**BESCHWERDEKAMMERN DES EUROPÄISCHEN PATENTAMTS**

**BOARDS OF APPEAL OF THE EUROPEAN PATENT OFFICE**

**CHAMBRES DE RECOURS DE L'OFFICE EUROPEEN DES BREVETS**

**Internal distribution code:**

(A) [ ] Publication in OJ
(B) [ ] To Chairmen and Members
(C) [X] To Chairmen
(D) [ ] No distribution

# DECISION
## of 23 February 2006

|  |  |
|---|---|
| **Case Number:** | T 0411/03 - 3.5.01 |
| **Application Number:** | 00110655.8 |
| **Publication Number:** | 1028377 |
| **IPC:** | G06F 9/46 |
| **Language of the proceedings:** | EN |

**Title of invention:**
Data transfer with expanded clipboard formats

**Applicant:**
MICROSOFT CORPORATION

**Opponent:**
-

**Headword:**
Clipboard formats II/MICROSOFT

**Relevant legal provisions:**
EPC Art. 52(1)(2)(3), 54, 56, 76(1) 83, 84, 123(2)
EPC R. 88
RPBA Art. 10b

**Keyword:**
"Late-filed request responding to objections - admitted (yes)"
"Divisional application extending beyond parent application (no-after correction)"
"Method of operating a computer - excluded as being a computer program (no)"
"Novelty/inventive step (yes)"

**Decisions cited:**
G 0003/89, T 1173/97, T 1194/97, T 0258/03

**Catchword:**
-

Europäisches Patentamt
European Patent Office
Office européen des brevets

Beschwerdekammern
Boards of Appeal
Chambres de recours

**Case Number:** T 0411/03 - 3.5.01

# DECISION
## of the Technical Board of Appeal 3.5.01
## of 23 February 2006

**Appellant:**

MICROSOFT CORPORATION
One Microsoft Way
Redmond, WA 98052-6399 (US)

**Representative:**

Grünecker, Kinkeldey,
Stockmair & Schwanhäusser
Anwaltssozietät
Maximilianstrasse 58
D-80538 München (DE)

**Decision under appeal:**

**Decision of the Examining Division of the European Patent Office posted 22 November 2002 refusing European application No. 00110655.8 pursuant to Article 97(1) EPC.**

**Composition of the Board:**

**Chairman:** S. Steinbrener
**Members:** K. Bumes
G. Weiss

## Summary of Facts and Submissions

I. The appeal lies from the Examining Division's decision to refuse divisional application No. 00 110 655.8 for lack of inventive step in view of

D1: EP-A-0 622 730 and

D2: "Method for multiple round data entry to an application with a multiple partition clipboard", IBM Technical Disclosure Bulletin, Vol. 36, No. 10, page 117, October 1993, ISSN 0018-8689, XP000412202.

II. The present application is one of six divisional applications originating from European patent application No. 95 119 547.8. The parent application and five of the divisional applications have proceeded to the appeal stage before the Board (T 424/03; T 411/03, T 425/03, T 467/03, T 468/03, T 469/03). The Board processed the parent and the five divisional applications in parallel to ensure a synoptical handling of the appeal cases.

III. The appellant requests that the decision under appeal be set aside and a patent be granted in the following version:

- claims 1 to 3 as submitted at oral proceedings before the Board on 23 February 2006;

- description: pages 1, 3, 5, 7 to 9 as originally filed;

- description: pages 2, 4, 6, 10 to 12 as submitted at the oral proceedings;

- drawings: Figures 1 to 6 as originally filed.

In addition, the appellant requested that claim 5 of the divisional application as filed be considered corrected, pursuant to Rule 88 EPC, to read "The method of one of claims 1 to 3 [...]" instead of "The method of one of claims 1 to 4 [...]".

