This document includes some recent decisions of the EPO in 2015 with regards to software related inventions and shows relevant extracts from the respective decisions.

T 1562/11 (Closing out white space/SAP) of 3.6.2015

European Case Law Identifier: ECLI:EP:BA:2015:T156211.20150603

Improving portal page personalization offering a direct manipulative window arrangement functionality

Inventive step - (no)
Efficient utilization of space within a graphical user interface - technical effect (no)

Application number: 07002856.8
IPC class: G06Q 10/00, G06F 17/30, G06F 3/048
Applicant name: SAP SE

Board: 3.5.01

http://www.epo.org/law-practice/case-law-appeals/pdf/t111562eu1.pdf

1. The invention

1.1 The invention concerns a graphical user interface called an "enterprise portal", which comprises "window panes" and "white space". The window panes may be tailored to a specific user role; the "white space" is unused space where new window panes might be added. The user can change the appearance of the portal by dragging a first window pane (e.g. "Employee Photo") from a first position to a second position.

However, the pane does not stay where the user left it. According to claim 1 of the main request, the first window pane is automatically shifted to a third position that is aligned with other window panes to "close out white space". In other words, the enterprise portal automatically eliminates the white space in-between window panes.

Claim 1 of the main request reads:

"A method comprising:

displaying an enterprise portal;

displaying role-specific window panes (104, 205, 206, 207) in the enterprise portal;

detecting a first position of a first window pane (104, 205);

detecting an input signal to drag-and-drop the first window pane (104, 205) to a second position, wherein a user drags the first window pane (104, 205) around the enterprise portal; and

displaying the first window pane (104, 205) in the second position where the user left the first window pane (104, 205);

characterized in, that the method further comprises

automatically shifting the first window pane (104, 205) in the second position to a third position where the first window pane (104, 205) is aligned with other window panes (206, 207) to close out white space (211, 400, 500)."

2. Main request, inventive step

2.1 The Examining Division considered D15 to be the most relevant state of the art vis-à-vis the subject-matter of claim 1, and the appellant agrees. The Board sees no reason to depart from this starting point for evaluating inventive step.

2.2 It is common ground that D15 discloses an enterprise portal having a drag-and-drop function, which allows the user to move window panes ("sheet objects") within the portal. There is also a "snap-to-grid" function, which aligns the objects that are being moved to a predefined design grid. As pointed out by the appellant, "snap-to-grid" does not unambiguously imply a move from the drop position to an aligned position. Nor does D15 disclose the closing-out of white space between window panes. Hence, the Board finds that document D15 does not disclose the following features of claim 1:

a) the first window pane is shifted to the third position after the user has left it in the second position;

b) the first window pane is shifted so as to close out white space.

2.3 **Feature (a) is not technical**, since it **provides no technical effect**. In particular, the Board does not follow the appellant's argument regarding a reduction in calculation. Claim 1 says nothing about what is and what is not calculated during drag-and-drop. There are **no clear implications as to resource requirements of the invention**, and so there is no basis for a comparison with the "snap-to-grid" method of D15.

In the Board's view, **any difference between the shift to a third position according to the invention and the "snap-to-grid" method of D15 is a matter of visual presentation relating to the "look-and-feel"** of the portal, and not a technical distinction.

2.4 Considering feature (b), the **Board does not accept that the arrangement of window panes according to the invention leads to better use of screen space in any technical sense**.

Firstly, since **claim 1 neither defines how the panes are aligned, nor their sizes**, **it covers inefficient usage of space**, e.g. large panes aligned in one column leaving only a narrow strip of white space on one side.

2.5 Secondly, the **Board does not consider more efficient usage of screen space to be a technical effect**. The Board rather agrees with the Examining Division that it **is a matter of layout design**. The arrangement of objects within available screen space does not involve any technical considerations of the computer screen, but follows the same layout principles as would apply to the layout of a magazine cover, for example. The "white space" is not a limited physical resource but rather part of the layout itself. In the Board's view, **the invention does not overcome any physical limitation of the computer screen.**

2.6 The **alleged improvement in readability of the displayed content is not an evident result of the claimed layout** of window panes without intervening space; nor has an improvement been substantiated by any real test or other evidence. **An improvement that resides merely in the cognitive process of reading is not a technical effect**, and is, therefore, irrelevant to inventive step. In this regard, the Board does not see T 49/04 as settled law.

2.7 T 49/04 itself declined to follow the earlier decision T 125/04 "Assessment system/COMPARATIVE VISUAL ASSESSMENTS". T 509/07, which the appellant cites as following T 49/04, did not do so; rather, it distinguished the case on the facts (point 5.2.2 of the reasons).

As the appellant pointed out***, T 49/04 was followed in T 1023/06 and T 1793/07, but then T 1143/06 specifically declined to follow T 49/04, as did a series of further decisions (see, for example, T 1575/07 "Managing maintenance/ACCENTURE, T 1741/08 "GUI layout/SAP", T 1214/09 "Information managing/SHARP").***

The present Board shares the view taken by the Board in T 1143/06 that a feature which relates to how cognitive content is conveyed to the user on a computer screen normally does not contribute to a technical solution to a technical problem. An exception would be if the manner of presentation could be shown to have a credible technical effect.

2.8 T 717/05 does not refer to T 49/04, but the appellant has sought to rely on it. It concerned the display of outcomes of games, which the Board considered to be technical as it provided information about the internal state of the gaming apparatus. The Board does not see how the factual situation aids the appellant, and notes that T 717/05 does not seem to set out a general point of law.

2.9 For the above reasons, **the Board does not consider that the invention contributes to the solution of a technical problem by providing a technical effect, and concludes that the subject-matter of claim 1 according to the main request lacks inventive step** (Article 56 EPC 1973).

