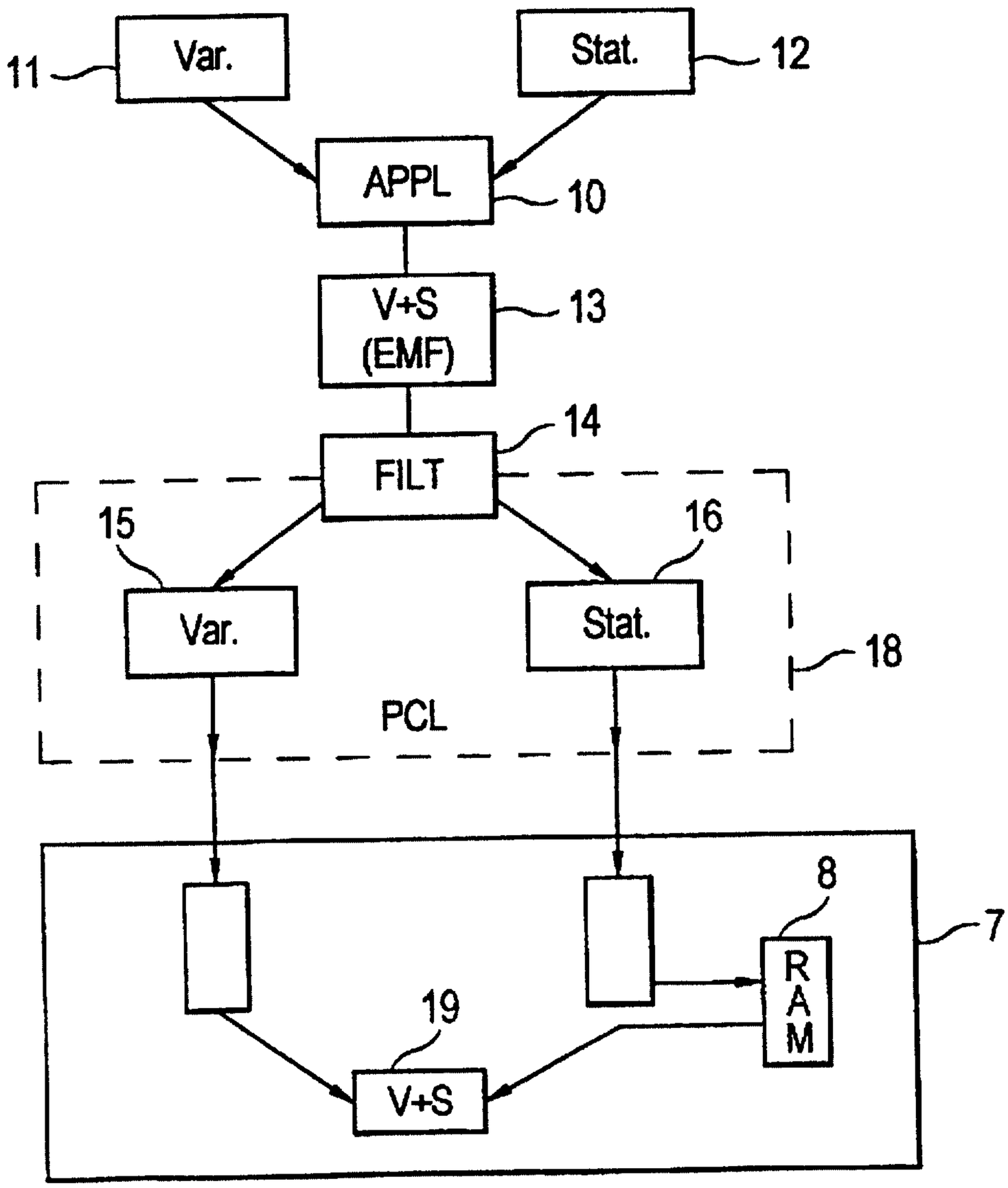

11
Var.
Stat.
12
APPL
10
V+S
(EMF)
13
14
FILT
15
16
Var.
Stat.
18
PCL
8
7
R
A
M
19
V+S

Fig.2

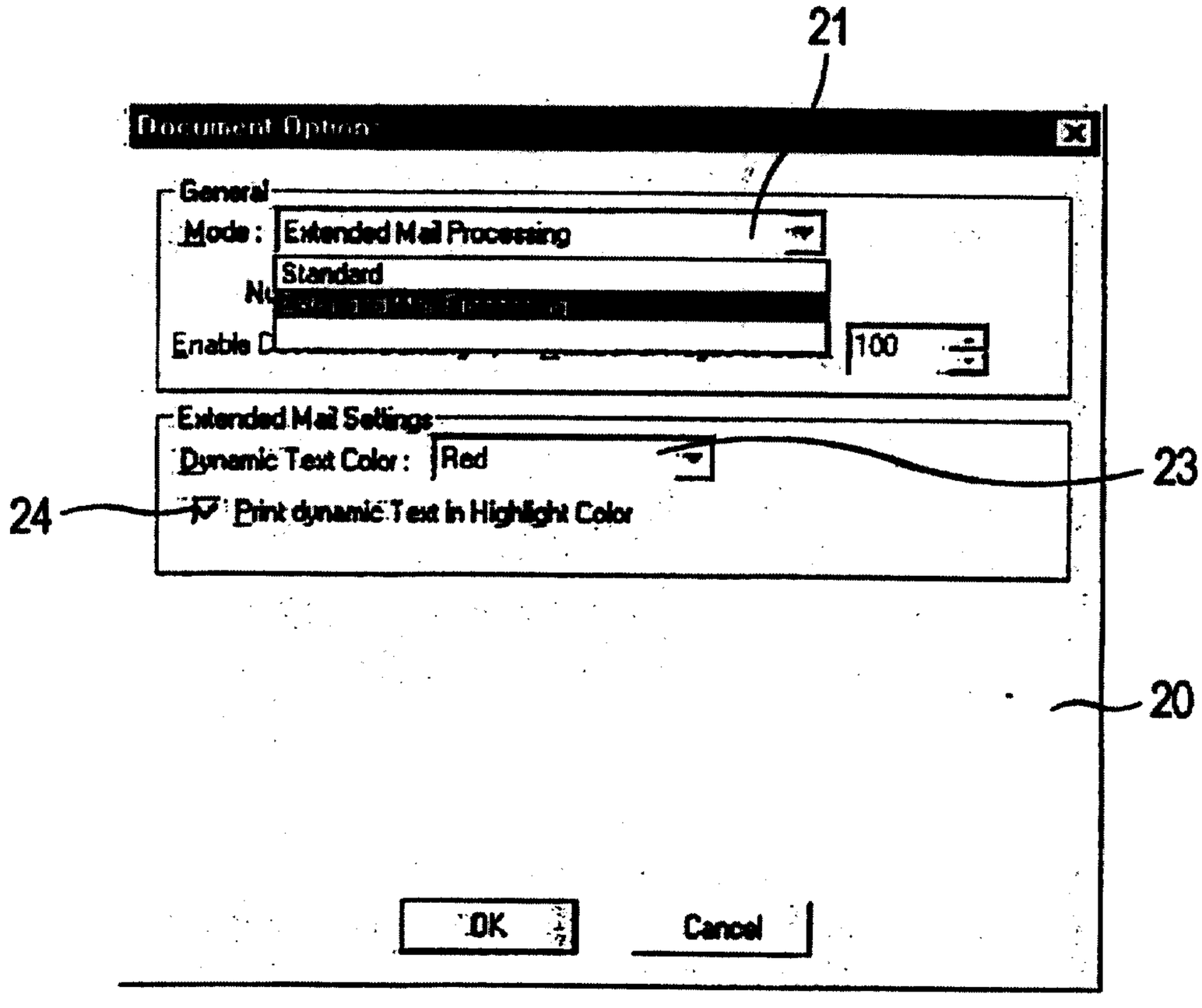

21
Document Options
General
Mode : Extended Mail Processing
Standard
Enable D
100
Extended Mail Settings
Dynamic Text Color : Red
23
24
Print dynamic Text in Highlight Color
20
OK
Cancel

