



- (51) **International Patent Classification:**  
*G05B 19/042* (2006.01) *G06F 9/44* (2006.01)
- (21) **International Application Number:**  
PCT/EP2010/006164
- (22) **International Filing Date:**  
8 October 2010 (08.10.2010)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (71) **Applicant (for all designated States except US):** **ABB TECHNOLOGY AG** [CH/CH]; Affolternstr. 44, CH-8050 Zürich (CH).
- (72) **Inventors; and**
- (75) **Inventors/Applicants (for US only):** **MESSINGER, Christian** [DE/DE]; Grosssachsener Str. 27a, 69493 Hirschberg (DE). **WILKE, Kay** [DE/DE]; Pfalzgrafenstr. 65, 67061 Ludwigshafen (DE). **GRAMBERG, Oliver** [DE/DE]; Heidelberger Str. 59, 69221 Dossenheim (DE). **JESKE, Ralf** [DE/DE]; Hoegenweg 10, 32469 Petershagen (DE).
- (74) **Agents:** **PARTNER, Lothar** et al.; ABB AG, GF-IP, Wallstadter Str. 59, 68526 Ladenburg (DE).
- (81) **Designated States (unless otherwise indicated, for every kind of national protection available):** AE, AG, AL, AM,

AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) **Designated States (unless otherwise indicated, for every kind of regional protection available):** ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Declarations under Rule 4.17:**

- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))
- of inventorship (Rule 4.17(iv))

**Published:**

- with international search report (Art. 21(3))

(54) **Title:** METHOD AND SYSTEM FOR PROPAGATION OF AMENDMENT MADE TO A MASTER TO COPIES

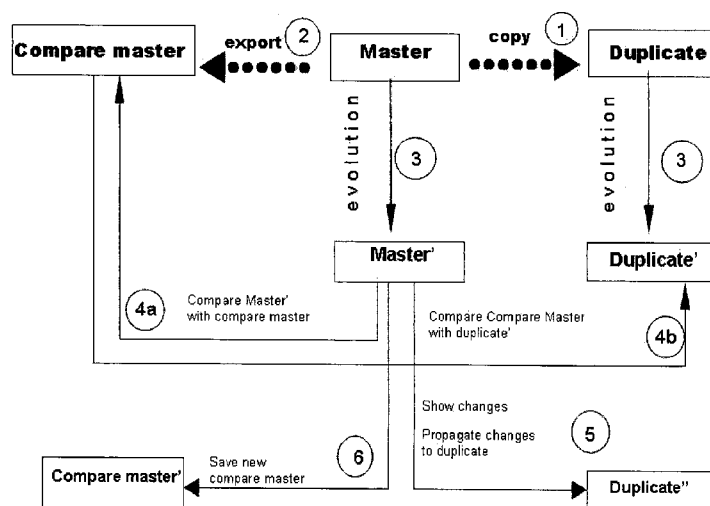


Fig. 3

(57) **Abstract:** The invention refers to a method and a system for propagation of amendments in the configuration of technical equipment, e.g., transformers, generators, mills, and other automated machines or devices, by means of transfer of an amended configuration of a master comprising a specific apparatus or device to a non-limited number of duplicates of said master, whereas any changes which have been done with the master are propagated from the master to the duplicate in a semi-automated or fully automated manner whereas possible conflicts are indicated automatically by means of graphic display.

## Method and System for Propagation of Amendments made to a Master to Copies

### Description

The invention refers to a method and a system for propagation of amendments in the configuration of technical equipment, e.g., transformers, generators, mills, and other automated machines or devices, by means of transfer of an amended configuration of a master comprising a specific apparatus or device to a non-limited number of duplicates of said master.

Many production plants have large subunits which are very similar but not identical to each other. As an example there may be several boilers in a chemical plant. Those boilers would be similar to a large extent but some aspects would be different. One aspect wherein subunits will be similar relating to their structure, i.e., the number, types, and arrangement of devices of which the subunit is made up. An aspect which will almost always differ is the naming of the tags and signals.

The optimal workflow would be to finish the configuration of the first subunit, to test and optimize it, and then to copy and adapt the configuration for use with further subunits of the same type. But this is a slow process, since the work on the second subunit can only be started after the first one has finished.

The process today is that the first subunit is configured, and then this configuration is copied to the other subunits, while the optimization and test of the configuration takes place after the copying. This leads to shorter overall project execution times but to a

higher engineering effort because all units have to be optimized and tested separately.

Accordingly, the challenge today is that the changes which are done to the first sub-unit during the test and optimization phase cannot be simply and automatically propagated to the other subunits for the following reasons:

- a) In most cases, there is no formal link between master and copy,
- b) there are differences between master and copy which should be kept,
- c) there may be conflicts between changes done in master and in copy, and finally,
- d) a completely automated process is unwanted by engineers who need to keep control of the engineered solution.

The result is that all changes which are done to the master after copying need to be done to the duplicate again.

Resulting from these findings it seems to be desirable to have a method and/or system at hand which avoid any excess of the efforts being necessary today and achieve systematically a standardization of the configuration of the respective sub-units of a plant.

Consequently, it is an objective of the present invention to provide a method and/or system which avoid the excess of efforts being necessary today and to achieve systematically a standardization of the configuration of the respective units.

