Internal distribution code:

(A) [-] Publication in OJ

(B) [-] To Chairmen and Members

(C) [-] To Chairmen

(D) [X] No distribution

Datasheet for the decision
of 17 April 2018

Case Number: T 2026/15 - 3.5.06

Application Number: 07016599.8

Publication Number: 1921572

IPC: G06N1/00

Language of the proceedings: EN

Title of invention:

Method for training a system to specifically react on a specific input

Applicant:

Pucher, Max J.

Headword:

Training method/PUCHER

Relevant legal provisions:

EPC 1973 Art. 84

EPC R. 137(3)

RPBA Art. 12(1), 12(4)

Keyword:

Non-admission of auxiliary request by the examining division -
full reasoning is incompatible with non-admission
Auxiliary request not admitted by examining - admitted in
appeal (yes)
Claims - clarity (no)

Decisions cited:

G 0007/93, T 0820/14, T 2324/14

Catchword:

Case Number: T 2026/15 - 3.5.06

D E C I S I O N
of Technical Board of Appeal 3.5.06
of 17 April 2018

Appellant: (Applicant)	Pucher, Max J. Weidstrasse 26 6300 Zug (CH)
Representative:	FDST Patentanwälte Nordostpark 16 90411 Nürnberg (DE)
Decision under appeal:	Decision of the Examining Division of the European Patent Office posted on 22 May 2015 refusing European patent application No. 07016599.8 pursuant to Article 97(2) EPC.

Composition of the Board:

Chairman	W. Sekretaruk
Members:	M. Müller
	A. Teale

Summary of Facts and Submissions

I.   The appeal is against the decision of the examining
        division, with reasons dated 22 May 2015, to refuse
        European patent application No. 07 016 599, because the
        main request lacked clarity and inventive step. An
        auxiliary request was not admitted pursuant to
        Rule 137(3) EPC because it did not overcome the
        objections against the main request and also "prima
        facie" introduced new clarity problems.

II.   Notice of appeal was filed on 22 July 2015, the appeal
        fee being paid on the same day. A statement of grounds
        of appeal was received on 1 October 2015. The appellant
        requested that the decision be set aside and that a
        patent be granted according to either of the two
        requests on file and subject to the decision, namely on
        the basis of claim 1 as filed by telefax dated
        25 April 2013 (main request) and 20 March 2015
        (auxiliary request), both in combination with
        claims 2-18 as originally filed.

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

        "A method for training a system to specifically react
        on a specific input, comprising the following steps:

        -    defining a set of binary data structures, whereas
             each data structure represents an abstract
             description of a certain type of a real-world
             component, a real-world item or a virtual object,
             and each data structure comprising as features of
             said represented type of a real-world component, a
             real-world item or a virtual object it's [sic]
             possible states, it's [sic] attributes, it's [sic]
             possible actions to be performed, and possible

relationships between said real-world components,
real-world items or virtual objects,

- storing each of the data structures directly as a
  binary pattern in a computer memory or a disk
  memory,
- creating uniquely identifiable copies of the binary
  data structures to represent an individual instance
  of said types of real-world components, real-world
  items or virtual objects,
- creating a virtual state space of said individual
  instances of types of real-world components, real-
  world items or virtual objects by grouping them as
  relevant for a specific situation,
- receiving via keybo[a]rd input, sensor devices or
  other technical means changes of said status or
  said attribute values of at least one of said real-
  world components, real-world items or virtual
  objects,
- storing said received changes as a new version of
  the respective data structure, representing a
  changed status and a transition process of said
  individual real-world components, real-world items
  or virtual objects,
- analyzing similarities of said binary patterns
  stored in said computer or disk memory related to a
  particular action performed,
- searching for a match in said binary patterns of
  said data structures, and
- if at least one matched binary pattern of said data
  structures is identified, at least one possible
  action related to the matched binary pattern is
  proposed to an operator or performed
  automatically."

Claim 1 of the auxiliary request differs from claim 1
of the main request in that the "defining" step, the

first "storing" step, and the final "if"-clause now
read as follows (additions and deletions marked by the
board):

