(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 795 015**

(51) Int. CI.:

B61L 19/06 (2006.01)
B61L 27/00 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **24.04.2017 PCT/EP2017/059631**

(87) Fecha y número de publicación internacional: **02.11.2017 WO17186629**

(96) Fecha de presentación y número de la solicitud europea: **24.04.2017 E 17720733 (9)**

(97) Fecha y número de publicación de la concesión europea: **29.04.2020 EP 3448735**

(54) Título: **Dispositivo de servidor que opera un software para el control de una función de un sistema de protección de transporte sobre carriles**

(30) Prioridad:

25.04.2016 DE 102016206988

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
20.11.2020

(73) Titular/es:

THALES MANAGEMENT & SERVICES DEUTSCHLAND GMBH (100.0%)
Thalesplatz 1
71254 Ditzingen, DE

(72) Inventor/es:

ERDMANN, CHRISTOPH

(74) Agente/Representante:

ISERN JARA, Nuria

DESCRIPCIÓN

Dispositivo de servidor que opera un software para el control de una función de un sistema de protección de transporte sobre carriles

La invención se refiere a un dispositivo de servidor que opera un software para el control de una función de un sistema de protección de transporte sobre carriles,
operando el software al menos dos procesos de forma separada físicamente entre sí cuyos resultados se comparan entre sí para realizar el control de la función.

Sistemas de protección de transporte sobre carriles, en particular puestos de enclavamiento y sistemas de protección de trenes, se automatizan cada vez más mediante ordenadores. A este respecto se debe garantizar un alto nivel de fiabilidad, disponibilidad, facilidad de mantenimiento y seguridad de personas (los denominados requisitos RAMS; Reliability Availability Maintainability Safety). Mientras que, por regla general, errores de software (errores de programación) se pueden descubrir y eliminar mediante una planificación y revisión adecuadas de escenarios de prueba hasta la puesta en funcionamiento, errores de hardware (en particular el fallo de componentes constructivos individuales, por ejemplo, transistores) se pueden producir en principio en cualquier momento durante el funcionamiento. Errores de hardware de este tipo se deben descubrir a tiempo de forma que no se produce un riesgo para personas (maquinistas de locomotora, pasajeros) en la protección de transporte sobre carriles y, preferiblemente, tampoco para equipos (locomotoras, vagones) valiosos o carga.

Por tanto, en aplicaciones relevantes para la seguridad en sistemas de protección de transporte se realiza habitualmente un procesamiento de múltiples canales y una revisión de componentes relevantes para la seguridad, véase, por ejemplo, M. Schäfer, F. Schneider, "Standardisierte Bedienoberflächen für Bahnsteuerungssysteme", Signal + Draht (98), 9-2006, páginas 50-52.

En el procesamiento de múltiples canales, varios procesos del mismo tipo se operan en paralelo separados físicamente entre sí, esto es, con hardware diferente, y los resultados se comparan entre sí. En caso de coincidir los resultados, se puede partir de que el hardware implicado funciona correctamente. En el caso de un error en un hardware implicado, se produce una divergencia de los resultados, lo que se puede detectar mediante una comparación de los mismos. La aplicación puede tomar entonces medidas de seguridad adecuadas, por ejemplo, poner de manera preventiva señales en "parada".

Software del ámbito de los sistemas de protección de transporte sobre carriles está instalado habitualmente en aparatos individuales en los que se puede asegurar bien la separación física de procesos. Para ello, el software y la arquitectura de aparato se ajustan de manera adecuada entre sí.

En el caso de ordenadores con denominados procesadores de múltiples núcleos es posible provocar una asignación fija de procesadores individuales a recursos de ordenador mediante una programación adecuada. Con respecto a la programación en Linux ha demostrado ser eficiente para ello el uso de denominados "cgroups", véase la entrada en inglés de Wikipedia "cgroups" del 31/3/2016. De manera correspondiente, para procesos cuyos resultados se deben comparar se pueden asignar diferentes núcleos de procesador (el denominado "core binding"), por lo que se puede asegurar la separación física de los procesos.

Mediante la virtualización de aplicaciones se puede renunciar a proporcionar aparatos individuales en muchos casos. Asimismo, el desarrollo de software y la integración están simplificados. Por ejemplo, la virtualización de un sistema de control de trenes se ha propuesto en el documento WO 2015/126529 A1.

Además, mediante la virtualización en un clúster de servidores a partir de varios servidores individuales es posible realizar una migración de procesos a otro servidor individual en el caso de un fallo de un servidor individual y, así, mejorar la disponibilidad de una aplicación.

Sin embargo, en el caso de una operación de un software en un nivel operativo virtualizado de un clúster de servidores, procesos individuales operados por el software ya no se pueden asignar a determinados recursos de ordenador; en particular, se asignan los procesos individuales de forma fundamentalmente aleatoria a uno de los ordenadores individuales. Entonces existe el riesgo (estadísticamente relevante) de que varios procesos, cuyos resultados se deben comparar entre sí, se ejecuten en el mismo hardware de forma que un error de hardware de este hardware genera los mismos cálculos erróneos en estos varios procesos y, de manera correspondiente, el error de hardware ya no se puede encontrar mediante una comparación de los resultados de los procesos. En este caso ya no está garantizada la seguridad operacional en el sistema de protección de transporte sobre carriles.

El documento EP 1 085 415 A2, que se considera el estado de la técnica más próximo, da a conocer un módulo de programa y un procedimiento para aumentar la seguridad de un sistema controlado por software, en particular de un puesto de enclavamiento electrónico para la técnica de señalización ferroviaria. En el ejemplo de realización de la figura 4 del mismo se emplea un conjunto de ordenadores compuesto por los ordenadores R8, R9, R10, R11 y un comparador V3. Los ordenadores R8 y R9 están conectados en serie y los ordenadores R10 y R11 están

conectados en serie. A este respecto, los ordenadores R10 y R11 están conectados en paralelo a los ordenadores R8 y R9. En los ordenadores R8 y R10 está instalada en cada caso la primera parte de programa de un módulo de programa, y en los ordenadores R9 y R11 está instalada en cada caso la segunda parte de programa del módulo de programa. Ambos ordenadores R8 y R10 reciben los mismos datos de entrada. Los datos de salida de los ordenadores R9 y R11 se comprueban por el comparador V3; solo si coinciden los datos de salida de los ordenadores R9 y R11 se libera un trayecto de desplazamiento.

