

**Internal distribution code:**

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

**Datasheet for the decision  
of 25 October 2016**

**Case Number:** T 0992/11 - 3.5.06

**Application Number:** 05775076.2

**Publication Number:** 1784723

**IPC:** G06F9/445

**Language of the proceedings:** EN

**Title of invention:**

METHOD AND APPARATUS FOR MODIFYING SOFTWARE

**Applicant:**

Extreme Networks, Inc.

**Headword:**

Modifying software/EXTREME NETWORKS

**Relevant legal provisions:**

EPC 1973 Art. 56

**Keyword:**

Inventive step - (no)

**Decisions cited:**

**Catchword:**



**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 0992/11 - 3.5.06

**D E C I S I O N**  
**of Technical Board of Appeal 3.5.06**  
**of 25 October 2016**

**Appellant:** Extreme Networks, Inc.  
(Applicant) 3585 Monroe Street  
Santa Clara, CA 95051 (US)

**Representative:** Regimbeau  
20, rue de Chazelles  
75847 Paris Cedex 17 (FR)

**Decision under appeal:** Decision of the Examining Division of the  
European Patent Office posted on 16 December  
2010 refusing European patent application No.  
05775076.2 pursuant to Article 97(2) EPC.

**Composition of the Board:**

**Chairman** W. Sekretaruk  
**Members:** M. Müller  
G. Zucka

## Summary of Facts and Submissions

I. The appeal lies against the decision of the examining division, with reasons dated 16 December 2010, to refuse European patent application No. 05 775 076.2 for lack of novelty or inventive step over the document

D1: US 2002/073410 A.

II. Notice of appeal was filed against the decision on 15 February 2011, the appeal fee being paid on the same day. A statement of grounds of appeal was received on 11 April 2011. The appellant requested that the decision be set aside and that a patent be granted on the basis of claims 1-20 as filed with the grounds of appeal.

III. In the annex to a summons to oral proceedings, the board informed the appellant of its opinion that claim 1 lacked inventive step in view of D1, Article 56 EPC 1973. Objections under Article 123(2) EPC and Article 84 EPC 1973 were also raised.

IV. In response to the summons, by online submission on 23 September 2016, the appellant filed amended sets of claims 1-20 according to a main and an auxiliary request.

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

"A method comprising:

receiving a software upgrade comprising one or more executable software modules of an operating system (OS) comprising a plurality of software modules;

installing some or all of the one or more modules of the OS;

providing a notice of impending termination to an instance of an executing software module of the OS that corresponds to an installed module;

terminating the instance of the executing software module; and

after the terminating, starting the corresponding instance of the installed module; and wherein an operation of a computer system on which the OS is running is not interrupted by the terminating of the instance of the executing software module."

Claim 1 of the auxiliary request differs from claim 1 of the main request in that the "some or all" in the installing step has been replaced by "some, but not all". Both requests also comprise an independent claim for an article of manufacture defined by reference *inter alia* to method claim 1.

VI. With letter dated 21 October 2016, the appellant's representative informed the board that the appellant would be neither present nor represented during the oral proceedings.

VII. Accordingly, oral proceedings were held as scheduled but in the appellant's absence. At the end of the oral proceedings, the chairman announced the board's decision.

## **Reasons for the Decision**

### *The appellant's absence from the oral proceedings*

1. The appellant was duly summoned but chose not to attend the oral proceedings. According to Article 15(3) RPBA, the board is not obliged to delay any step in the proceedings, including its decision, by reason only of

the absence at the oral proceedings of any party duly summoned who may then be treated as relying only on its written case.

