(19) Deutsches Patent- und Markenamt

(10) **DE 10 2022 204 719 A1** 2023.11.16

(12) # Offenlegungsschrift

(21) Aktenzeichen: **10 2022 204 719.2**
(22) Anmeldetag: **13.05.2022**
(43) Offenlegungstag: **16.11.2023**

(51) Int Cl.: ***H04L 69/28*** (2022.01)

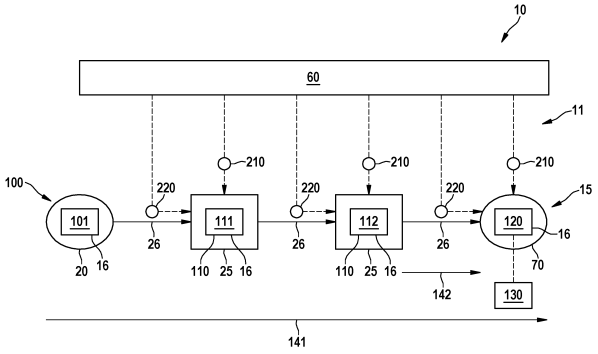
(71) Anmelder:
Robert Bosch Gesellschaft mit beschränkter Haftung, 70469 Stuttgart, DE

(72) Erfinder:
Ziegenbein, Dirk, 71691 Freiberg, DE; Dasari, Dakshina Narahari, 71034 Böblingen, DE; Hamann, Arne, 71634 Ludwigsburg, DE; Pressler, Michael, 76149 Karlsruhe, DE

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.

(54) Bezeichnung: **Verfahren für eine Bereitstellung einer Zeitinformation**

(57) Zusammenfassung: Die Erfindung betrifft ein Verfahren für eine Bereitstellung einer Zeitinformation (210) zu einem Kommunikationsinhalt (220) in einem verteilten System (10) wenigstens einer Anwendung (15), wobei die nachfolgenden Bereitstellungsschritte ausgeführt werden:
- Bereitstellen der Zeitinformation (210) über einen Zeitpunkt einer Erzeugung (101) des Kommunikationsinhalts (220),
- Bereitstellen des Kommunikationsinhalts (220) für eine verkettete Verarbeitung (100) des Kommunikationsinhalts (220) umfassend zumindest eine Zwischenverarbeitung (110) sowie eine Zielverarbeitung (120) durch unterschiedliche Verarbeitungsblöcke (16) der wenigstens einen Anwendung (15),wobei das Bereitstellen der Zeitinformation (210) für wenigstens einen der unterschiedlichen Verarbeitungsblöcke (16) erfolgt, um zumindest bei der Zielverarbeitung (120) eine Auswertung (130) der Zeitinformation (210) für eine Bewertung des verarbeiteten Kommunikationsinhalts (220) durchzuführen.



Beschreibung

[0001] Die vorliegende Erfindung betrifft ein Verfahren für eine Bereitstellung einer Zeitinformation zu einem Kommunikationsinhalt in einem verteilten System wenigstens einer Anwendung. Ferner bezieht sich die Erfindung auf ein Computerprogramm sowie ein verteiltes System zu diesem Zweck.

Stand der Technik

[0002] Aus dem Stand der Technik sind herkömmliche Middleware-Lösungen bekannt, welche Kommunikationsinhalte für Anwendungen bereitstellen und QoS (Quality of Service)-Richtlinien für Echtzeitanforderungen wie maximale Latenzen berücksichtigen. So erlaubt bspw. der Data Distribution Service (kurz „DSS“, s. https://www.dds-foundation.orgj) die deklarative Angabe einer „Deadline“, die eine maximale Dauer angibt, innerhalb der von einer Aktualisierung von ausgetauschten Kommunikationsinhalten, auch Topics bezeichnet, ausgegangen werden kann. In ähnlicher Weise erlaubt DDS die Spezifikation eines „Latenzbudgets“, das der Middleware vorgibt, wie viel Zeit für eine Lieferung von Daten zur Verfügung steht.

[0003] Herkömmliche Middleware-Lösungen implementieren diese QoS-Richtlinien jedoch nur in einem begrenzten Umfang, bspw. von einem Zeitpunkt, an dem eine Nachricht von einem Quellknoten eintrifft, bis zu dem Zeitpunkt, an dem die Nachricht zur Verarbeitung an einen Zielknoten weitergegeben wird. So können herkömmliche Lösungen einer Middleware nur die Zeitpunkte berücksichtigen, zu denen ein Datenpaket ihnen übergeben wird und zu dem sie dieses am Ziel wieder ausliefern (d.h. „Ende-zu-Ende“). Die Middleware kennt allerdings nicht die gesamten Nachrichten- und/oder Dateneigenschaften.

[0004] Darüber hinaus bieten Betriebssysteme wie QNX oder Linux reservierungsbasierte Scheduling-Mechanismen (z.B. Adaptive Partitioning für QNX oder SCHED_DEADLINE für Linux), die es erlauben, zeitliche Eigenschaften wie Reaktionszeiten (engl. Response Times) von ausgeführten Funktionalitäten zu kontrollieren. Allerdings beziehen sich diese auf einzelne Rechenknoten, und die bereitgestellten Mechanismen allein sind für verteilte Funktionen in einer verteilten Umgebung nicht geeignet. Es sind daher für Kommunikations- und Rechenressourcen im Stand der Technik Einzellösungen zur Kontrolle von Zeitanforderungen bekannt. Allerdings besteht ein Bedarf an Lösungen zur ausreichenden Behandlung von Ende-zu-Ende-Zeitanforderungen in einem verteilten System. Die hinreichende Beachtung der Anforderungen an Ende-zu-Ende-Latenzzeiten kann notwendig sein, um ein korrektes Funktionsverhalten und eine ausreichende Leistung zu gewährleisten.

Offenbarung der Erfindung

[0005] Gegenstand der Erfindung ist ein Verfahren mit den Merkmalen des Anspruchs 1, ein Computerprogramm mit den Merkmalen des Anspruchs 14 sowie ein verteiltes System mit den Merkmalen des Anspruchs 15. Weitere Merkmale und Details der Erfindung ergeben sich aus den jeweiligen Unteransprüchen, der Beschreibung und den Zeichnungen. Dabei gelten Merkmale und Details, die im Zusammenhang mit dem erfindungsgemäßen Verfahren beschrieben sind, selbstverständlich auch im Zusammenhang mit dem erfindungsgemäßen Computerprogramm sowie dem erfindungsgemäßen verteilten System, und jeweils umgekehrt, so dass bezüglich der Offenbarung zu den einzelnen Erfindungsaspekten stets wechselseitig Bezug genommen wird bzw. werden kann.

[0006] Es ist bei dem offenbarten Verfahren insbesondere vorgesehen, dass die nachfolgenden Bereitstellungsschritte ausgeführt werden, vorzugsweise automatisiert und/oder nacheinander in der angegebenen Reihenfolge, bevorzugt durch eine Middleware:

- Bereitstellen der Zeitinformation über einen Zeitpunkt einer Erzeugung des Kommunikationsinhalts, insbesondere einer erstmaligen Erzeugung des Kommunikationsinhalts, z. B. durch eine Erfassung eines Sensors und/oder eine Messung und/oder eine Eingabe und/oder dergleichen,

- Bereitstellen des Kommunikationsinhalts für eine verkettete Verarbeitung des Kommunikationsinhalts, insbesondere einer Funktionskette, umfassend zumindest eine Zwischenverarbeitung sowie eine Zielverarbeitung durch unterschiedliche Verarbeitungsblöcke der wenigstens einen Anwendung, d. h. bspw. einer einzigen oder einer Mehrzahl von Anwendungen.

