



(12) **Offenlegungsschrift**

(21) Aktenzeichen: **10 2018 210 711.4**

(22) Anmeldetag: **29.06.2018**

(43) Offenlegungstag: **02.01.2020**

(51) Int Cl.: **G06F 12/06 (2006.01)**

**G06F 16/00 (2019.01)**

(71) Anmelder:

**Continental Automotive GmbH, 30165 Hannover,  
DE**

(56) Ermittelter Stand der Technik:

<b>DE</b>	<b>100 32 962</b>	<b>A1</b>
<b>US</b>	<b>6 157 963</b>	<b>A</b>

(72) Erfinder:

**Castan, Jürgen, 35606 Solms, DE; Sagnes, Guy,  
35043 Marburg, DE; Kabatek, Ulrich, Dr., 64832  
Babenhause, DE**

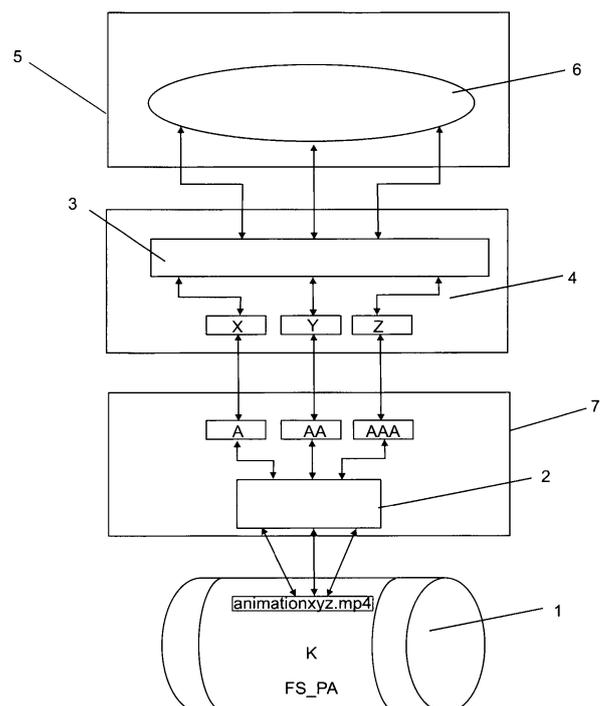
Prüfungsantrag gemäß § 44 PatG ist gestellt.

**Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.**

(54) Bezeichnung: **Verfahren und Computersystem zum priorisierten Dateizugriff, Computerprogramm und computerlesbarer Datenträger**

(57) Zusammenfassung: Die Erfindung betrifft ein Verfahren zum priorisierten Dateizugriff in einem Computersystem mit einem Betriebssystem umfassend eine von dem Betriebssystem bereitgestellte Verzeichnisstruktur, ein Speichermedium (1), ein Blockvirtualisierungssystem (2), zumindest ein erstes Blockgerät (A), welches über ein Blockvirtualisierungssystem (2) auf das Speichermedium (1) zugreift, einen ersten Einhängpunkt (X), welcher in die Verzeichnisstruktur eingehängt wird und auf das erste Blockgerät (A) zeigt, wobei zumindest ein zweites Blockgerät (AA), welches über ein Blockvirtualisierungssystem (2) auf das Speichermedium (1) zugreift und ein zweiter Einhängpunkt (Y), welcher in die Verzeichnisstruktur eingehängt wird und auf das zweite Blockgerät (AA) zeigt, umfasst wird, wobei eine erste Datei in dem Speichermedium (1), auf welches das erste Blockgerät (A) und das zweite Blockgerät (AA) zugreift, gespeichert wird, und der Zugriff auf die erste Datei über den ersten Einhängpunkt (X) mit einer ersten Priorität und der Zugriff auf die erste Datei über den zweiten Einhängpunkt (Y) mit einer von der ersten Priorität unterschiedlichen zweiten Priorität erfolgt.

Ferner betrifft die Erfindung ein Computersystem zum priorisierten Dateizugriff, ein Computerprogramm und einen computerlesbaren Datenträger



## Beschreibung

**[0001]** Die Erfindung betrifft ein Verfahren zum priorisierten Dateizugriff in einem Computersystem mit einem Betriebssystem umfassend eine von dem Betriebssystem bereitgestellten Verzeichnisstruktur, ein Speichermedium, ein Blockvirtualisierungssystem, zumindest ein erstes Blockgerät, welches über ein Blockvirtualisierungssystem auf das Speichermedium zugreift, einen ersten Einhängpunkt, welcher in die Verzeichnisstruktur eingehängt wird und auf das erste Blockgerät zeigt. Ferner betrifft die Erfindung ein Computersystem zum priorisierten Dateizugriff, ein Computerprogramm und einen computerlesbaren Datenträger.

**[0002]** Aktuelle Dateisysteme erlauben es nicht, einen lang anhaltenden und andauernden Zugriff, beispielsweise das Lesen einer großen Datei, anzuhalten, um zuerst einen wichtigeren Prozess durchzuführen.

**[0003]** Eingehende Anfragen für Lese- und Schreibzugriffe werden nach der Reihenfolge abgearbeitet, in der sie bei der jeweiligen Dateisystem-Software-Instanz eintreffen. Dies bezeichnet man auch als „FI-FO-First In - First Out“-Prinzip. Dabei werden Dateien, die zuerst gespeichert werden, auch zuerst dem Speicher wieder entnommen. Die Warteschlange in der Informatik beruht auf dem FIFO Prinzip. Die Dateien werden genau in der Reihenfolge abgerufen, in der sie vorher abgelegt wurden.

**[0004]** Es existiert jedoch keine Priorisierung von abzuarbeitenden Zugriffen bzw. Anfragen und keine Unterbrechung der aktuell zu bearbeitenden Anfrage.

**[0005]** In vielen Fällen ist es jedoch erforderlich, gewisse Zugriffe bzw. Datenströme gegenüber anderen zu bevorzugen.

**[0006]** Dies ist beispielsweise der Fall für Leseanfragen, die schnellstmöglich oder in einem bestimmten Zeitraster ausgeführt werden sollen, beispielsweise bei sicherheitskritischen Funktionalitäten, wie einer Darstellung eines Fahrzeugumfelds, in Form von einer Abstandswarnung oder aber eine Audio- oder Videowiedergabe des Fahrzeugumfelds. Diese Leseanfragen werden durch bereits wartende Anfragen verzögert, also auch durch Zugriffsanfragen, die keine oder niedrige Anforderungen bezüglich einer zeitnahen Durchführung haben, aber zeitlich früher beim Dateisystem eintreffen. Ferner werden diese Anfragen durch aktuell abzuarbeitende Anfragen verzögert. Dies kann insbesondere bei Lese- oder Schreibzugriffen, die auf große Datenmengen zugreifen, zu einer deutlichen Verzögerung führen.

**[0007]** Die bisherigen Lösungen für derartige Probleme sind aufwändige Konfigurationen oder Implemen-

tierungen, beispielsweise das sogenannte Caching für ein Audio/Video Playback. Dies erfordert jedoch einen höheren RAM-Bedarf. Eine weitere Möglichkeit liegt in der Konfiguration von Aufstart-Zeitpunkten oder der Aufteilung großer Datenmengen zum Lesen/Schreiben in kleinere Einheiten. Dadurch ergeben sich jedoch höhere Kosten.

**[0008]** Die US 20080126645 A1 offenbart ein Speichersystem und ein Dateneingabe- / Ausgabesteuerungsverfahren, beispielsweise eines, das zur Verwendung in einem Speichersystem geeignet ist, in dem eine Hostvorrichtung die Verarbeitung für verschiedene Arten von Arbeit unter Verwendung von Anwendungsprogrammen ausführt. Das Speichersystem führt dabei die Dateneingabe- / Ausgabeverarbeitung für die Prioritätsziele und die nicht vorrangige Zielgruppe bei den Frequenzen aus, die den jeweiligen Verarbeitungsprozentsätzen entsprechen, die durch die Verarbeitungsprozentsatzbestimmungseinheit bestimmt werden.

