

Internal distribution code:

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

**Datasheet for the decision
of 16 September 2015**

Case Number: T 0868/11 - 3.5.06

Application Number: 02750108.9

Publication Number: 1461692

IPC: G06F9/44

Language of the proceedings: EN

Title of invention:

GRAPHICALLY CONFIGURING PROGRAM INVOCATION RELATIONSHIPS

Patent Proprietor:

National Instruments Corporation

Opponent:

i f m electronic gmbh

Headword:

Program deployment GUI/National Instruments Corporation

Relevant legal provisions:

EPC 1973 Art. 54, 56, 84, 100(a)

EPC Art. 123(2)

Keyword:

Novelty - main request (no)

Inventive step - auxiliary requests (no)

Amendments - added subject-matter (no)

Claims - clarity - first auxiliary request (no)

Decisions cited:

Catchword:



**Beschwerdekammern
Boards of Appeal
Chambres de recours**

European Patent Office
D-80298 MUNICH
GERMANY
Tel. +49 (0) 89 2399-0
Fax +49 (0) 89 2399-4465

Case Number: T 0868/11 - 3.5.06

D E C I S I O N
of Technical Board of Appeal 3.5.06
of 16 September 2015

Appellant: National Instruments Corporation
(Patent Proprietor) 11500 North MoPac Expressway
Austin, Texas 78759-3504 (US)

Representative: Heselberger, Johannes
Bardehle Pagenberg Partnerschaft mbB
Patentanwälte, Rechtsanwälte
Postfach 86 06 20
81633 München (DE)

Respondent: i f m electronic gmbh
(Opponent) Teichstrasse 4
45127 ESSEN (DE)

Representative: Gesthuysen Patent- und Rechtsanwälte
Patentanwälte
Postfach 10 13 54
45013 Essen (DE)

Decision under appeal: **Decision of the Opposition Division of the
European Patent Office posted on 9 February 2011
revoking European patent No. 1461692 pursuant to
Article 101(3) (b) EPC.**

Composition of the Board:

Chairman W. Sekretaruk
Members: S. Krischer
M. Müller

Summary of Facts and Submissions

I. This appeal by the patent proprietor is directed against the decision of the opposition division, posted on 9 February 2011, to revoke the European patent EP 1 461 692 for lack of inventive step of the main request (Article 100(a) EPC 1973) and of auxiliary requests 1 and 2 (Article 56 EPC 1973) over D1:

D1 US 5 999 178 A, 7 December 1999.

II. A notice of appeal was received on 15 April 2011. The appeal fee was received the same day. A statement of the grounds of appeal was received on 15 June 2011.

III. The proprietor (appellant) requests that the decision be set aside and the patent be maintained as granted (main request) or in amended form based on claims 1-39 of the auxiliary request filed with the grounds of appeal or based on claims 1-35 of the second auxiliary request with letter of 24 July 2015, the further text on file being description paragraphs [1]-[366] and drawing pages 61-104 of the patent.

IV. The opponent (respondent) requests that the appeal be dismissed.

V. In its summons to oral proceedings, the board gave reasons for its preliminary opinion that claim 1 of the main request lacked novelty (Articles 54 and 100(a) EPC 1973) and claim 1 of the first auxiliary request was originally disclosed (Article 123(2) EPC), but not clear (Article 84 EPC 1973) and not inventive (Article 56 EPC 1973).

VI. In a letter dated 24 July 2015, the proprietor filed a second auxiliary request, maintained the main and the first auxiliary request, withdrew its request for oral proceedings, requested a decision "on the stand of the file" without oral proceedings, and announced that in any case it would not be represented at oral proceedings.

VII. In a communication dated 10 August 2015, the board informed the parties that the oral proceedings had been cancelled.

