(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 3 372 473 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
12.09.2018 Bulletin 2018/37

(21) Application number: **17000388.3**

(22) Date of filing: **10.03.2017**

(51) Int Cl.:
B61L 15/00 $^{(2006.01)}$ *B61L 3/00* $^{(2006.01)}$
G07C 5/08 $^{(2006.01)}$

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA ME
Designated Validation States:
MA MD

(71) Applicant: **KNORR-BREMSE**
Systeme für Schienenfahrzeuge GmbH
80809 München (DE)

(72) Inventors:
- **Csutoras, Marton Ferenc**
8000 Székesfehérvár (HU)
- **Nagy, Gabor**
9024 Györ (HU)

(54) METHOD FOR LOGGING AND SYNCHRONIZING DIAGNOSTIC RELATED EVENTS

(57) A method for logging diagnostic related events in a system for railway application is provided. The method includes the steps: requesting of sending of an amount of stored events from a second system (LAD) to a first system (CID); sending of the amount of stored events from the second system to the first system; requesting of sending of stored events from the second system to the first system; sending of the stored events as sent events to the first system; in case of correctly receiving the sent events as received events, storing received events in the memory of the first system and acknowledging receipt of the received events to the second system; in case of not correctly receiving the sent events, multiply resending the same stored events to the first system by the second system until the sent events are correctly received and acknowledged to the second system.

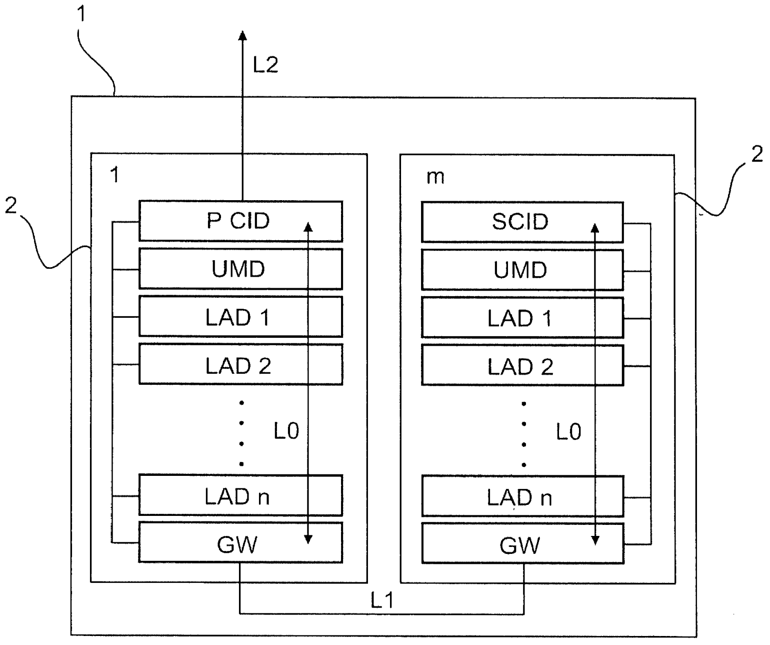

Fig. 1

Printed by Jouve, 75001 PARIS (FR)

Description

[0001] The invention relates to a method for logging and synchronizing diagnostic related events, in particular, to a method for logging and synchronizing diagnostic related events in a shared resource system for railway application.

[0002] In railway applications, shared resource systems are known. As shown in Fig. 1, a shared resource system is e.g. composed by segments 1. Each segment 1 contains one or more units 2. In one segment 1 there is at least one central intelligence device (CID), which is called the primary CID (P CID). In the case of redundant configuration, there is an additional CID in the same segment 1. The additional CID is called the secondary CID (S CID). CIDs are e.g. responsible for brake force distribution, i.e. for the so called blending. In one unit, there can be one or more local application devices (LAD) as stand-alone devices which are responsible for lower level tasks like measuring environmental information such as axle load or valve pressure and to carry out a wheel slide protection function. The minimal unit configuration contains one unit master device (UMD), one gateway (GW) and one CID and one or more LADs. The communication inside the unit is denominated LO (Level 0 communication), the communication between the units is denominated L1 (Level 1 communication), and the communication between the segments 1 is denominated L2 (level 2 communication). The UMD is responsible for managing the device addressing inside the unit 2. The GW is routing the messages between a LO bus and a L1 bus.

