

High Availability Boot and Update for train embedded electronic devices

5 The present invention relates to an electronic device, such as a train device, comprising a processor and a memory unit associated with the processor, the memory unit including a boot program and at least one boot partition.

The present invention also relates to a rail vehicle comprising such an electronic train device.

10 The present invention also relates to a rail system comprising at least one element from among a maintenance server and an electronic maintenance apparatus, and at least one rail vehicle including such an electronic train device.

15 An electronic device of the aforementioned type is known from document CN 102945176 A. This electronic device comprises a data storage unit containing a boot program, as well as first and second boot partitions to improve the boot robustness of the electronic device. The first and second boot partitions have the same boot file system.

20 When a boot command is received, the boot program loads the first boot partition, verifies the integrity of said partition and/or performs a cyclic redundancy check. The boot program then boots up from this first partition if the performed check is correct. Otherwise, the boot program loads the second boot partition, and performs the boot process from that second partition. Having two redundant boot partitions then makes it possible to improve the likelihood of being able to boot the electronic device correctly.

25 However, the boot partitions generally take up a relatively large amount of space in the data storage unit, and rebooting such a device is sometimes delicate.

The aim of the invention is therefore to propose an electronic device with an improved boot, while having one or several boot partitions occupying a small amount of memory space.

30 To that end, the invention relates to an electronic device of the aforementioned type, wherein each boot partition includes a kernel, a network configuration manager and a query manager, the query manager being configured to manage at least one query received from an electronic apparatus outside the electronic device,

the boot program being configured to command, for the or one of the boot partitions, the activation of the kernel, the network configuration manager and the query manager, and

the query manager is configured so as to be able, after it is activated, to continuously receive at least one query from the outside apparatus.

According to other advantageous aspects of the invention, the electronic device comprises one or more of the following features, considered alone or according to all technically possible configurations:

- the memory unit includes a non-volatile memory and a volatile memory, the non-volatile memory being able to store the boot program and the boot partition(s), the volatile memory including an execution memory zone configured to execute the boot program, the kernel, the network configuration manager and the query manager of the corresponding boot partition, and

- the device further comprises an allocation module configured to allocate part of the execution memory zone to the query manager, then to reserve said part allocated to execution of the query manager and keep the query manager resident in said allocated part;

- the query manager is a computer process configured to be run in the background;

- the device further comprises a primary activation module configured to create, in the volatile memory, a memory storage zone allocated to the corresponding boot partition, then to load, in the memory storage zone, the kernel, the network configuration manager and the query manager of said boot partition;

- the primary activation module is further configured to command the execution of the kernel, the network configuration manager and the query manager of said boot partition;

- the primary activation module is further configured to keep the allocation of said memory storage zone in the volatile memory throughout all of the operation of the device;

- each boot partition is made up of the primary activation module, the kernel, the network configuration manager and the query manager;

- when the query manager depends on at least one function library, the query manager is statically linked to each library on which it depends;

- the query manager further includes a web server;

- the boot program is further configured to verify the integrity of the or at least one of the boot partitions;

- the boot program is configured to select the or one of the boot partitions, and to command, for the selected boot partition, the activation of the kernel, the network configuration manager and the query manager,

in particular via the execution of a primary activation module, the device comprising said primary activation module;

- the memory unit further includes an application partition, the application partition including at least one software application and one secondary activation module that is configured to run at least one corresponding software application; and

- the memory unit includes at least two redundant boot partitions, each boot partition including the kernel, the network configuration manager and the query manager.

The invention also relates to a rail vehicle comprising an electronic train device, wherein the electronic device is as defined above.

The invention also relates to a system comprising:

- at least one element from among a maintenance server and an electronic maintenance apparatus, and

- at least one rail vehicle including an electronic train device,
each element from among the maintenance server and the maintenance apparatus being connected to the electronic device and being configured to send one or several queries to the electronic device,

wherein the electronic device is as defined above.

