



(10) DE 10 2016 119 178 A1 2017.04.13

(12)

Offenlegungsschrift

(21) Aktenzeichen: 10 2016 119 178.7

(22) Anmeldetag: 10.10.2016

(43) Offenlegungstag: 13.04.2017

(51) Int Cl.: G06F 17/30 (2006.01)

G07C 3/00 (2006.01)

G06F 15/163 (2006.01)

(30) Unionspriorität:

62/239,620

09.10.2015 US

15/274,244

23.09.2016 US

(72) Erfinder:

Nixon, Mark John, Round Rock, Tex., US;
Enver, Alper Turhan, Austin, Tex., US; Bell, Noel
Howard, Austin, Tex., US; Kidd, Joshua Brian,
Safety Harbor, Fla., US; Muston, Paul Richard,
Narborough, Leicester, GB

(71) Anmelder:

Fisher-Rosemount Systems, Inc., Round Rock,
Tex., US

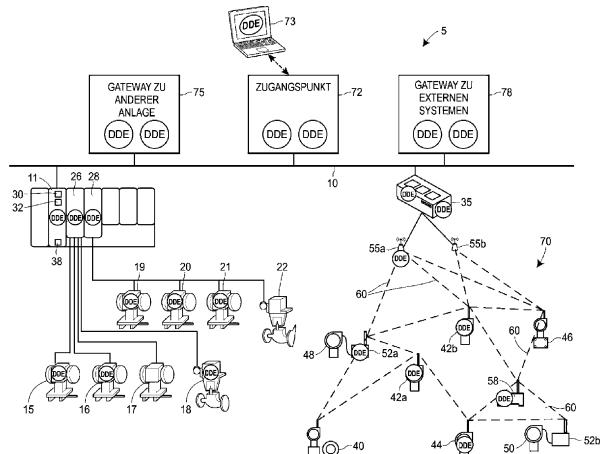
(74) Vertreter:

Meissner Bolte Patentanwälte Rechtsanwälte
Partnerschaft mbB, 80538 München, DE

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Bezeichnung: Quellenunabhängige Abfragen in einem verteilten Industriesystem

(57) Zusammenfassung: Bereitgestellt werden Techniken für das Abfragen und Bereitstellen von Prozessanlagendaten unter Verwendung einer quellenunabhängigen standardisierten Abfrage. Eine Abfragevorrichtung generiert eine standardisierte Abfrage, um Daten aus einer oder mehreren Datenquellen, beispielsweise aus relationalen oder nicht-relationalen Datenbanken, abzurufen. Die Abfrage nutzt ein standardisiertes Format, das nicht von der Datenquelle abhängig ist, und die Abfrage kann als JSON-Datei generiert werden. Die standardisierte Abfrage ist möglicherweise nicht direkt nutzbar für die Datenquellen. Stattdessen generiert eine Datenvorrichtung auf den Empfang einer standardisierten Abfrage hin eine oder mehrere quellspezifische Abfragen. Die quellspezifischen Abfragen nutzen eine Syntax, die für die einzelnen Datenquellen nativ ist, um Daten abzurufen. In manchen Fällen müssen die empfangenen Daten weiterverarbeitet werden, um sie an unterschiedliche Abtastzeiten oder Abtastraten anzupassen, beispielsweise durch Interpolation. Die resultierenden Daten aus allen Datenquellen können in einem Datenrahmen kombiniert werden, bevor sie an die Abfragevorrichtung zurückgeschickt werden.



Beschreibung**QUERVERWEIS AUF
VERWANDTE ANMELDUNGEN**

[0001] Diese Anmeldung beansprucht den Vorteil der am 9. Oktober 2015 eingereichten vorläufigen US-Anmeldung Nr. 32/239, 620 mit dem Titel DISTRIBUTED INDUSTRIAL PERFORMANCE MONITORING AND ANALYTICS, deren Gesamtheit hiermit ausdrücklich durch Bezugnahme mit aufgenommen wird. Die vorliegende Anmeldung betrifft: (i) die am 4. März 2013 eingereichte US-Anmeldung Nr. 13/784, 041 mit dem Titel „BIG DATA IN PROCESS CONTROL SYSTEMS“, (ii) die am 6. Februar 2014 eingereichte US-Anmeldung Nr. 14/174, 413 mit dem Titel „COLLECTING AND DELIVERING DATA TO A BIG DATA MACHINE IN A PROCESS CONTROL SYSTEM“, (iii) die am 11. August 2014 eingereichte US-Anmeldung 14/456, 763 mit dem Titel „SECURING DEVICES TO PROCESS CONTROL SYSTEMS“, (iv) die am 17. März 2014 eingereichte US-Anmeldung Nr. 14/216, 823 mit dem Titel „DATA MODELING STUDIO“, (v) die am 31. Januar 2014 eingereichte US-Anmeldung Nr. 14/169, 965 mit dem Titel „MANAGING BIG DATA IN PROCESS CONTROL SYSTEMS“, (vi) die am 14. März 2014 eingereichte US-Anmeldung Nr. 14/212, 411 mit dem Titel „DETERMINING ASSOCIATIONS AND ALIGNMENTS OF PROCESS ELEMENTS AND MEASUREMENTS IN A PROCESS“, (vii) die am 14. März 2014 eingereichte US-Anmeldung Nr. 14/212, 493 mit dem Titel „DISTRIBUTED BIG DATA IN A PROCESS CONTROL SYSTEM“, (viii) die am 6. Oktober 2014 eingereichte US-Anmeldung Nr. 14/506, 863 mit dem Titel „STREAMING DATA FOR ANALYTICS IN PROCESS CONTROL SYSTEMS“, (ix) die am 6. Oktober 2014 eingereichte US-Anmeldung Nr. 14/507, 188 mit dem Titel „REGIONAL BIG DATA IN PROCESS CONTROL SYSTEMS“, (x) die am 6. Oktober 2014 eingereichte US-Anmeldung Nr. 62/060, 408 mit dem Titel „DATA PIPELINE FOR PROCESS CONTROL SYSTEM ANALYTICS“ und (xi) die am 6. Oktober 2014 eingereichte US-Anmeldung Nr. 14/507, 252 mit dem Titel „AUTOMATIC SIGNAL PROCESSING-BASED LEARNING IN A PROCESS PLANT“, deren gesamte Offenbarungen hiermit ausdrücklich durch Bezugnahme mit aufgenommen werden.

TECHNISCHES GEBIET

[0002] Die vorliegende Offenbarung betrifft allgemein Prozessanlagen und Prozessleitsysteme und insbesondere eine Echtzeit-Leistungsüberwachung und Analytik von Echtzeitdaten, die von Prozessanlagen und Prozessleitsystemen generiert werden.

HINTERGRUND

[0003] Verteilte Prozessleitsysteme, wie sie in chemischen, Petroleum-, industriellen oder anderen Prozessanlagen zur Herstellung, Veredelung, Umwandlung, Erzeugung oder Produktion physischer Materialien oder Produkte eingesetzt werden, umfassen typischerweise eine oder mehrere Prozesssteuereinrichtungen, die über analoge, digitale oder analog/digitale Datenbusse oder über eine drahtlose Kommunikationsverbindung oder ein Netzwerk kommunikativ mit einem oder mehreren Feldgeräten verbunden sind. Die Feldgeräte, welche beispielsweise Ventile, Stellungsregler, Schalter und Sender sein können (z. B. Temperatur-, Druck-, Niveau- und Durchflussmessfühler), sind innerhalb der Prozessumgebung angeordnet und führen generell physische oder Prozesssteuerfunktionen durch, wie das Öffnen oder Schließen von Ventilen, die Messung des Prozesses und/oder von Umgebungsparametern, wie Temperatur oder Druck usw., zur Steuerung eines oder mehrerer Prozesse, die innerhalb der Prozessanlage oder des Systems ausgeführt werden. Intelligente Feldgeräte, wie beispielsweise die Feldgeräte, die dem bekannten Feldbusprotokoll entsprechen, können auch Steuerberechnungen, Alarmfunktionen und andere Steuerfunktionen, die üblicherweise innerhalb der Steuerung implementiert sind, ausführen. Die Prozesssteuereinrichtungen, die normalerweise ebenfalls innerhalb der Anlage angeordnet sind, empfangen Signale, die indikativ für die Prozessmessungen sind, die von den Feldgeräten durchgeführt werden, und/oder die andere Informationen im Zusammenhang mit den Feldgeräten sind, und führen eine Steuereinrichtungsanwendung aus, die beispielsweise unterschiedliche Steuermodule betreibt, Prozessleitentscheidungen trifft, Steuersignale auf Grundlage der erhaltenen Informationen erzeugt und mit den Steuermodulen oder Blöcken, die in den Feldgeräten ausgeführt werden, koordiniert, wie beispielsweise HART®, WirelessHART® znd FOUNDATION® Fieldbus Feldgeräte. Die Steuermodule in der Steuereinrichtung senden die Steuersignale über die Kommunikationsleitungen oder -verbindungen an die Feldgeräte, um dadurch den Betrieb von mindestens einem Teil der Prozessanlage zu steuern, beispielsweise um mindestens einen Teil eines oder mehrerer industrieller Prozesse, die innerhalb der Anlage ablaufen oder ausgeführt werden, zu steuern. Die Steuereinrichtungen und die Feldgeräte steuern beispielsweise mindestens einen Teil eines Prozesses, der von dem Prozessleitsystem der Prozessanlage gesteuert wird.

[0004] Informationen der Feldgeräte und der Steuereinrichtung werden üblicherweise über eine Datenautobahn oder ein Kommunikationsnetzwerk an ein oder mehrere andere Hardwaregeräte bereitgestellt, wie beispielsweise Bedienerarbeitsplätze, persönliche Computer oder Rechner, Datenhistoriker, Be-

richtgeneratoren, zentrale Datenbanken oder andere zentrale, administrative Rechner, die sich typischerweise in Kontrollräumen oder an anderen Standorten befinden, die von der rauen Anlagenumgebung entfernt liegen. Jedes dieser Hardwaregeräte ist typischerweise für die gesamte Prozessanlage oder für einen Teil der Prozessanlage zentralisiert. Diese Hardwaregeräte führen Anwendungen aus, die es einem Bediener beispielsweise ermöglichen, Funktionen zur Steuerung eines Prozesses und/oder für den Betrieb der Prozessanlage auszuführen, wie beispielsweise die Veränderung von Einstellungen der Prozessleitroutine, die Modifikation des Betriebs der Steuermodule innerhalb der Steuereinrichtung oder der Feldgeräte, die Beobachtung des aktuellen Prozessstatus, das Einsehen von Alarmen, die von Feldgeräten und Steuereinrichtungen erzeugt werden, die Simulation des Prozessbetriebs zum Zweck der Personalschulung oder zum Testen der Prozesssteuerungssoftware, die Erhaltung und Aktualisierung der Konfigurationsdatenbank usw. Die von den Hardwaregeräten, Steuereinrichtungen und Feldgeräten genutzte Datenautobahn kann einen verkabelten Kommunikationsweg, einen kabellosen Kommunikationsweg oder eine Kombination aus verkabelten und kabellosen Kommunikationswegen umfassen.

[0005] Das DeltaV™ Steuersystem zum Beispiel, das von Emerson Process Management vertrieben wird, enthält mehrere Anwendungen, die darin gespeichert sind und von unterschiedlichen Geräten, die sich an verschiedenen Orten innerhalb einer Prozessanlage angeordnet sind, ausgeführt werden. Eine Konfigurationsanwendung, die sich in einem oder mehreren Arbeitsplätzen oder Rechnern befindet, ermöglicht es Benutzern Prozessleitmodule zur erzeugen oder zu verändern und diese Prozessleitmodule über eine Datenautobahn auf spezielle, verteilte Steuereinrichtungen herunterzuladen. Diese Leitmodule bestehen typischerweise aus miteinander verbundenen Funktionsblöcken, die Objekte in einem objektorientiertem Programmierprotokoll sind, und innerhalb des Leitschemas auf Grundlage von Eingaben Funktionen ausführen und anderen Funktionsblöcken innerhalb des Leitschemas Ausgaben bereitstellen. Die Konfigurationsanwendung kann es einem Konfigurationsentwickler auch ermöglichen, Bedienerschnittstellen zu erzeugen und zu verändern, die von einer Beobachtungsanwendung zur Darstellung von Daten für einen Bediener verwendet werden und es dem Bediener ermöglichen, Einstellungen, beispielsweise Sollwerte, innerhalb der Prozessleitroutinen zu ändern. Jede spezielle Steuereinrichtung und in einigen Fällen ein oder mehrere Feldgeräte, speichert eine entsprechende Steuereinrichtungsanwendung und führt diese aus, die die zugewiesenen Leitmodule betreibt, und zur Implementierung der tatsächlichen Prozesssteuerfunktionalität darauf heruntergeladen wurde. Die Beobachtungsanwendungen, die an einem oder mehreren Bediener-Arbeitsplät-

zen (oder dezentral auf einem oder mehreren Rechnern, die kommunikativ mit den Bediener-Arbeitsplätzen und der Datenautobahn verbunden sind) ausgeführt werden können, empfangen über die Datenautobahn Daten von der Steuereinrichtungsanwendung und zeigen diese Daten den Entwicklern von Prozessleitsystemen, Bedienern oder Benutzern über Benutzerschnittstellen an. Sie können auch mehrere verschiedene Ansichten ermöglichen, wie eine Bedieneransicht, eine Ingenieuransicht, eine Technikeransicht usw. Eine Datenhistorikanwendung wird typischerweise in einem Datenhistorikgerät gespeichert und von diesem ausgeführt, das einige oder alle Daten, die über die Datenautobahn bereitgestellt werden, sammelt und speichert, während eine Konfigurationsdatenbankanwendung in einem weiteren Computer ausgeführt werden kann, der zur Speicherung der aktuellen Konfiguration der Prozessleitroutine und der damit verbundenen Daten mit der Datenautobahn verbunden ist. Alternativ hierzu kann sich die Konfigurationsdatenbank auch in dem gleichen Arbeitsplatz wie die Konfigurationsanwendung befinden.

[0006] In einer Prozessanlage oder einem Prozessleitsystem wird, wenn es einen Beleg für eine abnorme Bedingung gibt oder ein Fehler auftritt (beispielsweise wenn ein Alarm erzeugt wird oder wenn festgestellt wird, dass eine Prozessmessung oder ein Betätigungsorgan eine zu starke Abweichung aufweist), ein Bediener, ein Instrumententechniker oder ein Verfahrenstechniker typischerweise ein Analysewerkzeug in Kombination mit seiner Kenntnis des Prozesses, der von dem System gesteuert wird, und des Fließwegs durch das System zur Anwendung bringen, um zu versuchen, vorgeschaltete Messungen und Prozessvariablen festzustellen, die möglicherweise zur Erzeugung des Belegs der abnormen Bedingung oder des Fehlers beigetragen haben. Ein Bediener kann beispielsweise ein historisches Datenprotokoll, das im Laufe der Zeit aus der Ausgabe eines Prozessleitsystems (beispielsweise eines Feldgeräts, einer Steuereinrichtung usw.) in das DeltaV™ Batch-Analytikprodukt oder das Werkzeug für die kontinuierliche Datenanalyse erstellt wurde, zuführen, um zu versuchen, die Beiträge verschiedener Prozessvariablen und/oder -messungen zu einer abnormen oder Fehlerbedingung festzustellen. Typischerweise entscheidet ein Benutzer, welche historischen Datenprotokolle und/oder Zeitseriendaten dem Analysewerkzeug zugeführt werden, und identifiziert aufgrund seiner Kenntnis des Prozesses mögliche vorgeschaltete Faktoren (beispielsweise Messungen, Prozessvariablen usw.). Anschließend verwenden diese Datenanalytikwerkzeuge die Hauptkomponentenanalyse (Principal Component Analysis; PCA) oder eine andere Analysetechnik, um zu ermitteln, welche der möglichen vorgeschalteten Faktoren sich auf die erwarteten, nachgeschalteten Qualitätsparameter auswirken. Die Genauigkeit und Ef-

fektivität der von dem Analytikwerkzeug bereitgestellten Ausgabe beruht also auf der Kenntnis des Benutzers oder ist dadurch beschränkt und ermöglicht somit möglicherweise keine vollständigen oder richtigen Einsichten in den Ursprung der abnormen Bedingung oder des Fehlers.

[0007] Außerdem werden solche Analytiken typischerweise außerhalb des Prozesses durchgeführt und der Prozess kann sich somit verändern oder bewegen, während die Analyse durchgeführt wird. Eine typische Prozessanlage führt beispielsweise einen oder zwei Zyklen einer bestimmten Analyse pro Tag durch (beispielsweise einen Datensammelungs- und einen Analysezyklus) und die Ergebnisse werden erst einige Zeit nachdem die Analytik durchgeführt wurde analysiert und Vorgaben entwickelt und in der Anlage implementiert. Es kann daher nicht nur die Genauigkeit der Analytikergebnisse suspekt sein, sondern die Vorgaben, die daraus entwickelt werden, können ebenfalls nicht optimal sein oder für den derzeit ausgeführten Prozess nicht mehr gelten.

[0008] Weiterhin wird die Architektur der derzeit bekannten Prozessleitanlagen und Prozessleitsysteme stark von dem eingeschränkten Speicher der Steuereinrichtung und des Geräts, der Kommunikationsbandbreite und der Leistungsfähigkeit der Steuereinrichtung und des Geräteprozessors beeinflusst. In derzeit bekannten Architekturen für Prozessleitsysteme ist beispielsweise die Verwendung des dynamischen und statischen, nichtvolatilen Speichers in der Steuereinrichtung üblicherweise minimiert oder sie wird zumindest sorgfältig verwaltet. Infolgedessen muss ein Benutzer typischerweise während der Systemkonfiguration (z. B. a priori) wählen, welche Daten in der Steuereinrichtung archiviert oder gespeichert werden sollen, wie häufig sie gespeichert werden und ob sie komprimiert werden oder nicht und die Steuereinrichtung ist dann mit diesen eingeschränkten Datensatzregeln entsprechend konfiguriert. Demnach werden Daten, die für die Fehlerbehebung und Prozessanalyse nützlich sein könnten, oft nicht archiviert, und wenn sie gesammelt werden, können die nützlichen Informationen aufgrund der Datenkompression verlorengehen.

[0009] Noch weiter nimmt das Volumen der Datensets von industriellen oder Prozessleitanlagen stetig bis zu dem Punkt zu, an dem die bestehenden Datenverarbeitungsanalytikanwendungen nicht mehr ausreichen. Typischerweise versuchen bekannte Analytiktechniken nur einen Wert aus Daten zu extrahieren, sie gehen aber nicht auf besondere Volumen der Datensätze, aus denen der Wert extrahiert wird, ein, und sie können insbesondere nicht nahtlos mit sehr großen Datensätzen (beispielsweise alle Prozessdaten, die in einer Anlage generiert werden) arbeiten. Weiterhin können bekannte Analytiktechniken keine Streaming oder Stream-Daten bearbeiten.

[0010] Die Einschränkungen der derzeit bekannten Prozessanlagenüberwachungs- und Analytik- und Prozessleitsysteme, die vorstehend besprochen wurden, und andere Einschränkungen können sich unerwünscht im Betrieb und bei der Optimierung von Prozessanlagen oder Prozessleitsystemen manifestieren, beispielsweise im Anlagebetrieb, bei der Fehlerbehebung und/oder der Vorhersagemodellierung. Generell sind Echtzeitanalytiken unter Verwendung von aktuellen, industriellen Prozessleistungsdaten in Echtzeit mit den bekannten Überwachungs- und Analytikwerkzeugen nicht möglich.

KURZDARSTELLUNG

[0011] Hierin werden Techniken, Systeme, Vorrichtungen, Komponenten und Verfahren für verteilte, industrielle Prozessleistungsüberwachung und/oder Analytiken offenbart. Diese Techniken, Systeme, Vorrichtungen, Komponenten und Verfahren können für industrielle Prozessleitsysteme, Umgebungen und/oder Anlagen eingesetzt werden, die hierin austauschbar als „Automation“, „industrielle Steuerung“, „Prozessleitung/-steuerung“ oder „Prozess“ Systeme, Umgebungen und/oder Anlagen bezeichnet werden. Typischerweise ermöglichen diese Systeme und Anlagen auf verteilte Weise die Steuerung von einem oder mehreren Prozessen, die zur Herstellung, Verfeinerung, Umwandlung, Erzeugung oder Produktion physischer Materialien oder Produkte ablaufen. Diese Techniken, Systeme, Vorrichtungen, Komponenten und Verfahren umfassen generell die Einbettung von Datenüberwachungs- und/oder Datenanalytikmaschinen (die hierin austauschbar auch als „verteilte Datenmaschine“, „Datenmaschine“ oder „DDE“ (Distributed Data Engine) bezeichnet werden) auf verteilte Weise innerhalb von Geräten, die bei der Steuerung eines industriellen Verfahrens zusammenarbeiten. Eine verteilte Datenmaschine kann beispielsweise zu einer Vorrichtung hergestellt werden, die mit anderen Vorrichtungen arbeitet, um einen Prozess zu steuern, der in einer Prozessanlage oder in einem Prozessleitsystem ausgeführt wird (beispielsweise Prozessleitvorrichtungen, wie Feldgeräte, Steuereinrichtungen, I/O Karten usw.) und/oder eine verteilte Datenmaschine kann lokal oder direkt mit solch einer Vorrichtung gekoppelt sein. Es können zusätzliche Datenmaschinen in anderen Vorrichtungen, die in der Prozessanlage enthalten sind, wie beispielsweise in Kommunikationsknoten, Arbeitsplätzen oder anderen Bedienerchnittstellen, Servern und ähnlichem, eingebettet oder hergestellt sein. In manchen Konfigurationen sind Datenmaschinen mit mehreren Kommunikationsverbindungen innerhalb der Prozessanlage oder anderweitig mit der Prozessanlage verbunden, so dass sie eine Ansicht oder ein Fenster zu Echtzeit-Daten bieten, die während der Durchführung oder Steuerung des Prozesses innerhalb der Anlage übertragen werden.

[0012] Daten werden zwischen mehreren eingebetteten Datenmaschinen per Stream übertragen, beispielsweise durch Verwendung eines oder mehrerer Datenkommunikationskanäle und Netzwerke, die typischerweise außerhalb der herkömmlichen Kommunikationssysteme existieren, die man in verteilten Steuersystemen (Distributed Control Systems; DCS), programmierbaren Logiksystemen (Programmable Logic Systems; PLS) und Prozessleit- Sicherheits- und Gesundheitsüberwachungssystemen (Process Control Safety and Health Monitoring Systems; SHM) findet. Um eine Verwechslung mit solchen herkömmlich bekannten Prozessleitkommunikations- systemnetzwerken und -verbindungen zu vermeiden, werden die Kommunikationskanäle und Netzwerke, die zur Stream-Kommunikation zwischen Datenmaschinen genutzt werden, hierin austauschbar als „Dateianalytikkommunikationskanäle“, „Datenanalytikkanäle“, „Datenanalytikkommunikationsnetzwerke“ oder „Datenanalytiknetzwerke“ bezeichnet. Die Stream-Daten können Echtzeitdaten umfassen, die von der Datenmaschine eingesehen oder beobachtet werden. Wenn eine verteilte Datenmaschine beispielsweise mit einer herkömmlichen Prozesskontrollsystemkommunikationsverbindung verbunden ist (und somit die Daten, die darüber übertragen werden, einsieht), kann die Datenmaschine eine Kopie der Daten, die über die herkömmliche Kommunikationsverbindung übertragen werden, über das Datenanalytiknetzwerk per Stream an eine oder mehrere andere Datenmaschinen übertragen. In einem anderen Beispiel könnten, wenn eine verteilte Datenmaschine hergestellt wird oder in eine Prozessleitvorrichtung eingebettet ist, die Analytikdaten, die von der Datenmaschine, die das Datenanalytiknetzwerk nutzt, per Stream übertragen werden, Kopien der Daten umfassen, die von der Vorrichtung empfangen, erzeugt oder anderweitig verarbeitet werden. Weiterhin oder alternativ hierzu können die Stream-Daten Daten umfassen, die einer oder mehreren Analytiken entsprechen, die lokal an der Vorrichtung durchgeführt wurden, wie beispielsweise Analytikresultate, Vorgaben und dergleichen. Diese Architektur ermöglicht es, dass ein Analytikservice lokal gebunden ist und in der Nähe oder sogar an der Datenquelle bereitgestellt werden kann, während gleichzeitig Analytiken in größerem Umfang bereitgestellt werden, wodurch zeitnahe Resultate und Optimierungen ermöglicht werden, während die Bandbreitennutzung und Verarbeitungszyklen im gesamten System minimiert werden, wie nachfolgend noch im Detail erklärt wird.

[0013] In einem Aspekt umfasst ein verteiltes System zur Überwachung und Analyse eines industriellen Prozesses mehrere verteilte Datenmaschinen (Distributed Data Engines; DDEs), die innerhalb einer Prozessanlage, die zur Steuerung eines Prozesses betrieben wird, eingebettet ist. Jede der DDEs ist mit einer oder mehreren jeweiligen Datenquellen innerhalb der Prozessanlage gekoppelt, die jeweils als Re-

sultat der Prozesssteuerung Daten erzeugen. Weiterhin speichert jede der DDEs die von der einen oder den mehreren jeweiligen Datenquellen, die mit jeder DDE gekoppelt sind, erzeugten Daten. Das System umfasst auch ein Datenanalytiknetzwerk, das das Streaming von Analytikdaten zwischen den mehreren DDEs unterstützt, und das die Übertragung der Anfragen nach Daten, die auf den mehreren DDEs gespeichert sind, unterstützt.

[0014] In einem anderen Aspekt umfasst ein Verfahren die Mitteilung über das Vorhandensein eines Clusters in einem Datenanalytiknetzwerk einer Prozessleitanlage, die zur Steuerung eines Prozesses betrieben wird. Dieses Verfahren umfasst auch die Registrierung einer Datenquelle, die auf die Mitteilung reagiert und kontinuierlich Daten erzeugt, die aus der Steuerung des Prozesses durch die Anlage resultieren. Weiterhin umfasst das Verfahren den Empfang kontinuierlicher Daten, die von der Datenquelle erzeugt werden, und das Streaming von mindestens manchen der von der Datenquelle kontinuierlich erzeugten Daten über das Datenanalytiknetzwerk an einen Datenkonsumenten. Einer oder mehrere Teile des Verfahrens können beispielsweise von dem Cluster durchgeführt werden.

[0015] In noch einem anderen Aspekt umfasst ein Verfahren die Mitteilung über das Vorhandensein eines zentralen Clusters in einem Datenanalytiknetzwerk einer Prozessleitanlage, die zur Steuerung eines Prozesses betrieben wird. Dieses Verfahren umfasst auch die Registrierung eines lokalen Clusters, das auf die Mitteilung reagiert und konfiguriert ist, kontinuierliche Stream-Daten, die als ein Resultat der Steuerung des Prozesses erzeugt werden, zu speichern. Das lokale Cluster ist in die Prozessanlage eingebettet und kommunikativ mit einer oder mehreren Datenquellen gekoppelt, die mindestens einen Teil der kontinuierlichen Daten während des Betriebs zur Steuerung eines Prozesses generieren. Das Verfahren kann weiterhin den Empfang von mindestens einem Teil der kontinuierlichen Stream-Daten von dem lokalen Cluster umfassen, die Durchführung von einer oder mehreren Datenanalytfunktionen für die kontinuierlichen Stream-Daten, die von dem lokalen Cluster empfangen werden, und das Streaming einer Ausgabe der einen oder oder mehreren Datenanalytfunktionen über das Datenanalytiknetzwerk an einen Datenkonsumenten. Einer oder mehrere Teile des Verfahrens können beispielsweise von dem zentralen Cluster durchgeführt werden.

[0016] In einem anderen Aspekt umfasst das System ein Datenanalytikmodul, das an einen kontinuierlichen Datenstrom gebunden ist. Der kontinuierliche Datenstrom verfügt über Inhalte, die in Echtzeit von einer Datenquelle erzeugt werden, die in der Prozessanlage enthalten ist, die zur Steuerung eines Prozesses betrieben wird, und die Inhalte des kon-

tinuierlichen Datenstroms werden als Resultat des Betriebs zur Steuerung des Prozesses erzeugt. Das System umfasst weiterhin eine Benutzerschnittstelle, die die kontinuierliche Ausgabe darstellt, die in Echtzeit als Resultat des Datenanalytikmoduls, das in Echtzeit für den kontinuierlichen Datenstrom betrieben wird, erzeugt wird. Die kontinuierliche Ausgabe, die auf der Benutzerschnittstelle angezeigt wird, umfasst eine kontinuierliche Aktualisierung von einem oder mehreren vorhergesagten Werten.

[0017] In noch einem anderen Aspekt umfasst ein Leistungsüberwachungs- und Analytiksystem für die Steuerung eines industriellen Prozesses eine Plattform, die über einen Satz an Benutzersteuerungen und eine Oberfläche verfügt, die es einem Benutzer ermöglichen, ein Datendiagramm zu erzeugen, das repräsentativ für ein Datenmodell ist. Die Plattform ermöglicht es dem Benutzer weiterhin, das Datendiagramm zur Ausführung eines Dateneingabesatzes zu evaluieren oder dieses zusammenzustellen, wodurch Ausgabedaten generiert werden. Der Dateneingabesatz umfasst Zeitseriendaten (und optional andere Daten), die aus einer Online-Prozessanlage, die einen Prozess steuern, resultieren. Der Begriff „Zeitseriendaten“, wie er hierin verwendet wird, bezieht sich auf eine Sequenz von Datenpunkten, Werten oder Sätzen, die über ein Zeitintervall generiert werden, typischerweise von einer oder mehreren Datenquellen.

[0018] Das Datendiagramm umfasst einen Satz von Datenblöcken, die über einen Satz von Drähten, über die Daten zwischen den Datenblöcken übertragen werden, miteinander verbunden sind. Jeder Datenblock des Sets von Datenblöcken korrespondiert mit einer entsprechenden Datenoperation, einschließlich Null, einem oder mehreren Eingangskonnektoren, und umfasst Null, einen oder mehrere Ausgangskonnektoren. Die jeweiligen Eingangsdaten werden an jedem Datenblock über den oder die Eingangskonnektor(en) empfangen und jeder jeweilige Datenblock liefert entsprechende Ausgangsdaten, die daraus resultieren, dass jeder Datenblock seine jeweilige Datenoperation für die jeweiligen Eingangsdaten ausführt, über den oder die Ausgangskonnektor(en). Der Eingangskonnektor ist weiterhin in einem ersten Datenblock eines Satzes von Datenblöcken enthalten, wobei die jeweilige Datenoperation des mindestens einen Datenblock, der in dem Satz von Datenblöcken enthalten ist, eine Datenanalytfunktion umfasst, und unterschiedliche Teile des Datendiagramms asynchron sind und separat komplizierbar und ausführbar sind. In manchen Ausführungsformen, beispielsweise wenn keine Eingangskonnektoren und keine Ausgangskonnektoren konfiguriert sind, können Eingangs- und Ausgangsdaten intern von einem Datenblock bearbeitet werden.

[0019] In einem Aspekt umfasst ein Verfahren zur Durchführung von Echtzeitanalytiken in einer Prozesssteuerungsumgebung die Erzeugung eines ersten Diagramms auf einer grafischen Benutzerschnittstelle. Das erste Diagramm kann erste Programmierbefehle repräsentieren, die betreibbar sind, um einen Prozessor zu veranlassen, Daten, die zuvor von der Prozesssteuerungsumgebung erzeugt und gespeichert wurden, zu manipulieren. Das Verfahren umfasst auch das Kompilieren der ersten Programmierbefehle, die von dem ersten Diagramm repräsentiert werden, die Ausführung der kompilierten ersten Programmierbefehle zur Erzeugung einer resultierenden ersten Ausgabe, und die Auswertung der resultierenden ersten Ausgabe zur Ermittlung eines Vorhersagewerts eines Modells, das zur Erzeugung einer resultierenden ersten Ausgabe verwendet wird. Das Verfahren umfasst weiterhin die automatische Erzeugung eines zweiten Diagramms aus dem ersten Diagramm. Das zweite Diagramm kann zweite Programmierbefehle repräsentieren, die betreibbar sind, um einen Prozessor zu veranlassen, sich an eine Live-Datenquelle zu binden und Daten auszuwerten, die von der Live-Datenquelle empfangen werden, wobei zumindest das Modell verwendet wird, das zur Erzeugung der ersten resultierenden Ausgabe verwendet wurde. Das Verfahren umfasst weiterhin die Komplilation der zweiten Programmierbefehle, die von dem zweiten Diagramm repräsentiert werden, und die Ausführung der kompilierten zweiten Programmierbefehle zur Vorhersage eines Aspekts des Betriebs der Prozessleitumgebung.

[0020] In einem anderen Aspekt umfasst ein System zur Durchführung von Echtzeit-Analytiken in einer Prozessleitumgebung eine Vielzahl von Prozessleitvorrichtungen, die in einer Prozessanlage betrieben werden, und eine Steuereinrichtung, die über ein Steuernetzwerk kommunikativ mit der Vielzahl von Prozessleitvorrichtungen gekoppelt ist. Das System umfasst weiterhin einen großen Datenspeicherknoten, der ein greifbares, nicht-transitorisches Speichermedium umfasst, das Daten des Betriebs der Prozessleitumgebung speichert, einen Prozessor, der kommunikativ mit dem großen Datenspeicherknoten verbunden ist, und einen Programmspeicher, der kommunikativ mit dem Prozessor gekoppelt ist. Der Programmspeicher kann ein greifbares, nicht-transitorisches Speichermedium zur Speicherung von Befehlen umfassen, die, wenn sie vom Prozessor ausgeführt werden, den Prozessor veranlassen, eine grafische Benutzerschnittstelle anzuzeigen, die eine Bearbeitungsoberfläche umfasst und eine Auswahl einer Vielzahl grafischer Programmierelemente zur Platzierung auf der Bearbeitungsoberfläche zur Erzeugung eines ersten Diagramms umfasst. Das erste Diagramm kann erste Programmierbefehle repräsentieren, die betreibbar sind, um den Prozessor zu veranlassen, Daten, die auf dem großen Datenspeicherknoten gespeichert sind, zu manipulie-

ren. Außerdem kann der Prozessor, wenn die Befehle vom Prozessor ausgeführt werden, veranlasst werden, die ersten kompilierten Programmbefehle zur Erzeugung einer resultierenden ersten Ausgabe und zur Erzeugung eines Vorhersagemodells auf Grundlage der resultierenden ersten Ausgabe auszuführen. Wenn die Befehle vom Prozessor ausgeführt werden, kann der Prozessor weiterhin veranlasst werden, automatisch ein zweites Diagramm aus dem ersten Diagramm zu erzeugen. Das zweite Diagramm kann zweite Programmierbefehle repräsentieren, die betreibbar sind, um den Prozessor zu veranlassen, sich an eine Live-Datenquelle zu binden und Daten auszuwerten, die von der Live-Datenquelle empfangen werden, wobei zumindest das Modell verwendet wird, das zur Erzeugung einer resultierenden ersten Ausgabe verwendet wurde. Wenn Befehle vom Prozessor ausgeführt werden, kann der Prozessor noch weiterhin veranlasst werden, die zweiten Programmierbefehle, die von dem zweiten Diagramm repräsentiert werden, zu erzeugen, und die kompilierten zweiten Programmierbefehle zur Vorhersage eines Aspekts des Betriebs der Prozessleitumgebung ausführen.

[0021] In einem anderen Aspekt umfasst ein Analytikservice zur Durchführung von Datenanalytik in einer Prozessleitumgebung eine Bibliothek mit Blockdefinitionen. Die Bibliothek kann auf einem greifbaren, nicht-transitorischen Medium gespeichert sein und jede in der Bibliothek gespeicherte Blockdefinition kann (i) einen Zielalgorithmus zur Durchführung einer Aktion in Hinblick auf die Daten in der Prozessleitumgebung und (ii) eine oder mehrere Blockeigenschaften enthalten. Der Analytikservice umfasst einen Satz maschinenlesbarer Befehle, die auf dem greifbaren, nicht-transitorischen Medium gespeichert und die, wenn sie vom Prozessor ausgeführt werden, betreibbar sind, um (1) einem Benutzer über ein Display eine Oberfläche zu präsentieren; (2) dem Benutzer die Bibliothek der Blockdefinitionen zu präsentieren; (3) eine oder mehrere Auswahlen einer oder mehrerer der entsprechenden Blockdefinitionen zu empfangen; (4) einen oder mehreren Blöcke entsprechend der einen oder den mehreren Blockdefinitionen auf der Oberfläche zur Erzeugung eines Moduls, das den einen Block oder die mehrere Blöcke umfasst, platzieren; (5) die Konfiguration einer oder mehrerer Eigenschaften zu ermöglichen, entweder für (i) den einen Block oder die mehreren Blöcke oder (ii) das Modul oder (iii) den einen Block oder die mehreren Blöcke und das Modul; und (6) die Auswertung des (i) einen Block oder der mehreren Blöcke oder (ii) des Moduls zu veranlassen.

[0022] Der Analytikservice umfasst weiterhin einen Ausführungsservice, der auf einem Prozessor betrieben wird, wobei der Ausführungsservice konfiguriert ist (1) eine Anfrage zur Auswertung des einen oder der mehreren Blöcke oder des Moduls zu empfangen;

(2) Befehle, die mit dem einen oder den mehreren Blöcken oder dem Modul korrespondieren, zu kompilieren; (3) einen oder mehrere Jobbearbeitungen zur Ausführung der kompilierten Befehle zu erzeugen und zu veranlassen, dass die eine oder mehreren Jobbearbeitungen die kompilierten Befehle ausführen; und (4) Resultate der einen oder der mehreren Jobbearbeitungen zu empfangen. Der Analytikservice umfasst auch ein Kommunikationsnetzwerk, das einen oder mehrere Prozessoren, die die eine oder die mehreren Jobbearbeitungen ausführen, mit einer oder mehreren verteilten Datenmaschinen koppelt. Jede verteilte Datenmaschine kann auch ein greifbares, nicht-transitorisches Speichermedium umfassen, das Daten, die in der Prozessleitumgebung generiert werden, speichert.

[0023] In der Tat können Daten, die durch den Betrieb der Prozessanlage generiert werden oder diese betreffen, in einer Vielzahl von Datenspeichern, wie beispielsweise relationalen oder nicht-relationalen Datenbanken, gespeichert werden. Diese Datenspeicher können verschiedene Datenstrukturen und Abfragemechanismen nutzen, so dass unterschiedliche Abfragesyntaxen für den Zugriff auf die Daten in unterschiedlichen Datenspeichern benötigt werden. Hierin wird eine standardisierte Abfrage zur Ermöglichung eines Datenzugriffs auf die Datenspeicher unter Verwendung verschiedener Formate beschrieben. Die standardisierte Abfrage unter Verwendung eines standardisierten Datenabfrageformats enthält Informationen, die für den Zugriff auf Daten in einem Datenspeicher erforderlich sind, aber die standardisierte Abfrage kann zum Erhalt solcher Daten nicht direkt ausführbar sein. Stattdessen werden auf Grundlage der standardisierten Abfrage Datenquellen-spezifische Abfragen generiert. Dies kann die Extraktion von Abfrageparametern aus der standardisierten Abfrage und die Erzeugung von einer oder mehreren Datenquellen-spezifischen Abfragen unter Verwendung Datenquellen-spezifischer Abfrageformate, die mit bestimmten Datenquellen assoziiert sind, umfassen. Die Datenquellen-spezifischen Abfragen können zum Zugriff auf und die Auswahl von Daten aus ihren jeweiligen Datenquellen, die dann formatiert werden, um Daten-Frames zu generieren, die die Daten, die in der standardisierten Abfrage angegeben sind, in einem gewünschten Format darzustellen, ausgeführt werden. Dies kann die Ausrichtung von Samplingraten, Dateneigenschaften oder anderer Dateneigenschaften der Daten, die aus den Datenquellen-spezifischen Abfragen erhalten werden, umfassen. In manchen Ausführungsformen können Daten aus mehreren Datenquellen unter Verwendung unterschiedlicher Datenquellen-spezifischer Formate in einem aggregierten Daten-Frame zur weiteren Verwendung in Verfahrensprotokollen oder -analysen kombiniert werden.

[0024] In einem anderen Aspekt wird ein Verfahren, System und ein computerlesbares Medium beschrieben, das Befehle für den Erhalt von Zeitseriendaten aus einer elektronischen Datenquelle speichert und den Empfang einer Angabe der elektronischen Datenquelle aus der die Daten erhalten werden sollen, den Empfang einer Angabe einer Zeitspanne, während der die Daten erhalten werden, den Empfang einer Angabe einer Eigenschaft der zu erhaltenden Daten, die Ermittlung eines mit der elektronischen Datenquelle assoziierten Datenquellen-spezifischen Abfrageformats, die Erzeugung einer dem Datenquellen-spezifischen Format entsprechenden Datenquellen-spezifischen Abfrage auf Grundlage der Angaben der Zeitspanne und der Eigenschaften der zu erhaltenden Daten umfasst, was die auszuführende Datenquellen-spezifische Abfrage veranlasst, die Daten aus der elektronischen Datenquelle auszuwählen und/oder die Daten aus der elektronischen Datenquelle zu empfangen. Die Angaben der Zeitspanne und der Eigenschaften der Daten kann in einem standardisierten Abfrageformat ausgedrückt werden, das unabhängig von der elektronischen Datenquelle ist und die elektronische Datenquelle kann eine relationale Datenbank oder eine nicht-relationale Datenbank sein.

[0025] Das Verfahren kann eine standardisierte Abfrage umfassen, die das standardisierte Abfrageformat verwendet. Die Angabe der Zeitspanne und die Angabe der Eigenschaften können in der standardisierten Abfrage empfangen werden. Die Angabe der elektronischen Datenquelle kann ebenfalls in der standardisierten Abfrage empfangen werden. Die standardisierte Abfrage kann weiterhin ein Format für die Daten oder eine Datei für die Rücksendung der Daten umfassen, wie beispielsweise JavaScript Object Notation (JSON). Die standardisierte Abfrage kann auch eine JSON-formatierte Datei sein. Das standardisierte Abfrageformat kann eine Syntax verwenden, die sich von einer nativen Syntax, die von der elektronischen Datenquelle verwendet wird, unterscheidet. Solch eine von dem standardisierten Abfrageformat verwendete Syntax kann nicht direkt zum Erhalt der Daten von der elektronischen Datenquelle ausführbar sein.

[0026] Die Angabe der Zeitspanne kann mindestens eine Startzeit und mindestens eine Endzeit umfassen. Die Angabe der Zeitspanne kann ebenso eine Abtastrate für die zu erhaltenden Daten umfassen. Die Angabe der Eigenschaften der zu erhaltenden Daten kann eine Angabe von einem oder mehreren Typen von Messwerten im Zusammenhang mit dem Betrieb einer Prozessanlage umfassen. Die eine oder die mehreren Arten von Messwerten können Messwerte von einem oder mehreren in der Prozessanlage eingesetzten Feldgeräten umfassen. Die Angabe der einen oder der mehreren Arten von Messwer-

ten kann ein oder mehrere mit den Daten assoziierte Tags, Aliase und Datentypen umfassen.

[0027] Das Verfahren kann weiterhin den Empfang einer Angabe eines Formats, in dem die Daten von der elektronischen Datenquelle zu erhalten sind, umfassen und die Daten können von der elektronischen Datenquelle in dem angegebenen Format empfangen werden. In manchen Ausführungsformen kann eine Angabe eines Formats, in dem die Daten zu erhalten sind, empfangen werden und die Daten, die von der elektronischen Datenquelle erhalten werden, können in das angegebene Format konvertiert werden.

[0028] Das Datenquellen-spezifische Abfrageformat kann auf Grundlage eines Typs der elektronischen Datenquelle festgelegt werden. Die Erzeugung der Datenquellen-spezifischen Abfrage kann das Abbilden der angegebenen Zeitspanne und der angegebenen Eigenschaften der zu erhaltenden Daten für die Festlegung des Datenquellen-spezifischen Abfrageformats umfassen. Die Veranlassung der Ausführung der Datenquellen-spezifischen Abfrage kann das Senden der Datenquellen-spezifischen Abfrage an eine Schnittstelle der elektronischen Datenquelle umfassen, so dass die Schnittstelle die elektronische Datenquelle unter Verwendung der Datenquellen-spezifischen Abfrage abfragt.

[0029] Es kann ein Daten-Frame erzeugt werden, der die von der elektronischen Datenquelle empfangenen Daten enthält. Der Daten-Frame kann eine Vielzahl von Datenpunkten enthalten. Jeder Datenpunkt kann mit einem Zeitpunkt innerhalb der Zeitspanne assoziiert sein. Jeder Datenpunkt kann weiterhin mit einem Zeitpunkt innerhalb der Zeitspanne, der auf einen ganzzahligen Vielfachen einer Abtastperiode nach einer Startzeit ausgerichtet ist, ausgerichtet sein oder er ist die Startzeit.