**[0009]** Die US 6253260 B1 offenbart ein System und ein Verfahren zum Verarbeiten einer Dateizugriffsanforderung (DAR). Eine Verarbeitungseinheit, wie z. B. eine Speichersteuereinheit, empfängt eine DAR, die Daten anzeigt, die auf einem Kanal zurückkehren sollen, beispielsweise einen Kanal, der mit einem Hostsystem verbunden ist, und Prioritätsinformationen für die empfangene DAR. Die Verarbeitungseinheit ruft die angeforderten Daten für die empfangene DAR aus einem Speicherbereich wie einem Cache oder einer Direktzugriffsspeichervorrichtung (DASD) ab und bestimmt, ob es eine Warteschlange von Dateneinträgen gibt, die abgerufene Daten für DARs angeben, die auf dem Kanal übertragen werden sollen. Die in der Warteschlange befindlichen DARs enthalten Prioritätsinformationen. Die Verarbeitungseinheit verarbeitet mindestens einen Dateneintrag in der Warteschlange, die Prioritätsinformation für den Dateneintrag und die Prioritätsinformation für den empfangenen DAR aufweisen, um eine Position in der Warteschlange für den empfangenen DAR zu bestimmen. Die Verarbeitungseinheit zeigt dann an, dass sich der empfangene DAR an der bestimmten Position in der Warteschlange befindet, und verarbeitet die Warteschlange, um abgerufene Daten auszuwählen, die auf dem Kanal an das Hostsystem übertragen werden sollen.

**[0010]** In der DE 112008003256 T5 wird bestimmt, ob eine Datei auf einem langsameren Laufwerk eines logischen Speichervolumenbereichs oder auf einem schnelleren Laufwerk des logischen Speichervolumenbereichs abgelegt werden soll, wobei ein Nutzer keine Kenntnis darüber zu haben braucht, ob die Datei auf dem langsameren Laufwerk oder auf dem schnelleren Laufwerk abgelegt wurde.

**[0011]** In der DE 10032962 A1 wird ein verteiltes Dateisystem mit autonomen Platten beschrieben. Das Dateisystem umfasst autonome Platten, die Applikationsdaten speichern. Legacyattributplatten speichern Metadaten des Dateisystems und die Verzeichnisstruktur. Die Verzeichnisstruktur wird auf den Legacyattributplatten im ursprünglichen Dateisystem abgelegt. Einer der Klienten des Dateisystems wird als Konfigurationsmanager zur Zugriffsüberwachung für das System verwendet. Das verteilte Dateisystem verwendet Agenten um Echtzeitanwendungen und konkurrierende Lese/Schreib-Zugriffe auf Dateien zu unterstützen.

**[0012]** Die US 4616338 offenbart eine Einrichtung zur Zwischenspeicherung von Daten, organisiert gemäß einer Warteschlange nach dem Prinzip « first in-first out », umfassend einen Schreib-Lese-Zwischenspeicher mit wahlfreiem Zugriff und ein Regelungssystem, wobei die Einrichtung ein erstes und ein zweites Ausgangs-Pufferregister aufweist, die dem Zwischenspeicher in Reihe nachgeschaltet sind, und wobei sie außerdem ein dem Zwischenspeicher vorgeschaltetes Eingangs-Pufferregister aufweist.

**[0013]** Die Aufgabe der Erfindung ist daher die Angabe eines kostengünstigen Verfahrens und eines Computersystems, mit welchen sich ein priorisierter Datei- bzw. Datenzugriff bewerkstelligen lässt, sowie ein entsprechendes Computerprogramm und ein entsprechender computerlesbarer Datenträger.

**[0014]** Die Aufgabe wird durch ein Verfahren zum priorisierten Dateizugriff mit den Merkmalen des Anspruchs 1, ein Computersystem mit den Merkmalen des Anspruchs 10 als auch ein Computerprogramm mit den Merkmalen des Anspruchs 18 sowie ein computerlesbarer Datenträger mit den Merkmalen des Anspruchs 19 gelöst.

**[0015]** In den Unteransprüchen sind weitere vorteilhafte Maßnahmen aufgelistet, die beliebig miteinander kombiniert werden können, um weitere Vorteile zu erzielen.

**[0016]** Die Aufgabe wird gelöst durch die Angabe eines Verfahrens zum priorisierten Dateizugriff in einem Computersystem mit einem Betriebssystem umfassend eine von dem Betriebssystem bereitgestellte Verzeichnisstruktur, ein Speichermedium, ein Blockvirtualisierungssystem, zumindest ein erstes Blockgerät, welches über das Blockvirtualisierungssystem auf das Speichermedium zugreift, einen ersten Einhängpunkt, welcher in die Verzeichnisstruktur eingehängt wird und auf das erste Blockgerät zeigt.

**[0017]** Im Rahmen der Erfindung wird zumindest ein zweites Blockgerät, welches über das Blockvirtualisierungssystem auf das Speichermedium zugreift

und ein zweiter Einhängpunkt, welcher in die Verzeichnisstruktur eingehängt wird und auf das zweite Blockgerät zeigt, umfasst, wobei eine erste Datei in dem Speichermedium, auf welches das erste Blockgerät als auch das zweite Blockgerät zugreift, gespeichert wird, und der Zugriff auf die erste Datei über den ersten Einhängpunkt mit einer ersten Priorität und der Zugriff auf die erste Datei über den zweiten Einhängpunkt mit einer von der ersten Priorität unterschiedlichen zweiten Priorität erfolgt.

**[0018]** In anderen Worten wird zwischen Blockvirtualisierungssystem und einer Anwendungsschicht eine Speicherzugriffsschicht, in der die Blockgeräte angeordnet sind, so ausgestaltet, dass für das nur einmal physikalisch vorhandene Speichermedium, das das Dateisystem, beispielsweise das Dateisystem FS\_PA, enthält, statt einem Blockgerät mehrere Blockgeräte zur Verfügung gestellt werden. Diese können untereinander in der Speicherzugriffsschicht, in der die Blockgeräte angeordnet sind priorisiert, werden.

**[0019]** Dabei kann als Priorität auch ein Anwendungsfall vorgesehen sein, beispielsweise ein Speichervorgriff, eine normale Anwendungszugriffspriorität oder ein priorisierter Anwendungszugriff. Dabei versteht man unter Speichervorgriff, eine Technik die von Computerprozessoren verwendet wird, um die Ausführungsleistung zu erhöhen, indem Anweisungen oder Daten von ihrem ursprünglichen Speicher im langsameren Speicher zu einem schnelleren lokalen Speicher abgerufen werden, bevor sie tatsächlich benötigt werden. Dateien können auch einzelne Daten sein.

**[0020]** Ein Blockgerät, auch blockorientiertes Gerät genannt, überträgt Daten in vorhandene Speicherblöcke eines Speichermediums und wird daher oft für parallele Datenübertragungen genutzt. Blockgeräte weisen einen betriebssystemeigenen Puffer auf.

**[0021]** Die Beziehung von Instanzen des Dateisystems zu bestimmten Einhängpunkten innerhalb eines Dateisystems oder eines root-Dateisystems wird normalerweise in der Dateisystemschiicht behandelt. Dabei wird dem Speichermedium beispielsweise ein Dateisystem FS\_PA zugeordnet. Sind in einer Recheneinheit mehrere Speichermedien vorhanden, so besitzen alle Instanzen ein Dateisystem. Gegebenenfalls kann das Speichermedium in Partitionen aufgeteilt sein, denen ebenfalls jeweils ein eigenes Dateisystem zugeordnet ist.

**[0022]** Eine Verzeichnisstruktur ist die hierarchische Struktur des Dateisystems von einem einzelnen Computer.

**[0023]** Erfindungsgemäß wurde erkannt, dass lediglich das Einführen von Einhängpunkten innerhalb

desselben Dateisystems, beispielsweise des Dateisystems FS PA, nicht zum gewünschten Effekt führen würde. Während des Auflösens des Pfades innerhalb der Dateisystemschiicht löst ein virtueller Dateisystem-Switch die Zugriffe auf die gleichen Teile des Speichermediums nur auf, obwohl sie über verschiedene Pfade / Einhängpunkte gegeben sind. Daher erhält das Blockvirtualisierungssystem, beispielsweise der Blocktreiber, unterhalb des Dateisystems keine Informationen über die angeforderte Priorität für den Dateisystemzugriff. Erfindungsgemäß wurde weiter erkannt, dass die Information über die verschiedenen Prioritäten bis zum Speichermedium bereitgestellt werden müssen. Das Verfahren stellt für das Speichermedium verschiedene Blockgeräte bereit, die jedoch alle demselben physikalischen Speichermedium bzw. derselben physikalischen Partition des Speichermediums zugeordnet sind. Dadurch kann auf dieselbe Datei in dem Speichermedium mit unterschiedlicher Priorität zugegriffen werden.

**[0024]** Erfindungsgemäß wurde zudem erkannt, dass das Dateisystem Metadaten enthält, die nur einmal vorhanden sind, und die dem Dateisystem von allen Blockgeräten zur Verfügung gestellt werden müssen.