IV. Claim 1 reads:

"1. A method in a computer system (10) having a clipboard for performing data transfer of data in a clipboard format, said method comprising the steps of:

providing several clipboard formats including a text clipboard format, a file contents clipboard format and a file group descriptor clipboard format to hold a file group descriptor structure,

selecting data that is not a file for a data transfer operation,

using the file contents clipboard format to hold said data by

converting said selected data into converted data of said file contents clipboard format and storing the converted data as a data object,

using the file group descriptor structure of the file group descriptor clipboard format to hold a file descriptor holding descriptive information about the data that is to be encapsulated into a file during the data transfer operation,

completing the data transfer by providing a handle to said data object, using said handle to paste said data of said data object to a data sink, using said descriptive information to enable the computer system to create a file at the data sink and encapsulating said data object into said file,

wherein the file descriptor includes a dwFlags field indicating which fields hold legal data, a clsid field holding a class ID of the data object that encapsulates the data held in the file contents clipboard format, a sizel field holding a value specifying the size of the data held in the file contents clipboard format, a pointl field holding a pointer to the object that holds said data, a dwFileAttributes field holding a double word of attributes of said data, a ftCreationTime field holding a creation time, a ftLastAccessTime field holding a last access time, a ftLastWriteTime field holding a last write time, an nFileSizeLow field and an nFileSizeHigh field holding a value describing the length of the object in bytes and a cFileName field holding characters that specify a filename."

Method claim 2 depends on claim 1. Claim 3 is directed to a computer-readable medium having computer-executable instructions adapted to cause the computer system to perform a method of one of Claims 1 to 2.

V. The appellant's substantive arguments are summarised as follows.

The claimed method modifies the internal operation of a computer system and is therefore technical. Moreover, the method achieves a non-obvious improvement over the appellant's prior operating system Windows 3.1 because the available prior art does not suggest a clipboard format which enables non-file data to be transferred for encapsulation into a file at a receiving application program (data sink) which expects to handle a file format (rather than a text format, for example). The claimed combination of a file contents clipboard format and a file group descriptor format enables the computer system to create a file at the data sink and to encapsulate the transferred data into the file. Prior to the invention, the obvious approach was either to refrain from communicating non-file data to an application which cannot handle such data, or to set up a common standard for the sending and the receiving application programs in the first place (e.g. to send file data if the receiving application expects file data).

The claimed method transfers data in a plurality of clipboard formats whereas each data object (scrap object) created by the method of D1 stores data in a single clipboard format (D1, column 7, lines 24 to 29). While also setting out from Windows 3.1, D1 is not concerned with transferring non-file data but serves a different purpose (persistent data storage).

D2 describes a method for entering data into a multiple partition clipboard and transferring the data from the multiple partition clipboard into an application. However, where a large number of partitions is used, the user has to manage a large number of numeric codes to transfer multiple pieces of information via the clipboard. A file group descriptor format as claimed handles the transfer of a group of data automatically.

VI. With respect to the request for correction of claim 5 of the divisional application as filed, the appellant argues that both the presence of an error (technical incompatibility with claim 4 of the divisional application) and its correction (deletion of the incompatible reference) are immediately evident to the skilled person.

VII. The Board pronounced its decision at the end of the oral proceedings.

## Reasons for the Decision

1. *Article 10b RPBA - Late filed request*

The amended claim set submitted at the oral proceedings before the Board responded to objections raised when the appellant was summoned to oral proceedings in relation to the present case, its copending parent application and the copending divisional applications. One of those objections was that the description presented the file contents clipboard format and the file group descriptor clipboard format as closely interrelated. Since these formats were considered as essential to each other, it was not surprising that the appellant joined them to form the amended claim 1 which has been distinguished from claim 1 of the parent application by the inclusion of further details.

Therefore, the Board admitted the late-filed request into the proceedings.

2. *Article 76(1) EPC - Comparison with the parent application*

In the following, the published version of the parent application will be denoted

M0 = EP-A-0 717 354,

and the published version of the present divisional application will be denoted

Div1 = EP-A-1 028 377.

2.1 Paragraphs 0001 to 0032 of the description in Div1 are identical to the corresponding paragraphs of the parent application M0. Paragraphs 0033 to 0065 of Div1 reflect the 33 claims of M0. Paragraph 0066 of Div1 resumes the last paragraph of the description of M0. Figures 1 to 6 of Div1 duplicate those of M0. Method claims 1 to 3 of Div1 correspond to claims 6 to 8 of M0. System claims 6 to 8 of Div1 correspond to claims 11 to 13 of M0.

