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

Datasheet for the decision
of 15 January 2019

Case Number:	T 1042/17 - 3.5.06
Application Number:	14192798.8
Publication Number:	3007068
IPC:	G06F9/50
Language of the proceedings:	EN

Title of invention:
System and method for prioritizing on-access scan and on-demand scan tasks

Applicant:
Kaspersky Lab, ZAO

Headword:

Relevant legal provisions:
EPC Art. 56

Keyword:
Inventive step - no

Decisions cited:

Catchword:

Case Number: T 1042/17 - 3.5.06

D E C I S I O N
of Technical Board of Appeal 3.5.06
of 15 January 2019

Appellant:
(Applicant)

Kaspersky Lab, ZAO
39A/3 Leningradskoe Shosse
Moscow 125212 (RU)

Representative:

Sloboshanin, Sergej
V. Füner Ebbinghaus Finck Hano
Patentanwälte
Mariahilfplatz 3
81541 München (DE)

Decision under appeal: **Decision of the Examining Division of the European Patent Office posted on 22 November 2016 refusing European patent application No. 14192798.8 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 is against the decision of the examining
        division dated 22 November 2016 to refuse European
        patent application No. 14 192 798 for lack of inventive
        step over the document

        D1: Zhen Zhao, "A preemptive connection pool manager
            for web based application collaboration", 8th Int.
            Conf. on Collaborative Computing, Networking,
            Applications and Worksharing, 2012.

        Inter alia, the decision also mentioned the documents

        D2: US 8 479 294 B1 and
        D4: US 2005/0149749 A1.

II.   Notice of appeal was filed on 20 January 2017, the
        appeal fee being paid on the same day. A statement of
        grounds of appeal was received on 31 March 2017. The
        appellant requested that the decision under appeal be
        set aside and a patent be granted on the basis of
        claims 1-13 according to a main or an auxiliary request
        as filed with the grounds of appeal.

III.   In an annex to a summons to oral proceedings, the board
        informed the appellant of its preliminary opinion that
        the claimed invention lacked inventive step in view of
        documents D1, D2 and D4, Article 56 EPC.

IV.   In response to the summons, with a letter dated
        13 December 2018, the appellant filed amended claims
        1-13 to replace the pending auxiliary request.

        Claim 1 of the main request reads as follows:

"A method for prioritizing scan requests, the method comprising:

establishing and reserving, by a thin client (131), two or more connections between the thin client (131) and a first virtual machine (120) of a computer, wherein the thin client (131) is running on a second virtual machine (110) of the computer;

when one or more of the reserved connections are not being used for communicating on-access scan (OAS) requests or on-demand scan (ODS) requests, allocating, by the thin client (131), said one or more of the reserved connections to communicate OAS or ODS requests between the thin client (131) and the first virtual machine (120), wherein the OAS and ODS requests are requested by the thin client (131) for execution by the first virtual machine (120);

determining, by the thin client (131), whether all the reserved connections are currently being used for communicating OAS or ODS requests where at least one reserved connection is currently being used for communicating ODS requests;

reallocating, by the thin client (131), said at least one reserved connection that is currently being used for communicating ODS requests to be used for communicating at least one OAS request; and transmitting, by the thin client (131), said at least one OAS request on said reallocated reserved connection to the first virtual machine (120) for execution thereof."

Claim 1 of the auxiliary request differs from claim 1 of the main request in that it is specified that the thin client

"... comprises an on-access scan (OAS) module (150) and an on-demand (ODS) module (140); ..."

and in that at the end of the "reallocating" step it is
stated that

"... wherein the reallocated reserved connection is
dedicated for transmitting events of the OAS requests
and not for transmitting events for the ODS requests,
and wherein the thin client (131) conducts a forced
transfer of the at least one reserved connection from
the ODS module (140) to the OAS module (150); ...".

V.   Oral proceedings took place on 15 January 2019. At
        their end, the chairman announced the decision of the
        board.