**[0025]** Wird nun ein Zugriff auf eine Datei ausgeführt, beispielsweise wird auf die erste Datei unter dem Einhängpunkt mit niedriger Priorität zugegriffen, und greift das System auf eine zweite Datei mit höherer Priorität als der niedrigen Priorität zu, welche unter dem Einhängpunkt mit hoher Priorität angeordnet ist, so kann das Ausführen der ersten Datei unterbrochen werden und die zweite Datei zuerst ausgeführt werden. Zudem werden unabhängig von der „Warteschlange“ hochpriorisierte Dateizugriffe immer vor niedrigeren Dateizugriffen ausgeführt, unabhängig von dem „ankommen/anfrage“ des Dateizugriffes. Durch die Erfindung kann daher zwischen sicherheitsrelevanten und nicht sicherheitsrelevanten Daten unterschieden werden.

**[0026]** Die Erfindung bewirkt, dass das Blockvirtualisierungssystem auf das gesamte Speichermedium zugreift, den darüberliegenden Instanzen bzw. Schichten aber für dasselbe physikalische Speichermedium ein unterschiedliches (virtuelles) Blockgerät zur Verfügung stellen.

**[0027]** Das virtuelle Dateisystem sieht somit die Speichermedien:

/dev/virt-blockdevice-A

/dev/virt-blockdevice-AA

/dev/virt-blockdevice-AAA

**[0028]** Das Blockvirtualisierungssystem weist bevorzugt eine Tabelle auf, in der festgelegt ist, wie das Blockvirtualisierungssystem Zugriffe, die über die

verschiedenen Einhängpunkte hereinkommen, untereinander behandelt.

**[0029]** Durch die Erfindung ist es möglich eine HMI (Human Maschine Interface) Schnittstelle mit priorisiertem Zugriff auf ein und dieselbe Datei beispielsweise in einer Integrationsplattform bereitzustellen. Dazu greift beispielsweise ein Fahrerassistenzsystem einfach über den ersten Einhängpunkt mit erster Priorität auf diese Datei zu. Ist die Datei ausgeführt und beispielsweise der sicherheitsrelevante Aspekt wieder vorbei, so kann das Fahrerassistenzsystem wieder über einen anderen Einhängpunkt mit niedrigerer Priorität auf dieselbe Datei zugreifen.

**[0030]** Dadurch ist es möglich, eine Standard-Dateischnittstelle, beispielsweise POSIX, welche keine Zugriffsprioritäten bietet, mit unterschiedlich priorisiertem Dateizugriff, beispielsweise für elektronische Steuergeräte oder Apps etc. bereitzustellen.

**[0031]** Durch den mehrfachen, redundanten Zugriff ist zudem eine sichere Datenverwendung möglich.

**[0032]** Mittels der Erfindung können die Dateizugriffe für den Boot-Vorgang optimiert werden, beispielsweise durch einen verzögerten Start von weniger wichtigen Applikationen etc.

**[0033]** Weiterhin bevorzugt ist eine zweite Datei vorgesehen, welche über den zweiten Einhängpunkt mit der zweiten Priorität ausgeführt wird, wobei die Ausführung bei Zugriff auf die erste Datei über den Einhängpunkt mit der ersten Priorität unterbrochen wird. Dabei ist die erste Priorität höher als die zweite Priorität. Dadurch kann auf sicherheitsrelevante Daten bevorzugt zugegriffen werden und kurzfristig weniger priorisierte Zugriffe mit weniger sicherheitsrelevanten Daten unterbrochen werden, um die sicherheitsrelevanten Daten so schnell wie möglich zu erlangen.

**[0034]** In bevorzugter Ausgestaltung ist eine erste Datei vorgesehen, die über den zweiten Einhängpunkt mit der zweiten Priorität ausgeführt wird, wobei die Ausführung bei Zugriff auf die erste Datei oder auf Teile der ersten Datei über den ersten Einhängpunkt mit der ersten Priorität unterbrochen wird. Es wird daher bei einem Ausführen einer Datei über einen Einhängpunkt mit niedrigerer Priorität immer die Ausführung unterbrochen, sobald ein höherpriorisierter Dateizugriff ankommt.

**[0035]** Bevorzugt wird die Verzeichnisstruktur als eine als Baumstruktur ausgebildete hierarchische Verzeichnisstruktur mit einem Wurzelverzeichnis ausgebildet. Dabei wird die Verzeichnisstruktur zur Ablage von Dateien und/oder Verzeichnissen in jeder Instanz ausgebildet.

**[0036]** Weiterhin bevorzugt wird das Speichermedium zumindest als ein Speicherblock ausgestaltet. Ein Speicherblock, auch Datenblock genannt, stellt die kleinste in einem Zugriff lese- oder schreibbare Einheit eines Speichermediums dar.

**[0037]** In bevorzugter Ausgestaltung weist das Speichermedium zumindest eine Partition auf, wobei der Speicherblock der zumindest einen Partition zugeordnet wird. Auch kann der Speicherblock mehrere Partitionen aufweisen.

**[0038]** In weiterer bevorzugter Ausgestaltung wird zumindest ein drittes Blockgerät, welches auf das Speichermedium zugreift, und ein dritter Einhängpunkt, welcher in die Verzeichnisstruktur eingehängt wird und auf das dritte Blockgerät zeigt, umfasst, wobei die erste Datei in dem Speichermedium gespeichert wird, und der Zugriff auf die erste Datei über den dritten Einhängpunkt und das dritte Blockgerät mit einer dritten Priorität erfolgt, wobei die dritte Priorität von der ersten Priorität und der zweiten Priorität unterschiedlich ausgebildet wird. In einer bevorzugten weiteren Ausgestaltung wird die erste Priorität höher als die zweite Priorität und die zweite Priorität höher als die dritte Priorität ausgebildet.

**[0039]** Beispielsweise ist hier die Datei `beispiel.sql` gegeben. Diese wird durch drei verschiedene Einhängpunkte in die Verzeichnisstruktur eingehängt. Dabei weist der erste Einhängpunkt `/highprio/opt/hmi` eine erste hohe Priorität auf. Der zweite Einhängpunkt `/opt/hmi` weist eine zweite Priorität auf, welche niedriger als die erste Priorität ist, und beispielsweise als Standardpriorität gesetzt ist. Der dritte Einhängpunkte `/prefetch/opt/hmi` ist für Background Anwendungen gedacht und weist eine dritte Priorität auf, welche niedriger als die erste und die zweite Priorität ist. Von einer Anwendungsschicht kann daher auf die Datei mit einer ersten Priorität unter `/highprio/opt/hmi/beispiel.sql`, mit einer zweiten Priorität unter `/opt/hmi/beispiel.sql` und mit einer dritten Priorität unter `prefetch/opt/hmi/beispiel.sql` zugegriffen werden, um die Datei `beispiel.sql` auszuführen.

**[0040]** Auch können mehrere Einhängpunkte mit mehreren Prioritäten vorgesehen sein.

**[0041]** Besonders bevorzugt ist eine dritte Datei vorgesehen, die über den dritten Einhängpunkt mit der dritten Priorität ausgeführt wird, wobei die Ausführung bei Zugriff auf die erste Datei über den ersten Einhängpunkt mit der ersten Priorität oder bei Zugriff auf die erste Datei über den zweiten Einhängpunkt mit der zweiten Priorität unterbrochen wird. Die Ausführung einer Datei wird immer unterbrochen, wenn ein höherer priorisierter Zugriff auf eine andere Datei erfolgt.

**[0042]** Weiterhin bevorzugt wird das Speichermedium als embedded Multimedia Card (eMMC), also als eingebettete Multimedialkarte ausgebildet. Eine eMMC (embedded Multimedia Card) ist eine auf einem Flash-Speicher basierender Standard für einen universell einsetzbaren, integrierten Massenspeicher. Diese sind besonders kostengünstig.

**[0043]** In zusätzlicher bevorzugter Ausgestaltung ist ein VFS (Virtual file system, VFS) Switch vorgesehen, welcher die Einhängpunkte in die Verzeichnisstruktur einhängt. Ein VFS-Switch stellt hier auf einer VFS-Schicht Programmen eine einheitliche Schnittstelle zur Anwendungsprogrammierung zur Verfügung, um auf die unterschiedlichen Einhängpunkte zuzugreifen. Der VFS-Switch ist auf einer VFS-(Layer)schicht angeordnet.