*The invention*

2. The application is concerned with the problem of upgrading an operating system (OS) without interrupting the computer system on which the OS is running (see paragraphs 1, 4, 5, 10 and 18 of the application as originally filed).
- 2.1 It is disclosed that upgrading may affect the entire OS or only one or more of its modules (see paragraphs 10, 14, 15 and 18).
- 2.2 As a solution it is proposed that new modules to be installed are loaded (installed) before, but only started after, the old modules to be replaced are terminated. It is stated that "in this manner, certain modules can be upgraded and restarted, without interrupting operation of the computing system as a whole" (figure 2/3, steps 210, 230 and 240, as well as paragraphs 22 and 23, esp. paragraph 22, last sentence, and paragraph 23, lines 10-12).
- 2.3 As an example, reference is made to an "internetworking protocol" in a network switching device which might need upgrading, and it is disclosed that during the upgrade ongoing communication is terminated (paragraph 23, lines 12 *et seq.*).

*The prior art*

3. D1 discloses a "telecommunications platform that provides software upgrade capability without shutting

down the entire platform or wasting processing power" (see paragraph 7).

3.1 The upgrading protocol provided to achieve this purpose is depicted in figure 2 and described in paragraphs 36-47. It is disclosed that first any new program is loaded and started (2-3) and then the corresponding old program is shut down (2-4) and "unpublished" (2-5). Before being shut down, the old program saves its state information (see figure 5 and paragraph 41), which then may have to be converted for the new one (2-8, see also paragraph 43). The conversion may take place while the data structures are in use, which reduces undesirable system downtime (see paragraph 75). Once the conversion is finished (2-10), and only then, the new program is "activated" (2-11) and "published" (2-12) and fully takes over the task of the old program. It is only at this point that, according to D1, "the upgraded program is running" (2-13 and paragraph 47).

3.2 D1 discloses that programs are loaded by means of a so-called "load module". Moreover, it is disclosed that any program is "a running instance of a load module" and is created "as a consequence of the loading of the corresponding load module" (see paragraph 29). Furthermore, the load module for a particular program is "designed (coded) to know what data and process status information" is to be stored as state information and includes the shutdown function for the old program being replaced (see paragraphs 32 and 33).

*Clarity, Article 84 EPC 1973, and claim construction*

4. Claim 1 of both requests specifies a step of installing modules of the OS. In response to a concern raised by the board in its summons, the appellant referred to the

description, which introduces the term "modules" to denote "other parts" of the OS as opposed to the OS kernel (see paragraph 8 and the appellant's letter of 23 September 2016, page 1, point 8.2). For the purpose of this decision, the board adopts this interpretation.

5. Claim 1 of both requests qualifies the specified method with the clause "wherein an operation of a computer system on which the OS is running is not interrupted by the terminating of the instance of the executing software module".
  - 5.1 The board considers this phrase to be unclear, Article 84 EPC 1973. More specifically, what it considers to be unclear is exactly which "operation of [the] computer system" is not interrupted.
  - 5.2 When, as just explained, OS modules are distinguished from the OS kernel, the termination of OS modules need not affect the operation of the OS kernel. So, arguably, "core functionality" of the OS need not be interrupted when modules are upgraded (see the appellant's letter, *loc. cit.*).
  - 5.3 However, the executing software module is terminated before the corresponding installed module is started. Hence, the operation of at least that module will be interrupted for a short period of time. The functionality provided by that module might be crucial for a user of the system. For example, when communication is terminated in order to upgrade the internetworking protocol (see the description, paragraph 23), this might well be perceived by the user as an interruption of "an operation of" the computer system running as a whole.



6. This issue notwithstanding, the board considers that claim 1 is clear enough to allow an assessment of inventive step. For this purpose, claim 1 is interpreted as specifying that the operation of the computer system running the OS is not *entirely* interrupted.

*Novelty, Article 54 EPC 1973*

7. The examining division identified the "starting" of a module according to the invention with the "full activation" of the program according to D1 (see the decision, reasons 2). While the board agrees that the notion of "starting" a module is open to different interpretations, D1 distinguishes the concepts "load" and "start" in the labelling of step 2-3 and identifies a program as a "running instance of a load module" (paragraph 29). The board therefore takes the view that the skilled person would identify the claimed "starting" with the "start of the program" according to D1 (i.e. with the "start" aspect of step 2-3 rather than with step 2-11). Having said that, the board notes that D1 does not disclose anything substantial that a started new program would have to do before full activation. Rather, D1 does not exclude the possibility that the newly started program immediately hangs waiting for the data conversion to finish.
8. According to D1, the old and new versions will run in parallel for a short period of time (see e.g. paragraphs 29 and 75 and the grounds of appeal, paragraph bridging pages 4 and 5), whereas this is expressly excluded by the claimed method. Accordingly, the subject-matter of claim 1 is new over D1.