[0007] Die Funktionskette kann somit durch die Verarbeitungsblöcke bereitgestellt werden. Dennoch können die Verarbeitungsblöcke unabhängig voneinander ausgeführt sein, z. B. als Prozesse einer oder mehrerer Anwendungen oder jeweils als Teil einer eigenen Anwendung. Es ist daher auch denkbar, dass die Verarbeitungsblöcke jeweils Teil einer eigenen Anwendung sind, so dass die Funktionskette durch mehrere, unabhängig voneinander ausgeführte Anwendungen gebildet sein kann. Die unabhängige Ausführung kann sich darauf beziehen, dass die Verarbeitungsblöcke jeweils durch eigene Rechenknoten parallel und/oder hardwareunabhängig voneinander ausgeführt werden. Teilweise kann

es auch vorgesehen sein, dass Verarbeitungsblöcke keine eigenen Rechenknoten aufweisen, sondern gemeinsam auf einem Rechenknoten ausgeführt werden. Dennoch ist es vorteilhaft, jeden Verarbeitungsblock ausschließlich auf einen eigenen Rechenknoten bzw. durch eine eigene Anwendung auszuführen. Insbesondere bei einer heterogenen Ausgestaltung der Rechenknoten des Systems mit unterschiedlichen Leistungsangeboten und Ressourcen kann damit eine effizientere Verteilung der Funktionskette ermöglicht werden.

[0008] Es ist ferner denkbar, dass das Bereitstellen der Zeitinformation für wenigstens einen oder mehrere oder sämtliche der unterschiedlichen Verarbeitungsblöcke erfolgt, vorzugsweise durch die Middleware, um zumindest bei der Zielverarbeitung oder auch bei der Zwischenverarbeitung eine Auswertung der Zeitinformation für eine Bewertung des verarbeiteten Kommunikationsinhalts durchzuführen. Dies hat den Vorteil, dass zuverlässig eine Latenz nicht nur von einzelnen Verarbeitungsblöcken, sondern über die gesamte verkettete Verarbeitung hinweg ermittelt werden kann.

[0009] Die Verarbeitungsblöcke können ferner durch unterschiedliche Rechenressourcen des Systems mit unterschiedlicher Leistung bereitgestellt werden und/oder unterschiedliche Kommunikationsressourcen des Systems für die Bereitstellung des Kommunikationsinhalts verwenden. Damit kann das Verfahren den Vorteil haben, dass eine verbesserte Handhabung von Ende-zu-Ende-Zeitanforderungen auf Anwendungsebene über alle Rechen- und Kommunikationsressourcen hinweg möglich ist.

[0010] Der verarbeitete Kommunikationsinhalt kann der jeweils durch die Zwischen- und/oder Zielverarbeitung verarbeitete Kommunikationsinhalt sein, welcher daher von dem ursprünglich erzeugten Kommunikationsinhalt abweicht, jedoch darauf basiert. Dies ist z. B. der Fall, wenn der erzeugte Kommunikationsinhalt ein Bild über eine Fahrzeugumgebung umfasst und der verarbeitete Kommunikationsinhalt eine Liste von Objekten umfasst, welche im Bild detektiert wurden. Der Kommunikationsinhalt kann ggf. auch als Topic ausgeführt sein. Als ein Topic wird im Folgenden eine Nutzlast bezeichnet, also Daten, welche inhaltlich oder funktional zusammenhängen. Das Topic kann dann für die verkettete Verarbeitung bevorzugt zwischen zwei Verarbeitungsblöcken, z. B. unabhängigen Anwendungen oder Prozessen oder Softwareteilen, ausgetauscht werden.

[0011] Vorteilhaft ist es zudem, wenn die Bereitstellungsschritte durch eine Middleware ausgeführt werden. Die Middleware kann bevorzugt eine Kommunikation zwischen den Verarbeitungsblöcken bereitstellen, um die Zeitinformation zugeordnet zum Kommunikationsinhalt bei der verketteten Verarbeitung bereitzustellen. Eine Middleware, welche ggf. auch als Vermittlungssoftware bezeichnet werden kann, ist insbesondere ein anwendungsneutrales Programm, welches zwischen den Verarbeitungsblöcken vermittelt, um die Komplexität und Infrastruktur reduzieren zu können. Es kann vorgesehen sein, dass eine Bereitstellung von Mechanismen und einer entsprechenden API (engl. Application programming interface, also eine Programmierschnittstelle) zur Verbesserung der Middleware erfolgt. Eine Besonderheit kann hierbei die Möglichkeit sein, wenigstens ein oder mehrere QoS- (engl. Quality of Service-) Eigenschaften an kommunizierte Topics anzuhängen und zu untersuchen. Eine solche QoS-Eigenschaft kann bspw. durch die Zeitinformation bereitgestellt sein.

[0012] Weiter ist im Rahmen der Erfindung denkbar, dass das Bereitstellen der Zeitinformation dadurch erfolgt, dass die Zeitinformation dem Kommunikationsinhalt, insbesondere durch die Middleware, zugeordnet wird.

[0013] Hierbei kann die Zeitinformation ggf. eine Latenzinformation über den Zeitpunkt der Erzeugung, bevorzugt bis zu einer Zielverarbeitung des verarbeiteten Kommunikationsinhalts, umfassen. Ferner kann die zugeordnete Zeitinformation bei der Zielverarbeitung, bevorzugt über die Middleware, für die Auswertung abgerufen werden. Dies ermöglicht es z. B., dass bei der Bewertung eine Reaktionszeit der Anwendung und/oder ein Datenalter des Kommunikationsinhalts ausgewertet werden kann. Daraus lassen sich ggf. Konsequenzen für die Zielverarbeitung ziehen, welche bspw. die Aktivierung eines Aktuators beeinflussen.

[0014] Es kann von Vorteil sein, wenn im Rahmen der Erfindung die Middleware dazu ausgeführt ist, eine Echtzeitanforderung zu registrieren, um anhand der Zeitinformation insbesondere automatisch eine Verletzung der Echtzeitanforderung zu detektieren. Hierbei kann vorzugsweise die Echtzeitanforderung eine maximale Zeitdauer einer Latenz, vorzugsweise Ende-zu-Ende-Latenz, der Anwendung definieren. Die (Ende-zu-Ende-) Latenz kann die gesamte zeitliche Dauer der verketteten Verarbeitung bezeichnen. Die Bewertung des verarbeiteten Kommunikationsinhalts kann daher als eine Ende-zu-Ende-Zeitanalyse der verketteten Verarbeitung verstanden werden. Die (Ende-zu-Ende-) Latenz kann sich ferner auf das Zeitintervall zwischen dem Auftreten einer Ursache (wie der Erzeugung, also bspw. externe Aktivität oder Abtastung eines Sensorwerts) und einer erkennbaren Wirkung (wie einem Abschluss der Verarbeitung oder Aktivieren, z. B. Bewegung eines Aktors) dieser Funktionskette beziehen. Die Ende-zu-Ende-Zeitanalyse kann damit insbesondere zur Gewährleistung der Verhal-

tensrichtigkeit in verteilten cyberphysischen Systemen oder Steuerungsanwendungen mit Echtzeitanforderungen dienen.

[0015] Beispielsweise kann es vorgesehen sein, dass die Bewertung des verarbeiteten Kommunikationsinhalts den nachfolgenden Schritt umfasst:

- Vergleichen des Zeitpunkts der Erzeugung mit einem aktuellen Zeitpunkt und mit der Echtzeitanforderung, um eine Verletzung der Echtzeitanforderung zu detektieren.

[0016] Der aktuelle Zeitpunkt kann bspw. durch einen Zeitgeber eines Rechenknotens ermittelt werden. Somit ist eine zuverlässige Überwachung der Echtzeitanforderung möglich, um problematische Situation bei der Verarbeitung, z. B. hinsichtlich einer Aktivierung eines Aktuators, zu vermeiden.

[0017] Des Weiteren kann vorgesehen sein, dass die Middleware zur Registrierung eines Rückrufs (engl. callback) in der Middleware ausgebildet ist, um den Rückruf auszuführen, sobald die Verletzung der Echtzeitanforderung detektiert wird. In anderen Worten kann in der Middleware ein Callback registriert werden, welcher ausgeführt wird, wenn die Verletzung detektiert wird. Somit ist eine schnelle und programmatische Reaktion auf die Verletzung der Echtzeitanforderung möglich. Diese kann bspw. darin bestehen, dass der entsprechende Kommunikationsinhalt verworfen wird.

