

Internal distribution code:

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

**Datasheet for the decision
of 10 October 2024**

Case Number: T 2353/22 - 3.5.07

Application Number: 17851913.8

Publication Number: 3549036

IPC: G06F17/30

Language of the proceedings: EN

Title of invention:

Generating, accessing, and displaying lineage metadata

Applicant:

AB Initio Technology LLC

Headword:

Lineage metadata/AB INITIO

Relevant legal provisions:

EPC Art. 56

Keyword:

Inventive step - mixture of technical and non-technical features - after amendment - (yes)

Decisions cited:

G 0001/19, T 0115/85, T 0641/00, T 0619/02, T 1351/04,
T 0756/06, T 1670/07, T 0697/17, T 0731/17, T 2626/18,
T 3176/19, T 1272/20



Beschwerdekammern
Boards of Appeal
Chambres de recours

Boards of Appeal of the
European Patent Office
Richard-Reitzner-Allee 8
85540 Haar
GERMANY
Tel. +49 (0)89 2399-0

Case Number: T 2353/22 - 3.5.07

D E C I S I O N
of Technical Board of Appeal 3.5.07
of 10 October 2024

Appellant: AB Initio Technology LLC
(Applicant) 201 Spring Street
Lexington, Massachusetts 02421 (US)

Representative: Herrmann, Daniel
Boehmert & Boehmert
Anwaltspartnerschaft mbB
Pettenkoferstrasse 22
80336 München (DE)

Decision under appeal: **Decision of the Examining Division of the
European Patent Office posted on 14 June 2022
refusing European patent application
No. 17851913.8 pursuant to Article 97(2) EPC**

Composition of the Board:

Chairman J. Geschwind
Members: P. San-Bento Furtado
C. Barel-Faucheux

Summary of Facts and Submissions

I. The appeal lies from the decision of the examining division to refuse European patent application No. 17851913.8 for lack of inventive step of the subject-matter of the claims of the main request and of claim 1 of the first to sixth auxiliary requests over a "computer system comprising a query processing method" illustrated by prior-art document D1:

D1: L. Zamboulis: "XML Data Integration by Graph Restructuring", 19 June 2004, Key Technologies for Data Management, Lecture Notes in Computer Science (LNCS) 3112, pages 57 to 71.

The examining division considered that the subject-matter was essentially non-technical.

II. In the statement of grounds of appeal, the appellant requested that the decision under appeal be set aside and that a patent be granted on the basis of the main request or of one of the first to sixth auxiliary requests considered in the appealed decision.

III. In a communication accompanying a summons to oral proceedings, the board introduced the following documents into the proceedings:

D2: US 2010/0138420 A1, 3 June 2010;

D3: WO 2015/183738 A1, 3 December 2015;

D4: US 2016/0232230 A1, 11 August 2016;

D5: US 2016/0232229 A1, 11 August 2016;

D6: WO 2016/014615 A1, 28 January 2016;

D7: US 2016/0019057 A1, 21 January 2016;

D8: US 2014/0279979 A1, 18 September 2014.

These documents are publications of patent applications by the same applicant, document D2 being the publication of the international application corresponding to the US application cited in the description, page 12, last full paragraph.

The board expressed the view that the subject-matter of claim 1 of each of the requests lacked inventive step over the disclosure of document D2 in combination with common general knowledge. Documents D3, D6 and D8 disclosed some of the claim features.

- IV. With a letter of reply the appellant maintained its requests and submitted further arguments.
- V. Oral proceedings were held as scheduled. During the oral proceedings the appellant filed a revised new auxiliary request V, which amended the fifth auxiliary request, as the new main request and renumbered the previous main request and first to fourth auxiliary requests as first to fifth auxiliary requests. At the end of the oral proceedings, the Chair announced the board's decision.
- VI. The appellant's final requests were that the contested decision be set aside and that a patent be granted on the basis of the claims of the main request, corresponding to the revised new auxiliary request V filed during the oral proceedings before the board, or of one of the first to sixth auxiliary requests corresponding to the main request and first to fourth and sixth auxiliary requests considered in the decision under appeal.

