Internal distribution code:

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

Datasheet for the decision of 12 June 2023

Case Number: T 0480/18 - 3.5.06

Application Number: 03075821.3

Publication Number: 1339222

IPC: G06F9/455, H04N7/173, G06F9/46

Language of the proceedings: EN

Title of invention:
Multithread data processor

Applicant:
Thomson Licensing

Headword:
Multithread data processor/THOMSON LICENSING

Relevant legal provisions:
EPC 1973 Art. 56

Keyword:
Inventive step - (no)

Decisions cited:

Catchword:

Europäisches Patentamt
European Patent Office
Office européen des brevets

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 0480/18 - 3.5.06

D E C I S I O N
of Technical Board of Appeal 3.5.06
of 12 June 2023

Appellant: (Applicant)	Thomson Licensing 1-5, rue Jeanne d'Arc 92130 Issy-les-Moulineaux (FR)
Representative:	Morain, David InterDigital CE Patent Holdings 20, rue Rouget de Lisle 92130 Issy-les-Moulineaux (FR)
Decision under appeal:	**Decision of the Examining Division of the European Patent Office posted on 6 October 2017 refusing European patent application No. 03075821.3 pursuant to Article 97(2) EPC.**

Composition of the Board:

Chairman	M. Müller
Members:	A. Teale
	K. Kerber-Zubrzycka

Summary of Facts and Submissions

I. This is an appeal against the decision, dispatched with reasons on 6 October 2017, refusing European patent application No. 03 075 821.3 on the basis that the subject-matter of claim 1 according to a main and an auxiliary request lacked inventive step, Article 56 EPC, in view of the combination of D1 and D2, these documents being:

D1: W.E. Wall, "An Advanced Video Platform for the Cable Industry", Proceedings of the Fourth International Workshop on Community Networking Processing, 11-12 September 1997, Atlanta, GA, USA, pages 31-33, XP002061359.

D2: G. Hirtz, R. Zwing, "Open TV: Betriebssystem für interaktives Fernsehen", FKT Fernseh- und Kinotechnik, Fachverlag Schiele & Schon GmbH, Berlin, DE, vol. 50, no. 3, 1 March 1996, pages 84-89, ISSN: 1430-9947, XP000581417.

II. A notice of appeal and the appeal fee were received on 15 December 2017.

III. In a statement of grounds of appeal, received on 2 February 2018, the appellant requested that the decision be set aside and that the main request be granted. The appellant also made an auxiliary request for oral proceedings.

IV. In an annex to a summons to oral proceedings the board set out its preliminary view on the appeal, stating *inter alia* that the subject-matter of claim 1 seemed to lack inventive step, Article 56 EPC 1973, in view of

the combination of D1 and D2 and that the claim seemed to be unclear, Article 84 EPC 1973.

V. The appellant did not comment in substance on the board's preliminary opinion. Instead, in a letter received on 22 March 2023, they stated that they would not attend the oral proceedings. The board subsequently cancelled the oral proceedings and announced in a communication via its registry dated 23 March 2023 that it intended to issue a written decision at the end of May 2023.

VI. The application is being considered in the following form:

Description:
pages 1 to 20, as originally filed.

Claims:
1 to 11, received as main request on 25 August 2017.

Drawings:
Pages 1/7 to 7/7, as originally filed.

VII. Claim 1 reads as follows:

"An apparatus for processing digital audio-visual data, comprising a data processing system including a virtual machine adapted to, inter alia, receive at least one code file written in a first or in a second interpretative language and downloaded via one or more hardware devices of said apparatus, said virtual machine being adapted to perform an evaluation of code of said received file so as to distinguish between a code file written in said first interpretative language and a code file written in said second interpretative

language in dependence on the structure of the code of the received file and to pass said code of the received file, for interpretation and execution, to a first interpreter means adapted to interpret code written in said first interpretative language or to a second interpreter means adapted to interpret code written in said second interpretative language, depending on said evaluation."