**[0044]** Das Blockvirtualisierungssystem weist bevorzugt eine Tabelle auf, in der festgelegt ist, wie das Blockvirtualisierungssystem Zugriffe, die über die verschiedenen Einhängpunkte hereinkommen, untereinander behandelt und legt dadurch die Prioritäten fest. Die VFS-(Layer)schicht bleibt bevorzugt unangetastet.

**[0045]** Weiterhin bevorzugt wird das Betriebssystem als ein unix- oder linuxbasiertes Betriebssystem ausgestaltet.

**[0046]** Es kann vorteilhaft gesehen werden, dass bei einem Schreibzugriff auf das Speichermedium durch einen ersten Einhängpunkt der Schreibzugriff auf das Speichermedium durch einen zweiten Einhängpunkt blockiert wird. Dies gilt auch umgekehrt, das heißt, dass bei einem Schreibzugriff auf das Speichermedium durch einen zweiten Einhängpunkt der Schreibzugriff auf das Speichermedium durch einen ersten Einhängpunkt blockiert wird.

**[0047]** Weiter kann vorteilhafterweise vorgesehen sein, dass bei einem Schreibzugriff auf das Speichermedium durch einen ersten Einhängpunkt ein jeweils anderer, zweiter Einhängpunkte ausgehängt wird. Dies heißt, dass beim Senden oder beim Empfangen von Daten einer Datei immer gewartet (blockiert) wird, bis der jeweilige Schreibzugriff abgeschlossen ist.

**[0048]** Zur anschließenden Synchronisation des Speichermediums werden bevorzugt Synchronisationsmechanismen angewandt.

**[0049]** Ferner wird die Aufgabe gelöst durch die Angabe eines Computersystems mit einem Betriebssystem zum Ausführen eines wie oben beschriebenen Verfahrens, zum Ausführen eines priorisierten Dateizugriffs, umfassend eine von dem Betriebssystem bereitgestellten Verzeichnisstruktur, ein Speichermedium, ein Blockvirtualisierungssystem, zumindest ein

erstes Blockgerät, welches über das Blockvirtualisierungssystem auf das Speichermedium zugreift, ein erster Einhängpunkt, welcher in die Verzeichnisstruktur eingehängt ist und auf das erste Blockgerät zeigt, wobei

zumindest ein zweites Blockgerät, welches über das Blockvirtualisierungssystem auf das Speichermedium zugreift, und ein zweiter Einhängpunkt, welcher in die Verzeichnisstruktur eingehängt ist und auf das zweite Blockgerät zeigt, vorgesehen ist, wobei eine erste Datei in dem Speichermedium, auf welches das erste Blockgerät als auch das zweite Blockgerät zugreift, gespeichert ist, und der Zugriff auf die erste Datei über den ersten Einhängpunkt mit einer ersten Priorität und der Zugriff auf die erste Datei über den zweiten Einhängpunkt mit einer von der ersten Priorität unterschiedlichen zweiten Priorität erfolgt.

**[0050]** Das Computersystem ist wie das Verfahren, mit frei auf dem Markt verfügbaren Dateisystemen nutzbar, ohne dass diese angepasst oder geändert werden müssen. Die Erfindung ist damit unabhängig vom verwendeten Dateisystem und für bestehende Standarddateisysteme und Standardbetriebssysteme für Dateisystemzugriffe, z.B. POSIX nutzbar.

**[0051]** Bevorzugt ist eine zweite Datei vorgesehen, welche über den zweiten Einhängpunkt mit der zweiten Priorität ausgeführt ist, wobei die Ausführung bei Zugriff auf die erste Datei über den Einhängpunkt mit der ersten Priorität unterbrochen ist.

**[0052]** Weiterhin bevorzugt umfasst das Speichermedium zumindest einen Speicherblock. In weiterer bevorzugter Ausgestaltung ist zumindest ein drittes Blockgerät, welche auf das Speichermedium zugreift, und ein dritter Einhängpunkt, welcher in die Verzeichnisstruktur eingehängt ist und auf das dritte Blockgerät zeigt, umfasst, wobei die erste Datei auf dem Speichermedium gespeichert ist, und der Zugriff auf die erste Datei über den dritten Einhängpunkt und das dritte Blockgerät mit einer dritten Priorität erfolgt, wobei die dritte Priorität von der ersten Priorität und der zweiten Priorität unterschiedlich ist. Weiter bevorzugt ist die erste Priorität höher als die zweite Priorität und die zweite Priorität höher als die dritte Priorität.

**[0053]** Vorteilhafterweise ist eine dritte Datei vorgesehen, die über den dritten Einhängpunkt mit der dritten Priorität ausgeführt ist, wobei die Ausführung bei Zugriff auf die erste Datei über den ersten Einhängpunkt mit der ersten Priorität oder bei Zugriff auf die erste Datei über den zweiten Einhängpunkt mit der zweiten Priorität unterbrochen ist.

**[0054]** In weiterer bevorzugter Ausgestaltung ist bei einem Schreibzugriff auf das Speichermedium durch einen ersten Einhängpunkt der Schreibzugriff auf das Speichermedium durch einen zweiten Einhän-

gepunkt blockiert. Bevorzugt ist bei einem Schreibzugriff auf das Speichermedium durch einen ersten Einhängpunkt der jeweils andere, zweite Einhängpunkte ausgehängt.

**[0055]** Die Vorteile des Verfahrens können auch auf das Computersystem angewendet werden.

**[0056]** Ein weiterer Gegenstand der Erfindung betrifft ein Computerprogramm, umfassend Befehle, die bei der Ausführung des Programms durch den Computer diesen veranlassen, das wie oben beschriebene Verfahren auszuführen.

**[0057]** Ein weiterer Gegenstand der Erfindung betrifft einen computerlesbaren Datenträger, auf dem das wie oben beschriebene Computerprogramm gespeichert ist.

**[0058]** Weitere Merkmale, Eigenschaften und Vorteile der vorliegenden Erfindung ergeben sich aus der nachfolgenden Beschreibung unter Bezugnahme auf die beiliegenden Figuren. Darin zeigen schematisch:

**Fig. 1:** ein erstes Ausführungsbeispiel der Erfindung,

**Fig. 2:** ein zweites Ausführungsbeispiel gemäß der Erfindung,

**Fig. 3:** ein drittes Ausführungsbeispiel gemäß der Erfindung.

**[0059]** Obwohl die Erfindung im Detail durch das bevorzugte Ausführungsbeispiel näher illustriert und beschrieben wurde, ist die Erfindung nicht durch die offenbarten Beispiele eingeschränkt. Variationen hiervon können vom Fachmann abgeleitet werden, ohne den Schutzzumfang der Erfindung, wie er durch die nachfolgenden Patentansprüche definiert wird, zu verlassen.

**[0060]** **Fig. 1** zeigt ein erstes Ausführungsbeispiel der Erfindung. Das erfindungsgemäße Computersystem und Verfahren weist eine hierarchische Verzeichnisstruktur sowie - hier beispielhaft eine Datei mit einem Dateinamen animationxyz.mp4 - auf. Dabei ist eine hierarchische Verzeichnisstruktur die hierarchische Gestalt eines Dateisystems.

**[0061]** **Fig. 1** zeigt ein Speichermedium **1**. Das Speichermedium **1** kann als eMMC (emmedded MultiMedia Card) oder als Festplatte oder als ein anderes Speichermedium ausgebildet sein. Das Speichermedium **1** weist in diesem Beispiel eine Partition **K** mit dem Filesystem FS\_PA auf. Dabei kann das Filesystem auch als Dateisystem bezeichnet werden. Die Partition **K** umfasst mehrere Datenblöcke auch Speicherblöcke genannt.

**[0062]** Ferner ist ein erstes Blockgerät **A**, vorgesehen, welches auf das Speichermedium **1** bzw. auf

die Speicherblöcke der Partition **K** zeigt. Das heißt, das Blockgerät **A** wird auf die physikalische Adresse, hier auf die Speicherblöcke der Partition **K** abgebildet. Dies wird mit einem Blockvirtualisierungssystem **2**, das heißt einer Speicherverwaltungseinheit/Gerätetreiber realisiert. Bei der Blockvirtualisierung werden die Speicherblöcke eines Speichermediums **1** in einer Tabelle gespeichert, welche auch als „Mapping Table“ bezeichnet wird. Dieser ist zu entnehmen, wo genau sich bestimmte Speicherblöcke befinden. Die Blockvirtualisierung trennt somit das Blockgerät von dem physikalischen Speicher, so dass auf diesen Speicher zugegriffen werden kann, ohne Rücksicht auf das physikalische Speichermedium. Das Blockvirtualisierungssystem **2** ist auf der Speicherzugriffsschicht **7** angeordnet.