[0030] In einem anderen Aspekt, sind ein Verfahren, System und computerlesbares Medium zur Speicherung von Befehlen zur Bereitstellung von Daten von einer elektronischen Datenquelle beschrieben, die den Empfang einer standardisierten Abfrage von einer Daten-abfragenden Entität, die ein standardisiertes Abfrageformat, die Extraktion von Abfrageparametern aus der standardisierten Abfrage, die Erzeugung einer Datenquellen-spezifischen Abfrage in dem Datenquellen-spezifischen Abfrageformat auf Grundlage der extrahierten Abfrageparameter, die Ausführung der Datenquellen-spezifischen Abfrage zum Erhalt der Datenform der elektronischen Datenquelle und/oder die Bereitstellung der erhaltenen Daten an eine datenempfangende Entität umfasst. Das standardisierte Abfrageformat kann sich von einem Datenquellen-spezifischen Abfrageformat, das von der elektronischen Datenquelle verwendet wird, unterscheiden. Die elektronische Datenquelle kann

eine relationale Datenbank sein oder eine nicht-relationale Datenbank. Die Daten-anfordernde Entität kann ein Abfrageblock innerhalb eines Analytikprogramms sein, wie beispielsweise das hierin besprochene Data Analytics Studio. Die Daten-empfangende Entität kann die gleiche Entität sein, wie die Daten-abfragende Entität. Die standardisierte Abfrage kann eine Syntax verwenden, die nicht direkt ausführbar ist, um die Daten von der elektronischen Datenquelle zu erhalten.

[0031] Die Extraktion der Abfrageparameter von der standardisierten Abfrage kann die Festlegung eines Zeitraums und eines Datenparameters der Daten umfassen. Der Zeitraum kann eine Startzeit und eine Endzeit haben und der Datenparameter kann eine Art oder Eigenschaft der Daten, die von der elektronischen Datenquelle erhalten werden soll, angeben. Die Extraktion der Abfrageparameter aus der standardisierten Abfrage kann weiterhin die Festlegung der Samplingrate umfassen.

[0032] Die Bereitstellung der erhaltenen Daten an die abfragende Entität kann die Rücksendung eines mit dem Zeitraum assoziierten Satzes von Datenpunkten umfassen. Jeder Datenpunkt kann einen Zeitstempel umfassen, der einen Zeitpunkt für den mit dem Datenpunkt assoziierten Wert oder die Werte angeben. Solche Zeitstempel können auf Zeitpunkte beschränkt sein, die die Startzeit oder ein ganzzahliges Vielfaches des Zeitraums der Samplingrate nach der Startzeit sind. Zur Erzielung einer vollständigen Zeitserie können dem Datenset mit Zeitstempeln, die die ganzzahligen Vielfachen des Zeitraums der Samplingrate nach der Startzeit angeben, ein oder mehrere Datenpunkte hinzugefügt werden. Die hinzugefügten Datenpunkte können Werte von Dateneinträgen in die elektronische Datenquelle verwenden, die mit den Zeitpunkten assoziiert sind, die jedem Zeitstempel des einen oder der mehreren Datenpunkten am nächsten ist. Der Satz Datenpunkte kann auf Grundlage eines in der standardisierten Abfrage spezifizierten Formats formatiert werden.

[0033] Die Art oder Eigenschaft der Daten, die von dem Datenparameter angegeben wird, kann eine Art von Messung oder eine Messung von einem Typ Messgerät sein. Die Art oder Eigenschaft der Daten kann weiterhin ein spezifisches Messgerät angeben, welches ein Feldgerät sein kann, das innerhalb der Prozessanlage angeordnet ist. Der Datenparameter kann weiterhin ein Tag der zu erhaltenden Daten angeben, ein Alias für die Daten, die an die Daten-abfragende Entität bereitgestellt werden, und/oder ein Datenformattyp für die Daten, die der Daten-abfragen-Entität bereitzustellen sind.

[0034] Die Bereitstellung der Daten an die Daten-abrufende Entität kann das Senden eines Daten-Frame, der die erhaltenen Daten enthält, an die Da-

ten-abfragende Entität umfassen. Die standardisierte Abfrage kann eine Angabe eines Formats für den Daten-Frame umfassen. Der Daten-Frame kann gemäß der Angabe des Formats des Daten-Frames formatiert sein. Die standardisierte Abfrage kann angeben, dass der Daten-Frame der Daten-empfangenden Entität als eine JSON-Datei bereitzustellen ist.

[0035] In einem anderen Aspekt wird ein Verfahren, System und ein computerlesbares Medium, das Anweisungen für den Zugriff auf Prozessanlagedaten von einer Vielzahl elektronischer Datenquellen beschrieben, umfassend den Empfang einer standardisierten Abfrage, die ein standardisiertes Abfrageformat umfasst, die Erzeugung einer ersten Datenquellen-spezifischen Abfrage auf Grundlage der standardisierten Abfrage, die Erzeugung einer zweiten Datenquellen-spezifischen Abfrage auf Grundlage der standardisierten Abfrage, Veranlassung, dass die erste Datenquelle-spezifische Abfrage ausgeführt wird, um einen ersten Satz Daten von der ersten elektronischen Datenquelle zu erhalten, Veranlassung der Ausführung der zweiten Datenquellen-spezifischen Abfrage, um einen zweiten Satz Daten von der zweiten elektronischen Datenquelle zu erhalten, und/oder Erzeugung eines aggregierten Daten-Frames, der das erste und das zweite Datenset umfasst. Der erste und der zweite Datensatz kann jeweils eine Vielzahl von Datenpunkten enthalten, die Informationen enthalten, die von einem oder mehreren Feldgeräten innerhalb einer Prozessanlage gemessen wurden.

[0036] Die erste Datenquellen-spezifische Abfrage kann ein erstes Abfrageformat verwenden, das mit einer ersten elektronischen Datenquelle assoziiert ist, und die zweite Datenquellen-spezifische Abfrage kann ein zweites Abfrageformat verwenden, das mit einer zweiten elektronischen Datenquelle assoziiert ist. Die standardisierte Abfrage kann eine erste Angabe der ersten elektronischen Datenquelle und eine zweite Angabe der zweiten elektronischen Datenquelle enthalten. In manchen Ausführungsformen kann die erste elektronische Datenquelle eine relationale Datenbank sein und die zweite elektronische Datenquelle kann eine nicht-relationale Datenbank sein. Die erste Angabe kann das erste Abfrageformat identifizieren und die zweite Angabe kann das zweite Abfrageformat identifizieren. In manchen Ausführungsformen können den ersten und zweiten Angaben Angaben von Datenspalten vorangestellt sein.

[0037] Das standardisierte Abfrageformat kann eine Abfragesyntax verwenden, die sich sowohl von einer Abfragesyntax des ersten Abfrageformats, als auch einer Abfragesyntax des zweiten Abfrageformats unterscheidet. Die Abfragesyntax des standardisierten Abfrageformats kann nicht direkt zum Erhalt entweder des ersten Datensatzes von der ersten elektronischen Datenquelle oder des zweiten Datensatzes

von der zweiten elektronischen Datenquelle ausführbar sein. Die standardisierte Abfrage kann beispielsweise in einem oder mehreren Objekten oder Arrays einer JSON-Datei enthalten sein.

[0038] Die standardisierte Abfrage kann eine Angabe einer Zeitspanne enthalten, die einen Zeitraum identifiziert, der mit Dateneinträgen in dem ersten Datensatz und dem zweiten Datensatz korrespondiert. Die Zeitspanne kann eine Vielzahl von Zeiträumen umfassen, die durch ausgeschlossene Zeiträume, während denen keine Daten abgefragt werden, separiert sein können. Die Zeitspanne kann auch einen ersten Zeitraum identifizieren, der mit dem ersten Datensatz assoziiert ist, und einen zweiten Zeitraum, der mit dem zweiten Datensatz assoziiert ist. Die standardisierte Abfrage kann auch eine Angabe einer Abtastrate für den aggregierten Daten-Frame umfassen.

[0039] Der erste Datensatz kann Daten umfassen, die eine erste Abtastrate haben, und der zweite Datensatz kann Daten umfassen, die eine zweite Abtastrate haben. In solchen Fällen kann die Erzeugung des aggregierten Daten-Frames die Ausrichtung auf die erste und die zweite Abtastrate umfassen. Die Ausrichtung der ersten und der zweiten Abtastrate kann die Hinzufügung von Datenpunkten umfassen, die mit nicht abgetasteten Zeiten korrespondieren, die benötigt werden, um eine gewünschte Abtastrate zu erzielen. Solchen hinzugefügten Datenpunkten können Werte der abgetasteten Daten gegeben werden, die den hinzugefügten Datenpunkten zeitlich direkt vorausgehen. Die Ausrichtung der ersten und der zweiten Abtastrate kann ebenso die Entfernung abgetasteter Datenpunkte umfassen.

[0040] Systeme zur Implementierung solcher Verfahren können weiterhin einen oder mehrere Prozessoren umfassen, die kommunikativ mit einer oder mehreren Datenquellen und einem oder mehreren Programmspeichern, die computerlesbare Befehle speichern, die, wenn sie von dem einen oder den mehreren Prozessoren ausgeführt werden, veranlassen, dass das Computersystem alle vorstehend beschriebenen Systemfunktionen oder einen Teil davon durchführt, verbunden sind. Computerlesbare Medien können ähnliche computerlesbare Befehle speichern, die von einem oder mehreren Prozessoren eines Computersystems ausgeführt werden, wodurch das Computersystem veranlasst wird alle der vorstehend beschriebenen Systemfunktionen oder einen Teil davon durchzuführen. Zusätzliche oder alternative Aspekte der Verfahren, Systeme oder computerlesbaren Medien können implementiert werden, wie näher in der nachfolgenden detaillierten Beschreibung beschrieben.

[0041] Zusätzlich wird eine neuartige Frequenzanalyse-Analytiktechnik offenbart und diese kann von

einem der hierin beschriebenen Systeme, Verfahren, Vorrichtungen und Techniken bereitgestellt werden. Die neuartige Frequenzanalyse-Analytiktechnik kann Stream-Daten analysieren, um eine frühzeitige, warnende Fehlerentdeckung in Prozessanlagen oder Prozessleitsystemen zu ermöglichen. Insbesondere kann die Frequenzanalyse-Analytiktechnik einen neuen Satz von Prozessvariablen erzeugen, die mit identifizierten Anfangsindikatoren für Fehler, Abnormalitäten, abnehmender Leistung, Zielleistungsniveaus, unerwünschten Bedingungen und/oder gewünschten Bedingungen korrespondieren, und sie kann Zeitseriendaten der neuen Prozessvariablen durch Durchführung einer rollenden FFT für Stream-Prozessdaten ermitteln. Die folgende FFT kann die Stream-Prozessdaten von der Zeitdomäne in die Frequenzdomäne, in der die Werte des neuen Sets von Prozessvariablen ermittelt werden können, wandeln. Die ermittelten Werte der neuen Prozessvariablen können zur Überwachung zurück in die Zeitdomäne gewandelt werden. Die Erkennung einer Präsenz eines Anfangsindikators innerhalb der überwachten Zeitdomänedaten kann eine Angabe eines vorhergesagten Fehlers, Abnormalität, Leistungsrückgangs und/oder anderer unerwünschter Bedingung veranlassen, die produziert und/oder dargestellt werden und in einer Veränderung des Betriebs der Prozessanlage resultieren kann, wodurch die Auswirkung der unerwünschten Störungen, Ereignisse und/oder Bedingungen vermieden, verhindert und/oder vermindert werden kann. Ebenso kann die Erkennung des Vorhandenseins eines Anfangsindikators in den überwachten Zeitdomänedaten eine Angabe einer vorhergesagten, gewünschten Bedingung veranlassen, wie beispielsweise ein zu produzierendes und/oder vorliegendes Zielleistungsniveau, und in manchen Fällen dazu, eine Zeit oder ein Zeitintervall für die/das bzw. während der/dem der Eintritt der gewünschten Bedingung vorausgesagt wird.

KURZBESCHREIBUNG DER ZEICHNUNGEN

[0042] **Fig. 1** ist ein Blockdiagramm, das eine beispielhafte Systemprozessanlage oder ein beispielhaftes Prozessleitsystem veranschaulicht, das eine verteilte, eingebettete industrielle Prozessleistungsüberwachung und/oder Analytikmaschine umfasst;

[0043] **Fig. 2A** ist ein Blockdiagramm eines beispielhaften Systems für verteilte industrielle Prozessleistungsüberwachung/-analyse;

[0044] **Fig. 2B** zeigt eine Ausführungsform eines Arrangements eines beispielhaften industriellen Prozessleistungsüberwachungsanalytiksystems in Beziehung zu einem beispielhaften Prozessleitsystem;

[0045] **Fig. 3** ist ein Blockdiagramm einer beispielhaften industriellen Prozessleistungsüberwachungs- und/oder Analytikmaschine;

[0046] **Fig. 4A** ist ein beispielhaftes Datendiagramm, das mehrere Datenblockfunktionen und Verbindungen veranschaulicht;

[0047] **Fig. 4B** zeigt eine beispielhafte Data Analytics Studio Benutzerschnittstelle;

[0048] **Fig. 4C** zeigt eine beispielhafte Beziehung zwischen einer Datenblock-Definitionsbibliothek, Datenblockdefinitionen, Datenmodulen, Datenblöcken, Datenblockinstanzen und Drähten;

[0049] **Fig. 4D** zeigt ein Beispiel für den Entwurf oder die Erzeugung eines Datenmoduls;

[0050] **Fig. 4E** zeigt einen vergrößerten Bereich einer Data Studio Oberfläche, auf der ein Datenmodul erzeugt wird;

[0051] **Fig. 4F–Fig. 4H** zeigen eine beispielhafte Datenblockdefinitionsvorlage für verschiedene Arten von Datenblöcken;

[0052] **Fig. 4I** zeigt einen beispielhaften Evaluationsablauf, Kompilation oder Einsatz eines Datendiagramms;

[0053] **Fig. 4J** zeigt eine beispielhafte Architektur, die die Präsentation von Standard- und individuellen Visualisierungen veranschaulicht;

[0054] **Fig. 4K** zeigt ein beispielhaftes Szenario, in dem ein Verbunddatenblock erzeugt wird;

[0055] **Fig. 4L** zeigt ein beispielhaftes Offline-Datendiagramm;

[0056] **Fig. 4M** zeigt ein Online-Datendiagramm, das aus der Transformation des beispielhaften Offline-Datendiagramms in **Fig. 4L** generiert wurde;

[0057] **Fig. 4N-1** und **Fig. 4N-2** zeigen beispielhafte Data Analytics Dashboard Benutzerschnittstellen;

[0058] **Fig. 4O** ist ein High-Level-Blockdiagramm einer Steuerschleife, die die Prozessanlagenleistung steuert und optimiert;

[0059] **Fig. 4P** zeigt ein beispielhaftes Verfahren für die Bereitstellung lokalisierter Datenanalytik-Serviceleistungen;

[0060] **Fig. 4Q** zeigt ein beispielhaftes Verfahren für die Bereitstellung lokalisierter Datenanalytik-Serviceleistungen;

[0061] **Fig. 5A** ist ein Blockdiagramm einer beispielhaften Architektur eines Analytikservice in Übereinstimmung mit der vorliegenden Beschreibung;

[0062] **Fig. 5B** ist ein Blockdiagramm, das in größerem Detail eine beispielhafte Datenserviceentität der in **Fig. 5A** gezeigten Architektur veranschaulicht;

[0063] **Fig. 5C** ist ein Flussdiagramm, das ein Verfahren zur Darstellung eines Eigenschaftendialogs für einen Block auf einer Programmoberfläche in Übereinstimmung mit einer vor kurzem beschriebenen Ausführungsform zeigt;

[0064] **Fig. 5D** zeigt eine beispielhafte Programmoberfläche auf der ein beispielhaftes Offline-Diagramm auf der Programmoberfläche konfiguriert ist;

[0065] **Fig. 5E** zeigt ein beispielhaftes Online-Diagramm, das dem Offline-Diagramm in **Fig. 5D** entspricht;

[0066] **Fig. 5F** zeigt einen beispielhaften Eigenschaftendialog für ein Modul;

[0067] **Fig. 5G** zeigt einen beispielhaften Eigenschaftendialog für einen Block eines Offline-Moduls, in dem die Eigenschaften sich sowohl auf den Offline-, als auch den Online-Betrieb beziehen;

[0068] **Fig. 6A** ist ein Blockdiagramm eines beispielhaften analytischen Abfragesystems;

[0069] **Fig. 6B** ist ein Blockdiagramm einer beispielhaften Analysekonfiguration, die eine standardisierte Abfrage verwendet;

[0070] **Fig. 6C** zeigt eine beispielhafte standardisierte Abfrage;

[0071] **Fig. 6D** ist ein Flussdiagramm eines beispielhaften standardisierten Abfrageverfahrens;

[0072] **Fig. 6E** ist ein Flussdiagramm eines beispielhaften standardisierten Abfrageverfahrens;

[0073] **Fig. 7A** zeigt einen Graph der beispielhaften Prozessflussmessungen eines Fackelsystems einer Raffinerie während eines Zeitintervalls;

[0074] **Fig. 7B** zeigt einen Graph der beispielhaften Prozessflussmessungen in **Fig. 7A** während eines anderen Zeitintervalls;

[0075] **Fig. 7C** zeigt einen Graph beispielhafter Prozessflussdaten der Raffinerie, die Fackelereignisse in den Daten enthalten;

[0076] **Fig. 7D** zeigt einen Graph von Zeitseriendaten, die durch Anwendung eines rollenden FFT auf die Prozessflussdaten in **Fig. 7C** ermittelt wurden;

[0077] Fig. 7E zeigt einen Graph der Ergebnisse der Anwendung von PCA zur Analyse der Frequenzdomänedaten in Fig. 7D um ein Fackelereignis herum;

[0078] Fig. 7F zeigt einen Graph der Ergebnisse der Anwendung von PCA zur Analyse der Frequenzdomänedaten in Fig. 7D um ein anderes Fackelereignis herum; und

[0079] Fig. 7G ist ein Flussdiagramm eines beispielhaften Verfahrens zur Bereitstellung einer frühzeitigen Fehlererkennung in Prozessanlagen.

DETAILLIERTE BESCHREIBUNG

[0080] Hierin werden Techniken, Systeme, Vorrichtungen, Komponenten und Verfahren für verteilte, industrielle Leistungsüberwachung und Analytiken offenbart. Generell ermöglichen die Leistungsüberwachungs- und Analytiktechniken Wissensfindung und umsetzbares Wissen über eine industrielle Prozessanlage, die Prozessleitumgebung und/oder ein Prozessleitsystem, das zur Steuerung eines oder mehrerer industrieller Prozesse in Echtzeit betrieben wird. Typischerweise wird ein industrieller Prozess, der von solchen Prozessanlagen oder Kontrollsystmen darin gesteuert wird, zur Herstellung, Verfeinerung, Transformation, Erzeugung oder Produktion physischer Materialien oder Produkte dadurch durchgeführt. Beispiele solcher Prozessanlagen oder Leitsysteme umfassen Ölraffinerien, Papiermühlen, die chemische Fertigung, pharmazeutische Fertigung, Lebensmittelverarbeitung und -vertrieb usw.

[0081] Von größter Bedeutung in jedem Prozessleitsystem, Anlage oder Umgebung ist deren Sicherheit, Zuverlässigkeit und wirtschaftliche Effizienz. Die Betriebssicherheit ist für Prozessleitsysteme oder Anlagen besonders wichtig, da unkontrollierte Prozesse, Fehler, Ausfälle und/oder vermeidbares menschliches Versagen zu Explosionen, Bränden, dem Freisetzen gefährlicher Chemikalien, Umweltschäden, Maschinenschäden und/oder dem Verlust von Menschenleben führen können. Zuverlässigkeit von Maschinen und des Prozessbetriebs ist wichtig um die wirtschaftliche Effizienz und Rentabilität der Anlage zu erhalten und/oder zu verbessern. Weiterhin kann die wirtschaftliche Effizienz und Rentabilität einer Anlage dadurch verbessert werden, dass man den Betrieb der Anlage zur Verbesserung der Leistung abstimmt.

[0082] Fehler, die in einem Prozessleitsystem auftreten, können seine Sicherheit, Zuverlässigkeit und/oder wirtschaftliche Effizienz beeinträchtigen. Fehler in einem Prozessleitsystem treten im Allgemeinen auf, wenn Ereignisse zusammenfallen, so dass Situationen entstehen für die das Prozessleitsystem nicht ausgelegt ist, wie beispielsweise unbekannte Veränderungen der Zusammensetzung der Einsatzma-

terialien, Funktionsminderung von Maschinen, Maschinenversagen und abnorme (oder fehlerhafte) Betriebshandlungen von Benutzern. Weitere Beispiele für Fehler umfassen das ungeplante Abschalten von Maschinen, das Freisetzen von Chemikalien in die Atmosphäre, die Erhöhung von Druck in Überdruckventilen, Temperaturausbrüche in Maschinen und Leistungsverschlechterungen. Natürlich sind Fehler nicht auf System- oder Großereignisse beschränkt. Ein Fehler kann ein abnormes Ereignis sein oder ein Zusammenfallen von Ereignissen, die zu einer wirtschaftlichen, Sicherheits- und/oder Umweltbeeinträchtigung geführt haben oder führen können.

[0083] Normalerweise wird bei Erkennung eines Fehlers (zum Beispiel automatisch durch das Prozessleitsystem) an einer Bedienerchnittstelle ein Alarm erzeugt. Der Bediener kann dann versuchen, den Ursprung des Fehlers zu diagnostizieren und Korrekturmaßnahmen ergreifen. Einige der Schlüsselfaktoren zur Steuerung von Fehlern umfassen daher eine frühzeitige Fehlererkennung und die Reduzierung falscher Alarne. Dazu zählt beispielsweise eine zeitnahe und zuverlässige Fehlererkennung, die Diagnose der Ursache des Fehlers und die Durchführung von Korrekturmaßnahmen, durch die das Instrument, die Steuereinrichtung, das Verfahren und/oder die Maschine, die bzw. das die Fehlerquelle ist, wieder in den Normalbetrieb zurückgebracht wird.

[0084] Bekannte Datenüberwachungs- und Analytikwerkzeuge versuchen Prozessanlagen innerhalb normaler Betriebsregionen zu halten und bieten eine erste Verteidigungsline gegen das Auftreten von Fehlern. Diese Werkzeuge konzentrieren sich jedoch typischerweise auf eine einzelne Schleife oder auf einen extrem stark eingeschränkten Umfang innerhalb einer Prozessanlage. Sie sind nicht dazu in der Lage, auf die Gesamtheit (oder nur einen großen Teil) einer typischen, modernen Prozessleitanlage, in der Hunderte und Tausende von Prozessvariablen bei sehr hoher Geschwindigkeit beobachtet werden müssen, einzugehen. Bekannte Datenüberwachungs- und Analytikwerkzeuge sind weiterhin nicht dazu in der Lage, auf ungewöhnliche Ereignisse, wie eine sich verschlechternde Maschinenleistung, entfernte Feldgeräte oder größere Veränderungen der Betriebsbedingungen, die durch Situationen außerhalb des unmittelbaren, beschränkten Bereichs des Werkzeugs herbeigeführt werden, einzugehen.

[0085] Andererseits sind die neuartigen verteilten industriellen Prozessleistungsbeobachtungs- und Analytiktechniken, -systeme, -vorrichtungen, -komponenten und -verfahren, die hierin offenbart werden, dazu in der Lage, auf industrielle Prozessüberwachung und Analytik jedweder Größe-Umfangs, von der gesamten Prozessanlage bis hinunter zu einer einzelnen Schleife oder sogar einem einzigen Gerät, einzugehen (beispielsweise durch Überwachung und/oder

Analyse). In der Tat sind die hierin offenbarten neuartigen Techniken in manchen Konfigurationen dazu in der Lage simultan auf mehrere Prozessanlagen einzugehen (beispielsweise mehrere Ölraffinerien, die einer einzigen Gesellschaft gehören und von dieser betrieben werden oder sogar unterschiedlichen Gesellschaften, unter der Voraussetzung, dass die Daten verfügbar sind). Generell entdecken und liefern die neuartigen Leistungsüberwachungs- und Analytiktechniken, die hierin offenbart werden, Wissen, das für eine frühzeitige Erkennung indikativ ist und/oder Vorwarnungen über mögliche Fehler, die in Prozessanlagen und Kontrollsystmen auftreten können. Sie geben somit genug Zeit, um vorschriftsmäßige oder Korrekturmaßnahmen zu ergreifen, um ein Auftreten des Fehlers zu verhindern. In manchen Situationen entdecken und liefern die hierin offenbarten neuartigen Techniken auch vorschriftsmäßiges, umsetzbares Wissen zur Vermeidung des Eintretens möglicher Fehler und/oder um die Auswirkungen ihres Eintretens zu begrenzen. Die hierin offenbarten neuartigen Techniken können weiterhin Wissen entdecken und liefern, das indikativ für mögliche Verbesserungen der Anlageneffizienz ist und umsetzbares Wissen zur Realisierung der Effizienzverbesserungen entdecken und liefern.

BEISPIELHAFTES PROZESSLEITSYSTEM MIT VERTEILTER INDUSTRIELLER PROZESSÜBERWACHUNG UND ANALYTIK

[0086] Wie vorstehend besprochen wird eine Prozessanlage, ein Prozessleitsystem oder eine Prozesssteuerungsumgebung, die zumindest einen Teil der hierin beschriebenen neuartigen industriellen Prozessüberwachungs- und Analytiktechniken umfasst oder unterstützt zur Steuerung einer oder mehrerer industrieller Prozesse in Echtzeit betrieben. Als solches können in der Prozessanlage oder dem Steuersystem eines oder mehrere verkabelte oder kabellose Prozessleitgeräte, Komponenten oder Elemente enthalten sein, die physikalische Funktionen (wie beispielsweise das Öffnen oder Schließen von Ventilen, die Messung von Temperaturen, Druck und/oder anderer Prozess- und/oder Umgebungsparameter usw.) zur Steuerung eines Prozesses, der innerhalb der Prozessanlage oder des Systems ausgeführt wird, durchführen. Die Prozessanlage oder das Prozessleitsystem können beispielsweise eines oder mehrere verkabelte Kommunikationsnetzwerke und/oder eines oder mehrere kabellose Kommunikationsnetzwerke umfassen. Die Prozessanlage oder das Steuersystem kann zentralisierte Datenbanken, wie beispielsweise kontinuierliche, Batch- und andere Arten historischer Datenbanken, umfassen.

[0087] Zur Veranschaulichung zeigt **Fig. 1** ein detailliertes Blockdiagramm einer beispielhaften Prozessanlage oder einer Prozesssteuerungsumgebung **5**, die alle oder jedwede der hierin beschriebenen

verteilten industriellen Prozessüberwachungs- und Analytiktechniken umfasst oder unterstützt. Das Prozessleitsystem **5** umfasst mehrere verteilte Datenmaschinen eines verteilten industriellen Prozessüberwachungs- und Analytiksystems, das von der Prozessanlage oder Umgebung **5** unterstützt wird, oder die darin enthalten oder integriert sind. (Eine vollständigere Beschreibung eines verteilten industriellen Prozessüberwachungs- und Analytiksystems ist in nachfolgenden Abschnitten enthalten.) Jede der verteilten Datenmaschinen des industriellen Prozessüberwachungs- und Analytiksystems ist direkt oder indirekt mit irgendeiner Komponente oder einem Teil der Prozessanlage **5** verbunden (beispielsweise physisch verbunden oder kabellos verbunden). Eine verteilte Datenmaschine kann beispielsweise in ein bestimmtes Gerät oder einen Knoten der Prozessanlage **5** eingebettet oder hergestellt sein. Zur Bildung einer einheitlichen Entität kann eine Datenmaschine an ein bestimmtes Gerät oder Knoten der Anlage **5** angebracht oder damit gekoppelt sein oder die Datenmaschine kann an einer herkömmlichen Kommunikationsverbindung der Anlage **5** angebracht oder damit gekoppelt sein. In **Fig. 1** markiert ein eingekreistes „DDE“ ein jeweiliges Beispiel einer verteilten Datenmaschine.

[0088] Wie vorstehend besprochen können verteilte Datenmaschinen in Prozessleitgeräten eingebettet sein, deren Hauptfunktion die automatische Erzeugung und/oder der Empfang von Prozessleitdaten zur Durchführung von Funktionen zur Steuerung eines Prozesses in der Prozessanlagenumgebung **5** in Echtzeit ist. Jeweilige Datenmaschinen können beispielsweise in Prozesssteuereinrichtungen, Feldgeräten und I/O Geräten eingebettet oder hergestellt sein. In der Prozessanlagenumgebung **5** empfangen Prozesssteuereinrichtungen Signale, die indikativ für die von den Feldgeräten durchgeführten Prozessmessungen sind, verarbeiten diese Informationen zur Implementierung einer Steueroutine und erzeugen Steuersignale, die über herkömmliche verkabelte oder kabellose Prozesssteuerungskommunikationsverbindungen oder Netzwerke an andere Feldgeräte zur Steuerung des Prozessbetriebs in der Anlage **5** gesendet werden. Typischerweise führt mindestens ein Feldgerät eine physische Funktion zur Steuerung eines Betriebsprozesses aus (beispielsweise Öffnen oder Schließen eines Ventils, Erhöhung oder Verringerung einer Temperatur usw.) und einige Arten der Feldgeräte kommunizieren über I/O Geräte mit Steuereinrichtungen. Prozesssteuereinrichtungen, Feldgeräte und I/O Geräte können verkabelt oder kabellos sein und die Prozessanlagenumgebung oder System **5** können über eine beliebige Anzahl und Kombination von verkabelten und kabellosen Prozesssteuereinrichtungen, Feldgeräten und I/O Geräten verfügen und diese können jeweils eine verteilte Datenmaschine umfassen.

[0089] Fig. 1 veranschaulicht beispielsweise eine Prozesssteuereinrichtung **11**, die über die Input/Output (I/O) Karten **26** und **28** mit den verkabelten Feldgeräten **15–22** kommunikativ verbunden ist, und die über einen kabellosen Gateway **35** und eine Prozesssteuerungsdatenautobahn oder Backbone **10** (der einen oder mehrere verkabelte und/oder kabellose Kommunikationsverbindungen umfassen kann und unter Verwendung eines gewünschten oder geeigneten Kommunikationsprotokolls, wie beispielsweise einem Ethernet Protokoll, implementiert werden kann) mit den kabellosen Feldgeräten **40–46** kommunikativ verbunden ist. In einer Ausführungsform ist die Steuereinrichtung **11** unter Verwendung von einem oder mehreren anderen Kommunikationsnetzwerken als dem Backbone **10**, wie beispielsweise unter Verwendung einer beliebigen Anzahl anderer verkabelter oder kabelloser Kommunikationsverbindungen, die ein oder mehrere Kommunikationsprotokolle unterstützen, beispielsweise Wi-Fi oder andere IEEE 802.11 konforme kabellose Local Area Network Protokolle, mobile Kommunikationsprotokolle (beispielsweise WiMAX, LTE oder andere ITU-R kompatible Protokolle), Bluetooth®, HART®, Wireless-HART®, Profibus, FOUNDATION® Fieldbus usw., mit dem kabellosen Gateway **35** kommunikativ verbunden. Der Backbone **10** und diese anderen Kommunikationsnetzwerke sind Beispiele für „herkömmliche“ Prozessleitkommunikationsnetzwerke, wie sich hierin darauf bezogen wird.

[0090] Die Steuereinrichtung **11**, bei der es sich beispielsweise um die von Emerson Process Management vertriebene DeltaV™ Steuereinrichtung handeln könnte, kann zur Implementierung eines Batch-Prozesses oder eines kontinuierlichen Prozesses, der mindestens einige der Feldgeräte **15–22** und **40–46** nutzt, betrieben werden. In einer Ausführungsform ist die Steuereinrichtung **11**, zusätzlich zu der kommunikativen Verbindung mit der Prozesssteuerungsdatenautobahn **10**, unter Verwendung jeder gewünschten Hardware und Software, die beispielsweise mit Standard 4–20 mA Geräte, I/O Karten **26**, **28** und/oder einem intelligentem Kommunikationsprotokoll, wie dem FOUNDATION® Fieldbus Protokoll, dem HART® Protokoll, dem WirelessHART® Protokoll usw., assoziiert ist, mit mindestens einem der Feldgeräte **15–22** und **40–46** kommunikativ verbunden. In Fig. 1 sind die Steuereinrichtung **11**, die Feldgeräte **15–22** und die I/O Karten **26**, **28** verkabelte Geräte und die Feldgeräte **40–46** kabellose Feldgeräte. Natürlich könnten die verkabelten Feldgeräte **15–22** und die kabellosen Feldgeräte **40–46** mit jedem anderen gewünschten Standard oder Protokoll konform sein, wie beispielsweise verkabelten oder kabellosen Protokollen, einschließlich Standards und Protokollen, die in der Zukunft entwickelt werden.

[0091] Die Steuereinrichtung **11** in Fig. 1 enthält einen Prozessor **20**, der eine oder mehrere Prozess-

steueroutinen **38** (die beispielsweise in einem Speicher **32** gespeichert sind) implementiert oder überwacht. Der Prozessor **30** ist zur Kommunikation mit den Feldgeräten **15–22** und **40–46** konfiguriert und mit anderen Knoten kommunikativ mit der Steuereinrichtung **11** verbunden. Es sollte beachtet werden, dass Teile der hierin beschriebenen Steuer routinen oder Module (einschließlich Qualitätsvorhersagen und Fehlererkennungsmodule oder Funktionsblöcke) von unterschiedlichen Steuereinrichtungen oder anderen Geräten implementiert oder ausgeführt werden können, wenn dies gewünscht wird. Ebenso können die hierin beschriebenen Kontrollroutinen oder Module **38**, die innerhalb des Prozessleitsystems **5** implementiert werden sollen, jede Form haben, einschließlich Software, Firmware, Hardware usw. Steueroutinen können in jedem gewünschten Softwareformat, wie beispielsweise objektorientierte Programmierung, Leiterlogik, sequentielle Funktionspläne, Funktionsblockdiagramme, implementiert werden oder jede andere Softwareprogrammiersprache oder jedes andere Designparadigma verwenden. Die Steueroutinen **38** können in jeder gewünschten Art von Speicher **32**, wie beispielsweise Random Access Memory (RAM) oder Read-Only Memory (ROM), gespeichert werden. Ebenso können die Steueroutinen **38** hartkodiert werden, beispielsweise in einen oder mehrere EPROMs, EEPROMs, applikationsspezifischen integrierten Schaltkreisen (ASICs) oder anderen Hardware- oder Firmwareelementen. Die Steuereinrichtung **11** kann auf jede gewünschte Weise für die Implementierung einer Steuerstrategie oder Steueroutine konfiguriert werden.

[0092] In manchen Ausführungsformen implementiert die Steuereinrichtung **11** eine Steuerstrategie unter Verwendung dessen, was gemeinhin als Funktionsblöcke bezeichnet wird, wobei jeder Funktionsblock ein Objekt oder ein anderer Teil (beispielsweise eine Subroutine) einer Gesamtsteuerroutine ist und in Verbindung mit anderen Funktionsblöcken (über Verbindungen genannte Kommunikationen) der Implementierung der Prozesssteuerungsschleifen innerhalb des Prozesssteuersystems **5** dienen. Steuerbasierte Funktionsblöcke führen entweder eine Eingabefunktion aus, wie sie beispielsweise mit einem Sender, einem Messfühler oder einem anderen Messgerät für Prozessparameter assoziiert wird, eine Steuerfunktion, wie sie beispielsweise mit einer Steuerroutine, die PID, Fuzzy Logic usw. ausführt, assoziiert wird, oder eine Ausgabefunktion, die den Betrieb eines Geräts steuert, wie beispielsweise ein Ventil, um eine physische Funktion innerhalb des Prozessleitsystems **5** durchzuführen. Natürlich existieren auch Hybrid- und andere Arten von Funktionsblöcken. Funktionsblöcke können in der Steuereinrichtung **11** gespeichert und von ihr ausgeführt werden. Dies ist typischerweise der Fall, wenn diese Funktionsblöcke für Standard 4–10 mA Geräte und manche Arten von Smart Field Geräten, wie HART®, verwen-

det oder mit ihnen assoziiert werden, oder sie können in den Feldgeräten selbst gespeichert und implementiert werden, was im Fall von FOUNDATION® Fieldbus Geräten der Fall sein kann. Die Steuereinrichtung **11** kann eine oder mehrere Steuer routinen **38** umfassen, die eine oder mehrere Steuerschleifen implementieren können und durch Ausführung eines oder mehrerer Funktionsblöcke ausgeführt werden können.

[0093] Die verkabelten Feldgeräte **15–22** können jedwede Arten von Geräten sein, wie beispielsweise Messfühler, Ventile, Sender, Positionierer usw., während die I/O Karten **26** und **28** jedwede Arten von I/O Geräten sein können, die konform zu dem gewünschten Kommunikations- oder Steuereinrichtungsprotokoll sind. In **Fig. 1** sind die Feldgeräte **15–18** Standard 4–20 mA Geräte oder HART® Geräte, die über analoge Leitungen oder kombinierte analoge und digitale Leitungen mit der I/O Karte **26** kommunizieren, während die Feldgeräte **19–22** intelligente Geräte, wie beispielsweise FOUNDATION® Fieldbus Feldgeräte, sind, die unter Verwendung eines FOUNDATION® Fieldbus Kommunikationsprotokolls über einen digitalen Bus mit der I/O Karte **28** kommunizieren. In manchen Ausführungsformen kommunizieren jedoch mindestens einige der verkabelten Feldgeräte **15, 16** und **18–21** und/oder mindestens einige der Big Data I/O Karten **26, 28** zusätzlich oder alternativ unter Verwendung der Prozesssteuerungsdatenautobahn **10** und/oder unter Verwendung anderer, geeigneter Steuersystemprotokolle (beispielsweise Profibus, DeviceNet, Foundation Fieldbus, ControlNet, Modbus, HART usw.) mit der Steuereinrichtung **11** (und/oder anderen Big Data Knoten).

[0094] Wie in **Fig. 1** gezeigt enthalten alle, die Steuereinrichtung **11**, die I/O Karten **26** und **28** und die verkabelten Feldgeräte **15–16, 18–21**, eine jeweilige, eingebettete, verteilte Datenmaschine, wie durch das eingekreiste „DDE“ markiert, die über einen oder mehrere Datenanalytikkommunikationskanäle und/oder Netzwerke (in **Fig. 1** nicht gezeigt) mit anderen verteilten Datenmaschinen kommunizieren.

[0095] In **Fig. 1** kommunizieren die kabellosen Feldgeräte **40–46** unter Verwendung eines kabellosen Protokolls, wie beispielsweise das WirelessHART® Protokoll, über ein herkömmliches, kabelloses Prozesssteuerungskommunikationsnetzwerk **70**. Solche kabellosen Feldgeräte **40–46** können direkt mit einem oder mehreren anderen Geräten oder Knoten des Prozesssteuerungs-Datenanalytikkommunikationsnetzwerks **112**, die ebenfalls für eine kabellose Kommunikation konfiguriert sind (unter Verwendung des kabellosen Protokolls oder eines anderen kabellosen Protokolls zum Beispiel), kommunizieren. Zur Kommunikation mit einem oder mehreren anderen Knoten, die nicht für die kabellose Kommunikation konfiguriert sind, können die kabellosen Feld-

geräte **40–46** einen kabellosen Gateway **35** verwenden, der mit der Prozesssteuerungsdatenautobahn **10** oder mit einem anderen herkömmlichen Prozessleitkommunikationsnetzwerk verbunden ist.

[0096] Der kabellose Gateway **35** kann Zugang zu mehreren kabellosen Geräten **40–58** des kabellosen Kommunikationsnetzwerks **70** bieten. Der kabellose Gateway **35** ermöglicht insbesondere die kommunikative Kopplung der kabellosen Geräte **40–58**, der verkabelten Geräte **11–28** und/oder anderer Knoten oder Geräte der Prozesssteuerungsanlage **5**. Der kabellose Gateway **35** kann beispielsweise durch Verwendung der Prozesssteuerungsdatenautobahn **10** und/oder durch Verwendung von einem oder mehreren anderen herkömmlichen Kommunikationsnetzwerken der Prozessanlage **5** eine kommunikative Kopplung ermöglichen.

[0097] Ebenso wie die verkabelten Feldgeräte **15–22**, können die kabellosen Feldgeräte **40–46** des kabellosen Netzwerks **70** physische Steuerfunktionen innerhalb der Prozessanlage **5** ausführen, beispielsweise das Öffnen oder Schließen von Ventilen oder die Durchführung von Messungen der Prozessparameter. Die kabellosen Feldgeräte **40–46** sind jedoch für Kommunikation unter Verwendung des kabellosen Protokolls des Netzwerk **70** konfiguriert. Als solches sind die kabellosen Feldgeräte **40–46**, der kabellose Gateway **35** und die anderen kabellosen Knoten **52–58** des kabellosen Netzwerks **70** Produzenten und Konsumenten kabelloser Kommunikationspakete.

[0098] In manchen Szenarien kann das kabellose Netzwerk **70** nicht-kabellose Geräte umfassen. Ein Feldgerät **48** in **Fig. 1** könnte beispielsweise ein 4–20 mA Legacy-Gerät und ein Feldgerät **50** könnte ein herkömmliches verkabeltes HART® Gerät sein. Zur Kommunikation innerhalb des Netzwerks **70** können die Feldgeräte **48** und **50** über einen kabellosen Adapter oder eine Historisierung an diesem oder **52b** mit dem kabellosen Kommunikationsnetzwerk **70** verbunden sein. Die kabellosen Adapter **52a, 52b** können andere Kommunikationsprotokolle, wie Foundation® Fieldbus, PROFIBUS, DeviceNet usw., unterstützen. Das kabellose Netzwerk **70** kann weiterhin einen oder mehrere Netzwerk-Zugangspunkte **55a, 55b** umfassen, bei denen es sich um separate physische Geräte handeln kann, die in verkabelter Kommunikation mit dem kabellosen Gateway **35** stehen oder mit dem kabellosen Gateway **35** als eine integrale Vorrichtung bereitgestellt werden. Das kabellose Netzwerk **70** kann außerdem einen oder mehrere Router **58** zur Weiterleitung von Paketen von einem kabellosen Gerät zu einem anderen kabellosen Gerät innerhalb des kabellosen Kommunikationsnetzwerks **70** umfassen. Die kabellosen Geräte **40–46** und **52–58** können miteinander und über kabellose Verbindungen **60** des kabellosen Kommunikations-

netzwerks **70** und/oder über die Prozesssteuerungsdatenautobahn **10** mit dem kabellosen Gateway **35** kommunizieren.

[0099] Wie in **Fig. 1** gezeigt enthalten der Adapter **52a**, der Zugangspunkt **55a** und der Router **58** jeweils eine jeweilige, eingebettete, verteilte Datenmaschine, wie durch das eingekreiste „DDE“ markiert, die über einen oder mehrere Datenanalytikkommunikationskanäle und/oder Netzwerke (in **Fig. 1** nicht gezeigt) mit anderen verteilten Datenmaschinen kommunizieren.

[0100] In manchen Ausführungsformen umfasst das Prozesssteuerungssystem **5** einen oder mehrere kabellose Zugangspunkte **72**, die unter Verwendung anderer kabelloser Protokolle, wie beispielsweise WiFi oder anderen IEEE 802.11 konformen kabellosen Local Area Network Protokollen, mobilen Kommunikationsprotokollen, wie WiMAX (Worldwide Interoperability for Microwave Access), LTE (Long Term Evolution) oder anderen ITU-R (International Telecommunication Union Radio Communication Sector) kompatiblen Protokollen, Kurzwellen-Funkkommunikationen wie Nahfeldkommunikation (NFC) und Bluetooth, oder andere kabellose Kommunikationsprotokolle, kommunizieren. Typischerweise ermöglichen solche kabellosen Zugangspunkte **72** Handheld oder anderen tragbaren Computergeräten (beispielsweise Benutzerschnittstellengeräte **73**) Kommunikation über ein jeweiliges herkömmliches, kabelloses Prozesssteuerungskommunikationsnetzwerk, das sich von dem kabellosen Netzwerk **70** unterscheidet und ein anderes kabelloses Protokoll unterstützt als das kabellose Netzwerk **70**. Ein kabelloses oder tragbares Benutzerschnittstellengerät **73** kann beispielsweise eine mobile Arbeitsstation oder diagnostische Testausrüstung sein, die von einem Bediener innerhalb der Prozessanlage **5** verwendet wird. In manchen Szenarios kommunizieren zusätzlich zu tragbaren Computergeräten ein oder mehrere Prozesssteuerungsgeräte (beispielsweise Steuereinrichtung **11**, Feldgeräte **15–22** oder kabellose Geräte **35, 40–58**) ebenfalls unter Verwendung des von den Zugangspunkten **72** unterstützten Zugangspunkten **72**.

[0101] In manchen Ausführungsformen umfasst das Prozesssteuerungssystem **5** ein oder mehrere Gateways **75, 78** zu Systemen extern zu dem unmittelbaren Prozessleitsystem **5**. Typischerweise sind solche Systeme Kunden oder Lieferanten der Informationen, die von dem Prozessleitsystem **5** generiert oder berechnet werden. Die Prozessleitanlage **5** kann beispielsweise einen Gateway-Knoten **75** zur kommunikativen Verbindung der unmittelbaren Prozessanlage **5** mit einer anderen Prozessanlage umfassen. Zusätzlich oder alternativ kann die Prozesssteuerungsanlage **5** einen Gateway-Knoten **78** zur kommunikativen Verbindung der unmittelbaren Prozessanlage **5** mit einem externen öffentlichen oder privaten Sys-

tem, wie einem Laborsystem (beispielsweise Laboratory Information Management System oder LIMS), einer Operator Rounds Datenbank, einem Materialbearbeitungssystem, einem Wartungsverwaltungssystem, einem Produktbestandkontrollsysteem, einem Produktionsplanungssystem, einem Wetterdatensystem, einem Versand- und Bearbeitungssystem, einem Verpackungssystem, dem Internet, dem Prozessleitsystem eines anderen Anbieters oder anderen externen Systemen, umfassen.