VII. Claim 1 of the main request reads as follows:

"A method performed by a data processing apparatus, the method including:

receiving a portion of metadata (106, 112A, 112B, 112C, 128) from a data source (110A, 110B, 110C), the portion of metadata (106, 112A, 112B, 112C, 128) describing nodes and edges, wherein each node represents a metadata object and at least some of the edges each represent a one-way effect of one node upon another node, each edge having a single direction, and wherein a metadata object is a data element (202A) or a transformation (204A);

generating instances of a data structure (134, 300) representing the portion of metadata (106, 112A, 112B, 112C, 128) and storing the instances of the data structure (134, 300) in random access memory;

at least a first instance of the instances of the data structure (134, 300) including:

an identification value that identifies a node that corresponds to the first instance of the instances of the data structure (134, 300),

one or more property values representing respective properties of the corresponding node, and

a pointer (320A, 320B) that includes a reference to a memory location (322A, 322B) associated with a portion of the random access memory, the portion of the random access memory storing a second instance of the instances of the data structure (134, 300) that corresponds to another node, the pointer (320A, 320B) representing an edge (316A-316D) associated with the other node that corresponds to the second instance of the instances of the data structure (134, 300), wherein the edge represents an effect the node that corresponds to the first

instance has on the other node that corresponds to the second instance, or an effect the other node that corresponds to the second instance has on the node that corresponds to the first instance;

receiving a query (114, 126) for lineage metadata, said query including an identification of at least one particular element of data,

wherein the query (114, 126) includes an identification of a type of lineage and a walk plan (130, 132, 400) that identifies which types of nodes and edges are relevant to the identified type of lineage, and wherein the walk plan (130, 132, 400) includes conditions for following an edge or for collecting an edge or an instance of the data structure based on one or more property values representing respective properties of a corresponding node, wherein an edge is followed by accessing a memory location identified by a pointer representing the followed edge, and wherein an instance of a data structure is collected by adding data of the collected instance to lineage metadata to be returned in response to the query, and wherein an edge is collected by adding data representing the collected edge to lineage metadata to be returned in response to the query;

and

in response to receiving the query (114, 126), traversing the data structure (134) in accordance with the walk plan (130) to collect lineage metadata stored in the data structure (134) that is responsive to the query (114), including:

accessing data stored using the data structure (134, 300), including accessing the first instance of the instances of the data structure (134, 300), the first instance of the instances of the data structure (134,300) identified by a first identifier that

corresponds to the particular element of data identified by the query (114, 126);

accessing the pointer (320A, 320B) of the first instance;

accessing the second instance of the instances of the data structure (134, 300), the second instance of the instances of the data structure (134, 300) identified by a second identifier, the second identifier being stored at the memory location (322A, 322B) referenced by the pointer (320A, 320B) of the first instance;

collecting data of the second instance of the instances of the data structure (134, 300); and

based on the data stored using the data structure (134,300), generating a response to the query (114, 126), the response including the lineage metadata (106, 112A, 112B, 112C, 122, 128) responsive to the query, the lineage metadata (106, 112A, 112B, 112C, 122, 128) describing a sequence of nodes and edges, wherein one of the nodes of the sequence represents the particular element of data, and wherein the lineage metadata further includes data representing the other node that is associated with the second identifier; and

sending the response containing the lineage metadata (106, 112A, 112B, 112C, 122, 128) to a computer system (116) for causing a display of the computer system (116) to display a representation of lineage of the particular element of data in form of a lineage diagram (200A-200E) generated based on the lineage metadata,

wherein the relationship among the nodes shown in the lineage diagram (200A-200E) corresponds to the edge (316A-316D) represented by the pointer (320A, 320B)."

VIII. Claims 2 to 4 of the main request are dependent on claim 1.

- IX. Claim 5 of the main request reads as follows:
"A system including:
at least one non-transitory computer-readable storage medium storing executable instructions; and
one or more processors configured to execute the instructions, wherein execution of the instructions causes the system to perform the operations of the method of any one of claims 1 to 4."
- X. Claim 6 of the main request reads as follows:
"A non-transitory computer readable storage device storing instructions that, when executed, carry out the operations of the method of any one of claims 1 to 4."
- XI. The claims of the other requests are not relevant for this decision.