According to the invention, the aforementioned problem is being solved by a method which is characterized in that any changes which have been done in the master are propagated from the master to the copies in a semi-automated or fully automated manner whereas possible conflicts are indicated automatically by means of graphic display.

Accordingly, the claimed method is characterized in that the propagation of changes done in the master comprises the following three steps:

- matching whereas the data source is analyzed to identify master and duplicate or duplicates, thus a logical link between the objects of the master and the duplicate is present;
- comparing whereas different types of changes, e.g., structural changes are considered and checked for possible conflicts and
- synchronizing whereas it is decided by the user which changes are approved and applied to the duplicate in order to synchronize the duplicate with the master.

Generally, the final decision needs to be taken by the engineer using the system but the decision is prepared by the system whereas the term “master” and “duplicate” refer to a usually large set of data objects typically being organized in one or more hierarchies whereas each object typically has a common and a specific set of information items.

In such case, the common information comprises, e.g., an identifier like the name, a designation of the type of the object, and the creation time of the object; while the specific information depends on the type of the object, e.g., for an object representing a device of a certain kind, the specific information comprises, among others, the configuration parameters applicable to this kind of device.

One preferred embodiment of the claimed method is characterized in that with the matching step the respective data source is analyzed in order to identify whether it is master or duplicate whereas a logical link between the objects of the master and the duplicate is present.

According to a more detailed embodiment of the method, preferably a three-way match is used wherein the master is saved in a compare master directly after copying the master data.

Changes done to the master result in a data set master' and changes done to any of the copies result in a data set copy' whereas both master' and copy' will be compared to the original version, the compare master, in order to identify if changes have been made to the master and/or the copies, whereas at any time when changes with the master are propagated to the duplicate, the current state of the master, master' is stored as the compare master as basis for later synchronization.

Preferably according to another embodiment of the invention the so-called compare master is stored as an action log whereas this action log is provided to detect and to determine the side on which a change has been made even if it has been deleted in the duplicate. This action log is as well an embodiment of the claimed method as of the claimed system according to the invention.

Another advantageous embodiment of the method provides for the matching to be based on a set of prioritized rules. The method will apply the rule with the highest priority first, and only objects not matched so far will be considered for matching with lower-priority rules. An example of a matching rule is equality of the name of the objects. Another example is that the names are not identical but similar to a certain degree. Furthermore, metrics are used like the type of the object, the number of matching children, the position in the tree, property values and more, and suitable combinations of these.

The user can choose which rules should be used for matching and in which order. He can also configure the individual rules when those offer parameters.

A typical example is the following rule: "match by similar name". One useful parameter may then be a "threshold" where objects are never matched when their names differ by e.g. more than 4 characters. If the engineer knows that in his project the naming is not a good way to match, he can deselect this option and choose other algorithms instead.

Hence, after the matching step, potential matches are presented to the engineer as a preferred user of the invention. Furthermore it is possible to attach a weight (like "92% match") to each match in order to point out the quality of the match to the user who may review these matches.

Either the user may manually match pairs of objects which have not been matched by the system or he may unmatch pairs of objects which have been falsely matched. The matching step is configurable by the using engineer.

A preferred embodiment of the method according to the claimed invention provides that conflicts are resolved by the user since the invention is also able to detect conflicts and shows these to the user. There may be simple conflicts where, e.g. both in master and in copy a single value item has been changed to two different values.

There may also be more complex conflicts where multiple changes that belong together must be propagated together. E.g., when a function block has been inserted in a control diagram and its inputs and outputs have been connected, none of these single changes makes sense to propagate on its own. The combination of these changes is called a change group. If change groups are overlapping in the master and in the copy, this presents a conflict.

A further part of the invention relates to a system where the method as illustrated before is applied to. Accordingly, this document discloses a system for propagation of changes in the configuration of technical equipment whereas two groups of objects are compared where the second group has been created by copying the first group but after copying one or both groups of objects have been changed.

Such system provides for the claimed method an appropriate means for propagation of changes in the configuration of technical equipment whereas in a first step a user or the system identifies the roots of master and copy or copies to the matching step; then in a second step the system identifies matching objects in the first group and in the second group for comparison and synchronization; after that in a third step the system compares objects matched in the second step and detects changes in the first group and the second group and conflicts in the changes in the first group and the second group; and finally in a fourth step the system presents the found changes and conflicts to the user for change propagation and conflict resolution.

According to a preferred embodiment of the claimed invention the identification of matching objects is not based only on the objects having the same ID or the same name but also on the use of other algorithms like having similar names and the same type or the same number and types of children. There may be algorithms that are specifically tailored to a standardized naming scheme.

Accordingly the user decides which algorithms are applied for the matching step and in which order they are applied whereas the user parameterizes the algorithms if possible to do so.

Preferably an embodiment of the system according to the invention is using a three-way comparison to determine if an identified change has been performed in the first group or in the second group.

According to another preferred embodiment of the claimed invention the system analyzes the two groups of objects being located in the same data set or in different data sets for changes and communicates or displays these changes to the user by means of any communication media whereas any changes being detected are done either on a structural level when for example when object-data level.

A further preferred embodiment of the system according to the invention is characterized in that multiple changes are grouped if these changes depend on each other.