[0102] Wie in **Fig. 1** gezeigt enthalten der Zugangspunkt **72**, das Benutzerschnittstellengerät **73** und der Gateway **75** jeweils eine jeweilige, eingebettete, verteilte Datenmaschine, wie durch das eingekreiste „DDE“ markiert, die über einen oder mehrere Datenanalytikkommunikationskanäle und/oder Netzwerke (in **Fig. 1** nicht gezeigt) mit anderen verteilten Datenmaschinen kommunizieren.

[0103] Es wird angemerkt, dass obwohl **Fig. 1** nur eine einzelne Steuereinrichtung **11** mit einer endlichen Zahl von Feldgeräten **15–22** und **40–46**, kabellosen Gateways **35**, kabellosen Adapters **52**, Zugriffspunkten **55**, Routern **58** und kabellosen Prozesssteuerungskommunikationsnetzwerken **70**, die in einer Prozessanlage **5** enthalten sind, veranschaulicht, dies nur eine veranschaulichende und nicht einschränkende Ausführungsform ist. Es kann eine beliebige Anzahl an Steuereinrichtungen **11** in dem Prozessleitsystem oder System **5** enthalten sein und jede der Steuereinrichtungen **11** kann mit einer beliebigen Anzahl an verkabelten oder kabellosen Geräten und Netzwerken **15–22, 40–46, 35, 52, 55, 58** und **70** zur Steuerung eines Prozesses in der Anlage **5** kommunizieren.

BEISPIEL FÜR DIE ARCHITEKTUR EINES VERTEILTEN INDUSTRIELLEN PROZESSLEISTUNGSÜBERWACHUNGS/ -ANALYTIKSYSTEMS

[0104] **Fig. 2A** enthält ein Blockdiagramm eines beispielhaften Systems **100** für die verteilte industrielle Prozessleistungsüberwachung/Analytik, das hierin auch austauschbar als Daten-Analytik-System (DAS) bezeichnet wird. Das DAS **100** kann beispielsweise zusammen mit der Prozessanlage **5** in **Fig. 1** betrieben werden und wird hierin zur Vereinfachung der Darstellung erläutert. Es versteht sich jedoch, dass mindestens ein Teil des DAS **100** zusammen mit anderen Prozessanlagen und/oder Prozessleitsystemen als der Prozessanlage **5** betrieben werden kann.

[0105] Das DAS **100** unterstützt generell lokaliserte Leistungsüberwachung und/oder Analytik und unterstützt gleichzeitig eine umfangreiche (beispielsweise systemweite und/oder über mehrere Vorrichtungen oder Knoten der Prozessanlage **5** hinweg) Leistungsüberwachung, Data Mining und Datenana-

lytik für Prozessanlagenumgebungen. Insofern umfasst das System **100** eine Vielzahl von verteilten Datenmaschinen (Distributed Data Engines; DDE). Beispiele dafür werden in **Fig. 2A** durch die Bezugszeichen **102a–102e** und in **Fig. 2B** durch die Bezugszeichen **102f–102h** gekennzeichnet. Wie hierin verwendet bezieht sich das Bezugszeichen „**102x**“ auf eine oder mehrere der DDEs **102a–102h**. Mindestens einige der in **Fig. 2A** veranschaulichten verteilten Datenmaschinen korrespondieren mit den in **Fig. 1** veranschaulichten verteilten Datenmaschinen. Die Datenmaschine **102a** in **Fig. 2A** ist beispielsweise die Datenmaschine, die in dem kabellosen Gateway **35** von **Fig. 1** enthalten ist und die Datenmaschine **102b** in **Fig. 2A** ist die Datenmaschine, die in der Steuereinrichtung **11** der **Fig. 1** eingebettet ist. Zusätzlich umfasst das System **100** andere Datenmaschinen **102**, die in **Fig. 1** nicht gezeigt werden. Zum Beispiel ist eine verteilte Datenmaschine **102c** mit der Datenautobahn **10** verbunden, eine verteilte Datenmaschine **102d** ist in eine zentrale Big Data Anwendung **108** der Prozessanlage **5** eingebettet und eine verteilte Datenmaschine **102e** ist in einen Big Data Cloud Knoten **110** eingebettet, wobei der Big Data Cloud Knoten **110** den Datenanalytikanforderungen der Prozessleitanlage **5** nachkommen kann (und in manchen Konfigurationen auch anderen Prozessleitanlagen zuarbeiten kann). Selbstverständlich ist das System **100** nicht nur auf fünf Datenmaschinen **102a–102e** oder acht Datenmaschinen **102a–102h** beschränkt, sondern kann eine beliebige Anzahl an verteilten Datenmaschinen umfassen, von denen mindestens einige in ihre jeweiligen Datenquellen eingebettet oder darin hergestellt sind (beispielsweise in jeweiligen Prozesssteuerungsvorrichtungen der Prozessanlage **5**) und/oder von denen mindestens einige anderweitig mit einigen anderen Datenquellen (beispielsweise Komponenten, Teil usw.) der Prozessanlage **5** verbunden sind.

[0106] Wie vorstehend erwähnt kommunizieren verteilte Datenmaschinen typischerweise nicht mit anderen verteilten Datenmaschinen durch Verwendung der herkömmlichen Prozesssteuerungsanlagenkommunikationsnetzwerke (beispielsweise den Backbone **10**, das kabellose Netzwerk **70**, die Kabelverbindung zwischen der I/O Karte **28** und Geräten **19–22** usw. der **Fig. 1**), obwohl in manchen Konfigurationen manche verteilten Datenmaschinen **102x** möglicherweise mindestens einige Informationen durch Verwendung eines herkömmlichen Prozesssteuerungskommunikationsnetzwerks an andere Datenmaschinen kommunizieren. Generell kommunizieren die Datenmaschinen **102x** jedoch mit anderen Datenmaschinen **102x** durch Verwendung eines oder mehrerer Datenanalytikkommunikationsnetzwerke **112**, die getrennt und unterschiedlich von herkömmlichen Prozesssteuerungsnetzwerken sind. Analytikdaten werden zwischen verteilten Datenmaschinen oder Knoten des Netzwerks **112** Stream-

übertragen, beispielsweise durch Verwendung eines Analytik-Streamdienstes, eines Streaming- und/oder Warteschlangen-Protokolls und/oder durch Verwendung eines Messaging Brokers oder Systems **115**, das Streaming unterstützt, wie beispielsweise eine Streaming-Quelle, Flume, HDFS, ZeroMQ, Kafka, Microsoft Message Bus, MQTT, AMQP, RabbitMQ usw., ein benutzerdefiniertes spezialisiertes Historienobjektkommunikationsprotokoll, wie es in der vorstehend erwähnten US-Anmeldung Nr. 14 / 506,863 mit dem Titel „STREAMING DATA FOR ANALYTICS IN PROCESS CONTROL SYSTEMS“ beschrieben ist oder ein anderes geeignetes Kommunikationsprotokoll. In **Fig. 2A** werden durch die gestrichelten Linien mehrere Zweige des Datenanalytikkommunikationsnetzwerks **112** dargestellt, obwohl in **Fig. 2A** das Netzwerk **112** der Klarheit halber nicht vollständig dargestellt wird. In einer Ausführungsform kann somit mindestens ein Teil des Datenanalytiknetzwerks **112** in gewissem Sinne ein Überlagerungsnetzwerk auf den herkömmlichen Prozesssteuerungskommunikationsnetzwerken innerhalb der Anlage **5** sein. Zum Beispiel können mindestens einige der physischen Geräte, die die DDEs **102x** und das Datenanalytiknetzwerk **112** umfassen, sich auf dem Gelände der Prozessanlage **5** befinden und in manchen Fällen unter den Geräten der Prozessanlage **5**.

[0107] In einer anderen Ausführungsform kann mindestens ein Teil des Datenanalytikkommunikationsnetzwerks **112** entlang oder weitgehend parallel zu herkömmlichen Prozesssteuerungskommunikationsnetzwerken implementiert sein, wie in **Fig. 2B** gezeigt. In **Fig. 2B** enthält die Prozessanlage **5** mehrere Prozesssteuerungsgeräte oder -komponenten, die kommunikativ mit einem herkömmlichen Prozesssteuerungskommunikationsnetzwerk verbunden sind, zum Beispiel zur Unterstützung der Steuerung eines oder mehrerer Prozesse innerhalb der Anlage **5**. Drei solcher Prozesssteuerungsgeräte/-komponenten verfügen jeweils über darin eingebettete DDE **102f**, **102g** und **102h** mit denen sie assoziiert sind und jede der DDEs **102f–102h** ist kommunikativ mit dem Datenanalytiknetzwerk **112** verbunden. Der größte Teil der Geräte, die das Datenanalytiknetzwerk **112** unterstützen und/oder die es umfasst befindet sich jedoch nicht unter den Geräten der Prozessanlage **5**, sondern kann sich stattdessen dezentral in einem Unternehmens-Rechenzentrum befinden.

[0108] In einer anderen Ausführungsform kann mindestens ein Teil des Datenanalytikkommunikationsnetzwerks **112** als ein logisches Netzwerk innerhalb der Prozessanlage **5** implementiert sein. In dieser Ausführungsform können beispielsweise sowohl Prozessleitdaten, als auch Analytikdaten über eine gleiche physische Kommunikationsverbindung transportiert werden, die logisch das Aussehen unabhängiger Verbindungen hat, beispielsweise eine herkömmliche

Prozesskommunikationsverbindung und eine Datenanalytikkommunikationsverbindung.

[0109] Ungeachtet der Implementierung des Datenanalytiknetzwerks **112** überschneidet sich das Datenanalytiknetzwerk **112** jedoch mit herkömmlichen Prozesssteuerungskommunikationsnetzwerken an Geräten oder Knoten innerhalb des Prozesskontrollsystems **5**, die Informationen an andere Geräte oder Knoten innerhalb des Prozesskontrollsystems **5** über herkömmliche Prozesskontrollkommunikationsnetzwerke übertragen und in denen verteilte Datenmaschinen eingebettet sind (beispielsweise Datenmaschinen **102a**, **102b**, **102f**, **102g**, **102h**) und/oder aufgrund von verteilten Datenmaschinen, die an herkömmliche Prozesssteuerungskommunikationsnetzwerke zum Zweck eine Fenster zu den Daten, die darüber übertragen werden, angeschlossen sind (beispielsweise Datenmaschine **102c**).

[0110] Das Datenanalytiknetzwerk **112** kann jeweils gewünschte Kommunikationstechnologie und/oder Kommunikationsprotokolle nutzen. Mindestens ein Teil des Datenanalytiknetzwerks **112** kann verkabelte Technologie nutzen und mindestens ein Teil des Datenanalytiknetzwerks **112** kann kabellose Technologie nutzen. Generell kann das Datenanalytiknetzwerk **112** jedoch Datenpakettransportprotokolle und/oder Technologie nutzen, wie beispielsweise IP oder andere asynchrone Datenpaketprotokolle über Ethernet.

[0111] Auf höherer Ebene oder Schicht kann das Datenanalytiknetzwerk **112** ein Stream-fähiges Protokoll, einen Message-Broker, der Echtzeit-Datenfeeds bearbeitet, und/oder ein Messaging-System, das die Echtzeit-Datenfeeds **115** bearbeitet, wie Apache Kafka, nutzen. Wie in **Fig. 2A** gezeigt ermöglicht es die von dem Datenanalytiknetzwerk **112** bereitgestellte Streaming-Fähigkeit **115** dem verteilten Leistungsüberwachungs-Analytiksystem oder DAS **100** sich lokal zu binden und diverse analytische Services nahe an den Datenquellen bereitzustellen (beispielsweise nahe an oder an Vorrichtungen oder Knoten der Prozessanlage **5**, in die verteilte Datenmaschinen **102x** eingebettet sind, wie beispielsweise die Knoten **11**, **18**, **26**, **28**, **72**, **35**, **52a**, **55a**, **72** in **Fig. 1**). Gleichzeitig ermöglicht die Streaming-Fähigkeit **115** des Systems **100** dem System **100** auch umfangreichere Vorhersagen und Optimierungen bereitzustellen, da nur bestimmte, quellengenerierte Daten, die zur Unterstützung einer breiteren Datenanalytik oder einer Datenanalytik auf höherer Ebene notwendig sind, an die empfangenden Datenmaschinen kommuniziert werden müssen. Wenn die Datenmaschine **102d** beispielsweise nur eine bestimmte Datenanalytik für Daten ausführt, die von jeder der Steuereinrichtungen der Prozessanlage **5** generiert und innerhalb von einer Stunde nachdem eine bestimmte Konfigurationsveränderung an jeder Steuereinrich-

tung instantiiert wurde, erfasst wird, Stream-überträgt die an der Steuereinrichtung **11** eingegebettete Datenmaschine **102b** nur die erforderlichen Daten an die Maschine **102d** (sie Stream-überträgt beispielsweise nur die Ausgabedaten, die von der Steuereinrichtung **11** während der Stunde nachdem die bestimmte Konfigurationsveränderung instantiiert wurde generiert wurden, an die Datenmaschine **102d**; sie überträgt aber keine anderen Ausgabedaten, die von der Steuereinrichtung **11** generiert wurden).

[0112] Wie in **Fig. 2A** gezeigt sind Teile des verteilten Leistungsüberwachungs-/Analytiksystems oder DAS **100** lokal innerhalb der Prozesssteuerumgebung oder Anlage **5** angeordnet. Zum Beispiel befinden sich die verteilten Datenmaschinen **102a**, **102b**, **102c** und **102d** auf dem Gelände der Prozessanlage **5**. Wie in **Fig. 2A** ebenfalls gezeigt sind andere Teile des Systems **100** dezentral angeordnet. Die Datenmaschine **102** befindet sich beispielsweise in der Cloud **110**. Andere Datenmaschinen **102x** können sich an einem oder mehreren dezentralen Standorten befinden (nicht gezeigt), wie an einer Serverbank, die mehrere Prozessanlagen oder Prozesssteuersysteme bedient. Selbstverständlich kann das System **100** in manchen Konfigurationen über keine lokal angeordneten Datenmaschinen oder über keine dezentral angeordneten Datenmaschinen verfügen. Das heißt, das System **100** kann die gesamte Leistungsüberwachung und Analytik (beispielsweise über eine oder mehrere verteilte Datenmaschinen) auf dem Gelände der ihr unterstehenden Prozesssteuerungsumgebung **5** (beispielsweise lokal) in seiner Gesamtheit außerhalb des Geländes der ihr unterstehenden Prozesssteuerungsumgebung **5** (beispielsweise dezentral) oder durch Verwendung einer Kombination von auf und außerhalb des Geländes befindlicher verteilter Datenmaschinen (beispielsweise sowohl lokal, als auch dezentral) bereitstellen.

BEISPIELHAFTE ARTEN DER LEISTUNGSÜBERWACHUNG UND -ANALYTIK FÜR INDUSTRIELLE STEUERSYSTEME

[0113] Leistungsüberwachung und Analytiken industrieller Steuersysteme umfassen generell die Sammlung von Daten, die von dem ihnen unterliegenden Steuersystem generiert werden (beispielsweise in Echtzeit während das System zur Steuerung eines oder mehrerer Prozesse betrieben wird), und die Durchführung von einer oder mehreren Funktionen oder Operationen an mindestens einigen der gesammelten Daten, um Wissen darüber zu ermitteln, wie gut das Steuersystem arbeitet, und in manchen Fällen, um Maßnahmen zu ermitteln, die zur Verbesserung der Leistung des Systems ergriffen werden können. Datenanalytiken für industrielle Prozessleitsysteme lassen sich in drei allgemeine Kategorien unterteilen: deskriptive Analytik, prädiktive Analytik

und präskriptive Analytik. Es folgt eine allgemeine Diskussion dieser Analytic-Kategorien.

[0114] Deskriptive Analytiken ermöglichen es einem Benutzer zu entdecken, was innerhalb eines Prozessleitsystems oder einer Anlage geschehen ist, wann es geschehen ist, wie oft es geschehen ist und welches Problem oder welche Probleme aus dem, was geschehen ist, entstanden sind. Typischerweise werden deskriptive Analytiken für Daten, die aus der Überwachung des Prozesssystems oder der Anlage (beispielsweise *a posteriori*) gewonnen werden, durchgeführt, und können Techniken wie grundlegende Berechnungen, Skalierung und Standardstatistiken verwenden.

[0115] Präskriptive Analytiken erlauben es einem Benutzer die Betriebsabläufe innerhalb eines Prozessleitsystems oder einer Anlage zu optimieren. Präskriptive Analytik erlaubt es einem Benutzer zum Beispiel Fragen zu beantworten, wie beispielsweise: Was ist die beste Antwort? Was ist angesichts der Unsicherheit das beste Ergebnis? Was sind signifikant unterschiedliche und bessere Optionen? Prädiktive Analysen können wesentliche Qualitätsvariablen oder Kennzahlen der Prozess-Betriebsabläufe in industriellen Prozessleitanlagen und -systemen identifizieren, überwachen und steuern. Zusätzlich können prädiktive Analytiken identifizieren, was angesichts eines Satzes zukünftiger Eingaben oder kausaler Bedingungen mit wesentlichen Qualitätsvariablen oder Kennzahlen der Prozess-Betriebsabläufe geschieht. Die vorhergesagten Werte können dann von präskriptiver Analytik zur Generierung einer präskriptiven Maßnahme genutzt werden. Typischerweise nutzen präskriptive Analytiken Techniken, wie bedingungsbasierte Optimierung und Mehrzieloptimierung. Bekannte Überwachungs- und Analytikwerkzeuge sind jedoch rudimentär und in ihrem Umfang stark eingeschränkt.

[0116] Mit bekannten Datenüberwachungs- und Analytikwerkzeugen können Benutzer oder Bediener beispielsweise die Leistung eines einzelnen Geräts oder einer Einheit der Anlage durch Verwendung bedingungsbasierter Optimierungswerkzeuge, die auf Grundlage von Daten arbeiten, die aus der Überwachung eines einzelnen Geräts oder einer Einheit (beispielsweise *a posteriori*) gewonnen wurden, optimieren. Da ein Prozessleitsystem oder eine Anlage jedoch zahlreiche Geräte und Prozesseinheiten umfasst, könnte der Gesamtprozess aufgrund der Prozessinteraktionen zwischen mehreren Einheiten immer noch weit vom Optimum entfernt sein. Mehrzieloptimierungstechniken für interagierende Einheiten oder eine koordinierte Optimierung über einen Teil des Prozessleitsystems hinweg stehen in bekannten Datenüberwachungs- und Analytikwerkzeugen zwar zur Verfügung (beispielsweise First-Principle-Based-Modellierung), diese koordinierten Op-

timierungstechniken sind aber aufgrund der zunehmenden Komplexität der Prozesse schwierig und brauchen zu lang zur Ausführung (sofern sie sich überhaupt ausführen lassen), um nützlich zu sein. Bekannte Mehrzieloptimierungswerkzeuge arbeiten beispielsweise mit Daten, die aus der Beobachtung mehrerer Geräte oder Einheiten (beispielsweise *a posteriori*) gewonnen wurden, um optimale Betriebspunkte zu identifizieren und um Trajektorien zu bestimmen, um den aktuellen Betriebspunkt zum optimalen zu manövrieren. Jeder Zyklus der Datensammlung und deren Berechnung kann jedoch so lange dauern, dass der Prozess bis die präskriptiven Maßnahmen identifiziert werden in seinem Betrieb soweit fortgelaufen ist, dass die identifizierten präskriptiven Maßnahmen nicht mehr das Optimum sind oder sogar ungültig, kontraproduktiv und/oder gefährlich geworden sind. Wenn ein Benutzer weiterhin versucht, das Volumen und den Umfang (und somit die Zeit, die für sie benötigt wird) der Berechnungen zur reduzieren oder einzuschränken, indem der die Anzahl der Variablen, die in das Mehrzieloptimierungswerkzeug eingegeben wird, einschränkt, trifft der Benutzer ein menschliches Urteil darüber, welche Variablen als Eingaben ausgewählt werden, was nicht nur den Umfang einschränkt, sondern auch falsch sein könnte und daher irreführende oder falsche Ergebnisse produzieren könnte.

[0117] Weiterhin sind bekannte Überwachungs- und Analytikwerkzeuge häufig nicht dazu in der Lage, die Identität und Werte von wesentlichen Qualitätsvariablen bei Bedarf bereitzustellen (beispielsweise aufgrund von Einschränkungen der verfügbaren Messtechniken) und sie können keine zeitnahen, sachdienlichen Resultate liefern. Derzeit für die prädiktive Analytik von Prozessleitsystemen eingesetzte Techniken umfassen zum Beispiel entweder Offline-Techniken (zum Beispiel in einem Labor) oder Online-Techniken (zum Beispiel durch einen Analysator). Beide Techniken haben ihre jeweiligen Nachteile. Die Offline-Ermittlungen wesentlicher Qualitätsvariablen ist nicht wünschenswert, da durch die signifikante Verzögerung, die durch Labortests herbeigeführt wird, Ergebnisse suboptimal oder sogar für die aktuellen Prozessbedingungen ungültig werden (die gemessenen Signale können beispielsweise nicht als Rückmeldungen verwendet werden). Der Einsatz von Online-Analysatoren zur Ermittlung von wesentlichen Qualitätsvariablen ist ebenfalls weniger als wünschenswert, da bekannte Online-Analysatoren problematisch und teuer sind und eine regelmäßige und kostenintensive Wartung erforderlich machen. Solche Einschränkungen können die Qualität von Produkten, die Produktion von Abfällen und die Betriebssicherheit stark beeinflussen.

[0118] Außerdem ist ein herkömmliches, zentrales Leistungsüberwachungssystem in umfangreichen Überwachungssituationen stark eingeschränkt.

Ein geeignetes Leistungsüberwachungssystem sollte über Fehlertoleranzfähigkeit, Betriebszuverlässigkeit und wirtschaftliche Effizienz verfügen. Diese werden kompromittiert, wenn für umfangreiche Überwachungssituationen ein zentrales Leistungsüberwachungssystem eingesetzt wird. Ein zentrales Überwachungssystem kümmert sich insbesondere um Fehler in allen Einheiten des Prozesses gleichzeitig. Wird ein Fehler in einer Einheit gefunden, wird das zentrale Überwachungssystem in seiner Fähigkeit, weitere Fehler bei anderen Einheiten, die während des gleichen Zeitraums auftreten, eingeschränkt, da ein zentrales Überwachungssystem alle gemessenen Variablen in seine Berechnung mit einbezieht und wenn eine Variable nicht zur Verfügung steht oder ein Kommunikationskanal blockiert ist, kann das gesamte Überwachungssystem seine Funktion einstellen. Weiterhin ist die Echtzeit-Fähigkeit eines zentralen Überwachungssystems durch die niedrigste Abtastrate der Variablen eingeschränkt, was die Effizienz des Überwachungssystems verringert. Weiterhin können unterschiedliche Variablen in einer Anlage sehr unterschiedliche Dynamikmaßstäbe (selbst innerhalb der gleichen Einheit) haben und ein einzelnes Überwachungssystem gilt üblicherweise nicht für alle Variablen eines gesamten Prozesses oder einer gesamten Einheit und ist insbesondere für Situationen, in denen sich Variablen gegenseitig aufeinander auswirken, beispielsweise bei interagierenden Prozesseinheiten, suboptimal. Noch weiter ist, wenn Prozessleitsysteme geografisch verteilt sind (zum Beispiel bei großen Distanzen zwischen verschiedenen Prozesseinheiten, wie beispielsweise in einer Öl-Pipeline), jede Einheit typischerweise mit einer separaten Leistungsüberwachung ausgestattet, wodurch eine lokale (aber keine insgesamte) Leistungsüberwachung sowie Übertragungsverzögerungen, Datenverlust und Einschränkungsprobleme in Verbindung mit der Batterie ermöglicht werden.

[0119] Anders als die rudimentären, eindimensionalen und/oder zentralen prädiktiven Überwachungs- und Analytiktechniken, die derzeit in Verwendung sind, und anders als der a posteriori Charakter deskriptiver und präskriptiver Analytiken ermöglichen die hierin beschriebenen, neuartigen verteilten Leistungsüberwachungs- und Analytiksysteme, -verfahren, -vorrichtungen, -komponenten und -techniken für industrielle Steuersysteme und Anlagen es einem Benutzer zu überwachen, was derzeit in dem Prozessleitsystem oder der Anlage zu jedem beliebigen Zeitpunkt geschieht und auf Grundlage der vorliegenden Daten in Echtzeit vorauszusagen, was aller Wahrscheinlichkeit als nächstes geschieht oder später geschehen wird. Die hierin offenbare neuartige Technik ermöglicht es dem Benutzer Fragen zu beantworten, wie beispielsweise: „Was geschieht im Moment?“ „Was geschieht als nächstes?“ „Was, wenn diese Trends sich fortsetzen?“ Dies mit nur wenig oder keiner Benutzer- oder menschlichen Befan-

genheit in Hinblick auf die Eigenschaften der Prozessdaten.

[0120] Die neuartigen Techniken für die hierin beschriebenen verteilten Leistungsüberwachungs- und Analytiksysteme für die industrielle Steuerung umfassen generell eine Plattform (beispielsweise die in **Fig. 2A** gezeigte Systemarchitektur) und Anwendungen, die auf oder zusammen mit der Plattform betrieben werden. Die Plattform nutzt datengetriebene Verfahren, die mit Geräten und Prozesswissen ergänzt werden, zur Unterstützung beispielsweise der Schätzung, Überwachung und dem Data Mining eines Prozessleitsystems oder einer Anlage (beispielsweise das Prozessleitsystem **5**) vor, während und nachdem das Prozessleitsystem oder die Anlage zur Steuerung eines oder mehrerer Prozesse betrieben wird. In manchen Implementierungen kann die Plattform mehrere Prozessleitsysteme oder Anlagen unterstützen.

[0121] Die Leistungsüberwachungs-/Analytik-Applikationen, die auf oder zusammen mit der Plattform betrieben werden, können beispielsweise abgeleitete Messungen, Gerätüberwachung, Fehlererkennung, Prozessvorhersagen, Kausalität, andere Überwachungsanwendungen und/oder andere Analytik-Anwendungen umfassen. Techniken, die von den Applikationen verwendet werden, können Data Mining, Optimierung, Vorhersagemodellierung, maschinelles Lernen, Simulation, verteilte Zustandsschätzung und ähnliches umfassen. Als solches können die Leistungsüberwachungs-/Analytikapplikationen zur Überwachung, Vorhersage und Diagnose von Leistungsverschlechterungen und Fehlern einer beliebigen Anzahl von Teilen des Prozessleitsystems **5** verwendet werden, einschließlich von Bereichen, wie beispielsweise Instrumentierung, Geräten, Steuerung, Benutzerinteraktionen und Prozess.

[0122] Dementsprechend können die hierin beschriebenen Techniken zur verteilten industriellen Leistungsüberwachung und Analytik inferentielle Mechanismen umfassen, die kontinuierliche Online-Schätzungen kritischer Prozessvariablen aus den leicht verfügbaren Prozessmessungen bereitstellen. Zur Unterhaltung von Datenmodellen über einen längeren Zeitraum kann das System weiterhin die Fähigkeit zur Überwachung, Abstimmung und Optimierung der Datenmodelle unterstützen. Verfahren, die verwendet werden können, umfassen im einfachsten Fall die Verstellung der Modelle (beispielsweise Erhöhung der normalen Betriebsregion auf die Vorhersage $+/-3s$). Komplexere Beispiele umfassen Modellumschaltungen und Modellanpassungen.

[0123] In der Tat schufen die Erfinder im Rahmen eines Forschungs- und Prototyp-Testprogramms den Prototyp eines Frameworks für industrielle Prozessleistungsdatenüberwachung und Datenanalytik un-

ter Verwendung der hierin beschriebenen Techniken, um auf Probleme der Soft-Sensor-Entwicklung bei Vorhandensein von Nichtlinearität, Nicht-Gaussverteilung, Unregelmäßigkeit von Datenmustern, Prozessbeschränkungen und/oder anderen Faktoren einzugehen. Bei diesen Prototypen und Forschungsarbeiten wurde die Überwachung und Analyse von Daten, die von mehreren Soft-Sensoren für Spezialchemikalien-, Chargen- und Kohlenwasserstoffprozessen durchgeführt und das resultierende Wissen wurde zur Verbesserung der erhaltenen Prozesse genutzt. Unsere Industriepartner haben einen signifikanten wirtschaftlichen Nutzen und Umweltnutzen gemeldet. Beispiele für unsere Forschungen und Prototypen werden in einem späteren Abschnitt beschrieben.

[0124] Zurückkehrend zu **Fig. 2A** liefert das verteilte industrielle Leistungsüberwachungs- und Analytiksystem oder DAS **100** die Werkzeuge, Konfiguration und Schnittstellen, die zur Unterstützung einer lokalisierten Leistungsüberwachung erforderlich sind, und unterstützt gleichzeitig umfassendes Data Mining und Datenanalysen für Prozessanlagenumgebungen, wie beispielsweise die Prozessanlage **5** der **Fig. 1**. Das System **100** kann beispielsweise eine Benutzerschnittstellenanwendung (zum Beispiel das Data Analytics Studio) zur Konfiguration und Entwicklung von Datenanalytikmodellen, eine Laufzeitmaschine zur Ausführung der Modelle (die ganz oder teilweise auf Grundlage von Echtzeitdaten, die von dem Prozessleitsystem generiert werden, betrieben werden können) und die gleiche oder eine andere Benutzerschnittstellenanwendung (zum Beispiel ein Run-time Dashboard) zur Darstellung der Analyseergebnisse umfassen. Das System **100** kann auf mehreren Arten von Datenquellen betrieben werden oder unterstützen, einschließlich, wie bereits erwähnt, Echtzeitwerte (zum Beispiel kontinuierliche Daten in Echtzeit), Ereignissammlung, Chargendatenerfassung, Operator-Runden-Daten, LIMS-Daten, externe Daten wie Lieferketten-Daten und/oder Wetterdaten und jede andere Art von Daten, die mit dem Prozessleitsystem assoziiert sind, einschließlich sowohl strukturierter als auch unstrukturierter Daten. Das System **100** kann einen Satz von Standardanalytiken „aus dem Kasten“ bereitstellen, wie deskriptive Statistiken, Histogrammplots, Korrelationsdiagramme usw. Zusätzlich kann das System **100** eine strukturierte Umgebung für Benutzer zur Erzeugung der gewünschten Analytiken bereitstellen und um sowohl Quelldaten, als auch n-Auftragsdaten, die durch Anwendung mehrerer unterschiedlicher Datenanalysen zur Identifikation impliziter Beziehungen zwischen unterschiedlichen Datensätzen des Prozessleitsystems generiert wurden, einzusehen, und/oder um Produktfähigkeiten, -qualität und andere Eigenschaften vorherzusagen. Das System **100** kann generell die Entdeckung von Wissen über das Prozessleitsystem sowie von umsetzbarem Wis-

sen (beispielsweise zur Verbesserung oder Optimierung der Leistung des Prozessleitsystems, zur Behebung und/oder Vorhersage von Fehlern, zur Verbesserung der Zuverlässigkeit, zur Verbesserung der Sicherheit, zur Verbesserung der wirtschaftlichen Effizienz usw.) ermöglichen, die als Datenmodelle dargestellt werden können.

[0125] Wie in **Fig. 2A** gezeigt kann das System **100** stark verteilt sein, es kann beispielsweise zahlreiche verteilte Datenmaschinen **102x** umfassen. Das Netzwerk **112** der eingebetteten Datenmaschinen **102x** kann zahlreiche (beispielsweise Hunderte, Tausende oder sogar Millionen) Sensoren und Quellen von Prozessleitsystem-Informationen miteinander verbinden. Wie bereits erwähnt sind die Datenmaschinen **102x** in Form von Clustern in der Nähe, an oder innerhalb der Vorrichtungen und Knoten des Prozessleitsystems angeordnet (beispielsweise Fertigungsmaschinen, Prozesssteuerungsgeräten, Kommunikationsknoten, Materialbearbeitungssystemen, Laborsystemen, Benutzern der Anlage und auch dem Prozess selbst). Als solches ist der eingebettete Charakter des verteilten industriellen Leistungs- und Analytiksystems **100** eng mit der physischen Prozesssteuerungsanlage gekoppelt. Die Einbettung und Integration der Datenmaschinen **102x** in das Herstellungsgefüge der Prozesssteuerungssystemgeräte und Knoten kann die Datenmaschinen **102x** jedoch für Endbenutzer fast unsichtbar machen. Eine Datenmaschine **102x** ist typischerweise klein, kabellos mit dem Netzwerk **112** verbunden, Bandbreiten-eingeschränkt und wird unter physikalischen Beschränkungen betrieben, wie beschränkter Energie und das Erfordernis einer adäquaten Wärmeableitung, da sie in ein Prozesssteuerungsgerät, wie beispielsweise einen Temperatur- oder eine andere Art von Messfühler eingebettet ist.

[0126] Wie bereits erwähnt sind verteilte Datenmaschinen **102x** mit anderen Datenmaschinen **102x** über das Netzwerk **112** unter Verwendung von Streaming-Protokollen und/oder Warteschlangenprotokollen verbunden. Jede verteilte Datenmaschine **102x** kann eine oder mehrere Datenüberwachungs- und/oder Datenanalytikanwendungen unterstützen. Die bloße Anzahl an verbundenen Applikationsclustern macht die Verwendung statistisch korrekter (anstatt deterministischer) Algorithmen für die Ressourcenabrechnung, Fehlererkennung und -korrektur, Systemverwaltung usw. erforderlich und jedes Cluster kann eine Funktionalität aufnehmen, die für lokalisierter Bedürfnisse von Interesse ist.

[0127] Als solches kann das verteilte industrielle Leistungsüberwachungs- und Analytiksystem **100** die Sicherheit, Effizienz und Produktivität von Prozesssteuerungsanlagen erhöhen. Das System **100** kann zum Beispiel Prozessparameter genau steuern und somit die Gesamtkosten der Prozessfertigung redu-

zieren. Weiterhin kann die Integration des Systems **100** in die Prozesssteuerungsumgebung **5** aufgrund einer genauen Prozesssteuerung und Qualitätssicherung in Echtzeit in einer besseren Produktqualität und weniger Abfall; mehr Flexibilität, schnell konfigurierte Produktionslinien als Folge programmierbarer Untersysteme; Überwachung der Systemgesundheit, was zu einer effektiveren, vorbeugenden und kostengünstigeren Wartung führt; sicheren Arbeitsumgebungen aufgrund besserer Überwachung und Kontrolle; und besseren Komponentenmontagetechniken, wie beispielsweise durch die Verwendung intelligenter RFID-Tags, um nur einige der zahllosen Vorteile aufzuzählen, resultieren.

[0128] Weiterhin kann das verteilte industrielle Leistungsüberwachungs- und Analytiksystem **100** die Mensch-Maschine-Interaktionen zwischen Benutzern und dem Prozesssteuerungssystem **5** bis zum Punkt der Generation einer Mensch-plus-Maschine Steuerschleife in Echtzeit innerhalb der Prozessanlage **5** verbessern. Eine verbesserte Mensch-Maschine-Interaktion kann beispielsweise die Qualität und Produktivität dadurch verbessern, dass es keine Bediener-/Wartungs-/Zuverlässigkeitfehler gibt und Unfälle reduziert werden. Weiterhin lässt sich die Verfügbarkeit, Zuverlässigkeit und kontinuierliche Service-Qualität der Prozesssteuerungsanlage **5** durch eine erweiterte Kontrolle, Redundanz, intelligente Alarmauslösung, Selbstdiagnose und Reparaturen, die durch das System **100** ermöglicht werden, erreichen.

BEISPIEL EINER INDUSTRIELEN LEISTUNGSDATENÜBERWACHUNGS- / DATENANALYTIKMASCHINE

[0129] Wenn wir uns nun den verteilten, industriellen verteilten Leistungsdatenüberwachungs- und/oder Analytikmaschinen **102x** zuwenden, kann eine verteilte Datenmaschine **102x** ein Knoten des Datenanalytikkommunikationsnetzwerks **112** sein, der alle oder die meisten Daten mit Prozesssteuerungsbezug (beispielsweise Big Data), die von einer oder mehreren Datenquellen oder Quellen erzeugt oder anderweitig beobachtet werden, beispielsweise durch die Vorrichtung oder Komponenten, in die die Datenmaschine **102x** eingebettet ist oder durch die Komponente(n), mit der oder denen die Datenmaschine **102x** gekoppelt oder an die sie angebracht ist, sammelt, beobachtet, abruft, empfängt, verarbeitet, speichert, im Cache speichert und/oder analysiert. In manchen Situationen kann eine verteilte Datenmaschine **102x** zusätzliche Daten generieren (beispielsweise Resultate der Analysen, die sie durchführt) und/oder ausgewählte Daten an andere Knoten des Datenanalytiknetzwerks **112** senden oder weiterleiten. Die Begriffe „Prozesssteuerungs Big Data“, „Prozess Big Data“ und „Big Data“, die hierin austauschbar verwendet werden, beziehen sich ge-

nerell auf alle (oder fast alle) Daten, die von Vorrichtungen und/oder Komponenten (beispielsweise sowohl Prozesssteuerungsgeräte/-komponenten, als auch Analytikgeräte/-komponenten) generiert, empfangen und/oder beobachtet werden, die in einem Prozesssteuerungssystem oder -anlage enthalten oder damit assoziiert sind (beispielsweise das Prozesssteuerungssystem oder Anlage **5**) und insbesondere alle (oder fast alle) Daten, die generiert, empfangen und/oder während das Prozesssteuerungssystem oder die Anlage in Echtzeit zur Steuerung eines oder mehrerer Prozesse ausgeführt wird. In einer Ausführungsform werden alle Daten (einschließlich aller Prozessdaten und aller Analytikdaten), die von allen Vorrichtungen, die in der Prozessanlage **5** enthalten oder mit dieser assoziiert sind, generiert, erzeugt, empfangen oder anderweitig beobachtet werden, gesammelt und als Big Data innerhalb des Datenanalytikkommunikationsnetzwerk **112** gespeichert. In Prozessanlagen und Prozesssteuerungsumgebungen ist diese Sammlung und Analyse von Big Data der Schlüssel zur Verbesserung der Sicherheit, Zuverlässigkeit und wirtschaftlichen Effizienz, da die Zeitdimension und das Vorhandensein oder Fehlen bestimmter Datenpunkte kritisch sein kann. Wenn zum Beispiel ein bestimmter Datenwert nicht an eine Empfängerkomponente der Prozessanlage innerhalb eines bestimmten Zeitintervalls geliefert wird, kann ein Prozess unkontrolliert werden, was zu einem Brand, einer Explosion, Verlust von Maschinen und/oder dem Verlust von Menschenleben führen kann. Weiterhin können sich mehrere und/oder komplexe zeitbasierte Beziehungen zwischen verschiedenen Komponenten, Entitäten und/oder Prozessen, die innerhalb der Prozessanlage und/oder extern zur Prozessanlage betrieben werden, auf die Betriebseffizienz, Produktqualität und/oder Anlagensicherheit auswirken.

[0130] Die erzeugten, gesammelten, beobachteten, abgerufenen, erhaltenen, gespeicherten, zwischen gespeicherten, verarbeiteten, ausgewerteten und/oder von den Verteilte-Daten-Engines **102x** weitergeleiteten Prozesssteuerungs-Massendaten können Daten enthalten, die unmittelbar bei der Steuerung eines Prozesses verwendet oder durch diese Steuerung innerhalb der Anlage **5** erzeugt wurden, z. B. erstrangige Echtzeit- und Konfigurationsdaten, die durch Prozesssteuerungsgeräte wie Controllern, Eingabe-/Ausgabe-(I/O)-Geräten und Feldgeräten erzeugt oder verwendet werden. Zusätzlich oder wahlweise können die Daten-Engines **102x** Daten im Zusammenhang mit der Lieferung und Durchleitung solcher erstrangigen Prozesssteuerungsdaten und sonstigen Daten innerhalb der Prozessanlage **5** erzeugen, sammeln, beobachten, verarbeiten, auswerten, speichern, empfangen, abrufen, zwischenspeichern und/oder weiterleiten, z. B. Daten im Zusammenhang mit der Netzwerksteuerung des Datenauswertungskommunikationsnetzwerks **112** und/

oder anderer Kommunikationsnetzwerke in der Anlage 5, auf die Bandbreite hinweisende Daten, Netzwerkzugriffsversuche, Diagnosedaten etc. Weiterhin können einige Daten-Engines **102x** auf Ergebnisse, Erlerntes und/oder Informationen, die innerhalb des Prozesssteuerungsdaten-Auswertungskommunikationsnetzwerks **112** durch Verarbeitung und/oder Auswertung der von ihnen gesammelten Prozesssteuerungs-Massendaten erlernt wurden, hinweisende Daten erzeugen, sammeln, beobachten, speichern, zwischenspeichern, abrufen, empfangen und/oder weiterleiten. Üblicherweise werden derartige Auswertungsergebnisse, Erlerntes und/oder erlernte Informationen aus den von einer oder mehreren Daten-Engines **102x** durchgeführten Auswertungen erzeugt.

[0131] Als solches wird eine Verteilte-Daten-Engine (z. B. VDE **102x**) hier wechselweise als „Massendaten-Vorrichtung“, „Massendaten-Vorrichtungsknoten“ oder als „Vorrichtungsknoten“ bezeichnet. In den meisten Fällen enthält eine Verteilte-Daten-Engine des Massendaten-Vorrichtungsknotens **102x** Mehrkern-Hardware (z. B. Mehrkernprozessoren) zur Übermittlung und zum Empfang von Massendaten in Echtzeit (z. B. über Streaming) und, in einigen Ausführungsformen, zur Zwischenspeicherung und/oder Speicherung der Echtzeit-Massendaten für späteres Streaming oder sonstige Lieferung über das Prozesssteuerungsdaten-Auswertungskommunikationsnetzwerk **112**. Dementsprechend enthält eine Verteilte-Daten-Engine **102x** auch einen Speicher (z. B. hochdichten Speicher) zur Zwischenspeicherung und/oder Speicherung der Massendaten. Beispiele von Echtzeit-Daten die übermittelt, empfangen, gestreamt, zwischengespeichert, gesammelt, gespeichert, empfangen, abgerufen, zwischengespeichert und/oder in sonstiger Weise von den Daten-Engines **102x** beobachtet werden können, können Prozesssteuerungsdaten wie Messdaten, Konfigurationsdaten, Stapelverarbeitungsdaten, Ereignisdaten und/oder fortlaufende Daten enthalten. Beispielsweise können Konfigurationen, Stapelverarbeitungsrezepturen, Sollwerte, Datenausgaben, Quoten, Steuerungstätigkeiten, Diagnosen, Alarme, Ereignisse und/oder diesbezüglichen Änderungen entsprechende Echtzeitdaten gesammelt werden. Weitere Beispiele von Echtzeitdaten können Prozessmodelle, Statistiken, Statusdaten, Netzwerk- und Anlagenverwaltungsdaten und Auswertungsergebnisse sein. Beschreibungen verschiedener Arten von Massendaten-Beispielvorrichtungen und ihrer Komponenten, die mit beliebigen oder sämtlichen hier beschriebenen Techniken verwendet werden können, finden sich unter den Aktenzeichen der vorstehend genannten US-Patent-Anmeldungen 13/784,041, 14/174,413 und 14/212,493, obwohl selbstverständlich beliebige oder sämtliche hier beschriebenen Techniken mit sonstigen geeigneten Massendaten-Vorrichtungen verwendet werden können.

[0132] Üblicherweise arbeitet die Verteilte-Daten-Engine **102x** im Allgemeinen mit Massendaten, die von einer oder mehreren Datenquellen innerhalb der Prozesssteuerungsanlage oder des -systems 5 oder in sonstiger Verbindung mit der den Echtzeitbetrieb der Prozesssteuerungsanlage oder des -systems 5 stehenden Datenquellen erzeugt oder bereitgestellt werden. So kann beispielsweise eine VDE **102x** Daten sammeln und mit einem Zeitstempel versehen, die von dem Gerät, in welches sie eingebettet ist, empfangen und/oder erzeugt wurden, oder die von ihrer entsprechenden Datenquelle oder Datenquellen empfangen und/oder erzeugt wurden. Die gesammelten Daten können (zumindest vorübergehend) im lokalen Speicher der Daten-Engine **102x** gespeichert werden. In einigen Situationen kann zumindest ein Teil der Daten an eine oder mehrere weitere Daten-Engines **102x** mittels eines besonderen Historienobjektkommunikationsprotokolls wie in der vorstehend genannten US-Patentanmeldung Nr. 14/506,863 unter dem Titel „STREAMING DATA FOR ANALYTICS IN PROCESS CONTROL SYSTEMS“ beschrieben oder anderen geeigneten Kommunikationsprotokollen oder einem Nachrichtenübermittlungssystem wie Kafka übertragen oder gestreamt werden.