[0003] In the shared resource system both, the CID and the LAD, can generate diagnostics related events. These are stored in a non-volatile memory of these devices but the size of memory of these devices can be different. Usually, the size of the LAD's non-volatile memory is much less than that of the CID's memory. Therefore, the diagnostic related events are to be transferred or synchronized from the memory having the less size to the memory having the larger size of the shared resource system. However, a loss of the diagnostic related events, even not being safety critical, should be avoided. Therefore, the object underlying the invention is to provide a method for reliably transferring stored diagnostic related events from one memory to another memory.

[0004] The object is achieved by a method according to claim 1. Further developments of the invention are included in the dependent claims.

[0005] According to an aspect of the invention, a method for logging and synchronising diagnostic related events as events in a system for railway application is provided. The method includes the steps: step 1: requesting of sending of a number of not yet stored events from a second system to a first system by the first system; step 2: sending of the number of not yet acknowledged stored events from the second system to the first system by the second system; step 3: checking the number of the not yet acknowledged stored events by the first system, and proceeding to step 4 if the number of not yet acknowledged stored events is larger than zero, and proceeding to step 1 performed on a next second system if the number of not yet acknowledged stored events is equal to zero; step 4: requesting of sending a number of stored events from the second system to the first system by the first system; step 5: sending of the requested number of stored events as sent events to the first system by the second system; step 6: checking a number of correctly received events by the first system, and proceeding to step 7 if the number of correctly received events is equal to a number of requested events, proceeding to step 4 if the number of correctly received events is not equal to the number of requested events and a count of retries is smaller than a pre-defined parameter, and increasing the count of retries by one, and proceeding to step 1 performed on the next second system if the number of correctly received events is not equal to the number of requested events and the count of retries is greater than or equal to the pre-defined parameter; step 7: storing received events in the memory of the first system by the first system and acknowledging receipt of the received events to the second system by the first system; step 8: checking a number of stored events, sent in step 2, by the first system, and proceeding to step 1 performed on the next second system if the number of stored events sent in step 2 is equal to a number of successfully stored events in the first system, and proceeding to step 4 performed on the same second system if the number of stored events is larger than the number of successfully stored events in the first system.

[0006] By the provision of the method including these steps, the first systems knows the amount of the events to be transferred and, therefore, it can recognize whether received data concerning the amount of the events and received data concerning the events are consistent and whether the events have been transferred correctly. In particular, in case that the sent events have not been correctly transmitted, the first system can assure that it correctly receives the data by not acknowledging receipt of the data and, therefore, causing the second system to resend the data until the sent events are correctly received. This prevents a loss of the information since the transmission from the LAD side will always continue from where it has been interrupted until a correct acknowledge is received from the CID. By this way of data transfer, the loss of diagnostic information can be avoided in all scenarios.

[0007] In a first implementation of the method according to the aspect, the first system is a superior system and the second system is a subsystem.

[0008] By the provision of the first system being responsible for the data transmission or synchronization as a superior system, the second system as the subsystem must not have an undue performance and, therefore, it can be realized in a less expensive manner.

[0009] In a second implementation of the method according to the aspect or according to the first implementation, the stored events are sent as subsets of the events.

[0010] By sending the events as subsets of the events, data packets can be sufficient small for enabling a steady and quick data transmission by a bus for data transmission when multiple information are simultaneously to be sent by the bus.

[0011] In a third implementation of the method according to the aspect or according to anyone of the first and the second implementation, the second system sends an error code if a memory of the second system is corrupted.

[0012] Due to this feature, the shared resource system can recognize a fault in the second system and, therefore, it can execute countermeasures or signalize the fault so that the fault can be remedied as soon as possible.

[0013] In a fourth implementation of the method according to the aspect or according to anyone of the first to third implementation, the method further comprises steps of setting a state of a diagnostic state, generating one of the events in case of a state transition of the diagnostic state, and storing the event.

[0014] By these steps, a diagnostic related event can be provided when upon setting a state of a diagnostic state, the state transition appears and, therefore, a diagnostic related condition of a component has changed.