According to a preferred version of the invention the system can propagate changes to links between objects in the master. The new target of the link in the duplicate is the object in the duplicate that corresponds to the new target of the link in the master as determined in the matching step. I.e., if in the master an object A references an object B and is changed to reference an object C, then in the copy object A' will be changed from referencing B' to referencing C' even though the names or identifiers of these objects are different from the objects they correspond to in the master.

Likewise it is advantageously provided by the system that the user may filter for relevant changes by choosing relevant change cases where each change case is a type of change which may occur for a certain type of object.

Generally the system offers proposals to the user which changes should be propagated and which should not.

These and further advantageous embodiments and improvements of the invention are subject matter of the dependent claims.

By means of examples of various preferred embodiments of the invention which are shown in the attached drawing the invention, advantageous embodiments and improvements of the invention as well as special advantages of the invention shall be illustrated and described in more detail.

In this regard it is indicated that the comprehensiveness of the invention is not limited to the embodiments and configurations shown and illustrated in the figures but extended to other embodiments and configurations within the scope of the claims.

It is shown in

- Fig. 1 an object structure with master and copy, before any changes, only the names of the copied objects are changed to make them unique,
- Fig. 2 the same two sets of objects as in Fig. 1 after structural changes have been made,
- Fig. 3 the evolution of object structures “master” and “duplicate” over time and a three way comparison with a compare master after changes have been made,
- Fig. 4 a possible screenshot of the system output showing matching of objects, differences detected, and change actions proposed to the user,
- Fig. 5 a screenshot of a list of changes cases from which the user can select the ones relevant to him, and
- Fig. 6 a configuration where overlapping change groups present a conflict.

Fig. 1 shows a very simple object structure, where the site “PresentationPlant” contains two boiler areas: “cfg\_HYD\_Boiler\_301” and “cfg\_HYD\_Boiler\_302”, the latter being a – suitably renamed – duplicate of the first. They each contain four function diagrams and each function diagram has several signal objects (DI/DO) as children. In the matching step, the data source is analyzed to identify master and duplicate or duplicates. After this step, a logical link between the objects of the master and the duplicate is present.

If objects are arranged in a hierarchical fashion, as shown in Fig. 1, an iterative matching needs to be done. First it needs to be determined that “cfg\_HYD\_Boiler\_301” and “cfg\_HYD\_Boiler\_302” are the roots of sub trees that have a master-duplicate relationship. Then this needs to be repeated for all function diagrams below this level and again for each signal below the function diagrams.

Fig. 2 shows a simple object structure with changes which may have been done to the master and the duplicate. The matching process will leave the objects “fd\_HYD\_Boiler\_302” and “fd\_HYD\_Boilerl\_302\_Rtx1”, marked with a surrounding



rectangle, and their respective children unmatched, because there are no corresponding objects in the other hierarchy. The same applies to the objects deleted in only the master or the duplicate, but not both.

There are many tools like directory comparing elements which compare two trees. In nearly all cases, the comparison is done based on name or ID. Also determining a master-duplicate relationship is not done.

The matching step is configurable by the user. The user can choose which algorithms should be used for matching and in which order. He can also configure the individual algorithms when those offer parameters. A typical example is the algorithm "match by similar name". One useful parameter could then be a "threshold" where objects are never matched when they differ by more than 4 characters. If the engineer knows that in his project the naming is not a good way to match, he can deselect this option and choose other algorithms instead.

In the comparison step, there are different types of changes to be considered. One type of changes is structural changes. As shown for example in Fig. 2 there might be an object in the master which has no corresponding object in the duplicate.

Since changes occur concurrently in the master and the duplicate and no action log is written, there is no way of determining whether this object has been created in the master or if it has been deleted in the duplicate. Hence an action log can be used as a compare master to detect the side on which a change has been made.

For the comparison step, this might be negligible information, but for the next step, the change propagation, this is crucial information: If the object has been added in the master, this change should be propagated to the duplicate. If it has been deleted in the duplicate, it is assumed that this has been done deliberately so and the object should not be added back to the copy.

To be able to differentiate the two situations, the invention makes use of the concept of a three-way match. In Fig. 3 a scheme is shown how this process works.

After copying the master data (step 1), the master itself is saved either in a file or to some other kind of storage medium (step 2). Then the master or the duplicate or both are changed (step 3), resulting in master' and duplicate'.

When the user now applies the system according to the invention, the master' will be compared to the original version of the master which is called "compare master" (step 4a). The compare master will also be compared to the duplicate' (step 4b).

With the information contained in the compare master, it is now possible to find out which changes have been done to the duplicate and which to the master. This method is far superior to common methods like comparing time stamps.

Every time changes are propagated to the duplicate' (step 5), a new export of the current master' to compare master' will also be done (step 6).

Fig. 4 shows by means of a screenshot how the matching is displayed. Matched nodes are displayed on the same level. If the system detected a change that can be propagated to the duplicate, then this is shown as a labeled arrow from left to right in the middle column. Changed, deleted or added items are marked as such which is done by a background color in this embodiment of the invention.

In Fig. 4, in the section below the tile "**Change Actions**" a change group is shown whereas it can be seen that one change action is "**On page 1 add component add (1)**" and this action has child actions. So, the change group consists of one added function block and three added connections which connect the new function block to the rest of the logic. Since the child actions depend on the parent action, a change group is used to build a meaningful set of items for a better overview, as well as to protect the system against inconsistencies which would result from partial execution of the change group's actions.