T 0077/14 (Detection of scrolling gestures/PHILIPS) of 30.6.2015

European Case Law Identifier: ECLI:EP:BA:2015:T007714.20150630

Touch-screen image scrolling system and method

Inventive step (yes, after amendment): unobvious extension of detectable scrolling functions

Application number: 02790580.1
IPC class: G06F 3/033
Applicant name: Koninklijke Philips N.V.
Cited decisions: T 0641/00

Board: 3.5.05

http://www.epo.org/law-practice/case-law-appeals/pdf/t140077eu1.pdf

1.1.1 The present invention concerns a touch-screen device supporting gesture-based scrolling operations. According to the application, the problem to be solved by the claimed invention is to enable a user/viewer to access a desired portion of a long list of data by scrolling to the location of that portion rapidly and in a more natural manner.

1.1.2 Independent claims 1 and 8 are directed to an electronic touch-screen display device with scrolling capability configured to detect different gestures for data display and selection. In particular, process claim 8 includes the following features (as labelled by the board):

A method of controlling a scroll-like display of data on an electronic display screen, said method comprising the steps of:

a) sensing the speed and direction of motion of a finger-touch contact with said display screen having scrollable data displayed thereon;

b) sensing the duration of finger-touch contact time with said display screen;

c) if the sensed duration of finger-touch contact time is greater than a first preset minimum time and less than a second preset minimum time and is accompanied by motion along the surface of the display screen, moving said display in correspondence with movement of the finger touch, and - following separation of said finger touch from said display screen - initiating scrolling motion of said scrollable data on said display screen in said sensed direction and at said sensed speed;

d) upon sensing a finger touch during scrolling displacement of the image on said display screen, regardless of the length of the touch, stopping the motion of said display;

e) slowing the speed of said scrolling motion from the initiated speed thereof at a predetermined rate;

f) terminating said scrolling motion upon first sensing either a substantially stationary finger touch having a finite duration or an end-of-scroll signal;

g) if the sensed duration of said stationary finger-touch contact time is greater than a first preset minimum time and less than a second preset minimum time and is accompanied by motion of said finger touch along the surface of said display screen, and if, after subsequent moving of said display in correspondence with movement of the finger touch, there is no finger motion at the time that the finger contact with the display screen is broken, maintaining said display screen in the position it is at that time without further motion, and reverting the system to "waiting" status;

h) selecting an item touched if the sensed stationary duration of the finger-touch contact time is less than said second preset minimum time and if no motion occurs before separation of said finger from said display screen, wherein upon selection the selected item is highlighted.

1. MAIN (SOLE) REQUEST

Although this claim request was submitted for the first time during the oral proceedings before the board, i.e. at a very late stage in the overall procedure, the board admitted it into the appeal proceedings by virtue of Article 13(1) and (3) RPBA, since it was considered a legitimate and eventually successful attempt (see point 1.11.1 below) to overcome the objections raised by the board.

The main request differs from the claim requests refused by the examining division essentially in that present independent claims 1 and 8 now specify that (emphasis added by the board)

A) upon sensing a finger touch during scrolling displacement of the image on said display screen the motion of said display is stopped, regardless of the length of the touch;

B) if the sensed duration of a stationary finger touch contact is greater than a first preset minimum time and less than a second preset minimum time motion of said finger touch along the surface of said display screen, and if after subsequent moving of said display in correspondence with movement of the finger touch, there is no finger motion at the time when the finger contact with the display screen is broken, said display screen is maintained in the position it is at that time without further motion, and the system is reverted to "waiting" status;

C) if the sensed duration of a stationary finger touch contact said duration is less than said second preset minimum time and if no motion occurs before separation of said finger from said display screen, an item touched is selected and highlighted upon selection.

… the input gestures are made up of various ordered, sequence-sensitive touch events (like long/short touch, motion, separation, etc.) and are mapped to the corresponding output functions, thereby building the above "gesture matrix" of the underlying human-machine interaction scheme.

Document D7 teaches that the displayed part of a mobile terminal's touch screen is scrolled in the direction of the movement of the corresponding pointing means (e.g. finger) and at a rate proportional to the speed of that pointing means. Thus, in accordance with the finding of the decision under appeal, feature a) is known from D7. In addition, D7 states that an accelerated scrolling may be executed by touching the touch screen for a "longer" time. From this the

board concludes that, contrary to the finding in the appealed decision feature b) is also anticipated by D7.

As regards feature e) and the scrolling-termination criteria according to features d) and f), D7 teaches that list scrolling is retarded according to a predefined formula and is eventually stopped by touching the display control area or an element of the displayed list with the pointing means. Therefore and in the absence of a detailed and solid definition of an "end-of-scroll" signal, the board holds that features d) to f) are known from D7 as well.

1.1.5 However, the board finds that **D7 fails to directly and unambiguously disclose that the sensed duration of a touch contact within predetermined time limits is utilised to detect the gestures and perform the gesture-to-function mappings according to features c), g) and h)**. Consequently, the subject-matter of the present independent claims is found to be novel over D7.

1.1.7 As regards the alleged non-technical aspects of the objective problem formulated in the decision under appeal by applying the approach adopted in T 641/00 (OJ EPO 2003, 352; cf. headnote 2 and point 7 of the reasons), the board notes that the application itself is silent as to whether the selection of specific gesture-to-function mappings based on features c) to h) - out of a virtually infinite variety of devisable gesture-based functions - is predominantly based on human perception phenomena, as the examining division seems to suggest, or whether it rather depends on purely technical considerations relating to performance-related improvements of the touch-screen device in terms of e.g. its gesture recognition speed, accuracy and the like. Therefore, the board holds that **only through speculation could one derive from the application that the claimed gesture-to-function mappings are generally the mere result of non-technical considerations** (i.e. "attribution of meanings" as referred to in the appealed decision).