[0015] In a fifth implementation of the method according to the fourth implementation, the diagnostic state is one of a component diagnostic state, a functional diagnostic state, or a system diagnostic state.

[0016] Diagnostic states can be a component diagnostic state, a functional diagnostic state or a system diagnostic state. The component diagnostic state represent hardware related information, the functional diagnostic state and system diagnostic state represents a software component or a service.

[0017] In a sixth implementation of the method according to the fourth or fifth implementation, the setting of the state of the diagnostic state depends on a state of at least one fault linked to the diagnostic state.

[0018] One or more of the faults can be linked to a diagnostic state. Therefore, a state transition of at least one fault condition causes a change of the state of the diagnostic state.

[0019] In a seventh implementation of the method according to the sixth implementation, the method comprises the step of linking at least one of the faults to one or more related diagnostic states.

[0020] If appropriate, at least one of the faults can also be linked not only to one of the diagnostic states but also to multiple diagnostic states. Also, a set of faults can be linked to one or more diagnostic states.

[0021] In an eighth implementation of the method according to the sixth or seventh implementation, the method further comprises the step of prioritizing the faults depending on a degradation effect on a linked event.

[0022] By analysing the degradation effect on a linked event, the faults linked to the event can be prioritized and, therefore, appropriate counter measures against the faults can be adopted based on the severity of the fault.

[0023] In a ninth implementation of the method according to anyone of the sixth to eighth implementation, the method further comprises the step of defining a particular fault as a root cause if a specific diagnostic state is degraded by several faults and the particular fault linked to the specific diagnostic state has a highest degradation effect on the specific diagnostic state.

[0024] By the detection of the particular fault having the highest degradation effect on the linked specific diagnostic state, the root cause can be determined and related fault can be remedied.

[0025] In a tenth implementation of the method according to the aspect or according to anyone of the preceding implementations, the method comprises the step of storing the event including a source of the event and/or a time stamp and/or an environment information in an event log history.

[0026] Due to the storing of the event including additional information, a facilitated detection of the cause of the event is enabled. The environment information is defined by a designer of an application of the system since he is aware which environmental parameters are important to understand a possible root cause of the event.

[0027] In an eleventh implementation of the method according to the aspect or according to anyone of the preceding claims, the method further comprises the step of reading out diagnostic related events, and marking readout events as being readout.

[0028] By reading out the diagnostic related events, the events can be evaluated on external systems as e.g. a PC. In order to facilitate the detection of new events which had not been evaluated yet, the readout events are accordingly marked so that only newly raised events are considered upon the next reading out.

[0029] In a twelfth implementation of the method according to the eleventh implementation, a first user and a second user and a first level of information and a second level of information are defined. The first user is provided with the first level of information and second level of information and the second user is provided with the second level of information.

[0030] By defining at least two user having different rights to readout information, protected information of the system can be merely provided for e.g. an engineer of an owner of the system being allowed to handle e.g. intellectual property included in the system, whereas diagnostic states can be read by an operator for remedying faults.

[0031] The invention is now elucidated referring to the attached drawings by means of an embodiment.

[0032] In particular:

Fig. 1 shows a shared resource embedded system used by the method according to the invention; and

Fig. 2 shows possible linking solutions of a diagnostic concept according to the invention.

[0033] In the shared resource embedded system shown in Fig. 1 and described above, the diagnostic related information are to be transferred from the device having the memory having less size to the device having the memory having the larger size, i.e. from the LAD to the CID. The CID is responsible for collecting events from the LADs, therefore for a so-called mirroring. The mirroring also takes place in the case of an optional redundant CID configuration. In this case, the Primary CID is responsible for the mirroring. The mirroring can alternatively also be performed from a distant LAD which is not including in a P CID unit. In this case, the communication is performed via GW devices in the units 2.

[0034] In use, data are transferred from a smaller memory to a larger memory by a mirroring process. The CID is denominated as a first system and the LAD is denominated as a second system. Here, the first system is a superior system and the second system is a subsystem, however, the systems can alternatively also be equivalent systems or the first system can be the subsystem and the second system can be the superior system.

