

Internal distribution code:

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

**Datasheet for the decision
of 29 June 2021**

Case Number: T 2863/18 - 3.5.07

Application Number: 05766079.7

Publication Number: 1759306

IPC: G06F17/24

Language of the proceedings: EN

Title of invention:

PROCEDURALLY EXPRESSING GRAPHIC OBJECTS FOR WEB PAGES

Applicant:

Apple Inc.

Headword:

Procedurally expressing graphic objects/APPLE

Relevant legal provisions:

EPC Art. 56

RPBA Art. 12(4)

Keyword:

Claims - main request - inventive step - (no)

Admissibility - first and second auxiliary requests - not admitted



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
Fax +49 (0)89 2399-4465

Case Number: T 2863/18 - 3.5.07

D E C I S I O N
of Technical Board of Appeal 3.5.07
of 29 June 2021

Appellant: Apple Inc.
(Applicant) One Apple Park Way
Cupertino CA 95014 (US)

Representative: Lang, Johannes
Bardehle Pagenberg Partnerschaft mbB
Patentanwälte, Rechtsanwälte
Prinzregentenplatz 7
81675 München (DE)

Decision under appeal: **Decision of the Examining Division of the
European Patent Office posted on 19 June 2018
refusing European patent application No.
05766079.7 pursuant to Article 97(2) EPC.**

Composition of the Board:

Chairman J. Geschwind
Members: C. Barel-Faucheux
R. de Man

Summary of Facts and Submissions

- I. The appeal lies from the decision of the examining division to refuse European patent application No. 05766079.7, filed as international application PCT/US2005/022152 and published as WO 2006/012168 A2.
- II. The documents cited in a communication dated 12 June 2018 on which the decision, in the form of a decision "according to the state of the file", is based included:
- D1: "Objects, Images, and Applets", W3C Recommendation, 10 February 2003, retrieved from the Internet on 13 April 2006, <http://web.archive.org/web/20030210154019/http://www.w3.org/TR/REC-html40/struct/objects.html>
- D2: Christian Ullenboom, "Java ist auch eine Insel", 2003, retrieved from the Internet on 13 April 2006, http://www.galileocomputing.de/openbook/javainsel2/java_140000.htm#Xxx998138
- D3: Marx et Al, "JSP best practices", 1 November 2001, retrieved from the Internet on 15 October 2003, <http://www.javaworld.com/javaworld/jw-11-2001/jw-1130-jsp.html>
- III. In the communication dated 12 June 2018, the examining division informed the appellant that the subject-matter of claims 1, 17 and 23 of the sole request did not involve an inventive step within the meaning of Article 56 EPC having regard to document D1 in combination with document D2, in light also of the possibility to include the procedural language in the same file instead of loading it from a separate applet file disclosed in document D3.

- IV. By letter of 15 June 2018, the appellant requested a decision "on the basis of the documents presently on file" and informed the examining division that it would not attend the oral proceedings scheduled on 19 June 2018.
- V. With the statement of grounds of appeal, the appellant filed again the sole request subject to the decision under appeal as a main request and a first and second auxiliary request. It requested that the decision under appeal be set aside and that the application be remitted for grant or further prosecution on the basis of the main request or an auxiliary request.
- VI. In a communication according to Article 15(1) RPBA, the board expressed its preliminary opinion that claim 1 of the main request lacked clarity (Article 84 EPC) since the expression "arbitrary graphics object" was used in the application in a non-generally accepted meaning not further defined in the claim.

The board was of the preliminary opinion that claim 1 of the main request lacked inventive step over document D1 in combination with document D3 (Article 56 EPC).

Furthermore, the board explained why it was inclined not to admit the first and second auxiliary requests into the procedure. It was also of the preliminary opinion that claim 1 of the first and second auxiliary requests lacked inventive step.