However, the board accepts that **distinguishing features c), g) and h) not only relate to the definition of gesture-to-function mappings as such** (i.e. corresponding to the second and fourth columns labelled "input" and "output" of the gesture matrix depicted in point 1.1.31.1.3 above) **but in particular and most importantly to actually enabling at the implementation level a proper recognition of and differentiation between three distinct scroll-related functions on a touch-screen device, namely scrolling initiation (relating to feature c)), scrolling interruption (corresponding to the "waiting" status according to feature g)) and data-item selection (relating to feature h))**. … **The parameter thus defined represents a further degree of freedom in the design of a gestural human-machine interface**, in addition to the already utilised parameters relating to the occurrence of a touch, its motion as well as the motion's speed and direction. The board concurs with the appellant that **thereby the number of distinct ordered (sequence-sensitive) touch events and feasible gesture-based functions may be significantly extended.** …

Moreover, even though the operation of data-item selection according to feature h) is not directly interlinked with the actual data-scrolling procedure itself, the board believes that it is well suited to readily selecting a particular data item after scrolling through a long data list according to feature c) or g). Hence, feature h) in fact **seamlessly supplements the scrolling operation for the purpose of an efficient data search and selection on the touch screen**.

1.1.8 In view of the above, **the objective technical problem** to be solved by present independent claims 1 and 10 may be formulated as **"how to extend the number of recognisable gesture-based functions in the context of scroll-based data list search on a touch-screen device** as given in D7".

1.1.9 … D7 does not provide any hint towards duration-based discrimination between a variety of further gesture-based functionalities as regards scroll-based data list search.

The board agrees with the examining division that the implementation of detecting the individual time limits via the concept of thresholds belongs to the common general knowledge of the skilled person in the field of user interface design. It is also true that prior-art document D5 demonstrates detecting different touch-duration intervals. Therefore, the board concedes that the implementation of detecting time-related thresholds as such **may** be rendered obvious by the combination of D7 and D5. However, in the board's judgment, **this does not suffice to cogently arrive at the claimed subject-matter.**

Rather, the board finds that actually **incorporating the additional detection of predetermined time intervals for the purpose of defining and applying gesture-based scrolling initiation, according to feature c), as well as additional gesture-to-function mappings, being directly or indirectly connected with the scrolling operation, according to features g) and h), goes beyond enabling mere differentiation between distinct gestures and their resulting functions at a conceptual level.** In particular, there is no discernible encouragement whatsoever in D7 or D5 to markedly extend the number of detectable combinations of ordered touch events and thus the latitude for feasible gesture-based output functions. Nor is there any pointer to leverage distinct touch-duration intervals in addition to the physical touch-detection parameters already involved. This holds true even in the light of the standardised usability principles as defined in D10, which merely provides some very general guidance about interface design in terms of effectiveness, efficiency and user satisfaction. In view of the increased implementation complexity and difficulties (e.g. relating to gesture recognition speed, conflict resolution, calibration, noise tolerance) resulting from extending the number of physical parameters to be detected, the board believes that the skilled person would rather be led away from the claimed solution, especially at the time of D7's publication (i.e. 1998). Accordingly, starting from D7 and in view of the amendments made to claim 1, its distinguishing features cannot be considered as straightforward implementation measures which the skilled person in the field of touch-screen interface design would, depending on circumstances, inevitably choose at the application's priority date, without the benefit of hindsight knowledge of the invention.

1.2 Thus, having regard to the cited prior art, the subject-matter of present independent claims 1 and 10 is held to be new and to involve an inventive step within the meaning of Article 52(1) EPC.

T 1339/09 (Presentation system/SAP) of 12.3.2015

European Case Law Identifier: ECLI:EP:BA:2015:T133909.20150312

Computer based presentation system

Inventive step - (no)

Application number: 02017776.2
IPC class: G06F 17/30
Applicant name: SAP SE

Board: 3.5.07

http://www.epo.org/law-practice/case-law-appeals/pdf/t091339eu1.pdf

The invention

The application relates to improved computer-based presentation.

The invention described in the application comprises a computer running a presentation application, a control device and a screen to display the presentation. The control device can be for example a keyboard or a remote control device with radio or infra-red link. The screen may be a display or a projector.

The presentation application processes a data set (e.g. a company name) to a presentation displaying processed data (e. g. bar charts). In some embodiments the data set is pre-stored.

A controller in the computer receives an activation trigger from the user, via the control device, and causes the presentation application to read the data set from intermediate storage, process it and display the presentation with the processed data.

Since the data sets are pre-stored, the presenter does not have to type data and is free to walk while doing the presentation if he uses a remote control device. Another advantage of the invention is that members of the audience can also choose data sets, as it is possible to pass the remote control device to them or to use more than one control device.

Claim 1 of the main request defines a method for operating a computer with an application, wherein the application processes a data set to a presentation. It specifies that the following steps are performed in sequence

- the computer displays a substantially empty input view,

- a controller receives an activation trigger from a user and reads the data set from an intermediate storage,

- the computer displays a filled input view filled with the received data set, wherein the received data is a predefined data set, and

- the computer displays a presentation with processed data.

Taking into account the description, the Board interprets the "received data set" as the data set, for example a company name, read in reaction to the activation trigger. The feature "the received data is a a predefined data set" encompasses "the data set is pre-recorded". The "processed data" is the data (e.g. bar charts) obtained by the presentation application ("application (200)" in the claim) after processing the data set. Even though it is not explicitly mentioned in the application, the Board assumes that the processed data may be for instance the bar charts corresponding to the company identified by the company name in the data set.

The Board notes that claim 1 does not define the control device and only specifies the method performed at the computer. It describes the invention in such vague and broad terms that its subject-matter is obvious over a method for operating a computer using the well-known PowerPoint application with the functionality described in document D4.