[0035] In a first step of the mirroring process, the first system requests sending of a number of not yet acknowledged stored events from the second system to the first system. If all the required local resources in the LAD are working correctly, i.e. the storage is not corrupted, in a second step, the second system sends the number of not yet acknowledged stored events from the second system to the first system. If the memory of the second system is corrupted, the second system sends an error code in this step of the method or, alternatively, at another appropriate moment. In a third step, the first system checks the number of the not yet acknowledged stored events, and it is proceeded to a fourth step if the number of not yet acknowledged stored events is larger than zero, and it is proceeded to step 1 performed on a next second system if the number of not yet acknowledged stored events is equal to zero. In the fourth step, the first system requests sending a number of stored events from the second system to the first system. The number of the requested events may be less than the total number of stored events. Subsequently, in a fifth step, the second system sends the requested number of stored events as sent events to the first system. In a sixth step, the first system checks a number of correctly received events and it is proceeded to a seventh step if the number of correctly received events is equal to a number of requested events, and it is proceeded to the fourth step if the number of correctly received events is not equal to the number of requested events and a count of retries is smaller than a pre-defined parameter (e.g. 3), and the count of retries is increased by one, and it is proceeded to the first step performed on the next second system if the number of correctly received events is not equal to the number of requested events and the count of retries is greater than or equal to the pre-defined parameter. The stored events are optionally sent as subsets of events, however, the data of the events can also be sent as entire package. In the seventh step, the first system stores received events in the memory of the first system and acknowledges receipt of the received events to the second system. In an eighth step, the first system checks a number of stored events sent in the second step and it is preceded to the first step performed on the next second system if the number of stored events sent in the second step is equal to a number of successfully stored events in the first system, and it is proceeded to the fourth step performed on the same second system if the number of stored events is larger than the number of successfully stored events in the first system. This can prevent the loss of the information since the transmission from the LAD side will always continue from where it has been interrupted until a correct acknowledge is received from the first system. By this way of data transfer, the loss of diagnostic information can be avoided in all scenarios.

[0036] Fig. 2 shows possible linking solutions of a diagnostic concept according to the invention. In the shared resource system, the diagnostic related information are presented at different levels with different meanings.

[0037] The lowest level of the diagnostic related information is a fault. In Fig. 2, faults are denoted with "F_1", "F_2", ..., "F_k". A fault can represent a hardware related error or a software related error. A possible state of a fault is either "healthy" (no error) or "sick" (error).

[0038] The highest level of the diagnostic related information is a diagnostic state. The diagnostic state represents a higher abstraction of hardware and software. For different purposes, there are different types of abstraction, e.g., a component diagnostic state, a functional diagnostic state, and a system diagnostic state. The component diagnostic state represents hardware related information as e.g. a status of the brakes. The functional diagnostic state and system diagnostic state represent the status of a software component or of a service, as e.g. a CAN communication. The diagnostic state is one of the component diagnostic state, the functional diagnostic state, or the system diagnostic state. In Fig. 2, the diagnostic states are denoted with "FDS_1", "FDS_2", ..., "FDS-m" (Functional Diagnostic State) and "CDS-1", "CDS_2", ... "CDS_n" (Component Diagnostic State).

[0039] As also shown in Fig. 2, the diagnostic state (FDS_2) is linked to only one fault (F_3) or the diagnostic state (FDS_1) is linked to several faults (F_1, F_2"). However, it is also possible that the diagnostic state is not linked to any fault (FDS_3). Alternatively, one fault is linked to several diagnostic states (not shown).

[0040] A setting of the diagnostic states can be done in two ways. In case that no fault is linked to the diagnostic state, it can be directly set. Otherwise, the diagnostic state can be set by the linked fault or by the linked faults. The setting of the status of the diagnostic state depends on the state of at least one fault linked to the diagnostic state. If at least one of the faults is set to sick, the linked diagnostic state is also set to sick. Therefore, the state of the diagnostic states are in relationship with the linked fault or faults.

[0041] Upon every state transition of one of the faults or of one of the diagnostic states, one of the events is generated. This event is stored as a diagnostic related information in an event log history in the LAD. Alternatively, the event log history can be stored in a memory of another component of the system. The event contains information about a source of the event with time stamp and detailed environment information. Alternatively, the event can contain merely a part of the information, additional or other information.