El documento US 2003/0018927 A1 describe un sistema de servidor de clúster de alta disponibilidad. El clúster comprende varios servidores físicos/servidores individuales a los que se hace referencia como "nodes". En cada node se ejecutan uno o varios programas de software a los que se hace referencia como "servidor virtual". Si falla un node, un servidor virtual afectado se transfiere a otro node.

Objetivo de la invención

La invención se basa en el objetivo de proporcionar un dispositivo de servidor en el que se pueda garantizar una disponibilidad mejorada de una aplicación de software con una alta seguridad operacional del tráfico ferroviario al mismo tiempo.

Breve descripción de la invención

Este objetivo se consigue mediante un dispositivo de servidor del tipo mencionado al inicio con las características de la reivindicación 1 que está caracterizado por que el software se opera en un nivel operativo virtual del dispositivo de servidor,
por que el dispositivo de servidor comprende al menos dos clústeres de servidores separados físicamente entre sí,
comprendiendo cada uno de los clústeres de servidores del dispositivo de servidor al menos dos servidores individuales que permiten una migración de procesos entre sí en el caso de un fallo de un servidor individual,
ejecutándose los al menos dos procesos en máquinas virtuales,
y por que el software comprende al menos dos partes que están instaladas en clústeres de servidores diferentes de los al menos dos clústeres de servidores de forma que los al menos dos procesos se operan en clústeres de servidores diferentes de los al menos dos clústeres de servidores.

Por un lado, la invención permite que una aplicación de software acceda a la disponibilidad aumentada en clústeres de servidores. Sin embargo, por otro lado, asegura que procesos, cuyos resultados se deben comparar entre sí para conservar la seguridad operacional, se ejecutan separados físicamente entre sí. Para ello, el dispositivo de servidor, que se utiliza para la operación del software, se configura con al menos dos clústeres de servidores. Cada uno de los clústeres de servidores del dispositivo de servidor comprende al menos dos servidores individuales que permiten una migración de procesos entre sí en el caso de un fallo de un servidor individual (High Availability Cluster). De esta manera se asegura una alta disponibilidad (disponibilidad operativa). Por otro lado, el software se divide en al menos dos partes que se reparten por los al menos dos clústeres de servidores. En cada caso una parte del software, y, con ello, uno de los procesos, está asignada fijamente a uno de los clústeres de servidores. De esta manera se asegura que los procesos, cuyos resultados se deben comparar entre sí, se ejecutan en diferentes clústeres de servidores y, con ello, en hardware diferente. Esta separación física de los procesos asegura que un error de hardware individual, que provoca un resultado erróneo de un proceso, se puede descubrir mediante una comparación con el resultado de un proceso del mismo tipo calculado con otro hardware (impecable).

Los procesos, cuyos resultados se comparan entre sí, pueden ser procesos de verificación especiales que se ejecutan adicionalmente a la función de control de la aplicación de software (por ejemplo, el cálculo de dígitos de verificación/sumas de verificación), o también procesos principales que se usan por sí mismos para la función de control (por ejemplo, el cálculo de un esquema de las vías). Los procesos a comparar entre sí llevan a cabo las mismas operaciones de cálculo en el mismo orden (procesos del mismo tipo) para obtener el respectivo resultado de proceso. En general, los mismos resultados de proceso indican un funcionamiento correcto del dispositivo de servidor; en general, resultados de proceso diferentes indican un fallo.

Por ejemplo, uno de los procesos, cuyos resultados se deben comparar entre sí, es un proceso maestro, y un segundo proceso es un proceso esclavo. Si el resultado del proceso esclavo se diferencia del resultado anteriormente determinado del proceso maestro, el estatus de la aplicación de software se pone en "no seguro" (unsafe) (por ejemplo, mediante la parte de software del proceso maestro y/o la parte de software del proceso esclavo y/o una parte de software adicional para el proceso de comparación) y ya no se fía de ninguno de los resultados de los procesos. Por ejemplo, en una aplicación de puesto de enclavamiento, todas las señales afectadas se pueden poner entonces de manera preventiva en "parada".

Mediante la comparación de los resultados de los procesos se puede garantizar de manera fiable una operación segura del dispositivo de servidor o de la aplicación de software y, con ello, también de la función controlada del sistema de protección de transporte sobre carriles, por ejemplo, en un puesto de enclavamiento electrónico. Dado que los procesos están asignados en cada caso estrictamente a los clústeres de servidores individuales, la separación física de los mecanismos de verificación está asegurada en cualquier momento. A este respecto, la

expresión "separados físicamente entre sí" hace referencia a una separación de los procesos de cálculo con respecto al hardware utilizado.

Mediante la virtualización es posible operar el software en gran parte de manera independiente de un hardware local disponible. En particular se pueden remplazar fácilmente componentes individuales (tales como servidores individuales dentro de uno de los clústeres de servidores).

Formas de realización preferidas de la invención

En una forma de realización preferida del dispositivo de servidor de acuerdo con la invención, el software es una aplicación de puesto de enclavamiento. Debido a la arquitectura de acuerdo con la invención del dispositivo de servidor se puede garantizar un alto nivel de seguridad, tal como se exige habitualmente para aplicaciones de puesto de enclavamiento. Asimismo, es ventajosa la alta disponibilidad para evitar o minimizar retrasos en el funcionamiento del tráfico ferroviario.