Fig.3

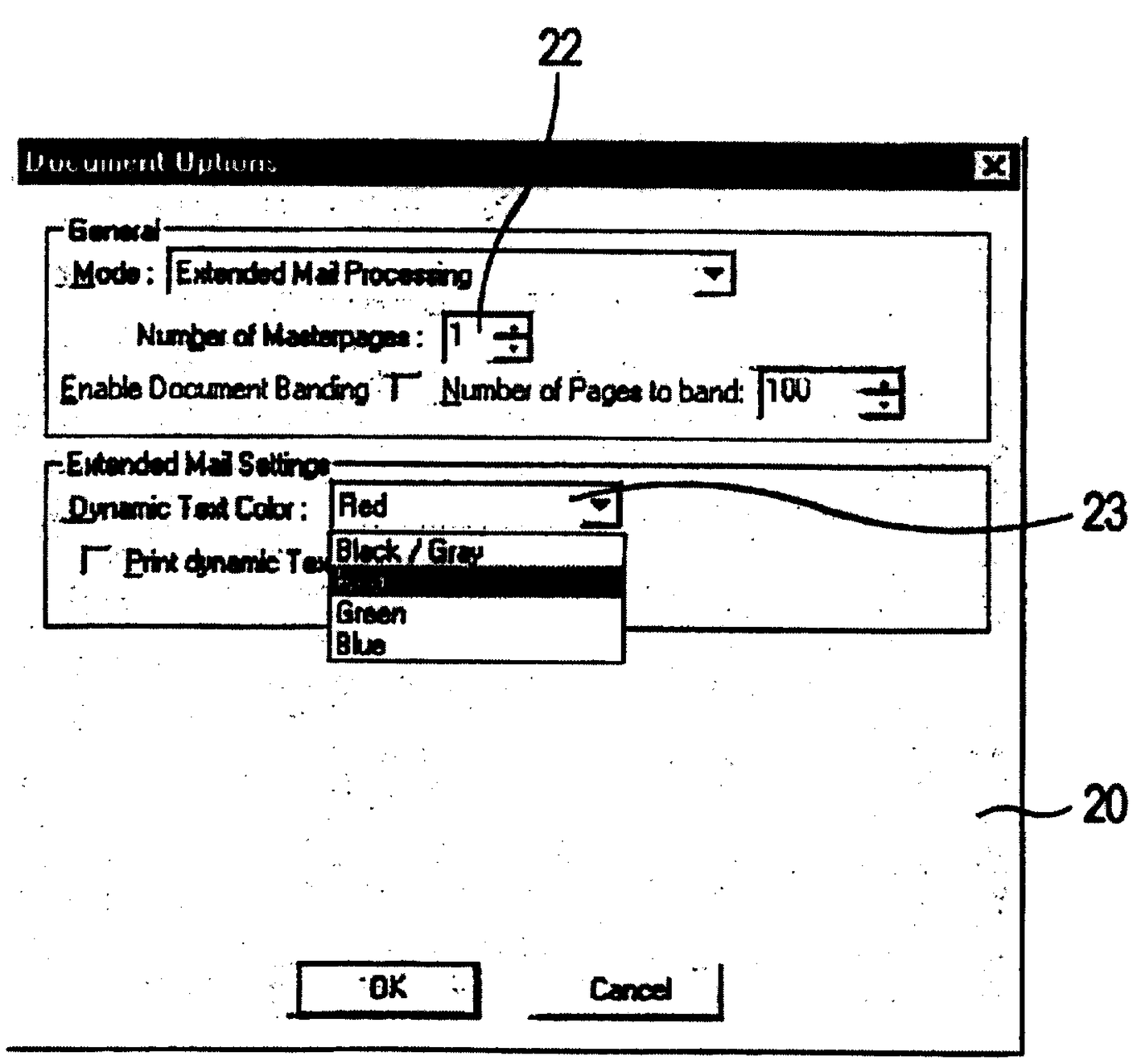

22
Document Options
General
Mode : Extended Mail Processing
Number of Masterpages : 1
Enable Document Banding
Number of Pages to band: 100
Extended Mail Settings
Dynamic Text Color : Red
23
Print dynamic Te
Black / Gray
Green
Blue
20
OK
Cancel