- VII. In reply to the board's communication, the appellant explained why it considered the subject-matter of claim 1 to be clear and provided further arguments in favour of inventive step for claim 1 of the main (and first and second auxiliary) request(s). It did not, however,

expressly provide arguments in favour of the admissibility of the first and second auxiliary requests.

VIII. Oral proceedings were held as a videoconference. At the end of the oral proceedings, the Chair pronounced the board's decision.

IX. The appellant's final requests were that the decision under appeal be set aside and that a patent be granted on the basis of the claims of the main request or, in the alternative, the first or second auxiliary request, all requests as filed with the statement of grounds of appeal.

X. Claim 1 of the main request reads as follows (itemisation added by the board):

"A computer-implemented method of drawing an arbitrary graphics object in a web page (140), the method characterized by comprising:

(A) accessing a file comprising coded markup language that specifies a drawing space (210) as an extent within the web page (140) and coded procedural language that specifies a drawing command to draw the arbitrary graphics object in the drawing space (210) in the web page (140);

(B) rendering the drawing space (210) within the web page (140) using the coded markup language; and

(C) drawing the arbitrary graphics object into the drawing space (210) within the web page (140) using the coded procedural language."

XI. The appellant's arguments, where relevant to this decision, are addressed in detail below.

Reasons for the Decision

The application

1. The application relates to "procedurally" expressing arbitrary graphic objects (AGO) in markup language documents (paragraph [0002] of the published application).
- 1.1 A drawing area in which anything can be drawn using drawing commands is described in a markup language. The AGO is then expressed in the form of arbitrary drawing commands, such as those provided in vector graphics, to draw in the drawing area. A markup language, such as HTML, is used to specify a "canvas". A procedural language such as JavaScript is used to draw in that canvas. Any graphics language can be used to specify the graphical content to be drawn in the element or canvas; such language can include vector graphics commands such as pathing, stroking and filling. The canvas itself may also be manipulated in terms of other markup constructs such as Content Style Sheets (CSS). During an event loop, the procedural commands are translated into graphics code, which is executed to dynamically generate the graphics object. The object is then composited for display (paragraph [0007]; paragraph [0021] and Figure 2).

A. Main request

Claim interpretation

2. The application uses the expression "arbitrary graphics object" to designate "any graphical output rendered procedurally, including, but not limited to, a two or

three dimensional image or scene, produced based on the execution of procedural commands" (paragraph [0006]). The AGO is expressed in the form of arbitrary drawing commands. The execution of the commands may be carried out in a graphics context that supports vector graphics, Scalable Vector Graphics, OpenGL or other types of existing and "emerging" graphics platforms, or may also utilise more conventional graphics formats such as Postscript, TIFF, PDF, PICT, BMP, WMF, GIF, JPEG, PNG and EPS (paragraphs [0006] and [0007]).

- 2.1.1 For the purpose of assessing inventive step, the board will interpret the expression "arbitrary graphics object" in accordance with these explanations.

3. *Inventive step*

- 3.1 Document D1 has been considered the closest prior art by the examining division. This has not been disputed by the appellant.

- 3.1.1 Document D1 discloses how to insert an "object" (applets, plug-ins, media handlers, etc.) in HTML documents (page 2, first paragraph).

The object element comprises, among other attributes, a "data" attribute which "may be used to specify the location of the object's data, for instance image data for objects defining images, or more generally, a serialized form of an object which can be used to recreate it" (section 13.3 on page 4; page 5, second full paragraph). The "Inline vs. external data" example at the bottom of page 7 shows that the "data" attribute may in fact be used to include a serialised form of an object (e.g. an applet) directly in an HTML document.

The object element also comprises a "type" attribute which specifies the content type for the data specified by data (page 5, third full paragraph). D1 explains that "[m]ost user agents have built-in mechanisms for rendering common data types such as text, GIF images, colors, fonts, and a handful of graphic elements. To render data types they don't support natively, user agents generally run external applications" (page 5, last paragraph; page 6, second paragraph).

