

**Internal distribution code:**

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

**Datasheet for the decision  
of 13 September 2013**

**Case Number:** T 0624/10 - 3.5.01

**Application Number:** 05106445.9

**Publication Number:** 1650675

**IPC:** G06F17/30

**Language of the proceedings:** EN

**Title of invention:**  
External metadata processing

**Applicant:**  
Microsoft Corporation

**Headword:**  
Data flow consistency check/MICROSOFT

**Relevant legal provisions:**  
EPC 1973 Art. 56

**Keyword:**  
Inventive step - (no)



**Beschwerdekammern  
Boards of Appeal  
Chambres de recours**

European Patent Office  
D-80298 MUNICH  
GERMANY  
Tel. +49 (0) 89 2399-0  
Fax +49 (0) 89 2399-4465

Case Number: T 0624/10 - 3.5.01

**D E C I S I O N**  
**of Technical Board of Appeal 3.5.01**  
**of 13 September 2013**

**Appellant:** Microsoft Corporation  
(Applicant) One Microsoft Way  
Redmond, WA 98052 (US)

**Representative:** Grünecker, Kinkeldey,  
Stockmair & Schwanhäusser  
Leopoldstrasse 4  
80802 München (DE)

**Decision under appeal:** **Decision of the Examining Division of the  
European Patent Office posted on 27 October 2009  
refusing European patent application No.  
05106445.9 pursuant to Article 97(2) EPC.**

**Composition of the Board:**

**Chairman:** S. Wibergh  
**Members:** P. Scriven  
P. Schmitz

## **Summary of Facts and Submissions**

- I. The appeal is against the Examining Division's decision to refuse European patent application 05106445.9 on the grounds that the main request involved too many independent claims in the same category (Rule 43(2) EPC), the first to third auxiliary requests involved added subject matter (Article 123(2) EPC, and because the subject matter of claim 1 according to the fourth and fifth auxiliary requests did not involve an inventive step (Article 56 EPC 1973).
- II. With the statement setting out its grounds of appeal, the appellant re-submitted the main and five auxiliary requests; and requested that oral proceedings be held, if the main request were not allowed.
- III. The Board arranged oral proceedings, and, in a communication sent with the summons, set out its provisional opinion.
- IV. In response, the appellant sent a letter dated 11 July 2013. With this letter, the appellant maintained the six substantive requests on file, but ordered them differently. Two new auxiliary requests, IV and VI, were also submitted.
- V. At oral proceedings, the appellant stated its requests as: that the decision under appeal be set aside, and that a patent be granted on the basis of the main request or of auxiliary requests I to VII filed with the letter dated 11 July 2013.
- VI. The appellant's arguments can be summarised as follows.  
  
The invention concerned a data flow which operated on

some external data file. An example would be a program which operated on a relational database. The programmer would expect the database to remain unchanged, but in fact changes did sometimes happen. For example, a column of prices expressed in dollars might be replaced by a column of prices expressed in yen, or a new column might be added, or a column might be removed. In such cases, the program would no longer work. Such was the problem addressed by the invention.

The invention solved the problem by storing information about what data was stored in the columns of the database. That allowed a check to be made: a comparison of the stored information with what was actually in the database would reveal if anything had changed, and that could be done before the program was actually run.