The features and advantages of the invention will be better understood upon reading the following description, provided solely as a non-limiting example, and done in reference to the appended drawings, in which:

- figure 1 is a diagrammatic illustration of a train maintenance system according to the invention, comprising several electronic devices, a maintenance server and an electronic maintenance device, connected to one another via a network;

- figure 2 is a diagrammatic illustration of one of the electronic devices of figure 1;

- figure 3 is a flowchart of a boot sequence in the nominal mode of the electronic device of figure 2; and

- figure 4 is a view similar to that of figure 3 for a fail-soft boot.

In figure 1, a system 10 comprises several electronic devices 12, a maintenance server 14 and an electronic maintenance apparatus 16, the electronic devices 12, the maintenance server 14 and the electronic maintenance device 16 being connected via a network 18, such as a local network.

The system 10 is for example a rail system and further comprises at least one rail vehicle, not shown. At least one of the electronic device 12 is then placed onboard one of the rail vehicles and is then called electronic train device. The maintenance apparatus 16, connected to the train device 12, is then configured to send one or several queries to the train device 12, in order to maintain the train device inside which the train device 12 is located.

Each electronic device 12 comprises an information processing unit 20 for example formed by a processor 22 and a memory unit 24 associated with the processor 22, the memory unit 24 including a boot program 25, at least one boot partition 26 and an application partition 28.

The maintenance server 14 is configured to load, in each electronic device 12, the boot partition 26 and the application partition 28, each in their most recent version. The maintenance server 14 then comprises the boot partition 26 and the application partition 28, as well as a module 30 for loading the corresponding partition 26, 28 in the corresponding electronic device 12, as shown by arrow F1 in figure 1.

As an optional addition, the maintenance server 14 is also configured to send queries to an electronic device 12 that is in the maintenance phase. The maintenance server 14 is then configured to receive, from the corresponding electronic device 12, the data sent in response to each query sent.

The maintenance apparatus 16 comprises a maintenance module 32, such as a test module, configured to exchange data with each electronic device 12 via the network 18, as shown by arrow F2 in figure 1.

The maintenance apparatus 16 is in particular configured to send queries, via its maintenance module 32, to an electronic device 12 that is in the maintenance phase. The maintenance module 32 is then configured to receive, from the corresponding electronic device 12, the data sent in response to each query sent.

As an optional addition, the maintenance apparatus 16 is configured to load, in each electronic device 12, the boot partition 26 and the application partition 28, each in their most recent version. The maintenance apparatus 16 then comprises the boot

partition 26 and the application partition 28, as well as a module for loading the corresponding partition 26, 28 in the corresponding electronic device 12.

The maintenance apparatus 16 is an apparatus outside each electronic device 12 and known in itself. The maintenance apparatus 16 is for example a traditional computer, for example working with the Windows operating system, on which the maintenance module 32 is installed, the maintenance module 32 for example being a program executable by the processor of that computer. Alternatively, the outside apparatus 16 is an onboard electronic device including a dedicated software component.

The network 18 is known in itself, and is for example an Ethernet network. The network 18 is for example a rail network.

The processor 22 is known in itself, and is also called CPU (Central Processing Unit). The processor 22 is for example a microprocessor. Alternatively, the processor 22 includes one or several programmable logic components.

The memory unit 24 includes a non-volatile memory 34 and a volatile memory 36, the non-volatile memory 34 being able to store the boot program 25 and the boot partition(s) 26.

In the example of figure 2, the memory unit 24 includes two redundant boot partitions 26. Alternatively, the memory unit 24 includes a single boot partition 26. Also alternatively, the memory unit 24 includes at least three redundant boot partitions 26.

Each boot partition 26 includes a kernel 40, a network configuration manager 42 and a query manager 44. As an optional addition, each boot partition 26 includes a primary activation module 46, able to be executed by the boot program 25.