[0042] Since it is possible to link a set of faults to one or more diagnostic states, the fault causing the state transition of the diagnostic state is ambiguous. In order to clarify a root cause for a sick diagnostic state, the faults are prioritized depending on a degradation effect on the linked event. A particular fault is defined as a root cause if a specific diagnostic state is degraded by several faults and the particular fault linked to the specific diagnostic state has a highest degradation effect on the specific diagnostic state. By checking the root cause of the diagnostic state, the fault having the highest degradation effect is reported to a user.

[0043] The CID is connected to an Ethernet based maintenance port which is used for reading out diagnostic events. A maintenance software, e.g. web browser based, is used for a connection to one or more of the CIDs of the segment 1. The maintenance software is able to readout the CID's events and the events mirrored from the LADs. Readout events including the stored information like timestamp, event type and root cause are shown e.g. on a user's PC. Optionally, it can be confirmed that the events stored in the CID have already been readout. In other words, the diagnostic related information is readout and readout events are marked as being readout. Therefore, the user is enabled to merely readout newly raised events.

[0044] In a phase of evaluation of diagnostic information, a first user and a second user, e.g. an operator of the system and an engineer of an owner of the system, can be distinguished as different users. The two different users are provided with two different levels of information, which are defined as a first level of information and a second level of information, since a certain set of the diagnostic information is intellectual property (IP) of the owner of the system and, therefore, the IP relevant information is hidden for the operator. An access level is determined by an allocation of the diagnostic states, i.e., the faults are IP protected information, whereas, the diagnostic states are not IP protected information. Alternatively, also still multiple different users can be distinguished and still multiple levels of information can be defined in accordance with respective requirements.

REFERENCE SIGNS LIST

[0045]

1	segment
2	unit
CDS_	Component Diagnostic State
CID	Central Intelligence Device
F_	Fault
FDS_	Functional Diagnostic State
GW	GateWay
LAD	Local Application Device
L0	Level 0 communication
L1	Level 1 communication
P CID	Primary Central Intelligence Device
S CID	Secondary Central Intelligence Device
UMD	Unit Master Device

Claims

1. Method for logging and synchronizing diagnostic related events as events in a system for railway application, including the steps:

step 1: requesting of sending of a number of not yet acknowledged stored events from a second system (LAD) to a first system (CID) by the first system (CID);
step 2: sending of the number of not yet acknowledged stored events from the second system (LAD) to the first system (CID) by the second system (LAD);
step 3: checking the number of the not yet acknowledged stored events by the first system (CID), and
proceeding to step 4 if the number of not yet acknowledged stored events is larger than zero, and
proceeding to step 1 performed on a next second system (LAD) if the number of
not yet acknowledged stored events is equal to zero;
step 4: requesting of sending a number of stored events from the second system (LAD) to the first system (CID) by the first system (CID);
step 5: sending of the requested number of stored events as sent events to the first system (CID) by the second system (LAD);
step 6: checking a number of correctly received events by the first system (CID), and
proceeding to step 7 if the number of correctly received events is equal to a number of requested events,
proceeding to step 4 if the number of correctly received events is not equal to the number of requested events and a count of retries is smaller than a pre-defined parameter, and increasing

the count of retries by one,
proceeding to step 1 performed on the next second system (LAD) if the number of correctly received events is not equal to the number of requested events and the
count of retries is greater than or equal to the pre-defined parameter;
step 7: storing received events in the memory of the first system (CID) by the first system (CID), and
acknowledging receipt of the received events to the second system (LAD) by the first system (CID);
step 8: checking a number of stored events, sent in step 2, by the first system, and proceeding to step 1 performed on the next second system (LAD) if the number of stored events, sent in step 2, is equal to a number of successfully stored events in the first system (CID), and
proceeding to step 4 performed on the same second system (LAD) if the number of stored events is larger than the number of successfully stored events in the first system (CID).

2. Method according to claim 1, wherein the first system (CID) is a superior system and the second system (LAD) is a subsystem.

3. Method according to claim 1 or 2, wherein the stored events are sent as subsets of the events.

4. Method according to anyone of the preceding claims, wherein
the second system (LAD) sends an error code if a memory of the second system (LAD) is corrupted.