[0018] Vorteilhafterweise kann im Rahmen der Erfindung vorgesehen sein, dass das System als ein cyber-physisches System ausgebildet ist, bei welchem in Abhängigkeit von der Zielverarbeitung ein Aktuator aktiviert wird, wobei der Rückruf dazu ausgeführt sein kann, die Aktivierung des Aktuators zu unterbinden und/oder zu verändern, insbesondere den Aktuator verlangsamt zu betreiben. Ein cyber-physisches System kann einen Verbund informatischer, softwaretechnischer Komponenten mit mechanischen und elektronischen Teilen bezeichnen, die vorzugsweise über eine Dateninfrastruktur, wie z. B. das Internet, kommunizieren können. Die voranstehend beschriebenen Komponenten und Teile können zusammengefasst als die Verarbeitungsblöcke bezeichnet werden.

[0019] Es ist ferner denkbar, dass das Bereitstellen der Zeitinformation für mehrere oder sämtliche der unterschiedlichen Verarbeitungsblöcke erfolgt, insbesondere, um bei der zumindest einen Zwischenverarbeitung und bei der Zielverarbeitung die Auswertung der Zeitinformation für die Bewertung des verarbeiteten Kommunikationsinhalts durchzuführen. Die Bewertung kann bspw. eine Bewertung der Latenz sein, mit welcher der verarbeitete Kommunikationsinhalt vorliegt. Somit kann eine umfassendere Überwachung von Echtzeitanforderungen wie der Latenz bei der verketteten Verarbeitung ermöglicht werden.

[0020] Vorteilhafterweise kann bei der Erfindung vorgesehen sein, dass der wenigstens einen Anwendung und/oder den Verarbeitungsblöcken jeweils eine Echtzeitanforderung oder unterschiedliche Echtzeitanforderungen zugewiesen sind, wobei die Echtzeitanforderung oder die unterschiedlichen Echtzeitanforderungen wenigstens eine der nachfolgenden Anforderungen, insbesondere QoS-Anforderungen, umfassen:

- eine maximale Ausführungszeit der Anwendung und/oder der zumindest einen Zwischenverarbeitung und/oder der Zielverarbeitung,
- eine maximale Reaktionszeit der Anwendung,
- ein maximales Datenalter des verarbeiteten Kommunikationsinhalts, wobei das Datenalter sich vorzugsweise aus der Zeitinformation über den Zeitpunkt der Erzeugung des Kommunikationsinhalts und einem aktuellen Zeitpunkt während der verketteten Verarbeitung ergibt,

wobei bevorzugt zumindest bei der Zielverarbeitung die bereitgestellte Zeitinformation ausgewertet wird, um eine Einhaltung der wenigstens einen Anforderung zu bewerten.

[0021] Anhand der bereitgestellten Zeitinformation kann bspw. ein Datenalter berechnet werden, d. h. insbesondere die Zeitdauer, die seit der Erzeugung des Kommunikationsinhalts verstrichen ist. Diese QoS-Eigenschaft kann z. B. verwendet werden, um auf der Empfängerseite, also insbesondere bei der Zielverarbeitung oder beim Aktivieren des Aktuators, festzulegen, wie lange die Daten für Funktionsberechnungen verwendet werden dürfen. Ferner kann anhand der bereitgestellten Zeitinformation auch eine Reaktionszeit berechnet werden, d. h. insbesondere die Zeitdauer, die verstrichen ist, bis der Kommunikationsinhalt einen Empfänger wie die Zielverarbeitung oder den Aktuator erreichen und zum ersten Mal verwendet werden kann. Diese QoS-Eigenschaft kann z. B. verwendet werden, um Anforderungen an die Reaktionsfähigkeit einer Funktionalität festzulegen. Auch ist es möglich, dass anhand der QoS-Eigenschaften eine Priorisierung verschiedener vorliegender Kommunikationsinhalte für die Verarbeitungen und/oder die Kommunikation und/oder eine Weiterleitung an einen der Verarbeitungsblöcke durchgeführt wird.

[0022] Der Ausdruck „maximale Reaktionszeit“ kann eine Dauer eines längsten Zeitintervalls vom Auftreten einer äußeren Ursache bis zum Zeitpunkt, an dem diese äußere Ursache vollständig verarbeitet ist, bezeichnen. Die äußere Ursache ist bspw. die Erzeugung des Kommunikationsinhalts, bspw. durch einen Sensor, und die vollständige Verarbei-

tung liegt bspw. nach Abschluss der Zielverarbeitung vor. Allgemeiner formuliert kann die maximale Reaktionszeit die Frage beantworten, wie lange es von dem Zeitpunkt, an dem der Kommunikationsinhalt erzeugt wird, maximal dauert, bis die verkettete Verarbeitung abgeschlossen ist.

[0023] Der Ausdruck „maximales Datenalter“ bezeichnet insbesondere eine Dauer eines längsten Zeitintervalls von einer Abtastung (engl. Sampling) eines Wertes bis zum letzten Zeitpunkt, an dem eine Betätigung auf der Grundlage dieses abgetasteten Wertes erfolgt. Die Abtastung erfolgt bspw. bei der Erzeugung des Kommunikationsinhalt, insbesondere durch einen Sensor, und die Betätigung ist bspw. eine Aktivierung eines Aktuators. Allgemeiner ausgedrückt beantwortet das maximale Datenalter die Frage, wie alt die Daten sind, die als Grundlage für eine Betätigung verwendet werden.

[0024] Bevorzugt kann im Rahmen der Erfindung vorgesehen sein, dass die unterschiedlichen Verarbeitungsblöcke jeweils als unterschiedliche Prozesse der wenigstens einen Anwendung oder als unterschiedliche Teile der wenigstens einen Anwendung oder als unterschiedliche Anwendungen oder Teile dieser ausgeführt sind und vorzugsweise durch unterschiedliche, heterogene Rechenknoten eines Netzwerkes ausgeführt werden. Somit kann eine verteilte Struktur bereitgestellt werden, um flexibel Ressourcen des Systems für die Verarbeitung nutzen zu können. Die Verarbeitungsblöcke, insbesondere die Prozesse, können dabei die Verarbeitungen wie die Ziel- und/oder Zwischenverarbeitungen bspw. dadurch ausführen, dass die Prozesse den Inhalt des Kommunikationsinhalts verarbeiten, also auswerten und/oder verändern.

[0025] Außerdem kann es im Rahmen der Erfindung von Vorteil sein, dass als die zumindest eine Zwischenverarbeitung mehrere Zwischenverarbeitungen vorgesehen sind, d. h., dass nicht nur eine sondern mehrere Zwischenverarbeitungen vorgesehen sind. Das Bereitstellen des Kommunikationsinhalts kann ferner dadurch erfolgen, dass der erzeugte Kommunikationsinhalt für die verkettete Verarbeitung mittels einer Kommunikation zunächst für die Zwischenverarbeitungen und ein Ergebnis der Zwischenverarbeitungen als der verarbeitete Kommunikationsinhalt für die Zielverarbeitung bereitgestellt wird. Hierbei kann die Kommunikation durch die Middleware, insbesondere eine Messaging-Middleware, zwischen den Verarbeitungsblöcken auf Anwendungsebene bereitgestellt werden, wobei das Bereitstellen der Zeitinformation ebenfalls auf dieser Anwendungsebene erfolgt. Für die Kommunikation zwischen den Verarbeitungsblöcken der Anwendung können somit bevorzugt sogenannte Messaging-Middleware-Systeme eingesetzt werden, wie z.B. Publish-Subscribe-Protokolle wie DDS oder MQTT. Auf der Grundlage der vorgeschlagenen Mechanismen und einer API können Messaging-Middleware-Systeme dabei vorteilhafterweise so erweitert werden, dass sie Ende-zu-Ende-Latenzen auf Anwendungsebene spezifizieren und bereitstellen, die die gesamte Verarbeitungskette abdecken. Oder allgemeiner ausgedrückt: Es kann eine Spezifizierung und Bereitstellung von Zeit- und Latenzinformationen auf Anwendungsebene erfolgen ab dem Zeitpunkt der Erzeugung des Kommunikationsinhalts bis zur finalen Verwendung des Kommunikationsinhalts bei der Zielverarbeitung. Hierbei können ggf. auch zwischen der Erzeugung und der finalen Verwendung bei den Zwischenverarbeitungen mehrere Hops vorgesehen sein, also Berechnungen mit jeweils einer Kommunikation dazwischen.