Es especialmente preferible un perfeccionamiento en el que el software es una aplicación para operar la interfaz de usuario de un puesto de enclavamiento controlado por ordenador, en particular con una funcionalidad para conectar terminales de operación móviles. Por ejemplo, el software puede ser una aplicación de servidor HIS (HIS = human machine interface for interlocking systems), en particular con función MPT y/o HHT (MPT = mobile possession terminal; HHT = hand held terminal). En esta aplicación ha demostrado ser especialmente eficaz la arquitectura de servidor de acuerdo con la invención. Como procesos a comparar o sus resultados se pueden usar en este caso esquemas calculados de las vías que se indican en terminales de operación, en particular terminales de operación móviles (tal como ordenadores tablet). Dado que el usuario puede encargarse temporalmente de la liberación de tramos de vía, en este caso debería estar disponible un estándar de seguridad elevado que puede proporcionar la invención.

También es preferible una forma de realización en la que el software es una aplicación de protección de trenes. Debido a la arquitectura de acuerdo con la invención del dispositivo de servidor se puede garantizar un alto nivel de seguridad, tal como se exige habitualmente para aplicaciones de puesto de enclavamiento. Aplicaciones de protección de trenes pueden incluir, por ejemplo, sistemas de frenado de emergencia al sobrepasar señales de "parada".

Además, es ventajosa una forma de realización en la que el software está configurado de acuerdo con el nivel de integridad de seguridad 2 (SIL2) o superior. Este nivel de seguridad SIL2 es suficiente para muchas aplicaciones de sistemas de protección de transporte sobre carriles y se puede alcanzar bien con la arquitectura de servidor de acuerdo con la invención, pudiendo posibilitarse al mismo tiempo una disponibilidad aumentada. El nivel de integridad de seguridad (SIL) viene determinado según la norma EN 61508 (en particular las normas EN 50128 y EN 50129) en la versión vigente del 4/4/2016. El software puede ser, por ejemplo, una aplicación de servidor HIS.

Es especialmente ventajosa una forma de realización en la que el software está configurado de acuerdo con el nivel de integridad de seguridad 4 (SIL4). Con ello, el software cumple con los requisitos de seguridad más elevados. El nivel de seguridad SIL4 también se puede alcanzar bien con la arquitectura de servidor de acuerdo con la invención, pudiendo posibilitarse al mismo tiempo una disponibilidad aumentada. El nivel de integridad de seguridad (SIL) viene determinado según la norma EN 61508 (en particular las normas EN 50128 y EN 50129) en la versión vigente del 4/4/2016. Por ejemplo, el software puede ser una aplicación de un bloque central de radio (RBC = Radio Block Centre) o de un puesto de enclavamiento electrónico (interlocking module), además también una aplicación SCM (SCM = safe communication module) o una aplicación FEC (FEC = field element controller).

Además, es ventajosa una forma de realización en la que el software opera exactamente dos procesos que están separados físicamente entre sí en exactamente dos clústeres de servidores diferentes. La configuración de dos clústeres de servidores para únicamente dos procesos a comparar entre sí (respectivamente en el caso de una respectiva operación de verificación) es relativamente fácil, aunque aumenta considerablemente la seguridad con una alta disponibilidad al mismo tiempo.

Una forma de realización ventajosa alternativa prevé que el dispositivo de servidor comprenda tres clústeres de servidores separados físicamente entre sí, que el software comprenda al menos tres partes que están instaladas en clústeres de servidores diferentes de los clústeres de servidores de forma que el software opera tres procesos en clústeres de servidores diferentes de los tres clústeres de servidores y que los resultados de los procesos se evalúen en el marco de una decisión de 2 de 3 para el control de la función del sistema de protección de transporte sobre carriles. Con la decisión de 2 de 3 es posible identificar aún resultados de proceso correctos también en el caso de un fallo de un hardware (en este caso de un error en uno de los clústeres de servidores), lo que aumenta adicionalmente la disponibilidad.

También es preferible una forma de realización en la que el dispositivo de servidor opera al menos un software adicional para el control de una función adicional de un sistema de protección de transporte sobre carriles y que el al menos un software adicional esté instalado y se opere únicamente en uno de los clústeres de servidores. El

respectivo software adicional no se divide en partes diferentes que se deben instalar en clústeres de servidores diferentes; de este modo se facilita considerablemente la operación del software adicional. Típicamente, el software adicional está configurado según SILO. Típicamente, en esta forma de realización están instaladas y se operan en cada caso una o varias aplicaciones de software adicionales en cada uno de los clústeres de servidores.

En un perfeccionamiento preferido de esta forma de realización, el al menos un software adicional comprende una o varias de las siguientes aplicaciones de software:

- un sistema de planificación de horario, en particular ARAMIS-D;
- un sistema de gestión de números de tren y dirección de tren, en particular ARAMIS-C;
- un sistema de análisis de datos y métrica (business intelligence);
- un sistema de mantenimiento de trenes (maintenance centre);
- un sistema de registro y control de datos de tren (checkpoint master node);
- un sistema de registro y evaluación de componentes operativos (service management tool). En la práctica, estas aplicaciones armonizan bien con el software repartido por diferentes clústeres de servidores, en particular cuando éste está configurado para la operación de una interfaz de usuario de un puesto de enclavamiento, por ejemplo, con conexión para terminales móviles.

Otras ventajas de la invención resultan de la descripción y del dibujo. Asimismo, las características mencionadas anteriormente y desarrolladas aún en mayor medida pueden utilizarse de acuerdo con la invención en cada caso en sí mismas por separado o varias de ellas en combinaciones cualquiera. Las formas de realización mostradas y descritas no han de entenderse como enumeración cerrada, sino que tienen más bien carácter a modo de ejemplo para la descripción de la invención.

Descripción detallada de la invención y dibujo

La invención está representada en el dibujo y se explica en más detalle mediante ejemplos de realización. Muestran:

La figura 1 una vista general esquemática de la estructura de una primera forma de realización de un dispositivo de servidor de acuerdo con la invención con dos clústeres de servidores;

La figura 2 una vista general esquemática de la estructura de una segunda forma de realización de un dispositivo de servidor de acuerdo con la invención con tres clústeres de servidores.