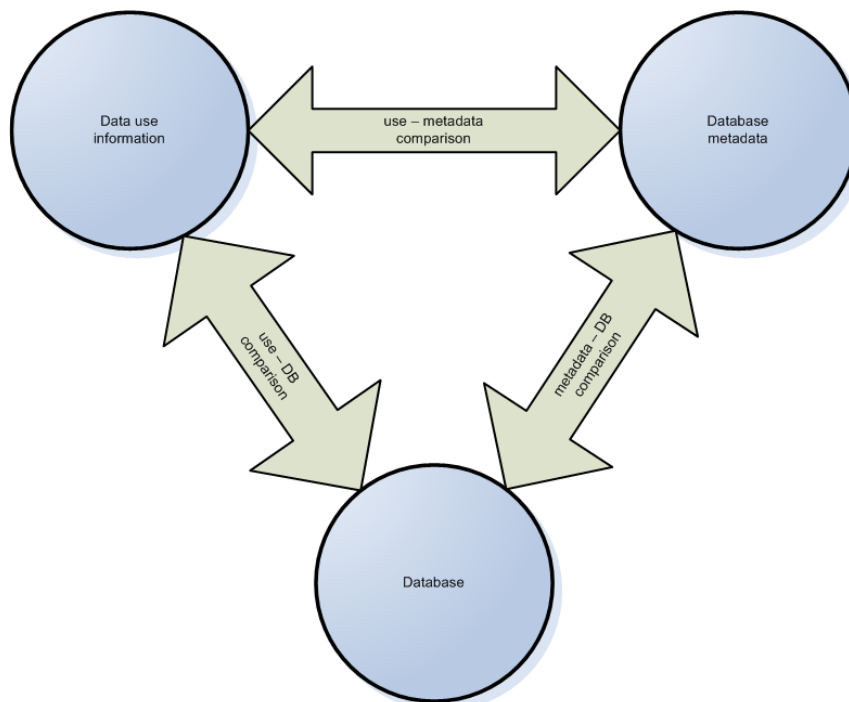
The prior art did not disclose or suggest such a comparison. Of the documents cited by the Examining Division, D4 (GB 1 494 505) was closest to the invention. It dealt with the same problem, that a database might have changed since the program was written, and stored some metadata, but used it differently. According to D4, if data was of the wrong type, a conversion could be performed, but there was no comparison of the actual contents of the database with the stored metadata to see whether they matched. Indeed, there was no obvious incentive for the skilled person, a programmer, to store metadata and compare it to actual contents. If, for example, the programmer noted that the program was not producing correct results, he would consider the actual contents of the database and re-write the program accordingly. There would be no need to make a comparison between the contents and any stored metadata.

The invention also stored information on the way the

program used data. That allowed an extra check to be made. It might happen, for example, that a database had been changed, so that the metadata check would indicate an inconsistency, but the inconsistency involved only columns which the program did not use. In such a case, the second check would indicate that the program would run successfully.

A third check was also possible: the stored use information could be compared with the stored metadata. If, for example, the database had changed and the metadata had been updated to match, and the database were not available for some reason, the comparison of the use data with the metadata would be able to reveal any mismatch.

The following diagram showed the entities and comparisons that could be made.



VII. Claim 1 according to the main request reads as follows.

*A computer implemented method for preparing data flow information regarding a data flow using at least one external data file, each of said external data files comprising columns of information, said method comprising:*

*storing a data flow model, said data flow model comprising use data describing, for each of said at least one external data file, a use of at least one column from said external data file;*

*storing metadata comprising data describing at least one column of each of said at least one external data file;*

*using said stored metadata to determine whether said data flow model and said external data files are consistent, where using said stored metadata to determine whether said data flow model and said external data files are consistent comprises for each external data file, verifying that said stored metadata is consistent with a current state of said external data file and where using said stored metadata to determine whether said data flow model and said external data files are consistent further comprises providing a validation state indicating whether said stored metadata is consistent with said current state of said external data file.*

VIII. Claim 15 according to the main request reads as follows.

*A data flow modeling system for preparing data flow information regarding a data flow using at least one external data file, each of said external data files comprising columns of information, said data flow modeling system comprising:  
data flow model storage storing a data flow model, said data flow model comprising use data describing, for each of said at least one external data file, a use of at least one column from said external data file;  
metadata storage storing metadata comprising data describing at least one column of each of said at least one external data file;  
consistency checker using said stored metadata to determine whether said data flow model and said external data files are consistent, where said consistency checker, for each external data file, verifies that said stored metadata is consistent with a current state of said external data file, and said consistency checker comprises validation state determiner for providing a validation state indicating whether said stored metadata is consistent with said current state of said external data file.*

- IX. The claims according to auxiliary request I read identically to claims 1 - 14 according to the main request.
  
- X. Claim 15 according to auxiliary request II reads identically to claim 15 according to the main request, but claim 1 has the following appended text:

*... ;*

*determining whether said data flow model and said external data files are consistent by determining if each said use of at least one column from said external data file described in said use data is possible given a current state for each of said external data file, and where determining whether said data flow model and said external data files are consistent by determining if each said use of at least one column from said external data file described in said use data is possible given a current state for each of said external data file comprises providing a validation state indicating whether said data flow model is consistent with said current state of said external data files.*

- XI. The claims according to auxiliary request III read identically to claims 1 - 14 according to auxiliary request II.
- XII. The claims according to auxiliary request IV read identically to claims 15 - 18 according to the main request and auxiliary request II.
- XIII. Claim 1 according to auxiliary request V appends the following to claim 1 according to auxiliary request II:

*...; and  
where said using said stored metadata to  
determine whether said data flow model and*



*said external data files are consistent comprises:  
verifying that said stored metadata is consistent with said data flow, where said verifying that said stored metadata is consistent with said data flow comprises determining if each said use of at least one column from said external data file described in said use data is consistent with said stored metadata.*

- XIV. Claim 15 according to auxiliary request V appends the following to Claim 15 according to auxiliary request II (or to the main request)

*...; and  
where said consistency checker comprises:  
consistency verifier for verifying that said stored metadata is consistent with said data flow where said consistency verifier determines if each said use of at least one column from said external data file described in said use data is consistent with said stored metadata.*

- XV. The claims according to auxiliary request VI read identically to claims 15 - 18 according to auxiliary request V.