- 3.1.2 D1 gives an example of inserting an analogue clock applet written in the Python language in a document via the object element (page 6, last paragraph). A "classid" attribute of the "object" element specifies the location of the applet via a URI (page 4, last paragraph; page 6, last paragraph). Parameters like "height" and "width" may be added to the object (page 9, first paragraph). D1 discloses that the "width" attribute specifies the initial width, while the "height" attribute specifies the initial height, of the applet's display area (page 12, third full paragraph).

It is well known to the skilled person that "procedural" programming is based on the concept of procedure calls, where the procedures (a type of routine or subroutine) contain a series of computational steps to be carried out. The Python language is well known to support procedural programming and is therefore a "procedural language".

- 3.1.3 Thus, D1 discloses a computer-implemented method of drawing an arbitrary graphics object in a web page (e.g. the clock applet written in the Python language) as well as feature (A). The "width" and "height" attributes specify the initial width and height of the

applet's display area, which can be seen as the drawing space.

- 3.2 In the appealed decision, the examining division had identified a drawing space as a first distinguishing feature of claim 1 with regard to document D1 (the underlined part), as well as the feature that the file also comprises "coded procedural language that specifies a drawing command to draw the arbitrary graphics object in the drawing space in the web page" in combination with the step of "drawing the arbitrary graphics object into the drawing space within the web page using the coded procedural language" (feature (C)).
 - 3.2.1 An HTML object having "height" and "width" parameters and containing an applet, for example, an analogue clock applet, written in Python language, in an HTML document as described in D1 is considered by the board to define the drawing space of an object for drawing an arbitrary graphics object coded in a procedural language.
- 3.3 The appellant further argued that "even if the skilled person were to assume that the downloaded programs [note from the board: the 'applet'] include procedural language, it is submitted that automatically downloading a program at run time does not constitute 'including' the procedural language of the program within the same file as the markup language that references the program as stipulated by the independent claims of the main request" (statement of grounds, page 6, item (bb)). Furthermore, the "classid" attribute of the object element of D1 specified a location of an object's implementation via a URI; even if a skilled person were to assume, based on this disclosure, that

the URI pointed to code that contained procedural language, the referenced code was still not included in the same file as the markup language file that contained the URI to the code (statement of grounds, page 6, item (cc)).

- 3.3.1 However, as pointed out in point 3.1.1 above, the "data" attribute of the object element of D1 should also be taken into account (page 4, section 13.3). This attribute may be used to "specify the location of the object's data, for instance image data for objects defining images, or more generally, a serialized form of an object which can be used to recreate it" (page 5, second full paragraph). The example given at the bottom of page 7 ("Inline vs. external data") shows that the "data" attribute can be used to include the code of an object directly in the HTML file in serialised form.
- 3.3.2 The board is therefore of the opinion that document D1 discloses, as one option, that the HTML file comprises the procedural applet code.
- 3.4 D1 does not explicitly disclose the rendering and drawing steps of features (B) and (C).

However, once the file of feature (A) - comprising coded markup language that specifies a drawing space (210) as an extent within the web page (140) and coded procedural language that specifies a drawing command to draw the arbitrary graphics object in the drawing space (210) in the web page (140) - has been accessed, it is immediately obvious to the skilled person that the corresponding web page should be displayed, which includes rendering the drawing space and drawing the graphics object.

The coded markup language that specifies a drawing space (210) as an extent within the web page (140) will directly enable the skilled person to implement the step of "rendering the drawing space (210) within the web page (140) using the coded markup language" (feature (B)), while the coded procedural language that specifies a drawing command to draw the arbitrary graphics object in the drawing space (210) in the web page (140) will directly enable the skilled person to implement the step of "drawing the arbitrary graphics object into the drawing space (210) within the web page (140) using the coded procedural language" (feature (C)).