Reasons for the Decision

1. The invention concerns a method, a system and a device for supporting the storage of, access to and display of lineage metadata about data stored in a storage system. The lineage metadata of a data object provides information about the sources from which the data object was derived. For instance, how the data object was generated, from which source it was imported, how it has been used by applications, how it relates to other datasets or how its modification will affect tables (see original description, page 1 to page 2, fifth line).
- 1.1 The method according to the claimed invention includes a first step of receiving metadata from a data source, the metadata describing nodes and edges. Each node represents a metadata object, which can be a data element or a transformation. An edge represents a one-

way effect of one node upon another node. According to the description, page 10, first paragraph, the data elements can represent, for instance, "datasets, tables within datasets, columns in tables, and fields in files, messages, and reports". An example of a transformation is "an element of an executable that describes how a single output of a data element is produced".

- 1.2 After the step of receiving metadata from a data source, the claimed method further includes steps of generating a data structure representing the received metadata and receiving a query for lineage metadata. In response to receiving the query, the data structure is accessed and a response to the query is generated and sent to a computer system for display. The response includes the lineage metadata responsive to the query.

Main request

2. *Admission*

The main request was filed at the oral proceedings before the board in order to overcome minor deficiencies of claim 1 of the then fifth auxiliary request. The deficiencies were identified by the board for the first time at the oral proceedings, which constitutes an exceptional circumstance under Article 13(2) RPBA. Consequently, the main request is admitted into the appeal proceedings.

3. *Claim construction - Claim 1*

- 3.1 Claim 1 specifies a method performed by a data processing apparatus including main steps which the board summarises as follows:
 - (a) receiving a portion of metadata from a data source,

- (a1) the portion of metadata describing nodes and edges, wherein
- (a2) each node represents a metadata object,
- (a3) each edge has a single direction and at least some of the edges each represent a one-way effect of one node upon another node, and
- (a4) a metadata object is a data element or a transformation;
- (b) generating instances of a data structure representing the portion of metadata, the instances including at least a first instance and a second instance;
- (c) storing the instances of the data structure in random access memory (RAM);
- (d) receiving a query for lineage metadata,
 - (d1) said query including an identification of at least one particular element of data, and
 - (d2) wherein the query includes an identification of a type of lineage and a walk plan;
- in response to receiving the query:
- (e) traversing the data structure in accordance with the walk plan to collect lineage metadata stored in the data structure that is responsive to the query,
- (f) generating a response to the query based on the data stored using the data structure, the response including lineage metadata responsive to the query, where the lineage metadata
 - (f1) describes a sequence of nodes and edges, one of the nodes representing the particular element of data,
 - (f2) includes data representing the other node associated with the second identifier; and
- (g) sending the response containing the lineage metadata to a computer system for causing display of a lineage diagram generated based on the lineage metadata,

- (g1) wherein the relationship among the nodes shown in the lineage diagram corresponds to the edge represented by the pointer.

Claim 1 further specifies more details about these main steps.

3.2 In particular, with regard to steps (b) and (c), claim 1 further specifies that at least a first instance of the instances of the data structure includes:

- (c1) an identification value that identifies a node that corresponds to the first instance ("first-instance node"),
- (c2) property value(s) representing respective properties of the corresponding node, and
- (c3) a pointer that includes a reference to a memory location associated with a portion of the RAM, the portion of the RAM storing a second instance of the instances of the data structure that corresponds to another node, the pointer representing an edge associated with the other node that corresponds to the second instance ("second-instance node"), wherein
- (c4) the edge represents an effect the first-instance node has on the second-instance node, or an effect the second-instance node has on the first-instance node.

3.3 Claim 1 further specifies, with regard to feature (d2), that:

- (d3) the walk plan identifies which types of nodes and edges are relevant to the identified type of lineage, and
- (d4) the walk plan includes conditions for following an edge or for collecting an edge or an instance of the data structure based on one or more

property values representing respective properties of a corresponding node, wherein an edge is followed by accessing a memory location identified by a pointer representing the followed edge, and wherein an instance of a data structure is collected by adding data of the collected instance to lineage metadata to be returned in response to the query, and wherein an edge is collected by adding data representing the collected edge to lineage metadata to be returned in response to the query.