**[0063]** Die physikalische Adresse der Speicherblöcke der Partition **K** wird daher anhand des „Blockgerätes **A**“ festgehalten. Weiterhin ist ein zweites Blockgerät **AA**, vorgesehen, welches auf das Speichermedium **1** bzw. auf die Speicherblöcke der Partition **K** zeigt. Zudem ist ein drittes Blockgerät **AAA**, vorgesehen, welches auf das Speichermedium **1** bzw. auf die Speicherblöcke der Partition **K** zeigt. Das erste Blockgerät **A**, das zweite Blockgerät **AA** und das dritte Blockgerät **AAA** werden mit drei Einhängepunkten, sogenannten MountPoints **X**, **Y**, **Z** verknüpft.

**[0064]** Durch einen VFS-Switch **3** (engl. virtual File System, virtuelles Dateisystem) auf einer VFS-(Layer)schicht **4**, werden die Einhängepunkte **X**, **Y**, **Z** in der hierarchischen Verzeichnisstruktur eingehängt und sind dort sichtbar. Der VFS-Switch **3** kann die Verknüpfung der Blockgeräte **A**, **AA**, **AAA** mit den Einhängepunkten **X**, **Y**, **Z**, bewerkstelligen.

**[0065]** VFS-Switches **3** sind Anwendungsprogrammierschnittstellen, welche das „unmounten“ und „mounten“ der Einhängepunkte **X**, **Y**, **Z** übernehmen. Die Einhängepunkte **X**, **Y**, **Z** werden somit in einer Verzeichnisstruktur des Rechners eingehängt, das heißt die vorherige Verzeichnisstruktur wird also um die Einhängepunkte **X**, **Y**, **Z** erweitert. Die Einhängepunkte **X**, **Y**, **Z** werden in einer Filesystem Table angegeben, in der auch Eigenschaften vordefiniert werden können. Hier werden über die jeweiligen Einhängepunkte **X**, **Y**, **Z** die verschiedenen Prioritäten oder die verschiedenen Anwendungsfälle (use-cases) zugeordnet, das heißt dass durch das Einhängen (Mounten) auf einen bestimmten Einhängepunkt **X,Y,Z**, die Zuordnung zu den Prioritäten / Anwendungsfällen erfolgt. Das VFS-Switch **3** und die Zuordnung von Dateisystem-Instanzen zu verschiedenen Einhängepunkten **X**, **Y**, **Z** bleiben aber unverändert, so wie vom jeweiligen Betriebssystem zur Verfügung gestellt, das heißt, gibt keine Änderung in der Dateisystem-Funktionalität und keine Änderung durch das VFS-Switch **3**. Dies bedeutet, dass die ganze VFS-(Layer)schicht **4** unangetastet bleibt. Die Priorität

sierung erfolgt in dem Blockvirtualisierungssystem **2** und der Speicherzugriffsschicht **7**.

**[0066]** Ein Anwendungsfall bündelt alle Möglichkeiten, die eintreten können, wenn ein Nutzer versucht, mit Hilfe eines Verfahrens / Computersystems ein bestimmtes Ziel zu erreichen, d.h. ein Anwendungsfall beschreibt beispielsweise, wie ein Nutzer ein System verwendet, um ein bestimmtes Ziel zu erreichen.

**[0067]** Dem ersten Einhängepunkt **X** wird eine erste Priorität, dem zweiten Einhängepunkt **Y** eine zweite Priorität und dem dritten Einhängepunkt **Z** eine dritte Priorität zugewiesen, wobei die erste Priorität größer als die zweite Priorität und die zweite Priorität größer als dritte Priorität ist.

**[0068]** Hier wird die Datei animationxyz.mp4 im Verzeichnispfad „/res/video/ animationxyz.mp4“ nun auf die Einhängepunkte **X**, **Y**, **Z** gemounted. Dadurch ergeben sich nun drei unterschiedliche Verzeichnispfade mit drei unterschiedlichen Prioritäten:

/X/res/video/animationxyz.mp4

/Y/res/video/animationxyz.mp4

/Z/res/video/animationxyz.mp4

**[0069]** Durch diese Pfade kann der Anwender oder ein System, beispielsweise ein Fahrerassistenzsystem, mit verschiedenen Prioritäten auf die Datei animationxyz.mp4 zugreifen. Auf die Datei kann von einer Anwendungsschicht **5** durch eine Anwendungsprogrammierschnittstelle **6** (API) zugegriffen werden.

**[0070]** Dabei hat ein Zugriff über den Einhängepunkt /X/res/video/ animationxyz.mp4 die höchste Priorität. Zugriffe auf andere Dateien oder dieselbe Datei werden unterbrochen, sofern diese mit einem Zugriff mit niedriger Priorität ausgeführt werden.

**[0071]** Der Zugriff wird von der Speicherzugriffsschicht **7** aus priorisiert.

**[0072]** Da die Dateien der verschiedenen Blockgeräte **A**, **AA**, **AAA** auf die verschiedenen Einhängepunkte **X**, **Y**, **Z** gesetzt werden, bzw. die Blockgeräte **A**, **AA**, **AAA** den verschiedenen Einhängepunkten **X**, **Y**, **Z** zugeordnet werden, landen Anwendungszugriffe auf die Datei animationxyz.mp4 durch die verschiedenen Einhängepunkte **X**, **Y**, **Z** mit den korrespondierenden Blockgeräten **A**, **AA**, **AAA** auf der Partition **K** des Speichermediums **1**. Der Zugriff auf das Speichermedium **1** unterliegt somit den verschiedenen Prioritäten.

**[0073]** Ein Schreibzugriff auf das Speichermedium **1** erfolgt durch ein Blockgerät **A** oder **AA** oder **AAA**. Wenn ein Schreibzugriff durch verschiedene Blockgeräte möglich ist, wird der Schreibzugriff entweder ganz blockiert oder es werden bis auf das eine Block-

gerät, durch das der Schreibzugriff erfolgen soll, alle anderen Blockgeräte mittels ihres Einhängepunktes **X**, **Y**, **Z** ausgehängt.

**[0074]** In diesem Beispiel ist das Speichermedium **1** hier als eMMC ausgebildet.

**[0075]** Es können auch mehr als drei Blockgeräte mit mehr als drei Einhängepunkten, welche auf die Partition **K** des Speichermedium **1** zugreifen, vorgesehen sein.

**[0076]** Wird nun ein Zugriff auf eine Datei mit niedriger Priorität ausgeführt, also wird beispielsweise auf eine erste Datei unter dem dritten Einhängepunkt **Z** zugegriffen, und greift das System auf eine zweite Datei zu, welche unter dem ersten Einhängepunkt **X** angeordnet ist, so wird das Ausführen der ersten Datei unterbrochen und die zweite Datei wird zuerst ausgeführt.

**[0077]** Somit kann beispielsweise ein Fahrerassistenzsystem priorisiert auf die Datei animationxyz.mp4 über den Dateinamen und den Pfadnamen, hier also über den ersten Einhängepunkt **X** und den Pfadnamen /X/res/video/animationxyz.mp4 zugreifen, wenn diese Datei bzw. Daten für sicherheitsrelevante Aspekte gerade von Bedeutung sind. Andere weniger hoch priorisierte Zugriffe werden dann unterbrochen.

**[0078]** Fig. 2 zeigt ein zweites Ausführungsbeispiel der Erfindung. Hier wird über eine Anwendungsprogrammierschnittstelle **6** eine Datei wichtig.sql aufgerufen. Die Datei wurde, wie unter Fig. 1 beschrieben, über die Einhängepunkte **X**, **Y**, **Z** und die Blockgeräte **A**, **A**, **AAA** in einer Partition **K** des Speichermediums **1** gespeichert.

**[0079]** Die Datei wird von einem beispielsweise Fahrerassistenzsystem über die Anwendungsprogrammierschnittstelle **6** mit höchster Priorität aufgerufen. Der VFS-Switch **3** greift (8) über den ersten Einhängepunkt **X** auf das Blockgerät **A** zu. Dieses wiederum greift (8) über das Blockvirtualisierungssystem **2** auf die Partition **K** zu, und die Datei wichtig.sql wird ausgeführt. Die Zugriffe von der Anwendungsprogrammierschnittstelle **6** bis zur Partition **K** über den ersten Einhängepunkt **X** sind mit dem Bezugszeichen **8** gekennzeichnet. Das heißt, solange der Zugriff andauert, werden Daten der ausgeführten Datei wichtig.sql zu der Anwendungsprogrammierschnittstelle **6** zurückgeliefert, dies ist mit dem Bezugszeichen **9** gekennzeichnet.