2.2 Method claim 4 of Div1 corresponds to claim 9 of M0 but the dependency of claim 4 has multiplied: It is appended to "one of claims 1 to 3" whereas claim 9 of M0 was appended to claim 6 of M0 only. System claim 9 of Div1 corresponds to claim 14 of M0 but the dependency of claim 9 has multiplied: It is appended to "one of claims 6 to 8" whereas claim 14 of M0 was appended to claim 11 of M0 only.

That does not add any matter since the feature of claims 4 and 9 (data transfer via the clipboard) reflects a general teaching of M0.

2.3 Method claim 5 of Div1 corresponds to claim 10 of M0 but the dependency of claim 5 has multiplied: It is appended to "one of claims 1 to 4" whereas claim 10 of M0 was appended to claim 6 of M0 only.

While the additional reference of claim 5 to claims 2 and 3 in Div1 remains within the content of the parent application as filed, the dependency of claim 5 on claim 4 in Div1 *prima facie* appears to add matter because it suggests formally that a drag-and-drop operation might be performed using the clipboard to effect the data transfer, whereas the disclosure of M0 points out that in drag-and-drop operations the data is *not* passed through the system store (= clipboard) used for cut and copy operations (M0, page 2, lines 25 to 27).

On the other hand, since the clipboard and drag-and-drop operations clearly relate to mutually exclusive technical alternatives (see Div1, paragraphs 0002 to 0004 and 0020), the incompatibility of claims 4 and 5 is obvious to the skilled reader of Div1. At the same time, its correction is immediately evident: the claim reference which does not make technical sense should be deleted. Hence, the conditions established by opinion G 3/89 (OJ EPO 1993, 117; see points 2 to 6 of the reasons) for allowing corrections under Rule 88 EPC, second sentence are met. Therefore, the Board accepts the appellant's request pursuant to Rule 88 EPC to consider claim 5 of Div1 as appended to one of claims 1 to "3".

Taking the correction into account, the divisional application Div1 does not extend beyond the content of the parent application M0 and, thus, meets the requirements of Article 76(1) EPC.

3. *Article 123(2) EPC - Admissibility of amendments*

The Board is satisfied that the amendments made after the filing date of the divisional application are based on original disclosure. As the content of the parent application M0 is fully included in the present divisional application (see point 2 *supra*) and in each of the other divisional applications processed by the Board, uniform reference will be made to the parent application.

3.1 The amended claim 1 sets out from claim 1 as originally filed and published (M0) and has been amended as follows.

3.1.1 While the original version of claim 1 (M0) specifies the file contents clipboard format as a format for holding "contents of a file" with non-file data being provided for a data transfer in that format (M0, page 6, lines 26/27), the amended version defines generally that the file contents clipboard format is to hold non-file "data". That generalisation is based on M0, page 4, line 24, and Figure 3, steps 37 and 38.

3.1.2 The claimed method is specified as using several clipboard formats including a text clipboard format, a file contents clipboard format and a file group descriptor format. The text clipboard format forms part of the starting point of the invention (Microsoft

Windows 3.1) mentioned in the introductory portion of the description (M0, page 2, lines 14 to 16). The clipboard may simultaneously hold the same data in different clipboard formats (M0, page 2, lines 16/17). M0 expands the number of clipboard formats (M0, page 2, lines 33/34) to supplement those provided by the Windows 3.1 operating system (M0, page 2, lines 28/29; page 3, lines 35/36). M0 enumerates a number of expanded clipboard formats of the preferred embodiment (M0, page 3, lines 34/35), in particular the file contents clipboard format and the file group descriptor format (M0, page 4, lines 22 to 31), those formats being closely interrelated (M0, page 2, lines 35 to 38; page 4, lines 52/53; page 5, lines 39 to 43).

3.1.3 The selected data is converted into the additional clipboard format and stored in a data object (M0, page 3, lines 37 to 40; page 5, lines 44 to 46; Figure 2, steps 30, 32). In particular, where the file contents clipboard format is used, the selected data must be converted into that format (M0, page 4, lines 26 to 28; Figure 3, step 38).