"...
- defining a set of binary data structures, <u>the data
  structures being binary pattern, not described as
  compiled machine-code or interpreted logic language
  based on a Boolean logic with data stored in data-
  base tables or as a rule based expert or business
  process system that requires knowledge to be
  encoded in fact and belief knowledge rules</u>, whereas
  each data structure represents an abstract
  description of a certain type of a real-world
  component, a real-world item or a virtual object,
  <u>(e.g. people, machines, robots, robotic entities,
  goods, tools, materials, physical objects, items in
  a warehouse, computer systems and components,
  manufacturing machines with their components and
  moving parts, physical or virtual documents,
  generic media content, abstract business entities,
  data manipulation, home automation system that is
  composed of sensors, control switches, actuators,
  motion sensor, light level sensor, light switch,
  light source, road vehicles, naval vessels,
  airplanes, process management system, business
  process management, navigation system)</u>, and each
  data structure comprising as features of said
  represented type of a real-world component, a real-
  world item or a virtual object it's ~~possible~~
  <u>allocated</u> states, it's attributes, it's ~~possible~~
  <u>allocated</u> actions to be performed, and ~~possible~~
  relationships between said real-world components,
  real-world items or virtual objects,
- storing each of the data structures <u>without
  transcoding, compilation or interpretation</u> directly

as ~~a~~ <u>the</u> binary pattern in a computer memory or a
disk memory

...
- if at least one matched binary pattern of said data
  structures is identified, at least one ~~possible~~
  action related to the matched binary pattern is
  ~~proposed to an operator or~~ performed
  automatically",

and that, in the second "creating" step, the phrase "as
relevant for a specific situation" was replaced by
"with regard to specific situations".

IV.   In an annex to a summons to oral proceedings, the board
        informed the appellant of its preliminary opinion that
        claim 1 of both requests was unclear, Article 84
        EPC 1973. Doubts regarding Articles 83 and 56 EPC 1973
        were also raised.

V.   In response to the summons, the appellant did not file
        either amendments or arguments.

VI.   The appellant was absent from the oral proceedings held
        on 17 April 2018, as announced in advance. At the end
        of the oral proceedings the board announced its
        decision.

Reasons for the Decision

*The non-admission of the auxiliary request by the examining
division*

1.   In its summons to oral proceedings, the examining
        division raised several clarity objections against
        claim 1 of the present main and then sole request (see

points 1 to 1.8). In response to the summons, the
appellant filed amended claim 1 of the present
auxiliary request. In its decision (see points 21.1 and
21.2 of the reasons), the examining division found that
the negative feature "the data structures [...] not
described as" was *prima facie* unclear and that the list
of examples of "real-world component[s]", "real-world
item[s]" and "virtual object[s]" introduced, *prima
facie* again, new clarity problems. Moreover several
objections raised in the summons were not overcome by
the amendments (see points 21.3 to 21.7 of the
reasons). On this basis, the examining division decided
not to admit the auxiliary request under Rule 137(3)
EPC (point 22 of the reasons).

1.1   The appellant challenged the non-admission decision,
        arguing as follows (see the grounds of appeal, page 5,
        paragraph 1). In "the decision the auxiliary request is
        discussed in detail, in particular with regard to
        Article 84 EPC. Hence, it is believed that the auxil-
        iary request is regarded as introduced during examining
        and has to be regarded during appeal proceedings."
        Furthermore, it was requested to "regard the auxiliary
        request as already admitted to the appeal proceedings".

1.2   In this request the appellant made implicit reference
        to Articles 12(1) and (4) RPBA and effectively argued
        that the board had no discretion to "hold inadmissible"
        the auxiliary request, which was filed with the grounds
        of appeal, even though it was, on the face of it, not
        admitted by the examining division. In other words, the
        appellant argued that the board was obliged to decide
        on the auxiliary request in substance and could not,
        instead, limit itself to a formal decision confirming
        the non-admission decision.

2.   The board agrees with the appellant's arguments.

2.1   Article 12(1)(a) and (4) RPBA states *inter alia* that "everything presented by the" appellant with the grounds of appeal "shall be taken into account by the Board", except that it has "the power [...] to hold inadmissible" in particular requests which were "not admitted in the first instance proceedings". This language implies that that which is "h[e]ld inadmissible" is not "taken into account".

2.2   Rule 137(3) EPC states that "No further amendment may be made without the consent of the Examining Division". Giving or denying "consent" to an amendment according to this rule is conventionally understood as "admitting" or "not admitting" an amended request. This is also the language used by the examining division. Correspondingly, Article 12(4) RPBA is taken to imply that the board has the power to "hold inadmissible" a request involving an amendment to the application to which the examining division has denied its consent under Rule 137(3) EPC.

2.3   The EPC does not expressly define what it means for an examining division to deny its consent to - or not admit - an amendment. It is clear, however, that once having given consent - or admitted - a request, the examining division is obliged to assess its merits in substance. In contrast, a request which is not admitted need not be further examined (or "taken into account" or "considered" as Articles 12(4), 13(1) and (3) RPBA put it). This is also the main reason why examining divisions do "not admit" a request, namely in order to "bring the examination procedure to a close"(see also G 7/93, point 2.5 of the reasons).