Fig.4

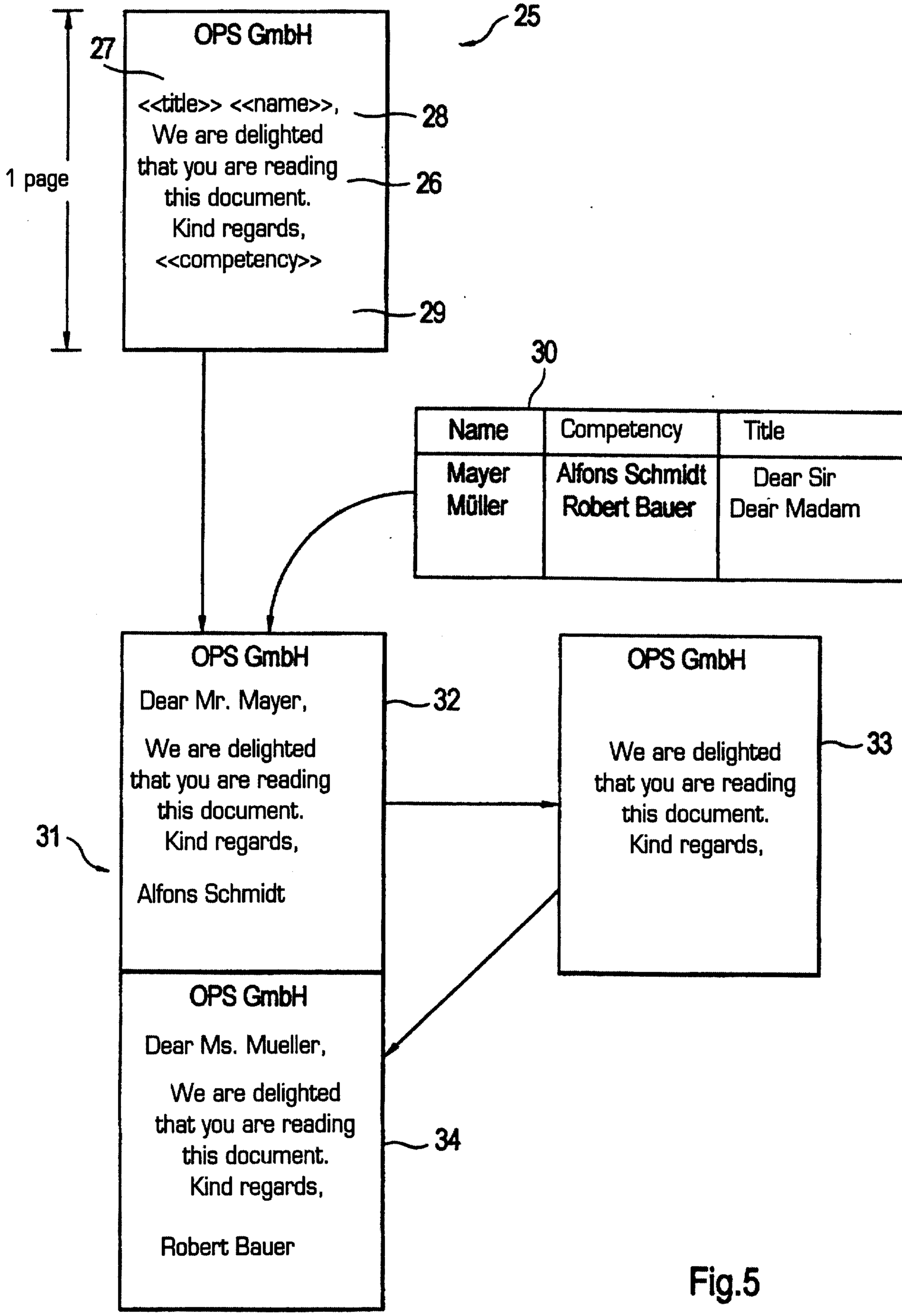

25
OPS GmbH
27
<<title>> <<name>>,
We are delighted
that you are reading
this document.
Kind regards,
<<competency>>
28
26
29
1 page
30
Name
Competency
Title
Mayer
Müller
Alfons Schmidt
Robert Bauer
Dear Sir
Dear Madam
OPS GmbH
Dear Mr. Mayer,
We are delighted
that you are reading
this document.
Kind regards,
Alfons Schmidt
32
31
OPS GmbH
We are delighted
that you are reading
this document.
Kind regards,
33
OPS GmbH
Dear Ms. Mueller,
We are delighted
that you are reading
this document.
Kind regards,
Robert Bauer
34