The boot program 25 is then configured to command, for the corresponding boot partition 26, the activation of the kernel 40, the network configuration manager 42 and the query manager 46, for example via the execution of the primary activation module 46.

The boot partitions 26 are said to be redundant when they each include the kernel 40, the network configuration manager 42 and the query manager 44. The redundant boot partitions 26 are preferably identical.

In the example of figure 2, each boot partition 26 is made up of the kernel 40, the network configuration manager 42, the query manager 44 and the primary activation module 46.

The boot program 25 is further configured to verify the integrity of the boot partition 26 or at least one of the boot partitions 26. The integrity check is for example done with a

checksum, such as a cyclical redundancy check calculation, also called CRC, or with a hash function.

The boot program 25 is also configured to select the or one of the boot partitions 26 and to command, for the selected boot partition 26, the activation of the kernel 40, the network configuration manager 42 and the query manager 44, for example via the execution of the primary activation module 46.

The application partition 28 includes at least one software application 50 and one secondary activation module 52 that is configured to run each software application 50 using the processor 22. The primary activation module 46 is then further configured to command the execution of the secondary activation module 52, for example after executing the kernel 40, the network configuration manager 42 and the query manager 44 of the corresponding boot partition 26. In other words, the software application(s) 50 are executed via the processor 22, for example after executing the various elements of the selected boot partition 26, i.e., after the execution, via the processor 22, of the kernel 40, the network configuration manager 42 and the query manager 44.

Alternatively or as an optional addition, one or several software applications 50 are executed via the processor 22 in parallel with the execution of the different elements of the selected boot partition 26, as shown in figures 3 and 4.

The volatile memory 36 includes a memory execution zone 60 and a memory storage zone 62, the memory execution zone 60 being configured to execute the boot program 25, the kernel 40, the network configuration manager 42 and the query manager 44 of the corresponding boot partition.

Each device 12 further comprises an allocation module 70 configured to allocate part 72 of the execution memory zone 60 to the query manager 44, then to reserve said part 72 allocated to execution of the query manager 44 and keep the query manager 44 resident in said allocated part 72. The allocation module 70 is preferably directly included in the kernel 40, and is then for example implemented by a management service of the volatile memory 36.

The kernel 40 is for example a kernel of the Linux type. In the example of figure 2, the kernel 40 is a kernel including a minimum of services, in order to occupy minimal memory space in the non-volatile memory 34.

The kernel 40 for example includes the following services: volatile memory management, input/output management, file system management, network management, command prompt.

The network configuration manager 42 is suitable for configuring one or several network layers, not shown, in order to allow the electronic device 12 to communicate with the network 18.

As an optional addition, each boot partition 26 comprises a proxy server, also called proxy, not shown, suitable for intercepting load queries by the maintenance module 32. In other words, the proxy is configured to redirect queries coming from the maintenance module 32.

Alternatively, the redirection of the queries is done by the query manager 44. Also alternatively, the queries from the maintenance module 32 are sent directly to the correct recipient, and do not need to be reoriented.

The query manager 44 is, according to the invention, configured so as to be able, after it is activated, to continuously receive at least one query from the outside apparatus 16.

The query manager 44 is for example a computer process configured to run in the background, like a daemon, when the kernel 40 is a Linux-type kernel.

The query manager 44 further depends on at least one function library 74 able to be stored in the non-volatile memory 34, the query manager 44 is then statically linked to each library 74 on which it depends. This static link of the query manager 44 to each library 74 on which it depends is for example done during the compilation of the different elements of the boot partition 26, in particular during the compilation of the query manager 44.

As an optional addition, the query manager 44 further includes a web server, not shown, suitable for sending and receiving web queries.

The primary activation module 46 is configured to create, preferably, in the volatile memory 36, the memory storage zone 62, the latter being allocated to the corresponding boot partition 26. The primary boot module 46 is further configured to next load, in the memory storage zone 62, the kernel 40, the network configuration manager 42, the query manager 44, and if applicable the proxy, of said boot partition 26.