Sumario de la invención

La presente invención se basa en el reparto de procesos de un control de software de un sistema de protección de transporte sobre carriles en un nivel operativo virtual por diferentes clústeres de servidores. De esta manera, los procesos se pueden someter a una migración en los servidores individuales de su clúster de servidores para asegurar una alta disponibilidad en el caso de un fallo de servidores individuales. Los procesos son del mismo tipo y los resultados de los procesos se comparan entre sí para fines de seguridad. Mediante el reparto de los procesos por diferentes clústeres de servidores está asegurado que los procesos siempre se ejecutan en diferentes servidores individuales de forma que errores de hardware individuales conducen a resultados de proceso diferentes que se pueden descubrir fácilmente en el marco de verificaciones de seguridad.

Aplicación HIS en el marco de la invención

A continuación, la invención se describe en más detalle mediante el ejemplo de la arquitectura de una aplicación HIS, en particular con respecto al reparto de proceso.

La aplicación HIS (HIS = Human machine interface for Interlocking Systems) es una aplicación SIL2 (Safety Integrity Level 2) que está desarrollada especialmente y autorizada según la norma CENELEC EN 50128. Fundamentalmente tiene la función de interfaz de usuario de un puesto de enclavamiento electrónico (ESTW) y puede estar configurada para diferentes mercados o aplicaciones en diferentes diseños para tener en cuenta las respectivas particularidades.

Todos los diseños tienen en común la arquitectura básica según la cual un proceso maestro realiza cálculos que finalmente conducen al denominado alumbrado (= representación visual en una pantalla) de estados de los elementos del puesto de enclavamiento. Estos cálculos también se realizan al mismo tiempo mediante uno o varios (según el diseño) proceso(s) esclavo y los resultados del cálculo se comparan de manera cruzada entre sí, es decir, tanto el proceso maestro como el/los proceso(s) esclavo comparan en cada caso el propio resultado de cálculo con los del otro o de los otros. En el caso de que los resultados de cálculo no coincidan, el sistema global se pone en un denominado "estado no seguro" que ya no permite determinadas acciones de operación relevantes para la seguridad.

Un diseño especial de la aplicación HIS es el denominado servidor HIS que fundamentalmente sirve para alimentar terminales de operación conectados con los alumbrados o estados calculados de los elementos del puesto de

enclavamiento.

Para cumplir con los requisitos según SIL2 de la norma EN 50128 la arquitectura HIS, según una característica, tiene que ser de tal manera que el proceso maestro y un proceso esclavo se ejecutan en diferentes procesadores (de hardware). En el caso de procesadores de múltiples núcleos, esto se puede conseguir al ligar los procesos fijamente a determinados núcleos de procesador (core binding; processor affinity). Con ello se puede garantizar que un error de cálculo de un procesador (o de un núcleo de procesador) nunca puede conducir al mismo resultado falso en el proceso maestro y en el proceso esclavo (errores dobles simultáneos se excluyen por la norma).

Al portar las aplicaciones de servidor a un nivel operativo virtual común (virtual platform) ya no se puede garantizar de manera sencilla que el proceso maestro y el proceso esclavo no se ejecutan a través del mismo error de cálculo de un procesador, ya que la asignación de un procesador virtual a un (núcleo de) procesador físico no se puede dar y demostrar sin más.

Los inventores han detectado que se pueden formar varios denominados clústeres de servidores a partir de ordenadores de servidor (servidores individuales) que ofrecen las ventajas de un nivel operativo virtual (alta disponibilidad, redundancia) y, al mismo tiempo, garantizan una separación física de procesos. Con dos clústeres de servidores de al menos dos ordenadores de servidor en cada caso, el proceso maestro se puede ejecutar en un clúster de servidores y el proceso esclavo se puede ejecutar en el otro clúster de servidores. A este respecto no se puede predecir qué (núcleo de) procesador se está utilizando en el clúster de servidores por un proceso, pero se puede excluir que los procesos en los diferentes clústeres de servidores vayan a utilizar en algún momento el mismo (núcleo de) procesador.

De esta manera también es posible la realización de la característica anteriormente descrita de la arquitectura HIS en el caso de emplear la aplicación HIS en un nivel operativo virtual.

Forma de realización de un dispositivo de servidor con dos clústeres de servidores

En la figura 1 se describe en más detalle una primera forma de realización de un dispositivo de servidor 1 de acuerdo con la invención con dos clústeres de servidores SC1, SC2. El dispositivo de servidor 1 se denomina también clúster virtual (virtual cluster).

En este caso, al dispositivo de servidor 1 pertenecen un primer clúster de servidores SC1 y un segundo clúster de servidores SC2 que están configurados separados físicamente entre sí, lo que está ilustrado en la figura 1 mediante un límite físico 2. La expresión "separados físicamente" se refiere a que los ordenadores de servidor (SRV) de los dos clústeres de servidores SC1, SC2 no están compuestos por el mismo hardware sino que son ordenadores independientes. Con ello, la separación local se puede realizar tanto mediante la formación de los clústeres de servidores SC1, SC2 en el mismo bastidor en una sala de servidores o en diferentes bastidores en la misma sala de servidores o en diferentes salas de servidores como en diferentes emplazamientos con una distancia de varios kilómetros. El factor limitador para la distancia máxima entre los clústeres de servidores SC1, SC2 es la velocidad y el tiempo de latencia de la red situada entre los mismos para la sincronización de los clústeres de servidores SC1, SC2. Las conexiones de red están representadas en la figura 1 mediante líneas de conexión sencillas.

En el primer clúster de servidores SC1 están agrupados al menos dos ordenadores de servidor (servidores individuales) SRV-1-1, SRV-1-2 de modo que forman un clúster. Asimismo, en el segundo clúster de servidores SC2 están agrupados al menos dos ordenadores de servidor (servidores individuales) SRV-2-1, SRV-2-2 de modo que forman un clúster.