5. Method according to anyone of claims 1 to 4, further comprising the steps:

setting a state of a diagnostic state;
generating one of the events in case of a state transition of the diagnostic state; and
storing the event.

6. Method according to claim 5, wherein
the diagnostic state is one of a component diagnostic state, a functional diagnostic state, or a system diagnostic state.

7. Method according to claim 5 or 6, wherein the setting of the state of the diagnostic state depends on a state of at least one fault linked to the diagnostic state.

8. Method according to claim 7, further comprising the step of linking at least one of the faults to one or more related diagnostic states.

9. Method according to claim 7 or 8, further comprising the step of prioritizing the faults depending on a degradation effect on a linked event.

10. Method according to anyone of claims 7 to 9, further comprising the step of defining a particular fault as a root cause if a specific diagnostic state is degraded by several faults and the particular fault linked to the specific diagnostic state has a highest degradation effect on the specific diagnostic state.

11. Method according to anyone of the preceding claims, further comprising the steps:

storing the event including a source of the event and/or a time stamp and/or an environment information in an event log history.

12. Method according to anyone of the preceding claims, further comprising the steps of reading out diagnostic related information, and marking readout events as being readout.

13. Method according to claim 12, wherein a first user and a second user and a first level of information and a second level of information are defined, and wherein
the first user is provided with the first level of information and second level of information, and
the second user is provided with the second level of information.

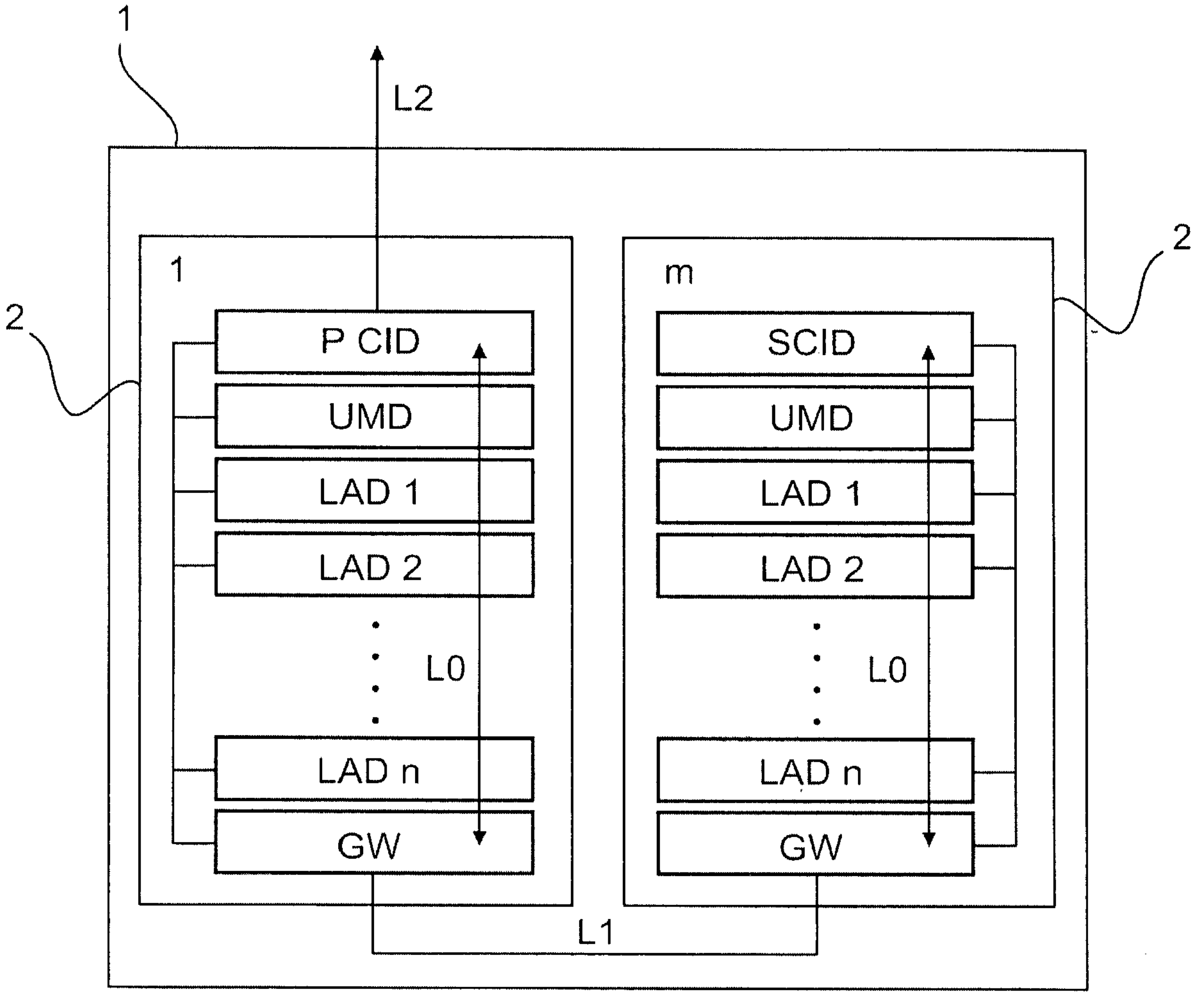
1
L2
1
m
2
2
P CID
UMD
LAD 1
LAD 2
L0
LAD n
GW
SCID
UMD
LAD 1
LAD 2
L0
LAD n
GW
L1