- 3.4 Step (e) of traversing the data structure is further defined as including:
- (e1) accessing data stored using the data structure and a first identifier that corresponds to the particular element of data identified by the query, including accessing the first instance of the data structure identified by the first identifier;
 - (e2) accessing the pointer of the first instance;
 - (e3) accessing the second instance identified by a second identifier, the second identifier being stored at the memory location referenced by the pointer of the first instance;
 - (e4) collecting data of the second instance.

4. *Articles 84 and 123(2) EPC*

- 4.1 Throughout the examination proceedings, the examining division did not raise any objections for lack of clarity or lack of support of the claims. The board is satisfied that the deficiencies identified in the appeal proceedings were overcome by amendment and that the claims of the main request meet the requirements of Article 84 EPC.

4.2 The subject-matter of claim 1 of the main request is directly and unambiguously derivable from the original claims 1 to 6 and 9 to 10 and the following passages of the description of the application as filed: page 9, first full paragraph; page 15, last paragraph, to page 16, line 2; page 16, lines 24 to 30; and page 20, lines 12 to 18. Therefore, claim 1 of the main request fulfils the requirements of Article 123(2) EPC.

4.3 Claims 2 to 4 correspond to original claims 7, 8 and 12. Claim 5 specifies a system by reference to claims 1 to 4, where the system includes at least one non-transitory computer-readable storage medium and one or more processors. It is based on original claim 13. Claim 6 specifies a non-transitory computer-readable storage device by reference to claims 1 to 4 and finds a basis in original claim 14. Therefore, claims 2 to 6 also meet the requirements of Article 123(2) EPC.

5. *Article 56 EPC*

5.1 In the decision under appeal, the examining division decided that the method of claim 1 defined "an abstract **graph model** of data lineage describing a network of nodes and edges, in the form of a customized query processing formulation" which was "void of any further technical considerations". The only features of claim 1 which were technical were the features "storing", "random access", "computer", and "for causing a display of the computer system". The application did not describe "any technical interaction between the apparent non-technical features and the technical features", besides the use of a computer to perform the method.

The examining division was of the opinion that no technical effect serving a technical purpose could be derived from the claim wording. The description disclosed non-technical purposes of the metadata, e.g. finding out the meaning of business terms, the relationship between those terms and the data to which the terms referred. These were non-technical aspects of an administrative task. In the context of general-purpose computers, pointers referencing memory locations inevitably had to be used. They were thus regarded as an integral part of the general purpose computer.

- 5.2 The appellant argued that the "problem-solution-approach" exercised by the examining division in points 11.2.20 to 11.2.27 of the decision under appeal was intrinsically biased and based on hindsight, as the objective technical problem included the solution. The examining division's assessment that several claim features were non-technical resulted from an incorrect legal approach for assessing the technical character and from an incorrect understanding of the claimed features. This assessment was arbitrary and involved an artificial, hypothetical separation of features that were actually claimed together. Claim 1 specified an instance of a data structure stored in RAM that included a pointer/reference to a further instance of the data structure stored in RAM as well. The pointer established a reference between two portions of RAM. It was not any pointer but rather a specific pointer representing the edge between the two nodes, i.e., representing an effect one node had on another node. All the features of claim 1 were based on technical considerations and made a technical contribution.

The appellant further argued that even commonplace or generic computer components could be involved in making a technical contribution. The examining division had failed to acknowledge that the computer components of the claimed system did, in fact, make a technical contribution. The examining division had not provided any evidence of the implicit presence of the instances of the data structures in a general purpose computer and of the alleged common general knowledge. The cited prior art did not disclose the claimed pointer and instances of the data structure.

5.3 The board agrees with the decision under appeal that the lineage data itself should not be considered "a technical state of the underlying hardware", as argued by the appellant, or as "data about a technical process" or "conditions prevailing in an apparatus or system" within the meaning of decisions T 1670/07, T 115/85 and T 756/06 cited by the appellant. A "data element" and a "transformation" are not further specified in the claim and can thus not be seen as technical entities (see also T 1670/07, Reasons 12; T 756/06, Reasons 13), except for the fact that they are stored in computer memory. The display of lineage information thus corresponds to presentation of information as such under Article 52(2)(d) and (3) EPC and does not make a technical contribution.

5.4 However, the fact that the ultimate goal of the claimed method is not technical is not sufficient to conclude that the whole implementation is not technical (T 619/02, Reasons 2.1). The board agrees with the appellant that the reasoning of the decision under appeal is not convincing.