Fig.5

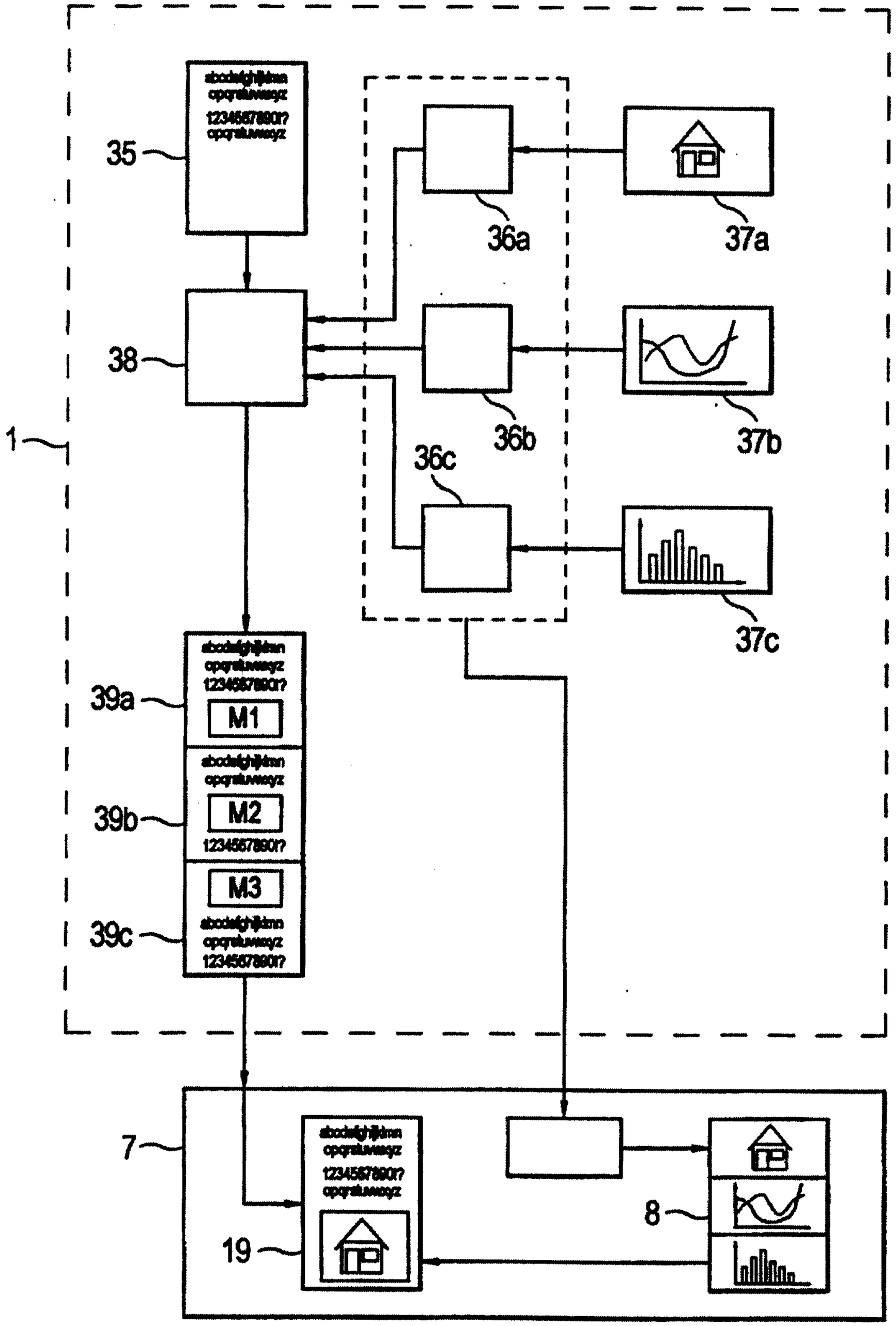

35
36a
37a
38
36b
37b
1
36c
37c
39a
M1
39b
M2
12345678901?
M3
39c
7
19
8

Fig.6

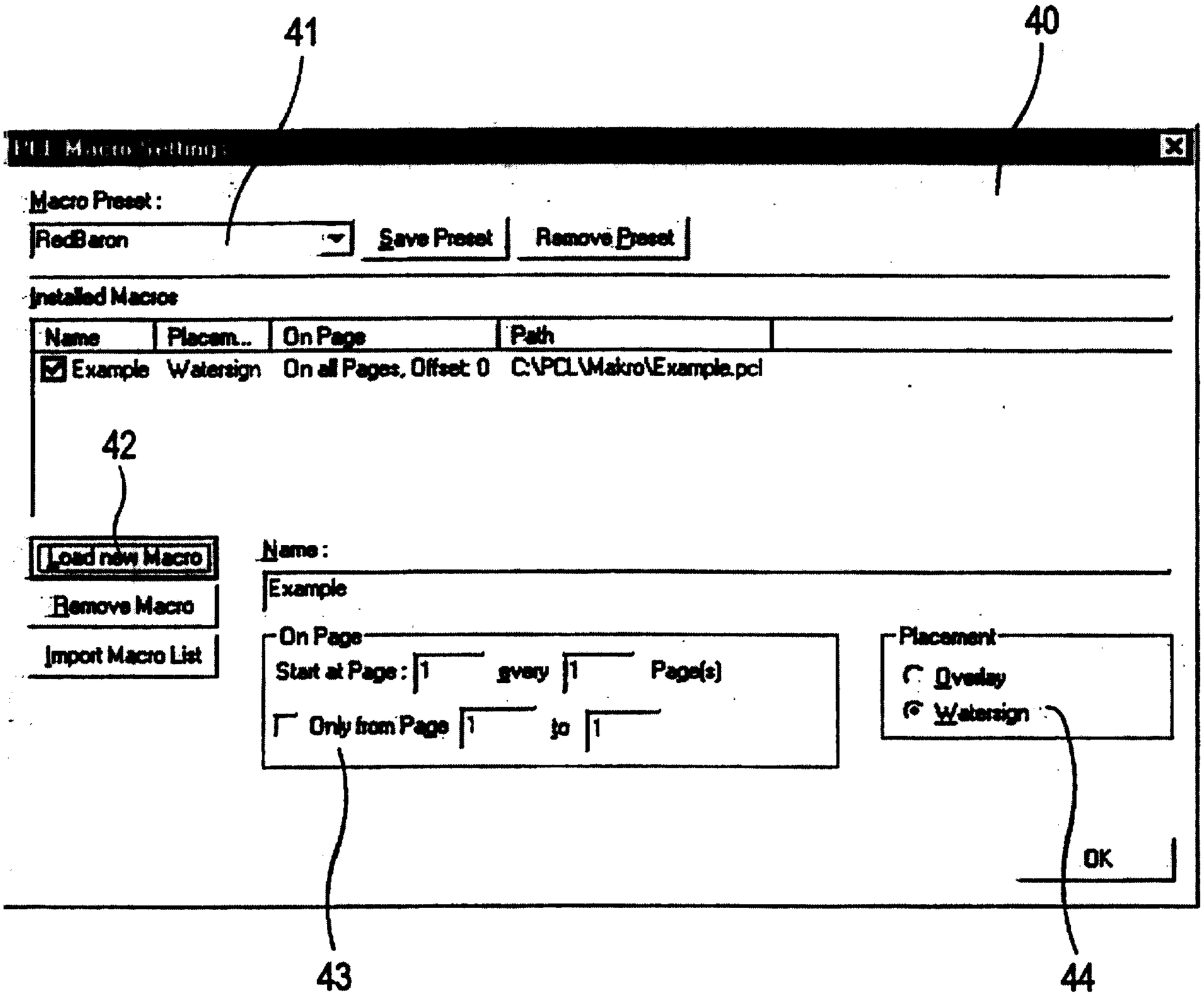

41
40
PCL Macro Setting
Macro Preset:
RedBaron
Save Preset
Remove Preset
Installed Macros
Name
Placem...
On Page
Path
Example
Watersign
On all Pages, Offset: 0
C:\PCL\Makro\Example.pcl
42
Load new Macro
Remove Macro
Import Macro List
Name:
Example
On Page
Start at Page: 1
every 1
Page(s)
Only from Page 1
to 1
Placement
Overlay
Watersign
OK
43
44