En un clúster de servidores SC1, SC2 se ejecutan diferentes máquinas virtuales VM en las que, a su vez, se ejecutan las aplicaciones más diversas o sus procesos. Éstas pueden ser aplicaciones cuyos procesos están repartidos por los clústeres de servidores individuales, de las que, sin embargo, solo su actuación conjunta da como resultado una funcionalidad común, así como aplicaciones que se ejecutan individualmente en un clúster de servidores y que dan como resultado una funcionalidad independientemente de los otros procesos y aplicaciones. Ejemplos de aplicaciones y procesos de las máquinas virtuales VM son:

- HIS-Master 11a (proceso de la aplicación HIS)
- HIS-Slave 11b (proceso de la aplicación HIS)
- Proceso de control de puesto de enclavamiento-1 12a (proceso de la aplicación de control de puesto de enclavamiento)
  (Interlocking-Control Process-1 = IL-Ctrl Proc-1)
- Proceso de control de puesto de enclavamiento-2 12b (proceso de la aplicación de control de puesto de enclavamiento)
  (Interlocking-Control Process-2 = IL-Ctrl Proc-2)

- Proceso de control de protección de trenes-1 13a (proceso de la aplicación de control de protección de trenes) (Train Control-Control Process-1 = TC-Ctrl Proc-1)
- Proceso de control de protección de trenes-2 13b (proceso de la aplicación de control de protección de trenes) (Train Control-Control Process-2 = TC-Ctrl Proc-2)
- Interfaz de usuario A 14 (Human Machine Interface A = HMI A)
- Aplicación B 15 (Application B = App B)
- Interfaz de usuario C 16 (Human Machine Interface C = HMI C)
- Aplicación D 17 (Application D = App D).

El dispositivo de servidor 1 tiene un dispositivo de control de clúster (cluster control) 18 común y un dispositivo de control de memoria (storage control) 19 común para ambos clústeres de servidores SC1, SC2. Cada clúster de servidores SC1, SC2 dispone de un control propio de alta disponibilidad (high availability = HA control) 20a, 20b con el que se pueden desplazar procesos de las aplicaciones entre los ordenadores individuales SRV-1-1, SRV-1-2 o SRV-2-1, SRV-2-2 dentro del respectivo clúster de servidores SC1 o SC2, en particular si se produce un defecto en un ordenador individual. Además, cada clúster de servidores SC1, SC2 dispone en cada caso de una memoria propia (storage vol 1, storage vol 2) 21a, 21b que se puede usar por los servidores individuales del respectivo clúster SC1, SC2.

En el ejemplo de realización mostrado, el software de servidor HIS 11 está dividido en dos partes: El proceso maestro HIS 11a está implementado en el primer clúster de servidores SC1, y el proceso esclavo HIS 11b (del mismo tipo con respecto al proceso maestro HIS 11a) está implementado en el segundo clúster de servidores SC2. Por tanto, el proceso maestro HIS 11a siempre se ejecutará en uno de los servidores individuales SRV-1-1 o SRV-1-2 del primer clúster de servidores SC1, pero no en los servidores individuales del segundo clúster de servidores SC2. A la inversa, el proceso esclavo HIS 11b siempre se ejecutará en uno de los servidores individuales SRV-2-1 o SRV-2-2 del segundo clúster de servidores SC2, pero no en los servidores individuales del primer clúster de servidores SC1. De esta manera se asegura que el proceso maestro HIS 11a y el proceso esclavo HIS 11b siempre están separados físicamente entre sí. Si los resultados de proceso coinciden, el resultado de proceso que coincide es fiable.

Asimismo, en este caso, los procesos 12a y 12b del mismo tipo del software de control de puesto de enclavamiento 12 están separados físicamente entre sí, y los procesos 13a y 13b del mismo tipo del software de control de protección de trenes 13 están separados físicamente entre sí; en el caso de resultados de proceso que coinciden, a su vez, el resultado de proceso que coincide es fiable en cada caso. Las aplicaciones de software adicionales 14, 15, 16, 17 o sus procesos existen aquí en cada caso sin una parte complementaria del mismo tipo en el respectivo otro clúster de servidores SC1, SC2, esto es, solo se realizan en cada caso de manera sencilla en uno de los clústeres de servidores SC1, SC2. Esto está previsto sobre todo para aplicaciones no relevantes para la seguridad.

Forma de realización con tres clústeres de servidores

En la figura 2 está representada una forma de realización de un dispositivo de servidor de acuerdo con la invención (virtual cluster) 30 que dispone de tres clústeres de servidores SC1, SC2, SC3. La estructura del dispositivo de servidor 30 con tres clústeres de servidores SC1, SC2, SC3 se corresponde en gran parte con la estructura con dos clústeres de servidores de la figura 1, de forma que, a continuación, solo se explican las diferencias fundamentales.

En el dispositivo de servidor 30 con tres clústeres de servidores SC1, SC2, SC3 se pueden ejecutar aplicaciones que siguen al denominado principio de 2 de 3 (2 out of 3 = 2oo3). En estas aplicaciones, tres procesos del mismo tipo realizan los mismos algoritmos de cálculo y, a este respecto, obtienen respectivamente un resultado de cálculo. Estos resultados de cálculo se comparan entre sí por un comparador. Siempre que al menos dos de los tres resultados de cálculo coincidan, este resultado que coincide se considera correcto. Si el comparador detecta tres resultados diferentes, el sistema se marca como "no seguro". Según este principio funcionan, por ejemplo, la aplicación de puesto de enclavamiento o la aplicación de protección de trenes.

Un criterio para la autorización según la norma EN 50128 en los sistemas 2oo3 es que los procesos individuales se ejecuten en hardware diferente. Esto se puede asegurar mediante el dispositivo de servidor 30 de acuerdo con la invención (virtual cluster) que se basa en tres clústeres de servidores SC1, SC2, SC3 separados por límites físicos 2. Por ejemplo, los procesos de la aplicación de puesto de enclavamiento se ejecutan incrustados en cada caso en una máquina virtual VM repartidos en los tres clústeres de servidores y, por tanto, nunca usan los mismos procesadores o núcleos de procesador. Con sistemas 2oo3 también se puede alcanzar el estándar de seguridad según SIL4.