5.4.1 In the decision under appeal, the examining division identified the computer programmer as the skilled person (point 11.2.24) and described the "expert in graph models" as a non-technical expert who defines the set of requirement specifications (point 11.2.26). The examining division argued that the abstract graph model was given "to the skilled person who will use a computer to implement it without the use of any 'further technical considerations', let alone technical considerations of the internal functioning of the computer" (point 11.2.18). However, in the board's opinion, in the conventional problem-solution approach as further developed by decision T 641/00 (COMVIK approach), the skilled person solves the objective technical problem by technical means based on technical considerations. If that is not the case, either the skilled person is inaccurately defined or the non-technical features were not added to the formulation of the technical problem to be solved. According to the COMVIK approach, the skilled person is a fictional person skilled in a technical field, who has the task of technically implementing the non-technical requirements passed on to them as part of the technical problem to be solved. The skilled person searches a technical solution based on their ordinary technical skills, common general knowledge and knowledge of the prior art. Each claim feature, or each aspect of a claim feature, is either a contribution of the non-technical expert, e.g. an expert on graph models, in which case it can appear in the formulation of the technical problem to be solved, or a contribution of the technical expert, in which case it is part of the technical solution (see also decision T 2626/18, Reasons 4.13).

5.4.2 The same comment applies to the examining division's argument that "When automating the method on the computer system comprising a query processing method the computer programmer does not have to overcome any technical problem, commonplace programming skills and computer knowledge will suffice." (point 11.2.27).

In addition, this statement confuses commonplace features and/or obvious solutions with non-technical subject-matter by expressing that the programmer, who was identified as the technically skilled person, does not have to overcome any technical problem because only commonplace programming skills and computer knowledge are necessary.

It is true that since computer programming involves technical and non-technical aspects, it is difficult to distinguish between the "programmer as such" who, as long as they only develop abstract algorithms, are not a skilled person within the meaning of the case law, and the "technical programmer" (see also T 697/17, reasons 5.2.4). But in the decision under appeal, the programmer was identified as the skilled person who receives the objective technical problem.

5.4.3 The decision under appeal did not take into account all the claim features making a technical contribution.

Citing decision G 1/19 of the Enlarged Board of Appeal, the examining division argued correctly that "merely" performing a method, the result of which did not cause any "technical effect(s) on a physical entity in the real world", did not suffice to contribute to the technical character of the invention, and that "the mere calculation of the behaviour of a (technical) system as it exists on the computer, and the numerical output of such calculation, should not be equated with

a technical effect" (see point 11.2.16). The board notes however, that according to decision G 1/19, "technical effects" or "technical interactions" do not occur only through the generation of "technical output". Technical contributions can result, for example, from "adaptations to the computer or its operation, which result in technical effects (e.g. better use of storage capacity or bandwidth)" and "technical effects can occur within the computer-implemented process (e.g. by specific adaptations of the computer or of data transfer or storage mechanisms)" (see pages 39 and 40, points c, 85 and 86).

Under point 14.2.1 of the decision under appeal, the examining division argued that "pointers referencing memory locations inevitably have to be used, therefore, they are regarded as an integral part of the general purpose computer that is always implicitly comprised within the general-purpose computer and its usage. Therefore, the reasoning of the applicant that the data structure with the pointer as claimed is a **specific implementation** is not convincing."

The board does not agree with this argument. In the assessment of technical contribution and inventive step, the claim should not be analysed as a collection of disconnected terms but as a whole (see e.g. T 731/17, Reasons 6.2 to 6.4; T 1272/20, Reasons 3.1). Even though a pointer to data in computer memory is commonplace, its purposive use in a method for retrieving data from computer memory makes a technical contribution (see e.g. T 1351/04, Reasons 7.2 to 7.4; T 697/17, Reasons 5.2.5; T 3176/19, Reasons 10.3) and cannot be ignored in the inventive-step assessment.

The data structure specified in the claim serves the technical purpose of providing access to the data stored in memory, as claimed for example in features (c3), (e) and (f) and thus makes a technical contribution.