[0026] Unter einer Anwendungsebene kann insbesondere eine funktionale Ebene verstanden werden, welche eine Kommunikation der mindestens einen Anwendung (an wenigstens eine weitere Anwendung) umfasst, bei welcher anwendungsübergreifende Informationen berücksichtigt werden. Bspw. kann „auf Anwendungsebene“ auch bedeuten, dass hierbei eine Kommunikation in einer Anwendungsschicht gemäß dem OSI-Modell und/oder durch Funktionen der Anwendung(en) erfolgt.

[0027] Es kann optional möglich sein, dass das System als ein cyber-physisches System ausgebildet ist, in welchem die verkettete Verarbeitung eine funktionale, insbesondere Ursache-Wirkungs-, Kette bildet. Diese Funktionskette kann durch mehrere Kommunikationsverbindungen zwischen den Verarbeitungsblöcken bereitgestellt werden. Dabei kann die bereitgestellte Zeitinformation verwendet werden, um eine Verarbeitungsdauer, insbesondere Ende-zu-Ende-Latenz, der gesamten Kette zu ermitteln. Das verteilte System kann vorteilhafterweise mehrere Rechen- und Kommunikationsressourcen aufweisen. Es kann vorgesehen sein, dass die Anwendungen als verteilte Anwendungen des Systems mit einer Ende-zu-Ende-Zeitanforderungen wie z. B. einer maximalen Reaktionszeit oder eines maximalen Datenalters ausführt werden. Das erfindungsgemäße Verfahren kann den Vorteil haben, dass eine verbesserte Handhabung dieser Ende-zu-Ende-Zeitanforderungen auf Anwendungsebene über die Rechen- und Kommunikationsressourcen und über die gesamte Funktionskette hinweg ermöglicht wird.

[0028] Des Weiteren ist es denkbar, dass die Erzeugung des Kommunikationsinhalts zumindest den nachfolgenden Schritt umfasst:

- Durchführen einer Erfassung einer Umgebung, insbesondere einer Vorrichtung wie eines Fahrzeuges, durch einen Sensor, um die erfasste Umgebung als den erzeugten Kommunikationsinhalt bereitzustellen.

[0029] Ferner ist es denkbar, dass die zumindest eine Zwischenverarbeitung und die Zielverarbeitung zumindest die nachfolgenden Schritte umfasst:

- Detektieren von mindestens einem Objekt in der erfassten Umgebung der Vorrichtung bzw. des Fahrzeuges anhand des erzeugten Kommunikationsinhalts, insbesondere durch eine erste Zwischenverarbeitung,

- Berechnen eines Pfades zur Fortbewegung der Vorrichtung bzw. des Fahrzeuges, insbesondere in der Umgebung, anhand des mindestens einen detektierten Objekts, um den berechneten Pfad als den verarbeiteten Kommunikationsinhalt bereitzustellen, insbesondere durch eine zweite Zwischenverarbeitung,

- Aktivieren eines Aktuators, um die Vorrichtung bzw. das Fahrzeug entlang des berechneten Pfades fortzubewegen.

[0030] Dabei kann das Aktivieren des Aktuators in Abhängigkeit von der Bewertung des verarbeiteten Kommunikationsinhalts erfolgen. Der Aktuator kann dabei bspw. ein Steuerungsgerät des Fahrzeuges sein, um das Fahrzeug zumindest teilweise autonom fortzubewegen und zu steuern. Das vorgeschlagene Verfahren und insbesondere die damit bereitgestellten Mechanismen und/oder die API haben den Vorteil, dass das Alter und die Reaktionszeiten von Kommunikationsinhalten bzw. Topics berücksichtigt werden können. Dadurch können die Funktionalitäten zeit- und latenzbewusst sein, was eine robustere Leistung ermöglicht, z. B. durch vernetzte Kontrollmethoden. Außerdem erleichtert dies die Fusion von Signalen, die von mehreren verschiedenen Sensoren geliefert werden (es kann somit eine Sensorfusion vorgesehen sein). Alternativ oder zusätzlich zur Anwendung für ein zumindest teilweise autonomes Fahren ist es denkbar, dass das erfindungsgemäße Verfahren bei anderen automatisierten Fahranwendungen und/oder bei einer verteilten Middleware und/oder einem Robotersystem und/oder bei einer industriellen Automatisierung zum Einsatz kommt.

[0031] Ebenfalls Gegenstand der Erfindung ist ein Computerprogramm, vorzugsweise Computerprogrammprodukt, umfassend Befehle, die bei der Ausführung des Computerprogramms durch einen Computer diesen veranlassen, das erfindungsgemäße Verfahren auszuführen. Damit bringt das erfindungsgemäße Computerprogramm die gleichen Vorteile mit sich, wie sie ausführlich mit Bezug auf ein erfindungsgemäßes Verfahren offenbart worden sind.

[0032] Als der Computer kann bspw. eine Datenverarbeitungsvorrichtung vorgesehen sein, welcher das Computerprogramm ausführt. Der Computer kann wenigstens einen Prozessor zur Ausführung des Computerprogramms aufweisen. Auch kann ein nicht-flüchtiger Datenspeicher vorgesehen sein, in welchem das Computerprogramm hinterlegt und von welchem das Computerprogramm durch den Prozessor zur Ausführung ausgelesen werden kann. Der Computer kann bspw. ein Rechenknoten des Systems sein.

[0033] Ebenfalls Gegenstand der Erfindung kann ein computerlesbares Speichermedium sein, welches das erfindungsgemäße Computerprogramm umfasst. Das Speichermedium ist bspw. als ein Datenspeicher wie eine Festplatte und/oder ein nicht-flüchtiger Speicher und/oder eine Speicherkarte ausgebildet. Das Speichermedium kann z. B. in den Computer integriert sein.

[0034] Darüber hinaus kann das erfindungsgemäße Verfahren auch als ein computerimplementiertes Verfahren ausgeführt sein.

[0035] Ebenfalls Gegenstand der Erfindung ist ein verteiltes System, welches eingerichtet ist, das erfindungsgemäße Verfahren auszuführen. Damit bringt das erfindungsgemäße verteilte System die gleichen Vorteile mit sich, wie sie ausführlich mit Bezug auf ein erfindungsgemäßes Verfahren beschrieben worden sind.

[0036] Weitere Vorteile, Merkmale und Einzelheiten der Erfindung ergeben sich aus der nachfolgenden Beschreibung, in der unter Bezugnahme auf die Zeichnungen Ausführungsbeispiele der Erfindung im Einzelnen beschrieben sind. Dabei können die in den Ansprüchen und in der Beschreibung erwähnten Merkmale jeweils einzeln für sich oder in beliebiger Kombination erfindungswesentlich sein. Es zeigen:

Fig. 1 schematisch eine Visualisierung eines erfindungsgemäßen Verfahrens und Systems,

Fig. 2 schematisch eine weitere Visualisierung eines erfindungsgemäßen Verfahrens und Systems,

Fig. 3 schematisch eine Visualisierung eines erfindungsgemäßen Verfahrens und Systems sowie eines Computerprogramms.

[0037] In den nachfolgenden Figuren werden für die gleichen technischen Merkmale auch von unterschiedlichen Ausführungsbeispielen die identischen Bezugszeichen verwendet.

[0038] In **Fig.** 1 ist beispielhaft eine Funktionskette in einem verteilten System 10 wenigstens einer Anwendung 15 dargestellt, welche auch als eine Ursache-Wirkungs-Kette verstanden werden kann. Durch die Funktionskette ist eine verkette Verarbeitung 100 eines Kommunikationsinhalts 220 möglich. Dabei kann die verkettete Verarbeitung 100 dazu dienen, den Kommunikationsinhalt 220 nach Auftreten der Ursache, d. h. nach der Erzeugung 101 des Kom-

munikationsinhalts 220, in mehreren Zwischenverarbeitungen 110 zu verarbeiten, um in einem letzten Verarbeitungsschritt bei der Zielverarbeitung 120 eine Wirkung zu erhalten. Die Ursache ist bspw. die Erzeugung 101 des Kommunikationsinhalts 220 durch einen Sensor 20 und die Wirkung die Aktivierung eines Aktuators 70.