Reasons for the Decision

The invention

1.   In general, the application relates to an architecture
        in which virus scanning requests from a "thin client"
        are scheduled and prioritized (see page 1, lines 5-7
        and figure 1, nos. 130 and 131).

1.1   The application discloses that (thin) clients - i.e.
        programs - may be installed on or operate in
        "communication with corresponding virtual
        machines" (VM; see page 5, lines 11-16, and figure 1).
        Moreover, it is disclosed that the thin clients - and
        their virtual machines - may be installed on the server
        ("system thin clients") or the user device ("user thin
        client"; *loc. cit.*).

1.2   Each VM is connected with the virus scanning service,
        which itself runs on a virtual machine (referred to as

a "security virtual machine" SVM), via one or more
(TCP-) connections (see figure 1; page 5, lines 25-28;
and page 8, lines 11-16 and 25-30).

1.3   The application further relates to "on-access" and "on-
demand" virus scanning, referred to as "OAS" and "ODS",
respectively. In the board's understanding, the first
term refers to the scanning of files when they are
opened, saved or run, whereas the second one refers to
user-triggered scanning of drives, folders or files
(see page 2, paragraphs 6 and 7). It is observed that
OAS is lean - in that it produces only "few events to
be transmitted" to the SVM - but urgent, whereas ODS
may be voluminous - in producing "many events to be
transmitted" - and less urgent (see paragraph bridging
pages 6 and 7, and page 8, paragraph 2).

1.4   In order to give the urgent OAS requests priority, the
application discloses that some of the connections may
be "reserved" for handling the OAS requests (see
sentence bridging pages 8 and 9).

1.5   Moreover, OAS requests may pre-empt ODS requests as
depicted in the flow chart of figure 2. An ODS request
for which no free connection is available has to wait
(step 225). An OAS request for which no free connection
is available may have to wait, too, but only if all
connections are used by other OAS requests (steps 245
and 255). If, however, at least one connection happens
to be allocated to an ODS request, the ODS request may
be pre-empted and the connection reallocated to the OAS
request (step 250).

1.6   It is disclosed that it is up to the client to
establish, reserve, allocate and re-allocate
connections (see e.g. the paragraph bridging pages 2

and 3; page 8, last paragraph; page 10, paragraph 2;
page 11, paragraph 2).

The prior art

2.   D4 relates to the virus scanning in a data processing
        network (see paragraph 2), in which several so-called
        "secondary servers" provide different virus checking
        services (see paragraph 25 and figure 1, nos. 24-26)
        operating on files obtained from a large "primary"
        network file server (figure 1, no. 27), or rather from
        one of several "data movers", themselves separate
        computers, which the primary file server may provide
        for "parallelism and scalability" (see paragraph 26).

2.1   D4 discloses the distinction between "on-access" and
        "on-demand" scans and the fact that on-access scans
        have priority (see paragraphs 10 and 32).

2.2   As a solution, D4 proposes that each data mover has a
        "virus scan request queue", shared between on-access
        and on-demand scan requests, in which it tries to
        maintain a balance between the two types of requests so
        as "to provide a relatively continuous flow of requests
        [...] without significantly degrading the response time
        [...] for [...] 'on-access' requests" (see
        paragraph 33).

3.   D1 relates to so-called http connection pools (see
        page 326, right column, last two lines) managed by a
        "connection pool manager" (abstract, lines 15-18, and
        figure 7).