3.1.4 The file group descriptor format is used to hold descriptive information about the non-file data that is to be encapsulated into a file (M0, page 2, lines 35 to 38; page 4, lines 51 to 54; claims 6 and 11).

3.1.5 The data transfer is completed (M0, page 4, line 28; Figure 3, step 39) by providing a handle to the data object (M0, page 3, lines 43/44; Figure 2, step 34), using the handle to paste the data of the data object to a data sink (M0, page 3, lines 44/45; Figure 2, step 36), using the descriptive information to enable the computer system to create a file at the data sink (M0, page 4, lines 52/53; page 5, lines 37/38), and encapsulating the data object into the file (M0, page 4, lines 29/30 and 52/53; page 5, line 43; Figure 3, step 40). The file group descriptor clipboard format supports the transfer of a data group into a group of files (M0, page 5, lines 35 to 43). As the file group descriptor may hold "one or more file descriptors" (M0, page 4, lines 51/52), the creation of one file at the data sink (as covered by claim 1) corresponds to the minimum number disclosed.

3.1.6 The field structure of the file descriptor recited in the last paragraph of amended claim 1 is disclosed from page 4, line 51 to page 5, line 34 of M0.

3.2 Claim 2 of the amended claim set is based on M0, page 2, lines 33 to 38; page 4, lines 51 to 53; page 5, lines 35 to 43; Figure 4.

The instruction medium claim 3 is based on the preceding method claims and the fact that the disclosed method for a computer system must be run by computer-executable instructions stored on a computer-readable medium (Figure 1, memory 20).

4. *Article 84 EPC - Clarity of claim 1 and support by the description*

4.1 The aforementioned bases for amended claim 1 show at the same time that the claim is supported by the description. Embodiments of the overall method of claim 1 are described in relation to the flow charts of Figures 2 to 4.

4.2 Moreover, the amended claim 1 makes clear that the file contents clipboard format and the file group descriptor format are provided in addition to the conventional text clipboard format and must not be confused with it. Further, although the application fails to provide a comprehensive definition of file data or non-file data, the Board is satisfied that the skilled person is able to draw a line on the basis of general knowledge according to which file data is considered to be organised in a file directory (e.g. a hierarchical directory).

It is further clear that the method steps make use of the claimed clipboard formats without being restricted to a clipboard operation. In other words, a drag-and-drop operation using clipboard formats but not using the clipboard is not ruled out by the wording of claim 1 (see M0, page 2, lines 25 to 27).

Finally, it is clear that the file descriptor structure specified in the last paragraph of claim 1 distinguishes the scope of that claim from the claims of the parent application and each copending divisional application. Hence, no issue of double patenting arises.

5. *Article 83 EPC - Enabling disclosure*

5.1 The application as filed seeks to increase the number of data formats usable to transfer data from a source (e.g. a first window, document or the like) to a destination (e.g. a second window, document or the like) via the clipboard of a data processing system or via a drag-and-drop operation which uses clipboard formats as part of an object linking and embedding (OLE) protocol (M0, statement of the problem, page 2, lines 28/29).

5.2 The solution consists generally in providing expanded clipboard formats (M0, page 2, lines 33/34). A specific embodiment is a file contents clipboard format (CF_FILECONTENTS) that allows "data that is not a file" to be transferred in such a way that the data is finally encapsulated into a file (M0, page 2, lines 34/35; page 4, lines 22 to 31; Figure 3). To this end, the file contents clipboard format is accompanied by a file group descriptor clipboard format (CF_FILEGROUPDESCRIPTOR) which enables the resulting file to be created (M0, page 2, lines 35 to 38; page 4, lines 22/23; page 4, line 51 to page 5, line 43).

Instead of transferring data directly, a "handle" (unique identifier) which points to the data is used to retrieve the data from a memory and to paste the data into a destination file (M0, page 2, lines 13/14; page 3, lines 37 to 45; Figure 2, step 34).

5.3 The function and structure of the file group descriptor format are disclosed from page 4, line 51 to page 5, line 34 of M0.