[0133] In einigen Ausführungsformen unterstützen eine oder mehrere Daten-Engines **102x** umfangreiche Datenextrahierungen und Datenauswertungen aus mehrdimensionalen Daten einschließlich fortlaufenden Echtzeit-Werte-, Ereignissammlungs-, Stapeldatensammlungs- oder Bedienerrunden-Daten und/oder sonstige Daten. Eine Verteilte-Daten-Engine **102x** kann zur Durchführung einer oder mehrerer Datenauswertungen ihrer lokal gesammelten Daten und/oder von durch andere VDE **102x** gesammelten Daten konfiguriert werden. Eine VDE **102x** kann beispielsweise verschiedene Werkzeuge enthalten, die sowohl mit strukturierten Daten (z. B. Zeitreihen und im Speicher gespeicherte tabellarische Daten, relationale und/oder nicht-relationale Datenbanken, oder Gestreamtem) als auch mit unstrukturierten Daten (z. B. PDF-Dateien) arbeiten. Zusätzlich kann eine VDE **102x** jede beliebige einzelne oder mehrere erwünschte Zielumgebungen unterstützen und hierin entsprechenden Zielcode ausführen (z. B. Java, C#, R-Skripte, Python-Skripte, Matlab® -Skripte, Statgraphics etc.). Die VDE **102x** kann lernfähige Algorithmen ausführen (z. B. Partial Least Square Regression [Partielle Kleinste-Quadrate-Regression], Principal Component Analysis [Hauptkomponentenanalyse] etc.), Einordnungstechniken (z. B. Random Forest [Randomisierte Entscheidungsbäume], Mustererkennung etc.) und/oder sonstige Datenauswertungen, um Ergebnisse und/oder nützliche Informationen wie die Vorhersage von Produktkapazitäten, -eigenschaften und/oder sonstigen erwünschten Merkmalen zu erzeugen. Die Ergebnisse der lokalen Auswertung können im lokalen Speicher der Daten-Engine **102x** gespeichert werden und selbst als

zusätzliche Massendaten behandelt werden. Weiterhin können einige Daten-Engines **102x** Schnittstellen zur Konfigurierung und Entwicklung von Modellen, Laufzeit-Engines zur Ausführung von Modellen und zentrale Schaltstellen zur Anzeige von Ergebnissen in einer Benutzeroberfläche enthalten. So kann beispielsweise eine Verteilte-Daten-Engine **102x** für Auswertungen bei Laufzeit in einer Weise wie in der vorstehend genannten US-Anmeldung Nr. 62/060, 408 mit dem Titel „DATA PIPELINE FOR PROCESS CONTROL SYSTEM ANALYTICS“ beschrieben und/oder wie in einem späteren Abschnitt dieser Offenlegung beschrieben, konfiguriert werden. Angezeigte Ergebnisse können beschreibende Standardstatistiken, Histogramme, Beziehungszeichnungen und/oder sonstige Datendarstellungen, die eine implizite Beziehung innerhalb der verschiedenen Datensätze kennzeichnen, enthalten.

[0134] In einigen Fällen wird eine Verteilte-Daten-Engine **102x** in einer übergeordneten Datenquelle eingebettet, in dieser eingebaut, direkt mit ihr verbunden oder in sonstiger Weise mit ihr gleichzeitig vorhanden sein (z. B. die in **Fig. 2A** gezeigten Daten-Engines **102a**, **102b** und **102c**). In einigen Fällen kann eine Verteilte-Daten-Engine **102x** ein eigenständiger Massendaten-Knoten des Prozesssteuerungsdaten-Auswertungskommunikationsnetzwerks **112** sein (z. B. die in **Fig. 2A** gezeigten Daten-Engines **102d** und **102e**). Das heißt, in diesen Fällen, dass die Daten-Engine **102x** nicht in eine Datenquelle des Prozesssteuerungssystems oder der -anlage **5** eingebettet oder mit ihr gemeinsam vorhanden ist, sondern in sonstiger Weise Daten beobachten kann, die von einer oder mehreren Datenquellen erzeugt wurden (zum Beispiel, wenn eine Verteilte-Daten-Engine mit einer herkömmlichen Prozesssteuerungskommunikationsverbindung verbunden ist, so wie Daten-Engine **102c**). Unabhängig davon, ob eingebettet oder eigenständig, wertet die Daten-Engine **102x** Massendaten, die lokal erzeugt und/oder von einer oder mehreren Datenquellen zur Erkenntnisentdeckung oder -gewinnung bereitgestellt wurden, aus. Diese gewonnenen Erkenntnisse können in der Daten-Engine **102x** gespeichert werden, durch die Daten-Engine **102x** lokal bearbeitet werden und/oder als Massendaten anderen Daten-Engines **102x** bereitgestellt oder übermittelt werden, z. B. Massendaten-Empfänger-Knoten. Zusätzlich kann die Verteilte-Daten-Engine **102x** einen Teil ihrer bekannten oder gespeicherten Massendaten anderen Daten-Engines **102x** und/oder anderen Knoten des Auswertungsnetzwerks **112** bereitstellen (z. B. einem lokalen oder entfernten Benutzeroberflächenknoten).

[0135] **Fig. 3** enthält ein vereinfachtes Blockschema eines Beispiels einer verteilten industriellen Prozessperformanz-Datenüberwachung und/oder Datenauswertungs-Engine **150**, deren Instanzen in das Prozesssteuerungsdaten-Auswertungskommunikationsnetzwerk **112** aus **Fig. 2A** einbezogen werden können (z. B. die Verteilte-Daten-Engines **102x**). Unter Bezugnahme auf **Fig. 3** enthält das Verteilte-Daten-Engine-Beispiel **150** ein Massendaten-Speicherareal **155** zur Zwischenspeicherung, Speicherung und/oder Historisierung von Massendaten, einen oder mehrere Massendaten-Vorrichtungsempfänger **160** und einen oder mehrere Massendaten-Vorrichtungsanforderungszuführer **165**. Jeder der Massendaten-Vorrichtungsempfänger **160** wird für den Empfang und/oder die Beobachtung von Daten von einer oder mehreren Datenquellen **168** konfiguriert. In einem Beispiel empfängt und/oder beobachtet ein Massendaten-Vorrichtungsempfänger **160** über eine Netzwerkschnittstelle zu einem herkömmlichen Prozesssteuerungskommunikationsnetzwerk wie die Datennetzwerke **10**, ein Fieldbus-Netzwerk, ein WirelessHART-Netzwerk etc., Daten, die über das herkömmliche Prozesssteuerungskommunikationsnetzwerk laufen. Zusätzlich oder wahlweise kann der Massendaten-Vorrichtungsempfänger **160** Daten über eine lokale Schnittstelle von seiner entsprechenden Datenquelle/-n **168** empfangen, wie von einem Prozesssteuerungsgerät oder einem sonstigen Gerät, in das die VDE **150** eingebettet oder eingebaut wurde oder mit dem die VDE **150** lokal, unmittelbar und/oder in sonstiger Weise fest verbunden ist. Weiterhin zusätzlich oder wahlweise kann der Massendaten-Vorrichtungsempfänger **160** Massendaten-Pakete empfangen, z. B. über eine Datenauswertungs-Netzwerkschnittstelle **175**. Die empfangenen Massendaten-Pakete können von einer anderen VDE **150** gestreamt worden sein und/oder von einer Massendaten-Datenquelle, mit der die Daten-Engine **150** zusammen vorhanden ist, erzeugt worden sein. Unabhängig von der/den Quelle/-n **168** der empfangenen/beobachteten Daten verarbeitet der Massendaten-Vorrichtungsempfänger **160** die empfangenen/beobachteten Datenpakete und/oder Nachrichten, um die hierin enthaltenen substanziellen Daten und Zeitstempel abzurufen, und speichert die substanziellen Daten und Zeitstempel im Massendaten-Speicherareal **155** der Daten-Engine **150**, z. B. als Zeitreihendaten und wahlweise auch als Metadaten. Der hier verwendete Begriff „Metadaten“ bezieht sich im Allgemeinen auf Daten über Daten, wie statistische Informationen zu Daten, einordnende Informationen zu Daten, zusammenfassende Informationen, beschreibende Informationen, Definitionen etc. Das Massendaten-Speicherareal **155** kann mehrere lokale und/oder entfernte physische Datenlaufwerke oder Speichereinheiten umfassen, wie RAID-Speicher (Redundant Array of Independent Disks [Redundante Anordnung unabhängiger Festplatten]), Halbleiterspeicher, Cloud-Speicher, hochdichte Datenspeicher und/oder sonstige geeignete Datenspeichertechnologien, die als Datenbank- oder Datenzentrumsspeicher geeignet sind und die die Erscheinung eines einzelnen oder einheitlichen logischen Datenspeicherareals oder eine Einheit mit anderen

Knoten aufweisen und die für eine lokale Speicherung und/oder Historisierung von Massendaten konfiguriert werden können.

[0136] Jeder der Massendaten-Vorrichtungsanforderungszuführer **165** wird für den Zugriff auf in dem Massendaten-Vorrichtungsspeicherareal **155** gespeicherten Zeitreihendaten und/oder Metadaten konfiguriert, z. B. durch Anforderung der anforderten Einheit oder Vorrichtung wie eine lokale oder entfernte Datenanalyseanwendung, eine Benutzeroberflächenanwendung oder eine sonstige Anwendung. So kann beispielsweise ein Massendaten-Vorrichtungsanforderungszuführer **165** im Massendaten-Vorrichtungsspeicherareal **155** gespeicherte ausgewählte Daten in festgelegten Zeitabständen an Abonnenten der ausgewählten Daten veröffentlichen. In einem anderen Beispiel kann ein Massendaten-Vorrichtungsanforderungszuführer **165** in einem lokalen Massendaten-Vorrichtungsspeicherareal **155** auf Anforderung durch eine entfernte Datenauswertungsausführungsanwendung Daten abrufen. Weitere Beispiele für den Zugriff auf in einer VDE **150** gespeicherte Daten werden in späteren Abschnitten beschrieben. Derartige Daten können in einer Vielzahl von Datenformaten gespeichert werden, einschließlich relationaler und nicht-relationaler Datenbanken oder sonstigen Datenstrukturen. In einigen Ausführungsformen kann ein standardisiertes Abfrageformat für den Zugriff auf in beliebigen dieser Datenquellen enthaltene Daten verwendet werden.

[0137] In einigen Ausführungsformen enthält eine Verteilte-Daten-Engine **150** eine oder mehrere Massendaten-Auswerter **170** zur Durchführung entsprechender Datenauswertungen und/oder zum Erlernen von zumindest Teilen der im Speicher **155** gespeicherten Massendaten. Die Ausführung lokaler Auswertungen und/oder des Erlernens kann infolge eines von einem Benutzer oder eines sonstigen Knotens erzeugten Befehls oder einer Anweisung erfolgen. Zusätzlich oder wahlweise kann die Ausführung der lokalen Auswertungen und/oder des Erlernens in automatischer und/oder autonomer Weise ohne weitere Dateneingaben eines Benutzers oder sonstigen Knotens zur Einleitung und/oder Ausführung der lernenden Auswertung erfolgen. Beispielsweise kann die Datenauswertung oder das Erlernen in einer bereits beschriebenen Weise erfolgen, in einer wie in der vorstehend genannten US-Anmeldung Nr. 62/060,408 mit dem Titel „DATA PIPELINE FOR PROCESS CONTROL SYSTEM ANALYTICS“ beschriebenen Weise oder in einer sonstigen geeigneten Weise erfolgen. In einer Ausführungsform führen die Massendaten-Auswerter **170** einzeln oder gemeinschaftlich umfangreiche Datenauswertungen der gespeicherten Daten durch (z. B. Datenextrahierung, Datenentdeckung etc.), um neue Informationen oder Erkenntnisse zu entdecken, erkennen oder zu gewinnen. Eine Datenextrahierung umfasst im Allgemei-

nen den Vorgang der Prüfung großer Datenmengen, um neue oder bislang unbekannte interessante Daten oder Muster wie ungewöhnliche Datensätze oder mehrfache Gruppen von Datensätzen zu extrahieren. Die Massendaten-Auswerter **170** können ebenfalls umfangreiche Datenauswertungen von gespeicherten Daten durchführen (z. B. maschinelle Lernauswertung, Datenmodellierung, Mustererkennung, vorhersagende Auswertung, Beziehungsauswertung etc.), um implizite Beziehungen oder Rückschlüsse innerhalb der gespeicherten Daten vorherzusagen, zu berechnen oder zu erkennen.

[0138] In einer Ausführungsform können mehrfache Massendaten-Auswerter **170** (und/oder mehrfache Instanzen mindestens eines Massendaten-Auswerts **170**) parallel und/oder gemeinsam betrieben werden, um im Massendaten-Speicherareal **155** der Verteilte-Daten-Engine **150** gespeicherte Daten auszuwerten und/oder in einem oder mehreren anderen Massendaten-Speicherarealen anderer Verteilte-Daten-Engines **102x** gespeicherte Daten auszuwerten. Weiterhin können mehrfache Massendaten-Auswerter **170** berechnete Parameter und Modellinformationen als Art und Weise einer gemeinschaftlichen Datenauswertung und Erlernens miteinander teilen, austauschen oder untereinander übertragen. Die mehrfachen Massendaten-Auswerter **170** können mit verschiedenen Massendaten-Knoten gemeinsam vorhanden sein oder mit verschiedenen Massendaten-Knoten vorhanden sein. Ein Beispiel einer mit einer oder sämtlichen hier beschriebenen Techniken verwendbaren gemeinschaftlichen Datenauswertung findet sich in der vorstehend genannten US-Anmeldung Nr. 62/060,408 mit dem Titel „DATA PIPELINE FOR PROCESS CONTROL SYSTEM ANALYTICS“, obwohl jede geeignete gemeinschaftliche Datenauswertungstechnik oder -techniken mit einer oder sämtlichen Merkmalen dieser Offenlegung verwendet werden können. Ergebnisse der von den Massendaten-Auswertern **170** durchgeführten Auswertungen können im Massendaten-Vorrichtungsspeicherareal **155** und/oder an eine anfordernde Einheit oder Anwendung zurückgegeben werden.

[0139] In einer Ausführungsform ist zumindest ein Teil der Massendaten-Empfänger **160**, der Massendaten-Vorrichtungsanforderungszuführer **165** und/oder Massendaten-Auswerter **170** enthalten oder in einer oder mehreren integrierten Schaltkreisen, Halbleitern, Chips oder sonstiger geeigneter Hardware eingebaut. Beispielsweise kann ein Spektralanalysen durchführender Massendaten-Auswerter **170** durch einen in einem Massendaten-Knoten integrierten Schaltkreischip umgesetzt werden, wie in der vorstehend genannten US-Anmeldung 14/507,252 mit dem Titel „AUTOMATIC SIGNAL PROCESSING-BASED LEARNING IN A PROCESS PLANT“ beschrieben. In einer Ausführungsform umfasst zumindest ein Teil

der Massendaten-Empfänger **160**, der Massendaten-Vorrichtungsanforderungszuführer **165** und/oder der Massendaten-Auswerter **170** in einem Speicher gespeicherte computerausführbare und durch einen in einer Verteilte-Daten-Engine laufenden Prozess ausführbare Anweisungen. Beispielsweise umfassen mindestens einige Teile der Massendaten-Anwendungsempfänger **160**, der Massendaten-Vorrichtungsanforderungszuführer **165** und/oder der Massendaten-Vorrichtungsauswerter **170** entsprechende in einem oder mehreren nicht-flüchtigen, dinghaften Speichern oder Datenspeichergeräten gespeicherte computerausführbare Anweisungen und können durch einen oder mehrere Prozessoren ausgeführt werden, um eine oder mehrere der entsprechenden Massendaten-Funktionen durchzuführen.

[0140] In einigen Ausführungsformen sind zumindest einige der Massendaten-Auswerter **170** nicht in einer Verteilte-Daten-Engine **150** enthalten, sondern sind gemeinsam mit der Verteilte-Daten-Engine **150** auf demselben übergeordneten Datenquellengerät oder Komponente und in einer Kommunikationsverbindung mit der Daten-Engine **150** vorhanden. Beispielsweise kann die Daten-Engine **150**, einschließlich des Speicherareals **155**, Empfängers **160** und Zuführers **165** durch einen ersten Satz computerausführbarer Anweisungen umgesetzt werden, und die Massendaten-Auswerter **170** können durch einen Halbleiter-Chip oder einen zweiten Satz computerausführbarer Anweisungen umgesetzt werden, die in demselben nicht-flüchtigen, dinghaften Speicher oder Datenspeichergerät wie der erste Satz computerausführbarer Anweisungen gespeichert sein können, aber nicht müssen. In einigen Ausführungsformen sind die Massendaten-Auswerter **170** nicht in einer Verteilte-Daten-Engine **150** enthalten, und sind nicht gemeinsam mit der Daten-Engine **150** auf demselben übergeordneten Datenquellengerät oder Komponente vorhanden, haben aber dennoch eine Kommunikationsverbindung zur Daten-Engine **150**. Beispielsweise kann sich der Massendaten-Auswerter **170** in einem anderen Knoten des Datenauswertungsnetzwerks **112** befinden, welcher keine VDE **150** ist, z. B. eine Benutzeroberfläche oder ein Auswertungsserver.

[0141] Die in **Fig. 3** gezeigte Verteilte-Daten-Engine **150** enthält ein oder mehrere Netzwerkschnittstellen **175**, die so konfiguriert sind, dass sie der Daten-Engine **150** die Übermittlung und den Empfang von Massendaten-Last über das Datenauswertungsnetzwerk **112** sowie die Kommunikation mit anderen Daten-Engines und Knoten des Datenauswertungsnetzwerks **112** ermöglichen (z. B. Meldungen und andere Arten der Kommunikation). Beispielsweise kann eine Daten-Engine **150** eine Art der von einem anderen Knoten veröffentlichten Daten durch Nutzung der Netzwerkschnittstelle **175** abonnieren, und die Art der Daten, die durch den Veröffentlichungsknoten er-

zeugt wird und die von der Daten-Engine **150** abonniert wurde, kann über die Netzwerkschnittstelle **175** empfangen werden.

[0142] Wie bereits beschrieben können verschiedene Arten von Echtzeitdaten, wie prozessbezogene Daten, anlagenbezogene Daten und andere Arten von Daten durch die Verteilte-Daten-Engines **102x** erzeugt, gesammelt, beobachtet, abgerufen, empfangen, gespeichert, zwischengespeichert, verarbeitet, ausgewertet und/oder gestreamt werden. Beispiele prozessbezogener Daten sind unter anderem fortlaufende, Stapelverarbeitungs-, Mess- und Ereignisdaten, die erzeugt werden, während ein Prozess in der Prozessanlage **5** gesteuert wird (und, in einigen Fällen, auf einen Effekt einer Echtzeitausführung des Prozesses hinweisen). Weiterhin können prozessbezogene Daten unter anderem Definitionen, Anordnungs- oder Einstellungsdaten wie Konfigurationsdaten und/oder Stapelverarbeitungsrezepturdaten und Einstellungen, Ausführung und Ergebnisse der Prozessdiagnose entsprechende Daten etc. umfassen.

[0143] Anlagenbezogene Daten, wie Daten in Bezug auf die Prozessanlage **5**, die jedoch nicht durch unmittelbar einen Prozess in der Prozessanlage **5** konfigurierende, steuernde oder diagnostizierende Anwendungen erzeugt werden können, können als Massendaten durch die Verteilte-Daten-Engines **102x** erzeugt, gesammelt, beobachtet, abgerufen, empfangen, gespeichert, zwischengespeichert, verarbeitet, ausgewertet und/oder gestreamt werden. Beispiele von anlagenbezogenen Daten sind unter anderem Erschütterungsdaten, Kondensatdaten, Daten, die auf einen Wert eines Parameters in Bezug auf Anlagensicherheit hinweisen (z. B. Korrosionsdaten, Gaserkennungsdaten etc.), Daten, die auf ein Ereignis in Bezug auf Anlagensicherheit hinweisen, Daten in Bezug auf Maschinengesundheit, Daten in Bezug auf Vermögenswerte in der Anlage wie Anlagenausstattung und/oder -geräte, Daten in Bezug auf Konfiguration, Ausführung und Ergebnisse von Ausstattungsgegenständen, Maschinen und/oder Gerätediagnose und Daten, die nützlich für Diagnose und Prognose sind.

[0144] Weiterhin können andere Arten von Daten einschließlich Datenausbauverkehrs- und Netzwerkmanagementdaten in Bezug auf die Prozesssteuerungs-Massendaten-Netzwerkhauptleitung und verschiedenen Kommunikationsnetzwerke der Prozessanlage **5**, nutzerbezogene Daten wie Daten in Bezug auf Nutzerverkehr, Einwahlversuche, Abfragen und Anweisungen, Textdaten (z. B. Journale, Bedienungsverfahren, Handbücher etc.), Raumdaten (z. B. ortsbasierte Daten), und Multimedia-Daten (z. B. Videoüberwachungssysteme, Videos etc.) durch die Daten-Engines **102x** als Massendaten erzeugt, gesammelt, beobachtet, abgerufen, empfangen, ge-

speichert, zwischengespeichert, verarbeitet, ausgewertet und/oder gestreamt werden.

[0145] In einigen Ausführungsformen können dynamische Mess- und Steuerungsdaten durch die Verteilte-Daten-Engines **102x** automatisch als Massendaten erzeugt, gesammelt, beobachtet, abgerufen, empfangen, gespeichert, zwischengespeichert, verarbeitet, ausgewertet und/oder gestreamt werden. Beispiele dynamischer Mess- und Steuerungsdaten sind unter anderem Daten, die Änderungen eines Prozessverfahrens bezeichnen, Daten, die Bedienparameter wie Sollwerte bezeichnen, Aufzeichnungen von Prozess- und Hardwarealarmen und Ereignissen wie Fehlern beim Herunterladen oder der Kommunikation. Zusätzlich können statische Daten wie Controller-Konfigurationen, Stapelverarbeitungsrezepturen, Alarne und Ereignisse standardmäßig automatisch gesammelt werden, sobald eine Änderung erkannt wird oder sobald ein Controller oder eine andere Einheit dem Datenauswertungskommunikationsnetzwerk **112** zum ersten Mal hinzugefügt wird.

[0146] Außerdem werden in einigen Szenarien zumindest einige statische Metadaten, die dynamische Steuerungs- und Messdaten beschreiben oder bezeichnen in den Verteilte-Daten-Engines **102x** erfasst, sobald eine Änderung der Metadaten erkannt wird. Wenn beispielsweise eine Änderung der Controller-Konfiguration vorgenommen wird, die durch den Controller zu versendende Mess- und Steuerungsdaten in Modulen oder Einheiten beeinflussen, so wird eine Aktualisierung der entsprechenden Metadaten automatisch von den Daten-Engines **102x** erfasst. Zusätzlich oder wahlweise können zu bestimmten Modulen gehörende zur Pufferung von Daten von externen Systemen oder Quellen (z. B. Wettervorhersagen, öffentliche Veranstaltungen, unternehmerische Entscheidungen etc.) verwendete Parameter, Überwachungsdaten und/oder sonstige Arten von Überwachungsdaten automatisch von den Daten-Engines **102x** erfasst werden.

[0147] In einigen Situationen werden vom Endanwender erstellte hinzugefügte Parameter automatisch in den Verteilte-Daten-Engines **102x** erfasst. Beispielsweise kann ein Endanwender bestimmte Berechnungen in einem Modul erstellen oder einer Einheit einen zu sammelnden Parameter hinzufügen, oder der Endanwender könnte einen Standard-Controller-Diagnose-Parameter sammeln wollen, der nicht standardmäßig übermittelt wird. Parameter, die der Endanwender wahlweise konfiguriert, können in derselben Weise übermittelt werden wie Standardparameter.

BENUTZEROBERFLÄCHENANWENDUNG FÜR INDUSTRIELLE LEISTUNGSÜBERWACHUNG/-AUSWERTUNG

[0148] Wie bereits erwähnt kann das Datenauswertungssystem (DAS) **100** eine oder mehrere Benutzeroberflächenanwendungen enthalten, über die datenbezogene Mensch-Maschine-Interaktionen vorgenommen werden. Das Vorhandensein von Beispieldateninstanzen solcher Benutzeranwendungen wird in **Fig. 2A** durch Bezugszeichen **120a–120d** dargestellt. Der aktuelle Abschnitt dieser Offenlegung und die **Fig. 4A–Fig. 4Q** beschreiben die Benutzeroberflächenanwendung für industrielle Leistungsüberwachung/-auswertung näher, die beispielsweise durch die Prozessanlage oder das -system **5** in **Fig. 1**, das industrielle Leistungsüberwachungs-/auswertungssystem **100** in **Fig. 2A** und/oder die Überwachungs- und Auswertungs-Engine **150** in **Fig. 3** bereitgestellt oder in Verbindung mit diesen betrieben wird. Die hier beschriebene Benutzeroberflächenanwendung für industrielle Leistungsüberwachung/-auswertung kann jedoch von anderen industriellen Leistungsüberwachungs- und -auswertungssystemen für Prozesssteuerungssysteme bereitgestellt oder in Verbindung mit diesen betrieben werden. Zur Erleichterung der Beschreibung wird die Benutzeroberflächenanwendung für Industrielle Leistungsüberwachung/-auswertung untenstehend unter gleichzeitiger Bezugnahme auf **Fig. 1**, **Fig. 2**, **Fig. 3** beschrieben. Weiterhin wird zur besseren Lesbarkeit die Benutzeroberflächenanwendung für Industrielle Leistungsüberwachung/-auswertung hier großgeschrieben, um sie von allgemeinen und/oder sonstigen Benutzeroberflächenanwendungen zu unterscheiden, und wird hier auch wechselweise als „Benutzeranwendung für Datenauswertung“, „Benutzeroberflächenanwendung für Datenauswertung“, „VDE-Benutzeroberflächenanwendung“ oder „Benutzeroberflächenanwendung“ bezeichnet.

[0149] Die VDE-Benutzeroberflächenanwendung bietet eine Oberfläche zur Interaktion eines Benutzers mit dem verteilten industriellen Prozessleistungsüberwachungs-/auswertungssystem (DAS) **100**, um Struktur und Abfragedaten festzulegen und Entwurf-Daten-Modelle zu erstellen und zu bewerten. Nach Fertigstellung der Entwurf-Daten-Modelle ermöglicht die VDE-Benutzeroberflächenanwendung das Herunterladen der Daten-Modelle in eine Laufzeit-Engine und den Einsatz für den Betrieb in Verbindung mit einem Online-Prozesssteuerungssystem. Auf ein eingesetztes Daten-Modell (auch als Ausführungs- oder Online-Daten-Modul bezeichnet) kann über eine zentrale Laufzeit-Schaltstelle der VDE-Benutzeroberflächenanwendung zugegriffen und eine Überwachung erfolgen. Die VDE-Benutzeroberflächenanwendung ist ebenfalls in der Lage, Alarne und Meldungen zu erzeugen, die den ausgeführten Daten-Modellen entsprechen.

[0150] Die VDE-Benutzeroberflächenanwendung ermöglicht dem Benutzer ausdrücklich, Daten-Modelle zu erstellen, zu betrachten und zu ändern, wobei jedes eine oder mehrere Datenauswertungen beschreibt (z. B. beschreibende, vorhersagende und/oder vorschreibende Auswertungen), die in einem Eingabedatensatz ausgeführt oder durchgeführt werden. Daten-Modelle werden in einem Offline-Modus entworfen oder erstellt (z. B. während das Daten-Modell von Echtdatenquellen in einem Online- oder Betriebsprozesssteuerungssystem losgelöst ist), wobei in dieser Anmeldung ein in diesem Modus befindliches Daten-Modell als „Offline-Daten-Modul“ bezeichnet wird. Im Allgemeinen wird ein Offline-Daten-Modul mittels eines vom Benutzer mittels der VDE-Benutzeroberflächenanwendung erstellten Offline-Datendiagramms durch Auswahl eines Satzes von „Blöcken“ oder „Daten-Blöcken“ und Verbindung der einzelnen Blöcke untereinander in der gewünschten Weise mit einem Satz „Drähten“ im Diagramm festgelegt. Jeder Daten-Block enthält eine Dateneingabe mittels im Block empfangener Daten. Jeder Daten-Block steht auch für oder kennzeichnet eine bestimmte Funktion, Aktion, Algorithmus und/oder Bedienung, die von jedem Daten-Block mit dessen Eingabedaten durchzuführen ist, wodurch Ausgabedaten erzeugt werden, die anderen Blöcken mittels einer oder mehrerer Datenausgaben bereitgestellt werden können. Jeder Daten-Block kann eigenständig bewertet werden, sodass die Offline-Diagramm-Darstellung des Blocks und seiner Verbindungsdrähte in ausführbaren Code kompiliert und ausgeführt wird, wobei die Ergebnisse der Ausführung der einzelnen Daten-Blöcke in dem Offline-Datendiagramm angezeigt werden. Wie in einem späteren Abschnitt beschrieben wird, kann die Kompilierung eines Daten-Blocks zu Code und die darauffolgende Ausführung verteilt über verschiedene Zielumgebungen und Orte erfolgen.

[0151] In ähnlicher Weise kann auch das Offline-Datendiagramm als Ganzes bewertet werden. Die Bewertung eines Offline-Datendiagramms besteht aus der Kompilierung der Daten-Blöcke und Drähte, Ausführung des kompilierten Codes zur Übermittlung von Eingabe- und Ausgabedaten über die Drähte und Durchführung von Funktionen, Aktionen, Algorithmen und/oder Vorgänge wie durch die jeweilige Konfiguration der Blöcke und Drähte des Offline-Datendiagramms festgelegt. Ebenfalls in ähnlicher Weise wie einzelne Daten-Blöcke kann auch die Kompilierung des Offline-Datendiagramms verteilt über verschiedene Zielumgebungen und -orte hinweg erfolgen.

[0152] Weiterhin ermöglicht die VDE-Benutzeroberflächenanwendung dem Benutzer, ein Offline-Daten-Modul in ein „Online-Daten-Modul“ umzuwandeln, sodass sich das Online-Daten-Modul des Daten-Modells anbindet an oder in sonstiger Weise von Echtdatenquellen erzeugte Echtdaten (z. B. Streaming)

des Online-Prozesssteuerungssystems empfängt, eine oder mehrere dort festgelegte Datenauswertungen der Echtdaten durchführt und die Datenausgabe der Benutzeroberfläche, dem Historiker oder sonstigen Anwendungen bereitstellt. Beispielsweise können die von einem Daten-Modell erzeugten Ausgabedaten beschreibende, vorhersagende und/oder vorschreibende, zur Prozessanlage und/oder einem dort gesteuerten Prozess gehörende Informationen oder Daten enthalten.

[0153] Insbesondere kann ein Benutzer ein Online-Datendiagramm eines bestimmten Daten-Modells in ein Online-Datendiagramm des bestimmten Daten-Modells umwandeln. Wie ein Offline-Datendiagramm enthält ein Online-Datendiagramm einen Satz durch einen Satz Drähte verbundene Daten-Blöcke, und da die Offline-Datendiagramme und Online-Datendiagramme im Allgemeinen demselben Daten-Modell entsprechen, entsprechen die durch das Online-Datendiagramm festgelegten Funktionen, Aktionen, Algorithmen und/oder Vorgänge denen des Offline-Datendiagramms. Zumindest einige der Blöcke und Verbindungen des Offline-Datendiagramms unterscheiden sich jedoch vom Online-Datendiagramm, in erster Linie (jedoch nicht unbedingt ausschließlich), um die Verbindung des Online-Daten-Modul zur Online-Prozessanlage zu ermöglichen.

[0154] Ähnlich wie Offline-Datendiagramme können Online-Datendiagramme ebenso als Ganzes in dem Online-Daten-Modul des Daten-Modells entsprechenden ausführbaren Code kompiliert werden. Die Kompilierung bestimmter Blöcke eines Online-Datendiagramms löst die Festlegung der Bindungen des jeweiligen Blocks an entsprechende Datenquellen und Datenverbraucher innerhalb der Online-Prozessanlage aus. Der Einsatz eines Online-Daten-Moduls instanziert diese Bindungen und löst den Ablauf oder die Ausführung des ausführbaren Codes aus, um dadurch das Online-Daten-Modul in die Online-Prozessanlage zu integrieren, damit das Online-Daten-Modul in Verbindung mit dem Betrieb der Online-Prozessanlage ausgeführt wird. Die Kompilierung eines Online-Datendiagramms und die Bindung und Ausführung des sich ergebenden Online-Daten-Moduls kann verteilt über verschiedene Zielumgebungen und Orte hinweg erfolgen.

[0155] In einer bestimmten nützlichen Ausführungsform empfängt und wird das Online-Daten-Modul zumindest teilweise mit fortlaufenden Echtzeitdaten betrieben, die durch Geräte oder Komponenten der Prozessanlage als Ergebnis einer fortwährenden Steuerung einer Ausführung eines Online-Prozesses in der Anlage erzeugt werden. Beispielsweise arbeitet das Online-Daten-Modul mit fortlaufenden durch die Online-Prozessanlage und/oder durch ein in der Anlage enthaltenes Prozesssteuerungssystem erzeugten Echtzeit-Zeitreihendaten, während die Prozessanla-

ge und das Prozesssteuerungssystem den Prozess steuern. Das Daten-Modul arbeitet fortwährend mit dem fortlaufenden Echtzeit-Datenstrom (z. B. durch Durchführung einer oder mehrerer seiner Datenauswertungsfunktionen und sonstiger Funktionen (sofern vorhanden), wie durch das Daten-Modell festgelegt), und erzeugt fortwährend einen Echtzeitstrom an Ergebnissen oder Ausgabedaten, die in einer Benutzeroberfläche angezeigt werden (z. B. als rollendes Liniendiagramm oder eine andere Darstellung) und fortwährend die aktuellen Echtzeitwerte der Datenausgabe des Daten-Modells wiedergeben können. In einem Beispiel enthält die von dem Daten-Modell erzeugte und in der Benutzeroberfläche angezeigte Datenausgabe mindestens einen Vorhersagewert und dessen zeitlichen Verlauf. Im Allgemeinen sind Daten-Modelle jedoch in der Lage, mit großen, von verschiedenen Datenquellen erzeugten Datenmengen innerhalb der Prozesssteuerungsanlage oder – umgebung 5 zum Zwecke der Fehlererkennung, Vorhersage und Vorschreibung für die Prozesssteuerungsanlage oder -umgebung 5 zu arbeiten.

[0156] In einer Ausführungsform ist die VDE-Benutzeroberflächenanwendung web-basiert, wobei der Zugriff über einen Internetbrowser erfolgt, sodass die verschiedenen Instanzen der Anwendung von verschiedenen Plattformen (z. B. Apple Macintosh, Linux, Windows etc.) und verschiedenen Benutzern von verschiedenen Rechnern, unter Umständen auch gleichzeitig, genutzt werden können. Die VDE-Benutzeroberflächenanwendung ist jedoch nicht auf eine web-basierte Umsetzung beschränkt und kann auf beliebige geeignete Weisen umgesetzt werden, die plattformunabhängig sind und auf mehrere gleichzeitige Benutzer und/oder Instanzen erweitert werden können.

[0157] Weiterhin dient diese Konfiguration nur der Veranschaulichung und soll nicht als beschränkend ausgelegt werden, obwohl die hier als auf eine einzelne Prozesssteuerungsanlage oder -umgebung 5 mit mehreren gleichzeitig ausgeführten Instanzen der VDE-Benutzeroberflächenanwendung anzuwendende VDE-Benutzeroberflächenanwendung beschrieben wird. Beispielsweise kann in einigen Konfigurationen eine VDE-Benutzeroberflächenanwendung auf verschiedene mehrfache Prozesssteuerungsanlagen oder -umgebungen angewendet werden, die unabhängig voneinander an verschiedenen Orten mit verschiedenen Prozessen arbeiten. Beispielsweise kann eine einzelne durch eine Reihe von Servern oder Computern bereitgestellte VDE-Benutzeroberflächenanwendung von mehreren Ölraffinerien eines Mineralölunternehmens genutzt werden, sodass einige Instanzen der VDE-Benutzeroberflächenanwendung in Raffinerie A und einige Instanzen in Raffinerie B ausgeführt werden.

[0158] Jedenfalls abstrahiert die VDE-Benutzeroberflächenanwendung, wie bereits beschrieben, die Funktionen, Aktionen, Algorithmen und/oder Vorgänge, die von Daten-Blöcken mit entsprechenden Eingabedaten als Blöcke oder Daten-Blöcke durchgeführt werden. Beispiele für Blöcke oder Daten-Blöcke werden durch ihre jeweilige Funktion benannt, z. B. Laden, Bereinigen, Verändern, Auswerten, Bildlich Darstellen etc. Jeder Daten-Block kann entsprechende Eingabedaten laden oder erlangen, eine oder mehrere seiner Funktionen, Aktionen, Algorithmen und/oder Vorgänge an den erlangten Eingabedaten durchführen und eine oder mehrere entsprechende Ergebnisse oder entsprechende Ausgabedaten erzeugen. Mehrfache Daten-Blöcke können je nach Wunsch miteinander verbunden werden (z. B. seriell, parallel, n:1, 1:n etc.), um ein Offline-Datendiagramm zu bilden, das ein Entwurfsmodell oder Offline-Daten-Modul wiedergibt oder festlegt, wobei dieses Entwurf-Daten-Modell/Offline-Daten-Modul durch Auslösen der Anwendung des Offline-Daten-Moduls (oder eines Teils davon) auf oder der Bearbeitung von einem Satzes eines oder mehrerer Datensätze oder Offline-Datenquellen ausgeführt oder bewertet werden kann. Beispielsweise wird während der Ausführung oder Bewertung eines Offline-Daten-Moduls das Offline-Datendiagramm zu ausführbarem Code kompiliert, ein vorgegebene Datensatz oder -sätze erlangt und in das Offline-Modul geladen oder durch dieses genutzt, wobei die miteinander verbundenen Blöcke des kompilierten Offline-Moduls jeweils ausgeführt werden, um ihre jeweiligen Vorgänge an ihren jeweiligen Dateneingaben durchzuführen und die entsprechenden Ergebnisse ihren jeweiligen Datenausgaben bereitzustellen, was zur Erzeugung einer oder mehrerer Auswertungsausgaben oder -ergebnisse durch sämtliche Offline-Daten-Module entsprechend dem grade entwickelten oder erstellten Daten-Modell führt. In einer Ausführungsform können während der Offline-Bewertung, anstatt Datensätze aus einer Datei in das Entwurf-Daten-Modell zu laden, Eingabedaten aus einer streamenden Offline-Datenquelle empfangen werden, wie einem Testgerät, umgebung oder -anlage. Weiterhin kann, wie später noch näher erläutert wird, ein Offline-Daten-Modul oder Entwurf-Daten-Modell noch während dessen Entwicklung auch schrittweise bewertet werden. In der VDE-Benutzeroberflächenanwendung ist die Plattform oder Infrastruktur für die Bewertung von Offline-Daten-Blöcken und Offline-Daten-Modulen die Datenauswertungsdienste, die in einem späteren Abschnitt beschrieben werden.

[0159] Nachdem ein Offline-Daten-Modul (oder ein Teil dessen) bewertet wurde, können die Ergebnisse der Bewertung vom Benutzer überprüft und untersucht und das Entwurfsmodell (oder ein Teil dessen) solange entsprechend verändert und erneut bewertet werden, bis ein zufriedenstellendes Ergebnis erzielt wird. Ein Benutzer kann ein Entwurf-Daten-

Modell/Offline-Daten-Modul fertigstellen, in ein Online-Daten-Modul umwandeln und das Online-Daten-Modul ausführen oder einsetzen, um mit Echtzeit-, Online oder gestreamten Echtdaten, die aufgrund von Echtzeitvorgängen der Prozesssteuerungsumgebung oder -anlage 5 erzeugt werden, zu arbeiten. Beispielsweise kann ein ausgeführtes oder eingesetztes Online-Daten-Modul oder Daten-Modell fortwährend mit Echtzeitdaten arbeiten, die fortwährend aufgrund von Online-Vorgängen der Prozessanlage erzeugt werden, und das Online-Daten-Modul oder -modell kann selbst Echtzeit-Ausgabedaten oder Ergebnisse seiner Ausführung erzeugen, die in einer Benutzeroberfläche angezeigt und fortwährend aktualisiert werden. Die Online-Ausführung oder der Betrieb des eingesetzten Online-Daten-Moduls und der entsprechenden Ergebnisse können an einer zentralen Schaltstelle der VDE-Benutzeroberflächenanwendung überwacht werden, die ebenfalls in einem späteren Abschnitt näher beschrieben wird.

[0160] Wie bereits beschrieben können innerhalb der VDE-Benutzeroberflächenanwendung, Daten-Blöcke, Daten-Module (unabhängig davon ob Online oder Offline) und Teile davon einem Benutzer bildlich und/oder grafisch als Diagramme oder Daten-diagramme auf einem von der VDE-Benutzeroberflächenanwendung bereitgestellten Arbeitsbereich oder einer Leinwand dargestellt werden. Im Allgemeinen enthält jedes Datendiagramm einen Satz miteinander durch einen Satz Drähte (die z. B. durch Linien dargestellt werden können) verbundener Daten-Blöcke (die z. B. doch zweidimensionale Formen dar gestellt werden können). Die VDE-Benutzeroberflächenanwendung enthält üblicherweise zwei Betriebs-modi (obwohl in einigen Ausführungsformen eine größere oder kleinere Anzahl an Modi enthalten sein kann), die jeweils Datendiagramme zur Darstellung von Daten-Modulen und den in den Daten-Modulen enthaltenen Daten-Blöcken verwenden.

[0161] Einer dieser Betriebsmodi wird hier wechselweise sowohl als „Datenauswertungsstudio“, „Daten-studio“ oder als „Studio“ bezeichnet. Üblicherweise wird das Datenstudio von den Benutzern zur Gestaltung und Entwicklung von Datenauswertungsmodel len verwendet. Zusätzlich ermöglicht das Datenstu dio einem Benutzer, ein Offline-Modul in ein Online-Modul umzuwandeln sowie ein fertiges Offline-Modul als entsprechendes Online-Daten-Modul einzusetzen. Das Datenstudio stellt einen Arbeitsbereich oder eine Zeichenleinwand bereit, auf der der Benutzer die Möglichkeit hat, ein Daten-Modell durch Erstellung und Verbindung von Blöcken zu einem Da-ten-diagramm zu entwickeln, welches eine bildliche, grafische Darstellung eines Daten-Modells oder ei-nes Teils davon bildet. Während der Entwicklung ei-nes Daten-Modells innerhalb des Datenstudios kann es als Entwurf-Daten-Modell oder Offline-Daten-Modul bezeichnet und als Entwurf oder Prototyp betrach-

tet werden. Ein Offline-Daten-Modul-Diagramm kann eine größere Anzahl Daten-Blöcke und/oder Verbindungen enthalten als das sich daraus ergebende On-line-Daten-Modul, da einige im Offline-Modul enthaltenen Blöcke und Verbindungen zur Auswertung und Betrachtung von Daten in verschiedenen Teilen des Moduls genutzt werden können, z. B. zum Zwecke der Auswertung und Prüfung, ob ein bestimmter Teil des Daten-Modells wie gewünscht bewertet und/oder hinreichend nützliche und/oder vorhersagende Da-ten bereitstellt. Im Allgemeinen ist ein Offline-Dia-gramm eines Daten-Moduls eine Darstellung eines Ablaufs, der (1) Rohdatensätze erkundet und berei-nigt und (2) für die gewünschten Auswertungsvor-gänge gebaut, eingestellt und bewertet werden kann, wie Einordnung, Regression, Clustering, Dimensionsreduzierungen etc. Daten-Module, die offline sind, kön-nen während ihrer Entwicklung schrittweise oder for-tlaufend bewertet werden. Die Ergebnisse des Bewer-tungsprozesses eines Offline-Daten-Moduls oder Da-ten-Modells werden dem Benutzer mittels des Daten-studios angezeigt.

[0162] In einigen Fällen kann die Ausführung oder Bewertung eines Offline-Daten-Moduls eine längere Zeit bis zur Fertigstellung in Anspruch nehmen. In diesen Situationen kann der Status und der Fort-schritt des Offline-Daten-Modul-Bewertungsprozes-ses über einen anderen Betriebsmodus der VDE-Benutzeroberflächenanwendung übertragen werden, der hier wechselweise als „zentrale Datenauswer-tungsschaltstelle“, „zentrale Auswertungsschaltstel-le“ oder einfach nur „zentrale Schaltstelle“ bezeichnet wird. Die zentrale Schaltstelle wird üblicherweise vom Benutzer zur Anzeige und/oder Verwaltung von Mo-dulen, die (i) Offline-Daten-Module (z. B. Entwurf-Da-ten-Module) in der Bewertungsphase sind und/oder (ii) Online-Daten-Module (z. B. fertige Daten-Modelle, die als Daten-Module eingesetzt wurden) sind, ver-wendet. Die zentrale Schaltstelle stellt ebenfalls Ent-wurf-Daten-Modelle während der Bewertung und ei-gesetzte Online-Daten-Modul mithilfe entsprechen-der Diagramme oder bildlicher, grafischer Darstel-lung, dar. Ein eingesetztes Daten-Modul wird als „On-line“ seiend bezeichnet, da das eingesetzte Modul mit Echtzeitdaten ausgeführt wird, die aufgrund von Online- oder Laufzeitvorgängen des Prozesssteue-rungssystems oder der -anlage 5 erzeugt werden. Im Allgemeinen ist ein Online-Diagramm eines Daten-Moduls eine Darstellung eines Ablaufs, der an eine oder mehrere Datenquellen innerhalb des Prozess-steuerungssystems oder der -anlage 5 (z. B. strea-mende Datenquelle) gebunden ist, um Echtzeitbe-schreibungen, Vorhersagen und/oder Vorschriften zu machen und/oder fortwährend Daten-Modelle wäh-rend oder bei Laufzeit einzustellen. Daten-Module, die eingesetzt wurden oder online sind, können so-lange ausgeführt werden, bis sie ausdrücklich von der zentralen Schaltstelle aus beendet werden.