As an optional addition, the primary activation module 46 is further configured to keep the allocation of said memory storage zone 62 in the volatile memory 36 throughout the entire operation on the electronic device 12.

The primary activation module 46 is further configured to command the execution of each of the elements of the corresponding boot partition 26 other than itself, i.e., to

command the execution by the processor 22 of the kernel 40, the network configuration manager 42, the query manager 44, and if applicable the proxy.

The boot sequence of the electronic device 12 according to the invention will now be explained using figures 3 and 4, showing a flowchart of the boot sequence in normal mode, and in fail-soft mode, respectively.

According to figure 3, the boot program 25 begins by verifying, during step 100, the integrity of the or one of the boot partitions 26, at least of the selected boot partition 26, for example with a checksum calculation. This makes it possible to ensure that the boot partition 26 that will be used to boot up the electronic device 12 is not corrupted.

As previously indicated, the integrity verification of the or one of the corresponding boot partitions 26 is done via a CRC calculation or via the application of a hash function on the content of said boot partition 26, next with a comparison of the value obtained by this hash function with a predetermined value.

During the integrity verification step 100, if the boot program 25 detects that a first boot partition 26 does not check out, i.e., is corrupted, then the boot program 25 boots up from the following boot partition 26.

When the memory unit 24 includes a single boot partition 26 and the boot program 25 detects corruption of this single boot partition 26, then the boot program 25 stops booting up the corresponding electronic device 12 and reports that corruption of the boot partition 26.

Having several redundant boot partitions 26 then makes it possible to reduce the risk of failure of the electronic device 12. This in particular makes it possible to boot up the electronic device 12 despite the corruption of one or several boot partitions 26 once there is still at least one intact boot partition 26 that makes it possible to boot up the electronic device 12.

One skilled in the art will more generally observe that increasing the number of redundant boot partitions 26 makes it possible to increase the uptime of the electronic device 12.

After checking the integrity of the boot partition(s) 26, the boot program 25 commands the activation of the kernel 40 by activating the execution, via the processor 22, of the primary activation module 46.

During the following step 110, the primary activation module 46 then begins by starting the kernel 40 and its associated utilities. The primary activation module 46 next

commands the activation of at least one software application 50, via the activation of the execution, via the processor 22, of the secondary activation module 52.

During step 120, the secondary activation module 52 commands the execution of the corresponding software application(s) 50, in order to perform certain tasks, in particular the following tasks: additional network management, specific hardware management (specialized coprocessor, sensor, hardware configuration key, serial connection, display, etc.), train supervision (signaling, traction, passenger information, etc.).

The secondary activation module 52 also commands the update of the network 18 if needed. When the different tasks at the application partition 28 level have been carried out, the secondary activation module 52 tells the primary activation module 46 that the application of the network confirmation has been done successfully. The primary activation module 46 then continues its boot sequence in parallel with the secondary activation module 52.

The primary activation module 46 next activates, during step 130, the configuration of the network 18 using the network configuration manager 42, in particular to configure the corresponding network layer(s).

When the network configuration is done, the primary activation module 46 commands, during the following step 140, the booting up of the query manager 44, which will then be able to continuously receive one or several queries from the outside apparatus 16 or the maintenance server 14, and to place itself in standby to wait for one or several requests (step 150).

During the following step 160, in particular following the reception of a query, the software application(s) 50 perform different tasks at the application partition 28 level.

The electronic device 12 is then booted up, and the latter is able to continuously receive at least one query from the outside electronic apparatus 16 or the maintenance server 14, in particular a query related to the implementation of a maintenance test, an identification request, or a save recovery request, from the outside apparatus 16 or the maintenance server 14. After receiving this query from the outside apparatus 16, the electronic device 12 is of course configured to process that query, prepare a response and next send the response corresponding to the query, to the outside device 16.

