

Internal distribution code:

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

**Datasheet for the decision
of 1 October 2019**

Case Number: T 1745/13 - 3.5.06

Application Number: 09775012.9

Publication Number: 2353086

IPC: G06F9/455

Language of the proceedings: EN

Title of invention:

SYSTEMS AND METHODS FOR FACILITATING VIRTUALIZATION OF A
HETEROGENEOUS PROCESSOR POOL

Applicant:

Citrix Systems, Inc.

Headword:

Heterogeneous processor pool/CITRIX

Relevant legal provisions:

EPC Art. 56

Keyword:

Inventive step - (no)

Decisions cited:

Catchword:



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 1745/13 - 3.5.06

D E C I S I O N
of Technical Board of Appeal 3.5.06
of 1 October 2019

Appellant: Citrix Systems, Inc.
(Applicant) 851 W. Cypress Creek Road
Fort Lauderdale, FL 33309 (US)

Representative: Murgitroyd & Company
Scotland House
165-169 Scotland Street
Glasgow G5 8PL (GB)

Decision under appeal: **Decision of the Examining Division of the
European Patent Office posted on 5 March 2013
refusing European patent application No.
09775012.9 pursuant to Article 97(2) EPC.**

Composition of the Board:

Chairman M. Müller
Members: S. Krischer
B. Müller

Summary of Facts and Submissions

- I. The appeal is directed against the decision of the examining division, dated 5 March 2013, to refuse application No. 09775012.9 for insufficient disclosure (Article 83 EPC). In the *obiter dicta* concerning novelty, document D1 was cited ("VMware VMotion and CPU Compatibility", white paper by VMWare, Inc., 2008, as of this date still available on the Internet at http://web.archive.org/web/20080804071052/http://www.vmware.com/files/pdf/vmotion_info_guide.pdf,XP2568512).
- II. A notice of appeal was received on 6 May 2013. The appeal fee was paid on the same day. A statement of grounds of appeal was received on 4 July 2013. In its section 1.5, a new document D3 (US 2008/183944 A1) is cited.
- III. In a communication dated 12 March 2019, the rapporteur raised objections of insufficient disclosure and clarity.
- IV. In a letter dated 27 June 2019, the appellant submitted arguments and filed amended claims according to a main request and auxiliary request A.
- V. In its summons to oral proceedings, the board addressed these arguments and gave reasons as to why the claims lacked an inventive step over D1 in combination with D3. Clarity objections were also raised.
- VI. In a letter dated 11 September 2019, the appellant withdrew its request for oral proceedings, submitted further arguments and filed a clarified set of claims

(marked "main request") as its sole (and final) request.

VII. Oral proceedings were cancelled.

VIII. Claim 1 reads as follows:

"1. A method for facilitating virtualization of a heterogeneous processor pool, in a system comprising a control operating system (105) executing within a first virtual machine (106a), a guest operating system (110) executing within a second virtual machine (106b), a virtual processor (132), a hypervisor (101), a processor allocation component (210) and a plurality of physical processors (221); the method comprising:

identifying, by the processor allocation component (210), a plurality of the physical processors (221) available for computing;

determining, by the processor allocation component (210), a set of flags identifying functionality common across the plurality of physical processors (221), each of the set of flags identifying a type of functionality provided by each of the plurality of physical processors (221); and

allocating, by the hypervisor (101) to the second virtual machine (106b), access to only the identified common functionality in one of the plurality of physical processors (221) by:

preventing an attempt by the second virtual machine to access functionality that is provided by a first physical processor of the plurality of physical processors (221), and

allowing an attempt by the second virtual machine to access functionality that is provided by a second physical processor of the plurality of physical processors (221), a first computing device comprising the first physical processor and a second computing device comprising the second physical processor."

Reasons for the Decision

1. *Summary of the invention*

The invention relates to "facilitating virtualization of a heterogeneous processor pool" in a "computing device 100a" (see figure 2A). This eases the migration (see paragraph [3] in the original description) of virtual machines (106a, 106b) between heterogeneous physical devices, in particular processors (221a, 221b). The "computing device 100a" can be virtually any type of computer ([36]: "workstation, ..., laptop, ..., server, handheld computer, mobile telephone, ..., a gaming system, ..."). The claimed facilitation is achieved by restricting the functionality (e.g. the instruction set) of each of the virtual processors to the functionality common to all (heterogeneous) physical processors. This common functionality is stored as a set of flags. For instance, the description mentions two flags "3DNOW" and "3DNOWEXT" as indicating whether a processor includes a certain multimedia extension (see [52]).

The board notes that it is known that processors of the x86 architecture (Intel, AMD) provide a set of flags as claimed which can be read by a program through the instruction "CPUID" (see for example D1, page 3, last paragraph, to page 4, last paragraph, and <https://en.wikipedia.org/wiki/CPUID>, which also mentions the 3DNOW instruction set extension).