VIII. Claim 1 of the main request reads as follows (with the numbering of section 13 of the appealed decision added by the board in square brackets):

"1. [1] A method for deploying a program in a system, wherein the method executes on a first computer system (82), the method comprising:

[1.1] displaying a plurality of program icons (402, 404) on a display of the first computer system (82),

[1.1.1] wherein each of the program icons (402, 404) corresponds to at least one program;

[1.2] displaying a device icon (412) on the display of the first computer system (82),

[1.2.1] wherein the device icon (412) corresponds to a device in the system,

[1.2.2] wherein the device is coupled to the first computer system (82);

characterized in that, the method also comprising:

[1.3] associating a first program icon (404) of the plurality of program icons (402, 404) with the device icon (412) in response to user input,

[1.3.1] wherein the first program icon (404) corresponds to a first program in the application,

[1.3.2] wherein the first program is configured to be invoked by a second program;

[1.4] deploying the first program onto the device in response to said associating; and

[1.5] automatically modifying the second program to invoke the first program on the device in response to said associating;

[1.5.1] wherein during execution the second program is operable to invoke the first program on the device; and

[1.5.2] wherein the first program executes on the device after being invoked by the second program."

IX. Claim 1 of the first auxiliary request reads as follows:

"1. A method for graphically configuring program invocation relationships and deploying a program in a distributed computer system using a graphical program development environment, wherein the method executes on a first computer system (82), the method comprising:

displaying a plurality of program icons (402, 404) in a graphical user interface of the graphical program development environment on a display of the first computer system (82), wherein each of the program icons (402, 404) corresponds to at least one program;

displaying a device icon (412) in the graphical user interface of the graphical program development environment on the display of the first computer system (82), wherein the device icon (412) corresponds to a programmable device in the distributed computer system, wherein the programmable device is coupled to the first computer system (82) and wherein the

programmable device includes at least one or more of a processor and memory or a programmable hardware element or reconfigurable logic;

characterized in that, the method also comprising:

drawing a link between two program icons (402, 404) to configure an invocation relationship between a first program icon (404) and a second program icon (402) to modify a second program represented by the second program icon (402) to invoke a first program represented by the first program icon (404);

graphically associating the first program icon (404) of the plurality of program icons (402, 404) with the device icon (412) in response to user input, wherein the first program icon (404) corresponds to a first program in the application, wherein the first program is configured to be invoked by a second program;

deploying the first program onto the programmable device in response to said associating; and

automatically modifying the second program to invoke the first program on the programmable device in response to said associating;

wherein during execution the second program is operable to invoke the first program on the programmable device; and

wherein the first program executes on the programmable device after being invoked by the second program."

- X. Claim 1 of the second auxiliary request differs from claim 1 of the first auxiliary request in that the step of drawing a link additionally includes the expression "by a user" and the step of deploying now reads:

"deploying the first program onto the programmable device in response to said associating, wherein deploying comprises transferring the first program onto the programmable device and transferring one or more other programs onto the programmable device, wherein the one or more other programs are called by the first program during execution; and".

Reasons for the Decision

1. *Overview of the invention*

The invention relates to a graphical user interface (GUI) for allowing a user to input instructions about the deployment of programs in a distributed computer system (section [1] of the patent). The distributed system may be a measurement and automation system ([3]; figures 2A, 2B), e.g. controlled by the LabVIEW program of the proprietor ([6]). Different alternatives of deployment are disclosed in [123]. In the context of the claims, deployment essentially means transferring a program via a network from a first computer ((82); e.g. in figures 2A and 3) to a "device" in order to be remotely invoked from the first computer on the device (claim 1 of the patent; [123], line 33, alternative 3)). The device can be a (second) computer or a reconfigurable circuit like an FPGA ([52]). Correspondingly, the "program" can be a computer program or a bit file for an FPGA ([57]).

In order to instruct the first computer as to which device it should deploy the program on, the user can drag and drop in the GUI of the first computer ([88]) a

program icon onto a device icon of the distributed system in order to create an "association" between them ([122], first and third sentence; figure 6 (208)). The program is then deployed (i.e. transferred) to the associated device.

Figure 16 shows the drag and drop operation to create an association: the main program icon 402 (with four sub-programs 404a-404d; see [239]-[241]) is dragged and dropped onto device icon 412 (computer "Shah") and becomes in figure 17 the main program 402a deployed on "Shah" (i.e. copied to it; see [242], first sentence).