Aplicaciones típicas de sistemas 2oo3 o sus procesos son:

- Proceso de operación-1 31a (proceso de la interfaz de usuario) (Operation Control Process-1 = OC Proc-1)
- Proceso de operación-2 31b (proceso de la interfaz de usuario) (Operation Control Process-2 = OC Proc-2)
- Proceso de operación-3 31c (proceso de la interfaz de usuario) (Operation Control Process-3 = OC Proc-3)
- Proceso de puesto de enclavamiento-1 32a (proceso de la aplicación de puesto de enclavamiento) (Interlocking Process-1 = IL Proc-1)
- Proceso de puesto de enclavamiento-2 32b (proceso de la aplicación de puesto de enclavamiento) (Interlocking Process-2 = IL Proc-2)
- Proceso de puesto de enclavamiento-3 32c (proceso de la aplicación de puesto de enclavamiento) (Interlocking Process-3 = IL Proc-3)
- Proceso de protección de trenes-1 33a (proceso de la aplicación de protección de trenes) (Train Control Process-1 = TC Proc-1)
- Proceso de protección de trenes-2 33b (proceso de la aplicación de protección de trenes) (Train Control Process-2 = TC Proc-2)
- Proceso de protección de trenes-3 33c (proceso de la aplicación de protección de trenes) (Train Control Process-3 = TC Proc-3)

En el presente caso, los procesos 31a, 31b, 31c del mismo tipo o partes asociadas del software de operación 31 están repartidos por los tres clústeres de servidores SC1, SC2, SC3 de forma que los procesos 31a, 31b, 31c nunca se ejecutan en el mismo procesador o en el mismo hardware y, con ello, sus resultados de proceso no pueden ser falsos del mismo modo debido a un error de hardware individual. Lo mismo se cumple para los procesos 32a, 32b, 32c del software de puesto de enclavamiento 32 y, además, los procesos 33a, 33b, 33c del software de protección de trenes 33. Asimismo, en un clúster virtual o un dispositivo de servidor 30 con tres clústeres de servidores SC1, SC2, SC3 se pueden ejecutar aplicaciones individuales adicionales o procesos adicionales que son independientes de los sistemas 2oo3, en este caso, las aplicaciones de software HMI A 34, App B 35, HMI C 36, App D 37, HMI E 38, App F 39 adicionales.

Lista de referencias

1	Dispositivo de servidor
2	Límite físico
11a, 11b	Procesos del mismo tipo
11	Software (servidor HIS)
12a, 12b	Procesos del mismo tipo
12	Software (dispositivo de control de puesto de enclavamiento)
13a, 13b	Procesos del mismo tipo
13	Software (dispositivo de control de protección de trenes)
14-17	Software adicional
18	Dispositivo de control de clústeres
19	Dispositivo de control de memoria
20a-20c	Dispositivo de control de alta disponibilidad
21a-21c	Memoria
30	Dispositivo de servidor
31a-31c	Procesos del mismo tipo
31	Software (operación)
32a-32c	Procesos del mismo tipo
32	Software (puesto de enclavamiento)
33a-33c	Procesos del mismo tipo
33	Software (protección de trenes)
34-39	Software adicional
SC1-SC3	Clúster de servidores
SRV-1-1	Ordenador de servidor (servidor individual)
SRV-1-2	Ordenador de servidor (servidor individual)
SRV-2-1	Ordenador de servidor (servidor individual)

SRV-2-2 Ordenador de servidor (servidor individual)
SRV-3-1 Ordenador de servidor (servidor individual)
SRV-3-2 Ordenador de servidor (servidor individual)

REIVINDICACIONES

1. Dispositivo de servidor (1; 30) que opera un software para el control de una función de un sistema de protección de transporte sobre carriles, operando el software (11, 12, 13; 31, 32, 33) de manera separada físicamente entre sí al menos dos procesos (11a-11b; 12a-12b; 13a-13b; 31a-31c; 32a-32c; 33a-33c) cuyos resultados se comparan entre sí para realizar el control de la función,
caracterizado
por que el software (11, 12, 13; 31, 32, 33) se opera en un nivel operativo virtual del dispositivo de servidor (1; 30),
por que el dispositivo de servidor (1; 30) comprende al menos dos clústeres de servidores (SC1, SC2, SC3) separados físicamente entre sí,
comprendiendo cada uno de los clústeres de servidores (SC1, SC2, SC3) del dispositivo de servidor (1; 30) al menos dos servidores individuales (SRV-1-1, SRV-1-2, SRV-2-1, SRV-2-2, SRV-3-1, SRV-3-2) que permiten una migración de procesos (11a-11b; 12a-12b; 13a-13b; 31a-31c; 32a-32c; 33a-33c) entre sí en el caso de un fallo de un servidor individual (SRV-1-1, SRV-1-2, SRV-2-1, SRV-2-2, SRV-3-1, SRV-3-2),
ejecutándose los al menos dos procesos (11a-11b; 12a-12b; 13a-13b; 31a-31c; 32a-32c; 33a-33c) en máquinas virtuales (VM),
y por que el software (11, 12, 13; 31, 32, 33) comprende al menos dos partes que están instaladas en clústeres de servidores diferentes de los al menos dos clústeres de servidores (SC1, SC2, SC3) de forma que los al menos dos procesos (11a-11b; 12a-12b; 13a-13b; 31a-31c; 32a-32c; 33a-33c) se operan en clústeres de servidores diferentes de los al menos dos clústeres de servidores (SC1, SC2, SC3).

2. Dispositivo de servidor (1; 30) de acuerdo con la reivindicación 1, caracterizado por que el software (11, 12, 13; 31, 32, 33) es una aplicación de puesto de enclavamiento.

3. Dispositivo de servidor (1; 30) de acuerdo con la reivindicación 2, caracterizado por que el software (11, 12, 13; 31, 32, 33) es una aplicación para operar la interfaz de usuario de un puesto de enclavamiento controlado por ordenador, en particular con una funcionalidad para conectar terminales de operación móviles.

4. Dispositivo de servidor (1; 30) de acuerdo con la reivindicación 1, caracterizado por que el software (11, 12, 13; 31, 32, 33) es una aplicación de protección de trenes.