Fig. 1

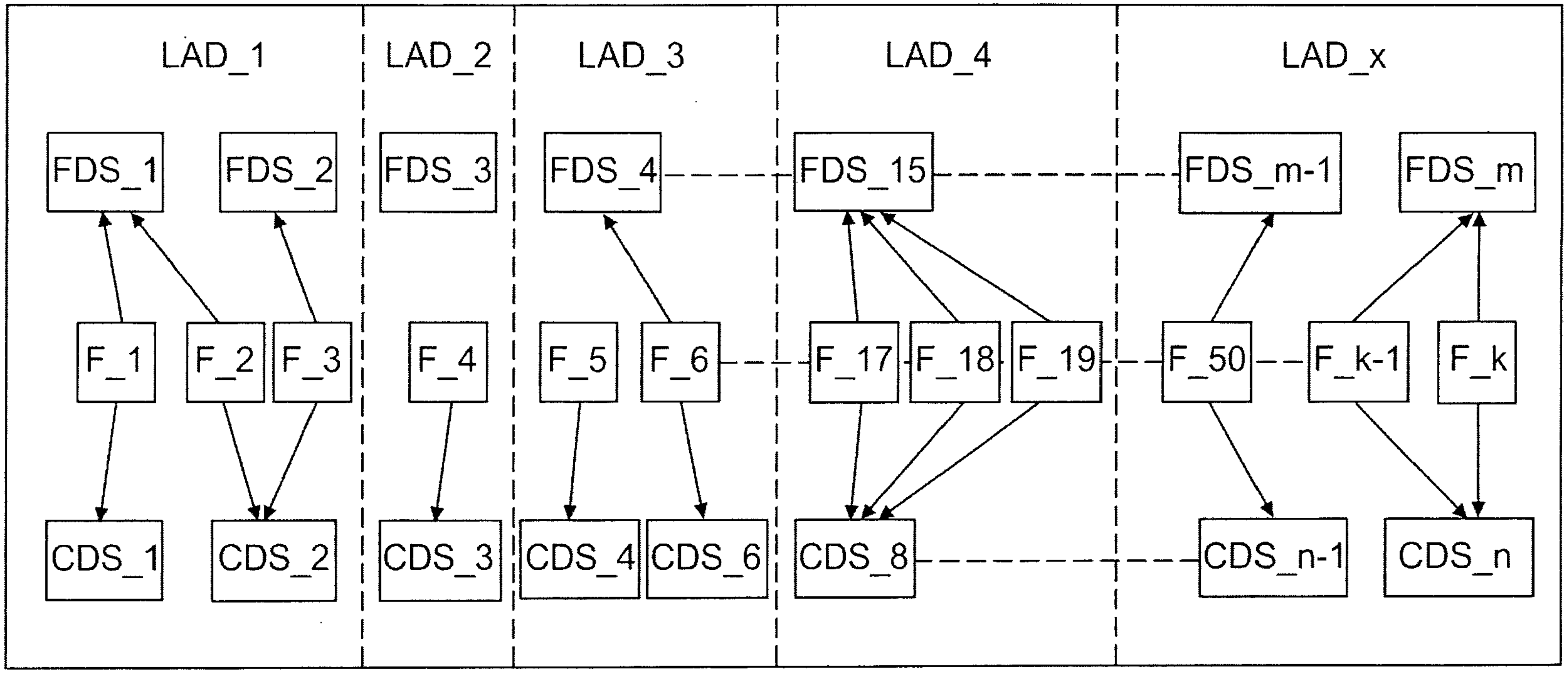
LAD_1
LAD_2
LAD_3
LAD_4
LAD_x
FDS_1
FDS_2
FDS_3
FDS_4
FDS_15
FDS_m-1
FDS_m
F_1
F_2
F_3
F_4
F_5
F_6
F_17
F_18
F_19
F_50
F_k-1
F_k
CDS_1
CDS_2
CDS_3
CDS_4
CDS_6
CDS_8
CDS_n-1
CDS_n