The file contents clipboard format is described by way of its function, i.e. its ability to transfer non-file data in such a manner that the non-file data can be encapsulated into a file at the data sink with the help of descriptive information provided by the file group descriptor clipboard format. Thus, the file contents clipboard format and the file group descriptor clipboard format jointly assist in changing the data format during the transfer operation, whereas in conventional clipboard formats the data either has to be accepted in the form offered or cannot be used by the receiving application program.

An exemplary property of the file contents clipboard format is disclosed in functional terms: "for holding the contents of a file" (M0, page 2, lines 34/35; claims 1 and 5). Although that format is not described in detail, the Board judges that the skilled person is able to implement the desired function of the format on the basis of common knowledge and relying on the accompanying file group descriptor format documented in M0.

6. *Article 52(1)(2)(3) EPC - Eligibility for patent protection*

6.1 The Board concurs with the appellant in considering the claimed method as an invention within the meaning of Article 52(1) EPC.

Claim 1 relates to a method implemented in a computer system. T 258/03 - *Auction method/Hitachi* (OJ EPO 2004, 575) states that a method using technical means is an invention within the meaning of Article 52(1) EPC. A computer system including a memory (clipboard) is a technical means, and consequently the claimed method has technical character in accordance with established case law.

Moreover, the Board would like to emphasise that a method implemented in a computer system represents a sequence of steps actually performed and achieving an effect, and not a sequence of computer-executable instructions (i.e. a computer program) which just have the potential of achieving such an effect when loaded into, and run on, a computer. Thus, the Board holds that the claim category of a computer-implemented method is distinguished from that of a computer program. Even though a method, in particular a method of operating a computer, may be put into practice with the help of a computer program, a claim relating to such a method does not claim a computer program in the category of a computer program. Hence, present claim 1 cannot relate to a computer program as such.

6.2 The Board also considers the claimed method steps to contribute to the technical character of the invention.

These steps solve a technical problem by technical means in that functional data structures (clipboard formats) are used independently of any cognitive content (see T 1194/97 - *Data structure product/Philips;* OJ EPO 2000, 525) in order to enhance the internal operation of a computer system with a view to facilitating the exchange of data among various application programs. The claimed steps thus provide a general purpose computer with a further functionality: the computer assists the user in transferring non-file data into files.

6.3 Claim 3 is directed to a computer-readable medium having computer-executable instructions (i.e. a computer program) on it to cause the computer system to perform the claimed method. The subject-matter of claim 3 has technical character since it relates to a computer-readable medium, i.e. a technical product involving a carrier (see decision T 258/03 - *Auction method/Hitachi* cited above). Moreover, the computer-executable instructions have the potential of achieving the above-mentioned further technical effect of enhancing the internal operation of the computer, which goes beyond the elementary interaction of any hardware and software of data processing (see T 1173/97 - *Computer program product/IBM*; OJ EPO 1999, 609). The computer program recorded on the medium is therefore not considered to be a computer program as such, and thus also contributes to the technical character of the claimed subject-matter.

## 7. *Article 54 EPC - Novelty*

7.1 The Board agrees with the appellant in regarding the operating system Windows 3.1 (released in 1992) as the closest available prior art. As acknowledged in the introductory portion of the description (M0, page 2, lines 8 to 24), that operating system uses two clipboard formats: a bitmap clipboard format and a text clipboard format. The same data may be held simultaneously in different clipboard formats (M0, page 2, lines 16/17) which have to be created by the sending application program. The receiving application program will pick up the data in a format understood by the receiving application, if such a format is included in the range of clipboard formats.

While Windows 3.1 is known to use handles (M0, page 2, lines 13/14), it does not appear to provide a clipboard format which enables non-file data (e.g. an embedding in a mail message, see M0, page 4, lines 22 to 31) to be transferred to applications designed to encapsulate data into a file. (Non-file data in the file contents clipboard format is not a conventional text block copied into the clipboard since text is to be stored in the conventional text clipboard format which exists in addition to the file contents clipboard format.)

Therefore, the method of claim 1 is novel over Windows 3.1 in that the method adds a pair of clipboard formats (the file contents clipboard format in conjunction with the file group descriptor format) which allows non-file data to be transferred with additional descriptive information such that the data sink is in a position to create a file for encapsulation of the data. The specific fields of the file descriptor format provide additional detail to claim 1.