3.1   It discloses connection requests and connections with
        high priority (HP) and with low priority (LP) (see
        page 329, left column, section III, lines 1-13). An HP

connection request may have to wait if all connections
are other HP connections whereas if an HP connection
request "sees" an LP connection, it can "free up
capacity by preempting [the] LP connection" (*loc. cit.;*
page 327, left column, the paragraph below figure 2, to
right column, paragraph 1; and figures 2-4 and 7). D1
also uses the language that LP "connections are removed
from the pool" (loc. cit.) or "dropped" (see page 330,
right column, paragraph 3 from the bottom).
Alternatively, D1 discloses that "the connection pool
capacity" can be adjusted by providing, for instance,
an additional HP connection when needed (*loc. cit.).*

3.2   A major part of D1 contains a probabilistic performance
        analysis of the proposed scheduling model, e.g. by
        determining the probabilities that an HP or LP request
        is blocked (see page 329, right column, section "A" *et
        seq.).* D1 does not expressly disclose where the
        "connection pool manager" is located, in the board's
        understanding because this is not crucial for the
        theoretical analysis in question.

4.   D2 mentions anti-malware scanning on and for virtual
        machines (see e.g. column 1, lines 26-40, and
        figure 2).

Inventive step, Article 56 EPC

5.   In the decision, D1 was chosen as the starting point
        for the inventive step assessment. In its preliminary
        opinion, the board followed this proposal. Even though,
        however, D1 indeed shares a number of central features
        with the claimed invention, it is invariant to which
        specific LP and HP requests are being considered. In
        particular, it does not relate to virus scanning. On
        the other hand, D4 mentions on-access and on-demand

virus scan requests and discloses the problem of
scheduling them so that "on-access requests" are given
"appropriate priority" (see especially paragraph 10).
Therefore, because D4 discloses the same application
context as the invention and shares the main goal with
it, the board considers that D4 is the more appropriate
starting point for the assessment of inventive step.

5.1   The solution disclosed in D4 based on data movers and
        shared virus scan request queues is different from the
        invention - based on reserving, allocating and re-
        allocating "connections" - and the board does not
        consider it obvious to modify the solution of D4 so as
        to arrive at the invention.

5.2   However, in the board's judgment it would be obvious
        for the person skilled in the art to address the
        problem of providing an alternative solution for the
        scheduling of OAS and ODS requests and their relative
        priority in a network with several remote virus
        checking servers.

6.   The board considers that the skilled person addressing
        this problem would make reference to document D1
        because it discusses the scheduling of several kinds of
        requests with one of two different priorities (see e.g.
        page 327, left column, lines 1-4, and page 329,
        section III).

6.1   Applying the solution of D1 to the mentioned problem of
        D4 would yield a system that differed from the inven-
        tion according to claim 1 of the main request in that

        i)   the client is "thin",
        ii) the thin client and the virus scanners run on
             "first" and "second" virtual machines,

iii) the thin client is responsible for "reserving" connections and for "allocating" and "reallocating" the reserved connections, and

iv) the pre-emption is implemented by reallocating an existing "reserved" connection to a different type of request.

Differences i)-iii) correspond, essentially, to differences IIA, IIb and IV according to the decision under appeal. Difference iv) was added in view of the appellant's arguments during the oral proceedings.

6.2   Features i) and ii) are unrelated to the problem of scheduling OAS and ODS requests because the scheduling problem and its solution would remain the same if a different type of client program was used and if virtual machines were dispensed with. The board notes that the description explains the term "thin client" (see page 5, lines 13-15) but does not justify the use of thin clients as opposed to "fatter" ones. Hence, the skilled reader of the application must be assumed to know the concepts of thin and fat client programs and understand their main properties and advantages. The skilled person also knows the concept of virtualization based on virtual machines (see also the description, page 4, penultimate paragraph). Hence, the board considers that the skilled person would make use of virtual machines and "thin" clients to achieve their known purposes, when carrying out the system obtained by applying the solution of D1 to the problem of D4, without exercising an inventive step. Only, in passing, the board notes that from the appellant's explanations and from D2 it appears that the use of virtual machines and thin clients is, in principle, known in the pertinent field.