[0039] Gemäß dem Verfahren und insbesondere durch eine Middleware 60 kann ferner ein Bereitstellen des Kommunikationsinhalts 220 für die verkettete Verarbeitung 100 des Kommunikationsinhalts 220 erfolgen. Die Zwischenverarbeitungen 110 sowie die Zielverarbeitung 120 können dabei als Teile der verketteten Verarbeitung 100 durch unterschiedliche Verarbeitungsblöcke 16 der wenigstens einen Anwendung 15 bereitgestellt werden. Die unterschiedlichen Verarbeitungsblöcke 16 können jeweils als unterschiedliche Prozesse der wenigstens einen Anwendung 15 oder als unterschiedliche Teile der wenigstens einen Anwendung 15 oder vollständig als eigenständige Anwendungen 15 ausgeführt sein und vorzugsweise durch unterschiedliche, heterogene Rechenknoten 105 eines Netzwerkes 11 ausgeführt werden.

[0040] Mit herkömmlichen Middleware-Lösungen wären die Möglichkeiten begrenzt, um eine gesamte Ende-zu-Ende-Latenz der Funktionskette zu ermitteln. Folglich wäre es bei der Aktivierung des Aktuators 70 nicht möglich, Rückschlüsse auf das Alter der Daten des Kommunikationsinhalts 220 zu ziehen. Es kann ggf. nur auf ein Datenalter für einen kleinen Teil der Ursache-Wirkungs-Kette geschlossenen werden, z. B. die Latenz 142 des letzten Teils vor der Zielverarbeitung 120. Daher ist gemäß einem erfindungsgemäßen Verfahren eine Verbesserung dadurch vorgesehen, dass eine Zeitinformation 210 über einen Zeitpunkt der Erzeugung 101 des Kommunikationsinhalts 220 bereitgestellt wird. Es erfolgt dabei das Bereitstellen der Zeitinformation 210 für wenigstens einen der unterschiedlichen Verarbeitungsblöcke 16, um zumindest bei der Zielverarbeitung 120 eine Auswertung 130 der Zeitinformation 210 für eine Bewertung des verarbeiteten Kommunikationsinhalts 220 durchzuführen. Dies ermöglicht eine Auswertung der Ende-zu-Ende-Latenz 141 der gesamten Funktionskette, und somit eine höhere Sicherheit bei der Zielverarbeitung 120.

[0041] In den **Fig.** 1 bis **Fig.** 3 ist ferner schematisch dargestellt, dass das Bereitstellen des Kommunikationsinhalts 220 und dazu zugeordnet der Zeitinformation 210 durch eine Middleware 60 erfolgen kann. Die Middleware 60 kann zudem eine Kommunikation 26 zwischen den Verarbeitungsblöcken 16 ermöglichen, um die Zeitinformation 210 und den Kommunikationsinhalt 220 bei der verketteten Verarbeitung 100 bereitzustellen. Es kann vorgesehen sein, dass die Zeitinformation 210 eine Latenzinformation 141 über den Zeitpunkt der Erzeugung 101 bis zu einer Zielverarbeitung 120 des verarbeiteten Kommunikationsinhalts 220 umfasst. Die zugeordnete Zeitinformation 210 kann bei der Zielverarbeitung 120 über die Middleware 60 für die Auswertung 130 abgerufen werden, ggf., um bei der Bewertung eine Reaktionszeit der Anwendung 15 auszuwerten.

[0042] Die Zuordnung der Zeitinformation 210 zum Kommunikationsinhalt 220 kann z. B. durch den folgenden Mechanismus erfolgen, wobei der Kommunikationsinhalt 220 als Topic bezeichnet wird:

- data_produced(topic_name, topic_instance):

  Damit kann ein Mechanismus bereitgestellt sein, durch welchem die Middleware 60 der gegebenen Instanz eines Topics („topic_instance“) mit dem angegebenen Namen („topic_name“) die aktuelle Zeit zuordnet (z. B. mit Hilfe eines Zeitstempels), die bspw. dem Zeitpunkt der Erzeugung 101 entspricht.

[0043] Eine weitere Möglichkeit ist es, diese Zuordnung unter Angabe eines zusätzlichen manuellen Zeitwerts durchzuführen:

- data_produced(topic_name, topic_instance, elaped_time in ms):

  Bei diesem Mechanismus kann zusätzlichen zum vorgenannten Mechanismus mit dem Parameter „elapsed_time in ms“ manuell ein Zeitwert als Produktionszeit festgelegt werden. Dies kann nützlich sein, um Verzögerungen von physischen Komponenten (z. B. Abtast- und Betätigungsverzögerungen) zu berücksichtigen, die nicht direkt auf Softwareebene berücksichtigt werden können. Darüber hinaus kann diese Funktionalität genutzt werden, um eine Ursache-Wirkungs-Kette einschließlich Datentransformationen durch ein verteiltes System zu verfolgen. Zum Beispiel in Fällen, in denen empfangene Topics für Berechnungen verwendet werden, die zu neuen Topics führen, die dann durch das System weitergeleitet werden.

[0044] Die Bereitstellung der Zeitinformation 210 z. B. bei der Zielverarbeitung 120 kann beispielhaft durch folgenden Mechanismus ermöglicht werden:

- getTimeFromProduction(topic_name, topic_instance):

  Durch diesen Mechanismus kann die Middleware 60 einen Zeitpunkt zurückgeben, zu dem das Topic „topic_instance“ von „topic_name“ erzeugt wurde.

[0045] In **Fig.** 3 ist gezeigt, dass die Middleware 60 dazu ausgeführt sein kann, eine Echtzeitanforderung 140 zu registrieren, um anhand der Zeitinformation

210 eine Verletzung der Echtzeitanforderung 140 zu detektieren. Die Echtzeitanforderung 140 kann konkret eine maximale Zeitdauer einer in **Fig.** 1 veranschaulichten Ende-zu-Ende-Latenz 141 der Anwendung 15 definieren. Die Bewertung des verarbeiteten Kommunikationsinhalts 220 kann ein Vergleichen des Zeitpunkts der Erzeugung 101 mit einem aktuellen Zeitpunkt und mit der Echtzeitanforderung 140 umfassen, um die Verletzung der Echtzeitanforderung 140 zu detektieren. Dieser Vergleich kann z. B. bei der Zielverarbeitung 120 stattfinden.

[0046] Es können zur Registrierung der Echtzeitanforderung 140 bspw. noch die folgenden Mechanismen vorgesehen sein:

- setMaxAge(topic_name, max_age in ms):

Es kann hiermit ein Mechanismus für die Middleware 60 bereitgestellt sein, der verwendet wird, um alle empfangenen Topics (also „topic_instances“ des Typs „topic_name“) mit einer Altersbeschränkung (bspw. angegeben in Millisekunden) zu versehen. Bspw. können die Topics dann automatisch nach Überschreiten der Altersbeschränkung verworfen oder gesperrt werden.

[0047] Die Middleware 60 kann ggf. außerdem Kommunikationsinhalte 220, die bereits gegen eine Altersbeschränkung verstoßen haben, nicht an Verarbeitungsblöcke weiterleiten, um so den Datenverkehr im Netz zu verringern. Darüber hinaus können auch andere Möglichkeiten vorgesehen sein. Bspw. kann auch funktional reagiert werden, d. h. die Daten können anders oder angepasst verarbeitet werden.

- setMaxReactionTime(topic_name max_reaction in ms):

Zusätzlich zum vorgenannten Mechanismus kann hierbei die angegebene Zeit „max_reaction“ einer maximal erforderlichen Reaktionszeit z. B. in Millisekunden angegeben werden.