Fig. 2

Europäisches Patentamt
European Patent Office
Office européen des brevets

EUROPEAN SEARCH REPORT

Application Number
EP 17 00 0388

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
Y	EP 2 765 053 A2 (INSIGHT DESIGN SERVICES LTD [IE]) 13 August 2014 (2014-08-13) * paragraph [0010] - paragraph [0021] * * paragraph [0032] - paragraph [0035] * * paragraph [0051] - paragraph [0077] * * paragraph [0083] - paragraph [0088] * * paragraph [0098] - paragraph [0100] * * figures 1-7 * -----	1-13	INV. B61L15/00 ADD. B61L3/00 G07C5/08
Y	US 2013/304896 A1 (COLLINS PAUL FRANCIS [US] ET AL) 14 November 2013 (2013-11-14) * paragraph [0005] - paragraph [0008] * * paragraph [0014] - paragraph [0016] * * paragraph [0021] - paragraph [0031] * * figures 1-3 * -----	1-13	
Y A	US 2007/179691 A1 (GRENN DANIEL P [US] ET AL) 2 August 2007 (2007-08-02) * paragraph [0006] - paragraph [0012] * * paragraph [0046] - paragraph [0055] * * figures 1-2 * -----	4 1-3,5-13	
			TECHNICAL FIELDS SEARCHED (IPC)
Y A	US 2017/024943 A1 (WODECKI ARTHUR R [US] ET AL) 26 January 2017 (2017-01-26) * paragraph [0021] - paragraph [0040] * * paragraph [0063] - paragraph [0064] * * paragraph [0072] - paragraph [0073] * * figures 3 - 8 * -----	9,10 1-8, 11-13	B61L G07C
A	WO 2004/024531 A1 (BOMBARDIER TRANSP GMBH [DE]; SMEDLEY VINCENT ARTHUR [GB]; STEIJGER LAU) 25 March 2004 (2004-03-25) * page 2, line 9 - page 4, line 11 * * page 7, line 4 - line 19 * * page 17, line 5 - page 20, line 21 * * page 23, line 21 - page 24, line 12 * * page 28, line 11 - line 27 * * figures 1-2,6,20,21- 31 * -----	1-13	

The present search report has been drawn up for all claims

1

Place of search	Date of completion of the search	Examiner
The Hague	29 August 2017	Alvado Cárcel, Lucía

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons
....
& : member of the same patent family, corresponding document

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 17 00 0388

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

29-08-2017

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 2765053 A2	13-08-2014	EP 2765053 A2	13-08-2014
		IE S86224 B2	17-07-2013
US 2013304896 A1	14-11-2013	US 2013304896 A1	14-11-2013
		US 2014359068 A1	04-12-2014
US 2007179691 A1	02-08-2007	DE 102007004634 A1	27-09-2007
		US 2007179691 A1	02-08-2007
US 2017024943 A1	26-01-2017	NONE	
WO 2004024531 A1	25-03-2004	AU 2003269181 A1	30-04-2004
		GB 2392983 A	17-03-2004
		GB 2409904 A	13-07-2005
		WO 2004024531 A1	25-03-2004

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82