A. Daten-Modul-Diagramme

[0163] Ein Beispieldiagramm eines Daten-Moduls enthält üblicherweise mehrere Daten-Blöcke und Drähte, die verschiedene Blöcke miteinander verbinden, um eine oder mehrere Datenflussleitungen zu bilden. Wie bereits erwähnt ist ein Daten-Block im Allgemeinen eine Abstrahierung einer Funktion oder eines Vorgangs, den ein Benutzer auf einen Datensatz anwenden möchte. Beispielsweise kann ein bestimmter Block einen Datensatz aus dem Datenspeicher oder einer Datei auf einem Speichermedium laden, ein anderer bestimmter Block kann alle fehlenden Werten in dem Datensatz ersetzen (z. B. jene Werte, die keinen gemessenen Wert zu einem dem Zeitpunkt entsprechenden Zeitpunkt haben, zu denen ein anderer Parameter/eine andere Variable einen Messwert hat), während ein weiterer bestimmter Block eine Random-Forest-Auswertung (Randomisierte Entscheidungsbäume-Auswertung) durchführen könnte. Üblicherweise entsprechen verschiedene Blöcke verschiedenen Funktionen oder Vorgängen, die an Datensätzen durchgeführt werden können und als solches können verschiedene Daten-Blöcke jeder mit einer entsprechenden Art oder einem Namen benannt werden, z. B. „Laden“, „Abfrage“, „Befüllen“, „Spalten“, „PCA (Principal Component Analysis – Hauptkomponentenanalyse)“, „PLS (Partial Least Squares – Partielle Kleinsten Quadrate)“, „Erkunden“, „Schreiben“ etc.

[0164] Jeder Block kann keine oder mehrere entsprechende Eigenschaften haben. Der Eigenschaftensatz eines Blocks (der für einige Blöcke ein Null-Satz sein kann) entspricht jeweils seinem Blocktyp, sodass sämtliche Instanzen desselben Blocktyps den gleichen Satz an Eigenschaften haben. Für einige Blöcke können die voreingestellten Eigenschaften-Werte von der VDE-Benutzeroberflächenanwendung bereitgestellt werden, während für andere Blöcke dem Benutzer gestattet werden kann, einen oder mehrere Eigenschaften-Werte einzugeben und/oder zu verändern. Eine Daten-Blockdefinition legt die entsprechenden Eigenschaften und sämtliche voreingestellten Eigenschaften-Werte (sowie weitere Information für einige Blöcke) eines Daten-Blocktyps fest. Daten-Block-Definitionen werden in der Daten-Block-Definitionenbibliothek gespeichert, die von der VDE-Benutzeroberflächenanwendung bereitgestellt wird. Im Allgemeinen ist die Daten-Block-Definitionenbibliothek in sämtlichen offenen Instanzen der VDE-Benutzeroberflächenanwendung verfügbar, sodass mehrere Datenauswertungsbenutzer oder -ingenieure gleichzeitig Daten mithilfe der von der Bibliothek bereitgestellten Ressourcen entwickeln und/oder erkunden können.

[0165] Jeder Block hat keinen, einen oder mehrere Dateneingabe-Anschlüsse, die die Daten (sofern vorhanden) festlegen, die von einem oder mehreren an-

deren Blöcken oder Datenquellen in den Block einfließen. Zusätzlich hat jeder Block keinen, einen oder mehrere Datenausgabe-Anschlüsse, die die Daten (sofern vorhanden) festlegen, die aus einem Block herausfließen (und über ihre jeweiligen Datenausgabe-Anschlüsse möglicherweise in einen oder mehrere Empfänger-Blöcke). Die Verbindungen zwischen Dateneingaben und Datenausgaben verschiedener Blöcke untereinander werden in Daten-Modul-Diagrammen durch Drähte dargestellt. Sämtliche Datenarten können entlang eines Drahts fließen oder übertragen werden, von einfachen Skalierungswerten bis hin zu Datenpaketen, die jeweils Millionen Werte bis hin zu Objektcode enthalten.

[0166] Zusätzlich hat jeder Block einen Zustand. Wenn beispielsweise ein Block erstellt wird, befindet sich der Block im Zustand „Konfiguration“, „wird konfiguriert“ oder „nicht konfiguriert“. Nachdem der Block konfiguriert wurde, erhält der Block den Zustand „konfiguriert“. Solange ein Offline-Block bewertet wird, befindet er sich im Zustand „Bewertung“. Nachdem der Offline-Block bewertet wurde, erhält der Block entweder den Bewertungszustand „erfolgreich“ oder den Bewertungszustand „fehlgeschlagen“. Durchläuft ein Online-Block eine Konfiguration oder Anbindung an Datenquellen und/oder Datenverbraucher, so befindet er sich im Zustand „wird konfiguriert“ oder „wird angebunden“. Nachdem ein Online-Block in das Prozesssteuerungssystem eingesetzt wurde und mit Echtdaten arbeitet, befindet er sich im Zustand „eingesetzt“ oder „ausgeführt“. Natürlich sind auch andere Zustände möglich. Üblicherweise enthält jeder Block eine optische Anzeige, die den aktuellen Zustand anzeigt. Zusätzlich kann jedes Daten-Modul als Ganzes über eine Anzeige verfügen, die den aktuellen Zustand anzeigt.

[0167] Ein Beispiel-Datendiagramm **200** in **Fig. 4A** zeigt verschiedene mögliche Daten-Block-Funktionen und -verbindungen. Dieses Beispiel-Datendiagramm **200** enthält einen mit einem über einen Draht **205** mit einem FülleNaN-Block **202b** verbundenen LadeDaten-Block **202a**. Die Darstellung des Lade-Daten-Blocks **202a** enthält eine Anzeige des aktuellen Blockzustands **208a**, eine Anzeige des Blocktyps oder Namens **210a**, eine Anzeige der Ergebnisse der Blockbewertung **212a** und einen Datenausgabe-Anschluss **215a**, über den zumindest einige der als Ergebnis der LadeDaten-Block-Bewertung erzeugten Daten über den Draht **205** an einen Empfänger geliefert werden (in diesem Szenario an den FülleNaN-Block **202b**).

[0168] Der FülleNaN-Block **202b** enthält einen Dateneingabe-Anschluss **218b**, der die über den Draht **205** fließenden Daten aus dem LadeDaten-Block **202a** empfängt. Ähnlich wie LadeDaten-Block **202a** enthält der FülleNaN-Block **202b** eine Anzeige seines aktuellen Blockzustands **208b**, eine Anzeige seines

Blocktyps oder -namens **210b**, eine Anzeige der Ergebnisse seiner Bewertung **212b** und einen Datenausgabe-Anschluss **215b**, über den zumindest einige der als Ergebnis der FüllNaN-Block-Bewertung erzeugten Daten über einen Draht **220** an einen Empfänger oder Verbraucher geliefert werden (nicht dargestellt).

[0169] Es ist zu beachten, dass der LadeDaten-Block **202a** in **Fig. 4A** über keinen Dateneingabe-Anschluss verfügt, da der LadeDaten-Block **202a** keine Daten empfängt, die eine Datenausgabe eines beliebigen anderen Blocks im Datendiagramm **200** sind. Stattdessen kann der LadeDaten-Block **202a** so konfiguriert werden, dass er einen oder mehrere Datensätze aus einer oder mehreren Datenquellen lädt oder erlangt. Die Datenquellen können unter anderem Offline-Datenquellen wie eine Datendatei, eine Datenquelle (z. B. ein Seeq-System), eine relationale oder nicht-relationale Datenbank sein und/oder die Datenquellen können Online- oder Streaming-Datenquellen, wie von der Verteilte-Daten-Engine **202x** erzeugte Datenströme enthalten.

B. Datenauswertungsstudio

[0170] Wie bereits erwähnt ist einer der Betriebsmodi der VDE-Benutzeroberflächenanwendung das Datenauswertungsstudio. Das Datenauswertungsstudio kann vom Benutzer zur Gestaltung, Entwicklung, Anzeige oder Erkundung von Daten-Modellen verwendet werden. **Fig. 4B** zeigt eine vom Datenauswertungsstudio **240** dargestellte Beispielbenutzeroberfläche, die eine Navigationsleiste **242** und einen Arbeitsbereich oder Zeichenleinwand **245** enthält. Die Navigationsleiste **242** stellt Steuerelemente und Anzeigen bereit, mittels derer ein Benutzer in der Lage ist, Offline- und Online-Daten-Module zu verwalten, z. B. indem sie dem Benutzer ermöglichen, Aktionen wie die Erstellung eines neuen Offline-Daten-Moduls durchzuführen, aktuell geöffnete und auf der Leinwand **245** angezeigte Offline- oder Online-Daten-Modul zu erkennen, auf einfache Weise den Status (z. B. offline oder online) eines aktuell geöffneten und auf der Leinwand **245** angezeigten Daten-Moduls zu betrachten, aktuell geöffnete und auf der Leinwand **245** angezeigte Offline-Daten-Module zu speichern/sichern, Offline-Module in Online-Module umzuwandeln, zwischen der Anzeige von Offline- und Online-Datendiagramm eines Daten-Moduls umzuschalten, Offline-Daten-Module zu bewerten, Online-Daten-Module einzusetzen, andere Daten-Module zu durchblättern und sonstige ähnliche Modulverwaltungsfunktionen. Als solches enthält das Datenauswertungsstudio **240** eine Vielzahl von Benutzer-Steuerelementen und Anzeigen **248a–248b**, wie unter anderem:

- ein Modulnavigationssteuerelement **248a**, um dem Benutzer die Suche und das Durchblättern anderer Daten-Module zu ermöglichen;
- eine Kennzeichnung **248b** des aktuell auf der Leinwand **245** geöffneten Daten-Moduls;
- eine oder mehrere Anzeigen **248c, 248d**, die anzeigen, ob die Anzeige eines aktuell auf der Leinwand **245** geöffneten Daten-Moduls eine Offline- oder Online-Anzeige ist;
- ein oder mehrere Steuerelemente **248e, 248f**, mittels derer ein Benutzer zwischen einer Online- und einer Offline-Anzeige des aktuell auf der Leinwand **245** geöffneten Daten-Moduls umschalten kann;
- ein Benutzer-Steuerelement **248g**, mittels dessen ein Benutzer Eigenschaften des aktuell auf der Leinwand **245** geöffneten Daten-Moduls betrachten und/oder festlegen kann;
- ein Benutzer-Steuerelement **248h**, mittels dessen ein Benutzer das aktuell geöffnete Daten-Modul speichern kann;
- ein Benutzer-Steuerelement **248i**, mittels dessen ein Benutzer zumindest einen Teil des aktuell geöffneten Daten-Moduls bewerten kann;
- ein Benutzer-Steuerelement **248j**, mittels dessen ein Benutzer das aktuell geöffnete Daten-Modul einsetzen kann;
- eine Anzeige **248k**, die den Betriebsstatus des aktuell geöffneten Moduls anzeigt; und/oder
- ein oder mehrere sonstige Benutzer-Steuerelemente und/oder Anzeigen (nicht gezeigt).

[0171] **Fig. 4B** zeigt ebenfalls ein Benutzer-Steuerelement **248m**, mittels dessen ein Benutzer Block-Definitionen von einer Daten-Block-Definitionenbibliothek betrachten oder auswählen und/oder dieser hinzufügen kann (dies wird nicht in **Fig. 4B** gezeigt). In der in **Fig. 4B** gezeigten Ausführungsform wird das Benutzer-Steuerelement **248m** für die Bibliothek als auf der Leinwand **245** befindlich angezeigt, in anderen Ausführungsformen kann sich dieses Steuerelement **248m** jedoch auch in der Navigationsleiste **242** oder an jedem anderen gewünschten Ort befinden.

[0172] In der Tat sind die Zahlen, Typen, Orte/Positionen, Formfaktoren, Anordnungen etc. der in dem Datenauswertungsstudio **240**-Beispiel gezeigten Benutzer-Steuerelemente und Anzeigen nur eine von vielen möglichen Ausführungsformen. Eine größere oder geringere Anzahl und/oder Typen von Benutzer-Steuerelementen und/oder Anzeigen kann enthalten sein. Andere Orte/Positionen solcher Benutzer-Steuerelemente und/oder Anzeigen können genutzt werden, als auch andere Formfaktoren, Anordnungen etc. In einigen Ausführungsformen kann die Navigationsleiste **242** weggelassen werden und eine andere Methode zum Zugriff des Benutzers auf Steuer-Elemente bereitgestellt werden (z. B. Überlagerungsfenster, Auswahlmenü etc.).

1. Datenauswertungsstudio – Offline-Daten-Module

[0173] Im Datenauswertungsstudio **240** ist die Zeichenleinwand oder der Arbeitsbereich **245** der Bereich, mittels dessen Offline-Daten-Module entwickelt, definiert und bewertet werden können. Beispielsweise kann ein Benutzer von der Zeichenleinwand oder dem Arbeitsbereich **245** auf die Block-Definitionenbibliothek der VDE-Benutzeroberflächenanwendung zugreifen (z. B. über ein Benutzer-Steuerelement **248m**), verschiedene dort gespeicherte Daten-Block-Definitionen auswählen und die ausgewählten Block-Definitionen zu einem Datendiagramm eines Daten-Modells miteinander verbinden (z. B. verdrahten). Zusätzlich ist ein Benutzer in der Lage, auf der Zeichenleinwand oder dem Arbeitsbereich **245** die Eigenschaften einer bestimmten Daten-Blockinstanz zu bearbeiten, einen Teil des Entwurf-Daten-Modells zu bewerten, einschließlich der Betrachtung des Fortschritts der Bewertung sowie deren Ergebnisse, und/oder andere Aktionen in Bezug auf Offline-Daten-Module durchführen.

[0174] Wie bereits beschrieben, liegt jedem in einem Datendiagramm verwendbaren Daten-Block eine Blockdefinition eines Daten-Blocktyps zugrunde. Das heißt, eine bestimmte Instanz eines Daten-Blocks eines vorgegebenen Typs verfügt über einen Satz Eigenschaften entsprechend der Block-Definitionen des vorgegebenen Typs, wobei die bestimmte Instanz jedoch von anderen Instanzen des Blocks des gleichen vorgegebenen Typs abweichen kann, z. B. in einer Weise, die Objektklassen und Instanzen von Objektklassen ähnlich ist. Wie ebenfalls bereits beschrieben werden Block-Definitionen in der Block-Definitionenbibliothek gespeichert, auf die über ein Benutzer-Steuerelement **248m** zugegriffen werden kann (z. B. umschalten zwischen Anzeige und Versteckt), das sich auf der Leinwand **245** befinden kann.

[0175] Eine Veranschaulichung einer beispielhaften Beziehung **250** zwischen der Daten-Block-Definitionenbibliothek, den Daten-Block-Definitionen, Daten-Modulen, Daten-Blöcken, Daten-Blockinstanzen und Drähten ist in **Fig. 4C** zu sehen. Wie in **Fig. 4C** gezeigt, werden Block-Definitionen über eine von der VDE-Benutzeroberflächenanwendung bereitgestellte Block-Definitionenbibliothek **252** verfügbar gemacht. In einigen Ausführungsformen können verschiedenen Benutzern und Gruppen von Benutzern andere Zugriffsrechte (z. B. Nur-Lesen, Lesen-Schreiben etc.) für bestimmte Block-Definitionen und/oder andere Datenauswertungsbibliothek-Ressourcen (und/oder Teilen davon) gewährt werden.

[0176] Eine in der Bibliothek **252** gespeicherte Blockdefinition **255** kann zur Erstellung einer Instanz **258** des innerhalb eines Daten-Moduls **260** genutzten Blocks **255**, verwendet werden. Eine oder meh-

rere Eigenschaften **262** der Blockdefinition **255** können speziell für die Blockinstanz **258** festgelegt werden. Die Blockinstanz **258** kann so konfiguriert oder gestaltet werden, dass sie eine oder mehrere Dateneingabe-Anschlüsse **265** enthält, über die Daten in der Blockinstanz **258** empfangen werden, und die Blockinstanz **258** kann so konfiguriert oder gestaltet werden, dass sie einen oder mehrere Datenausgabe-Anschlüsse **270** enthält, über die Ausgabedaten (z. B. Daten als Ergebnis einer Bewertung der mit über den Dateneingabe-Anschluss **265** empfangenen Eingabedaten arbeitenden Blockinstanz **258**) bereitgestellt werden, z. B. an eine andere Blockinstanz **258**, zum Schreiben oder Speichern, an eine Benutzeroberfläche etc. Jeder Dateneingabe-Anschluss **265** einer Blockinstanz **258** kann Daten über einen oder mehrere Drähte **272** des Daten-Moduls **260** empfangen, und jeder Datenausgabe-Anschluss **270** einer Blockinstanz **258** kann Daten über einen oder mehrere Drähte **272** des Daten-Moduls **260** bereitstellen. Jeder Draht **272** des Daten-Moduls **260** stellt eine Verbindung zwischen einem bestimmten Datenausgabe-Anschluss **270** einer bestimmten Blockinstanz **258** und einem bestimmten Dateneingabe-Anschluss **265** einer anderen bestimmten Blockinstanz **258** bereit und ermöglicht so die Übertragung von Daten zwischen den beiden Blockinstanzen.

[0177] In einer Ausführungsform werden Block-Definitionen innerhalb der Bibliothek **252** in Funktions- oder Aktionskategorien unterteilt. Kategorien können Unterkategorien enthalten, Unterkategorien können Unter-Unterkategorien enthalten und so weiter. In einem veranschaulichenden, jedoch nicht beschränkenden Beispiel enthält eine Block-Definitionenbibliothek **252** mindestens fünf Kategorien Block-Definitionen: Datenquellen, Filter, Umwandler, bildliche Darstellungen und Datenverbraucher.

[0178] Block-Definitionen, die in der Datenquellenkategorie enthalten sind, legen üblicherweise verschiedene Aktionen in Bezug auf Datenquellen fest, die Eingabedaten, mit denen ein Daten-Modul arbeitet, bereitstellen. Beispielsweise können Block-Definitionen in der Datenquellenkategorie „ErstelleDatensatz“ enthalten, um einen Eingabedatensatz zu erzeugen, „LadeDatensatz“, um einen vorhandenen Datensatz aus einer Offline- oder Online-Datenquelle zu laden oder zu erlangen, „LadeDB“, um Daten aus einer Datenbank (wie einer Prozesssteuerungsdatenbank oder einer Auswertungsdatenbank) zu laden oder zu erlangen, „SpeicherDatensatz“, um einen Eingabedatensatz in einen Langzeitdatenspeicher zu überführen (z. B. nach dessen Erstellung, wie für Test-Eingabedatensätze) etc. Weiterhin können einige Datenquellenkategorien eine oder mehrere Unterkategorien enthalten. Beispielsweise können die ErstelleDatensatz- und LadeDatensatz-Kategorien jeweils entsprechende Unterkategorien für bestimmte Datensatzformate und/oder Umgebungen enthal-

ten, in denen die Datensätze erzeugt wurden, z. B. CSV (Comma Separated Values – durch Kommata getrennte Werte), Raspberry Pi, Seeq etc. Im Allgemeinen werden Datenquellen-Blöcke so konfiguriert, dass sie statische Datenquellen erlangen oder darauf zugreifen, wie Datendateien, Prozesssteuerungsdatenbanken, Auswertungsdatenbanken etc. Andererseits werden Online-Datenquellen üblicherweise mit einer Bindungsdefinition konfiguriert, die die jeweilige Instanz des Online-Datenquellen-Blocks mit einer oder mehreren Online-Datenquellen innerhalb der Prozessanlage in Beziehung setzt. Eine Kompilierung der konfigurierten Online-Datenquelle instantiiert die Bindung, sodass der Online-Datenquellen-Block von einer oder mehreren Datenquellen, an die er gebunden ist, erzeugten gestreamten Daten empfängt.

[0179] Die Datenquellenkategorie kann Block-Definitionen enthalten, die sich auf die Suche nach oder die Erlangung ausgewählter Daten von einer oder mehreren Datenquellen beziehen, z. B. „Abfrage-DatenQuelle“ oder „Abfrage“. Abfrage-Blöcke können mit Offline-Datenquellen und/oder Online-Datenquellen arbeiten. Im Allgemeinen erlaubt der Abfrage-Block einem Daten-Modul, bestimmte Arten oder Identitäten von Daten abzufragen, zu erlangen oder anzufordern (z. B. wie durch Spalten, Markierungen oder sonstige geeignete Kennzeichnungen angezeigt) und/oder nur während bestimmter Zeitabschnitte erzeugte Daten abzufragen, zu erlangen oder anzufordern, die auch unzusammenhängende Zeitabschnitte sein können. Weiterhin ist der Abfrage-Block in der Lage, jegliche Art von Daten abzufragen, unabhängig von ihrem Format und/oder ihrer Umgebung, in der die Daten erfasst oder gespeichert wurden. Zusätzliche Einzelheiten zum Abfrage-Block werden in einem späteren Abschnitt dieser Anmeldung gegeben.

[0180] In Filter-Kategorien enthaltene Block-Definitionen legen üblicherweise verschiedene Filtertechniken fest, die mit einem Datensatz durchgeführt werden können. Beispielsweise können Block-Definitionen in der Filter-Kategorie „Beschneiden“, „HochPass“, „TiefPass“, „SGF“ (z. B. für Savitsky-Golay-Filter), „Exponential“, „Mittelwert“, „Wellen“ etc. enthalten.

[0181] In der Umwandler-Kategorie enthaltene Block-Definitionen der Daten-Block-Definitionenbibliothek **252** legen üblicherweise verschiedene Techniken fest, mit denen die Inhalte eines Datensatzes oder eines gefilterten Datensatzes bearbeitet, ausgewertet und/oder in sonstiger Weise umgewandelt werden. Beispielsweise kann die Umwandler-Kategorie Daten-Block-Definitionen-Unterkategorien enthalten, die sich auf die Bereinigung von Eingabedatensätzen beziehen, z. B. „FülleNaN“ (z. B. Einträge des Datensatzes einfügen, die keine Zahl sind), „Ent-

fAusreisser“ (z. B. Ausreißerdaten entfernen), „KorrigierSchlecht“ (z. B. innerhalb des Eingabedatensatzes erkannte schlechte Daten korrigieren), „AusschlussSchlecht“ (z. B. erkannte schlechte Daten ausschließen), etc. Zusätzlich kann die Umwandler-Kategorie Daten-Block-Definitionen-Unterkategorien enthalten, die der Bearbeitung von Inhalten des Eingabedatensatzes entsprechen, wie „Skalieren“, „Verschieben“, „Aufteilen“, „Verschmelzen“, „ZentrierNormal“, „Heruntertakten“, „Zeitversatz“, „Spalten“ etc. Weiterhin kann in einigen Ausführungsformen die Umwandler-Kategorie Block-Definitionen-Unterkategorien enthalten, die der Ausrichtung von Daten innerhalb des Eingabedatensatzes entsprechen, z. B. „ZeitVerzögerung“, „VerzögerungKorrektur“ etc.

[0182] Die Umwandler-Kategorie kann Daten-Block-Definitionen-Unterkategorien enthalten, die einer Auswertung von Eingabedatensätzen entsprechen, um Erkenntnisse zu erlangen und den Inhalt zu erlernen, wie „Sensitivität“, „Anhäufung“, „RandomForest (Randomisierte Entscheidungsbäume)“, „CBP“ (Bedingte Bayes'sche Wahrscheinlichkeitsanalyse), „K Mittel“, „FourierTransform“, „SchnelleFourierTransform“, „PLS“, „PCA“ etc. Einige Unterkategorien der Umwandler-Kategorie können eine oder mehrere Unter-Unterkategorien enthalten. Beispielsweise kann die PCA-Unterkategorie der Umwandler-Kategorie Daten-Block-Definitionen-Unter-Unterkategorien enthalten, die verschiedenen PCA-(Principal Component Analyses-Hauptkomponentenanalyse)-Techniken entsprechen, wie „PCA_NIPALS“ (PCA und nicht-lineare iterative Partial Least Squares [Partielle Kleinste Quadrate]), „PCA-SVD“ (PCA und Singulärwertzerlegung), „PCA_Test“ etc.

[0183] Block-Definitionen der Bildliche-Darstellung-Kategorie legen üblicherweise verschiedene Techniken zur Darstellung von durch ein Daten-Modul erzeugten Datenausgaben fest. Beispielsweise kann die Bildliche-Darstellung-Kategorie Unterkategorien enthalten, die einer grafischen und/oder sonstigen bildlichen Darstellung entsprechen, wie „LinienDiagramm“, „Diagramm“, „BalkenDiagramm“, „StreuDiagramm“, „Histogramm“, „DatenRaster“, „DatenWolke“, „Animation“ etc. Die Bildliche-Darstellung-Kategorie kann Unterkategorien enthalten, die der Vorbereitung von Ausgabedaten für eine bestimmte grafische Darstellung entsprechen, wie „RundenSigFig“.

[0184] Block-Definitionen der Datenverbraucher-Kategorie legen üblicherweise Aktionen in Bezug auf verschiedene Verbraucher oder Empfänger von durch ein Daten-Modul erzeugten Ausgabedaten fest. Beispielsweise kann die Datenverbraucher-Kategorie eine Unterkategorie „Schreiben“ enthalten, die dafür sorgt, dass Ausgabedaten in eine Datei, Datenbank oder ein sonstiges statisches Speicherareal geschrieben oder gespeichert werden. Die Unter-

Kategorie „Schreiben“ kann wiederum über entsprechende Unterkategorien verfügen, die den verschiedenen Arten von Orten, Datenbanken, Datenbankformaten, Dateiformaten etc. entsprechen, die darauf hinweisen, wo oder wie die Ausgabedaten geschrieben oder gespeichert werden können. Die Datenverbraucher-Kategorie kann eine Unterkategorie „Veröffentlichen“ enthalten, die dafür sorgt, dass die Ausgabedaten gestreamt werden (z. B. über das Auswertungsdatennetzwerk **112**) oder in sonstiger Weise an einen Abonnenten oder Empfänger der Daten verschickt oder geliefert werden (z. B. an eine Verteilte-Daten-Engine **102x**, ein Online-Daten-Modul, eine vom Prozesssteuerungssystem ausgeführte Anwendung, eine Benutzeroberfläche etc.). In einigen Ausführungen wird der Veröffentlichen-Daten-Block konfiguriert und an einen Echtdaten-Verbraucher (z. B. an ein Online-Daten-Modul, eine ausgeführte Anwendung, ein Prozesssteuerungsgerät, -element oder -komponente etc., das über ein Abonnement für die Datenausgabe des Daten-Modells verfügt) gebunden, um einen Offline-Veröffentlichen-Block in seine Online-Version umzuwandeln. In einer Ausführungsform kann die Datenverbraucher-Kategorie eine Unterkategorie „KonvertierenZuSteuerelement“ enthalten, die dafür sorgt, dass Ausgabedaten in ein Signal eines Formats konvertiert werden, die das herkömmliche Prozesssteuerungskommunikationsnetzwerk versteht und dafür sorgt, dass das Signal über das herkömmliche Prozesssteuerungskommunikationsnetzwerk an eine Prozesssteuerungseinheit, ein -element, ein -gerät oder eine -komponente geliefert wird, um eine Veränderung in der Prozessanlage **5** hervorzurufen oder zu beeinflussen. Die Unterkategorie „KonvertierenZuSteuerelement“ kann Unter-Unterkategorien enthalten, die beispielsweise verschiedenen herkömmlichen Prozessprotokollformaten entsprechen (z. B. HART, PROFIBUS, WirelessHart etc.) und/oder verschiedenen Zielen innerhalb der Prozessanlage **5**, die potenzielle Empfänger des Veränderungssignals sind (z. B. Steuercresdateneingaben, Steuercreskonfigurationen, Meldungsprioritäten, Parameterwerte etc.). Beispielsweise kann ein KonvertierenZuSteuerelement-Block konfiguriert werden, um an eine bestimmte Prozesssteuerungseinheit, ein -element, ein -gerät oder eine -komponente innerhalb der Prozessanlage **5** gebunden zu werden, die die Datenausgabe des KonvertierenZuSteuerelement-Blocks empfangen und den Betrieb entsprechend verändern soll.

[0185] Die Daten-Block-Definitionenbibliothek **252** kann weitere Daten-Block-Definitionen enthalten, von denen jede zu einer Kategorie gehören kann oder nicht. Beispielsweise kann der Erkunden-Block einem Benutzer ermöglichen, von einem oder mehreren anderen Daten-Blöcken erzeugte Ausgabe-Daten in einem Offline-Modus zu erkunden. Beispielsweise kann der Erkunden-Daten-Block mehrere ver-

schiedene Datendrähte an seinen Dateneingabe-Anschlüsse empfangen und eine bildliche Darstellung erzeugen, die beide Sätze von Eingabedaten in vergleichender Weise enthalten, z. B. durch zeitliche Ausrichtung der beiden Eingabedatensätze und Überlagerung eines Liniendiagramms mit einem anderen Liniendiagramm, durch Erstellung von Histogrammen, bei dem die jeweiligen Datensatz-Informationen neben den Informationen des jeweiligen anderen Datensatzes angezeigt werden etc. Der Erkunden-Block ermöglicht dem Benutzer, die bildliche Darstellung seiner Datenausgabe zu bearbeiten, z. B. durch Vergrößern und Verkleinern des Maßstabs einer X-Achse und/oder einer Y-Achse, Sortieren und/oder Filtern von im Diagramm angezeigten statistischen Daten etc. Es ist zu beachten, dass der Erkunden-Block ein Beispiel für einen Block darstellt, der möglicherweise kein Online-Gegenstück hat, da seine Funktion in erster Linie darin besteht, dem Nutzer die Erkundung und das Verständnis der vom Entwurf-Daten-Modell erzeugten Datenausgabe zu ermöglichen.

[0186] Damit ein Benutzer ein neues Daten-Modell entwickeln kann, kann der Benutzer die Daten-Block-Definitionenbibliothek **252** aufrufen (z. B. durch Umschalten mittels des Benutzer-Steuerelements **248m**). Der Benutzer kann der Leinwand **245** durch Ziehen-und-Fallenlassen der Definition des gewünschten Daten-Blocks aus der Bibliothek **252** an die gewünschte Stelle der Leinwand **245** einen gewünschten Daten-Block hinzufügen (und dadurch den Daten-Block dem in der Entwicklung befindlichen auf der Leinwand **245** angezeigten Daten-Modul hinzufügen). (Natürlich ist Ziehen-und-Fallenlassen nur eine Möglichkeit, um einen bestimmten Daten-Block auszuwählen und innerhalb des Daten-Moduls zu positionieren und in dieses einzubinden, und eine Vielzahl anderer geeigneter Weisen, dies zu tun, sind möglich.) Nach dem Ziehen-und-Fallenlassen-Vorgang wird die ausgewählte Blockdefinition mit dem Entwurf-Daten-Modul verknüpft und eine entsprechende Instanz dieses Daten-Blocks für das Entwurf-Daten-Modul wird erstellt und benannt. In einem in **Fig. 4D** veranschaulichten Beispieldaten-Szenario befindet sich das Entwurf-Modul A1 in der Entwicklung und sein entsprechendes Entwurf-Offline-Datendiagramm wird auf der Datenstudio-Leinwand **245** angezeigt. Wie durch Anzeige **248c** angezeigt, hat das Modul A1 den Status „offline“. Der Benutzer hat dem Entwurf-Modul A1 bereits die Blöcke A1-B1, A1-B2 und A1-B3 hinzugefügt und mittels der Drähte A1-W1 und A1-W2 verbunden. Der Benutzer hat die Bibliothek **252** mittels des Steuerelements **248m** aufgerufen, Block B4 in der Bibliothek aufgerufen und Block B4 mittels Ziehen-und-Fallenlassen der Leinwand **245** hinzugefügt (wie durch die gestrichelte Linie angezeigt) und wird nun die Instanz von Block B4 als „A1-B4“ benennen. Nachdem die Instanz von Block B4 benannt wurde, kann der Benutzer

zer A1–B4 mit einem oder mehreren anderen Daten-Blöcken des Entwurf-Moduls A1 mit Drähten miteinander verbinden. Beispielsweise kann der Benutzer den Cursor über einen Dateneingabe-Anschluss von A1–B4 bewegen und durch einen Klick eine neue Drahtverbindung zum ausgewählten Dateneingabe-Anschluss zu erstellen. Der Benutzer kann dann den gewünschten Datenausgabe-Anschluss einer anderen Blockinstanz auf der Leinwand **245** anklicken, um eine neue Drahtverbindung zu dem ausgewählten Dateneingabe-Anschluss zu erstellen und so die beiden Blockinstanzen miteinander zu verbinden. In einem anderen Beispiel kann der Benutzer den Cursor über einen Datenausgabe-Anschluss einer anderen Blockinstanz bewegen, mit einem Klick eine neue Drahtverbindung zu dem ausgewählten Datenausgabe-Anschluss erstellen und dann mit einem Klick auf den gewünschten Dateneingabe-Anschluss von A1–B4 eine neue Drahtverbindung erstellen. Natürlich kann jede geeignete Benutzer-Steuerungsmethode zur Erstellung von Drahtverbindungen zwischen Blockinstanzen verwendet werden.

[0187] Auf der Datenstudio-Leinwand **245** kann der Benutzer Werte vorgegebener Eigenschaften eines Blocks für eine bestimmte Blockinstanz verändern. Zur Veranschaulichung stellt **Fig. 4E** einen Teil der Datenstudio-Leinwand **245** dar, auf der ein Entwurf-Daten-Modul B1 entwickelt wird. Zwei in einem Entwurf-Modul B1 enthaltene Blockinstanzen wurden auf der Leinwand **245** positioniert und miteinander verbunden, d. h. B1-FülleNaN und B1-PCA. B1-FülleNaN empfängt seinen Eingabedatensatz von einer in **Fig. 4E** nicht angezeigten Datenquelle und B1-PCA stellt entsprechenden Verbraucher- oder Empfänger-Blöcken (ebenfalls nicht angezeigt) zwei Datenausgaben „Modell“ **280a** und „Punktzahl“ **280b** bereit. Der Benutzer hat kenntlich gemacht, dass er oder sie die Werte der der Blockinstanz B1-PCA entsprechenden Block-Definitioneneigenschaften verändern möchte (z. B. durch Bewegen des Cursors über die Blockinstanz B1-PCA, durch Doppelklicken auf den Block B1-PCA oder durch Nutzung einer beliebigen sonstigen geeigneten Benutzer-Steuerungsmethode). Als Folge der Betätigung des Benutzer-Steuererelements ist ein Fenster **282** mit einer Liste der der PCA-Block-Definitionen entsprechenden festgelegten Eigenschaften erschienen (z. B. als schwedende Dialogbox, Überlagerungsfenster oder eine sonstige geeignete Darstellung). Ein Benutzer kann nun die Werte der verschiedenen Block-Eigenschaften von B1-PCA wie gewünscht über das Fenster **282** verändern.

[0188] Wie bereits beschrieben werden die Eigenschaften jedes Daten-Blocks (sofern vorhanden) über die jeweiligen in der Block-Definitionenbibliothek **252** gespeicherten Block-Definitionen festgelegt. Veranschaulichende (jedoch nicht beschränken-

de) Beispiele von Block-Definitionen und ihrer jeweiligen Eigenschaften werden in **Fig. 4F–Fig. 4H** gezeigt. Es versteht sich, dass die Block-Definitionenbibliothek **252** ein oder mehrere der in **Fig. 4F–Fig. 4H** gezeigten Block-Definitionenbeispiele, keine dieser Block-Definitionenbeispiele und/oder andere Block-Definitionen enthalten kann. Weiterhin können die Anzahl, die Typen und die voreingestellten Werte der Eigenschaften der in **Fig. 4F–Fig. 4H** gezeigten einzelnen Block-Definitionenbeispiele ebenfalls von den hier beschriebenen Beispielen abweichen.

[0189] In **Fig. 4F** wird eine Daten-Block-Definitionenvorlage eines „LadeDB“-Daten-Blocks **285a** durch die VDE-Benutzeroberflächenanwendung dargestellt, z. B. als Ergebnis eines Ziehens der LadeDB-Blockdefinition aus der Bibliothek **252** auf die Leinwand **245** durch einen Benutzer und der anschließenden Kenntlichmachung, dass er oder sie die Block-Eigenschaften-Werte des LadeDB-Daten-Blocks **285a** betrachten und/oder verändern möchte, um eine bestimmte Instanz des Blocks **285a** zu erstellen. Die Aktion oder Funktion, die der LadeDB-Block **285a** innerhalb des Offline-Diagramms eines Daten-Modells durchführt, umfasst das Laden (oder Erlangen der Inhalte in sonstiger Weise) eines bestimmten Datensatz, mit dem das Daten-Modell arbeiten soll. Beispielsweise kann der bestimmte Eingabe-Datensatz aus einer Datenbank oder einer Datei geladen oder erlangt werden. Der Benutzer ist in der Lage, den bestimmten in das Daten-Modell zu ladenden Eingabe-Datensatz durch Eingabe des gewünschten Namens, der Markierung oder des Wertes in das DatenbankName-Eigenschaften-Feld **285b** und DatenSatz-Eigenschaften-Feld **285c** zu kennzeichnen, z. B. mittels Auswahlmenü, Suche und Auswahl (z. B. einer Prozesssteuerungsdatenbank oder einer anderen mit dem Prozesssteuerungssystem verknüpften Datenbank), Freitexteingabe etc. Beispielsweise kann ein Benutzer einen gewünschten DatenbankNamen **285b** aus einem Auswahlmenü auswählen, wobei die Auswahl der gewünschten Datenbank **285b** zur Bereitstellung eines entsprechenden Auswahlmenüs für das DatenSatz-Feld **285c** führt, welches nur die Datensätze **285c** der ausgewählten/festgelegten Datenbank **285b** enthält. Der LadeDB-Block **285a** enthält auch ein Zeichnen-Eigenschaften-Feld **285d**, das in seiner Umsetzung einer Boole'schen Markierung entspricht, deren Wert anzeigt, ob eine Zeichnung eines Datensatzes **285c** aus der Datenbank **285b** erzeugt/dargestellt werden soll oder nicht, wenn die LadeDB-Blockinstanz bewertet wird. Sofern Zeichnen **285d** auf „ja“ gesetzt ist, würde nach der Block-Bewertung eine Zeichnung des geladenen Datensatzes **285c** in der grafischen Darstellung der auf der Leinwand **245** angezeigten LadeDB-Blockinstanz gezeigt werden, z. B. im Block-Ergebnis-Anzeige-Areal **212** der LadeDB-Blockinstanz. Für den LadeDB-Block **285a** ist die Zeichnen-Eigenschaft **285d** optional und des-

sen Wert auf „nein“ voreingestellt (z. B. es soll keine Zeichnung der geladenen Eingabedaten erzeugt/dargestellt werden, während der Block bewertet wird). Nachdem der Benutzer alle gewünschten Werte in die Eigenschaften-Felder **285b**, **285c** und **285d** des LadeDB-Daten-Blocks **285a** eingegeben hat, kann der Benutzer die geänderten Eigenschaften-Werte speichern und somit die LadeDB-Daten-Blockvorlage **285a** konfigurieren, um eine bestimmte Instanz des LadeDB-Daten-Blocks **285a**, die in dem Daten-Modell, das der Benutzer grade erstellt/entwickelt, verwendet werden soll. Der Benutzer kann eine bestimmte Instanz benennen, z. B. durch Eingabe eines gewünschten Namens (z. B. „LadeDB-1“) in das Block-Namensfeld **210** der auf der Leinwand **245** angezeigten Blockinstanz.

[0190] **Fig. 4G** veranschaulicht eine Daten-Block-Definitionenvorlage eines von der VDE-Benutzeroberflächenanwendung dargestellten „Spalten“-Daten-Blocks **286a**. Eine bestimmte Instanz eines Spalten-Daten-Blocks **286a** extrahiert oder erlangt ausgewählte Gruppen von Daten aus einem geladenen/erlangten Datensatz (hier als „Spalte“ von Daten bezeichnet oder als durch eine „Markierung“ referenzierte Daten), wie durch seine Eigenschaften-Werte des AusgewählteSpalten-Eigenschaften-Felds **286b** vorgegeben. Beispielsweise empfängt ein Spalten-Daten-Block **286a** einen Datensatz (oder einen Hinweis darauf, wie einen Zeiger, Anzeiger oder einen sonstigen Bezug auf den Datensatz) über seinen Dateneingabe-Anschluss, z. B. von einer Instanz eines LadeDB-Daten-Blocks **285a**. Das AusgewählteSpalten-Eigenschaften-Feld **286b** des Spalten-Daten-Blocks **286** ermöglicht dem Benutzer, eine oder mehrere Spalten, Markierungen oder sonstige Teile des Eingabe-Datensatzes auszuwählen, die der Benutzer für eine Bearbeitung durch andere Blöcke des Daten-Modells nutzen möchte. Beispielsweise scrollt der Benutzer wie in **Fig. 4G** grade durch eine Liste mit Markierungen von DatensatzA und hebt gewünschte Markierungen hervor. Üblicherweise (jedoch nicht notwendigerweise) sind die ausgewählten Spalten oder Markierungen eine Teilmenge der gesamten im Eingabe-Datensatz enthaltenen Spalten oder Markierungen. Ein Benutzer kann die Spalten- oder Markierungsauswahl **286b** (und andere Block-Eigenschaften, sofern vorhanden) speichern und somit den Spalten-Daten-Block **286a** konfigurieren, um eine bestimmte Instanz zu erstellen, z. B. die bestimmte Instanz, die im Daten-Modell verwendet werden soll, das der Benutzer grade erstellt/entwickelt. Der Benutzer kann die bestimmte Instanz der Spalten **286a** benennen, z. B. durch Eingabe eines gewünschten Namens in dessen Block-Namensfeld **210**.

[0191] **Fig. 4H** veranschaulicht eine Daten-Block-Definitionenvorlage eines von der VDE-Benutzeroberflächenanwendung dargestellten PCA_NIPALS-

Daten-Blocks **287a**. Der PCA-NIPALS-Daten-Block **287a** führt eine Hauptkomponentenanalyse (Principal Components Analysis – PCA) und einen nichtlineare iterative Partial-Least-Squares-(Partielle Kleinste Quadrate)-Vorgang mit einem Satz an Daten durch. Beispielsweise kann der PCA_NIPALS-Daten-Block **287a** über seinen Dateneingabe-Anschluss die Spalten oder Markierungen (oder Hinweise oder Bezüge darauf) von durch eine Instanz des Spalten-Daten-Blocks **286a** bestimmten Daten empfangen. In einem anderen Beispiel kann der PCA_NIPALS-Daten-Block **287a** über seinen Dateneingabe-Anschluss einen gesamten Datensatz (oder einen Hinweis darauf oder einen Bezug darauf) empfangen, der zur Verwendung durch das Daten-Modell durch einen LadeDB-Daten-Block **285a** geladen wurde. Die Vorlage des PCA_NIPALS-Blocks **287a** ermöglicht dem Benutzer, sofern gewünscht, eine Reihe Komponenten **287b** der PCA und/oder eines statistischen Sicherheitsniveaus **287c** auszuwählen. Beide Eigenschaften-Felder **287b** und **287c** sind optional für den PCA_NIPALS-Block **287a** und können auf ihre voreingestellten Werte (die z. B. in der Block-Definition des PCA_NIPALS festgelegt sind) oder auf Null gesetzt werden. In dem in **Fig. 4H** gezeigten Beispielszenario hat der Benutzer angezeigt, dass die aktuelle Instanz des PCA_NIPALS so konfiguriert werden soll, dass zwei Komponenten **287b** und mindestens ein statistisches Sicherheitsniveau **287c** von 95 % erzeugt werden. Die PCA_NIPALS-Daten-Block-Vorlage **287a** kann so konfiguriert werden (mit oder ohne geänderten Eigenschaften-Werten), dass eine bestimmte Instanz erstellt wird, die in dem Daten-Modell verwendet werden soll, das der Benutzer grade erstellt/entwickelt, wobei der Benutzer die bestimmte Instanz benennen kann, z. B. durch Eingabe eines gewünschten Namens in dessen Block-Namensfeld **210**.

[0192] Einige Daten-Block-Typen sind Verbraucher- oder Empfänger-Daten-Blöcke, die mit von einem vorangegangenen Block erzeugten Ausgangsdaten arbeiten. Diese Verbraucher-Daten-Block-Typen erfordern möglicherweise eine Bewertung des vorangegangenen Blocks, damit die Verbraucher-Daten-Blöcke konfiguriert werden können. Wenn beispielsweise eine Instanz eines LadeDB-Blocks **485a** verdrahtet ist, um Daten an eine Instanz eines Spalten-Blocks **486a** bereitzustellen, so würde eine Bewertung des LadeDB-Blockinstanz **485** den Datensatz bereitstellen, aus dem der Benutzer bestimmte Spalten oder Markierungen auswählen kann, um die Spalten-Blockinstanz **486a** zu konfigurieren.