Document D4 therefore discloses an application (PowerPoint) which, upon receipt of an activation trigger by a user, reads and processes a data set, for example a link or an object, to processed data, for example a video (page 900) or data of an Excel file (page 910), and to a presentation with the processed data. The data set can be seen as being predefined or pre-stored, since the object is predefined in the presentation, for example by the link or the file name of the file containing the data (page 903, first four paragraphs). Even though it is not explicitly mentioned in document D4, a system running PowerPoint also includes a controller, because it detects activation triggers from a user.

Document D4 does not disclose a "substantially empty input view" and a "filled input view". **What is "substantially empty" is not defined in the claim and is subjective.** Any input view can be seen as empty in comparison to the next (filled) view on which data is presented. PowerPoint offers the necessary functionality to support such features, so that it is trivial to use it to create a presentation including an empty and a filled input view. Any PowerPoint slide accepting input from a user to display an object can be seen as "an empty input view". After the user input the slide is filled with a generated view of the object, corresponding to a "filled input view". Furthermore, it should be noted that **the fact that data is displayed in a previously empty view is a non-technical aspect related to presentation of information as such**.

Since the **claim does not define any particular technical features of an input view**, any PowerPoint slide can in principle be seen as an input view accepting input to control the transition to the next slide, the display of an object, or other effects (document D4, pages 921 and 922). The Board considers that a linked or embedded object in a PowerPoint presentation offers the same functionality as that described in the application, including automatic and dynamic updates.

The Board notes that **the claim does not define a real-time presentation of data,** since the data according to the claim could have been processed in advance. Additionally, in the opinion of the Board, adding dynamic processing of data to a known presentation system would be obvious. An inventive step could only rely on the implementation, but the claim does not define implementation details. Finally, PowerPoint also provides real-time presentation (for example of a video) and dynamic generation of a view such as a bar chart.

From the above reasoning, the Board concludes that the subject-matter of claim 1 of the main request does not involve an inventive step (Articles 52(1) and 56 EPC).

T 1368/12 (Abschätzung der Datenrate/BOSCH) of 6.7.2015

European Case Law Identifier: ECLI:EP:BA:2015:T136812.20150706

VERFAHREN ZUM AUSLESEN VON SENSORDATEN

Klarheit nach Änderung (ja)
Ausreichende Offenbarung der beanspruchten Erfindung - (ja)
Technische Wirkung - (ja)
Erfinderische Tätigkeit - (ja)

Anmeldenummer:	05777849.0
IPC-Klasse:	G06F 5/06, B60R 21/01
Name des Anmelders:	ROBERT BOSCH GMBH

Kammer: 3.5.06

http://www.epo.org/law-practice/case-law-appeals/pdf/t121368du1.pdf

Anspruch 1 des Antrags lautet wie folgt:

"Verfahren zum Auslesen von Sensordaten aus einem Zwischenspeicher (110), welche mit einer Datenrate (Tpas) von mindestens einem Sensor (10) in den Zwischenspeicher (110) geschrieben werden, wobei eine Abtastrate (Tsg) so gewählt wird, dass ein Überlaufen des Zwischenspeichers (110) vermieden wird und alle zwischengespeicherten Sensordaten mit der vorgegebenen Abtastrate (Tsg) in einen Steuergerätespeicher (130) eingelesen werden, dadurch gekennzeichnet, dass

der Zwischenspeicher (110) einen Meldung (RBE) erzeugt, wenn zum Abtastzeitpunkt keine neuen Sensordaten im Zwischenspeicher (110) vorhanden sind, wobei aus der Anzahl der in einem vorgegebenen Zeitraum ausgelesenen Sensordaten, der vorgegebenen Abtastrate (Tsg) und der Anzahl der in diesem Zeitraum empfangenen Meldungen (RBE) die Datenrate (Tpas) des Sensors (10) berechnet wird."

Klarheit, Artikel 84 EPÜ 1973

5. Die angefochtene Entscheidung stützt sich unter anderem auf die Feststellung, dass die auf Seite 6 der Beschreibung offenbarten Gleichungen nur für eine wenigstens "zeitweise konstante Datenrate" korrekt seien und daher die Konstanz der Datenrate ein wesentliches Merkmal sei, das im Anspruch nicht fehlen dürfe (vgl. Entscheidung, Gründe 1-1.2). Die Kammer folgt diesem Einwand nicht. Zum einen ist sie der Meinung, dass der Fachmann ohne Weiteres annehmen würde, dass für einen gegebenen Sensor die Datenrate im Rahmen der üblichen Toleranzen konstant ist. Darüber hinaus würde der Fachmann erkennen, dass die Güte der Abschätzung von den im Bezugszeitraum auftretenden, tatsächlichen Schwankungen der Datenrate abhängt und somit ebenfalls schwanken kann.

6. Anspruch 1 ergänzt die Begriffe der Datenrate und der Abtastrate mit den "Bezugszeichen" Tpas und Tsg, obgleich diese Formelzeichen nicht etwa Raten bezeichnen, sondern deren Inverses, also Perioden (s. Abb. 1-3). Die Kammer meint jedoch, dass **der Fachmann diese begriffliche Ungenauigkeit - die übrigens auch in der Beschreibung enthalten ist (bspw. S. 1, letzter Absatz, Zn. 4-6) - ohne Weiteres als solche erkennen würde** und dass daraus kein Mangel an Klarheit folgt.

Erfinderische Tätigkeit, Artikel 56 EPÜ 1973

8.1 D1 offenbart ein Messwertverarbeitungssystem (MVS7), in dem die von Sensoren in regelmäßigen Intervallen erfassten Daten in einen FIFO-Zwischenspeicher geschrieben und so dem HOST-Rechner zur Weiterverarbeitung zur Verfügung gestellt werden (Abschnitte 2.1-2.2, Abb. auf S. 3-4). D1 offenbart auch, dass die "Meßzeit [...] größer als die Zeit sein [solle], die zur Übernahme der Meßwerte in den HOST-Rechner benötigt wird", dass also die "Abtastrate" größer als die "Datenrate" gewählt werden solle, um einen Überlauf des Zwischenspeichers zu vermeiden (vgl. S. 4, Abschnitt 2.2, und S. 15, Abschnitt 7.2.2).