The flow chart of figure 4 illustrates the case of booting up of the electronic device 12 in fail-soft mode, when the application partition 28 does not react following the

activation of the secondary activation module 52, by the primary activation module 46, as shown by the failure step 220 following the boot step 110 of the kernel.

In the example of figure 4, steps 100 and 110 are identical to those previously described in the example of figure 3, and are not described again.

When, after the boot step of the kernel 110, the primary activation module 46 commands the activation of one or several software applications 50 of the application partition 28 via the secondary activation module 52, it also triggers a maximum time delay $T_{\max}$ in order to perform the network configuration step 130 no later than at the end of that maximum time delay $T_{\max}$, in case of absence of response from the application partition 28.

In the example of figure 4, when the activation of the application partition 28 leads to a failure (step 220), then the primary activation module 46 activates, at the end of the maximum time delay $T_{\max}$, the configuration of the network 18 during step 130, step 130 being identical to that previously described in light of figure 3.

At the end of the network configuration step 130, the primary activation module 46 also commands, similarly to what was described for figure 3, during step 140, the booting up of the query manager 44, this boot step 140 of the query manager being identical to that previously described. The electronic device 12 is then on standby while awaiting one or more queries (step 150), similarly to what was previously described for figure 3.

Thus, the electronic device 12 is, after it is booted up, continuously available to receive a query from the outside apparatus 16 or the maintenance server 14, which improves its maintenance, and in particular facilitates updating thereof. This ability of the electronic device 12 to continuously receive queries from the outside apparatus 16 or the maintenance server 14, after it is booted up, is obtained via the implementation of the query manager 44 configured to be able to continuously receive, after its activation, at least one query from the outside apparatus 16 or the maintenance server 14.

This ability of the query manager 44 to continuously receive a query from the outside apparatus 16 or the maintenance server 14 is for example obtained by allocating the allocated part 72 of the memory execution zone 60 to the query manager 44, then reserving this allocated part 72 to execute the manager 44, the query manager 44 then being kept resident in the allocated part 72 throughout all of the operation of the electronic device 12. In other words, memory locking primitives are used to lock part of the volatile memory 36, in order to ensure that the query manager 44 will not lack memory space to run.

The fact that the query manager 44 is a computer process configured to be run in the background further makes it possible to guarantee the persistence of the query manager. In particular, the fact that the query manager 44 is in the form of a daemon when the kernel 40 is of the Linux type makes it possible to further improve the uptime of the query manager 44.

The loading of the set of minimum elements of the boot partition 26, i.e., the kernel 40, the network configuration manager 42 and the query manager 44 in the memory storage zone 62, this memory storage zone 62 being allocated to the corresponding boot partition 26, makes it possible to improve the accessibility of these elements. The memory storage zone 62 is also called RAM disk.

The fact that the kernel 40, the network configuration manager 42 and the query manager 44 are loaded in the memory storage zone 62 makes it possible to update the entire boot partition 26, since no file needs to be locked.

As an optional addition, this memory storage zone 62 is further not released during the operation of the electronic device 12 in order to guarantee more accessibility of the kernel 40, the network configuration manager 42 and the query manager 44.

The static connection of the query manager 44 with each function library 74 on which it depends makes it possible not to have a dynamic dependence with outside libraries, minimize the memory space occupied by the query manager 44 and further reduce the boot time of the electronic device 12.

Furthermore, the fact that the query manager 44 includes, as an optional addition, its own web server, makes it possible not to depend on a third-party web server and to still further reduce the boot time of the electronic device 12.

One can thus see that the invention has a minimal impact on the boot time of the electronic device 12, and despite the presence of one or several boot partitions 26, the invention also has a minimal impact on the occupancy of the memory space.

Additionally, when the electronic device 12 includes several redundant boot partitions 26, the electronic device 12 further offers better uptime and greater reliability, since it remains possible to boot or reboot the electronic device 12, even in case of failure of one of the boot partitions 26.