[0048] Ferner kann es gemäß **Fig.** 3 auch möglich sein, dass die Middleware 60 zur Registrierung eines Rückrufs 150 (engl. callbacks) ausgeführt ist, um den Rückruf 150 auszuführen, sobald die Verletzung der Echtzeitanforderung 140 detektiert wird. Im konkreten Beispiel kann das System 10 als ein cyber-physisches System ausgebildet sein, bei welchem in Abhängigkeit von der Zielverarbeitung 120 ein Aktuator 70 aktiviert wird, wobei der Rückruf 150 dazu ausgeführt ist, die Aktivierung des Aktuators 70 zu unterbinden, worunter auch verstanden wird, dass angepasst reagiert wird. Bspw. ist hierdurch der Einsatz für ein autonomes Fahren bzw. für eine Pfadberechnung bei einem zumindest teilweise autonomen Fahren möglichen. Für die Erzeugung 101 des Kommunikationsinhalts 220 kann bspw. eine Umgebung eines Fahrzeuges 50 durch einen Sensor 20 erfasst werden, um die erfasste Umgebung als den erzeugten Kommunikationsinhalt 220 bereitzustellen. Die verkettete Verarbeitung kann sodann ein Detektieren von mindestens einem Objekt in der erfassten Umgebung des in **Fig.** 2 dargestellten Fahrzeuges 50 anhand des erzeugten Kommunikationsinhalts 220 umfassen. Anschließend kann ein Berechnen eines Pfades zur Fortbewegung des Fahrzeuges 50 anhand des mindestens einen detektierten Objekts durchgeführt werden, um den berechneten Pfad als den verarbeiteten Kommunikationsinhalt 220 bereitzustellen. Sowohl das Detektieren als auch das Berechnen können durch eigene Zwischenverarbeitungen 110 durchgeführt werden. Anschließend kann bei der Zielverarbeitung 120 ein Aktivieren des Aktuators 70 erfolgen, um das Fahrzeug 50 entlang des berechneten Pfades fortzubewegen. Das Aktivieren kann jedoch in Abhängigkeit von der Bewertung des verarbeiteten Kommunikationsinhalts 220 erfolgen. Im gegebenen Beispiel kann die Aktivierung unterbunden werden, wenn anhand der Bewertung erkannt wird, dass der geplante Pfad auf veralteten Daten basiert. Dies hat den Vorteil, dass ungewünschte Situationen vermieden werden kann, z. B. falls der verfolgte Pfad auf Objekten basiert, die vor zu langer Zeit erkannt wurden, da die tatsächliche Situation mittlerweile völlig anders sein kann (hinsichtlich der Position der Objekte, neue Objekte usw.).

[0049] Die Registrierung eines callbacks kann z. B. durch folgenden Mechanismus erfolgen:

- registerExceptionCallback(callback_method, topic_name):

Dieser Mechanismus der Middleware 60 ermöglicht es, Rückrufe (d.h. engl. „callbacks“) zu registrieren, die automatisch von der Middleware 60 aufgerufen werden, wenn an dem Rechenknoten, der den Rückruf registriert hat, eine Instanz des Topics des Typs „topic_name“ empfangen wird, die die angegebenen Zeitanforderungen verletzt.

[0050] Gemäß **Fig.** 2 kann die zumindest eine Zwischenverarbeitung 110 eine erste und zweite Zwischenverarbeitung 111,112 sowie ggf. weitere Zwischenverarbeitungen 110 umfassen. Das Bereitstellen des Kommunikationsinhalts 220 kann dadurch erfolgen, dass der erzeugte Kommunikationsinhalt 220 für die verkettete Verarbeitung 100 mittels der Kommunikation 26 zunächst für die Zwischenverarbeitungen 111,112 und ein Ergebnis 220 der Zwischenverarbeitungen 111,112 für die Zielverarbeitung 120 bereitgestellt wird, wobei die Kommunikation 26 durch die Middleware 60, insbesondere eine Messaging-Middleware, zwischen den Verarbeitungsblöcken 16 auf Anwendungsebene bereitgestellt werden kann. Auch das Bereitstellen der Zeit-

information 210 kann auf dieser Anwendungsebene erfolgen.

[0051] In **Fig.** 3 ist ebenfalls ein Computerprogramm 200 visualisiert, welches durch einen Rechenknoten 25 ausgeführt werden kann, um die Schritte eines erfindungsgemäßen Verfahrens bereitzustellen.

[0052] Die voranstehende Erläuterung der Ausführungsformen beschreibt die vorliegende Erfindung ausschließlich im Rahmen von Beispielen. Selbstverständlich können einzelne Merkmale der Ausführungsformen, sofern technisch sinnvoll, frei miteinander kombiniert werden, ohne den Rahmen der vorliegenden Erfindung zu verlassen.

Patentansprüche

1. Verfahren für eine Bereitstellung einer Zeitinformation (210) zu einem Kommunikationsinhalt (220) in einem verteilten System (10) wenigstens einer Anwendung (15), wobei die nachfolgenden Bereitstellungsschritte ausgeführt werden:
- Bereitstellen der Zeitinformation (210) über einen Zeitpunkt einer Erzeugung (101) des Kommunikationsinhalts (220),
- Bereitstellen des Kommunikationsinhalts (220) für eine verkettete Verarbeitung (100) des Kommunikationsinhalts (220) umfassend zumindest eine Zwischenverarbeitung (110) sowie eine Zielverarbeitung (120), welche durch unterschiedliche Verarbeitungsblöcke (16) der wenigstens einen Anwendung (15) ausgeführt werden, wobei das Bereitstellen der Zeitinformation (210) für wenigstens einen der unterschiedlichen Verarbeitungsblöcke (16) erfolgt, um zumindest bei der Zielverarbeitung (120) eine Auswertung (130) der Zeitinformation (210) für eine Bewertung des verarbeiteten Kommunikationsinhalts (220) durchzuführen.

2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass die Bereitstellungsschritte durch eine Middleware (60) ausgeführt werden, welche eine Kommunikation (26) zwischen den Verarbeitungsblöcken (16) bereitstellt, um die Zeitinformation (210) zugeordnet zum Kommunikationsinhalt (220) bei der verketteten Verarbeitung (100) bereitzustellen.

3. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass das Bereitstellen der Zeitinformation (210) dadurch erfolgt, dass die Zeitinformation (210) dem Kommunikationsinhalt (220) zugeordnet wird, wobei die Zeitinformation (210) eine Latenzinformation (141) über den Zeitpunkt der Erzeugung (101), insbesondere bis zu einer Zielverarbeitung (120) des verarbeiteten Kommunikationsinhalts (220), umfasst, wobei die zugeordnete Zeitinformation (210) bei der Zielverarbeitung (120) für die Auswertung (130) abgerufen wird, um bei der Bewertung eine Reaktionszeit der Anwendung (15) und/oder ein Datenalter des Kommunikationsinhalts (220) auszuwerten.

4. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass die Bereitstellungsschritte durch eine Middleware (60) ausgeführt werden, welche dazu ausgeführt ist, eine Echtzeitanforderung (140) zu registrieren, um anhand der Zeitinformation (210) eine Verletzung der Echtzeitanforderung (140) zu detektieren, wobei vorzugsweise die Echtzeitanforderung (140) eine maximale Zeitdauer einer Latenz (141) der Anwendung (15) definiert.

5. Verfahren nach Anspruch 4, **dadurch gekennzeichnet**, dass die Bewertung des verarbeiteten Kommunikationsinhalts (220) den nachfolgenden Schritt umfasst:
- Vergleichen des Zeitpunkts der Erzeugung (101) mit einem aktuellen Zeitpunkt und mit der Echtzeitanforderung (140), um eine Verletzung der Echtzeitanforderung (140) zu detektieren.

6. Verfahren nach Anspruch 5, **dadurch gekennzeichnet**, dass die Middleware (60) zur Registrierung eines Rückrufs (150) ausgebildet ist, um den Rückruf (150) auszuführen, sobald die Verletzung der Echtzeitanforderung (140) detektiert wird.

7. Verfahren nach Anspruch 6, **dadurch gekennzeichnet**, dass das System (10) als ein cyber-physisches System ausgebildet ist, bei welchem in Abhängigkeit von der Zielverarbeitung (120) ein Aktuator (70) aktiviert wird, wobei der Rückruf (150) dazu ausgeführt ist, die Aktivierung des Aktuators (70) zu unterbinden und/oder zu verändern, insbesondere den Aktuator (70) verlangsamt zu betreiben.

8. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass das Bereitstellen der Zeitinformation (210) für mehrere oder sämtliche der unterschiedlichen Verarbeitungsblöcke (16) erfolgt, um bei der zumindest einen Zwischenverarbeitung (110) und bei der Zielverarbeitung (120) die Auswertung (130) der Zeitinformation (210) für die Bewertung des verarbeiteten Kommunikationsinhalts (220) durchzuführen.

9. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet,
dass der wenigstens einen Anwendung (15) und/oder den Verarbeitungsblöcken (16) jeweils eine Echtzeitanforderung oder unterschiedliche Echtzeitanforderungen (140) zugewiesen sind, wobei die Echtzeitanforderung oder die unterschiedlichen

Echtzeitanforderungen wenigstens eine der nachfolgenden Anforderungen umfassen:
- eine maximale Ausführungszeit der Anwendung (15) und/oder der zumindest einen Zwischenverarbeitung (110) und/oder der Zielverarbeitung (120),
- eine maximale Reaktionszeit der Anwendung (15),
- ein maximales Datenalter des verarbeiteten Kommunikationsinhalts (220), wobei das Datenalter sich aus der Zeitinformation (210) über den Zeitpunkt der Erzeugung (101) des Kommunikationsinhalts (220) und einem aktuellen Zeitpunkt während der verketteten Verarbeitung (100) ergibt,
wobei zumindest bei der Zielverarbeitung (120) die bereitgestellte Zeitinformation (210) ausgewertet wird, um eine Einhaltung der wenigstens einen Anforderung zu bewerten.

10. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass die unterschiedlichen Verarbeitungsblöcke (16) jeweils als unterschiedliche Prozesse der wenigstens einen Anwendung (15) oder als unterschiedliche Teile der wenigstens einen Anwendung (15) ausgeführt sind und vorzugsweise durch unterschiedliche, heterogene Rechenknoten (105) eines Netzwerkes (11) ausgeführt werden.

11. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass als die zumindest eine Zwischenverarbeitung (110) mehrere Zwischenverarbeitungen (111,112) vorgesehen sind, wobei das Bereitstellen des Kommunikationsinhalts (220) dadurch erfolgt, dass der erzeugte Kommunikationsinhalt (220) für die verkettete Verarbeitung (100) mittels einer Kommunikation (26) zunächst für die Zwischenverarbeitungen (111,112) und ein Ergebnis (220) der Zwischenverarbeitungen (111,112) als der verarbeitete Kommunikationsinhalt (220) für die Zielverarbeitung (120) bereitgestellt wird, wobei die Kommunikation (26) durch eine Middleware (60), insbesondere eine Messaging-Middleware, zwischen den Verarbeitungsblöcken (16) auf Anwendungsebene bereitgestellt wird, wobei das Bereitstellen der Zeitinformation (210) ebenfalls auf dieser Anwendungsebene erfolgt.

12. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass das System (10) als ein cyber-physisches System ausgebildet ist, in welchem die verkettete Verarbeitung (100) eine funktionale, insbesondere Ursache-Wirkungs-, Kette bildet, die durch mehrere Kommunikationsverbindungen (26) zwischen den Verarbeitungsblöcken (16) bereitgestellt wird, wobei die bereitgestellte Zeitinformation (210) verwendet wird, um eine Verarbeitungsdauer (141), insbesondere Ende-zu-Ende-Latenz (141), der gesamten Kette zu ermitteln.

13. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet,
dass die Erzeugung (101) des Kommunikationsinhalts (220) zumindest den nachfolgenden Schritt umfasst:
- Durchführen einer Erfassung einer Umgebung eines Fahrzeuges (50) durch einen Sensor (20), um die erfasste Umgebung als den erzeugten Kommunikationsinhalt (220) bereitzustellen,
und dass die zumindest eine Zwischenverarbeitung (110) und die Zielverarbeitung (120) zumindest die nachfolgenden Schritte umfasst:
- Detektieren von mindestens einem Objekt in der erfassten Umgebung des Fahrzeuges (50) anhand des erzeugten Kommunikationsinhalts (220),
- Berechnen eines Pfades zur Fortbewegung des Fahrzeuges (50) anhand des mindestens einen detektierten Objekts, um den berechneten Pfad als den verarbeiteten Kommunikationsinhalt (220) bereitzustellen,
- Aktivieren eines Aktuators (70), um das Fahrzeug entlang des berechneten Pfades fortzubewegen,
wobei das Aktivieren des Aktuators (70) in Abhängigkeit von der Bewertung des verarbeiteten Kommunikationsinhalts (220) erfolgt.

14. Computerprogramm (200), umfassend Befehle, die bei der Ausführung des Computerprogramms (200) durch einen Computer diesen veranlassen, das Verfahren nach einem der vorhergehenden Ansprüche auszuführen.