The comparison step is configurable by the user, in that the user can choose which types of changes he wants to be detected. In almost any case, he would not like to compare the name or the creation date.

To make this easily configurable, the concept of "change cases" has been introduced by the invention. For a function diagram, there are around 100 change cases. They range from "constant value has changed" and "diagram formatting has changed" over "execution order has changed" to "function block has been added".

Change cases are defined for every object type once. The user chooses during run-time which change cases he would like to see and which should be ignored. Fig. 5

shows some of the change cases defined for function diagrams. In this dialog, a user can choose which change cases are relevant for him.

Furthermore the invention is also able to detect conflicts and to show these to the user. There may be direct conflicts where, e.g. both in the master and in the duplicate, a constant value has been changed. The more complex cases are found by checking if change groups are overlapping.

Here Fig. 6 shows an example whereas in the master, the “output” of the diagram reference on the top left has been negated. This is a simple change, which can also be regarded to be a change group consisting of just that change.

In Fig. 6, this change group is indicated by the oval shape with the crossed pattern. In the duplicate, an “AND” block has been inserted between the two function blocks on the right side, and connected to these, indicated by the larger shape with the hatched pattern. This is a complex change. There is a conflict between these two changes because the two marked regions overlap.

### Claims

1. Method for propagation of changes in the configuration of technical equipment, e.g. transformers, generators, mills, and other automated machines or devices, by means of transfer of a changed configuration of a master comprising a specific apparatus or device to a non-limited number of duplicates of said master and for detection of conflicts with the propagation of changes whereas the propagation from the master to the at least one duplicate is executed by transferring a selected subset of the configuration of the master including the amendments,  
characterized in that all or some of the changes which have been done with the master are propagated from the master to the at least one duplicate in a semi-automated or fully automated manner whereas possible conflicts are indicated automatically by means of graphic or textual display.
2. Method according to claim 1 whereas the propagation of changes with the master comprises the following three steps:
  - a) matching whereas master and duplicate or duplicates are analyzed to identify which objects in the duplicate have been copied from which object from the master and thus correspond to each other ;
  - b) comparing whereas different types of changes, e.g. structural changes are considered and checked for possible conflicts and
  - c) synchronizing whereas it is decided by the user which changes are applied to the duplicate in order to synchronize the duplicate with the master.
3. Method according to claim 2 characterized in that with matching the respective data source is analyzed in order to identify master and duplicate automatically.
4. Method according to claim 2 whereas the concept of a three-way comparison is used wherein the master itself is saved in a compare master directly after copying the master, changes to master result in master' and changes to the dupli-

cate result in duplicate', whereas the master' as well as the duplicate' will be compared to the original version of the compare master.

5. Method according to claim 2 or 4 whereas at any time when changes with the master are propagated to the duplicate, the master is again saved to the compare master.
6. Method according to at least one of the preceding claims whereas for the matching definite identifiers of an object are used.
7. Method according to at least one of the preceding claims whereas conflicts are resolved by the user.
8. Method according to at least one of the preceding claims whereas an action log is provided in place of the compare master to detect and to determine the side on which a change has been made.
9. A system for propagation of changes in the configuration of technical equipment whereas two groups of objects are compared where the second group has been created by copying the first objects but after copying one or both groups of objects have been changed whereas
  - a) a user or the system identifies the roots of master and duplicate to the matching step;
  - b) then the system identifies matching objects in the first group and in the second group for comparison and synchronization and
  - c) detects possible conflicting changes in the first group and the second group if any.
10. System according to claim 9 whereas the identification of matching objects is not only based on the same ID or the same name but also on the use of other algorithms like having similar names and the same type or the same number and types of child objects.
11. System according to claim 9 or 10 whereas any of a plurality of matching algorithms is specifically tailored to a standardized naming scheme.

12. System according to at least one of the preceding claims 9 to 11 where the user decides which algorithms are applied for the matching step and in which order,
13. System according to claim 12 whereas the user parameterizes the algorithms if possible to do so.
14. System according to at least one of the preceding claims 9 to 13 using a three-way comparison to determine if an identified change has been performed in the first group or in the second group.
15. System according to at least one of the preceding claims 9 to 14 whereas the system analyzes the two groups of objects being located in the same data set or in different data sets for changes and displays these changes to the user.
16. System according to at least one of the preceding claims 9 to 15 whereas the system is able to detect changes done on a structural level and on an object-data level .
17. System according to at least one of the preceding claims where multiple changes are grouped if these changes depend on each other.
18. System according to at least one of the preceding claims where the user may filter for relevant changes by choosing relevant change cases where each change case is a type of change which may occur for certain types of objects.
19. System according to at least one of the preceding claims which is able to propagate changed links between objects.
20. System according to at least one of the preceding claims which offers proposals to the user which changes should be propagated and which should not.