8.2 Anspruch 1 unterscheidet sich von D1 darin, dass

a) "eine Meldung (RBE) erzeugt [wird], wenn zum Abtastzeitpunkt keine neuen Sensordaten im Zwischenspeicher (110) vorhanden sind", und dass

b) aus der vorgegebenen Abtastrate und der Anzahl dieser Meldungen in einem Zeitraum die Datenrate berechnet (also abgeschätzt, vgl. Punkt 3.3) wird.

9. Unterschiedsmerkmal a) allein hat die Wirkung, Informationen über den Füllstand des Zwischenspeichers verfügbar zu machen und ist zu diesem Zweck im Stand der Technik bekannt (vgl. bspw. D3, S. 10, Abb. 10, Flag EF).

9.1 D1 offenbart (loc. cit.), dass die Abtastrate größer als die Datenrate sein solle, damit keine Messwerte verloren gehen. Gleichzeitig offenbart die Anmeldung als eine durch die Erfindung gelöste Aufgabe, die Verdoppelung von Sensordaten zu vermeiden (vgl. S. 2, letzter Abs., Zn. 1-6). Beides hat nach Ansicht der Kammer die offensichtliche Konsequenz, - dass von Zeit zu Zeit keine "neuen Sensordaten" im Zwischenspeicher vorliegen und es ist naheliegend, diesen Umstand durch eine entsprechende Meldung anzuzeigen, um Abtastfehler zu vermeiden.

9.2 Unterschiedsmerkmal a) allein kann somit nach Ansicht der Kammer einen erfinderischen Schritt gegenüber D1 nicht begründen.

10. Die Prüfungsabteilung war der Ansicht (vgl. Entscheidung, Gründe 4.3), dass die Unterschiede **a) und b) "zwei völlig unterschiedliche" Aufgaben lösten** und "**keine funktionelle Wechselwirkung**" erzielten. Die **Kammer kann sich dieser Einschätzung nicht anschließen**. Die gemäß Merkmal a) erzeugten RBE-Meldungen werden gemäß Merkmal b) gezählt und gehen so in die Abschätzung der unbekannten Datenrate ein. Daher ist die Kammer der Ansicht, dass Unterschiedsmerkmale **a) und b) eine gemeinsame Aufgabe lösen**.

11. Die angefochtene Entscheidung kam zu dem Schluss, dass die Berechnung unter anderem der Datenrate keinen erfinderischen Schritt aufweise, "da die **Ermittlung einer mathematischen Größe (ohne Weiterverwendung) keinen technischen Effekt erziel**[e] und somit gegenüber D1 keinen technischen Vorteil bewirk[e]" (Gründe 4.3.2).

11.1 Die Kammer ist hingegen der Meinung, dass die Unterschiedsmerkmale **a) und b) über die reine Ermittlung einer mathematischen Größe hinausgehen**. Das gilt nach Ansicht der Kammer schon für die Erzeugung der RBE-Meldungen, die in D1 nicht offenbart ist, aber auch für die Zählung und Verarbeitung der durch den Zwischenspeicher erzeugten RBE-Meldungen.

11.2 Die Kammer ist weiterhin der Ansicht, dass **die Bereitstellung der abgeschätzten Datenrate die "Güte" der abgetasteten Sensorsignale gemessen an der erfassten Realität steigert und somit zur Lösung der Aufgabe beiträgt**, den Präzisionsverlusten zu begegnen, die durch Unkenntnis des verwendeten Sensors, durch die Verwendung eines weniger genauen und daher günstigeren Sensors, oder aus anderen Gründen wie etwa erhöhte Temperaturschwankungen entstehen (vgl. S. 1., 2. Abs., Zn. 4-6; S. 3, vorletzter Abs., Zn. 10-13). Die Kammer ist der Ansicht, dass es sich **dabei um eine technische Aufgabe handelt**.

11.3 Die Kammer kommt daher zu dem Schluss, dass im Kontext des **beanspruchten, über einen Zwischenspeicher vermittelten, Auslesens von Sensordaten die Bereitstellung und Verarbeitung der RBE-Meldungen zur Bestimmung des unbekannten Sensorparameters "Datenrate" eine technische Wirkung hat und eine technische Aufgabe löst**.

12. Weder D1 noch D3 enthalten einen Hinweis darauf, RBE-Meldungen in einem vorgegebenen Zeitraum zu zählen und auf dieser Basis die Datenrate des Sensors abzuschätzen. Unterschiedsmerkmale a) und b) sind daher durch den vorliegenden Stand der Technik nicht nahegelegt. **Die Kammer weist darauf hin, dass die konkreten Berechnungsschritte zur Abschätzung der Datenrate nicht beansprucht sind und es somit nicht entscheidungserheblich ist, ob diese Berechnungsschritte selbst naheliegen oder nicht.** Insgesamt kommt die Kammer zu dem Ergebnis, dass die beanspruchte Erfindung gegenüber dem vorliegenden Stand der Technik den unter Artikel 56 EPÜ 1973 erforderlichen erfinderischen Schritt aufweist.