2.4   In its decision, the examining division gave extensive
        reasons why claim 1 of the auxiliary request was
        unclear, namely two full pages relating to seven
        separate clarity objections (see pages 8 and 9 of the
        decision, reasons 21-3 to 21.7). Only two of these were
        qualified as merely "prima facie". At least the other,
        fully reasoned, objections establish in substance why,
        according to the examining division, a European patent
        could not be granted on the basis of the auxiliary
        request. Therefore, the board agrees with the appellant
        that the examining division has "considered" and "taken
        into account" the auxiliary request.

2.5   The board hence also finds that the examining division
        implicitly admitted the auxiliary request and,
        equivalently, gave its consent under Rule 137(3) EPC.
        In other words, the examining division wrongly
        exercised its discretion because, having implicitly
        admitted the request, it no longer had a discretion not
        to admit it (see also T 2324/14, point 2.6 of the
        reasons).

3.   For applicants, and under the EPC, a decision in
        substance would have the same effect as a non-admission
        decision for substantive reasons, if it were not for
        the appeal. No patent is granted on the basis of the
        request in question.

3.1   Since, however, the boards of appeal may limit
        themselves to confirming the first instance decision
        not to admit a request rather than "take[it] into
        account", the appellant's position on appeal is worse
        if the examining division decided not to admit a
        request than if it had admitted it and decided on its
        substance.

3.2   If in situations like the one to hand, where it has
        been determined that a request has a substantive
        deficiency, the examining division were free to either
        refuse the request or not to admit it for one and the
        same reason, it would thus have exclusive control over
        the appellant's options on appeal. The board considers
        this to be obviously undesirable (see also T 820/14,
        reasons 11).

4.   The board's discretion under Article 12(4) RPBA assumes
        that the non-admission decision of the examining
        division was correct.

4.1   As already mentioned, the board agrees with the
        appellant that the present non-admission decision was
        incorrect.

4.2   Since the auxiliary request was filed with the grounds
        of appeal, Articles 12(1)(a) and 12(4) RPBA oblige the
        board to consider it in substance.

The invention

5.   The application relates to a system which monitors
        users' activities to identify their intentions, so that
        it can "specifically react on a specific input in real-
        time" (see page 1, paragraphs 1 and 2, and page 2,
        paragraph 5).

5.1   The description states that the methods according to
        the invention are applicable to essentially any system,
        for instance a robot control, a business application or
        a home automation system (see page 1, paragraph 3, to
        page 2, paragraph 2; page 45, paragraph 3 *et seq.*;
        page 48, paragraph 3 *et seq.*; figures 7 and 12).

5.2   The adaption of the system in response to user input is
        modelled by a state space. User activities are
        monitored (see page 1, paragraph 3, and page 2,
        paragraphs 2 and 3), yielding information which is used
        to "train" a program, referred to as a "User Trained
        Agent" (UTA) (see page 32, paragraph 4).

5.3   The system components are modelled in terms of "binary
        data structures", each representing "a certain type of
        real-world component" etc. and its "attributes",
        "actions" and "relationship" (paragraph bridging
        pages 2 and 3). Each "instance" of such a definition is
        stored "directly" in computer memory as a "uniquely
        identifiable cop[y] of the binary data structure[]"
        (see page 3, lines 3-5), where "unique identifiability"
        is achieved by what are called "Global Unified
        Identifiers" (GUIDs) (see page 26, paragraph 3). A
        "virtual state space" is generated from groups of such
        instances and transitions between them (page 3,
        paragraph 1).

5.4   The binary data structures are not disclosed in detail.
        It is however stated that they allow access "without
        any compilation or interpretation" of a programmed
        taxonomy and are thus faster than conventional methods
        (see page 3, last 3 lines; page 5, paragraph 3; page 6,
        paragraph 3; page 12, paragraph 3). It is also
        mentioned that the format is "proprietary" and "native"
        and executable by a "UTE transaction kernel" (page 25,
        paragraph 3 from the bottom; figure 1).

5.5   A "Trainer" module of the UTA is stated to produce a
        "decision tree" from the "raw data" relating to the
        state space as input by the user (see page 34,
        paragraph 2 *et seq.*). The internal nodes of that tree
        are said to store the "deciding feature", the leaves

storing one or more "data clusters" (see page 37, paragraph 3).

6.   An "Actor" module of the UTA performs "the necessary actions" whenever "a transaction is committed and a pattern is modified" (page 37, paragraph 4). The decision tree is used to select one leaf if possible (page 37, paragraph 5 *et seq.*). The selected leaf (if any) represents an "action". This action is either carried out or only proposed to the user (see page 38, paragraphs 1 and 2).

Clarity, Article 84 EPC 1973, and claim construction

7.   Claim 1 of both requests comprises broad and partly vague language. In the following, the board explains how, in its view, the claim language would have been construed by the skilled person and where the claims are unclear.