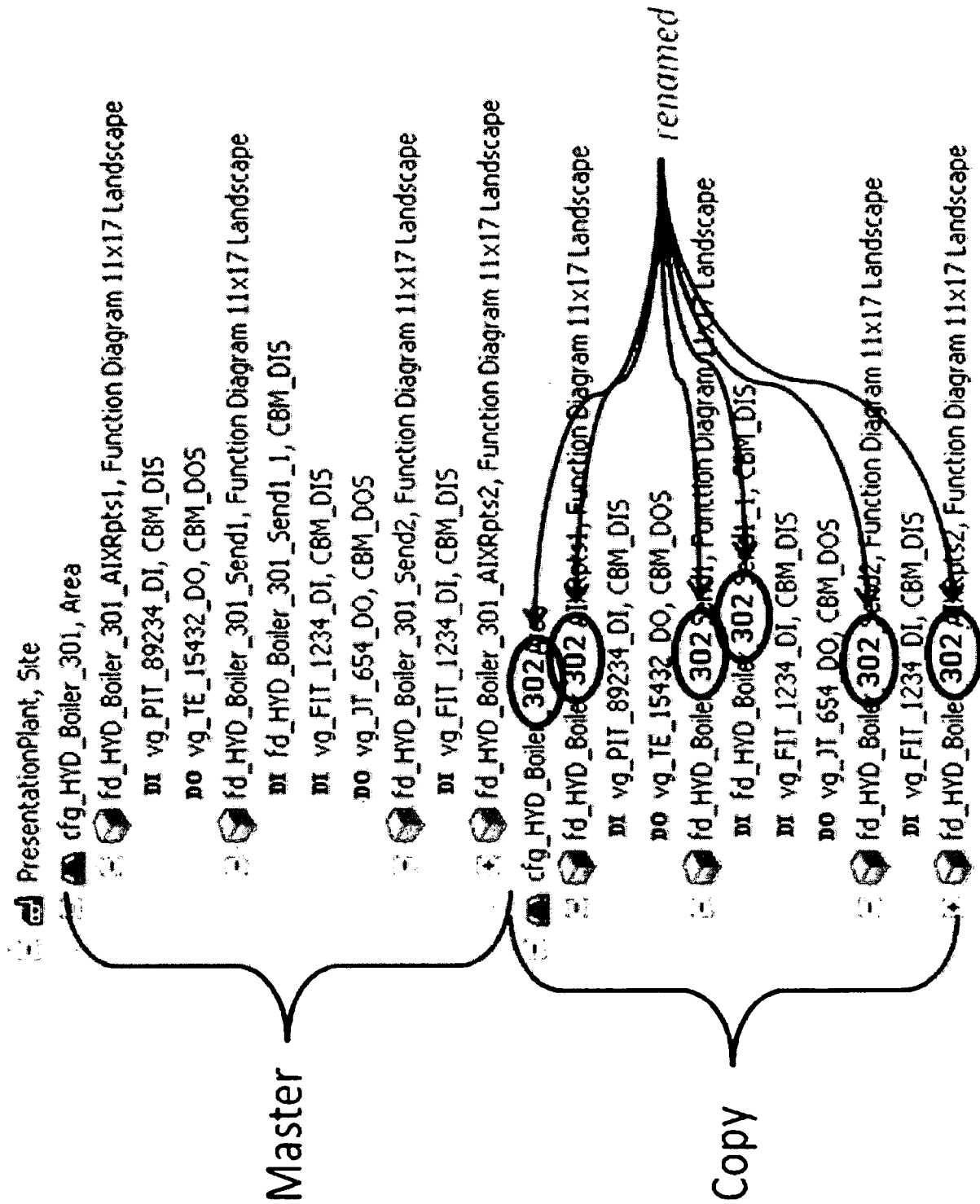


Fig. 1

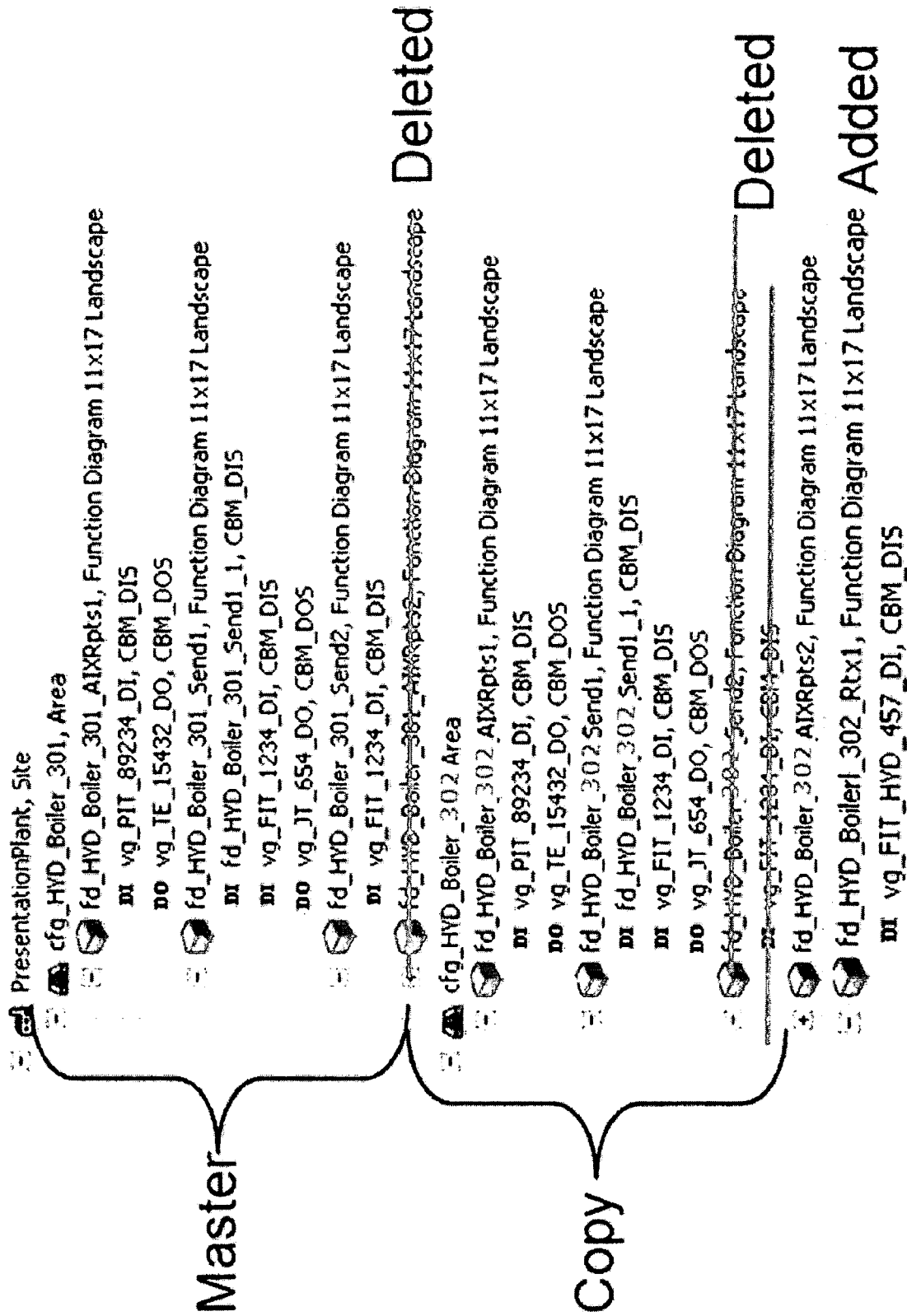


Fig. 2



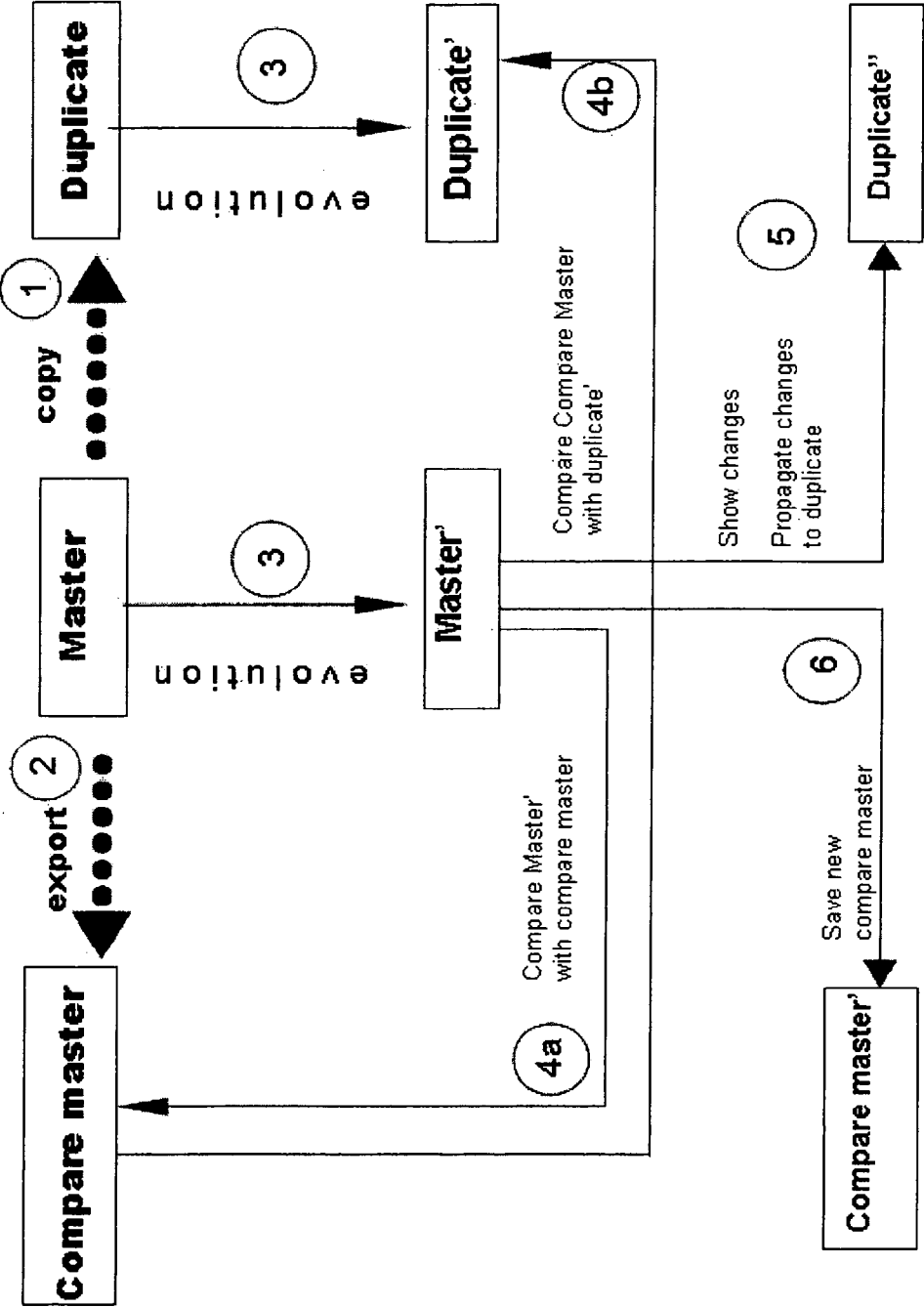
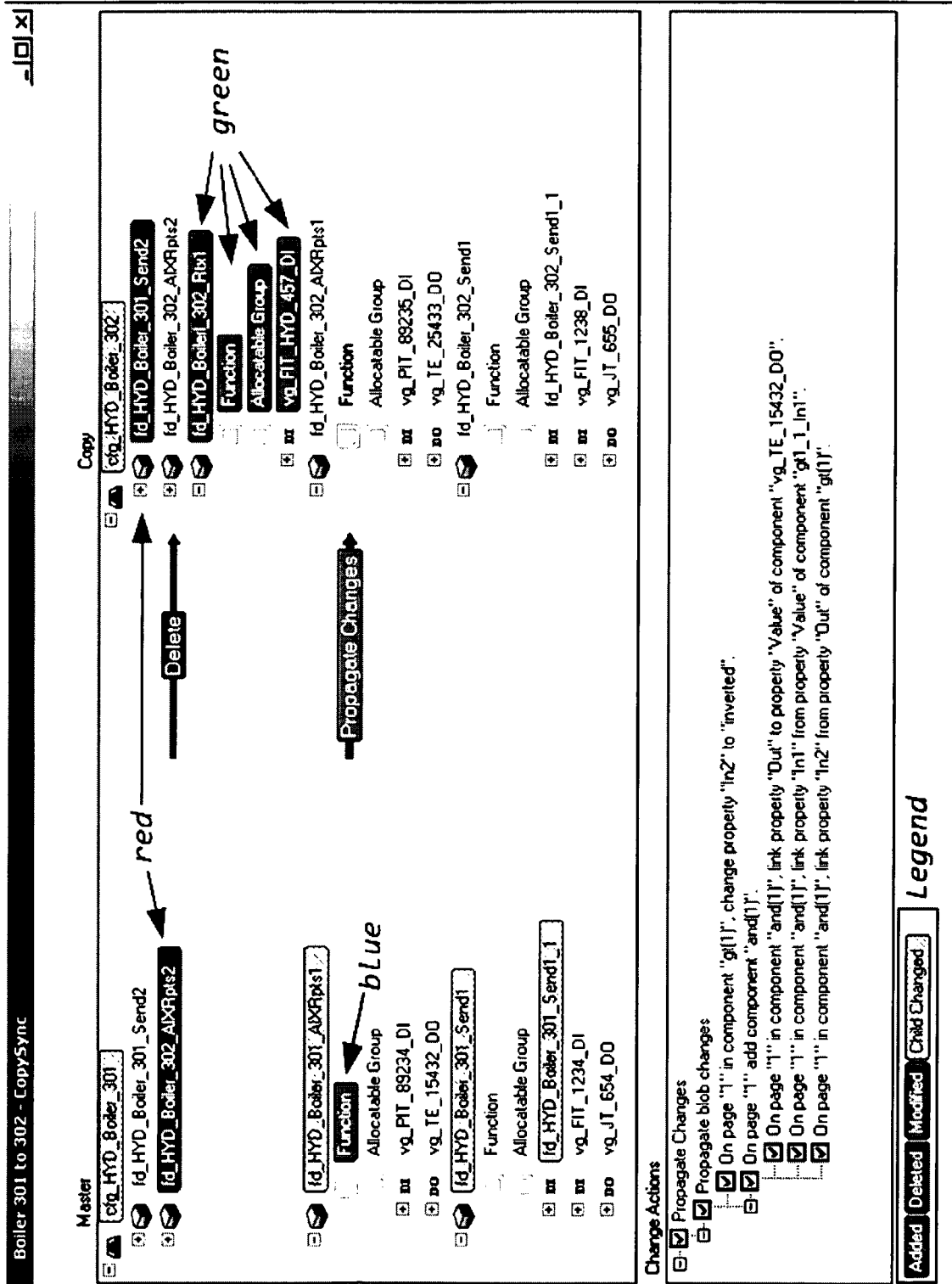


Fig. 3



**Fig. 4**

Change Cases

Symbols

☒ Add new Symbol

☐ Delete a Symbol

☐ Add a new Variable

☐ Delete Variable

☐ Change datatype of a variable

☐ New Diagram Input/Output Reference (DIOR)

☐ Delete DIOR

☐ Connect DIOR to Communication Variable


☐ Rename DIOR

☐ Change DIOR description

☐ Change "On Copy"-Rule

☐ Symbol Aspect Properties

☐ Changed Name

 Restore Defaults

This could be by a Function-Invocation (add(uint)) or a Function Block (RegisterBool1)