Reasons for the Decision

1. Admissibility of the appeal

In view of the facts set out at points I to III above, the appeal fulfills the admissibility requirements under the EPC and is consequently admissible.

2. Summary of the invention

2.1 The application relates to a data processor for use as an MPEG-2 decoder for digital audio-visual data in a digital television receiver system. The data processor comprises a virtual machine for processing events stored in an input queue.

2.2 The invention addresses the problem of the language used to write the application code running on the virtual machine. This code, which may be written in a variety of languages, is downloaded to the receiver from a broadcast centre and interpreted by an interpreter in the virtual machine; see page 2, lines 13 to 19 and figure 4; 4277. If the decoder can only handle one such language, then it may have to download several versions of an application in order to obtain the

language version that it can handle. This constitutes an inefficient use of bandwidth.

2.3 Hence it is desirable that the decoder be able to process applications written in two different languages, namely Java bytecode and m-code, "a proprietary code developed by the applicants" (see page 14, lines 22 to 26)(potentially a form of compiled Modula), or Java bytecode and p-code (understood to be compiled Pascal); see figure 5; 4278, 4279 and page 14, line 33, to page 15, line 4.

2.4 The invention solves the above problem by distinguishing between code written in two languages and passing code to the appropriate interpreter for the recognised language.

2.5 Figure 1 (see page 8, lines 8 to 36) illustrates a digital television system using MPEG-2 compression, the figure showing the transmitter side (2002-2008), an uplink (2012), satellite transponder (2014), downlink (2016) and "Earth receiver" (2018), such as a satellite dish. The received signals pass via an integrated receiver/decoder (2022) which decodes the compressed MPEG-2 signal into a television signal for display on a television set (2022). The system allows for "interactive television" (4000) in the sense that signals from the receiver/decoder can pass via a "modemmed back-channel" (4002) to the multiplexer/scrambler (2004) in the broadcast centre.

2.6 As shown in figure 2 (see page 9, line 8, to page 10, line 4), applications are created and tested at the broadcast centre using an authoring tool (4004) before being stored on an application and data server (4006) for transmission in the MPEG-2 transport stream to the

receiver/decoder (2020). Here the applications are executed in the data processing system (4008) of the receiver/decoder by a virtual machine with a run time engine (RTE) implemented as executable code installed on the receiver/decoder.

2.7 Figure 3 (see page 10, line 6, to page 11, line 11) shows the layered software architecture in the receiver/decoder. A low level operating system (OS) forms the bottom layer (4100). Event messages pass from the operating system to the middleware layer (4200). This comprises a virtual machine (4250) and, on top of that, a number of interfaces. On top of the middle layer there is the application interface (AP) layer (4300) comprising "packages" (4310-4315) written in an object-oriented interpretative language, such as Java; see page 11, lines 14 to 16. The packages provide an interface between applications created by the service provider, such as an interactive program guide, teleshopping and an internet browser, and the virtual machine (4250). The details of the Java class libraries of the API packages are set out from page 11, line 13, to page 12, line 12.

2.8 The structure of the virtual machine is illustrated in figure 4; see page 13, line 22, to page 14, line 30. The virtual machine comprises *inter alia* an interpretation package (4277-79), comprising a bytecode interpretation service (4278), and an "m-code" or "p-code" interpretation service (4279), meaning that the interpreter can handle applications written in either Java bytecode or m-code or Java bytecode or p-code; see page 14, lines 22 to 26, and figure 5.

2.9 As shown in figure 5, files arriving in the system, i.e. bytecode classes or p-code/m-code modules, are

evaluated by the module class manager (4500) based on their structure to distinguish between Java bytecode and p-code/m-code either based on a header or the file name; see page 15, lines 14 to 16.