[0193] Wenn wir uns der Bewertung von Offline-Daten-Blöcken zuwenden, so kann ein Offline-Daten-Modul, während es entwickelt wird, wiederholt bewertet werden und muss nicht fertiggestellt sein, bevor es bewertet werden kann. Als solches kann die Bewertung eines Offline-Daten-Modul asynchron erfolgen,

sodass ein Benutzer in der Lage ist, Teile eines Entwurf-Daten-Moduls zu bewerten, das Entwurf-Daten-Modul je nach Bewertung zu verändern, neu zu bewerten, erneut zu verändern, erneut neu zu bewerten etc. Beispielsweise kann ein Benutzer einen ersten Daten-Block für das Daten-Modul erstellen, den einzelnen Daten-Block bewerten, dann einen zweiten Daten-Block mit dem ersten Daten-Block verbinden, nur den zweiten Daten-Block bewerten (oder sowohl den ersten als auch den zweiten Daten-Block als Ganzes bewerten), vier weitere Daten-Blöcke hinzufügen, nur die neu hinzugefügten Daten-Blöcke bewerten (oder alle sechs Daten-Blöcke als Ganzes bewerten) etc. Das heißt, dass sobald dem Arbeitsdatendiagramm eines Offline- oder Entwurf-Daten-Moduls eine gewünschte Anzahl an Daten-Blockinstanzen und/oder zugehörigen Drähten hinzugefügt wurde, kann das auf der Leinwand **245** dargestellte Modul (oder ein Teil dessen) bewertet werden, beispielsweise durch Betätigung des Bewertungs-Benutzer-Steuerelements **248i** in der Navigationsleiste **242**. Wenn beispielsweise ein Benutzer die Gesamtheit der auf der Leinwand **245** dargestellten Blöcke und Drähte bewerten will, so kann der Benutzer einfach das Benutzer-Steuerelement **248i** bewerten. Wenn andererseits der Benutzer nur eine Teilmenge der auf der Leinwand **245** dargestellten Blöcke und Drähte bewerten will, so kann der Benutzer die gewünschten Blöcke und/oder Drähte auswählen (z. B. durch Anklicken, Ziehen einer Markierungsschlinge oder eine sonstige geeignete Methode) und dann das Steuerelement **248i** betätigen, um den ausgewählten Satz Blöcke und Drähte zu bewerten.

[0194] Fig. 4I veranschaulicht ein Beispiel, wie ein Offline-Datendiagramm **288** bewertet werden kann. Während das Offline-Datendiagramm **288** erstellt oder entwickelt wird, wird eine auf der Datenstudio-Leinwand **245** angezeigte Darstellung seiner Blöcke und Drähte in eine Transport-Datei oder ein -Dokument **290** gespeichert (hier auch als „Konfigurationsdatei“ bezeichnet), welche in einem kompakten Austauschformat wie JSON (Java Script Object Notation) oder einem sonstigen gewünschten Format vorliegt. Als solches ist das grafische Offline-Datendiagramm **288** an die Transportspeicher-Datei oder das -Dokument **290** gebunden. Wenn der Benutzer anzeigt, dass er oder sie das Offline-Datendiagramm **288** oder einen Teil dessen bewerten möchte (z. B. durch Betätigen des Benutzer-Steuerelements **248i**), so wird die Transport-Datei oder das -Dokument **290** in eine Sprache der Zielausführungsumgebung kompiliert, wodurch ausführbarer Code **292** für das Datendiagramm **288** erzeugt wird. In einer Ausführungsform kann die Transport-Datei oder das – Dokument **290** in mehrere Teile aufgeteilt werden, von denen jeder Teil in eine andere Zielsprache, die in verschiedenen Zielumgebungen (z. B. können mehrere Stücke oder Segmente ausführbaren Codes **292** von verschiedenen Zielsprachen aus der Transportspei-

cher-Datei oder des – Dokuments **290** erzeugt werden) ausgeführt werden kann, kompiliert wird. Nachdem der ausführbare Code **292** des Offline-Datendiagramms **288** erzeugt wurde, wird ein Prozess erstellt, um die Ausführung des Codes **292** zu verwalten und zu koordinieren, was über verschiedene Zielumgebungen hinweg und/oder durch verschiedene Prozessoren erfolgen kann oder nicht.

[0195] Während verschiedene Blöcke des Offline-Datendiagramms **285** ausgeführt oder bewertet werden, kann der Prozess Rückmeldung an das Datenstudio senden. Beispielsweise aktualisiert das Datenstudio auf Grundlage der Rückmeldung aus dem Prozess die entsprechenden Statusanzeigen **208** der einzelnen Blöcke des Offline-Diagramms **285**, um anzusehen, ob der Block grade kompiliert oder bewertet wird, die Bewertung erfolgreich abgeschlossen oder die Bewertung nicht erfolgreich abgeschlossen wurde (z. B. fehlgeschlagen). In der Tat kann während des gesamten in Fig. 4I veranschaulichten Kompilierungs- und Ausführungsvorgangs Rückmeldung an das Datenstudio gegeben werden und in der Benutzeroberfläche angezeigt werden. Beispielsweise aktualisiert das Datenstudio auf der Grundlage der Rückmeldung aus dem Prozess die Modul-Statusanzeige **248k** und/oder die Offline-Statusanzeige **248c**, um anzusehen, ob das Modul als Ganzes grade einer Kompilierung oder einer Bewertung unterzogen wird, die Bewertung erfolgreich abgeschlossen wurde oder die Bewertung nicht erfolgreich abgeschlossen wurde.

[0196] In der VDE-Benutzeroberflächenanwendung werden die zur Bewertung der Offline-Daten-Blöcke, Diagramme und Module verwendete Infrastruktur und Aktionen von den Datenauswertungsdiensten bereitgestellten. Beispielsweise ruft die Betätigung des Bewertungs-Benutzer-Steuerelements **248i** die Datenauswertungsdienste auf, um eine Bewertung des Offline-Daten-Blocks, Diagramms oder Moduls durchzuführen, dass gerade auf der Leinwand **245** entwickelt wird, das z. B. mehrfache verschiedene Backend-Plattformen verwendet (z. B. mehrfache verschiedene Zielsprachen, Kompilierer, Prozessoren und/oder Zielumgebungen). Dementsprechend schirmt die VDE-Benutzeroberflächenanwendung Benutzer von der Notwendigkeit ab, sich über die verwendeten Backend-Plattformen zur Umsetzung von Daten-Blöcken, Diagrammen oder Modulen Gedanken zu machen (oder Kenntnisse zu besitzen). Das heißt, ein Benutzer kann die Datenstudio- und Zentrale-Schaltstelle-Funktionen (z. B. die Daten-Block-Definitionenbibliothek **252**, die Leinwand **245** oder zugehörige Benutzer-Steuerelemente) nicht nur, um Daten-Module mit Datendiagrammen zu gestalten oder zu konstruieren, nutzen, sondern auch seine Gestaltungen in Echtzeit und unabhängig von den Backend-Plattformen, die „hinter den Kulissen“ von den Datenauswertungsdiensten verwaltet wer-

den. Darüber hinaus ermöglicht die VDE-Benutzeroberflächenanwendung in vorteilhafter Weise das Hinzufügen und/oder Löschen zusätzlicher Backend-Plattformen und/oder Komponenten, ohne die benutzerorientierte Daten-Modul-Gestaltung, Bewertung, Prüfung oder Einsatzfunktionen zu beeinflussen.

[0197] Eine nähere Beschreibung der Datenauswertungsdienste und deren Unterstützung der Auswertung erfolgt in einem späteren Abschnitt.

[0198] In Bezug auf **Fig. 4A** kann wie bereits beschrieben die Bewertung eines Daten-Moduls im Offline-Modus ein schrittweiser Vorgang sein. Während ein Benutzer Blöcke für ein Entwurf-Daten-Modul hinzufügt und konfiguriert, kann der Benutzer die hinzugefügten Blöcke bewerten, wobei der Status jedes einzelnen Blocks durch sein jeweiliges Status-Symbol dargestellt wird (z. B. die in **Fig. 4A** gezeigte Anzeige **208**). In Situationen, in denen die Bewertung des Blocks nicht erfolgreich oder fehlgeschlagen ist, kann der Benutzer Einzelheiten zum Fehlschlag erhalten (z. B. durch Auswahl oder Anklicken eines „Fehlgeschlagen“-Statusanzeigers **208**), sodass sich der Benutzer mit den Punkten, die zum Fehlschlag geführt haben, befassen kann. In Situationen, in denen die Bewertung erfolgreich war, kann der Benutzer die Ergebnisse der Block-Bewertung betrachten, z. B. durch Auswählen oder Anklicken des zum Block gehörenden Block-Ergebnisse-Anzeigen-Symbols **212**. In einer Ausführungsform kann ein modaler oder Bildliche-Darstellung-Dialog erscheinen, wenn ein Benutzer „Block-Ergebnisse anzeigen“ **212** auswählt (z. B. als Überlagerungsfenster, schwebendes Dialogfenster oder in einem sonstigen geeigneten Format) und der Benutzer kann die verschiedenen bildlichen Darstellungen betrachten und erkunden, um die Daten-Block-Ergebnisse anzuschauen.

[0199] In der Tat kann der Daten-Block, während er ausgeführt oder bewertet wird, beliebige für den Benutzer hilfreiche Ergebnisarten oder Ergebnisse speichern, um die Auswirkung/-en des Daten-Blocks, der mit Daten arbeitet, die an ihn übertragen wurden (z. B. über einen Draht und Dateneingabe-Anschluss) festzustellen. Die Arten der gespeicherten Ergebnisse sind üblicher blockspezifisch (und in einigen Fällen blockinstanzenspezifisch) und können von einem Block-Entwickler festgelegt oder bestimmt werden. Die Ergebnisse können dem Benutzer über das modale oder Bildliche-Darstellung-Dialogfenster angezeigt werden.

[0200] Zusätzlich zu den block- und/oder blockinstanzenspezifischen Ergebnissen kann die VDE-Benutzeroberflächenanwendung bildliche Standarddarstellungen bereitstellen, die auf einen (und in einigen Fällen die meisten oder sogar alle) der Daten-Blöcke anwendbar sind. Beispielsweise kann während der Ausführung eines Blocks ein Standardsatz verschie-

dener Statistiken zum Zustand der Daten bei Beendigung der Ausführung des Blocks gesammelt werden, sodass für jede Spalte, Markierung oder einem Teil des Datensatzes, der Durchschnitt, die Standardabweichung oder sonstige derartige Statistiken berechnet und zusammen mit dem entstandenen Datensatz gespeichert werden können. Wenn der Bildliche-Darstellung-Dialog einer bestimmten Blockinstanz angezeigt wird (z. B. durch Betätigen des entsprechenden Benutzer-Steuerelements **212**), werden der berechnete Standardsatz der Statistiken für jede Spalte, Markierung oder einem Teil davon aus dem Datenspeicher abgerufen und dem Benutzer angezeigt. Der Benutzer kann dann die ihn interessierenden Spalten/Markierungen/Teile auswählen und vom Datenstudio die Erzeugung entsprechender Grafiken oder sonstiger bildhafter Darstellungen anfordern, die die Statistik der besagten Spalten/Markierungen/Teile wiedergeben (z. B. Liniendiagramm, Streudiagramm, Histogramm, Datengitter, Datenzusammenfassungsgitter, berechnete Statistiken und Histogramme, die die Verteilung der Daten anzeigen etc.). In einer Ausführungsform werden die Statistiken und der Datensatz unabhängig voneinander gespeichert, da eine Speicherung der Statistiken unabhängig von dem erzeugten Datensatz der VDE-Benutzeroberflächenanwendung in vorteilhafter Weise ermöglicht, nur die erforderliche Datenmenge in den Browser zu laden.

[0201] Eine Beispielarchitektur **300** zur Anzeige von Standard- und benutzerdefinierten bildlichen Darstellungen wird in **Fig. 4J** gezeigt. In der Beispielarchitektur **300** wird der Daten-Block-Code für „Block1“ **302** ausgeführt und der sich daraus ergebende Datensatz **305**, die berechneten Standardstatistik- und sonstigen bildlichen Darstellungen **308** und Block1-spezifischen Ergebnisse **310** werden erzeugt und in einem lokalen oder entfernten von der VDE-Benutzeroberflächenanwendung verwalteten Speicherareal **312** gespeichert. In einer Datenstudio-Instanz **315** (z. B. einem Browser-Fenster) werden nach Auswahl des auf der Block1-Grafik angezeigten „Block-Ergebnisse anzeigen“-Benutzer-Steuerelements **212** durch den Benutzer die berechneten Statistiken **308** (z. B. der Standardsatz und/oder eine beliebige benutzerdefinierte bildliche Darstellung) von Block1 in die Datenstudio-Instanz **315** geladen **318**, und der Benutzer kann die ihn interessierenden gewünschten Spalten, Markierungen oder Teile auswählen. Nach Auswahl der gewünschten Spalten/Markierungen/Teile von Block1 durch den Benutzer werden die entsprechenden Daten in die Datenstudio-Instanz **315** zur Betrachtung und Erkundung durch den Benutzer geladen **320**.

[0202] Mit der Zeit können mit der Konfiguration mehrfacher Daten-Module gemeinsame Muster für Block-Konfigurationen und -Verwendung auftauchen, z. B. innerhalb desselben Prozesssteuerungssys-

tems 5 und/oder über ein Unternehmen hinweg, das über mehrere Prozesssteuerungssysteme verfügt. Falls derartige Gemeinsamkeiten erkannt werden, kann es wünschenswert sein, einen Satz bestimmter Daten-Blöcke zu gruppieren, um einen neuen Block zu bilden, der das gemeinsame Verhalten einkapselt, z. B. ein zusammengesetzter Block. In einigen Fällen kann es wünschenswert sein, einen einheitlichen benutzerdefinierten Block festzulegen, z. B. wenn sich im Laufe der Zeit ein Satz bestimmter Eigenschaften-Werte immer wieder wiederholt oder wenn ein Benutzer eine benutzerdefinierte Datenbearbeitung oder -funktion festlegen möchte. Benutzerdefinierte und/oder zusammengesetzte Blöcke können erstellt und in der Bibliothek 252 gespeichert werden, sodass sie zur Verwendung in anderen Daten-Modulen zur Verfügung stehen sind. Ein Beispielszenario 330 zur Veranschaulichung der Erstellung eines zusammengesetzten Blocks wird in **Fig. 4K** gezeigt. In einem Datendiagramm 332 wählt ein Benutzer zwei einheitliche Daten-Blöcke mit einer bestimmten Beziehung aus (z. B. „FüllenNaN“ und „Skalieren“) 335, um einen neuen zusammengesetzten Block zu bilden. Mittels eines vom Datenstudio bereitgestellten modalen Dialogfensters, Überlagerungsfensters oder einer sonstigen geeigneten Schnittstelle konfiguriert oder legt der Benutzer den neuen zusammengesetzten Block mit einem gewünschten Namen „Füllen & Skaliere“ fest und speichert den neuen zusammengesetzten „Füllen & Skaliere“-Block in der Daten-Block-Definitionenbibliothek 252 (Bezugszeichen 338). Nach Festlegung und Speicherung des „Füllen & Skaliere“-Blocks kann dieser zu einem beliebigen Zeitpunkt in ein anderes Datendiagramm 339 anstelle der eigenständigen „FüllenNaN“- und „Skaliere“-Daten-Blöcke eingefügt werden.

[0203] **Fig. 4L** zeigt ein Beispiel-Offline-Datendiagramm 340, welches einige der vorstehend beschriebenen Funktionen und Prinzipien veranschaulicht und hier mit gleichzeitigem Bezug auf **Fig. 4A**, **Fig. 4B**, und **Fig. 4F–Fig. 4H** beschrieben wird. In **Fig. 4L** wurde von einem Benutzer ein Beispiel-Offline-Datendiagramm 340 auf der Leinwand 245 des Datenstudios erstellt. Genauer gesagt hat der Benutzer die Block-Definitionenbibliothek 252 aufgerufen, z. B. durch Betätigen des Steuerelements 248m auf der Leinwand 245 und eine LadeDB-Block-Vorlage 285a auf die Leinwand 245 gezogen und fallengelassen. Weiterhin hat der Benutzer diese bestimmte Instanz des LadeDB-Blocks 285a als „LadeDB4M“ (Bezugszeichen 342a) benannt und die LadeDB4M-Blockinstanz 342 so konfiguriert, dass sie Datensatz4 aus Datenbank M lädt, z. B. durch Einstellen der Eigenschaften-Werte in den Feldern 285c und 285b des LadeDB4M-Blocks 342a. Zusätzlich hat der Benutzer, auch wenn dies nicht ausdrücklich in **Fig. 4L** gezeigt wird, die Zeichnen-Eigenschaft 285d des LadeDB4M 342a auf „Wahr“ eingestellt, sodass bei Bewertung des Blocks LadeDB4M 342a,

eine Zeichnung des geladenen Datensatz4 erzeugt und im Block-Ergebnisse-Anzeigen-Feld 212a des LadeDB4M-Blocks 342a angezeigt wird.

[0204] Nachdem der Benutzer die LadeDB4M-Blockinstanz 342a konfiguriert hat, hat der Benutzer zwei Empfänger-Spalten-Blockinstanzen 342b und 342c mit der Datenausgabe der LadeDB4M-Blockinstanz 342a verbunden. Beispielsweise hat der Benutzer zwei verschiedene Instanzen der Spalten-Daten-Block-Vorlage 286a auf die Leinwand 245 gezogen und fallengelassen und die Instanzen jeweils als „Spalten X“ (Bezugszeichen 342b) und „Spalten Y“ (Bezugszeichen 342c) benannt. Weiterhin hat der Benutzer die jeweiligen Dateneingaben des Spalten-X-Blocks 342b und Spalten-Y-Blocks 342c mit der Datenausgabe des LadeDB4M-Blocks 342a mittels verbindender Drähte verbunden.

[0205] Der Benutzer hat ebenfalls den Spalten-X-Block 342b und den Spalten-Y-Block 342c auf Grundlage der Bewertung des LadeDB4M-Blocks 342a konfiguriert. Insbesondere hat der Benutzer zuerst den LadeDB4M-Block 342a bewertet (z. B. durch Auswählen des Bildes des Blocks 342a auf der Leinwand 245 und Betätigen des „Bewerten“-Benutze-Steurelements 248i), und dadurch die Kompilierung und Ausführung des Ladens oder Erlangens von Datensatz4 aus Datenbank M durch LadeDB4M-Block 342a ausgelöst. Da die Zeichnen-Eigenschaft des LadeDB4M-Block 342a auf „Wahr“ gesetzt wurde, hat die Bewertung des LadeDB4M-Blocks 342a außerdem die Anzeige einer Zeichnung des geladenen Datensatz4 ausgelöst, z. B. im Block-Ergebnisse-Anzeigen Feld 202a des LadeDB4M-Blocks 342a (nicht in **Fig. 4L** gezeigt). Mittels dieser Zeichnung hat der Benutzer verschiedene Spalten, Markierungen oder Untergruppierungen von Daten innerhalb von Datensatz4 betrachtet und erkundet und anschließend den Spalten-X-Block 342b so konfiguriert, dass er Untergruppen oder Spalten von Daten mit der Beschriftung oder Markierung „X“ in Datensatz4 empfängt und den Spalten-Y-Block 342c so konfiguriert, dass er Untergruppen oder Spalten von Daten mit der Beschriftung oder Markierung „Y“ in Datensatz4 empfängt, z. B. durch das jeweilige Auswählen der richtigen Untergruppen-, Spalten- oder Markierungsanzeige im Eigenschaften-Feld 286b der Spalten-Blockvorlage 286a für jeden der Spalten-Blöcke 342b und 342c). Als solches führt eine Bewertung des Spalten-X-Blocks 342b nur zum Laden oder zum Zugriff auf Daten in Datensatz4 in Block 342b, die mit „X“ gekennzeichnet, beschriftet oder markiert sind, während eine Bewertung des Spalten-Y-Blocks 342c nur zum Laden oder zum Zugriff auf Daten in Datensatz4 in Block 342c führt, die mit „Y“ gekennzeichnet, beschriftet oder markiert sind.

[0206] Der Benutzer hat auch einen Partial-Least-Squares-(Partielle Kleinste Quadrate – PLS)-Block

in das Offline-Datendiagramm **340** gezogen, fallen gelassen und konfiguriert. Insbesondere hat der Benutzer die PLS-Blockinstanz **342d** als „PLS4M_X_Y“ benannt und den PLS4M_X_Y-Block **342d** so konfiguriert, dass er die Datenausgabe des Spalten-X-Blocks **342b** und die Datenausgabe des Spalten-Y-Blocks **342c** an seinen jeweiligen Dateneingaben empfängt. Der PLS4M_X_Y-Block **342c** bearbeitet oder führt eine Partielle-Kleinste-Quadrat-Funktion, -Aktion, -Algorithmus oder -Bearbeitung der durch Spalten-X **342b** und Spalten-Y **342c** bereitgestellten Daten durch, wobei das Ergebnis (z. B. ein entsprechendes auf Grundlage der Beziehung zwischen den Spalten-X- und Spalten-Y-Daten erzeugtes PLS-Modell) über die Datenausgabe des PLS4M_X_Y-Blocks **342d** an einen Erkunde4M_X_Y-Block **342e** bereitgestellt wird. Ähnlich wie die anderen Blöcke **342a–342d** ist der Erkunde4M_X_Y-Block **342e** eine Instanz einer entsprechenden Erkunden-Block-Definition, die der Benutzer auf die Leinwand **245** gezogen und fallengelassen, benannt und konfiguriert hat.

[0207] Zusätzlich zu **342d** und der Konfigurierung des Erkunde4M_X_Y-Blocks **342e** für den Empfang der von PLS4M_X_Y-Block **342d** erzeugten Datenausgabe im Offline-Diagramm **340** hat der Benutzer den Erkunde4M_X_Y-Block **342e** so konfiguriert, dass er auch die direkte Datenausgabe des Spalten-Y-Blocks **342c** als Dateneingabe empfängt, z. B. über einen Draht **342f**. Diese Konfiguration ermöglicht dem Erkunde4M_X_Y-Block **342e**, den Benutzer in die Lage zu versetzen, die Beziehung zwischen der Datenausgabe von PLS4M_X_Y-Block **342c** und der Datenausgabe des Spalten-Y-Blocks **342c** zu erkunden, z. B. durch Anzeige einer oder mehrerer bildlicher Darstellungen und/oder Statistiken. Beispielsweise enthält eine Datenausgabe des PLS4M_X_Y-Blocks **342c** üblicherweise eine oder mehrere Vorhersagewerte. Der Erkunde4M_X_Y-Block **342e** ermöglicht dem Benutzer einen Vergleich der in der Datenausgabe des PLS4M_X_Y-Blocks **342c** enthaltenen einen oder der mehreren Vorhersagewerte mit den tatsächlichen Werten des Spalten-Y-Blocks **342c**, z. B. zur Bestimmung, ob die Nutzung der Spalten-X-Daten **342b** im PLS4M_X_Y-Modell **342d** hinreichend vorhersagend für die Spalten-Y-Daten **342c** sind.

[0208] Es ist zu beachten, dass in diesem Offline-Datendiagramm weder der PLS4M_X_Y-Block **342d** noch der Erkunde4M_X_Y-Block **342e** eine Bewertung ihres oder ihrer unmittelbar vorangehenden Blocks oder Blöcke erfordert, bevor ihre eigene Konfiguration abgeschlossen werden kann. Das heißt, die Konfigurierung sowohl des PLS4M_X_Y-Blocks **342d** als auch des Erkunde4M_X_Y-Block **342e** kann unabhängig von der Bewertung der anderen Blöcke und jederzeit erfolgen. Da jedoch das Offline-Datendiagramm **340** einen oder mehrere Datenleitungsströme darstellt, erfordert üblicherweise die Bewertung

eines nachgelagerten Daten-Blocks eine Bewertung seines oder seiner vorgelagerten Daten-Blocks oder Daten-Blöcke, bevor der nachgelagerte Daten-Block bewertet werden kann (es sei denn, der empfangende oder nachgelagerte Daten-Block empfängt einen Test- oder Schein-Eingabedatensatz, der lediglich dem Zweck der Blockbewertung dient). Tatsächlich konfiguriert in einigen Szenarien ein Benutzer ein gesamtes Offline-Datendiagramm **340** als Ganzes und/oder kann ein gesamtes Offline-Datendiagramm **340** als Ganzes bewerten, anstatt Block für Block oder Teil für Teil.

2. Datenauswertungsstudio – Online-Daten-Module

[0209] Wenn wir nun zu **Fig. 4B** zurückgehen, wurde dort ein Offline-Datendiagramm eines Daten-Modells zur Zufriedenheit eines Benutzers im Datenstudio fertiggestellt und bewertet, und das Offline-Datendiagramm kann in seine äquivalente Online-Form übersetzt oder konvertiert werden. Zur Umwandlung eines Offline-Datendiagramms in seine Online-Form kann ein Benutzer das Online-Umschalt- oder Benutzer-Steuerlement **248f** in der Navigationsleiste **242** auswählen oder betätigen und dadurch die Umwandlung des Offline-Datendiagramms in sein Online-Gegenstück-Datendiagramm durch die VDE-Benutzeroberflächenanwendung auslösen sowie die Anzeige des Online-Datendiagramm des Daten-Modells auf der Leinwand **245** auslösen. Bestimmte Offline-Daten-Blöcke können über eine Online-Gegenstück-Definition verfügen (z. B. die Offline- und Online-Versionen des „Lade“-Daten-Blocks), andere Offline-Daten-Blöcke erfordern möglicherweise kein unterschiedliches Online-Gegenstück, sind jedoch in einem Online-Daten-Modul enthalten (z. B. ein „FüllNaN“-Block oder eine „PCA_NIPALS“-Block, während wiederum andere Offline-Daten-Blöcke im Online-Daten-Modul weg gelassen werden (z. B. ein „Erkunde“-Block). Die Infrastruktur und während einer Umwandlung eines Offline-Diagramms in sein Online-Gegenstück durch geführten Aktionen werden von den Datenauswertungsdiensten bereitgestellt. Beispielsweise ruft eine Betätigung des Benutzer-Steuerlements **248** die Durchführung einer Umwandlung eines Offline-Datendiagramm in sein entsprechendes Online-Datendiagramm durch die Datenauswertungsdienste auf. Eine genauere Beschreibung, wie die Datenauswertungsdienste die Umwandlung durchführen, erfolgt in der nachstehenden Beschreibung der Datenauswertungsdienste.

[0210] Wenn das Online-Datendiagramm des Daten-Modells auf der Leinwand **245** des Datenstudios dargestellt wird, kann ein Benutzer das Online-Daten-Modell konfigurieren. Üblicherweise beinhaltet die Konfigurierung des Online-Daten-Modells die Kennzeichnung der Prozesssteuerungs-Echtdaten, die sich auf die Steuerung des Prozesses beziehen, der als Eingabe-Daten für das Online-Daten-Modul

zu erlangen ist, z. B. durch Festlegen der Bindungen, die das Online-Daten-Modul der entsprechenden Datenquelle der Eingabedaten zuordnen. Zusätzlich kann eine Konfigurierung eines Online-Daten-Moduls einen Verweis auf den Ort oder die Orte und/oder Verbraucher beinhalten (z. B. durch Festlegen der Bindungen), an die vom Online-Daten-Modul erzeugte Ausgabedaten bereitgestellt werden sollen (z. B. eine oder mehrere Verbraucher-VDE und/oder -anwendungen, Datenspeicher, Dateien, Historiker, Prozesssteuerungsgeräte, Routinen, Elemente, Komponenten etc.). Die Konfigurierung und/oder Änderung von Daten-Blöcken, ihren jeweiligen Eigenschaften und Verbindungen von Online-Daten-Modulen untereinander erfolgt in ähnlicher Weise wie vorstehend in einer Ausführungsform hinsichtlich der Offline-Daten-Module beschrieben.

[0211] Falls der Benutzer ein Problem oder eine Fragestellung bezüglich eines Online-Daten-Moduls erkennt oder in sonstiger Weise das Online-Daten-Modul verändern möchte, kann der Benutzer das Online-Daten-Modul umdrehen oder zurück in dessen Offline-Gegenstück schalten. z. B. durch Nutzung des Offline-Umschalt- oder Benutzer-Steuerlements **248d**, und der Benutzer kann weiter das Offline-Datendiagramm des Daten-Modells wie vorstehend beschrieben verändern und bewerten. Anschließend wird, wenn der Benutzer das Offline-Datendiagramm zurück in dessen Online-Gegenstück dreht oder schaltet, das veränderte Offline-Datendiagramm in dessen entsprechendes Online-Datendiagramm des Daten-Modells umgewandelt.

[0212] **Fig. 4M** veranschaulicht das dem Beispiel-Offline-Datendiagramm **340** aus **Fig. 4L** entsprechende Online-Datendiagramm **345**, wobei **Fig. 4M** nachstehend mit gleichzeitigem Bezug auf **Fig. 4A**, **Fig. 4B**, **Fig. 4F–Fig. 4H** und **Fig. 4L** beschrieben wird. In dem in **Fig. 4M** gezeigten Beispielszenario hat der Benutzer entschieden, dass das Offline-Datendiagramm **340** in **Fig. 4L** das Daten-Modell wie beabsichtigt oder gewünscht festlegt und dass das Daten-Modell bereit für den Einsatz im Online-Prozesssteuerungssystem ist. Als solches, hat der Benutzer während der Betrachtung des Offline-Datendiagramms **340** (z. B. wie in **Fig. 4L** gezeigt) den Online-Umschalter **248f** des Datenstudios betätigt und dadurch das Offline-Datendiagramm **340** in dessen entsprechendes Online-Datendiagramm **345** umgewandelt, welches auf der Leinwand **245** des Datenstudios wie in **Fig. 4M** gezeigt dargestellt wird. Diese Umwandlung erfolgt durch die Datenauswertungsdienste der VDE-Benutzeroberflächenanwendung, wobei die jeweiligen von den Datenauswertungsdiensten zur Durchführung dieser und anderer Umwandlungen verwendeten Aktionen, Methoden und Architektur in einem späteren Abschnitt genauer beschrieben werden.

[0213] Es ist zu beachten, dass sich bei dem Online-Datendiagramm **345** und Offline-Datendiagramm **340**, obwohl beide Repräsentationen desselben Datenmodells sind, die Blöcke und Linien der beiden Datendiagramme **345**, **340** voneinander unterscheiden. So wurde zum Beispiel die Datenpipeline des Offline-Datendiagramms **340**, die an der Ausgabe des LoadDB4M-Blocks **342a** ihren Ausgang nimmt, den Spalten-Y-Block **342e** durchläuft und am Eingang des Explore4M_X_Y-Blocks **342e** endet, im Online-Datendiagramm **345** weggelassen, da die Datenpipeline im Offline-Datendiagramm **340** verwendet wurde, um den Offline-PLS4M_X_Y-Block **342d** zu testen und zu validieren, und sie nun, da die Effizienz des PLS4M_X_Y-Blocks **342d** zur Zufriedenheit des Nutzers nachgewiesen wurde, in der Online-Umgebung nicht benötigt wird. Es ist möglich, dass der Nutzer sich in einigen Ausführungsformen entscheidet, diese Validierungsdatenpipeline im Online-Datendiagramm **345** zu verwenden, z. B. falls er wünscht, dass das Testen und Validieren des Datenmodells in der Online-Umgebung durchgehend erfolgen soll.

[0214] Ein weiterer Unterschied zwischen dem Offline-Datendiagramm **340** und dem Online-Datendiagramm **345** ist der LoadDB4M-Block. Im Offline-Datendiagramm **340** ist der LoadDB4M-Block **342a** so konfiguriert, dass er Daten aus der Offline-Datenquelle Dataset4 aus der Datenbank M lädt. Im Online-Datendiagramm **345** wurde der LoadDB4M-Block **347a** dagegen in einen Block umgewandelt, der sich mit einer Online-Datenquelle verbinden kann, wie zum Beispiel eine Online-Streaming-Datenquelle innerhalb der Prozessanlage **5**. Der Nutzer kann den LoadDB4M-Block **347a** so konfigurieren, dass er an eine gewünschte Online-Datenquelle gebunden wird, wobei die Bindung bei der Kompilierung des konfigurierten LoadDB4M-Blocks **347a** instanziert wird.

[0215] In dem Online-Datendiagramm **345** ähnelt oder gleicht der Spalten-X-Block **347b** im Allgemeinen seiner Offline-Form **342b**. Wie jedoch bereits erläutert, erhält die Online-Form des PLS4M_X_Y-Blocks **347c** nur die Eingangsdaten, die der PLS4M_X_Y-Block **347c** benötigt, um in der Online-Umgebung arbeiten zu können, d. h. die Ausgabedaten des Spalten-X-Blocks **347b**.

[0216] Des Weiteren gibt es in dem Online-Datendiagramm **345** kein Gegenstück zu dem Offline-Explore4M_X_Y-Block **342e**, da der Explore4M_X_Y-Block **342e** in der Offline-Umgebung verwendet wurde, um dem Nutzer einen Einblick zu gewähren, wie gut die Ziele des Nutzers mit dem Entwurfsdatendiagramm **340** erreicht werden. In dem Online-Datendiagramm **345** werden die Ausgabedaten des PLS4M_X_Y-Blocks **347c** jedoch einem „Schreib“-Datenblock **347d** bereitgestellt, bei dem es sich um eine Art Datenverbraucher-Datenblock handelt. Der Schreib-Da-

tenblock **347d** bewirkt, dass die Echtzeit-Ausgabedaten des PLS4M_X_Y-Blocks **347c** in der Online-Umgebung in eine Datei oder Datenbank geschrieben werden, die in der Konfiguration des Schreib-Blocks **347c** angegeben ist. Falls der Nutzer es wünscht, könnten die Ausgabedaten des PLS4M_X_Y-Blocks **347c** natürlich zusätzlich oder ersatzweise einem oder mehr anderen Datenverbrauchern des Datenmodells bereitgestellt werden, indem die Ausgabe des PLS4M_X_Y-Blocks **347c** und die Eingänge der anderen Datenverbraucher-Blöcke untereinander verbunden werden. So könnte der PLS4M_X_Y-Block **347c** zum Beispiel mit einem Veröffentlichungs-Block verbunden sein, so dass die Ausgabe des PLS4M_X_Y-Blocks **347c** in dem Datenanalytik-Netzwerk **112** veröffentlicht und von anderen Datenanalytik-Anwendungen verwendet werden kann. In einem anderen Beispiel könnte der PLS4M_X_Y-Block **347c** mit einem Convert2Control-Block verbunden sein, so dass die Ausgabe des PLS4M_X_Y-Blocks **347c** eine Änderung im Online-Prozessleitsystem bewirkt. So kann ein Convert2Control-Block zum Beispiel so konfiguriert sein, dass die Ausgabedaten des PLS4M_X_Y-Blocks **347c** in ein Eingangssignal für eine Prozessleit-Funktion, eine Regelkreis oder eine Anwendung innerhalb der Prozessanlage **5** umgewandelt werden, und bewirken, dass das Eingangssignal dem entsprechenden Prozessleit-Sektor, -Element, -Gerät oder der entsprechenden Prozessleit-Komponente bereitgestellt wird.

3. Einsatz von Datenmodulen

[0217] Typischerweise werden Online-Datenmodule nicht asynchron evaluiert, wie dies für Offline-Datenmodule möglich ist. Stattdessen werden Online-Datenmodule zunächst konfiguriert (z. B. an eine Live-Datenquelle und/oder einen oder mehr Datenverbraucher/-empfänger gebunden) und werden dann in ihrer Gesamtheit eingesetzt, um das Online-Prozessleitsystem unterbrechungslos auszuführen und mit ihm zu interagieren. Ein Nutzer kann ein Datenmodul in einem Online-Prozessleitsystem zum Einsatz bringen, in dem er zum Beispiel auf der Navigationsleiste **242** ein Nutzer-Steuerelement **248j** „einsetzen“ anklickt oder aktiviert und damit den Datenanalytikdiensten mitteilt, dass diese den Einsatz des Online-Datenmoduls abzuwickeln haben. Die Infrastruktur und Aktionen, die für den Einsatz von Datenmodulen verwendete werden, werden in der Beschreibung der Datenanalytikdienste detaillierter beschrieben. Im Allgemeinen gilt jedoch, dass ein Online-Datenmodul, wenn es zum Einsatz kommt, einem Komplierungs- und Ausführungsprozess folgt, der dem eines Offline-Datenmoduls, das evaluiert wird, wie z. B. in **Fig. 4** dargestellt, folgt. Die Ziel-Umgebung(en) eines Online-Datenmoduls kann (können) sich jedoch von der (den) Ziel-Umgebung(en) seines (ihres) korrespondierenden Offline-Datenmoduls unterscheiden und Online-Datenmodule sind an

Live-Datenquellen innerhalb des Prozessleitsystems oder der Prozessanlage gebunden.

[0218] In einer Ausführungsform unterstützt die DDE-Nutzerschnittstellenanwendung einen Einsatz durch einmaliges Anklicken. Dies bedeutet, dass ein Nutzer, wenn ein Offline-Datenmodul zur Zufriedenheit des Nutzers konfiguriert wurde, keine Konvertierung des Datenmoduls in seine Online-Repräsentation und Konfigurierung der Bindungen für die Datenblöcke der Online-Repräsentation, die das Lesen und Schreiben von Daten von Live-Datenquellen übernehmen, durchführen muss. Stattdessen werden bei einem Einsatz durch einmaliges Anklicken nach der Initiierung durch den Nutzer alle notwendigen Bindungen für das jeweilige Datenmodul automatisch von der DDE-Nutzerschnittstellenanwendung ermittelt und konfiguriert. In einem beispielhaften Szenario kann ein Nutzer die Ausführung eines Datenmoduls einem bestimmten DCS-Controller zuweisen, z. B. indem er ein vom Datenstudio bereitgestelltes Dialogfeld oder -fenster verwendet, das Informationen zur Hierarchie des DCS-Systems aufruft. Der Nutzer kann die Hierarchie durchgehen und das Datenmodul entsprechend zuweisen. Die DDE-Nutzerschnittstellenanwendung (z. B. das Datenstudio und/oder die Datenanalytikdienste) verwendet daraufhin das zugewiesene Prozessleitsystem-Element (in diesem Szenario den bestimmten DCS-Controller), um automatisch die notwendigen Bindungen zu ermitteln. Die für den Einsatz durch einmaliges Anklicken von Datenmodulen verwendete(n) beispielhafte(n) Infrastruktur und Aktionen werden in der Beschreibung der Datenanalytikdienste detaillierter beschrieben.

[0219] Nachdem ein Online-Datenanalytikmodul erfolgreich eingesetzt wurde, können alle daraus resultierenden (z. B. deskriptiven, prädiktiven und/oder präskriptiven) Werte, die durch seine Ausführung erzeugt werden, dem Nutzer angezeigt werden, z. B. über das Daten-Dashboard der DDE-Nutzerschnittstellenanwendung. In einer Ausführungsform umfassen die daraus resultierenden Werte zum mindesten einen prädiktiven Wert und der (die) korrespondierende(n) Echtzeit-Live-Datenwert(e), der (die) von dem Prozessleitsystem **5** für den (die) Tag(s) oder die Prozessleit-Elemente erzeugt wurde(n), für die die prognostizierten Werte gelten, kann (können) ebenfalls zusammen mit dem prognostizierten Wert angezeigt werden.

C. Datenanalytik-Dashboard

[0220] Wie zuvor erläutert umfasst die DDE-Nutzerschnittstellenanwendung einen zweiten Arbeitsmodus, das Datenanalytik-Dashboard. Das Dashboard wird vom Nutzer typischerweise dazu verwendet, sich Module anzeigen zu lassen und/oder diese zu managen, die (i) zu evaluierende Offline-Datenmodule (z. B. Entwurfsdatenmodelle) oder (ii) in einem On-

line-Prozessleitsystem oder einer Online-Prozessleitanlage zum Einsatz kommende Online-Datenmodule sind. Durch visuelle und anderweitige Repräsentationen stellt das Dashboard einem Nutzer eine Ansicht zu Status- und anderen Informationen, die mit der Evaluierung von Offline-Datenmodellen korrespondieren, als auch eine Ansicht zum Einsatz kommender Online-Datenmodule und der Echtzeit- oder Live-Informationen, die von den Online-Datenmodulen erzeugt werden, bereit. Allgemein gesagt, stellt das Datenanalytik-Dashboard dem Nutzer eine Schnittstelle bereit, über die der Nutzer Online- und Offline-Datenmodule überwachen und managen kann. Der Nutzer kann zudem die Evaluierung von Offline-Datenmodulen und die Ausführung von Online-Datenmodulen vom Dashboard aus beenden. Im Allgemeinen, aber nicht notwendigerweise, werden zumindest einige der auf dem Datenanalytik-Dashboard angezeigten Informationen dem Dashboard **350** durch die Datenanalytdienste der DDE-Nutzerschnittstellenanwendung bereitgestellt.

[0221] Fig. 4N-1 zeigt eine beispielhafte Nutzerschnittstelle **350**, die vom Datenanalytik-Dashboard der DDE-Nutzerschnittstellenanwendung präsentiert wird. Ein erster Bereich **352** der beispielhaften Dashboard-Nutzerschnittstelle **350** stellt aktuelle Datenanalytik-Überblicksinformationen bereit, wie eine Angabe zur Anzahl der Online-Datenmodule **355**, die gegenwärtig von der DDE-Nutzerschnittstellenanwendung überwacht werden, und eine Angabe zur Anzahl der Live-Datenströme **358**, die von den Online-Datenmodulen **355** bearbeitet werden. Wie zuvor erläutert, werden die Datenströme **358** von einer oder mehr Datenquellen empfangen, die stetig Echtzeit-Daten erzeugen, bei der Steuerung der einen oder mehr Industrieverfahren durch die Online-Prozessleitanlage oder -umgebung **5** entstehen, und ausführende Online-Datenmodule **355** wurden so konfiguriert, dass sie Live-Datenströme **358** empfangen können. Die Datenanalytik-Überblicksinformationen umfassen zudem eine Angabe zur Anzahl der Offline-Datenmodule **360**, deren Evaluierungen gegenwärtig von der DDE-Nutzerschnittstellenanwendung überwacht werden. In einigen Ausführungsformen umfasst die Dashboard-Nutzerschnittstelle **350** eine oder mehr anderweitige Mitteilungen oder anderweitige Informationen **362**, die die überwachten Datenanalytik-Module, unabhängig davon ob on- oder offline, betreffen.

[0222] Klickt der Nutzer auf eine der Überblickssteuerelemente **355**, **358**, **360**, **362** oder aktiviert sie anderweitig, werden weitere zusammenfassende Informationen, die der Auswahl entsprechen, in einem zweiten Bereich **365** des Dashboards **350** präsentiert. In Fig. 4N-1 hat der Nutzer eine Auswahl getroffen, um sich die zusammenfassenden Informationen zu den aktuell ausführenden Online-Datenmodulen **355** anzeigen zu lassen, von denen es zu diesem

Zeitpunkt zwei gibt, nämlich Diagramm2 mit der Bezugsnummer **368** und Diagramm3 mit der Bezugsnummer **370**. Für jedes ausführende Online-Datenmodul **360**, **370** werden im zweiten Bereich **365** des Dashboards **350** die jeweiligen zusammenfassenden Informationen angezeigt, wie die Bezeichnung des Moduls **372**, die Startzeit für den Einsatz des Moduls **375**, eine Trendangabe **378** (die, wie in Fig. 4N-1 gezeigt, grafisch sein kann) der Ausgabe des Moduls während eines letzten Zeitraums, die aktuellste letzte Prognose oder der (die) aktuellste(n) prognostizierte(n) Wert(e) **380** und/oder weitere Informationen (nicht gezeigt). Es sei besonders hervorgehoben, dass auf dem Dashboard **350** die Trendangabe **378** eines jeden Online-Datenmoduls **368**, **370** fortlaufend aktualisiert wird, um die fortlaufend erzeugten Ausgabedaten des jeweiligen Datenmoduls **368**, **370** wiederzugeben und es dem Nutzer zu ermöglichen, die Ausführung von Online-Datenmodulen und die dadurch fortlaufend generierten Ergebnisse in Echtzeit zu überwachen.

[0223] Darüber hinaus ermöglicht es das Datenanalytik-Dashboard **350** einem Nutzer, auf mit den Datenmodulen **360**, **378** in Zusammenhang stehende, zusammenfassende Informationen zu klicken, um sich detaillierte Informationen anzeigen zu lassen, z. B. in einem Pop-Up-Fenster, Dialogfeld oder mit einem anderen geeigneten Anzeigeverfahren (nicht gezeigt). Klickt der Nutzer zum Beispiel auf die Bezeichnung des Diagramm2 **368** kann das Dashboard **250** das Online-Datenmoduldiagramm des Diagramm2 **368** und Angaben zu einem oder mehr bestimmten Datenströmen **358**, an die das Diagramm2 **368** gebunden ist, präsentieren. Klickt der Nutzer auf die Start-Informationen des Diagramm2 **368**, wird eine Angabe zum Urheber des Diagramm2 **368**, dem Nutzer, der das Diagramm2 **368** zum Einsatz gebracht hat, und/oder einem Eigentümer des Diagramm2 **368** präsentiert. Durch Klicken auf die Trendinformationen **378** zum Diagramm2 **368** wird es dem Nutzer möglich, den Trend der Ausgabedaten des Diagramm2 **368** zu bearbeiten und durchsuchen, z. B. durch Ausweiten oder Verkürzen des Anzeigezeitraums, Umwandlung der Trendinformationen in eine andere Visualisierungsart (z. B. Balkendiagramm, Kreisdiagramm, Streudiagramm usw.), Anwendung und/oder Abrufen von Statistiken bzgl. der Trenddaten über einen gegebenen Zeitraum (z. B. Durchschnitt, gleitender Durchschnitt, Maximalwert, Minimalwert usw.) und andere derartige Vorgänge. In ähnlicher Weise wird es einem Nutzer durch Klicken auf die Information Letzte Prognose **380** zum Diagramm2 **368** möglich, den oder die letzten prognostizierten Wert(e) zu bearbeiten und zu durchsuchen, z. B. indem er sich die Zeit der letzten Prognose und, wie häufig Prognosen erfolgen, anzeigen lässt, die letzte Prognose mit dem aktuellen Live-Datenwert vergleicht usw.