T 2349/13 (Modelling heart states/GRIPPING HEART) of 14.7.2015

European Case Law Identifier: ECLI:EP:BA:2015:T234913.20150714

State machine user and validation interface system

Clarity - (no)
Sufficiency of disclosure - (no)

Application number: 08857700.2

IPC class: A61B 5/02, G06F 19/00, G09B 23/28
Applicant name: Gripping Heart AB

Board: 3.5.05

http://www.epo.org/law-practice/case-law-appeals/pdf/t132349eu1.pdf

Claim 1 of the main request reads as follows:

"State machine interface system comprising an input means, a processing means and a graphical user interface, wherein said input means is adapted to receive signals from at least one sensor device, wherein said signals are related to physiological activities of the heart and/or the circulatory system of a living being and are transformed to time related triggering points, and wherein said input means is adapted to apply said triggering points to said processing means which is adapted to use state machine analyzer algorithms to determine phases of heart cycles based upon said signals including said triggering points, wherein said different phases of the heart cycle are determined by said state machine algorithms in a heart cluster state machine simulating the heart, and optionally the circulatory system, achieved by fusions of finite heart muscle cell state machines to form a DELTAV-pump state machine,

characterized in that said processing means is adapted to:

- evaluate said determined heart cycle phases by determining their respective local state diagram based upon said signals and determine the respective time duration for each heart cycle phase,

- determine the statistically most representative global state diagram, made up by said local state diagrams, and to

- present said determined local and global state diagrams at the graphical user interface such that the temporal relations between the different phases are illustrated,

wherein the heart cycle phases are graphically presented as one or several graphical illustrations, e.g. overlapping circle diagrams, rings or bars, presenting different activities of the heart and circulatory system at one to several locations, arranged as state diagrams, where the phases are represented as time segments with lengths depending on the duration of the respective phase, and wherein each presented heart cycle phase, and/or sub-part of heart cycle phase, has been assigned related values from said input signals and/or other related signals, and wherein the displayed information is continuously updated in real-time."

2.2 Articles 84 and 83 EPC

The board holds that claim 1 of both claim requests does not meet the requirements of Article 84 and/or 83 EPC either, for the following reasons:

2.2.1 The term "**triggering points**" is **not clear**, since the claim and **the description fail to reveal what should be actually triggered** by the heart-related measurement points applied to the processing means.

2.2.2 Moreover, it is **unclear** what the "**local state diagram**(s)" really are and **how they are supposed to be established and presented**. In this regard, the description as filed merely states that they are "registered from one or more sites", while the statement setting out the grounds of appeal indicates that the "local state diagrams are put together to a global state diagram". Also, the claim and the **description are silent as to which properties exactly qualify a global state diagram as the "most representative global state diagram" and how it ought to be determined, i.e. what kind of statistics should be applied**.

2.2.3 Also, the board finds that it is **unclear** to what "**related values**" and "**said established values**" actually refer and what "**individual and/or global related values" are meant to be**.

2.2.4 Lastly, as regards the feature that "the displayed information is **continuously updated in real-time**", **real-time updating of the displayed data is nowhere disclosed in the description, nor is it clear how this is supposed to be achieved according to the present invention**.

2.3 In conclusion, neither the main nor the auxiliary request is allowable under (at least) Articles 123(2), 84 and 83 EPC.

T 1958/13 (Single-drag gesture/LG) of 12.6.2015

European Case Law Identifier: ECLI:EP:BA:2015:T195813.20150612

Terminal and method for entering command in the terminal

Inventive step - (no): alternative gesture definition

Application number:	07002223.1
IPC class:	G06F 3/048
Applicant name:	LG Electronics Inc.
Cited decisions:	T 0643/00, T 0482/02, T 1284/04, T 1567/05, T 1841/06, T 1900/09, T 1192/10, T 0407/11

Board: 3.5.05

http://www.epo.org/law-practice/case-law-appeals/pdf/t131958eu1.pdf

Claim 1 of the main request reads as follows:

"A terminal, comprising:

a touch-screen display configured to be touched by a user, and

a controller configured to determine a portion of the touch-screen display that is touched during a dragging motion by the user,

characterized in that the controller is configured

- to recognize the dragging motion in which the user touch-drags from a first position to a second position on the touch-screen display,

- to determine a direction of the dragging motion on the touch-screen display, and

- to perform a delete or cut operation based on the direction of the dragging motion on the touch-screen display in order to delete or cut the text data between the first and second positions."

2.1 Article 84 EPC 1973: support by the description

2.1.1 The examining division held that the claims were not supported by the description, since the breadth of the term "direction" was unjustified in view of the allegedly unclear description of detecting the "up or down direction" according to paragraph [0046] of the application as filed (cf. appealed decision, sections 4 and 11).

2.1.2 The **board**, however, holds that this **objection is unfounded** because **the term "direction", although broad, is not found to be unsupported or unclear in view of the corresponding teaching of the application as filed** (cf. [0046]: "... the user can delete data by dragging the pointer in a left to right direction, and cut and paste data by dragging the pointer in a right to left direction. An up or down direction, etc. may also be used ...").

2.2 Article 52(1) EPC: novelty and inventive step

The board judges that claim 1 lacks an inventive step, for the following reasons:

2.2.3 Document D2 teaches that the text data between the start point and the end location (corresponding to the "first position" as claimed) of the initial stroke ("stroke 60"), rather than the subsequent stroke, is supposed to be copied or cut (see D2,[0029] and [0033] in conjunction with Figs. 6 and 9). Therefore, the board agrees with the examining division and the appellant that D2 fails to disclose that

i) the text data within an area between the first and the second position of the dragging motion is cut or deleted based on the direction of the dragging motion.

2.2.4 From distinguishing feature i)i) it follows that the drag gesture according to claim 1, i.e. dragging from the beginning of the text to be removed until the end of it in a certain direction, is made up of a single stroke. By contrast, the drag operation according to D2, i.e. first dragging from the beginning of the respective text to the end of it in a certain direction (i.e. to the right) and then dragging from the end of that text in a different direction (i.e. upwards or downwards), relies essentially on two strokes.

2.2.5 The appellant argued at the oral proceedings before the board that the so-called "single-drag gesture" according to feature i)i) had the effect of making text editing more convenient or simpler for the user and thus better compared to the solution of D2.