3. Clarity, Article 84 EPC 1973

3.1 The expression "interpretative language", present in claims 1 and 4, is used in the application with two different meanings. According to page 11, lines 14 to 16, the packages (4310-4315) in the application interface (AP) layer (4300) are written in an object-oriented "interpretative" language, such as Java. In contrast, page 15, lines 18 to 27, refers to both a Java bytecode interpreter (4278) and an m-code interpreter (4278). Hence the board has considered (see point 4.1 below) whether an "interpretative language" refers to a language such as Java or to the result of its compilation, namely Java bytecode.

3.2 Nevertheless, despite the doubts regarding clarity expressed in the summons, the board finds that the claims are sufficiently clear for the assessment of inventive step.

4. The board's understanding of the invention

4.1 The meaning of the expression "interpretative language"

4.1.1 The applicant argued that HTML (HyperText Markup Language) is not an "interpretative language". The examining division disagreed, arguing that HTML had to be parsed and interpreted before it could be presented; see reasons 1.7.

4.1.2 On appeal the appellant has argued that HTML is not an interpretative language but a markup language, so that code written in HTML did not qualify as "code written in an interpretative language", as it did not define an application by means of an instruction set such as Java bytecode. According to the appellant, the skilled person would make a distinction between a descriptive language, such as HTML, and a programming language, including interpreted languages, such as Java (bytecode).

4.1.3 The board finds that being "interpretive" is not a property of code *per se*, but rather indicates that the code is executed by an interpreter, since the same high-level code can be executed either by compiling it, meaning that it is converted into a lower level code before execution, or by interpreting it, meaning that it is executed directly.

4.1.4 The fact that the term "interpretative language" includes object-oriented languages (see original claim 4) does not restrict the broad notion of "interpretative languages" according to claim 1, and specifically does not exclude HTML. In the board's view object-oriented languages need not be Turing-complete programming languages. Hence the expression "interpretative language" according to claim 1 is understood to cover both Java bytecode, which is mentioned in the application, and HTML, which is known from D1; see below.

5. The prior art on file

5.1 Document D1

5.1.1 In the decision, inventive step was assessed starting from D1. The examining division did not accept the

applicant's argument, reiterated on appeal, that D1 (see abstract and section 4, lines 20 to 26) discloses a decoder handling a mixture of HTML and Java. In view of the reference to Java and HTML as "principal authoring environments" (see section 4, lines 19 to 25), the board understands D1 to disclose applications being written in one language only, be it the "native API of the operating system", Java or HTML. Thus HTML and Java are understood to be alternatives.

5.1.2 D1 relates to a set-top box (STB) which establishes a bidirectional - hybrid optical fibre to the STB, coaxial cable from the STB - communication link with a cable TV company. In addition to this giving the STB access to the Internet, video on demand (VOD) services can also be provided; see page 31, right column, lines 3 to 12. A STB can also download and execute applications, such as interactive games, from the network. In the case of a "digital home terminal" (see page 32, section 4), the STB downloads a different application program to support each service, such as web browsing, VOD and games.

5.1.3 The appellant has argued that D1 is a "high level" article on a video platform for the cable industry which does not discuss the architecture of the software in the set-top box. The appellant has also argued that, in addition to the difference features identified in the decision, D1 also does not disclose the following features of claim 1:

i. [the system being adapted to] receive at least one code file written in a first or a second interpretative language and

ii. [evaluating file code] so as to distinguish between a code file written in said first interpre-

tative language and a code file written in said second interpretative language.

5.1.4 In view of its understanding of the expression "interpretative language" (see above), the board is not persuaded that difference feature "i" exists, both HTML and Java qualifying as "interpretative languages". The board does however agree that difference "ii" exists; see the discussion of inventive step below.

5.1.5 The appellant has argued that D1 discloses a single interpreter. The board is however not persuaded, even in the light of the references in D1 to HTML and Java, that it is directly and unambiguously derivable from D1 that the digital home terminal comprises an interpreter.

5.1.6 In the words of claim 1, D1 discloses an apparatus (page 32, section 4, digital home terminal) for processing digital audio-visual data, comprising a data processing system adapted to, *inter alia*, receive at least one code file written in a first interpretative language (HTML/Java) and downloaded via one or more hardware devices of said apparatus.