**[0080]** Während des Zugriffs des Fahrerassistenzsystems über die Anwendungsprogrammierschnittstelle **6** und den ersten Einhängepunkt **X**, greift die Anwendungsprogrammierschnittstelle **6** erneut auf eine Datei wenigerwichtig.sql zu. Die Datei wird von

dem Fahrerassistenzsystem über die Anwendungsprogrammierschnittstelle **6** mit niedrigster Priorität aufgerufen. Der VFS-Switch **3** greift (10) über den dritten Einhängepunkt **Z** auf das Blockgerät **AAA** zu. Dieses versucht, über das Blockvirtualisierungssystem **2** auf die Partition **K**, zuzugreifen, um die Datei wenigerwichtig.sql auszuführen; es wird jedoch vom Blockvirtualisierungssystem **2**, das heißt der Speicherzugriffsschicht **7**, der Zugriff verweigert. Dieser Zugriff, hier mit dem Bezugszeichen **10** gekennzeichnet, wird solange verweigert bis das Ausführen der Datei wichtig.sql beendet worden ist; hier als Beendigungspunkt **11** gekennzeichnet. Anschließend wird die Datei wenigerwichtig.sql ausgeführt und Daten der ausgeführten Datei wenigerwichtig.sql werden zurückgeliefert, hier mit dem Bezugszeichen **12** gekennzeichnet.

**[0081]** Fig. 3 zeigt ein drittes Ausführungsbeispiel der Erfindung. Hier wird über eine Anwendungsprogrammierschnittstelle **6** eine Datei wenigerwichtig.sql aufgerufen. Die Datei wurde, wie unter Fig. 1 beschrieben, über die Einhängepunkte **X**, **Y**, **Z** und die Blockgeräte **A**, **A**, **AAA** in einer Partition **K** des Speichermediums **1** gespeichert.

**[0082]** Die Datei wird beispielsweise von einem Fahrerassistenzsystem über die Anwendungsprogrammierschnittstelle **6** mit niedrigster Priorität aufgerufen. Der VFS-Switch **3** greift über den dritten Einhängepunkt **Z** auf das Blockgerät **AAA** zu. Dieses wiederum greift über das Blockvirtualisierungssystem **2** auf die Partition **K** zu, und das Programm wenigerwichtig.sql wird ausgeführt. Die Zugriffe von der Anwendungsprogrammierschnittstelle **6** bis zur Partition **K** über den dritten Einhängepunkt **Z** sind mit dem Bezugszeichen **10** gekennzeichnet.

**[0083]** Das heißt, die Daten der ausgeführten Datei wenigerwichtig.sql werden zu der Anwendungsprogrammierschnittstelle **6** zurückgeliefert, dies wird mit dem Bezugszeichen **12** gekennzeichnet.

**[0084]** Während der Ausführung wird über die Anwendungsprogrammierschnittstelle **6** eine Datei wichtig.sql aufgerufen. Die Datei wurde, wie unter Fig. 1 beschrieben, über die Einhängepunkte **X**, **Y**, **Z** und die Blockgeräte **A**, **A**, **AAA** in einer Partition **K** des Speichermediums **1** gespeichert. Die Datei wird von dem Fahrerassistenzsystem über die Anwendungsprogrammierschnittstelle **6** mit höchster Priorität über den ersten Einhängepunkt **X** und das Blockgerät **A** aufgerufen. Der VFS-Switch **3** greift über den ersten Einhängepunkt **X** auf das Blockgerät **A** zu. Dieses greift über das Blockvirtualisierungssystem **2** auf die Partition **K** zu, um die Datei wichtig.sql auszuführen. Das Speichermedium **1**, das heißt die Partition **K** unterbricht an einem Unterbrechungspunkt **13** in diesem Fall sofort die Ausführung der Datei wenigerwichtig.sql und beginnt mit der Ausführung der Da-

tei wichtig.sql. Dieser Zugriff, hier mit dem Bezugszeichen **8**, wird solange ausgeführt bis die Ausführung der Datei wichtig.sql beendet worden ist; hier als Endzeitpunkt **14** gekennzeichnet. Anschließend wird wieder die Datei wenigerwichtig.sql ausgeführt.

#### Bezugszeichenliste

<b>1</b>	Speichermedium
<b>2</b>	Blockvirtualisierungssystem
<b>3</b>	VFS-Switch
<b>4</b>	VFS-Layerschicht
<b>5</b>	Anwendungsschicht
<b>6</b>	Anwendungsprogrammierschnittstelle
<b>7</b>	Speicherzugriffsschicht
<b>8</b>	Zugriffe
<b>9</b>	Zurückgelieferte Daten von 8
<b>10</b>	Zugriffe
<b>11</b>	Beendigungspunkt
<b>12</b>	Zurückgelieferte Daten von 10
<b>13</b>	Unterbrechungspunkt
<b>14</b>	Endzeitpunkt
<b>X,Y,Z</b>	Einhängepunkte
<b>A,AA, AAA</b>	Blockgeräte
<b>K</b>	Partition
<b>FS PA</b>	Dateisystem

**ZITATE ENTHALTEN IN DER BESCHREIBUNG**

*Diese Liste der vom Anmelder aufgeführten Dokumente wurde automatisiert erzeugt und ist ausschließlich zur besseren Information des Lesers aufgenommen. Die Liste ist nicht Bestandteil der deutschen Patent- bzw. Gebrauchsmusteranmeldung. Das DPMA übernimmt keinerlei Haftung für etwaige Fehler oder Auslassungen.*

**Zitierte Patentliteratur**

- US 20080126645 A1 [0008]
- US 6253260 B1 [0009]
- DE 112008003256 T5 [0010]
- DE 10032962 A1 [0011]
- US 4616338 [0012]

## Patentansprüche

1. Verfahren zum priorisierten Dateizugriff in einem Computersystem mit einem Betriebssystem umfassend eine von dem Betriebssystem bereitgestellten Verzeichnisstruktur, ein Speichermedium (1), ein Blockvirtualisierungssystem (2), zumindest ein erstes Blockgerät (A), welches über ein Blockvirtualisierungssystem (2) auf das Speichermedium (1) zugreift, einen ersten Einhängpunkt (X), welcher in die Verzeichnisstruktur eingehängt wird und auf das erste Blockgerät (A) zeigt,

**dadurch gekennzeichnet**, dass zumindest ein zweites Blockgerät (AA), welches über ein Blockvirtualisierungssystem (2) auf das Speichermedium (1) zugreift und ein zweiter Einhängpunkt (Y), welcher in die Verzeichnisstruktur eingehängt wird und auf das zweite Blockgerät (AA) zeigt, umfasst wird, wobei eine erste Datei in dem Speichermedium (1), auf welches das erste Blockgerät (A) und das zweite Blockgerät (AA) zugreift, gespeichert wird, und der Zugriff auf die erste Datei über den ersten Einhängpunkt (X) mit einer ersten Priorität und der Zugriff auf die erste Datei über den zweiten Einhängpunkt (Y) mit einer von der ersten Priorität unterschiedlichen zweiten Priorität erfolgt.

2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass eine zweite Datei vorgesehen ist, welche über den zweiten Einhängpunkt (Y) mit der zweiten Priorität ausgeführt wird, wobei die Ausführung bei Zugriff auf die erste Datei über den ersten Einhängpunkt (X) mit der ersten Priorität unterbrochen wird.

3. Verfahren nach Anspruch 1 oder 2, **dadurch gekennzeichnet**, dass eine erste Datei vorgesehen ist, die über den zweiten Einhängpunkt (Y) mit der zweiten Priorität ausgeführt wird, wobei die Ausführung bei Zugriff auf die erste Datei oder auf Teile der ersten Datei über den ersten Einhängpunkt (X) mit der ersten Priorität unterbrochen wird.

4. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass zumindest ein drittes Blockgerät (AAA), welches auf das Speichermedium (1) zugreift, und ein dritter Einhängpunkt (Z), welcher in die Verzeichnisstruktur eingehängt wird und auf das dritte Blockgerät (AAA) zeigt, umfasst wird, wobei die erste Datei in dem Speichermedium (1) gespeichert wird, und der Zugriff auf die erste Datei über den dritten Einhängpunkt (Z) und das dritte Blockgerät (AAA) mit einer dritten Priorität erfolgt, wobei die dritte Priorität von der ersten Priorität und der zweiten Priorität unterschiedlich ausgebildet wird.