15. Verteiltes System (10), welches eingerichtet ist, das Verfahren nach einem der Ansprüche 1 bis 13 auszuführen.

Es folgen 3 Seiten Zeichnungen

Anhängende Zeichnungen

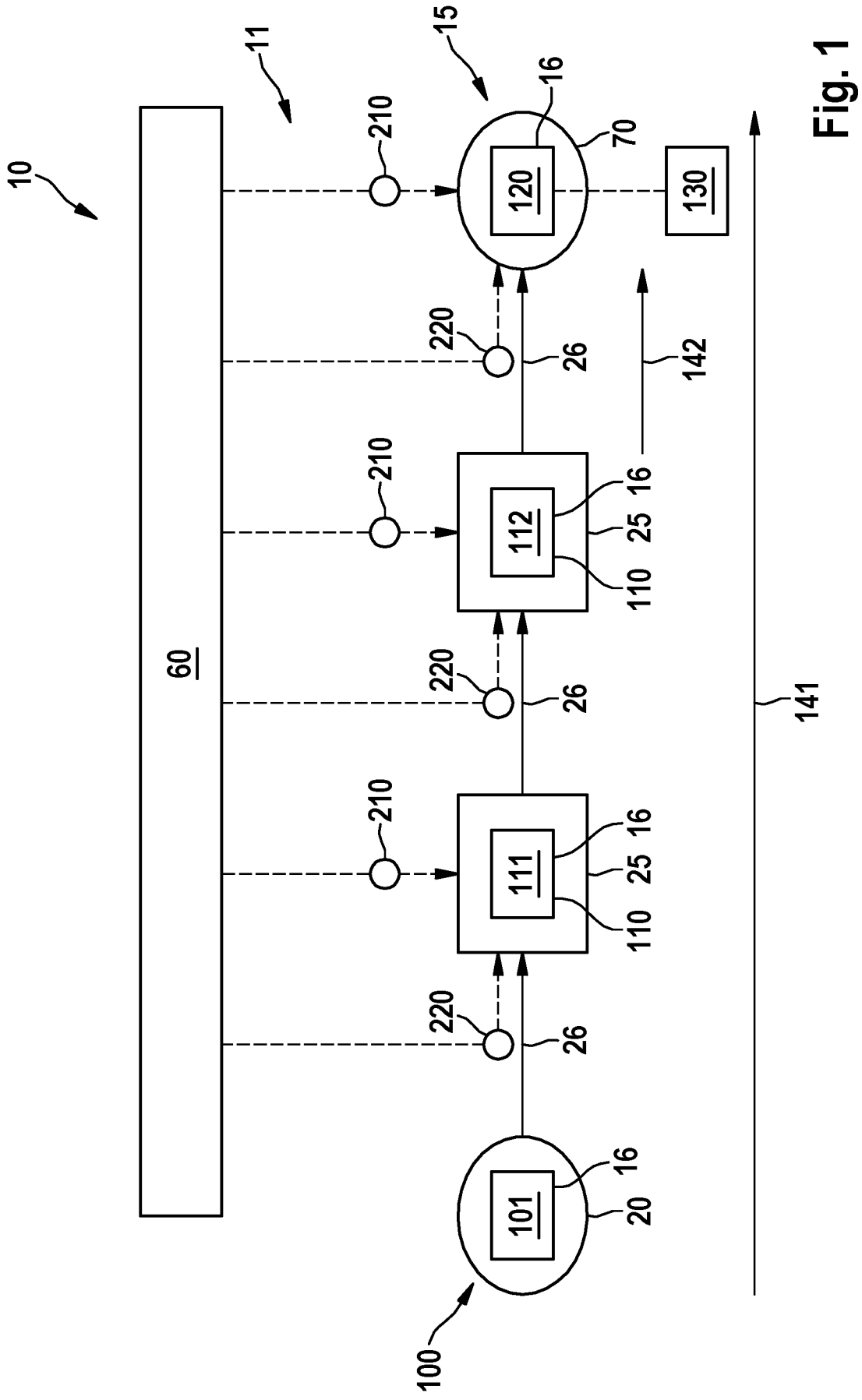


Fig. 1

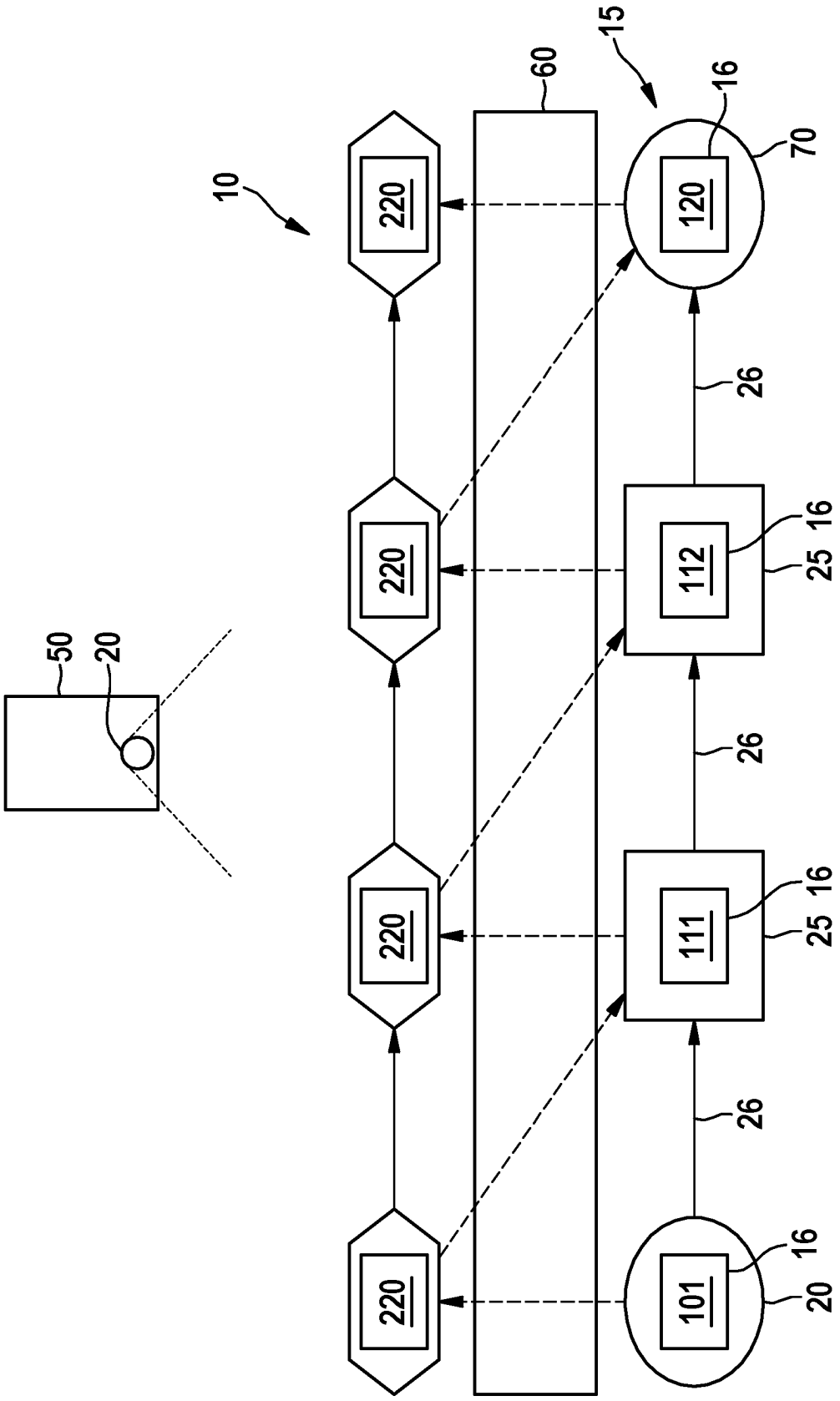
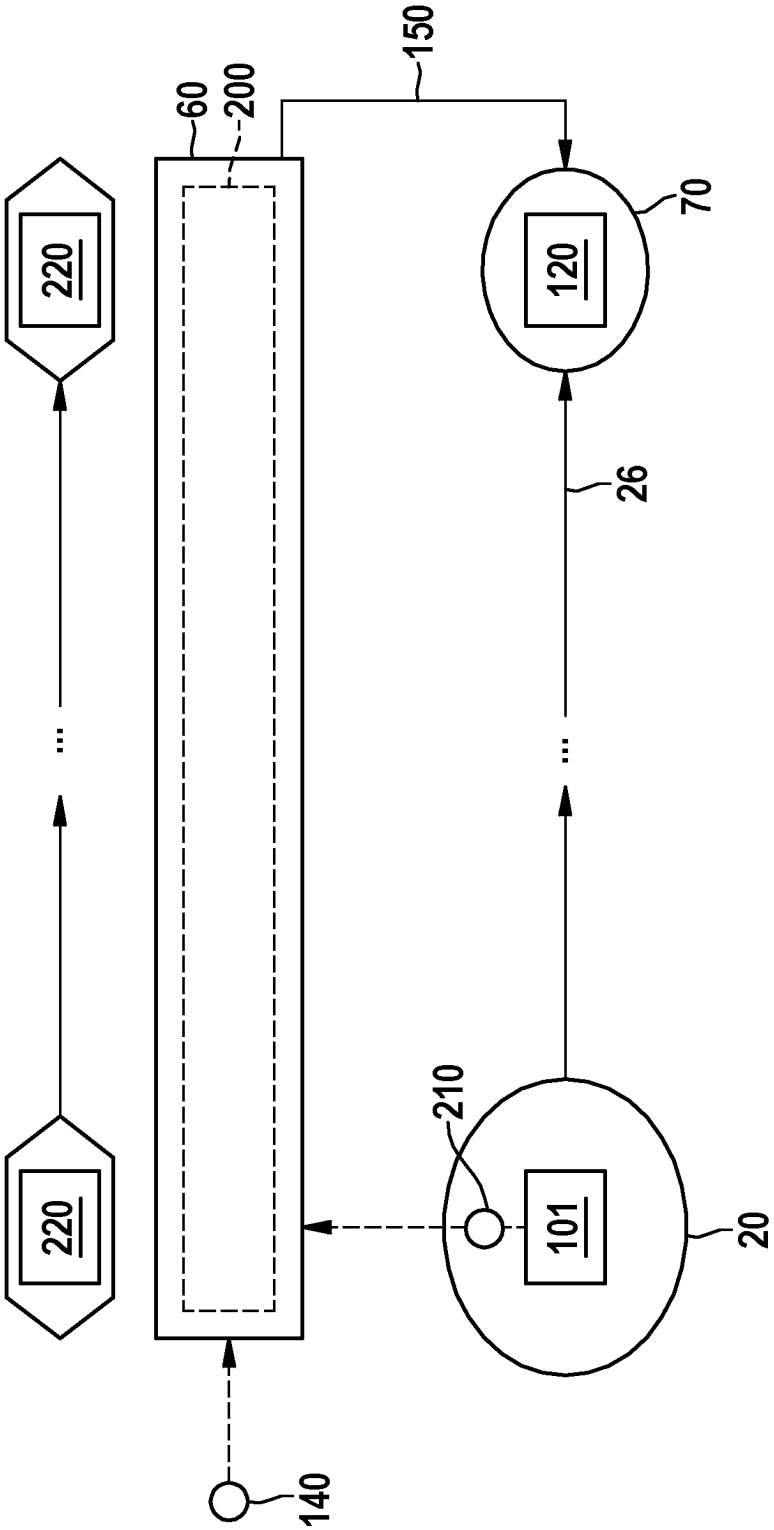


Fig. 2

Fig. 3



 Das Dokument wurde durch die Firma Luminess hergestellt.