However, the board does not consider the alleged effects attributed to distinguishing feature i)i) like **simplifying the user's operation** (see also appealed decision, page 5, fourth paragraph), **improving the user experience or providing more user-convenient text editing functions** (see also statement setting out the grounds of appeal, page 3, penultimate paragraph) to be persuasive. Although **the board deems those effects, in principle, to be technical effects, since in the end they aim at providing tools which serve or assist user activities** (see e.g. T 643/00 of 16 October 2003, point 16), in the present case the question **whether they are actually achieved depends exclusively on subjective user skills or preferences**. Therefore, the board is not satisfied that they may be regarded as objectively credible technical effects for the purpose of formulating the objective problem to be solved (cf. T 1567/05 of 30 April 2008, point 3.6; T 1841/06 of 21 January 2011, point 5, third paragraph; T 407/11 of 10 April 2014, point 2.1.4). For example, one user would prefer to delete or cut a certain text as fast as possible, without worrying about a possible error in selecting the text to be removed and thus an unintended deletion. For this user the single-drag gesture would be appropriate. Another user would however be more concerned with the precise selection of the text before actually removing that text. Such a user would rather opt for a two-stroke drag operation in order to be able to check beforehand whether the text to be removed has been correctly selected.

2.2.6 Furthermore, it is apparent to the board that feature i)i) constitutes a direct consequence of the definition of the gesture/function mapping as illustrated in point 2.2.12.2.1 above. The view of the appellant that the new gesture itself improved the system of D2 (cf. appellant's letter dated 12 May 2015, page 3, third paragraph) does not convince the board. The board rather holds that, unless the type of graphical user interface (GUI) technology and its application scenarios used are decisive for the definition of specific gestures, gestures are primarily aimed at a user familiar with the basic computer interaction concepts. Consequently, they are typically defined based on experimental or empirical studies on test users with the aim of reducing the user's cognitive load, or provide for intuitive gesture/function mappings, rather than being directed to purely implementational improvements of a touch-screen device relating e.g. to its processing load, its gesture recognition speed or the like. Also in the present case, there is no solid indication that the single-drag gesture claimed is actually devised for achieving device-specific and/or performance-oriented improvements at the implementation level of a touch-screen device, nor are there any details derivable from claim 1 or the present description with regard to the implementation of its recognition (as opposed e.g. to the cases underlying T 1900/09 of 15 March 2013 and T 1192/10 of 7 April 2014).

In view of the above, the board regards **such gesture definition - whether deemed to be technical or non-technical - as a preliminarily obtained precondition, i.e. a user-specific fact, to be taken into account in the user interface design as belonging to the conception or motivation phase normally preceding an invention** (see e.g. T 482/02 of 13 December 2005, point 5.3; T 1284/04 of 7 March 2007, point 3.1, second paragraph). However, the board agrees with the appellant that, in the present case, **incorporating the very specific gesture type of the claimed invention into the objective problem would include a clear pointer to the solution and inevitably amount to an ex-post-facto analysis.**

The board formulates the **objective problem to be solved by claim 1 as "how to implement an alternative direction-sensitive gesture for removing text on the touch-screen system of D2"**.

Concerning the mapping of the function "delete" to the drag gesture according to feature F)F) of claim 1, the board agrees with the finding of the decision under appeal (see section 10.1) that this represents a straightforward alternative to the "copy" operation based on the detected stroke direction, as taught in D2 (see e.g. [0029] and [0033] in conjunction with Figs. 6 and 9) to properly implement the given gesture definition.

Hence, in the present circumstances, applying a retention period in D2 for triggering a "delete" function does not lead the skilled person away from using a single-drag gesture, whether or not D2 relies on a two-step or three-step approach. **In summary, the pure definition of a gesture such as the present "single-drag gesture" at a conceptual level only, without addressing implementation details as to its recognition, cannot contribute to an inventive step.**

T 1211/12 (Compiling concurrent object-oriented programs/MICROSOFT) of 1.7.2015

European Case Law Identifier: ECLI:EP:BA:2015:T121112.20150701

Implementation of concurrent programs in object-oriented languages

Claims - clarity (no)
Sufficiency of disclosure - (no)

Applicant name: Microsoft Technology Licensing, LLC
Application number: 05106321.2
IPC class: G06F 9/52, G06F 9/54, G06F 9/46, G06F 9/44

Board: 3.5.06

http://www.epo.org/law-practice/case-law-appeals/pdf/t121211eu1.pdf

The invention

1. According to its initial paragraph, the application relates to "language extensions in an object oriented environment to support asynchronous programming through message passing, contracts, and orchestration". The application explains that shared-memory communication as used in certain object-oriented frameworks is "one of the main obstacles to simple support for concurrency" and proposes as a solution to incorporate asynchronous message passing in such object-oriented environment.

1.1 The proposed solution is based on several "services", each running in its own thread, which communicate with each other according to message-based interfaces based on "contracts". A contract is defined as "a formal specification of the allowable sequences of invocation of the members of an interface". Contract declarations in the program code are, at run-time, represented by non-deterministic finite state machines.

1.2 The contracts are said to be enforced at run-time by "validat[ing] method invocations ... against a contract specification" - presumably producing some sort of error message if a method call cannot be so validated. Alternatively, it is disclosed that contracts may be enforced at compile-time.

1.3 Another aspect of the invention is referred to in the description as "orchestration", which is said to encompass "the collection of mechanisms for coordinating communication between concurrent services". An "orchestration component" is disclosed comprising what is called a "schedule component" and a "compiler component" (see fig. 6). The compiler is said to "[break] the co-routine-based code down into pieces to allow parallel waits to occur without blocking thread context". Elsewhere, the compiler is disclosed as producing the schedule component as a "runtime object" - with no "source-code manifestation" - that is used to allow message-oriented code to wait for more than one message in parallel".