According to the dependent claims of the main request and in all the claims of the auxiliary requests, a *second type of association* can be entered (also via drag and drop; [122], second sentence; figure 6 (208), 2); see also "drawing a link between two program icons" in the claims concerned), namely an *invocation relationship* between the two programs. This means that a first program is executed on the associated device *after* being invoked by a second program. The claims do not state which computer the second program is executed on.

2. *Overview of the decision*
- 2.1 Claim 1 of the main request (i.e. the patent) lacks novelty (Articles 54 and 100(a) EPC 1973) over D1.
- 2.2 Claim 1 of the first and second auxiliary requests are originally disclosed (Article 123(2) EPC), but not inventive (Article 56 EPC 1973).
- 2.3 Claim 1 of the first auxiliary request is not clear (Article 84 EPC 1973).

Main request

3. *Novelty and inventiveness*

3.1 The closest prior art document D1 discloses that a first program icon (see figure 3C: the icon 350 with the caption "DiskUtil") is "associated" with a device icon "in response to user input" (column 5, lines 44-45: drag & drop DiskUtil icon 350 onto Computer-1 icon 340). The first program corresponding to the first program icon 350 is deployed onto the associated device (column 8, lines 23-28, 47-49: the "ExecProgram" corresponding to the DiskUtil icon 350 is sent to any computers which have been selected by drag&drop) and executed thereon (lines 49-51).

3.2 The grounds of appeal (page 6, paragraph 2) express the opinion that the preamble of claim 1 (i.e. features 1, 1.1, 1.1.1, 1.2, 1.2.1 and 1.2.2 in the numbering of the appealed decision; see section VIII above) of the main request is disclosed in D1. The board agrees.

3.3 According to the decision (25., 26., 28.), the claim only differs from D1 in that the claim specifies the device on which the first program is deployed and executed (namely on the associated device), whereas D1 does not disclose on which device the first program ("DiskUtil" in D1) is deployed and executed. This difference thus concerns the device in features 1.4 and 1.5.

3.4 As to feature 1.3.2, the proprietor argues in its letter of 11 January 2012 (page 5, paragraphs 3 and 4, last sentence) that the first program "DiskUtil" of D1 "should be configured to be invoked ... by the

AppManager program" (i.e. the second program) and that this was not the case in D1.

However, the board cannot see in what way such a "configuration to be invoked" should concern the invoked program. In the case of a computer program, the board is not aware of any configuration in the invoked program necessary so that it can be invoked by another program.

Thus, the board construes feature 1.3.2 as "wherein the first program is invoked by the second program" and thus regards it as being implied by feature 1.5.

3.5 The board further considers that (in the numbering of the decision):

- 1.3.1 follows from feature 1.1.1, and
- 1.5.1 and 1.5.2 are implied by feature 1.5, when properly construed.

3.6 Therefore, the board only has to discuss features 1.3, 1.4 and 1.5 of the characterising portion of the claim.

3.7 As to feature 1.3, the grounds of appeal (page 6, paragraph 4) argue that, since the first program was not deployed in D1, the related association in D1 was different to that in the claim. However, as shown below, the board takes the view that D1 discloses a deployment of the first program. Therefore, this argument fails.

3.8 As to features 1.4 and 1.5, the opponent argues in its letter of 18 November 2011 (page 5, last paragraph to page 7, second paragraph) that figure 5 and column 8,

lines 23-53 of D1 disclose the deployment and the execution on the associated device. Reference is made to D1, column 8, lines 47-51, which reads:

"The ExecPrograms are then sent to the machines having the resource objects referenced in their respective CopyPrograms. Finally, in stage 516 the ExecPrograms are executed on the machines and the operation terminates."

Furthermore, figure 3C discloses that the first program ("DiskUtil" in D1) is associated with computer 1 and that it follows from the above that this program is transferred to (i.e. deployed on) this computer and executed by it (opponent's letter, page 6, paragraph bridging pages 6 and 7).