7.2 According to D1, so-called scrap objects may be created by either a drag-and-drop mechanism or by a clipboard mechanism (D1, column 2, lines 38 to 51). To use the clipboard mechanism, data items are selected to be extracted (copied or cut) from a document (Figure 4, steps 44 and 46). The user switches to the application where he wishes to add the extracted information and chooses a "Create Scrap" command (Figure 4, step 48).

Unlike a clipboard, the scrap objects hold extracted information persistently until the information is to be transferred (D1, column 2, line 57 to column 3, line 2). D1 thus addresses the problem that users often inadvertently delete the contents of the clipboard by transferring other information to the clipboard (D1, column 1, lines 22 to 29). Moreover, the scrap object allows transfers between application programs that support different transfer mechanisms. For instance, a first application that supports only drag-and-drop operations may transfer information via a scrap object to a second application program that supports only clipboard transfer operations (D1, column 3, lines 2 to 8).

As compared to D1, the method of claim 1 uses specific clipboard formats (a file contents clipboard format and a file group descriptor format) for transferring non-file data and encapsulating it into a file, whereas the transferable scrap objects of D1 are generated in one or more of the conventional clipboard formats of Windows 3.1 (D1, column 1, lines 10 to 22; column 3, lines 47 to 49) which formats are not concerned with transferring non-file data.

Hence, the method of claim 1 is novel over D1.

7.3 D2 describes a clipboard having multiple partitions which may be used to transfer multiple data strings in a single operation. However, D2 does not deal with changing a data format during data transfer, and it is not concerned with transferring non-file data.

8. *Article 56 EPC - Inventive step*

8.1 With respect to the closest prior art (Windows 3.1), the method of claim 1 solves the problem of how to facilitate a data exchange across different data formats, in particular when transferring non-file data.

8.2 The solution provides for a file contents clipboard format and a file group descriptor format which interact to allow data to be sent in a first format (non-file data) and processed in a second format (file encapsulation) at the receiving data sink.

8.3 While Windows 3.1 is able to copy a block of text or graphics into the clipboard in corresponding clipboard formats at the same time and to paste the data into a receiving document or window in one of the clipboard formats offered, a clipboard format adding the aforementioned formatting metamorphosis (encapsulation into a file upon transfer of non-file data) does not derive in an obvious manner from the pre-existing operating system.

8.4 Like the present application, D1 sets out from Windows 3.1 (D1, column 1, lines 10 to 29), but the method of D1 serves a different purpose: It stores each data object persistently (scrap object having a file name, D1, column 6, line 42 to column 7, line 15) in order to avoid a data loss when the user shuts down the computer system or copies fresh data into the clipboard. D1 does not suggest expanding the number of clipboard formats. In particular, it does not suggest a clipboard format capable of converting a data format upon a data transfer. Notably, D1 is not concerned with transferring non-file data and, thus, does not suggest encapsulating non-file data into a file at a data sink.

8.5 Remaining document D2 does not give any hint to the claimed solution since, as mentioned before (point 7.3 *supra*), D2 does not deal with changing a data format during data transfer, and it is not concerned with transferring non-file data. Finally, the details of the file group descriptor are neither anticipated nor rendered obvious by the available prior art.

8.6 Therefore, the Board considers the method of claim 1 as involving an inventive step.

8.7 The dependent method claim 2 includes the steps considered non-obvious in claim 1.

8.8 The computer-readable medium according to claim 3 is regarded as non-obvious by virtue of its reference to one of the method claims.

9. The description has been adapted to the present version of claims.

10. The Board concludes that the application and the invention to which it relates meet the requirements of the EPC.

## Order

**For these reasons it is decided that:**

1. The decision under appeal is set aside.

2. The case is remitted to the department of first instance with the order to grant a patent in the following version:

   - claims 1 to 3 as submitted at the oral proceedings;

   - description: pages 1, 3, 5, 7 to 9 as originally filed;

   - description: pages 2, 4, 6, 10 to 12 as submitted at the oral proceedings;

   - drawings: Figures 1 to 6 as originally filed.

The Registrar: The Chairman:

P. Guidi S. Steinbrener