5. Verfahren nach Anspruch 4, **dadurch gekennzeichnet**, dass die erste Priorität höher als die zweite Priorität und die zweite Priorität höher als die dritte Priorität ausgebildet wird.

6. Verfahren nach einem der Ansprüche 4 bis 5, **dadurch gekennzeichnet**, dass eine dritte Datei vorgesehen ist, die über den dritten Einhängpunkt (Z) mit der dritten Priorität ausgeführt wird, wobei die Ausführung bei Zugriff auf die erste Datei über den ersten Einhängpunkt (X) mit der ersten Priorität oder bei Zugriff auf die erste Datei über den zweiten Einhängpunkt (Y) mit der zweiten Priorität unterbrochen wird.

7. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass das Betriebssystem als ein unix- oder linuxbasiertes Betriebssystem ausgestaltet wird.

8. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass bei einem Schreibzugriff auf das Speichermedium (1) durch einen ersten Einhängpunkt (X) der Schreibzugriff auf das Speichermedium (1) durch einen zweiten Einhängpunkt (Y) blockiert wird.

9. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass bei einem Schreibzugriff auf das Speichermedium (1) durch einen ersten Einhängpunkt ein jeweils anderer, zweiter Einhängpunkte (Y) ausgehängt wird.

10. Computersystem mit einem Betriebssystem zum Ausführen eines Verfahrens, zum Ausführen eines priorisierten Dateizugriffs, nach einem der vorhergehenden Ansprüche umfassend eine von dem Betriebssystem bereitgestellten Verzeichnisstruktur, ein Speichermedium (1), ein Blockvirtualisierungssystem (2), zumindest ein erstes Blockgerät (A), welches über ein Blockvirtualisierungssystem (2) auf das Speichermedium (1) zugreift, ein erster Einhängpunkt (X), welcher in die Verzeichnisstruktur eingehängt ist und auf das erste Blockgerät (A) zeigt, **dadurch gekennzeichnet**, dass zumindest ein zweites Blockgerät (AA), welches über ein Blockvirtualisierungssystem (2) auf das Speichermedium (1) zugreift, und ein zweiter Einhängpunkt (Y), welcher in die Verzeichnisstruktur eingehängt ist und auf das zweite Blockgerät (AA) zeigt, vorgesehen ist, wobei eine erste Datei in dem Speichermedium (1), auf welches das erste Blockgerät (A) und das zweite Blockgerät (AA) zugreift, gespeichert ist, und der Zugriff auf die erste Datei über den ersten Einhängpunkt (X) mit einer ersten Priorität und der Zugriff auf die erste Datei über den zweiten Einhäng-

gepunkt (Y) mit einer von der ersten Priorität unterschiedlichen zweiten Priorität erfolgt.

11. Computersystem nach Anspruch 10, **dadurch gekennzeichnet**, dass eine zweite Datei vorgesehen ist, welche über den zweiten Einhängpunkt (Y) mit der zweiten Priorität ausgeführt ist, wobei die Ausführung bei Zugriff auf die erste Datei über den ersten Einhängpunkt (X) mit der ersten Priorität unterbrochen ist.

12. Computersystem nach einem der Ansprüche 10 bis 11, **dadurch gekennzeichnet**, dass das Speichermedium (1) zumindest einen Speicherblock umfasst.

13. Computersystem nach einem der vorhergehenden Ansprüche 10 bis 12, **dadurch gekennzeichnet**, dass zumindest ein drittes Blockgerät (AAA), welches auf das Speichermedium (1) zugreift, und ein dritter Einhängpunkt (Z), welcher in die Verzeichnisstruktur eingehängt ist und auf das dritte Blockgerät (AAA) zeigt, umfasst ist, wobei die erste Datei in dem Speichermedium (1) gespeichert ist, und der Zugriff auf die erste Datei über den dritten Einhängpunkt (Z) und das dritte Blockgerät (AAA) mit einer dritten Priorität erfolgt, wobei die dritte Priorität von der ersten Priorität und der zweiten Priorität unterschiedlich ist.

14. Computersystem nach Anspruch 13, **dadurch gekennzeichnet**, dass die erste Priorität höher als die zweite Priorität und die zweite Priorität höher als die dritte Priorität ist.

15. Computersystem nach Anspruch 13 oder 14, **dadurch gekennzeichnet**, dass eine dritte Datei vorgesehen ist, die über den dritten Einhängpunkt (Z) mit der dritten Priorität ausgeführt ist, wobei die Ausführung bei Zugriff auf die erste Datei über den ersten Einhängpunkt (X) mit der ersten Priorität oder bei Zugriff auf die erste Datei über den zweiten Einhängpunkt (Y) mit der zweiten Priorität unterbrochen ist.

16. Computersystem nach einem der vorhergehenden Ansprüche 10 bis 15, **dadurch gekennzeichnet**, dass bei einem Schreibzugriff auf das Speichermedium (1) durch einen ersten Einhängpunkt (X) der Schreibzugriff auf das Speichermedium (1) durch einen zweiten Einhängpunkt (Y) blockiert ist.

17. Computersystem nach einem der vorhergehenden Ansprüche 10 bis 16, **dadurch gekennzeichnet**, dass bei einem Schreibzugriff auf das Speichermedium (1) durch einen ersten Einhängpunkt (X) ein jeweils anderer, zweiter Einhängpunkte (Y) ausgehängt ist.

18. Computerprogramm, umfassend Befehle, die bei der Ausführung des Programms durch den Com-

puter diesen veranlassen, das Verfahren nach einem der Ansprüche 1 bis 9 auszuführen.