5. Dispositivo de servidor (1; 30) de acuerdo con una de las reivindicaciones anteriores, caracterizado por que el software (11, 12, 13; 31, 32, 33) está configurado de acuerdo con el nivel de integridad de seguridad 2 (SIL2) o superior.

6. Dispositivo de servidor (1; 30) de acuerdo con una de las reivindicaciones anteriores, caracterizado por que el software (11, 12, 13; 31, 32, 33) está configurado de acuerdo con el nivel de integridad de seguridad 4 (SIL4).

7. Dispositivo de servidor (1; 30) de acuerdo con una de las reivindicaciones 1 a 6, caracterizado por que el software (11, 12, 13) opera exactamente dos procesos (11a-11b; 12a-12b; 13a-13b) que están separados físicamente entre sí en exactamente dos clústeres de servidores diferentes (SC1, SC2).

8. Dispositivo de servidor (1; 30) de acuerdo con una de las reivindicaciones 1 a 7, caracterizado por que el dispositivo de servidor (30) comprende tres clústeres de servidores (SC1, SC2, SC3) separados físicamente entre sí, por que el software (31, 32, 33) comprende al menos tres partes que están instaladas en clústeres de servidores diferentes de los clústeres de servidores (SC1, SC2, SC3) de forma que el software (31, 32, 33) opera tres procesos (31a-31c; 32a-32c; 33a-33c) en clústeres de servidores diferentes de los tres clústeres de servidores (SC1, SC2; SC3), y por que los resultados de los procesos (31a-31c; 32a-32c; 33a-33c) se evalúan en el marco de una decisión de 2 de 3 para el control de la función del sistema de protección de transporte sobre carriles.

9. Dispositivo de servidor (1; 30) de acuerdo con una de las reivindicaciones anteriores, caracterizado por que el dispositivo de servidor (1; 30) opera al menos un software adicional (14-17; 34-39) para el control de una función adicional de un sistema de protección de transporte sobre carriles, y por que el al menos un software adicional (14-17; 34-39) está instalado y se opera únicamente en uno de los clústeres de servidores (SC1, SC2, SC3).

10. Dispositivo de servidor (1; 30) de acuerdo con la reivindicación 9, caracterizado por que el al menos un software adicional (14-17; 34-39) comprende una o varias de las siguientes aplicaciones de software:

- un sistema de planificación de horario, en particular ARAMIS-D;
- un sistema de gestión de números de tren y dirección de tren, en particular ARAMIS-C;
- un sistema de análisis de datos y métrica (business intelligence);
- un sistema de mantenimiento de trenes (maintenance centre);
- un sistema de registro y control de datos de tren (checkpoint master node);
- un sistema de registro y evaluación de componentes operativos (service management tool).

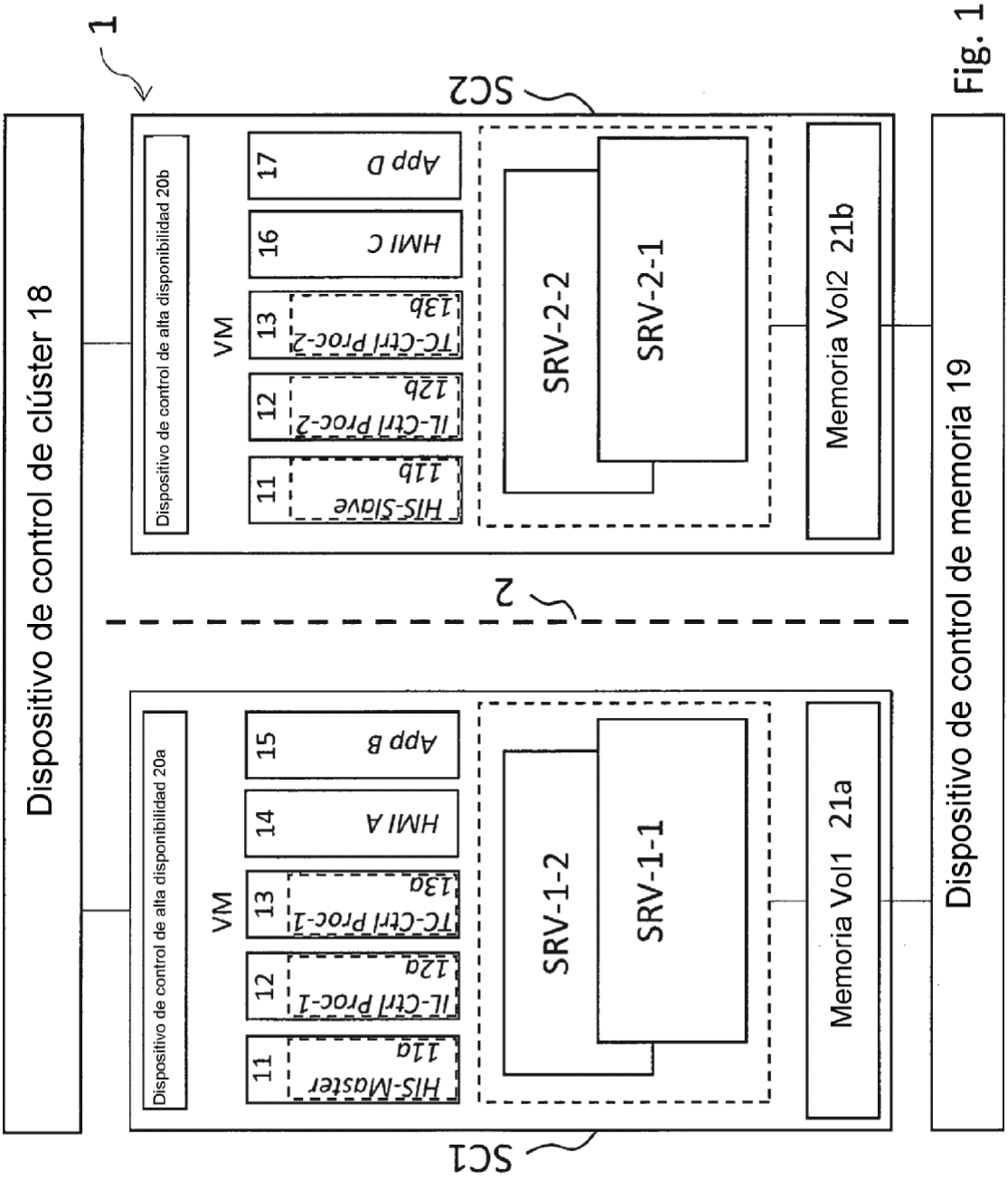
1
SC2
SC1
2
Dispositivo de control de clúster 18
Dispositivo de control de memoria 19
Dispositivo de control de alta disponibilidad 20b
Dispositivo de control de alta disponibilidad 20a
VM
11 HIS-Slave 11b
12 IL-Ctrl Proc-2 12b
13 TC-Ctrl Proc-2 13b
16 HMI C
17 App D
11 HIS-Master 11a
12 IL-Ctrl Proc-1 12a
13 TC-Ctrl Proc-1 13a
14 HMI A
15 App B
SRV-2-2
SRV-2-1
SRV-1-2
SRV-1-1
Memoria Vol2 21b
Memoria Vol1 21a