[0224] Wählt der Nutzer das Nutzer-Steuerelement **360** aus, um sich aktuell ausführende Offline-Datenmodule (von denen im beispielhaften Szenario der **Fig. 4N-1** keine gezeigt werden) anzeigen zu lassen, werden zusätzliche zusammenfassende Informationen, die mit den Offline-Datenmodulen korrespondieren, im zweiten Bereich **365** des Dashboards **350** präsentiert. Im Allgemeinen ähneln die zusammenfassenden Informationen zu einem Offline-Datenmodul denen eines Online-Datenmoduls, statt jedoch die zugehörigen Live-Datenströme anzuzeigen, können die zusammenfassenden Informationen des Offline-Datenmoduls die Testdatendatei(en) oder Testdatenströme (z. B. Testtools, -geräte und/oder -umgebungen) anzeigen, die von Offline-Datenquellen erzeugt wurden, auf denen das Offline-Datenmodul ausgeführt wird. Außerdem können die zusammenfassenden Informationen zu einem Offline-Datenmodul den aktuellen Status, Zustand und/oder Fortschritt seiner Ausführung anzeigen.

[0225] In einigen Szenarien kann ein Nutzer ein Offline-Datenmodul aus dem Dashboard (statt dem Datenstudio) konfigurieren und/oder zum Einsatz bringen. Zum Beispiel kann ein Nutzer die Evaluierung eines Offline-Datenmoduls aus dem Dashboard überwachen und dieses Offline-Datenmodul schließt seine Evaluierung erfolgreich ab. Der Nutzer kann ein Nutzer-Steuerelement des Dashboards (nicht gezeigt) aktivieren, über das der Nutzer das Datenmodul so konfigurieren kann, dass es auf die Live-Datenquelle(n) und auf durch die Live-Datenquelle(n), auf der (denen) das Online-Datenmodul ausgeführt wird, erzeugte Daten Bezug nimmt. Die Konfiguration des Online-Datenmoduls zum Binden der Datenquelle(n) ähnelt dem oben beschriebenen, nur dass es vom Dashboard und nicht dem Datenstudio aus initiiert wird. Ersatzweise kann der Nutzer einen Einsatz des Offline-Datenmoduls durch einmaliges Anklicken auf dem Dashboard verwenden.

[0226] Wählt der Nutzer das Nutzer-Steuerelement **358** aus, um sich Informationen anzeigen zu lassen, die mit Online-Datenströmen in Zusammenhang stehen, die aktuell an ausführende Online-Datenmodule (von denen im beispielhaften Szenario der **Fig. 4N-1** drei gezeigt werden) gebunden sind, werden zusätzliche zusammenfassende Informationen, die mit den Online-Datenströmen korrespondieren, im zweiten Bereich **365** des Dashboards **350** (nicht gezeigt) präsentiert. Die zusammenfassenden Informationen zu jeden Live-Datenstrom können eine Kennung des Datenstroms, eine Angabe zur Datenquelle, von der der Datenstrom empfangen wird, eine Angabe über ein korrespondierendes Prozessleitsystem-Tag oder eine andere herkömmliche Prozessleitsystem-Kennung der Live-Datenquelle, Informationen zu Abonnements und/oder der Veröffentlichung des Datenstroms, eine Angabe zu einem oder mehr Online-Datenmodulen, die gegenwärtig auf dem Live-Daten-

strom ausgeführt werden, eine fortlaufend aktualisierte Visualisierung des Live-Datenstroms (z. B. Liniendiagramm, Balkendiagramm, Streudiagramm usw. und/oder grundlegende Statistiken zum Datenstrom) und/oder weitere Informationen enthalten.

[0227] Es sei insbesondere darauf hingewiesen, dass in dem beispielhaften Szenario aus **Fig. 4N-1** der Datenstromüberblick **358** anzeigt, dass die Online-Datenmodule gegenwärtig auf drei Live-Datenströmen arbeiten, während der Online-Datenmodul-überblick **355** anzeigt, dass nur zwei ausführende Online-Module vorhanden sind. Dies bedeutet, dass eins der zwei ausführenden Online-Module **368, 370** auf zweien der Live-Datenströme arbeitet. Der Nutzer kann die Beziehung herausfinden, indem er sich entweder zur Ansicht der jeweiligen Online-Datenmodule **368, 370** bewegt, in der eine Angabe zu den Datenströmen, an die das jeweilige Modul **368, 370** gebunden ist, bereitgestellt wird, oder indem er sich zur Ansicht mit den detaillierten Angaben zu jedem Datenstrom **358** bewegt, in der eine Angabe zu den ausführenden Online-Datenmodulen, die auf jedem der Datenströme **358** arbeiten, bereitgestellt wird.

[0228] **Fig. 4N-2** zeigt eine beispielhafte Nutzerschnittstelle **382**, die vom Datenanalytik-Dashboard der DDE-Nutzerschnittstellenanwendung präsentiert werden kann. In einer Ausführungsform wurde die Nutzerschnittstelle **382** in den zweiten Bereich **365** der in **Fig. 4N-1** gezeigten Nutzerschnittstelle **350** aufgenommen. In einer Ausführungsform handelt es sich bei der Nutzerschnittstelle **382** um eine Vollansicht.

[0229] Die Nutzerschnittstelle **382** umfasst eine oder mehr Kacheln oder Karten **385a–385j**, die alle jeweils einem ausführenden Online-Datenmodul zugeordnet sind und die zusammen als eine Palette oder ein Kartenstapel **385x** bezeichnet werden können. Jede Karte **385x** in der Palette kann Informationen enthalten, die den für die ausführenden Online-Datenmodule **360, 370** aus **Fig. 4N-1** angezeigten Informationen ähneln, z. B. die Bezeichnung des Moduls, eine Startzeit für den Einsatz des Moduls, eine Trendangabe für die Ausgabedaten des Moduls während eines letzten Zeitraums, die aktuellste letzte Prognose oder der (die) aktuellste(n) prognostizierte(n) Wert(e) und/oder weitere Informationen. Darüber hinaus wird ähnlich wie bei den Anzeigen für die ausführenden Online-Datenmodule **360, 370** die Trendangabe einer jeden Karte **385x** fortlaufend aktualisiert, um die fortlaufend erzeugten Ausgabedaten seines jeweiligen Online-Datenmoduls wiederzugeben und es dem Nutzer so zu ermöglichen, seine Ausführung und die von ihm fortlaufend generierten Ergebnisse in Echtzeit zu überwachen.

[0230] Bei jeder Karte **385x** kann der Nutzer die Größe und die Position nach seinen Wünschen anpas-

sen. In einer Ausführungsform wird eine bestimmte Karte **385x** vergrößert, wenn der Fokus auf ihr liegt. Eine Karte **385x** kann manuell in den Fokus gebracht werden, indem der Nutzer zum Beispiel auf die bestimmte Karte **385x** klickt, sich über die bestimmte Karte **385x** bewegt usw. Zusätzlich oder ersatzweise kann eine Karte **385x** automatisch in den Fokus gebracht werden, wenn zum Beispiel ein bestimmtes Datenmodul dringend Aufmerksamkeit bedarf (z. B. ein prognostizierter Wert übersteigt einen Schwellwert, das Datenmodul selbst benötigt eine Nutzereingabe oder einen Nutzereingriff, Rangfolge der Schwere der bei den Karten **385x** vorliegenden Bedingungen usw.). Andere automatisch erzeugte Hinweise, dass Aufmerksamkeit erforderlich ist, können auf der Nutzerschnittstelle **382** angezeigt werden, zum Beispiel, indem eine bestimmte Karte **385x** hervorgehoben wird oder einer anderen Bearbeitung unterliegt, d. h. die Karten **385x** aus dem Satz erhalten eine Größe je nach Dringlichkeit, eine bestimmte Karte **385x** wird im Vordergrund oder am Anfang der Palette angezeigt, die Karten **385x** werden in der Reihenfolge, in der sie Aufmerksamkeit bedürfen, angeordnet usw. In einer Ausführungsform kann ein Nutzer definieren welche bestimmten Bearbeitungen automatisch in welchen bestimmten Situationen auf die Karten **385x** angewendet werden.

[0231] Es soll nun auf Datenblockdefinitionen **255**, die in der Datenblockdefinitionen-Bibliothek **252** der DDE-Nutzerschnittstellenanwendung (z. B. wie **Fig. 4C** gezeigt) gespeichert sind, zurückgekommen werden: die Datenblockdefinitionen, die in der Bibliothek **252** gespeichert sind, können von der DDE-Nutzerschnittstellenanwendung dynamisch während des Hochfahrens oder der Initialisierung erkannt werden. In einer Ausführungsform werden Datenblockdefinitionen definiert, indem ein kompaktes Datenaustauschformat, wie JSON-Dateien oder -Dokumente, oder ein anderes geeignetes Format verwendet wird. Die Definitionsdatei oder das Definitionsdokument eines bestimmten Datenblocks beinhaltet eine Blockdefinitionsbezeichnung und führt die genauen Block-eigenschaften und -anschlüsse auf, die den bestimmten Datenblock definieren. In einigen Fällen kann eine bestimmte Blockdefinition eine Definition für eine Offline-Repräsentation und eine Definition für eine Online-Repräsentation bereitstellen.

[0232] Da die DDE-Nutzerschnittstellenanwendung fortlaufend und dynamisch Blockdefinitionen entdecken kann, können zusätzliche Blockdefinition (z. B. für spezielle Datenblöcke, zusammengesetzte Datenblöcke und/oder andere Datenblöcke) jederzeit zur Bibliothek **252** hinzugefügt werden, während die DDE-Nutzerschnittstellenanwendung ausgeführt wird. In einer Ausführungsform werden zusätzliche Blockdefinitionen durch eine Dienstanwendung erzeugt, die in dieser Schrift als „Datenblockdefinitions-Wrapper“ bezeichnet wird. Der Datenblockdefini-

tions-Wrapper kann zum Beispiel aufgerufen werden, wenn ein Nutzer einen speziellen oder zusammengesetzten Datenblock, wie bereits in Bezug auf **Fig. 4K** beschrieben, speichert. Wird ein Blockdefinitions-Wrapper ausgeführt, erstellt er eine Instanz der zusätzlichen Blockdefinition und bewirkt, dass ein entsprechendes JSON-Dokument oder Ähnliches, durch das die zusätzliche Blockdefinition definiert wird, erstellt und gespeichert wird. Zum Beispiel erzeugt die erstellte Instanz der zusätzlichen Blockdefinition das JSON-Dokument und speichert es in der Bibliothek **252**. **Fig. 4K** zeigt ein beispielhaftes Szenario, in dem, wenn der Nutzer die zusammengesetzte Blockdefinition **338** speichert, der Blockdefinitions-Wrapper so ausgeführt wird, dass ein JSON-Dokument oder Ähnliches für die zusammengesetzte Blockdefinition **338** erzeugt und gespeichert wird.

[0233] Während der Konfiguration eines Datenblocks (unabhängig davon ob on- oder offline) muss die DDE-Nutzerschnittstellenanwendung eventuell eine Funktion (z. B. Geschäftslogik) durchführen, um die Konfigurationserfahrung zu erreichen, die erforderlich ist, damit der Nutzer bei der Konfiguration und Verwendung des Datenblocks erfolgreich unterstützt wird. Da Blockdefinitionen von der DDE-Nutzerschnittstellenanwendung dynamisch erkannt werden, ist diese Funktion in die DDE-Nutzerschnittstellenanwendung selbst eingebettet. Daher können in einer Ausführungsform alle damit in Zusammenhang stehenden Funktionen, die für einen bestimmten Datenblock spezifisch sind, ebenfalls dynamisch geladen werden, wenn die DDE-Nutzerschnittstellenanwendung initialisiert wird. Anschließend wird eine Referenz zu der dynamisch geladenen Funktion, die zu dem bestimmten Datenblock gehört, in der Blockdefinition des bestimmten Datenblocks beibehalten und von der DDE-Nutzerschnittstellenanwendung verwendet, wenn eine Instanz der Blockdefinition des bestimmten Blocks konfiguriert wird. Weitere Details zu den Blockdefinitionen und den damit in Zusammenhang stehenden Funktionen werden in dieser Patentschrift weiter unten erläutert.

IN PROZESSELEITSYSTEM INTEGRIERTES ÜBERWACHUNGS- UND ANALYTIKSYSTEM FÜR INDUSTRIEPROZESSE

[0234] Wie aus der vorstehenden Erläuterung der **Fig. 1–Fig. 3** und **Fig. 4A–Fig. 4N-2** ersichtlich, stellt das verteilte Leistungsüberwachungs- und/oder Analytiksystem oder DAS **100** eine Plattform bereit, die Datenanalytikkonfiguration, Datenmodellierung, Ausführung von Datenmodellen, Datenvisualisierung und Streamingdienste für industrielle Prozessanlagen unterstützt. Wie bereits erläutert, besteht das Datenkommunikationsnetzwerk **112** der Analytikplattform **100**, obwohl verschiedene Knoten des Überwachungs- und Analytiksystems **100** in dem Prozesseleitsystem oder der Prozessleitanlage **5** einge-

bettet sind, typischerweise außerhalb bestehender DCS, PLS, Health-Überwachung und anderer Systeme der Prozessanlage **5**. In einer beispielhaften Konfiguration wird das Datenanalytiknetzwerk **112** über die Prozessanlage **5** gelegt, womit das Netzwerk **112** von den meisten oder allen herkömmlichen Prozessleit-Kommunikationsnetzwerken **10, 70 usw.** der Prozessanlage **5** getrennt wird. Herkömmliche Prozessleit-Kommunikationsnetzwerke werden weiterhin Prozessleitdaten übertragen, um die Steuerung der Verfahren und des Betriebs der Anlage **5** zu steuern, regeln, managen und/oder verwalten, während das Datenanalytiknetzwerk **112** zur Ermittlung von Wissen zu den Operationen der Prozessanlage **5**, z. B. deskriptives Wissen, prädiktives Wissen und/oder präskriptives Wissen, Analytikdaten überträgt (die Kopien der über das herkömmliche Prozessleit-Kommunikationsnetzwerk übertragenen Daten beinhalten können). Die Architektur des Datenanalytiksystems **100** macht es möglich, dass das Datenanalytiksystem **100** einfach bestehende Alt-Prozessanlagen **5** hinzugefügt oder in diese integriert werden kann.

[0235] Das Wissen zur Prozessanlage **5**, das von dem Datenanalytiksystem **100** ermittelt wurde, kann verwendet werden, um Parameter, Ausrüstung, Operationen und andere Bereiche der Prozessanlage **5** zu ändern, um die Leistung der Anlage **5** zu verbessern und das Auftreten von Störfällen, Ausfällen und anderen unerwünschten Zuständen zu verringern. Das Datenanalytiksystem **100** ermittelt in vorteilhafter Weise Wissen zu der Prozessanlage **5** in Echtzeit in Übereinstimmung mit den aktuellen Online-Operationen der Prozessanlage, so dass das ermittelte Wissen für die aktuellen Operationen der Prozessanlage **5** deskriktiv, prädiktiv und/oder präskriktiv ist. Auf diese Art bilden das Datenanalytiksystem **100** und die Prozessanlage **5** einen Regelkreis, um die Leistung der Prozessanlage **5** zu steuern, regeln und optimieren.

[0236] Ein beispielhaftes Blockdiagramm **400** der oberen Ebene des Regelkreises, der die Leistung der Prozessanlage **5** steuert, regelt und optimiert, ist in **Fig. 4O** dargestellt. Wie in **Fig. 4O** dargestellt, erzeugt, bearbeitet und überträgt die Online-Prozessanlage **5** über herkömmliche Prozessleit-Kommunikationsnetzwerke (z. B. Netzwerke **10, 70 usw.**) verschiedene Arten von Steuerdaten **402**, um einen oder mehr Industrieprozesse zu steuern oder regeln und Prozessanlagen-Ausgabedaten **405** zu erzeugen. Das verteilte Leistungsüberwachungs-/Analytiksystem für Industrieprozesse **100** ist mit der Prozessanlage **5** über verteilte Datenengines **102x** verbunden und in ihr integriert, die Analytikdaten **408** erfassen, die erzeugt werden, da die Prozessanlage **5** einen oder mehr Prozesse steuert oder regelt und so die Ausgabedaten **405** erzeugt. Wie bereits erläutert, können die erfassten Analytikdaten **408** Kopien

von Prozessleitdaten **402** der obersten oder einer höheren Ebene oder auch die Ergebnisse/Ausgabedaten von Analytik-Funktionen beinhalten, die lokal von den DDEs **102x** und/oder von Offline- und Online-Datenanalytikmodulen ausgeführt werden, die über das Datenstudio des Datenanalytiksystems **100** entwickelt wurden. In einigen Fällen beinhalten die erfassten Analytikdaten **408** neue Analytik-Funktionen, die über das Datenstudio erzeugt werden und/oder autonom von den DDEs **102x** erzeugt werden.

[0237] Das Datenanalytiksystem bzw. die Plattform **100** empfängt, erzeugt und überträgt Analytikdaten **408** und führt auf ihnen Operationen durch, um eine Analytikausgabe **410** zu erzeugen. Die Analytikausgabe **410** kann ermitteltes Wissen zur Prozessanlage **5** enthalten, wie deskriptives Wissen zu den aktuellen Operationen der Prozessanlage **5**, Wissen, das basierend auf den aktuellen Operationen der Prozessanlage **5** das Auftreten von Störfällen, Ausfällen, Zeitintervallen, Leistung, Ereignissen usw. prognostiziert, und/oder Wissen, das eine oder mehr präskriptive Maßnahmen vorschreibt, die ergriffen werden können, um unerwünschte Merkmale der aktuellen Anlagenoperationen abzuschwächen und/oder basierend auf den aktuellen Operationen der Prozessanlage **5** die Wahrscheinlichkeit des Auftretens von unerwünschten prognostizierten Störfällen, Ausfällen, Zeitintervallen, Leistungen, Ereignissen usw. abzuschwächen. In einigen Ausführungsformen beinhaltet das ermittelte Wissen **410** Wissen zu den Analytikdaten **408** selbst. So kann das ermittelte Analytikwissen **410** zum Beispiel einen optimalen Offline-Datensatz beinhalten, der verwendet werden kann, um verschiedene Offline-Datenmodule zu evaluieren, eine Abfolge von Aktionen, die wiederholt von Data Engineers verwendet wurden und die als ein neues Datenanalytik-Nutzerverfahren definiert wurden, eine neue Konfiguration von Datenblöcken, die als eine neue Datenanalytiktechnik definiert wurden, usw. Tatsächlich wurde durch den Regelkreis **400** während der Tests für einen Prototypen des Datenanalytiksystems **100**, das in eine Raffinerieprozessanlage integriert war, eine neue Datenanalytiktechnik für Industrieprozessanlagen entdeckt und wird in einem späteren Abschnitt weiter unten erläutert.

[0238] Zumindest Teile des in der Analytikausgabe **410** enthaltenen Wissens können einer Nutzerschnittstelle (Human-Machine-Interface, HMI) **412**, wie der Datenanalytik-Nutzerschnittstellenanwendung, bereitgestellt werden. Über die HMI **412** kann ein Nutzer das ermittelte Wissen **410**, das von dem Datenanalytiksystem **100** in Bezug auf die Prozessanlage **5** erzeugt wurde, durchsuchen. In einigen Fällen umfasst das Durchsuchen des ermittelten Wissens **410** die Verwendung des Datenstudios durch den Nutzer, um zusätzliche Datenanalytikmodule zu erstellen und auszuführen, um zusätzliches ermitteltes Wissen **410** für eine Untersuchung zu erzeu-

gen. Der Nutzer kann basierend auf der Gesamtheit des ermittelten Wissens **410** irgendwann eine oder mehr Änderungen **418** an einem oder mehr Werten, Parametern, Ausrüstungsgegenständen, Komponenten, Regelkreisen und/oder anderen aktuellen Operationen der Online-Prozessanlage **5** vornehmen und somit die Leistung und Ausgabedaten **405** der Prozessanlage **5** optimieren und/oder das Auftreten von Störfällen, Ausfällen und anderen unerwünschten Bedingungen verhindern oder ihnen zuvorkommen.

[0239] In einigen Situationen wird zumindest ein Teil des ermittelten Wissens, das in der Analytikausgabe **410** enthalten ist, direkt der Prozessanlage **5** bereitgestellt, um, wie in **Fig. 4O** durch den Pfeil **420** dargestellt, eine oder mehr präskriptive Änderungen vorzunehmen oder zu implementieren. So kann das ermittelte Wissen **410** zum Beispiel eine präskriptive Aktion enthalten, die eine Änderung an einem Sollwert, eine Änderung an einer Konfiguration eines Controllers, eine Änderung an einer Priorität einer Prozessleitmitteilung oder eine Änderung an einem anderen Wert, Parameter, einer anderen Konfiguration usw. umfasst. Das Datenanalytiksystem **100** kann automatisch oder direkt die in der Prozessanlage **5** vorzunehmende Änderung herunterladen oder anderweitig bewirken, ohne dass ein Nutzereingriff **418** erforderlich ist, und somit basierend auf dem ermittelten Wissen **410** die Leistung und Ausgabedaten **405** der Prozessanlage **5** automatisch optimieren und/oder das Auftreten von Störfällen, Ausfällen und anderen unerwünschten Bedingungen automatisch verhindern oder ihnen zuvorkommen. Es sei darauf hingewiesen, dass diese Strecke des Regelkreises **400** die Online-Prozessanlage **5** automatisch aktualisiert, um deren aktuelle Operationen in Echtzeit in dem Maße, wie das Analytiksystem **100** basierend auf den aktuellen Operationen der Prozessanlage **5** aktuelle Analytikdaten **408** verarbeitet und ermitteltes Wissen **410** erzeugt, zu verbessern. Daher werden hier im Gegensatz zu bestehenden Analytiktechniken, bei denen Stunden oder sogar Tage für die Berechnung, Ermittlung und Implementierung präskriptiver Aktionen benötigt werden, die vom Analytiksystem **100** erzeugten präskriptiven Aktionen für die aktuellen Betriebsbedingungen der Anlage **5** optimiert und unverzüglich auf die Prozessanlage **5** angewendet und werden in einigen Fällen angewendet, bevor der Nutzer eine unerwünschte oder erwünschte Bedingung überhaupt wahrnimmt.

[0240] Natürlich benötigt der Regelkreis **400** für eine optimale Implementierung von präskriptiven Aktionen in der Prozessanlage **5** als Ganzes sowohl die autonome Strecke **420** als auch die HMI-Regelstrecke **418**. Einige präskriptive Aktionen erfordern zum Beispiel einen menschlichen Eingriff, wie den Austausch eines Ventils, Sensors oder einer anderen Komponente.

[0241] Wie bereits erläutert, wird durch die Datenanalytikplattform **100** das Streamen von Analytikdaten zwischen einer Vielzahl von eingebetteten verteilten Datenengines (DDEs) **102x** ermöglicht. Alle Arten von Daten der obersten Ebene, die von der Prozessanlage **5** bei der Steuerung des Prozesses erzeugt werden, können gestreamt werden. Zum Beispiel können die Prozessleitdaten, die in von Prozessleitgeräten (z. B. Controller, E/A-Karten, Feldgeräte usw.) erzeugten Signalen enthalten sind, die typischerweise zur Steuerung oder Regelung des Prozesses über herkömmliche Prozessleit-Kommunikationsnetzwerke übermittelt werden, auch im Datenanalytiknetzwerk **112** auch zur Datenanalytik gestreamt werden. Damit kann der Inhalt oder die Daten, die in Prozessleitsignalen, Sensorsignalen usw. enthalten sind, kopiert und/oder vervielfältigt und über das Datenanalytiknetzwerk **112** gestreamt werden. Außerdem können Prozessleitdaten der zweiten (oder einer oberen) Ebene, die über herkömmliche Prozessleit-Kommunikationsnetzwerke erzeugt und übermittelt werden (z. B. Daten, die in einer Prozessleit-Datenbank oder einem Historian zu historisieren sind, Prozessleit-Konfigurationen, Prozessleit-Nutzerschnittstellenbefehle, Prozessleit-Netzwerkmanagementsignale und andere Signale, die bei der Steuerung und Regelung eines Prozesses üblicherweise über herkömmliche Prozessleit-Kommunikationsnetzwerke gesendet und empfangen wurde) in dem Datenanalytiknetzwerk **112** zur Datenanalytik gestreamt werden. Darüber hinaus kann das Datenanalytiknetzwerk **112** berechnete Daten oder Ergebnisse der Analytik-Funktionen oder -Techniken, die auf den herkömmlichen Prozessleitdaten und/oder auf berechneten Analytikdaten durchgeführt wurden, streamen.

[0242] Daher wird die Datenanalytik-Plattform **100** teilweise dadurch in die Prozessanlage **5** integriert, dass sie lokal an Datenquellen der Prozessanlage **5** gebunden wird und/oder indem sie nah bei den Datenquellen in der Prozessanlage **5** Datenanalytdienste oder -funktionen bereitstellt und gleichzeitig weitreichendere Prognose und Optimierungen bereitstellt. Nur die Daten, die für die Unterstützung von Prognosen und Optimierungen außerhalb des Regelkreises benötigt werden, müssen an Analytikserver einer höheren Ebene/mit besseren Fähigkeiten und Cloud-Plattformen des Analytiksystems **100** übertragen werden.

[0243] Wie bereits erläutert, sind einige DDEs aus der Vielzahl der DDEs **102x** des Datenanalytiksystems **100** in physische Geräte innerhalb der Prozessanlage **5** eingebettet. Zum Beispiel sind einige DDEs aus der Vielzahl der DDEs **102x** jeweils in einem entsprechenden Prozessleit-Gerät (z. B. Controller **11**; E/A-Karte **26, 28**; Feldgeräte **15–16, 18–21, 42A, 42B, 44** usw.) und/oder einem entsprechenden Netzwerk-Routing- und Managementgerät (z. B.

Gateways **35, 75, 78**; Adapter **52a**; Router **58**; Zugangspunkte **55a, 72** usw.) fest eingebaut. In einer Ausführungsform kann eine DDE statt in ein eine Einheit darstellendes Gerät oder derartige Komponente der Prozessanlage **5** integriert zu sein, ein Einzelgerät sein, das mit einer Prozesskommunikationsverbindung oder einem Prozesskommunikationsnetzwerk der herkömmlichen Art (z. B. Netzwerk **10, 70**, PLS-Netzwerk, Überwachungssystem zur Sicherheits-Health usw.) als optisches Mess-, Monitor- oder Voyeur-Gerät verbunden ist und ausgewählte Prozessleitdaten kopieren/vervielfältigen kann, die über die herkömmlichen Prozesskommunikationsverbindungen oder -netzwerke zum Streamen über das Datenanalytiknetzwerk **112** zur Datenanalytik übermittelt werden.

[0244] In einigen Szenarien kann aus mehr als einer DDE ein lokales Datenanalytik-Cluster gebildet werden, bei dem sich Datenquellen registrieren und dem sie sich anschließen können und über das Analytikdaten, die von registrierten Datenquellen erzeugt wurden, als Analytikdaten in das Datenanalytiknetzwerk **112** empfangen werden können. In einem Beispiel bildet ein Menge DDEs ein lokales Cluster und gibt das Vorhandensein des lokalen Clusters bekannt. Das lokale Cluster erscheint, obwohl es mehrere DDEs umfasst, die über das Datenanalytiknetzwerk **112** untereinander verbunden sind, als ein eine Einheit darstellender Knoten oder eine einzelne DDE-Instanz **150** des Datenanalytiknetzwerks **112**. Nachdem eine neue Datenquelle auf dem Gelände der Prozessanlage **5** aktiviert oder hochgefahren wurde, führt die Datenquelle Scans nach lokalen Cluster-Bekanntmachungen durch oder sucht diese. Die Datenquelle kann die Bekanntmachung des lokalen Clusters feststellen und sich beim lokalen Cluster registrieren und damit sich dem Datenanalytiknetzwerk **112** anschließen. Anschließend können registrierte Datenquellen vom Analytiksystem **100** eine Bandbreite anfordern (z. B. über ihr lokales Cluster) und können Daten in ihrem lokalen Cluster, einem anderen Knoten des Datenanalytiknetzwerk **112** oder direkt im Analytiksystem **100** zur Veröffentlichung bringen. Da lokale Cluster als Instanzen von verteilten Datenengines **150** erscheinen, erhalten lokale Cluster über entsprechende Big-Data-Empfänger **160** lokal erzeugte Analytikdaten (z. B. von registrierten Quellen und/oder anderen Quellen) und speichern die lokal erzeugten Analytikdaten auf einem entsprechenden lokalen Permanentenspeichermedium **155**. Zusätzlich können lokale Cluster zumindest einen Teil der gespeicherten Analytikdaten abrufenden Anwendungen (z. B. einer anderen Datenanalytik-Anwendung, einem Datenmodul, einer Nutzerschnittstelle, dem Datenstudio, einer anderen Anwendung usw.) z. B. über einen entsprechenden Big-Data-Abbrufservice **165** bereitstellen. In einigen Fällen können lokale Cluster z. B. über entsprechende Big-Data-Abbrufanalyser **170** auf gespeicherten Daten, die

von registrierten Datenquellen erhalten wurden, und/oder auf anderen innerhalb des Datenanalytiknetzwerks **112** gestreamten Daten entsprechende lokale Datenanalytik-Funktionen ausführen. Das Ergebnis oder die Ausgabe einer lokalen Datenanalytik-Funktion kann lokal in einem entsprechenden Big-Data-Speichermedium **155** gespeichert werden und/oder kann zu einer oder mehr DDEs außerhalb des lokalen Clusters, z. B. einer anderen lokalen DDE oder einem anderen lokalen Cluster, einem zentralen Datencluster oder einer Anwendung, die ein Abonnement hat, gestreamt werden, z. B. über das entsprechende eine oder die entsprechenden mehr Datenanalytiknetzwerk-Schnittstellen **175** des lokalen Clusters.

[0245] In einigen Ausführungsformen registrieren sich lokale Cluster bei mehr zentralen Clustern, die Analytikdaten höherer Ebene erhalten und speichern und die entsprechende Datenanalytik-Funktionen höherer Ebene ausführen. Zum Beispiel kann ein bestimmtes zentrales Cluster (z. B. ein regionales Cluster) von allen lokalen Clustern in einem bestimmten Bereich der Anlage **5** gestreamte Daten erhalten und auf diesen Datenanalytik ausführen. In einem anderen Beispiel kann ein bestimmtes zentrales Cluster von allen Regelkreisen mit einer bestimmten Konfiguration innerhalb der Anlage **5** gestreamte Daten erhalten und auf diesen Datenanalytik ausführen. Ähnlich wie ein lokales Cluster erscheint ein zentrales Cluster im Datenanalytiknetzwerk **112** als ein einzelner Knoten oder eine einzelne Instanz einer DDE **150**.

[0246] Daher erlaubt es die Fähigkeit des verteilten Leistungsüberwachungs- und Analytiksystems für Industrieprozesse **100**, DDEs und/oder Cluster von DDEs auf verschiedenen Stufen oder Ebenen örtlicher Verteilung bereitzustellen, Störfallerkennung und -prognosen nah an den Datenquellen der Prozessanlage **5** auszuführen und somit Echtzeit-Störfallerkennung und -prognosen, die mit den von den Online-Datenquellen erzeugten Echtzeit-Daten korrespondieren, bereitzustellen. Eine derartige mehrstufige oder aus Clustern gebildete Architektur ermöglicht darüber hinaus die Speicherung von Analytikdaten im gesamten System **100** und tatsächlich in der gesamten Prozessanlage **5**, da das Analytiksystem **100** darin eingebettete Bereiche umfasst. Darüber hinaus stellt das Datenanalytiksystem **100** die Fähigkeit bereit, auf Daten, egal auf welcher DDE sie gespeichert sind, zuzugreifen, und stellt für Analytik-Konfiguration, Diagnose und Überwachung von Leistung, Störfällen und Ausfällen ein zentrales System bereit.

[0247] In einer Ausführungsform stellen ein oder mehr Cluster für einen Nutzer oder Systemadministrator einen zentralen Ort bereit, um Regeln für die Sicherung oder Historisierung von Analytikdaten von der Vielzahl von DDEs **102x** aufzustellen. In einem

Beispiel befinden sich das eine oder die mehr zentralen Cluster auf dem Gelände der Prozessanlage **5** und stellen der Prozessanlage **5** insgesamt Dienste bereit. In einem anderen Beispiel befinden sich das eine oder die mehr zentralen Cluster in der Cloud oder werden in ihr gehostet und stellen der Prozessanlage **5** insgesamt Dienste bereit. In einigen Konfigurationen stellen das (die) zentrale(n) Cluster, die Sicherung und Historisierung der Prozessanlage **5** bereitstellen, auch anderen Prozessanlagen Dienste bereit. Im Allgemeinen werden Analytikdaten, wie bereits erläutert, auf allen DDE oder Clustern permanent gespeichert, z. B. in den entsprechenden Big-Data-Speicherbereichen **155**. In einigen Fällen können einige Analytikdaten, die von der DDE erhalten oder erzeugt werden, auf der DDE oder dem Cluster für einen schnelleren Zugriff gecachet werden (z. B. basierend auf Kriterien wie die Anzahl der Abonnenten, Häufigkeit von Veröffentlichungen und Übertragungen usw.), aber im Allgemeinen werden alle lokalen Analytikdaten auf ein Permanentspeichermedium **155** auf der DDE oder dem Cluster bewegt. In regelmäßigen Abständen oder zu vordefinierten Zeiten können die permanent auf den jeweiligen DDE gespeicherten Analytikdaten auf Langzeit-Datenanalytikspeichermedien gesichert, gespeichert oder historisiert werden, z. B. einem zentralen Analytikdaten-Historian, auf den über das Datenanalytiknetzwerk **112** zugegriffen werden kann. Der Datenanalytik-Historian kann zum Beispiel ein Datenanalytik-Historien für die gesamte Anlage sein und/oder in der Cloud gehostet werden. Der Datenanalytik-Historian ist jedoch typischerweise von Prozessleitdaten-Historians unabhängig und getrennt. Über eine Nutzerschnittstelle, die auf das eine zentrale oder die mehr zentralen Cluster zugreift, kann ein Nutzer oder Systemadministrator einen Satz Regeln definieren, die angeben, wie oft jeder bestimmte DDE-Speicherbereich **155** gesichert wird, wann die Sicherung stattfindet, wo die Analytikdaten historisiert werden usw.

[0248] In einer Ausführungsform stellen ein oder mehr Cluster für einen Nutzer oder Systemadministrator einen zentralen Ort bereit, um Regeln für die Vervielfältigung von Analytikdaten unter den Clustern, Knoten und/oder DDEs **102x** des Datenanalytiksystems **100** aufzustellen. Zum Beispiel kann mit dem Satz Vervielfältigungsregeln definiert werden, welche bestimmten Analytikdaten unter den bestimmten Knoten des Datenanalytiknetzwerks **112** zu vervielfältigen sind und wie oft bestimmte Analytikdaten unter den bestimmten Knoten zu aktualisieren sind. So kann mit dem Satz Vervielfältigungsregeln definiert werden, wie oft die Ergebnisse von Analytik-Funktion A, die vom Knoten A ausgeführt wird, an den Knoten B-D zu vervielfältigen sind (z. B. durch Streaming), so dass die vervielfältigten Ergebnisse den Knoten B-D bei der Ausführung ihrer jeweiligen Datenanalytik-Funktionen zur Verfügung stehen. Ähnlich wie bei den Regeln zur Historisierung und Si-

cherung kann der Satz Vervielfältigungsregeln von einem Nutzer oder Systemadministrator über eine Nutzerschnittstelle, die auf das eine zentrale oder die mehr zentralen Cluster zugreift, definiert werden. Zum Beispiel kann der Systemadministrator die Nutzerschnittstelle verwenden, um den Satz Vervielfältigungsregeln zu administrieren und der Satz Vervielfältigungsregeln kann auf dem einen zentralen oder den mehr zentralen Clustern gespeichert sein. Bei dem einen zentralen oder den mehr zentralen Clustern, die die Administration der Analytikdaten-Vervielfältigung bereitstellen, kann es sich um dieselbe Menge von Clustern, die die Administration von Analytikdaten-Sicherung/-Historisierung bereitstellt, oder eine andere Menge von Clustern handeln.

[0249] In einer Ausführungsform stellen der eine zentrale oder die mehr zentralen Cluster einen Standortdienst, einen Streamingdienst (z. B. den Streamingdienst **115** aus Fig. 1), ein Datenwörterbuch und/oder andere Funktionalität bereit, die von lokalen Clustern, anderen zentralen Clustern und individuellen DDEs **102x** verwendet werden kann, um einen Austausch von Analytikdaten zwischen ihnen einzurichten. Bei dem einen zentralen oder den mehr zentralen Clustern, die den Streamingdienst, das Datenwörterbuch und/oder die andere Funktionalität bereitstellen, kann es sich um dieselbe Menge an Clustern, die die Administration der Analytikdaten-Vervielfältigung und/oder die Administration von Analytikdaten-Sicherung/-Historisierung bereitstellt, oder eine andere Menge von Clustern handeln. Ein Systemadministrator oder Nutzer kann eine Nutzerschnittstelle verwenden, um auf den einen oder die mehr zentralen Cluster zuzugreifen, um die Dienste, Wörterbücher und die andere Funktionalität zu administrieren, die verwendet wird, um einen Austausch von Analytikdaten zwischen Clustern und Knoten des Datenanalytiknetzwerks **112** einzurichten und bereitzustellen.

[0250] Fig. 4P zeigt ein beispielhaftes Verfahren **450** zur Bereitstellung von lokalisierten Datenanalytikdiensten. In einer Ausführungsform führt eine verteilte Datenengine **102x** oder ein Cluster aus DDEs zumindest einen Teil des Verfahrens **450** aus. Im Allgemeinen kann zumindest ein Teil des Verfahrens **450** von jedem beliebigen Teil eines verteilten Leistungsüberwachung- und/oder Analytiksystems für Industrieprozesse, wie das System **100**, ausgeführt werden. Tatsächlich kann das Verfahren **450** unter Verwendung eines jeden beliebigen Teils eines jeden Systems, einer jeden Vorrichtung, eines jeden Geräts und/oder einer jeden Technik, die in dieser Patentschrift erläutert werden, implementiert werden. Um die Erläuterung zu vereinfachen, wird das Verfahren **450** jedoch nachfolgend in Bezug auf ein Cluster aus einer oder mehr DDEs **102x** des Analytiksystems **100** beschrieben, das als eine eine Einheit darstellende DDE-Instanz **150** erscheint, obwohl dies nur eine

Ausführungsform von vielen ist und nicht einschränkend sein soll.

[0251] Das beispielhafte Verfahren **450** umfasst die Bekanntmachung des Vorhandenseins eines Clusters (Block **452**) durch das Cluster. Das Cluster ist Teil eines Datenanalytiknetzwerks **112** einer Prozessleitanlage **5**, die online ist und betrieben wird, um einen Prozess zu steuern oder regeln, und die Bekanntmachungen des Clusters können über das Datenanalytiknetzwerk **112**, über ein oder mehr andere Netzwerke, über ein Funkprotokoll für kurze Distanzen, wie Bluetooth, NFC usw. oder andere geeignete Mittel zur Bekanntmachung übermittelt werden.

[0252] Das Verfahren **450** umfasst des Weiteren den Empfang einer Antwort auf die Bekanntmachung des Clusters von einer Datenquelle durch das Cluster. Zum Beispiel kann die Antwort von der Datenquelle ein Antrag der Datenquelle zur Registrierung beim Cluster sein und das Verfahren **450** kann die Registrierung der Datenquelle beim Cluster und/oder beim Datenanalytiknetzwerk **112** (Block **455**) umfassen, womit die Datenquelle an das Datenanalytiknetzwerk **112** angeschlossen wird. Die Datenquelle kann ein Gerät oder eine Komponente sein, das oder die Teil der Prozessanlage **5** ist und das oder die bei der Steuerung des Prozesses durch die Online-Prozessanlage **5** Daten erzeugt. In einem Beispiel erzeugt die Datenquelle bei der Steuerung des Prozesses kontinuierliche Zeitreihendaten.

[0253] Zusätzlich umfasst das Verfahren **450** den Empfang der von der Datenquelle erzeugten Daten (Block **458**) durch das Cluster. Zum Beispiel kann die Datenquelle die von ihr erzeugten Daten über das Datenanalytiknetzwerk **112** zum Cluster streamen. Zusätzlich oder ersatzweise kann das Cluster bestimmte Daten abonnieren, die von der Datenquelle erzeugt und veröffentlicht werden. In einigen Ausführungsformen kann das Cluster die Datenquelle nach bestimmten Daten abfragen, z. B. indem das in einem späteren Abschnitt dieser Anmeldung beschriebene Abfrageverfahren verwendet wird. Wenn die Datenquelle ein Gerät oder eine Komponente ist, die typischerweise Prozessleitdaten über ein oder mehr herkömmliche Prozessleit-Kommunikationsnetzwerke (z. B. Netzwerke **10, 70**) überträgt, kann die Datenquelle damit fortfahren, Prozessleitdaten wie üblich über das eine oder die mehr herkömmlichen Prozessleit-Kommunikationsnetzwerke zu übertragen, während es darüber hinaus über das Datenanalytiknetzwerk **112** eine Kopie oder Vervielfältigung des Inhalts der übertragenen Prozessleitdaten zum Cluster überträgt.

[0254] Nach Empfang der von der Datenquelle erzeugten Daten kann das Cluster zumindest einige der empfangenen Daten cachen, z. B. wenn die empfangenen Daten eine oder mehr Caching-Bedingungen

erfüllen. Die gesamten empfangenen Daten werden jedoch unabhängig davon, ob empfangene Daten ge- cacht werden oder nicht, lokal in dem Permanentenspeichermedium **155** des Clusters gespeichert.

[0255] Zusätzlich umfasst das Verfahren **450** das Streamen von zumindest einem Teil der von der Datenquelle erzeugten Daten an einen Datenverbraucher (Block **460**) durch das Cluster über das Datenanalytiknetzwerk **112**. Ein Datenverbraucher kann zum Beispiel ein anderer Analytik-Knoten oder ein anderes Analytik-Cluster, eine Datenanalytik-Anwendung, eine Nutzerschnittstelle und/oder Nutzerschnittstellenanwendung, ein Online-Datenanalytikmodul, eine andere Anwendung und/oder eine Datenbank oder ein Daten-Historian sein. Der Datenverbraucher kann an die Datenquelle und/oder den bestimmten Datenstrom gebunden sein, der zumindest einige der von der Datenquelle erzeugten Daten beinhaltet. Die bestimmten Daten, von denen erwünscht ist, dass sie von dem Datenverbraucher empfangen werden, sind typischerweise Zeitreihendaten und können kontinuierliche Daten sein.

[0256] Der Datenverbraucher kann das Cluster nach bestimmten von der Datenquelle erzeugten Daten abfragen, z. B. indem er das in einem späteren Abschnitt dieser Anmeldung beschriebene Abfrageverfahren verwendet, und das Streaming (Block **460**) kann als Antwort auf die von dem Datenverbraucher erzeugte Abfrage erfolgen. In einer Ausführungsform kann der Datenverbraucher bestimmte Daten abonniere n, die von der Datenquelle erzeugt und von der Datenquelle oder dem Cluster veröffentlicht werden. Auf jeden Fall ruft das Cluster die gewünschten Daten, die von der Datenquelle erzeugt wurden, aus seinem Cache oder seinem lokalen Permanentenspeichermedium **155** ab und streamt die Daten über das Datenanalytiknetzwerk **112** (Block **460**) zum Datenverbraucher.

[0257] In einigen Ausführungsformen umfasst das Verfahren **450** das Ausführen von einer oder mehr Datenanalytik-Funktionen oder -Techniken auf zumindest einem Teil der von der Datenquelle erzeugten Daten (in Fig. 4P nicht gezeigt) durch das Cluster. Die eine oder mehr Analytik-Funktionen oder -Techniken können deskriptive Analytik, prädiktive Analytik und/oder präskriptive Analytik umfassen. Die eine oder mehr Analytik-Funktionen oder -Techniken können von dem Cluster auf Anforderung zum Beispiel eines Online-Datenanalytikmoduls, einer anderen Datenanalytik-Anwendung oder eines Nutzers ausgeführt werden. In einem anderen Beispiel können die eine oder mehr Analytik-Funktionen oder -Techniken automatisch initialisiert und von dem Cluster ausgeführt werden. Die Ergebnisse der Ausführung der einen oder der mehr Analytik-Funktionen oder -Techniken können an die anfragende Partei ausgegeben und/oder lokal auf dem Cluster in ei-

nem Permanentdatenspeichermedium **155** gespeichert werden. In einigen Ausführungsformen werden die eine oder mehr Analytik-Funktionen oder -Techniken auf Daten ausgeführt, die von mehreren, beim Cluster registrierten Datenquellen erzeugt wurden und die auf dem Cluster gespeichert (und wahlweise gecacht) wurden.