1.4 In an appendix, the application contains a scientific paper which appears to provide a theoretical basis for aspects of the present application. The paper refers to model checking to verify that a "message-passing program" conforms to its "contract" and is thus "stuck-free", in that it "cannot deadlock waiting for messages that are never sent or send messages that are never received". Contracts appear to correspond to the ones described in the application. The appendix also reports on a study in which a service implementation was automatically detected not to conform to its contract specification in various ways.

Claim 1 of auxiliary request "3" reads as follows:

"A computer system employing language extensions in an object-oriented environment (100) that supports concurrency to [sic] object-oriented languages, via message passing to/from services (108) in the object-oriented environment (100), wherein a service executes its own algorithmic thread and does not share state with any code outside the service the system comprising:

a contract component (104) being defined as interface declarations for asynchronous message passing for managing communication between multiple services (108) simultaneously, the contract component including:

a message component (302) specifying a set of messages; and

a protocol component (304) specifying allowable sequences of message exchange;

a component (106) for the multiple services configured to facilitate the handling of multiple messages and multiple message targets, the component comprising:

a compiler component (604) configured to generate a schedule for the messages in accordance with the set of messages and the allowable sequences, wherein the schedule is a runtime object that allows message-oriented code to wait for more than one message in parallel."

Clarity, Article 84 EPC 1973

2. The board is of the opinion that the wording of claim 1 of all requests is fundamentally unclear, in particular as regards the nature and functionality of the claimed "schedule" and how the schedule is generated by the claimed "compiler".

3. The board first notes that the term "schedule" has no unique established meaning in the relevant art of concurrent and distributed programming languages, environments and compilers. This also applies to the term "implementation schedule" as used in auxiliary request 5.

3.1 The "schedule" is claimed as being a "runtime object" (auxiliary request 3, claim 1, penult. line) and disclosed as having no "source-code manifestation" (p. 37, last para.). It is claimed as being generated by a compiler "in accordance with" the contract component, in particular with "the set of messages and the allowable sequences". This language suggests, in conformity with the description, that the schedule is a compiler product meant to aid "enforcement" or "validation" of contracts (see p. 3, 3rd line from the bottom - p. 4, line 3).

3.2 The claims do not specify the desired behaviour of the schedule component at run-time, nor how it is intended to represent, enforce or validate the contract component.

3.2.1 It is noted that a "contract" may be breached by

a) the fact that a message is sent which is not "allowed" at a certain point in time, after a certain sequence of messages, or by the fact that

b) none of the possible messages allowed (and expected) at a particular point in time arrive.

3.2.2 Accordingly, "enforcing" or "validating" a contract can mean different things.

3.2.3 An obvious breach of contract is the sending of a message which is not allowed at a particular point in time. If a service receives such a message, an error message might be produced or an interrupt raised. The service might then process the message nonetheless (if it provides the corresponding method) or it may refuse to process it. In the latter case the sender of the message might be blocked ("stuck") waiting for a response, even though the deviation from the contract was detected.

3.2.4 Another possible breach of contract is that an expected or prescribed message is not received by a service so that the service is blocked waiting for that message, whether or not this is a response message. In this case, it must be defined when this non-compliance is said to occur. The description states that even a long delay of an expected message does not "technically" constitute a breach of contract (p. 64, lines 7-9) but that missing messages constitute a violation of the contract only when the entire schedule is terminated. The board notes that some services are meant to run forever. On the assumption that the "schedule" of

such a service never terminates, such a service will, from this perspective, never exhibit a breach of contract even when it is blocked ("stuck") waiting for a message that never arrives.

3.3 Moreover, it may be possible that the compliance of services with a contract can be determined at compile-time (see description, p. 28, lines 4-5). The board notes that this may not be generally possible (for fundamental reasons relating to what is known as the "halting problem") and that the description gives no indication as to the circumstances under which compile-time validation is possible and/or meant to be performed. Moreover, everything that has been checked at compile-time need not be checked at run-time any more. The skilled person would thus assume that the runtime object "schedule" only performs checks which have not already been carried out at compile-time.

3.4 Claim 1 of all requests does not specify any of the above, be it directly as claimed properties of the schedule or indirectly by way of claimed properties of the "compiler component". As a consequence, it is **entirely unclear what the schedule component is meant to do at run-time and thus in what way it is "generated [...] in accordance with the set of messages and the allowable sequences**" ….

4. Claim 1 of all requests specifies that "the schedule component [...] allows message-oriented code to wait for more than one message in parallel".

4.1 The board considers that **this phrase is ambiguous**. Firstly, code comprising multiple threads will typically have, at any point in time, some threads which are ready to proceed and other threads which block while waiting for some message to arrive. The latter threads wait "in parallel". Secondly, any concurrent object must be prepared to process any message corresponding to its methods. Hence, any idle concurrent object providing at least two methods can be said to be waiting "in parallel". Thirdly, according to the claims it is the "schedule", generated "in accordance with" the contract component, which "allows [...] code to wait [...] in parallel"; this appears to suggest that parallel waiting is only allowed to the extent that the protocol part of the contract component specifies it.

4.2 The wording of **claim 1 of all requests leaves open which of the three interpretations is the intended one**, and this also renders the claim unclear.

4.3 The board further notes that the **inventive contribution** which the "parallel wait" feature **may possibly make depends significantly on which of these three interpretations is chosen**. As suggested above, the possibility of threads waiting in parallel appears to be implicit in any multi-threaded message-based system, the possibility of objects waiting in parallel appears to be implicit in any concurrent object-oriented system, and the possibility for a service to wait in parallel for only those messages which the contract component happens to allow appears to follow from the fact that the specified contract should be "enforced" at run-time. While in the first two cases no dedicated compiler support might be needed at all to "allow [...] waiting in parallel", in the third case such compiler support is needed but not claimed.