Fig. 5

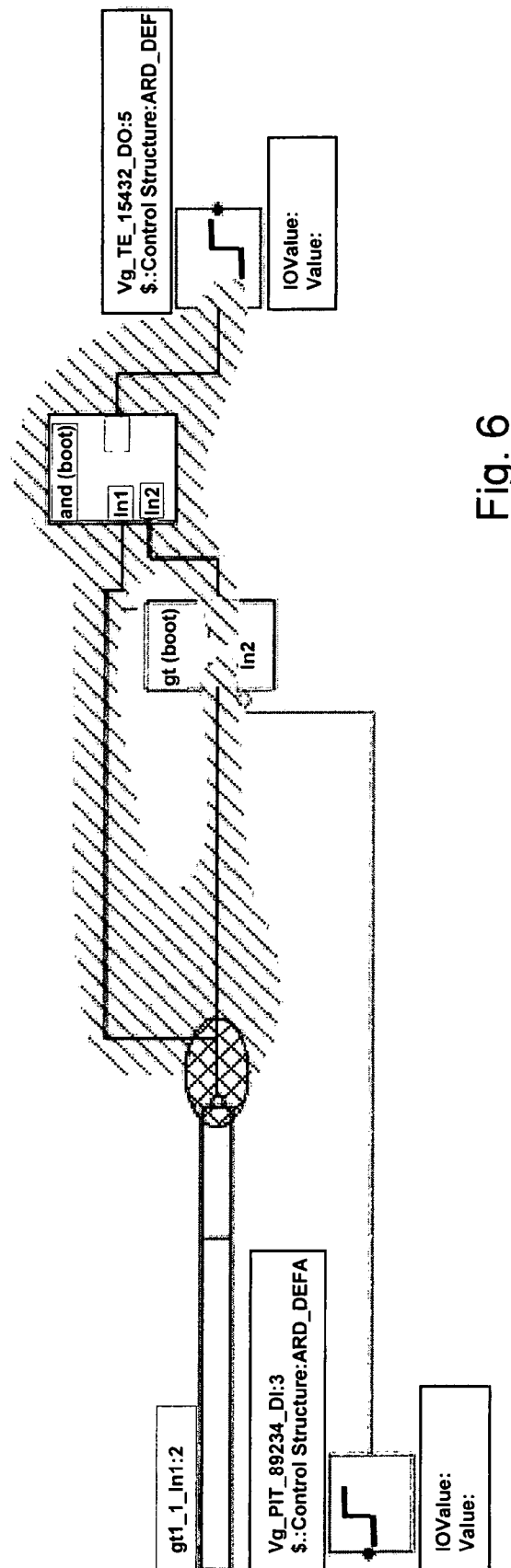


Fig. 6

## INTERNATIONAL SEARCH REPORT

International application No  
PCT/EP2010/006164

A. CLASSIFICATION OF SUBJECT MATTER  
INV. G05B19/042 G06F9/44  
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)  
G05B H04L G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2002/120723 A1 (FORTH J BRADFORD [CA] ET AL) 29 August 2002 (2002-08-29)	1
A	paragraphs [0001] - [0016], [0057], [0080] - [0089], [0100] - [0105] -----	2-20
X	US 2007/061033 A1 (LUCAS J M [GB] ET AL LUCAS J MICHAEL [GB] ET AL) 15 March 2007 (2007-03-15)	1
A	paragraphs [0001] - [0013], [0062], [0063], [0165] - [0184]; figures 24-27 -----	2-20
A	US 6 449 624 B1 (HAMMACK STEPHEN GERARD [US] ET AL) 10 September 2002 (2002-09-10) column 1 - column 3 column 24, line 5 - column 25, line 7 ----- -/--	1-20



Further documents are listed in the continuation of Box C.



See patent family annex.

\* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

25 August 2011

Date of mailing of the international search report

05/09/2011

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040,  
Fax: (+31-70) 340-3016

Authorized officer

José Luis Meseguer

## INTERNATIONAL SEARCH REPORT

International application No

PCT/EP2010/006164

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2008/228957 A1 (MEYER CHRISTIAN [DE] ET AL) 18 September 2008 (2008-09-18) the whole document	1-20
A	----- EP 1 770 459 A2 (ROCKWELL AUTOMATION TECH INC [US]) 4 April 2007 (2007-04-04) paragraph [0001] - paragraph [0040] -----	1-20

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/EP2010/006164

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2002120723 A1	29-08-2002	CA 2369965 A1	23-08-2002
		US 2006230394 A1	12-10-2006
		US 2004122833 A1	24-06-2004
US 2007061033 A1	15-03-2007	US 2010228373 A1	09-09-2010
		US 2005027377 A1	03-02-2005
US 6449624 B1	10-09-2002	DE 10051645 A1	09-08-2001
		GB 2363486 A	19-12-2001
		JP 2001242906 A	07-09-2001
US 2008228957 A1	18-09-2008	DE 112005002653 A5	15-11-2007
		WO 2006069762 A1	06-07-2006
EP 1770459 A2	04-04-2007	CN 1940785 A	04-04-2007
		JP 2007164764 A	28-06-2007
		KR 20070037426 A	04-04-2007
		US 2007078956 A1	05-04-2007