- 3.9 The proprietor argues in its letter of 11 January 2012 (page 6, paragraph 5 to page 7, first paragraph) that the above passage in D1 merely discloses that a modified copy of the first program "DiskUtil" (and not the program itself) is sent to a machine, i.e. computer 1, rather than to the selected resource object or device, namely the hard disk of computer 1. The transferred program might be only stored in the memory of the computer and not on its hard disk.
- 3.10 The board follows the arguments of the opponent in that the above cited passage in D1, column 8, lines 47-51 indeed discloses that a first program is sent to its associated devices to be executed on them.
- 3.11 The board agrees with the proprietor that only a modified copy of the DiskUtil program, the ExecProgram, is sent to the computer. However, the board considers

that this ExecProgram must be considered to "correspond" to the first program icon as the claim requires, and therefore identifies the claimed "first program" with the ExecProgram rather than the original DiskUtil program (see point 3.1 above). On this reading, D1 discloses that the first program is transferred.

3.12 The board also agrees with the proprietor that the program may not be transferred to the hard disk of the target computer. Therefore, the question to be answered is whether deploying necessarily includes storing the program on the hard disk of the associated device or whether it is sufficient to transmit the program to the target computer and store it in its memory. To answer this question, it is necessary to look at section [123] in the patent, which explains "deploying". In the whole section, there seems to be no statement about whether the program is stored on the hard disk or in the memory of the target device. However, there is an alternative 5) of deploying a program *without even transferring it to the target device* (lines 47-52). The board concludes that it can hardly be assumed that deploying necessarily implies storing the program *on the hard disk* if not even the transmission of the program is necessary for a deployment. Thus, the board considers a transferral (or sending) of a program to a target computer so that it can be executed on the latter to be a deployment within the meaning of the claim. This is the case in D1.

3.13 In its letter dated 24 July 2015 (page 4, first complete paragraph), the proprietor argues with respect to feature 1.5 that D1 does not disclose that the

second program is modified to be able to invoke the first program.

The board is not convinced by this argument. It is clear that the first program of D1 (e.g. the ExecProgram version of the DiskUtil program) *is* automatically executed (see column 8, lines 49-50: "Finally, ... the ExecPrograms are executed on the machines ..."). Since an execution can only happen by invocation by another program, there must be another program in D1 (e.g. the AppManager program) which invokes the first program automatically. Thus, this invoking program must be "automatically modified in order to be able to invoke the first program" and can be identified with the second program of the claim.

3.14 Thus it is confirmed that D1 discloses all the features of claim 1 of the main request, so that this claim is not new over D1 (Article 54 EPC).

First auxiliary request

4. *Original disclosure*

4.1 According to the opponent's letter of 18 November 2011 (pages 9-10, 2.), the feature of "drawing a link between two program icons ... to modify a second program ... to invoke a first program", introduced in claim 1 of the auxiliary request filed with the grounds, adds subject-matter, contrary to Article 123(2) EPC. The reason given is that the passage indicated by the proprietor as the basis for this amendment, i.e. description section [124] of the patent (corresponding to original description, page 25, last paragraph) discloses modifying the *first* program

to invoke the *second* program ([124], fourth sentence) whereas in the claim, the *second* program invokes the *first* one.

4.2 The board is not convinced by this reasoning. It is clear that the first and second program of the claim have to be matched to the first and second program of [124] in order to identify the programs of the claim with the programs in [124]. In this case, the calling program of the claim is the second one, whereas the calling program of [124] is the first one. Therefore, the second program of the claim has to be identified with the first program of [124], and vice versa. Furthermore, there is another passage ([26], seventh sentence) disclosing a "second program" invoking a "first program".

4.3 Therefore, the respondent's argument as to why the subject-matter of claim 1 of the auxiliary request is not originally disclosed fails.

5. *Clarity*

The board considers that it is unclear whether the expression added at the beginning of the characterising portion of claim 1 of the first auxiliary request, namely "drawing a link between two program icons (402, 404) to configure an invocation relationship ... to invoke a first program represented by the first program icon (404)", is meant as a mere *output* or as an *input* (with a corresponding output).

If it is meant as an output, the *first computer system* (82) merely *displays* a link (as in claim 2 of the

current main request and as in claim 1 of the second auxiliary request of the decision).

If it is meant as an input (with corresponding output), the user draws a link on the first computer which displays that link (as in [124], first two sentences).