Fig.7

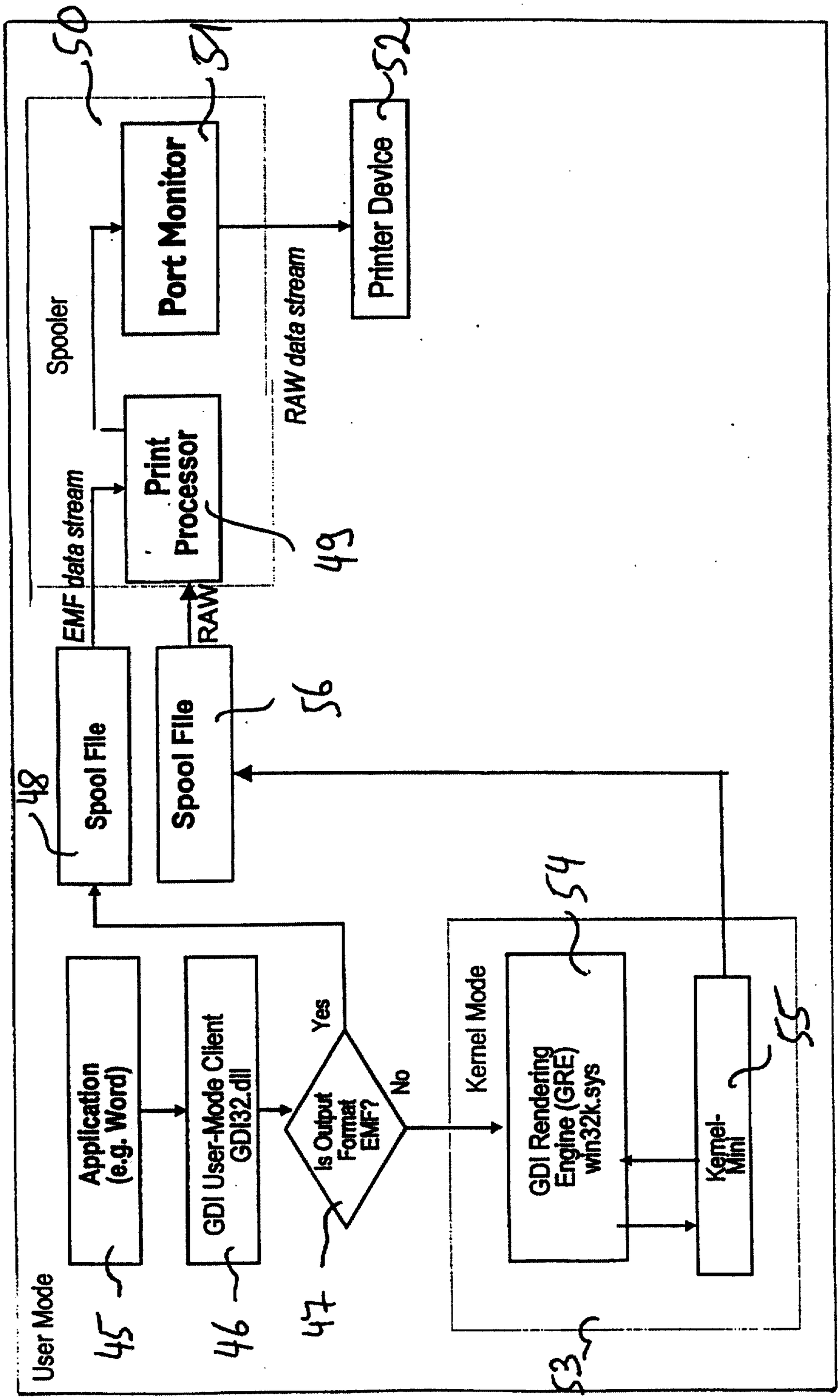

50
51
52
Port Monitor
Printer Device
Spooler
EMF data stream
Print Processor
RAW data stream
RAW
49
Spool File
Spool File
48
56
Application (e.g. Word)
GDI User-Mode Client GDI32.dll
Is Output Format EMF?
Yes
No
Kernel Mode
GDI Rendering Engine (GRE) win32k.sys
Kernel-Mini
54
55
User Mode
45
46
47
53