5.2 Document D2

5.2.1 D2 was cited in the decision (reasons, 1.6) as evidence that the combination of an audio/video receiver and interactive television applications was known in the prior art, applications being downloaded onto the receiver and executed by an interpreter. There was however no explicit mention of a virtual machine, although this was argued to be implicit or at least obvious.

5.2.2 D2 concerns the "OpenTV" operating system which is suitable for a set-top box for providing interactive TV applications, such as video on demand (VOD) (see page 84, section 2), teleshopping, home banking and interactive games; see page 84, right column, lines 15 to 19, and page 85, section 3 and figure 2. The figure shows a digital interactive decoder (DID) which can be integrated into a set-top box (STB), the DID receiving a digital video/audio signal and Open TV data via satellite and also having a back-channel link with the transaction server of the service provider. The DID decodes the OpenTV data to provide, for instance, a selection menu on the TV screen.

5.2.3 According to section 3.2 (see page 85), the OpenTV application runs on the application system layer which is separate from the hardware specific layers, allowing OpenTV to run on hardware from many manufacturers without requiring device drivers; see sentence bridging pages 85 and 86.

5.2.4 Figure 5 illustrates the layer structure of the OpenTV software in which a downloaded application (as opposed to the applications in ROM; see page 88, section 4.2.1, lines 4 to 8) is executed by an interpreter having access to libraries, the libraries running on a micro kernel (understood by the board to be a simple operating system) having hardware device drivers; see page 87, section 4.2. The downloaded applications are written in "O-code", understood by the board to result from compiling C/C++ code; see page 89, section 7, lines 11 to 12. The O-code is executed by the interpreter; see section 4.2.1, lines 15 to 19, and section 4.2.2 "Interpreter Layer".

6. Inventive step, Article 56 EPC 1973

6.1 According to the appealed decision, the subject-matter of claim 1 differs from the disclosure of D1 in that the apparatus further comprised a virtual machine adapted to perform an evaluation of code of said received file so as to distinguish between codes in dependence on the structure of the code of the received file and to pass said code to a first interpreter means adapted to interpret code written in said first interpretative language or to a second interpreter means adapted to interpret code written in said second interpretative language, depending on said evaluation. These features solved the problem of how to implement the home terminal of D1. D1 disclosed an advanced video platform for which applications could be written in languages such as Java or HTML (see abstract and page 32, section 4) or using the native API (Application Programming Interface) of the operating system. D1 provided few details of the internal architecture of the software in the home terminal.

6.2 The skilled person starting from D1 and seeking to implement the data processing system would have looked for implementations of similar advanced video platforms which combined audio-visual processing and software applications and would thus have considered D2. D2 related to the "OpenTV architecture which combined an audio/video receiver and interactive applications (see page 85, section 3.1). D2 provided an overview of the OpenTV Software architecture (see fig. 5) which followed a layered model and comprised *inter alia* download applications and an interpreter for interpreting and executing the code (see page 88, section 4.2.2 "Interpreter Layer"). While D2 did not explicitly mention a virtual machine, it did disclose separating the

application layer from the hardware specific layers to allow the applications to run on different receivers from different manufacturers (see first point of section 3.2). This was also the stated purpose of the virtual machine in the application (see page 10, lines 24-27). The presence of a virtual machine was consequently implicit, or at least obvious, from D2. As the system of D2 received download applications in a single language (O-code), it only required a single interpreter. The system of D1, having multiple languages, would however have required multiple interpreters, one for each language. Hence the difference features were known from D2 or would have been obvious to the skilled person implementing D1 in the light of D2 and common general knowledge.