- XVI. Claim 1 according to auxiliary request VII reads as follows.

*A method for preparing data flow information regarding a data flow using at least one*

*external data file, each of said external data files comprising columns of information, said method comprising: storing a data flow model, said data flow model comprising use data describing, for each of said at least one external data file, a use of at least one column from said external data file; storing metadata comprising data describing at least one column of each of said at least one external data file; and using said stored metadata to determine whether said data flow model and said external data files are consistent.*

### **Reasons for the Decision**

1. The invention concerns a data flow and a file that comprises columns. The data flow uses, in some way, the data in the file, but the file may change independently of the data flow.
2. The invention addresses the problem that the data in the file may not match what was originally there, or what the program expects. It provides an indication of whether the program and the data file match or not.
3. In order to do that, metadata and use data are stored. The metadata indicate what data are stored in a column, whereas the use data indicate how a column is used in

the program. The stored metadata and use data allow checks to be made that seek to ascertain whether or not the file has changed in a way that affects the program.

4. In each of the requests, the file on which the data flow operates is defined as "external". It is common ground that the intended meaning is simply that the file can be modified independently of the data flow.
5. The meaning of "column" is doubtful, but it is common ground that the columns of a table in a relational database will do.

*The main request*

6. Claim 1 defines a "computer implemented method". The method comprises the storing of use data and of metadata, as explained above. For each external file, the metadata are compared with the state of the file. What is produced is an indication of whether the two are consistent.
7. The claim does not specify what is meant by "consistent". The description does give some indications. In paragraph [0012] of the published application, it is stated that, "... the external metadata [can be] used in order to determine whether any changes have occurred to the data collections, and if so, what actions might be taken." Paragraph [0034] says much the same. Paragraph [0036], in speaking of an embodiment, says "The stored external metadata describes the columns present in the external data files/data collections." Given the lack of definition in the claim, and the somewhat vague statements in the description, the Board considers claim 1 as defining a

rather abstract and vague test: do the data contained in the at least one column of the file fall under the description in the metadata?

8. The Board takes, as starting point, a conventional computer system with a file, arranged with columns, that is subject to change (e.g. a table in a relational database). A programmer has the task of designing a data flow which uses that file. The issue of inventive step comes down to this: would it have been obvious for the programmer to store information about what is in the columns of data, and later to check whether or not it is still there?
9. The appellant has two arguments. The first is that a programmer would consider a database as a static thing, a thing that does not change. She would not anticipate changes and, for that reason, would not take measures to detect them. The Board cannot follow that argument. A programmer is not willfully blind. She knows that databases are subject to change. For example, as a response to the "millennium bug", date formats were changed; and when the Euro was introduced, prices in national currencies were changed to prices in Euros. The Board also notes that D4 recognises the issue, and proposes a solution using metadata, albeit in a different manner from the present invention.
10. The appellant's second argument is that the programmer would never have needed to compare the state of the database with stored metadata. According to this argument, if the programmer notes that the data flow no longer works correctly, she would find the new specification of the database, and modify her program to fit. There would be no need for a comparison with what was previously there. The Board cannot accept that

argument either. It is based on the idea of a programmer who does nothing unless a program fails. Some programs are so important that they cannot be allowed to fail. It would be better not to run them, and checking that the required data are available would have been obvious in that context. Similarly, a programmer may work on many programs which all rely fundamentally on some database over which she has no control. Such a programmer would have an interest in identifying changes in the database, whether in order to prevent her programs failing, with potentially catastrophic results, or to identify that new and useful information had been added. It would be a valid end in itself to identify that some information has been moved, removed, re-formatted, or added. In the Board's view, if the programmer wants to be able to spot changes in database outside her control, the idea of storing an indication of what used to be there and comparing that with what is there now would have been obvious.

11. The Board is, therefore, satisfied that the programmer would have had clear incentives to store information on the content of columns, and to make a subsequent comparison. Furthermore, it would have been obvious to store that information in the computer, and that would have constituted metadata in the sense of the claim. The Board is also satisfied that the use of metadata describing the contents of columns was known (see D4, for example).
  