Fig. 8

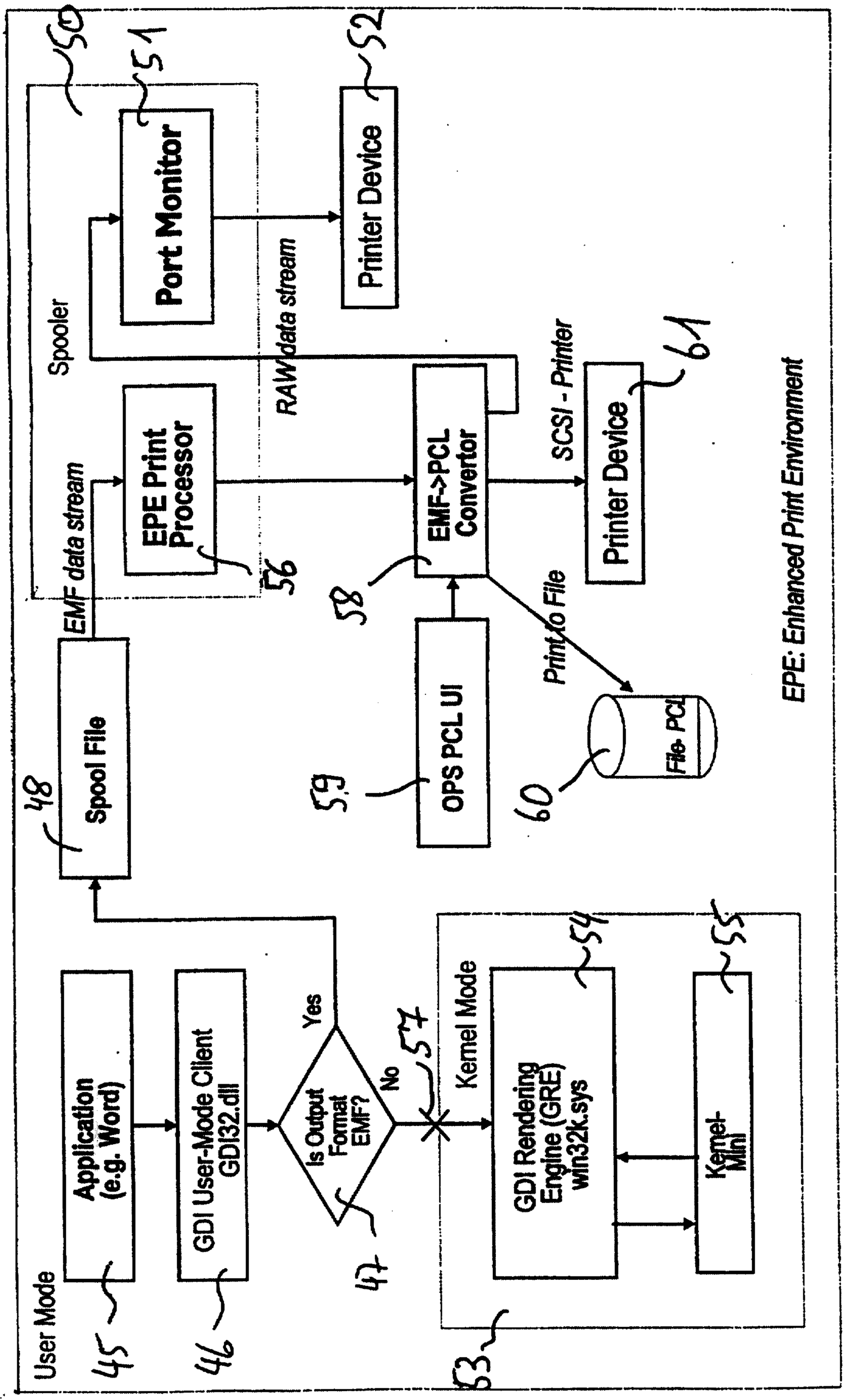
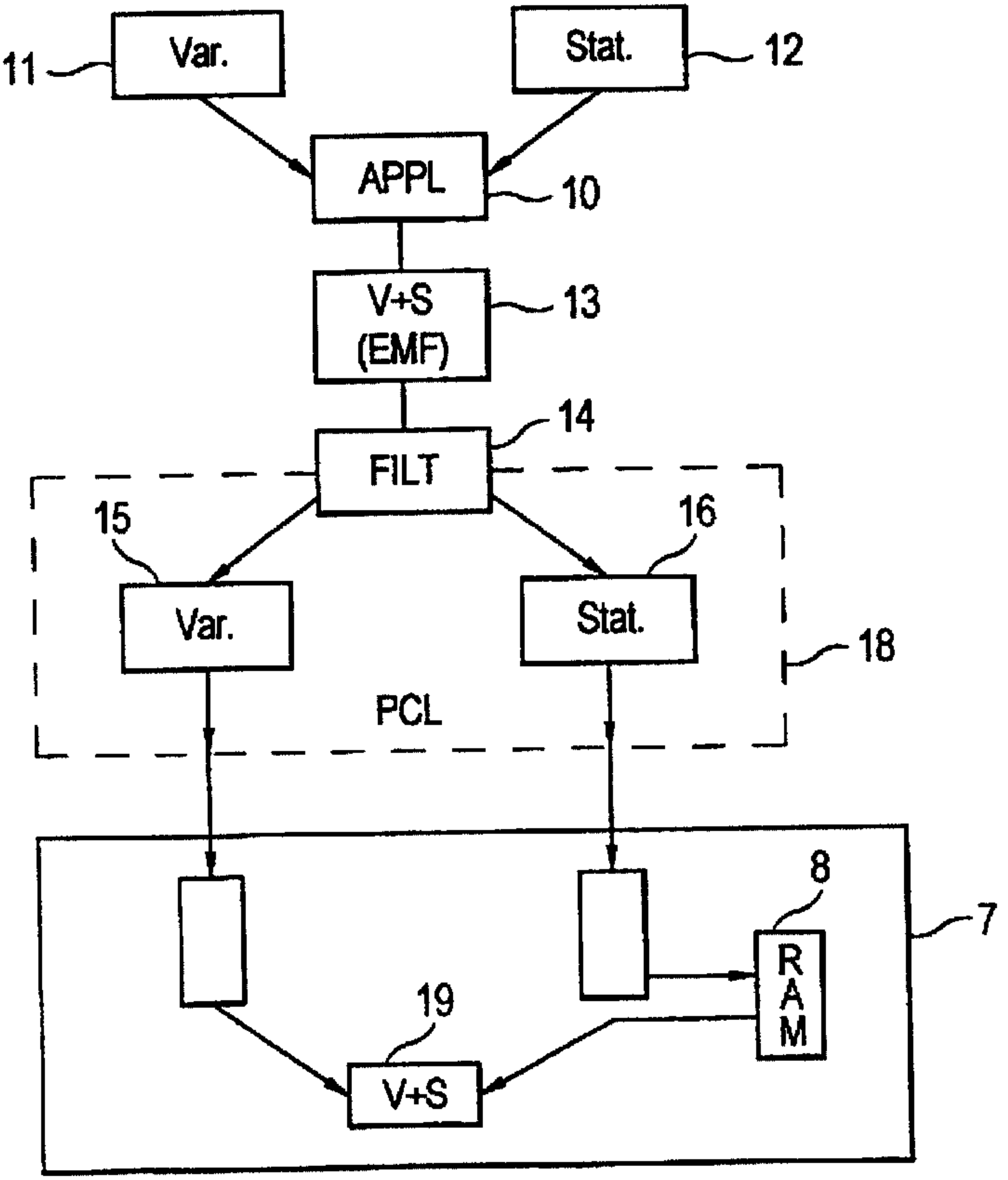

50
51
52
Spooler
Port Monitor
Printer Device
RAW data stream
EMF data stream
EPE Print Processor
56
58
EMF->PCL Convertor
SCSI - Printer
Printer Device
61
Print to File
59
OPS PCL UI
60
File- PCL
EPE: Enhanced Print Environment
48
Spool File
Application (e.g. Word)
GDI User-Mode Client GDI32.dll
Is Output Format EMF?
Yes
No
57
47
Kernel Mode
GDI Rendering Engine (GRE) win32k.sys
54
Kernel-Mini
55
User Mode
45
46
53

Fig. 9


11
Var.
Stat.
12
APPL
10
V+S
(EMF)
13
14
FILT
15
16
Var.
Stat.
18
PCL
8
7
R
A
M
19
V+S