7.1   Claim 1 of both requests refers to a "system" which is trained to react to user input, and to "real-world component[s]", "real-world item[s]" and "virtual object[s]". Although claim 1 does not state this explicitly, the board takes it that the skilled person would understand the components, items and objects to relate to the modelled system.

7.2   That said, the board agrees with the examining division (see the decision, point 13.1 of the reasons) that no clear distinction can be made between "component", "item" and "object". The board takes the view that the skilled person would however understand these three terms broadly as being parts of the modelled system. Whether they are "real-world" or "virtual" would seem to depend on the nature of the system. The board

understands the application to mean that a robot system
has "real-world" parts and a business process "virtual"
ones. The long list of examples added to claim 1 of the
auxiliary request for clarification ("e.g., people,
machines, robots, [...]") does not change the scope of
the claim.

7.3   The examining division also took issue with the
        formulation that the binary data structures
        "represent[]" components, items, and objects (still
        point 13.1 of the reasons). The board regards the term
        "represent" as clear but broad. However, this breadth
        makes it impossible to attribute to the claimed
        invention any effect on any particular "real-world"
        system (see below).

7.4   Claim 1 refers to an "abstract description" of types of
        real-word objects in terms of "states", "attributes",
        "actions" and "relationships". Again, the board regards
        this language as clear but broad. However, on this
        reading, the aspects being modelled are entirely
        conventional ones, used in any object-based model.

7.5   Claim 1 refers to the "grouping" of components (etc.)
        "as relevant for a specific situation" (see the second
        "creating" step). It is noted that neither the
        "situations" in question nor the "relevance" to a
        situation is explained. While it would be reasonable
        for the skilled person to ignore the terms "relevant"
        and "specific situation" and reduce this feature to
        specifying that components are grouped, the board
        considers it to be unclear how the grouping contributes
        to the creation of a "virtual state space" as the
        claims require ("creating a virtual state space [...]
        by grouping them [...]").

7.6   The board also agrees with the decision under appeal
        that the term "binary data structures" is unclear.

7.6.1   Firstly, every data structure in computer memory is
        stored in binary form, i.e. as a binary "pattern". For
        instance, every C++ object is. It is noted, however,
        that storing the same object in different memory
        locations might produce different binary "patterns",
        e.g. if the data structure contains pointers.

7.6.2   The application discloses (see page 3, paragraph 2) -
        and claim 1 states - that the binary data structures
        are bit patterns "not described as compiled machine-
        code or interpreted logic language based on a Boolean
        logic with data stored in data-base tables or as a rule
        based expert or business process system that requires
        knowledge to be encoded in fact and belief knowledge
        rules". It is not clear what exactly this language
        excludes. Apart from the fact that it is doubtful
        whether the terms "logic language", "expert system",
        "business process system" and "knowledge rules" can be
        precisely defined, it is questionable whether the claim
        excludes, for instance, a logic programming language
        based on first-order logic (such as Prolog) or a system
        that works without data-base tables. Moreover, the
        board finds the exclusion of "compilation" or
        "interpretation" unclear. Claim 1 specifies that
        changes of status or attributes are received and stored
        "as a new version of the [...] data structure". Hence,
        it would seem that the new status and attribute values
        must somehow be mapped onto the data structure. In a
        broad sense, the board considers this to be a
        compilation step. Claim 1 also specifies that an action
        "related to a matched binary pattern" must be
        identified, which seems to imply that the binary data
        structures must retain sufficient structure to allow

the retrieval of a related action; it cannot be a
merely unstructured "bit pattern". In a broad sense,
the board considers this to be an interpretation step.
Claim 1 is thus self-contradictory.

7.6.3   The board consequently takes the view that the cited
         passage in the application (and in claim 1 of the
         auxiliary request) is unclear *per se* and thus unable to
         clarify the feature in question.

7.7   The board considers the term "uniquely identifiable" to
         be unclear, since every data structure in computer
         memory is identifiable, even uniquely so. In object-
         oriented systems, every "instance" of a class (or type)
         is "uniquely identifiable" as a matter of course. The
         board understands claim 1 to mean that all copies of a
         particular class instance, i.e. object, share some
         "identifier" (see page 26, paragraph 3). The claim
         language does not however express this. Apart from
         that, it is not clear from the claim what role the
         "copies" play in the overall invention.

7.8   The board also agrees with the examining division's
         finding that claim 1 is unclear because it specifies a
         "method for training" but no training step. The claim
         specifies aspects of the model (defining data
         structures and storing them, creating copies) and its
         use to determine and propose a "possible action" in
         response to some user input (analyzing similarities,
         searching and proposing), but not how the model is
         obtained by training. It is only stated that a "virtual
         state space" is "creat[ed]".

8.   In summary, the board agrees with the decision under
         appeal that claim 1 of both requests is unclear,
         Article 84 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