6. *Inventiveness*

- 6.1 If the "drawing a link" is to be interpreted as a mere *output*, the board follows the opinion of the decision (42.-44. about the then second auxiliary request) that outputting relates to excluded subject-matter (Article 52(2)(d) EPC, presentation of information) and does not contribute to the technical character of the invention.

Therefore, claim 1 of the first auxiliary request is not inventive.

It should be noted that the formulation in the then second auxiliary request ("displaying a link icon") leaves no doubt about the fact that the feature merely relates to an output, in contrast to the present auxiliary request ("drawing a link").

- 6.2 If the "drawing a link" is to be interpreted as an *input* (with an associated output), the corresponding feature of inputting an invocation relationship constitutes a new feature not present in any of the claims of the patent.

However, the board considers inputting that one program calls another one to be a *mere automation* of the behaviour of an engineer who is going to build a

program sequence in a measurement and automation system (see also [3]-[6]). To input such an invocation relationship in a graphical way is furthermore well known from the field of visual programming.

Therefore, claim 1 of the first auxiliary request is not inventive with this interpretation either.

6.3 In its letter dated 24 July 2015, the proprietor argues in section (B), 2. (in particular on page 8, first paragraph) that the second program may be on another device than the deployed first program.

The board is not convinced. Since the claim leaves it open on which computer the second program is running, the implementation variant that they are running on the same computer is covered by the claim. Therefore, a simple definition of a (non-distributed) invocation relationship as well-known from visual programming is also covered by the claim, and the argument of the board still holds.

Furthermore, it should be noted that the AppManager program of D1 runs on a distributed system (column 9, first paragraph, lines 1-4) and consists of three components (lines 10-15): the AppManager GUI Console program running on the system administrator's console computer (lines 15-19; figure 8), the AppManager Server program running on the server computer (lines 19-22; the first computer 82 of the claim fulfills the functions of both, the console device and the server computer) and the AppManager Agent programs running on the target computers (of type server or workstation; lines 22-26; the programmable device of the claim corresponds to such a target computer). The target

computers are meant to receive the requests from the administrator's console computer (lines 24-25). Thus, the program structure of D1 also discloses a graphical input of a remote invocation relationship.

6.4 Thus it is confirmed that claim 1 of the first auxiliary request is not inventive over D1.

Second auxiliary request

7. Original disclosure

The features which were added to claim 1 of the first auxiliary request in order to come up with claim 1 of the second auxiliary request are originally disclosed in [124], first sentence (drawing a link *by a user*), in [175], second sentence (transferring), and in [176] (deploying/transferring first program and one or more other programs which are called by the first program).

8. Clarity

Since claim 1 of the second auxiliary request additionally contains the clarifying expression "by the user" in the drawing step, the objection raised above with respect to lack of clarity of the first auxiliary request does not apply to the second auxiliary request.

9. Inventiveness

9.1 In its letter dated 24 July 2015, the proprietor argues in section (C), 3. (in particular on page 11, paragraph 4, last three lines) that transferring and executing sub-programs of the deployed first program were not disclosed in D1 and that, together with the

- other differences (modified and compiled version of the first program; input of a remote invocation relationship in a distributed system), they establish an inventive step.
- 9.2 The board disagrees. Although the board deems it likely that the first program of D1 (i.e. the ExecProgram version of the DiskUtil program) is composed of a main program and of sub-programs called by the main program, it may not be the case. However, enabling the user to write the program in the well-known structured way (i.e. as a main program with sub-programs) is so commonplace that no inventive step can be recognised in it. This holds all the more, since the structured program may result in a single file, so that not even the transfer procedure would have to change.
- 9.3 As to the other alleged differences (modified and compiled version of the first program; input of a remote invocation relationship in a distributed system), the board has established above that it does not regard any of these as distinct from D1.
- 9.4 Therefore, claim 1 of the second auxiliary request is not inventive either.

Order

For these reasons it is decided that:

The appeal is dismissed.

The Registrar:

The Chairman:



B. Atienza Vivancos

W. Sekretaruk

Decision electronically authenticated