Fig. 1

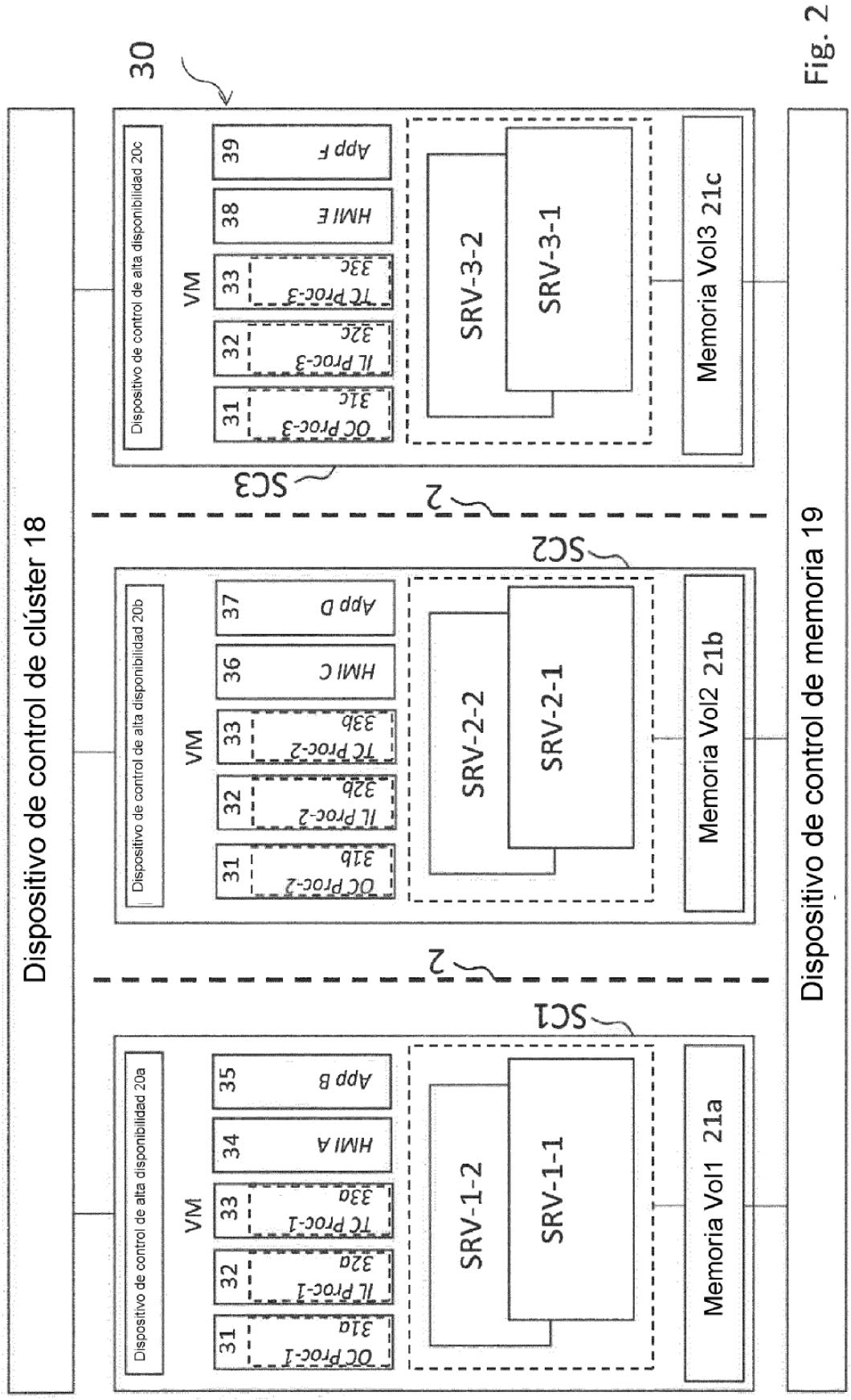
Dispositivo de control de clúster 18
30
Dispositivo de control de alta disponibilidad 20c
VM
31 OC Proc-3 31c
32 IL Proc-3 32c
33 TC Proc-3 33c
38 HMI E
39 App F
SRV-3-2
SRV-3-1
Memoria Vol3 21c
SC3
Z
Dispositivo de control de alta disponibilidad 20b
VM
31 OC Proc-2 31b
32 IL Proc-2 32b
33 TC Proc-2 33b
36 HMI C
37 App D
SRV-2-2
SRV-2-1
Memoria Vol2 21b
SC2
Z
Dispositivo de control de alta disponibilidad 20a
VM
31 OC Proc-1 31a
32 IL Proc-1 32a
33 TC Proc-1 33a
34 HMI A
35 App B
SRV-1-2
SRV-1-1
Memoria Vol1 21a
SC1
Dispositivo de control de memoria 19

Fig. 2