2. *Amended claims*

Claim 1 according to previous auxiliary request A contains a step of intercepting requests for access to unallowed functionality of a virtual processor and a step of responding to them with an indication that this functionality is not available. In claim 1 according to the new sole request, these steps have been replaced by a step of preventing an attempt to access functionality and a step of allowing such access. The intercepting step is omitted. The board considers that the new step of preventing corresponds substantially to the responding step of previous auxiliary request A, and that the new step of allowing complements it. Thus, given that amended claim 1 corresponds closely to the claims discussed in the summons to oral proceedings, the board admits the new set of claims (Article 13(1) RPBA) and, in coming to a decision on new claim 1, can rely on arguments on which the appellant has had an opportunity to comment (Article 113(1) EPC).

3. *Insufficient disclosure and clarity*

- 3.1 The board considers the invention sufficiently disclosed (Article 83 EPC), in contrast to the appealed decision (2-2.10), since the description ([68]) discloses a way of carrying out the invention, namely

in that the hypervisor intercepts "a request ... for access to functionality not identified by a flag" (e.g. an unallowed instruction) and responds with an indication that the functionality is not available. Although the decision finds that the intercepting is not sufficiently disclosed (2.2), it agrees in principle with the appellant that such an interception can be implemented by an (ordinary) emulator that interprets all instructions and does not execute them natively (2.8, second paragraph). The board agrees, and considers the arguments in the grounds of appeal (3.7-3.14) convincing. Since emulation is one of two basic methods of "processor virtualization" (see D3, [38]), the board considers that the invention is disclosed in a manner sufficiently clear and complete for it to be carried out by a person skilled in the art.

3.2 The board interprets the step of allocating (including preventing and allowing) in the sense of paragraph [68], which is the only pertinent passage of the description and the only paragraph in the description that discloses (in its last sentence) the word "preventing", which has been newly introduced in the claims. Accordingly, the allocating step is taken to refer to intercepting the requests by the second virtual machine and responding with an error message if the requested functionality (i.e. instruction) does not belong to the common functionality (marked in the set of flags).

4. *Inventiveness of the subject-matter of claim 1*

4.1 The board takes the same *starting point* (D1) as the *obiter dicta* of the decision (sections 5 and 6).

D1 discloses easing the live migration of virtual machines between heterogeneous processors by enabling the entire cluster of processors to use the same set of basic features (page 6, fifth paragraph). That is, in the words of claim 1, D1 discloses a set of flags of the common functionalities.

- 4.2 However, in D1 processor features (i.e. functionalities) are masked (or hidden) by modifying the semantics of the CPUID instruction (page 6, third paragraph). The features are not disabled. An application that does not use the CPUID instruction to determine the supported features may simply call an instruction, which is then executed (fourth paragraph).
- 4.3 Thus, the claim *differs* from D1 in that it prevents access to functionality of the first physical processor which is not in the set of flags of the common functionalities, and in allowing access to functionality of the second physical processor which is in the set of flags.
- 4.4 The objective technical *problem* to be solved may therefore be regarded as how to enforce the meaning of the set of flags according to D1. This problem essentially corresponds to the one proposed in the board's summons to oral proceedings (namely "how to prevent any guest program from executing on the (real) processor an instruction which is not part of the flagged functionalities").
- 4.5 In its letter dated 11 September 2019 (page 4, paragraphs 4 and 5), the appellant proposes using a different objective technical problem (how to facilitate virtualisation of a heterogeneous processor

pool). However, the appellant does not give any reasons why the problem used by the board is not suitable. Therefore, the board is not convinced and continues to consider the problem specified above.

4.6 The solution proposed cannot be considered as involving an inventive step, since the skilled person would first look at the two known basic methods for virtualising a processor, namely emulation and direct native execution (see D3, [38], sentences 4 and 6). He or she would immediately recognise that direct native execution does not easily allow the hypervisor to disable the execution of instructions. On the other hand, when the hypervisor emulates the processor (either by interpreting or by binary translation), it has to handle each program instruction in turn. In the course of this, the hypervisor could easily detect any instruction which is not in the set of flags of the virtual processor it emulates. It would be obvious to program the hypervisor such that in this situation it outputs, for example, an error message saying that the functionality of this instruction is not available. This prevents (or allows, in the converse situation) a request to access a functionality. No inventive activity can be recognised in applying to the hypervisor this standard programming methodology of preventing (e.g. by outputting an error message) an unallowable access to a resource (here, the functionality of a processor).

4.7 Therefore, the subject-matter of claim 1 is not inventive (Article 56 EPC) over D1 in combination with D3.

Order

For these reasons it is decided that:

The appeal is dismissed.

The Registrar:

The Chairman:



I. Aperribay

M. Müller

Decision electronically authenticated