Contrary to the appellant's argument submitted during the oral proceedings, the board does not consider that it uses "hindsight" by stating that it is obvious for the skilled person to use the information in the accessed file to implement the rendering step (B) and the drawing step (C).

- 3.5 With respect to the appellant's argument concerning the alleged technical effect of the current invention to dynamically generate the arbitrary graphics object "in a resolution-independent and scalable form", "while maintaining the same image quality", and "often consuming less space than conventional graphics elements" (letter of reply dated 14 June 2021, points 1.d, 2 and 3), the board considers that this effect is not apparent from claim 1 of the main request. The skilled person is aware that these properties of being resolution-independent and scalable are properties of vector graphics.

3.6 Therefore, the board is of the opinion that claim 1 of the main request lacks inventive step over document D1 (Article 56 EPC).

B. First and second auxiliary requests

4. Admissibility

4.1 The first and second auxiliary requests were filed with the statement of grounds.

4.2 The board has the discretion to hold inadmissible facts, evidence or requests which could have been presented in the first-instance proceedings (Article 12(4) RPBA 2007).

4.3 In response to the clarity and inventive step objections raised by the examining division in the communication dated 1 September 2017 accompanying the summons to attend oral proceedings, the appellant found it appropriate to file a single, amended set of claims on 18 May 2018.

4.4 After receipt of the examining division's communication dated 12 June 2018 maintaining its inventive step objection for claim 1 of the amended set of claims dated 18 May 2018, the appellant decided not to attend the scheduled oral proceedings before the examining division and requested a decision "on the basis of the documents presently on file, in particular the requests on file of May 18, 2018".

4.5 In the board's view, the sets of claims of the first and second auxiliary requests could have already been filed on 18 May 2018 in reply to the examining division's summons to oral proceedings.

- 4.6 During the oral proceedings before the board, the appellant argued only that the claims of the first and second auxiliary request submitted with the statement of grounds had been filed for clarifying the expression "arbitrary graphics object", as an anticipation to the board's communication, and should therefore be seen as a reaction to the clarity objection raised for the first time in the board's communication.
- 4.7 Claim 1 of the first auxiliary request additionally defines, in comparison with claim 1 of the main request, that the drawing command comprises a vector graphics or OpenGL command. This corresponds to the additional features of claim 16 as originally filed with regard to claim 1 as originally filed.
- 4.8 Claim 1 of the second auxiliary request additionally defines, in comparison with claim 1 of the first auxiliary request, the feature "wherein rendering the drawing command comprises translating the drawing command into graphics code and executing the graphics code to generate the arbitrary graphics object" at the end of the claim.
- 4.9 Thus, these amendments further specify the "drawing command" of feature (A) for drawing the AGO but do not clarify the definition of the "arbitrary graphics object" per se. The appellant's only argument in support of the admissibility of the first and second auxiliary requests therefore cannot convince the board.
- 4.10 The features "a drawing command within the drawing space (210) in the web page (140)" and "drawing the arbitrary graphics object into the drawing space (210) using the coded procedural language" had been

identified by the examining division as distinguishing features in its communication annexed to the summons to oral proceedings.

Therefore, the board is of the opinion that specifying the drawing command could have been an option for the appellant, in reply to the summons to oral proceedings, for reacting to the lack of inventive step objection against claim 1 of the main request raised in this communication.

4.11 Moreover, the board is of the opinion that these amendments *prima facie* do not overcome the lack of inventive step objection since vector graphics and OpenGL commands are obvious options for drawing objects in a drawing space providing the advantage of being resolution-independent and scalable, while rendering the drawing command by translating the drawing command into graphics code and executing the graphics code to generate the arbitrary graphics object is also obvious.

4.12 Therefore, these requests are not admitted into the proceedings.

Order

For these reasons it is decided that:

The appeal is dismissed.

The Registrar:

The Chairman:



T. Buschek

J. Geschwind

Decision electronically authenticated