- 5.5 In its preliminary opinion, the board introduced prior-art documents D2 to D9 into the appeal proceedings. Of all the cited prior art, document D2 is the best starting point for assessing inventive step and closest prior art. It discloses a metadata viewing environment which displays a data lineage diagram (paragraph [0028]). The system receives metadata from data sources, the metadata including metadata objects (paragraphs [0017], [0023], [0024] and [0027]). The metadata objects can represent different types of data elements (e.g., data used as input or output of an executable program) and/or transformations (for instance, any type of data manipulation associated with a data processing entity). The metadata objects are represented as nodes in the diagram. The system can automatically extract relationships (i.e. lineage information) between the metadata objects and compute the lineage information (paragraph [0027]). The computed lineage information corresponds to a data structure according to feature (b) of claim 1. In the board's opinion, it is implicit that the instances of the data structure are stored at least temporarily in RAM. Since the nodes represent data elements and transformations, and are connected by edges representing their relationships, an edge between two nodes represents a one-way effect as in feature (a3). Therefore, document D1 discloses a method comprising features (a) to (c), (a1) to (a4) and (f1).

5.6 The subject-matter of claim 1 differs from the method of D2 in that it includes features (c1) to (f), (f2), (g) and (g1). These features specify details of the data structure (features (c1) to (c4)), the step of receiving a query for lineage metadata including the identification of a data element, a type of lineage and a walk plan (features (d) to (d2)), the steps of traversing the data structure and collecting data using the data structure and the walk plan (features (d3), (d4) and (e) to (e4)), the step of generating a response including lineage metadata responsive to the query ((f) and (f2)) and the steps of sending the response to a computer system and displaying the lineage data (features (g) and (g1)).

5.7 The appellant argued that in the system of document D2 the metadata was not kept in RAM but in persistent storage. The walk plan instructed the computer how to create a lineage diagram according to the selection of the user and was not a part of said selection. By using a walk plan in the traversal of the data structure, data which was not responsive to the query was not collected. The distinguishing features collectively enhanced speed and computational efficiency when provisioning lineage data, thereby addressing a technical problem that was not solved by the prior art.

According to the description, the walk plan can be selected by the metadata server using the identity of the data element or taking into account the type of lineage (paragraph bridging pages 6 and 7; page 20, lines 12 to 18).

5.8 The board agrees with the appellant that creating the specific data structure of claim 1 and generating lineage metadata in RAM, which are not disclosed in document D2, contribute to a more efficient generation

of the lineage metadata. However, loading data into RAM for more efficient processing is well-known to the skilled person, for example from in-memory databases, as the board explained at the oral proceedings. In addition, not all the distinguishing features are directed to improve efficiency.

- 5.9 In the board's opinion, obtaining and displaying lineage information for a particular data element are non-technical requirements. The distinguishing features solve the technical problem of supporting in the system of document D2 the functionality for obtaining and displaying, in a computer system, lineage metadata of a given type for a particular element.
- 5.10 None of the cited prior-art documents disclose the combination of distinguishing features (c1) to (f), (f2), (g) and (g1). In the board's opinion, it would be within the ordinary skills of the computer expert to arrive at the data structure specified in the distinguishing features, which corresponds directly to the non-technical lineage structure. However, the board is not convinced that the skilled person would arrive at the combination of all the distinguishing features, including a walk plan to direct the way the data structure is traversed and the data is collected as claimed.
- 5.11 Therefore, the subject-matter of claim 1 of the main request involves an inventive step (Article 56 EPC).
- 5.12 The same applies to dependent claims 2 to 4 of the main request and to claims 5 and 6, which define the claimed subject-matter by reference to claims 1 to 4. Therefore, claims 2 to 6 fulfil the requirements of Article 56 EPC.

6. *Conclusion*

6.1 The board is satisfied that the claims fulfil the requirements of the EPC. However, the description may need to be adapted. Therefore, the case is to be remitted to the examining division with the order to grant a patent on the basis of the claims of the main request, drawing pages as published and description pages to be adapted.

Order

For these reasons it is decided that:

1. The decision under appeal is set aside.
2. The case is remitted to the examining division with the order to grant a patent based on:
 - claims 1 to 6 of the main request corresponding to the revised new auxiliary request V filed during the oral proceedings before the board,
 - a description to be adapted and
 - drawing pages 1/13 to 13/13 as published.

The Registrar:

The Chairman:



M. Schalow

J. Geschwind

Decision electronically authenticated