CLAIMS

1. An electronic device (12), such as a train device, comprising:
 - a processor (22),
 - an allocation module (70),
 - a memory unit (24), associated with the processor (22) and including:
 - + a boot program (25),
 - + at least one boot partition (26),
 - + a non-volatile memory (34), and
 - + a volatile memory (36),

characterized in that each boot partition (26) includes a kernel (40), a network configuration manager (42) and a query manager (44), the query manager (44) being configured to manage at least one query received from an electronic apparatus (16) outside the electronic device (12),

the boot program (25) being configured to command, for the or one of the boot partitions (26), the activation of the kernel (40), the network configuration manager (42) and the query manager (44),

the query manager (44) being configured so as to be able, after it is activated, to continuously receive at least one query from the outside apparatus (16),

the non-volatile memory (34) being able to store the boot program (25) and the boot partition(s) (26), and

the volatile memory (36) including an execution memory zone (60) configured to execute the boot program (25), the kernel (40), the network configuration manager (42) and the query manager (44) of the corresponding boot partition,

wherein the allocation module (70) is configured to allocate part (72) of the execution memory zone (60) to the query manager (44), then to reserve said part (72) allocated to execution of the query manager (44) and keep the query manager (74) resident in said allocated part (72).

2. The device (12) according to claim 1, wherein the query manager (44) is a computer process configured to be run in the background.

3. The device (12) according to any one of the preceding claims, wherein the device (12) further comprises a primary activation module (46) configured to create, in a volatile memory (36), a memory storage zone (62) allocated to the corresponding boot partition (26), then to load, in the memory storage zone (62), the kernel (40), the network configuration manager (42) and the query manager (44) of said boot partition (26).

4. The device (12) according to claim 3, wherein the primary activation module (46) is further configured to command the execution of the kernel (40), the network configuration manager (42) and the query manager (44) of said boot partition (26).

5. The device (12) according to claim 3 or 4, wherein the primary activation module (46) is further configured to keep the allocation of said memory storage zone (62) in the volatile memory (36) throughout all of the operation of the device.

6. The device (12) according to any one of claims 3 to 5, wherein each boot partition (26) is made up of the primary activation module (46), the kernel (40), the network configuration manager (42) and the query manager (44).

7. The device (12) according to any one of the preceding claims, wherein, when the query manager (44) depends on at least one function library (74), the query manager (44) is statically linked to each library (74) on which it depends.

8. The device (12) according to any one of the preceding claims, wherein the query manager (44) further includes a web server.

9. The device (12) according to any one of the preceding claims, wherein the boot program (25) is further configured to verify the integrity of the or at least one of the boot partitions (26).

10. The device (12) according to any one of the preceding claims, wherein the boot program (25) is configured to select the or one of the boot partitions (26), and to command, for the selected boot partition (26), the activation of the kernel (40), the network configuration manager (42) and the query manager (44),

in particular via the execution of a primary activation module (46), the device (12) comprising said primary activation module (46).

11. The device (12) according to any one of the preceding claims, wherein the memory unit (24) further includes an application partition (28), the application partition (28) including at least one software application (50) and one secondary activation module (52) that is configured to run at least one corresponding software application (50).

12. The device (12) according to any one of the preceding claims, wherein the memory unit (24) includes at least two redundant boot partitions (26), each boot partition (26) including the kernel (40), the network configuration manager (42) and the query manager (44).

13. A rail vehicle comprising an electronic train device (12), characterized in that the electronic device (12) is according to any one of the preceding claims.

14. A rail system (10), comprising:

- at least one element from among a maintenance server (14) and an electronic maintenance apparatus (16), and

- at least one rail vehicle including an electronic train device (12),

each element from among the maintenance server (16) and the maintenance apparatus (14) being connected to the electronic device (12) and being configured to send one or several queries to the electronic device (12),

characterized in that the electronic device (12) is according to any one of the preceding claims.