6.3 As discussed above, the appellant has argued that the subject-matter of claim 1 differs from the disclosure of D1 in two further difference features. The objective technical problem was, as stated in paragraphs [8-10 and 12] of the published application (see original page 2, lines 12 to 31, and page 3, lines 7 to 12), to avoid repeatedly downloading a software application in a broadcast system comprising decoders adapted to receive different codes. In contrast, the objective technical problem used in the decision, namely implementing the home terminal of D1 was vague and unrelated to the difference features over D1. If, for the sake of argument, the skilled person starting from D1 had sought to solve the problem used in the decision (implementing the home terminal of D1), they would have searched for a way of implementing the home terminal of D1 with the open TV architecture known from D2. D2 taught to compile applications written in different languages into a single "pivot" code type called O-code, both the client and the server using the pivot

language. The argument in point 1.8 of the decision, that the skilled person applying D2 would have only implemented part of the teaching of D2 on the client side but not on the server side, was unreasonable. According to the appellant, the skilled person would consequently not have applied the teaching of D2 to D1 at all. If the skilled person starting from D1 had learned anything from D2, it was to compile code of applications written in different languages into O-code before transmitting them to the set-top box of D1, so that the skilled person would have used a single O-code interpreter in the set-top box, as know from D2. Neither D1 nor D2 would have incited the skilled person to add a second interpreter for interpreting another language.

6.4 The appellant has also disputed the argument in point 1.6 of the decision that the use of several languages to write applications necessarily meant using several interpreters. This was not so; a single interpreter could handle more than one language, for instance HTML and Java. As stated above, the board is not persuaded that D1 discloses using a mixture of two application languages, so that the question of a single interpreter able to handle such a mixture does not arise.

6.5 In view of the above analysis of D1, the board finds that the subject-matter of claim 1 differs from the disclosure of D1 in the following features:

a. the data processing system includes a virtual machine;

b. the code file can also be written in a second interpretative language;

c. said virtual machine is adapted to perform an evaluation of code of said received file so as to distinguish between a code file written in said first interpretative language and a code file written in said second interpretative language in dependence on the structure of the code of the received file and

d. the virtual machine is adapted to pass said code of the received file, for interpretation and execution, to a first interpreter means adapted to interpret code written in said first interpretative language or to a second interpreter means adapted to interpret code written in said second interpretative language, depending on said evaluation.

6.6 The board regards difference feature "a" ("group 1") as solving a different technical problem, namely providing a platform for running application code, to the remaining difference features "b", "c" and "d" ("group 2"), which address the problem of extending the software capabilities of the digital home terminal. Hence the contribution of the two groups of differences to inventive step must be considered separately.

6.7 Regarding "group 1", in the context of the digital home terminal of D1, an article of consumer electronics, the use of a virtual machine would have been known to the skilled person as a means of decoupling the characteristics of the application code from those of the hardware, thus allowing the same software to run on a range of different hardware devices. Hence feature "a" cannot lend inventive step to the claim.

6.8 Regarding "group 2", the board agrees with the decision that it would have been obvious to apply the teaching of D2 to D1. In doing so, the use of a second interpretative language (difference "b") would have been an obvious design choice for the skilled person seeking to extend the software capabilities of the digital home terminal. Differences "c" (language recognition) and "d" (use of the appropriate interpreter for each language) would have been obvious consequences of difference "b". In connection with difference "c" (language evaluation), the board notes that it was well known at the priority date, for instance on PCs running DOS and/or Windows, to use file names to indicate the language of a code file (such as ".BAS" for Basic or ".PAS" for Pascal), as also mentioned in the description; see page 15, lines 14 to 16.

6.9 The appellant has argued that D2 teaches to compile all application languages into O-code, thereby using a single interpreter for O-code. The board finds that D2 does not touch on the question of writing downloaded applications in several languages. Compiling all application code to O-code would have been one possible interpretation by the skilled person reading D2. Another possible interpretation would have been to use a different interpreter for each language. The board considers both to be obvious alternatives.

6.10 The board concludes that the subject-matter of claim 1 lacks inventive step, Article 56 EPC 1973, in view of the combination of D1 and D2.

Order

For these reasons it is decided that:

The appeal is dismissed.

The Registrar: The Chairman:

L. Stridde M. Müller

Decision electronically authenticated