12. From this, it follows, that the subject matter of claim 1 does not involve an inventive step, and that the main request is, consequently, not allowable.

13. It is convenient to note, here, that claim 15 is directed to a system which carries out the method defined in claim 1, although it does not actually define the use of a computer. The appellant agreed that this claim would stand or fall with claim 1.

*Auxiliary request I*

14. Claim 1 according to this request is identical to that according to the main request. This request is not allowable for the same reasons.

*Auxiliary request II*

15. Claim 15 according to this request is identical to claim 15 according to the main request. As already noted (point 13.), its subject matter does not involve an inventive step, and the request is not allowable for that reason.
16. It is also convenient to consider the method defined by claim 1. As compared with claim 1 according to the main request, there is an additional step: not only are the metadata compared with the external data file, but the use data are also compared with it. There are, therefore, two indications of consistency.
17. The appellant explained that whereas the metadata comparison might indicate that a particular column had changed, the use data comparison might show that the data flow made no use of that column and would work just as before.

18. In the Board's view, the second comparison includes the case of a checking that data types are correct: does the database provide data of the type the program expects? An example would be a program that expects a particular variable to contain a date in the format "DD.MM.YYYY", and which obtains the value of the variable from a particular column of a database. Type checking would involve looking to see whether the data in the column were really of the correct type, in this case, was it a date in the correct format. The Board considers such type checking as part of a programmer's general knowledge and would be routinely applied.
19. The two checks together provide some information that cannot be derived from one alone. From the metadata check, we learn that the database has changed; from the use data check, that the file has the data needed by the data flow. Together, we might, for example, learn that although the data file has changed, the data flow will be unaffected. In the Board's view, a programmer, who already knew that the data file had changed, and how, would also have gone on to consider whether the data flow would be affected. The method defined by claim 1 implements an obvious manner of answering that question.
20. The Board, therefore, considers the request unallowable because each of claims 1 and 15 defines subject matter which does not involve an inventive step.

*Auxiliary request III*

21. Claim 1 according to this request is identical to claim 1 according to auxiliary request II. This request is

not allowable for the reasons already set out.

*Auxiliary request IV*

22. Claim 1 according to this request is identical to claim 15 according to auxiliary request II and according to the main request. This request is not allowable for the reasons already given.

*Auxiliary request V*

23. Claim 1 according to this request adds a third check to the two already defined in claim 1 according to auxiliary request II. This third test is a comparison of use data with metadata. The appellant explained that this is useful when the data file itself is not available. The Board, however, notes that the claim is not limited to the third check being carried out under those circumstances. According to the claim, all three checks are always performed.
24. As set out regarding auxiliary request II, the metadata need not agree with the contents of the data file. The comparison between the use data and the metadata, however, requires the metadata and the data file to be consistent. Otherwise, nothing useful is obtained. The appellant explained that, although not defined in claim 1 according to auxiliary request II, the metadata can be updated when the external data file is changed. If that is done, the metadata and the external data file will be consistent, and the first check (the comparison of the metadata with the actual file contents) will show no inconsistency; whereas the second check (the comparison of use data with the actual file contents)



might well. In that case, the third check does not provide any information that is not already obtained from the second check. On the other hand, if the metadata are not updated, the first check would show an inconsistency, and the second might do so. Either way, the third check provides no new information.

25. The Board, therefore, considers that this method does not solve any problem (technical or not) that is not already solved by the obvious method defined in claim 1 according to auxiliary request II, and, therefore, does not involve an inventive step.
26. The Board is furthermore of the opinion that the idea of comparing metadata with use data, both of which are maintained for the purposes set out above, would have been obvious in the particular situation that the data file itself was not readily available.
27. Claim 15 according to this request defines a system that implements a more general method than that defined by claim 1. Only two of the three checks are necessarily performed, while the third is not excluded. As the more specific method lacks inventive step, so does the more general. The corresponding system defined by claim 15 lacks inventive step for the same reasons.

*Auxiliary request VI*

28. Claim 1 according to this request is identical to claim 15 according to auxiliary request V. As already noted, the Board does not see an inventive step. This request, therefore, is not allowable.

*Auxiliary request VII*

29. Claim 1 according to this request defines a method that is broader than that defined in claim 1 according to the main request. As the subject matter of the latter does not involve an inventive step, that of the former cannot. This request, therefore, is not allowable.

**Order**

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

The appeal is dismissed.

The Registrar:

The Chairman:



T. Buschek

S. Wibergh

Decision electronically authenticated