19. Computerlesbarer Datenträger, auf dem das Computerprogramm nach Anspruch 18 gespeichert ist.

Es folgen 3 Seiten Zeichnungen

Anhängende Zeichnungen

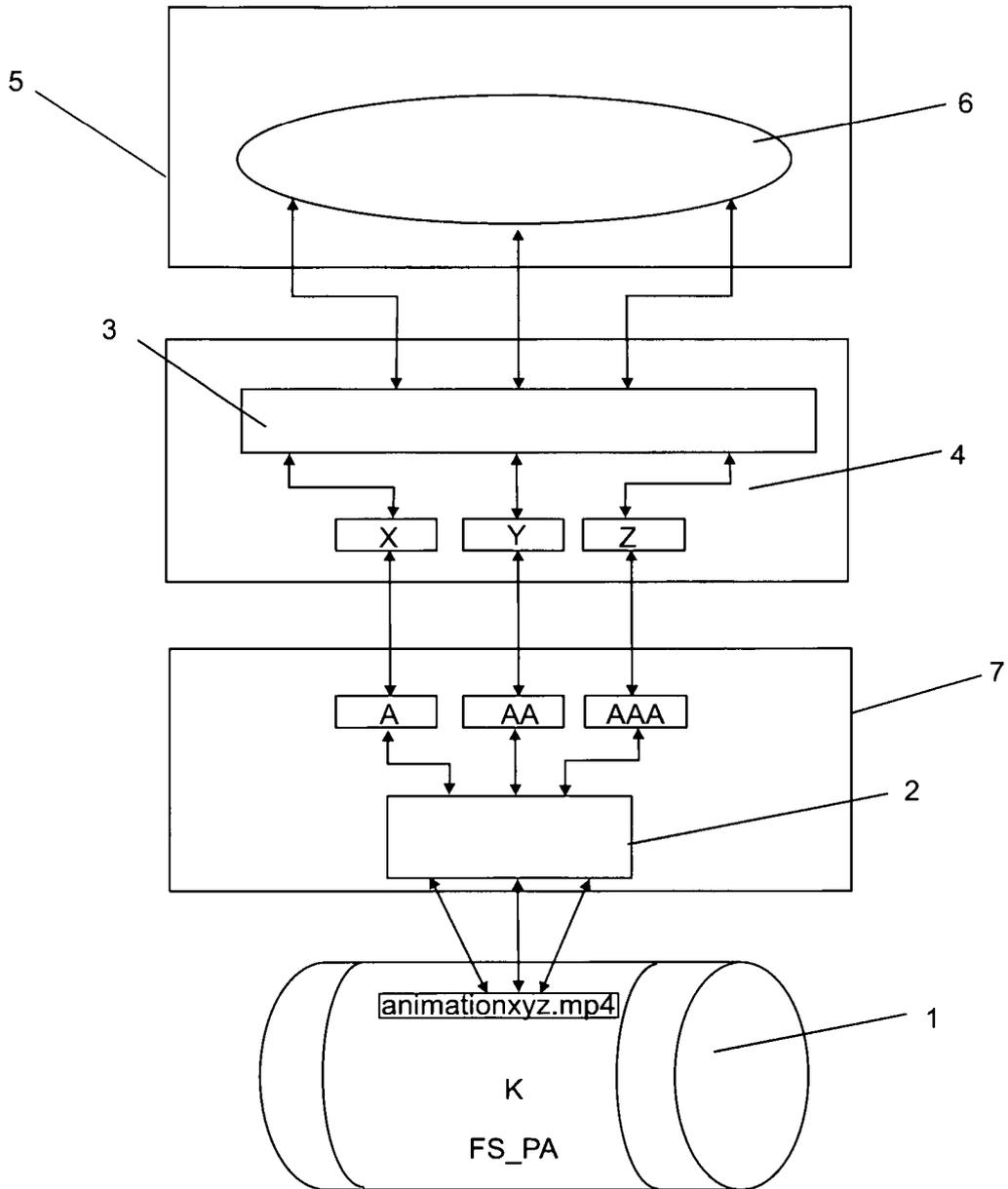


Fig. 1

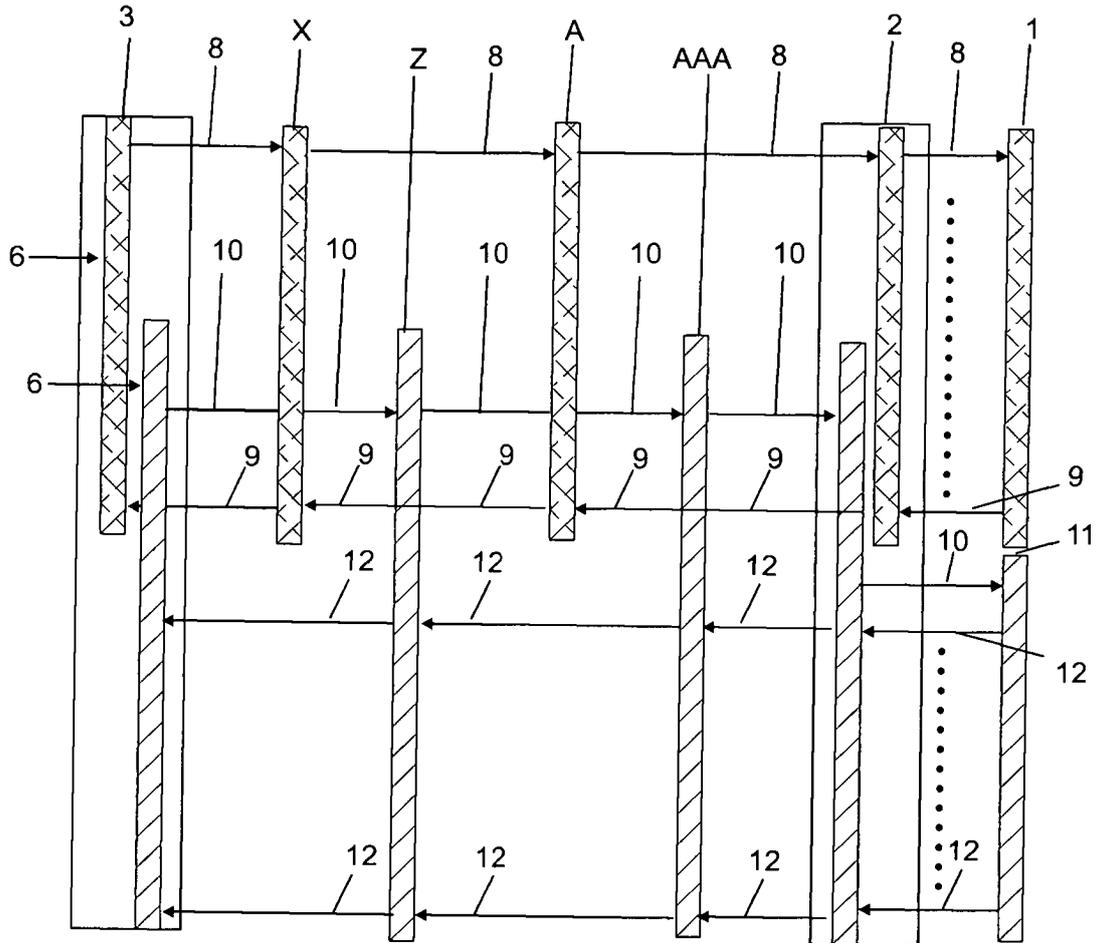


Fig. 2

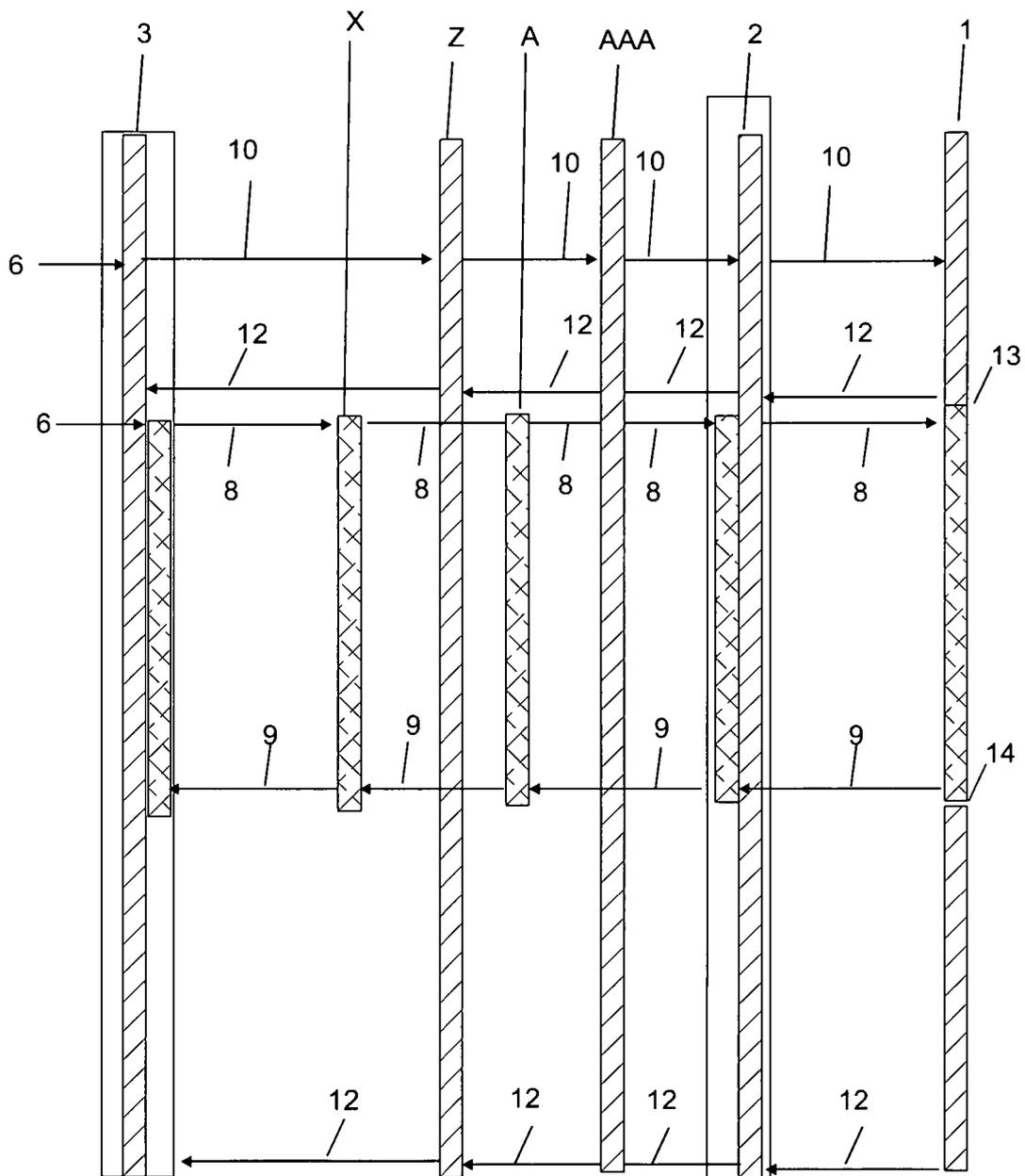


Fig. 3