7.   As regards feature iii), the board maintains that D1
        leaves open where exactly the "connection pool" is
        located. Although the appellant is right that D1 does
        not disclose it to be located at the client machine,
        the board does not accept that it must be located at
        the server (see the grounds of appeal, page 5, last
        paragraph). That D1 leaves this open is consistent with
        its abstract nature and its focus on a performance
        analysis which does not depend on the precise placement
        of the connection pool between the applications
        (clients) and the resources. Moreover, D1 leaving this
        genuinely open, does not expressly discourage the
        placement of the connection pool manager at the client.

7.1   In the board's judgement, the main effect of having the
        client manage the connection pools is that different
        clients could decide to make different decisions, for
        instance as regards "connection control" (see, for
        instance, D1, page 327, left column, below figure 2).
        For example, one client might use pre-emption whereas
        another one might make do with rate adaptation (see D1,
        figure 4). Thereby, the overall scheduling becomes more
        flexible in reacting to different client requirements.

7.2   However, if such flexibility was of interest, the board
        considers that it would be obvious to task the clients
        themselves with the management of their connection
        pools.

7.3   The board also notes that D4 discloses a separate
        request queue at each data mover, i.e. at the client
        side with respect to the virus checking services. As an
        aside, it is noted that virus checking requests with
        the same priority are distributed over several servers,
        so that it is impossible to perform the scheduling at
        the individual virus scanning servers (nos 32-34 in

D1). The board therefore considers that D4 itself
suggests making scheduling decisions at the client side
rather than at the server side.

8.   As regards feature iv) the appellant refers to the
        statement in D1 that a preempted LP connection is
        "removed" from the pool or "dropped" (loc. cit.) and
        states that this would have to be read as closing a
        connection (e.g. a socket) rather than maintaining and
        reallocating it.

8.1   In this regard the board first notes that D1 discloses
        not only the removal of LP connections but also the
        creation of new ones (see figure 3, "capacity
        adaptation"). The combination of both measures, i.e.
        the replacement of one connection by another one, is
        thus suggested by D1.

8.2   In order to actually carry out this solution, the
        skilled person would have to make a few implementation
        decisions. For example, the skilled person might use a
        "socket connection" to implement the connections and
        decide to "close a socket" and "open" a new one. Or, it
        could maintain an existing socket connection and
        dedicate it to a different task.

8.3   The board does not accept the appellant's argument that
        the claim language or the reference to TCP in the
        description implies specifically the use of socket
        connections. More generally, while it may be assumed
        that removing and creating the software objects -
        sockets or other - referred to as "connections"
        requires some computational effort, the same applies to
        the "allocation" of an existing connection to a type of
        request. There is no basis in the application for
        knowing whether the latter is smaller than the former.

Hence, no efficiency gain can be attributed to the claimed re-allocation of connections.

8.4   However, for the sake of argument, and to the appellant's benefit, it might be assumed that "reallocating" an existing connecting as claimed requires less computational effort different from removing a connection and creating a new one. For lack of detail in the description, this fact must be assumed to belong to the skilled person's common knowledge. In the board's judgment, the skilled person would thus choose this option for its known advantage without exercising an inventive step.

9.   In summary, the board concludes that claim 1 of the main request lacks inventive step over D4 and D1, in combination with common knowledge in the art.

10.   As regards the auxiliary request, the board considers that the use of dedicated, but otherwise unspecified, "OAS" and "ODS" modules for allocating requests to connections does not go beyond saying that two dedicated pieces of program code are used for this purpose. In the board's judgement, this would be obvious in the software implementation to hand. The remaining language added to claim 1 of the auxiliary request does not imply any specific feature that was not already assumed in the analysis of claim 1 of the main request. Therefore, this analysis carries over to claim 1 of the auxiliary request and shows that also this claim lacks inventive step, Article 56 EPC.

Order

For these reasons it is decided that:

The appeal is dismissed.

The Registrar: The Chairman:

B. Atienza Vivancos   W. Sekretaruk

Decision electronically authenticated