*Inventive step, Article 56 EPC 1973*

9. The subject-matter of claim 1 (both requests) differs from D1 in precisely this feature. While D1 discloses starting the new program before terminating the old program, the invention proposes first terminating the old program and then starting the new one.
  
- 9.1 The appellant argues (see grounds of appeal, paragraph bridging pages 4 and 5) that this difference has the effect that the invention uses fewer "processing and memory resources" than D1. The board notes that, according to claim 1, the new programs are installed before the old program is terminated. Old and new programs therefore coexist in memory also according to the claimed invention. However, according to the invention the old program can be removed from memory earlier than according to D1. Also, the invention enables the saving of processing resources. Therefore, the board accepts the appellant's argument that the claimed invention uses fewer resources than the system of D1 and, thus, can be considered to solve the problem of "[h]ow to reduce the use of resources during the installation of modules on a computer system" (see the appellant's letter of 23 September 2016, page 2, paragraph 6).
  
- 9.2 That said, the board also considers the following. If the skilled person were to realise that the upgrading process according to D1 required an undesirable, unacceptable or unavailable amount of resources, he would not need an inventive step to determine that this is, at least partly, due to the fact that old and new programs coexist in memory until the new program is started and may be running in parallel. Furthermore, the skilled person would, in addressing this problem,

obviously contemplate the possibility of terminating the old program before starting the new one.

10. The appellant argues (see its letter of 23 September 2016, page 2, last paragraph) that according to D1 any new program is tied to its load module, that "the load module of the new program is required to shutdown the old program, and thus it is not possible [...] to terminate the old program without running the new program". The appellant also points out that "the transfer of state information from the old program to the new program could [...] not happen if the load module and the new program were not started before the shutdown of the old program."
- 10.1 The board agrees that the load module of D1 would have to be modified in order to incorporate the distinguishing feature of terminating the old program before starting the new one. However, the board disagrees with the appellant that this "teaches away" from the invention (see letter of 23 September 2016, page 3, paragraph 2).
- 10.2 It is a matter of necessity that the state information of the old program is stored for conversion before the old program is shut down. There is no logical necessity, however, to start the new program before the old program is shut down. Also, as already mentioned, D1 does not teach that the new program, albeit started, has anything significant to do before the old program is terminated and therefore could easily be started later instead.
- 10.3 Therefore, there would be no difficulty for the skilled person to rearrange the tasks of the load module in the claimed manner.

11. Also the board does not see any "prejudice in the art" which the skilled person would have to overcome to modify D1 towards the invention, contrary to the appellant's allegation (see grounds of appeal, page 4, paragraphs 11 and 12).
- 11.1 Although the skilled person would understand that it might increase the downtime of the system to terminate an old module before starting a new one, for instance from paragraph 75 of D1, this would merely be an obvious disadvantage which the skilled person would factor into the trade-off decision to be made. For instance, a slightly increased system downtime might have to be accepted if the available memory did not allow the two module versions to run in parallel. In any case, in the board's view the skilled person would not need to take an inventive step to make that trade-off.
- 11.2 The board understands the appellant to argue that the claimed solution is made possible only because not all OS modules are upgraded at the same time (grounds of appeal, page 4, penultimate paragraph). The board disagrees. The same trade-off would exist if all modules were upgraded.
12. In summary, the board concludes that claim 1 of the sole request lacks inventive step over D1, Article 56 EPC 1973.

**Order**

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

The appeal is dismissed.

The Registrar:

The Chairman:



B. Atienza Vivancos

W. Sekretaruk

Decision electronically authenticated