[0258] Da das Cluster von der Datenquelle in der Prozessanlage **5** erzeugte Daten der ersten Ebene empfängt und sich nah an diesen Datenquellen in der Prozessanlage **5** befindet, wird das Cluster in der vorliegenden Patentschrift als „lokales“ Cluster bezeichnet. In einigen Ausführungsformen umfasst das Verfahren **450** die Feststellung einer von einem zentralen Cluster, wie einem regionalen Cluster, einem Bereichscluster, einem Cluster für die Anlage insgesamt usw., erzeugten Bekanntmachung durch das lokale Cluster. Das lokale Cluster kann sich beim zentralen Cluster registrieren und zumindest einen Teil der lokalen in seinem Cache und/oder seinem Permanentdatenspeichermedium **155** gespeicherten Analytikdaten an das zentrale Cluster streamen. Zum Beispiel können die Analytikdaten, die zum zentralen Cluster gestreamt werden, Daten enthalten, die von der Datenquelle erzeugt wurden und/oder das Ergebnis von einer oder mehr von dem lokalen Cluster ausgeführten Analytik-Funktionen sind. Das lokale Cluster kann Daten gemäß einer von dem zentralen Cluster initiierten Abfrage an das zentrale Cluster streamen und/oder das lokale Cluster kann Daten veröffentlichen, die das zentrale Cluster abonniert hat.

[0259] In einigen Ausführungsformen umfasst das Verfahren **450** die Historisierung und Sicherung der Daten, die im lokalen Permanentdatenspeichermedium **155** des Clusters gespeichert sind (nicht gezeigt in **Fig. 4P**). Die Datensicherung kann von dem Cluster initiiert werden oder die Datensicherung kann von einem zentralen Analytikdaten-Historian initiiert werden. In einer Ausführungsform können das Cluster und der zentrale Analytikdaten-Historian eine Veröffentlicher-Abonent-Beziehung haben. Der Zeitpunkt und/oder die Menge der vom Cluster zu sichernden Daten können in einer Ausführungsform von einem Systemadministrator des Systems **100** definiert werden. Die zu sichernden Daten können für eine Langzeitspeicherung und -sicherung vom Cluster an einen Analytikdaten-Historian übermittelt und/oder gestreamt werden.

[0260] **Fig. 4Q** zeigt ein beispielhaftes Verfahren **470** zur Bereitstellung von lokalisierter Datenanalytikdiensten. In einer Ausführungsform führt eine verteilte Datenengine **102x** oder ein Cluster aus DDEs **102x** zumindest einen Teil des Verfahrens **470** aus. Im Allgemeinen kann zumindest ein Teil des Verfahrens **470** von jedem beliebigen Teil eines verteilten Leistungsüberwachung-s und/oder Analytiksystems für Industrieprozesse, wie das System **100**, aus-

geführt werden. Tatsächlich kann das Verfahren **470** unter Verwendung eines jeden beliebigen Teils eines jeden Systems, einer jeden Vorrichtung, eines jeden Geräts und/oder einer jeden Technik, die in dieser Patentschrift erläutert werden, implementiert werden. Um die Erläuterung zu vereinfachen, wird das Verfahren **470** jedoch nachfolgend in Bezug auf ein Cluster aus einer oder mehr DDEs **102x** des Systems **100** beschrieben, das als eine eine Einheit darstellende DDE-Instanz **150** erscheint, obwohl dies nur eine Ausführungsform von vielen ist und nicht einschränkend sein soll.

[0261] Das beispielhafte Verfahren **470** umfasst die Bekanntmachung des Vorhandenseins des zentralen Clusters (Block **472**) durch das Cluster. Das zentrale Cluster unterscheidet sich vom lokalen Cluster aus **Fig. 4P**, da sich Datenquellen typischerweise nicht direkt beim zentralen Cluster registrieren. Stattdessen registrieren sich in den meisten Fällen lokale Cluster oder andere Arten von Clustern bei zentralen Clustern und zentrale Cluster führen Datenanalytik-Funktionen auf den Daten durch, die von solchen anderen Clustern empfangen wurden. Ähnlich wie lokale Cluster sind zentrale Cluster jedoch Teil eines Datenanalytiknetzwerks **112** der Prozessleitanlage **5**, die online ist und betrieben wird, um einen Prozess zu steuern oder regeln, und die Bekanntmachungen des zentralen Clusters können über das Datenanalytiknetzwerk **112** gestreamt und/oder als Broadcast verschickt werden. Die Bezeichnung ein „zentrales“ Cluster bezieht sich in dieser Patentschrift im Allgemeinen auf ein Cluster, bei dem sich andere Cluster registrieren. Ein zentrales Cluster kann Dienste für einen Regelkreis (z. B. ein „Regelkreis“-Cluster), einen Bereich der Prozessanlage (z. B. ein „Bereichs“-Cluster), eine Region einer Prozessanlage (z. B. ein „regionales“ Cluster), die gesamten Prozessanlage (z. B. ein „Anlagen“-Cluster), mehrere Prozessanlagen (ein „Unternehmens“-Cluster) und weitere derartige Cluster ausführen. Typischerweise umfasst ein zentrales Cluster eine Vielzahl von DDEs, die über ein Datenanalytiknetzwerk **112** untereinander verbunden sind und als eine einzige sich als Einheit darstellende Instanz **150** erscheinen.

[0262] Das Verfahren **470** umfasst des Weiteren den Empfang einer Antwort auf die Bekanntmachung des zentralen Clusters durch das zentrale Cluster von einem anderem Cluster. Zum Beispiel kann die Antwort von dem anderen Cluster ein Antrag des anderen Clusters zur Registrierung beim zentralen Cluster sein und das Verfahren **470** kann die Registrierung des anderen Clusters beim zentralen Cluster (Block **475**) umfassen. Das sich registrierende Cluster kann ein lokales Cluster, wie das lokale Cluster aus **Fig. 4P**, oder das sich registrierende Cluster kann ein anderes zentrales Cluster sein.

[0263] Zusätzlich umfasst das Verfahren **470** den Empfang von gestreamten Daten beim zentralen Cluster von einem registrierten Cluster (Block **478**). Zum Beispiel kann das registrierte Cluster einen Teil seiner gespeicherten und/oder gecachten Daten über das Datenanalytiknetzwerk **112** zum zentralen Cluster streamen. Die gestreamten Daten beinhalten typischerweise Zeitreihendaten, die zum Beispiel kontinuierliche Daten sein können. Zusätzlich oder ersatzweise kann das zentrale Cluster bestimmte Daten abonnieren, die von dem registrierten Cluster veröffentlicht werden. In einigen Ausführungsformen fragt das zentrale Cluster das registrierte Cluster nach bestimmten Daten ab, z. B. indem das in einem späteren Abschnitt dieser Anmeldung beschriebene Abfrageverfahren verwendet wird. Die gestreamten Daten können Daten umfassen, die von dem registrierten Cluster erzeugt wurden, und/oder können Daten umfassen, die zu dem registrierten Cluster von Datenquellen und/oder anderen Clustern gestreamt wurden.

[0264] In einigen Ausführungsformen cacht das zentrale Cluster zumindest einige der gestreamten Daten, die es empfängt, z. B. wenn die empfangenen Daten eine oder mehr Caching-Bedingungen erfüllen. Die gesamten empfangenen Daten werden jedoch unabhängig davon, ob Daten, die im zentralen Cluster empfangen werden, gecacht werden oder nicht, lokal in dem Permanentsspeichermedium **155** des zentralen Clusters gespeichert.

[0265] Im Block **480** umfasst das Verfahren **470** das Ausführen von einer oder mehr Datenanalytik-Funktionen oder -Techniken auf zumindest einem Teil der empfangenen, gestreamten Daten (z. B. die Daten, die im Cache und/oder seinem Permanentsspeicherbereich **155** gespeichert sind). Die eine oder mehr Analytik-Funktionen oder -Techniken können deskriptive Analytik, prädiktive Analytik und/oder präskriptive Analytik umfassen. Die eine oder mehr Analytik-Funktionen oder -Techniken können zum Beispiel von dem zentralen Cluster auf Anforderung zum Beispiel eines Online-Datenanalytikmoduls, einer anderen Datenanalytik-Anwendung oder eines Nutzers ausgeführt werden. In einem anderen Beispiel können die eine oder mehr Analytik-Funktionen oder -Techniken automatisch und/oder in regelmäßigen Abständen von dem zentralen Cluster ausgeführt werden. Die Ergebnisse oder die Ausgabe der Ausführung der einen oder der mehr Analytik-Funktionen oder -Techniken können an die anfragende Partei ausgegeben und/oder lokal auf dem Cluster in einem Permanentdatenspeichermedium **155** des zentralen Clusters gespeichert werden. In einigen Ausführungsformen werden die eine oder mehr Analytik-Funktionen oder -Techniken auf Daten ausgeführt, die von mehreren, registrierten Clustern erzeugt wurden und die auf dem zentralen Cluster gespeichert (und wahlweise gecacht) wurden.

[0266] Das Verfahren **470** umfasst das Streamen von zumindest einem Teil der Ergebnisse oder Ausgaben der einen oder mehr Analytik-Funktionen oder -Techniken an einen oder mehr Datenverbraucher durch das zentrale Cluster über das Datenanalytiknetzwerk **112** (Block **482**). Der eine oder die mehr Datenverbraucher können zum Beispiel einen anderer Analytik-Knoten oder ein anderes Analytik-Cluster, eine Datenanalytik-Anwendung, eine Nutzerschnittstelle und/oder Nutzerschnittstellenanwendung, ein Online-Datenanalytikmodul, eine andere Anwendung und/oder eine Datenbank oder ein Daten-Historian umfassen. Der eine oder die mehr Datenverbraucher können an das zentrale Cluster, an das bestimmte Ergebnisse/die bestimmte Ausgabe und/oder an einen bestimmten Datenstrom gebunden sein, der das bestimmte Ergebnis/die bestimmte Ausgabe beinhaltet.

[0267] Der eine oder die mehr Datenverbraucher können das zentrale Cluster nach dem bestimmten Ergebnis/der bestimmten Ausgabe abfragen, z. B. indem er das in einem späteren Abschnitt dieser Anmeldung beschriebene Abfrageverfahren verwendet, und das Streaming (Block **482**) kann als Antwort auf die von dem einen oder den mehr Datenverbrauchern erzeugte Abfrage erfolgen. Zusätzlich oder ersatzweise kann der Datenverbraucher die bestimmten Ergebnisse/Ausgaben abonnieren, die von dem zentralen Cluster veröffentlicht werden. Auf jeden Fall ruft das zentrale Cluster die gewünschten Daten aus seinem Cache oder seinem lokalen Permanentsspeichermedium **155** ab und streamt die gewünschten Daten über das Datenanalytiknetzwerk **112** zu dem einen oder den mehr Datenverbrauchern. In einigen Fällen werden die gewünschten Analytikergebnisse/Ausgabedaten nach ihrer Erzeugung durch die eine oder die mehr Datenanalytik-Funktionen oder -Techniken gestreamt.

[0268] In einigen Ausführungsformen umfasst das Verfahren **470** die Registrierung von einem oder mehr zusätzlichen lokalen Clustern bei dem zentralen Cluster und den Empfang entsprechender Analytikdaten von dem einem oder den mehr lokalen Clustern (nicht gezeigt in **Fig. 4Q**). In solchen Ausführungsformen kann die Ausführung der einen oder der mehr Datenanalytik-Funktionen oder -Techniken (Block **480**) die Ausführung der Datenanalytik-Funktionen oder -Techniken auf Analytikdaten, die beim zentralen Cluster von mehreren lokalen Clustern empfangen wurden, umfassen.

[0269] In einigen Ausführungsformen umfasst das Verfahren **470** die Historisierung und Sicherung der Daten, die im lokalen Permanentsspeichermedium (z. B. die in dem Big-Data-Speicherbereich **155** gespeicherten Daten) des zentralen Clusters gespeichert sind (nicht gezeigt in **Fig. 4Q**). Die Datensicherung kann von dem zentralen Cluster initiiert werden oder die Datensicherung kann von einem zentralen Ana-

lytikdaten-Historian initiiert werden. In einem Beispiel kann der zentrale Cluster im Analytikdaten-Historian eine Veröffentlicher-Abonent-Beziehung haben. Der Zeitpunkt und/oder die Menge der vom zentralen Cluster zu sichernden Daten können von einem Systemadministrator des Systems **100** definiert werden. Zum Beispiel kann der Systemadministrator eine Nutzerschnittstelle verwenden, um einen Satz Regeln für die Sicherung in dem System **100** und/oder in Teilen des Systems zu administrieren und der Satz Regeln kann auf dem zentralen Cluster gespeichert sein. Die zu sichernden Daten können für eine Langzeitspeicherung und -sicherung vom zentralen Cluster **152** und Analytikdaten-Historian übermittelt und/oder gestreamt werden.

[0270] Zusätzlich oder ersatzweise umfasst das Verfahren **470** in einigen Ausführungsformen die Bereitstellung von Regeln für die Vervielfältigung von Analytikdaten unter den Clustern, Knoten und/oder DDEs **102x** des Datenanalytiksystems **100**. Zum Beispiel kann mit dem Satz Vervielfältigungsregeln definiert werden, welche bestimmten Analytikdaten unter den bestimmten Knoten des Datenanalytiknetzwerks **112** zu vervielfältigen sind und wie oft bestimmte Analytikdaten unter den bestimmten Knoten zu aktualisieren sind. So kann mit dem Satz Vervielfältigungsregeln definiert werden, wie oft die Ergebnisse von Analytik-Funktion A, die vom Knoten A ausgeführt wird, an den Knoten B-D zu vervielfältigen sind (z. B. durch Streaming), so dass die vervielfältigten Ergebnisse den Knoten B-D bei der Ausführung ihrer jeweiligen Datenanalytik-Funktionen zur Verfügung stehen. Ähnlich wie bei den Regeln zur Historisierung und Sicherung kann der Satz Vervielfältigungsregeln von einem Systemadministrator des Systems **100** auf einem oder mehr zentralen Clustern definiert werden. Zum Beispiel kann der Systemadministrator die Nutzerschnittstelle verwenden, um den Satz Vervielfältigungsregeln zu administrieren und der Satz Vervielfältigungsregeln kann auf seinem einen zentralen oder seinen mehr zentralen Clustern gespeichert sein.

[0271] In einigen Ausführungsformen umfasst das Verfahren **470** die Feststellung einer von einem anderen zentralen Cluster erzeugten Bekanntmachung durch das zentrale Cluster. Das die Bekanntmachung sendende Cluster kann zum Beispiel ein regionales Cluster, ein Bereichs-Cluster oder ein Anlagen-Cluster sein, das sich auf dem Gelände der Prozessanlage **5** befindet. In einigen Fällen kann das die Bekanntmachung sendende Cluster in der Cloud gehostet werden und kann Analytikdienste für einen Teil der Anlage **5**, die gesamte Anlage **5** oder verschiedene Anlagen bereitstellen. Das zentrale Cluster kann sich beim die Bekanntmachung sendenden Cluster registrieren und in seinem Cache und/oder seinem Permanentdatenspeichermedium **155** gespeicherte Analytikdaten an das die Bekanntmachung sendende Clus-

ter streamen. Das zentrale Cluster kann zum Beispiel zumindest einige der Ergebnisse von Analytik-Funktionen streamen, die das zentrale Cluster ausgeführt hat. Das zentrale Cluster kann Daten an das die Bekanntmachung sendende Cluster streamen, bei dem das zentrale Cluster registriert ist, und/oder das zentrale Cluster kann Daten veröffentlichen, die das die Bekanntmachung sendende Cluster abonniert hat.

[0272] In einigen Ausführungsformen umfasst das Verfahren **470** die Bereitstellung eines Standortdiensts, eines Streamingdiensts (z. B. des Streamingdiensts **115** aus **Fig. 1**) und/oder eines Datenwörterbuchs, die von lokalen Clustern und zentralen Clustern als auch individuellen DDEs **102x** verwendet werden können, um einen Austausch von Analytikdaten zwischen ihnen einzurichten (in **Fig. 4Q** nicht gezeigt).

DIENSTE FÜR INDUSTRIELLE LEISTUNGSÜBERWACHUNG / -ANALYTIK

[0273] Die Umwandlung der Offline-Datendiagramme in Online-Datendiagramme durch Datenstudio, die Ausführung von sowohl Online- als auch Offline-Blöcken und -Modulen und die durch die Diagramme repräsentierte Funktionalität der Analytik wird durch einen Analytikdienst erreicht, der durch ein industrielles Leistungsüberwachung-/Analytiksystem, wie das Datenanalytiksystem (DAS) **100** bereitgestellt wird. Der Analytikdienst wird mit den austauschbar verwendbaren Begriffen „Dienst“, „Dienste“ oder „Analytikdienste“ bezeichnet. Jeder einzelne oder alle Teile der in der vorliegenden Patentschrift beschriebenen Analytikdienste können in Verbindung mit jedem einzelnen oder allen Teilen der **Fig. 1**, **Fig. 2**, **Fig. 3** und/oder **Fig. 4A–Fig. 4Q** und/oder mit einer beliebigen Anzahl von Funktionen und/oder Techniken arbeiten, die in den Abschnitten der vorliegenden Offenbarung beschrieben werden, die mit den **Fig. 1**, **Fig. 2**, **Fig. 3** und/oder **Fig. 4A–Fig. 4Q** korrespondieren.

[0274] Im Allgemeinen ist der Analytikdienst in der Lage, Offline-Diagramme zu komplizieren, auszuführen und/oder umzuwandeln (z. B. in dem Datenstudio und/oder dem Dashboard) und Online-Diagramme zu komplizieren, auszuführen und zu erstellen (im z. B. dem Datenstudio und/oder dem Dashboard). Der Analytikdienst vereinfacht das Durchsuchen großer Datenmengen (Big-Data), ohne dass unbedingt Kenntnisse zu dem (den) bestimmten Datentyp(en) benötigt werden und ohne dass der Nutzer des Analytikdiensts die Programmierung auf niedriger Ebene durchführen muss. Eine beispielhaftes Blockdiagramm-Architektur eines Analytikdiensts **500** wird in **Fig. 5A** im Kontext eines Bereichs des Systems **100** gezeigt. Genauer ist der Analytikdienst **500** einer aus einer Vielzahl von computer-implementierten Prozessen, wobei jeder der computer-implementierten Prozesse mit verschiedenen Softwares (z. B. computer-

ausführbare Anweisungen und Daten, die auf einem oder mehr materiellen, nicht-flüchtigen Computermedium gespeichert sind) assoziiert ist. Wie nachstehend noch detailliert erläutert werden wird, interagiert der Analytikdienst **500** mit dem umfassenderen System **100** zu großen Teilen über die Daten, die von verschiedenen Einheiten des Systems **100** erzeugt und/oder gespeichert werden.

[0275] Wie oben beschrieben ist die Anwendung der DDE-Benutzeroberfläche in dieser Ausführungsform eine webbasierte Anwendung. **Fig. 5A** zeigt eine Ausführungsform des Analysedienstes **500**, der in einer solchen Ausführungsform, d. h. mit einer webbasierten Anwendung der DDE-Benutzeroberfläche, implementiert ist. Im weitesten Sinn beinhaltet ein auf diese Weise implementierter Analysedienst **500** Client-seitige, Server-seitige und Back-End-computerimplementierte Prozesse. Wie beispielsweise in **Fig. 5A** gezeigt, beinhaltet der Analysedienst **500** einen Web-Client-Prozess **502**, einen Web-Server-Prozess **504**, einen Ausführungsdienstprozess **506** und einen oder mehrere Aufgabenprozesse **508**. Die verschiedenen Prozesse **502**, **504**, **506**, **508** können auf einem oder mehreren Prozessoren, auf einer bzw. einem oder mehreren Workstations oder Servern, an einem oder mehreren physikalischen und/oder geographischen Standorten ausgeführt werden. Das heißt, während eine Instanz des Web-Client-Prozesses **502** auf einer Workstation ausgeführt wird, die sich fern von einem Server befindet, auf dem der Web-Server-Prozess **504** ausgeführt wird, kann eine andere Instanz des Web-Client-Prozesses **502** auf demselben Server ausgeführt werden, auf dem der Web-Server-Prozess **504** ausgeführt wird, und faktisch auf demselben Prozessor laufen. In einem anderen Beispiel kann der Ausführungsdienstprozess **506** auf einer Workstation innerhalb der Prozesssteuerungsumgebung ausgeführt werden, während der eine oder die mehreren Aufgabenprozesse **508** auf einem oder mehreren Prozessoren einer verteilten Prozessumgebung (z. B. einer Serverfarm) ausgeführt werden, die sich fern von der Prozesssteuerungsumgebung befinden. Einfach gesagt ist der Analysedienst **500** so konzipiert, dass er flexibel genug ist, dass es egal ist, ob die verschiedenen Prozesse **502**, **504**, **506**, **508** auf demselben Prozessor, derselben Workstation, Prozessorenbank, in einer oder mehreren virtuellen Maschinen und/oder am selben physikalischen oder geographischen Ort ausgeführt werden oder nicht. Günstigerweise ermöglicht die Architektur nicht nur Fernzugriff auf Datenanalysen, die vom Analysedienst **500** bereitgestellt werden, und die Möglichkeit zur Verwendung von weit verstreuten Prozessor- und Rechnerressourcen zur Durchführung von Analysen, sondern erlaubt dem Analysedienst **500** auch, Daten über mehrere Anlagen hinweg zu konsumieren und/oder zu analysieren, unabhängig davon, wie verstreut die geographischen Positionen auf sein mögen.

[0276] Der Web-Client-Prozess **502** läuft gemäß Prinzipien, die sich nicht wesentlich von dem unterscheiden, was allgemein verstanden wird, sondern der Beschreibung der oben beschriebenen Anwendung der DDE-Benutzeroberfläche entsprechen. Beispielsweise beinhaltet der Web-Client-Prozess **502** eine Vielzahl von Softwareeinheiten inklusive beispielsweise einer Anzeigeeinheit **510**, die dem Benutzer die Anwendung der DDE-Benutzeroberfläche anzeigt (z. B. den Hintergrund **245** und Benutzersteuerelemente **248a-n** des Datenanalysestudios **240**, das Datenanalyse-Dashboard usw.); eine Ansichtsmodelleinheit **512**, mit der Transfer und Übersetzung von Anwendungsdaten wie Blockstatus in eine Form verwaltet wird, die eine Benutzeroberfläche verwenden kann, und die eine Ansicht erzeugt, die Feedback bietet und/oder auf andere Art Auskunft über den Betrieb eines Online-Datenmoduls liefert; eine Anwendungseinheit **514**, die die Softwareeinheit (d. h. webbasierte Anwendung) darstellt, die auf den Web-Client-Prozess **502** heruntergeladen wird und sich auf der Client-Workstation oder dem Gerät befindet, das der Benutzer zur Interaktion mit der Anwendung der DDE-Benutzeroberfläche verwendet; und eine Datendiensteinheit **516**, die Daten zur und von der Anwendung der DDE-Benutzeroberfläche transportiert.

[0277] Die Datendiensteinheit **516** empfängt beispielsweise Daten, die von verschiedenen Aufgaben zurückgegeben werden, die als Reaktion auf Eingaben und Anfragen durch den Benutzer ausgeführt wurden. Wie oben und in zusätzlichen Details nachfolgend beschrieben, kann die Anwendung der DDE-Benutzeroberfläche verschiedene Analysen anfordern, die mit Daten aus der Prozesssteuerungsumgebung ausgeführt (und in einigen Fällen gerade von der Prozesssteuerungsumgebung erzeugt) werden, entweder in einem Untersuchungsmodus (z. B. im Datenanalysestudio), mit dem Beziehungen zwischen verstreuten Daten (oder Daten von bekannten Beziehungen) erkannt werden sollen, oder in einem Vorhersagemodus (z. B. im Dashboard), mit dem eine ständige Analyse in Echtzeit (oder Beinahe-Echtzeit) durchgeführt wird, um zumindest einen Wert in der Prozesssteuerungsumgebung vorherzusagen. In diesem Dokument wird der Ausdruck „Echtzeit“ in der Bedeutung von „ausreichend gleichzeitig, um nützlich für die Anpassung der oder anderweitige Beeinflussung der aktuellen Aktion der Prozessanlage zu bleiben“ verwendet. Wenn beispielsweise ein oder mehrere Blöcke auf dem Hintergrund **245** platziert werden und einer oder mehrere dieser Blöcke ausgeführt werden, können die Ergebnisse dieser Ausführung über die Datendiensteinheit **516** (z. B. mithilfe einer WebSocket-Protokoll-Verbindung aus dem Web-Server-Prozess **504**) an die Anwendungseinheit **514** zurückgegeben und dem Benutzer über die Anzeigeeinheit **510** angezeigt werden.

[0278] Gleichzeitig kann die Datendiensteinheit **516** mit dem Web-Server-Prozess **504** über andere Protokolle (z. B. HTTP) kommunizieren, um verschiedene Arten von Informationen anzufordern und abzurufen, die für die Ausführung der Anwendungseinheit **514** erforderlich sind. Beispielsweise kann die Anwendungseinheit **514**, wenn sie erstmals ausgeführt wird, über die Datendiensteinheit **516** mithilfe des HTTP-Protokolls die neuesten Blockdefinitionen **255** aus der Blockdefinitionsbibliothek **252** anfordern und abrufen. Alternativ kann die Anwendungseinheit **514** ursprünglich nur die Kategorien und Listen der verfügbaren Blockdefinitionen **255** abrufen und die Anwendungseinheit **514** kann, wenn vom Benutzer ausgewählt, über die Datendiensteinheit **516** mithilfe des HTTP-Protokolls die spezifischen vom Benutzer ausgewählten Blockdefinitionen **255** anfordern und abrufen.

[0279] Wenn wir uns jetzt dem Web-Server-Prozess **504** zuwenden, kommunizieren ein oder mehrere Serverkommunikationseinheiten **518** mit dem Web-Client-Prozess **502**. Wie oben beschrieben, können für die Kommunikation zwischen Serverkommunikationseinheit/-einheiten Protokolle wie HTTP, HTTPS und WebSocket-Protokolle implementiert werden, obwohl die beschriebenen spezifischen Protokolle nicht als Einschränkung verstanden werden sollen, da auch jedes andere für die Anwendung geeignete Protokoll implementiert werden kann. Zusätzlich zur Kommunikation mit dem Web-Client-Prozess **502** kann die Serverkommunikationseinheit **518** Daten von einem oder mehreren Prozessen erhalten, die Anweisungen ausführen (z. B. Analyse, Laden, Filtern von Daten usw.), die von einem oder mehreren Blöcken (d. h. schematische Programmierelemente) angefordert werden, die auf dem Hintergrund **245** platziert sind. Daten können aus den Prozessen über beispielsweise ein WebSocket (nicht dargestellt) in der Serverkommunikationseinheiten **518** empfangen und an den Web-Client-Prozess **502** ebenfalls über das WebSocket-Protokoll kommuniziert werden.

[0280] Eine Datendiensteinheit **520** stellt für den Web-Client-Prozess **502** andere Daten bereit, als jene, die von Prozessen abgerufen werden, in denen die mit den verschiedenen Blöcken verknüpften Anweisungen ausgeführt werden. Beispielsweise kann die Datendiensteinheit **520** die Blockdefinitionen **255** aus der Blockdefinitionsbibliothek **252** abrufen und an die Anwendungseinheit **514** kommunizieren und/oder Statusinformationen über die Blockausführung aus dem Ausführungsdienstprozess **506** an die Anwendungseinheit **514** kommunizieren, damit der Status durch die Anzeigeeinheit **510** angezeigt werden kann.

[0281] Wenden wir uns nun **Fig. 5B** zu, in der die Datendiensteinheit **520** ausführlicher abgebildet ist. Insbesondere beinhaltet die Datendiensteinheit **520** eine

Kopie der (bzw. Zeiger auf die) Blockdefinitionen **255** in der Blockdefinitionsbibliothek **252** und einen Satz Daten-Plug-ins **523**. Jedes der Daten-Plug-ins **523** besteht aus einem Satz von computerausführbaren Anweisungen, die einer Schnittstelle entsprechen, die von der Anwendungseinheit **514** aufgerufen werden kann, um eine spezielle Aufgabe auszuführen. Jedes der Plug-ins **523** ist von der Anwendungseinheit **514** unabhängig und wird von der Anwendungseinheit **514** geladen, wenn die Anwendungseinheit **514** auf Metadaten stößt, in denen ein zu ladendes Plug-in **523** angegeben ist. Die Plug-ins **523** können Logik beinhalten, die spezifisch für die Anwendung und/oder die Betriebsumgebung ist, in der der Analysedienst **500** betrieben wird. Das heißt, dass sich der Satz von Plug-ins **523** für eine Prozesssteuerungs-umgebung, in der der Analysedienst **500** implementiert wird, von einem Satz von Plug-ins für einen Finanzdienst unterscheiden kann, in dem der Analysedienst **500** implementiert wird (dabei wird berücksichtigt, dass der Analysedienst **500** die Untersuchung von großen Datensätzen – Big Data – ermöglicht, ohne dass dazu notwendigerweise Kenntnisse der speziellen Datentyp(en) erforderlich sind, und ohne dass der Benutzer des Analysediensts **500** eine Programmierung auf niedriger Stufe vornehmen muss, damit die Analyse durchgeführt wird) und sich praktisch von einem Satz von Plug-ins für eine andere Prozesssteuerungsumgebung unterscheiden kann, in der der Analysedienst **500** implementiert wird.

[0282] Die Trennung der Blockdefinitionen **255** von der Anwendungseinheit **514** erlaubt der Anwendungseinheit **514** unabhängig von den Anforderungen des Kunden andere Blockdefinitionen zu verwenden und bereitzustellen. Blockdefinitionen können erstellt, hinzugefügt und/oder entfernt werden, ohne dass dabei die zugrundeliegende Ausführung der Anwendungseinheit **514** beeinträchtigt wird, da die für die Anwendung und/oder Betriebsumgebung spezifische Logik nicht fest in der Anwendungseinheit **514** programmiert ist. Durch eine von der Anwendungseinheit **514** separate Bereitstellung der Plug-ins **523** und der Blockdefinitionen **255** kann die Anwendungseinheit **514** unabhängig von den für eine bestimmte Anwendung spezifischen Blockdefinitionen **255** und Plug-ins **523** mit einer ähnlichen Funktionsweise ausgeführt werden.

[0283] Mit erneutem Bezug auf **Fig. 5A**, wie oben beschrieben, wird die Anwendungseinheit **514** so betrieben, dass die Erstellung von Offline-Diagrammen wie etwa dem Offline-Diagramm **340** aus **Fig. 4L** auf dem Hintergrund **245** möglich ist. Um das Offline-Diagramm zu erstellen, wählt der Benutzer einen oder mehrere Blöcke zur Platzierung auf dem Hintergrund **245** aus (z. B. durch Ziehen und Ablegen) und „verdrahtet“ die Blöcke, damit die Daten zwischen den Blöcken kommuniziert werden. Jeder Block ist eine Instanz eines Vorlagenblocks, der in der Block-

definitionsbibliothek **252** als eine der Blockdefinitionen **255** gespeichert ist. Wenn der Benutzer das Benutzersteuerelement **248m** aktiviert, veranlasst die Anwendungseinheit **514** die Anzeigeeinheit **510**, die verfügbaren Kategorien der Blockdefinitionen **255** und/oder verfügbaren Blockdefinitionen **255** anzuzeigen, aus denen der Benutzer einen Block auswählen und auf dem Hintergrund **245** platzieren kann. In einer Ausführungsform wird die Blockdefinitionsbibliothek **252** in einem Speichergerät gespeichert, das von einem oder beiden der Ausführungsdienstprozesse **506** oder dem Web-Server-Prozess **504** geteilt wird, und Metadaten der Blockdefinitionsbibliothek **252** (z. B. Namen, Kategorien der Blockdefinitionen usw.) werden an die Anwendungseinheit **514** übertragen, wobei Blockdefinitionen nur an die Anwendungseinheit **514** übertragen werden, wenn ein Block ausgewählt oder auf dem Hintergrund **245** platziert wird. Alternativ dazu wird die Blockdefinitionsbibliothek **252** in ihrer Gesamtheit an die Anwendungseinheit **514** übertragen, wenn die Anwendungseinheit **514** im Web-Client instanziert wird, und verbleibt während der gesamten Ausführung des Web-Client-Prozesses **502** auf dem Web-Client. Eine Masterkopie der Blockdefinitionsbibliothek **252** kann in einer Datenbank (nicht dargestellt) innerhalb der Prozesssteuerungsumgebung, in einem über das Netzwerk erreichbaren Cloud-Speichergerät, in einer Workstation usw. existieren.

[0284] Die Blockdefinitionen **255** können gemäß der Funktionalität jedes Blocks kategorisiert werden. Während **Fig. 5A** die Blockdefinitionen **255** zeigt, die auf mehrere abgebildete Kategorien **521** aufgeteilt sind, soll **Fig. 5A** nicht nahelegen, dass die Blockdefinitionen **255**, die in einer bestimmten Kategorie **521** kategorisiert sind, notwendigerweise gemeinsam gespeichert werden oder notwendigerweise getrennt von anderen Blockdefinitionen **255** gespeichert werden. Während das in einigen Ausführungsformen der Fall sein kann, werden die Blockdefinitionen **255** in anderen Ausführungsformen nicht in einer bestimmten Reihenfolge gespeichert, sondern können stattdessen jeweils durch Metadaten getaggt sein, die auf die Kategorie hinweisen, zu der die Blockdefinition gehört. Die Metadaten für eine bestimmte Blockdefinition **255** können, mittels Beispiel und nicht durch Beschränkung, auch auf die Anzahl der Eingänge und Ausgänge, die für die Ausführung des Blocks erforderlichen Eigenschaften, den Typ des Steuerelements für die Benutzeroberfläche jeder Eigenschaft, wann (d. h. unter welchen Bedingungen) das Steuerelement für jede Eigenschaft aktiviert bzw. deaktiviert wird, Plug-in-Informationen in Bezug auf die Eigenschaften, Standardwerte für die Eigenschaften usw. hinweisen.

[0285] In vielen, wenn auch nicht allen Ausführungsformen beinhaltene Blöcke Eigenschaften, die angegeben werden können bzw. müssen, bevor der Block

kompliert und ausgeführt werden kann. Da die Blockdefinitionen **255** nicht in der Anwendungseinheit **514** integriert sind, können die Werte, aus denen ein Benutzer eine Eigenschaft für einen bestimmten Block auswählen kann, nicht vordefiniert und/oder für die Anwendungseinheit **514** bereitgestellt werden, wenn die Anwendungseinheit **514** instanziert wird. **Fig. 5C** ist ein Flussdiagramm, in dem ein Verfahren **551** zur Anzeige eines Eigenschaftendialogs für einen auf dem Hintergrund **245** platzierten Block abgebildet wird. Wenn der Block auf dem Hintergrund **245** platziert wird, ruft die Anwendungseinheit **514** die zugehörige Blockdefinition **255** aus der Datendiensteinheit **520** oder, in Ausführungsformen, aus der Datenbank **529** (Block **553**) ab. Danach kann die Anwendungseinheit **514** einen Befehl zur Anzeige des Eigenschaftendialogs für den Block erhalten, der auf dem Hintergrund **245** platziert wurde (Block **555**), beispielsweise, wenn der Benutzer auf den Block doppelklickt.

[0286] Wenn die Anwendungseinheit **514** den Befehl zur Anzeige des Eigenschaftendialogs für den Block (Block **555**) erhält, parst die Anwendungseinheit **514** die Blockdefinition Eigenschaft für Eigenschaft. Für jede Eigenschaft ruft die Anwendungseinheit **514** die Metadaten für die Eigenschaft ab (Block **557**), indem sie die Metadaten für die Eigenschaft parst und nach Plug-in-Informationen sucht (Block **559**). Wenn die Anwendungseinheit **514** Plug-in-Eigenschaften für die Eigenschaft findet (Block **561**), erkennt und lädt die Anwendungseinheit **514** das angegebene Plugin (z. B. aus den Plug-ins **523** in der Datendiensteinheit **520**) (Block **563**). Die Anwendungseinheit **514** aktiviert das durch die Metadaten angegebene Plug-in für die Eigenschaft, um die erforderlichen Eigenschaftswerte zu erhalten (Block **565**). Wenn die Anwendungseinheit **514** andererseits keine Plug-in-Informationen für die Eigenschaft findet (Block **561**), dann findet die Anwendungseinheit **514** in den Metadaten für die Eigenschaft einen Standardwert bzw. Standardwerte für die Eigenschaften (Block **567**).

[0287] In jedem Fall findet die Anwendungseinheit **514** nach der Aktivierung des durch die Metadaten für die Eigenschaft angegebenen Plug-ins zum Abrufen der Eigenschaftswerte (Block **565**) bzw. Finden des Standardwerts bzw. der Standardwerte für die Eigenschaft (Block **567**) den Typ des Steuerelements für die Benutzeroberfläche (z. B. TextBox, Dropdown-Auswahl, Optionsschaltfläche, Kontrollkästchen usw.) in den Metadaten für die Eigenschaft (Block **569**) und fügt den Wert bzw. die Werte und das Steuerelement zu einem Eigenschaftendialog (Block **571**) hinzu. Wenn in der Blockdefinition (Block **573**) zusätzliche Eigenschaften vorhanden sind, dann ruft die Anwendungseinheit **514** die Metadaten für die nächste Eigenschaft (Block **557**) ab, und das Verfahren wird fortgeführt, bis es keine zusätzlichen Eigenschaften mehr gibt (Block **573**). Wenn kei-

ne zusätzlichen Eigenschaften mehr in der Blockdefinition gefunden werden, zeigt die Anwendungseinheit **514** den Eigenschaftendialog (Block **575**) an, damit der Benutzer die Eigenschaften festlegen kann.

[0288] In einigen Ausführungsformen und/oder für einige Blockdefinitionen **255** können eine oder mehrere erste Eigenschaften des Blocks in einem inaktiven oder deaktivierten Status angezeigt werden, bis eine oder mehrere zweite Eigenschaften des Blocks konfiguriert worden sind. Nach der Konfiguration der einen oder mehreren zweiten Eigenschaften kann die Anwendungseinheit **514** einen oder mehrere Schritte des Verfahrens **551** erneut ausführen und/oder eines oder mehrere der mit der ersten Eigenschaften verknüpften Plug-ins **523** erneut ausführen.

[0289] Nehmen Sie als Beispiel an, dass ein Benutzer einen „Lade“-Block auf dem Hintergrund **245** platziert. Die Anwendungseinheit **514** ruft die Blockdefinition **255** ab. In einer Ausführungsform stellt die Blockdefinition **255** für den Ladeblock der Anwendungseinheit **514** Metadaten über den Block zur Verfügung und zeigt insbesondere an, dass der Ladeblock drei Eigenschaften hat: einen Blocknamen, eine ausgewählte Datenbank (Datenbankname) und einen ausgewählten Datensatz (Datensatzname), der aus der ausgewählten Datenbank abgerufen werden kann. Wenn der Dialog zur Auslösung der Eigenschaftskonfiguration ausgelöst wird (z. B. durch Doppelklicken auf den Ladeblock), muss die Anwendungseinheit **514** festlegen, was angezeigt wird. Die Anwendungseinheit **514** kann feststellen, dass keine Plug-in-Informationen in den Metadaten für die Eigenschaft des Blocknamens vorliegen und ruft stattdessen einen Standardwert ab, beispielsweise „Laden<##>“. Die Anwendungseinheit **514** kann ferner feststellen, dass der Typ des Steuerelements für den Blocknamen, der in den Metadaten für die Eigenschaft des Blocknamens angegeben ist, eine Textbox angibt. Die Anwendungseinheit interpretiert den Standardwert (z. B. „Laden01“) und fügt sie in den Dialog als Textbox ein.

[0290] Die Anwendungseinheit **514** findet Metadaten für die Eigenschaft Datenbankname, indem sie die Blockdefinition parst. Durch Überprüfen der Metadaten für die Eigenschaft Datenbankname findet die Anwendungseinheit **514** Plug-in-Informationen für die Eigenschaft und lokaliert, ladet und aktiviert entsprechend das angegebene Plug-in. Das angegebene Plug-in, das für die spezifische Anwendung und/oder Betriebsumgebung geschrieben wurde (z. B. für eine bestimmte Prozesssteuerungsumgebung), ist so programmiert, dass es die Liste der Datenbanknamen (und zugehörigen Speicherorte) auffindet, von denen Datensätze geladen werden können. Nach dem Abrufen der Liste der Datenbanknamen parst die Anwendungseinheit **514** die Metadaten für die Eigenschaft Datenbankname und bestimmt, dass sie in

einem Auswahl-Steuerelement Dropdown-Liste angezeigt werden sollen. Die Anwendungseinheit **514** fügt das Auswahl-Steuerelement Dropdown-Liste mit der Liste der vom Plug-in zurückgegebenen Datenbanknamen zum Eigenschaftendialog hinzu.

[0291] Die Anwendungseinheit **514** findet Metadaten für die Eigenschaft Datensatzname, indem sie die Blockdefinition parst. Die Metadaten für die Eigenschaft Datensatzname benennen ein Plugin und geben an, dass die Eigenschaft Datensatzname ein Auswahl-Steuerelement Dropdown-Liste ist, weist aber auch darauf hin, dass das Steuerelement deaktiviert sein muss, solange der Datenbankname nicht vom Benutzer ausgewählt ist. Da die Anwendungseinheit **514** keine weiteren Blockdefinitionen für den Ladeblock findet, wird der Dialog angezeigt. Sobald der Benutzer einen der Datenbanknamen aus dem Auswahl-Steuerelement Dropdown-Liste auswählt, weiß die Anwendungseinheit **514** (aus der Datenblockdefinition), dass das in den Metadaten für die Eigenschaft Datensatzname angegebene Plug-in aktiviert werden muss, und übergibt den ausgewählten Datenbanknamen als Argument an das aktivierte Plug-in. Das Plug-in gibt die Namen der Datensätze zurück, die im ausgewählten Datenbanknamen verfügbar sind, und befüllt das Auswahl-Steuerelement Dropdown-Liste für die Eigenschaft Datensatzname mit den Namen der verfügbaren Datensätzen, indem der Dialog erneut angezeigt wird und das Steuerelement diesmal statt deaktiviert aktiviert ist. Der Benutzer kann dann den Datensatznamen auswählen, um die Konfiguration des Datenblocks abzuschließen.

[0292] Natürlich kann ein Block, nachdem er einmal auf dem Hintergrund **245** platziert wurde und nachdem der Block so konfiguriert wurde, dass er die jeweils erforderlichen Eigenschaften anzeigt, ausgeführt werden (d. h. die in der Blockdefinition angegeben Programmieranweisungen können ausgeführt werden), so dass jeder nachfolgende Block (d. h. jeder Block, der über einen mit dem Ausgang des Blocks „verknüpften“ Eingang verfügt) einen gültigen Eingabewert hat, aus dem ein gültiger Ausgabewert erzeugt wird. Mit Bezug auf das Offline-Datendiagramm **340**, das in **Fig. 4L** gezeigt wird, muss beispielsweise der LoadDB4M-Block **342a** evaluiert oder ausgeführt werden, bevor die Blöcke Spalten X **342b** und Spalten Y **342c** konfiguriert werden können, da die Konfiguration der Blöcke Spalten X **342b** und Spalten Y **342c** auf der Benutzeroberfläche aus der Ausgabe des LoadDB4M-Blocks **342a** basiert. Andererseits, und mit zusätzlichen Bezug auf **Fig. 4L**, müssen, obwohl der PLS4M_X_Y-Block **342d** unabhängig von der Evaluierung der Blöcke Spalten X **342b** und Spalten Y **342c** konfiguriert werden kann, die Blöcke Spalten X **342b** und Spalten Y **342c** evaluiert werden, bevor der PLS4M_Y_Y-Block **342d** auf der Grundlage des im Offline-Datendiagramm **340** geladenen Eingangsdatensatzes evaluiert werden kann.

[0293] Die Ausführung und Evaluierung der Blöcke und/oder Module verläuft asynchron. Das erlaubt dem Benutzer, mit der Modifizierung der Blöcke bzw. Module fortfahren, während sie evaluiert werden. Beispielsweise kann der Benutzer die Ausführung eines Blocks starten und gleichzeitig einen oder mehrere weitere Blöcke weiter bearbeiten und/oder konfigurieren. Günstigerweise kann jeder Block, da er separat kompiliert und ausgeführt werden kann, evaluiert und die Ergebnisse können ausgewertet werden, bevor ein nachfolgender Block ausgewählt und/oder konfiguriert und/oder ausgeführt wird. Als Resultat ist weder eine a priori-Kenntnis des erwarteten Ausgangswerts eines bestimmten Blocks erforderlich, noch ist es überhaupt notwendig, zu wissen welches der nächste Block im Datenfluss sein wird, bevor die Ergebnisse eines bestimmten Blocks vorliegen.

[0294] Jede Ausführung eines Offline- oder Online-Diagramms aktiviert die Funktionalität des Ausführungsdienstprozesses **506** und der Aufgabenprozesse **508**. Im Allgemeinen können die Aufgabenprozesse **508**, wie oben beschrieben, auf einem oder mehreren Prozessoren, auf denen Ausführungsdienstprozesse **506** und/oder Web-Server-Prozesse **506** ausgeführt werden (die derselbe Prozessor bzw. dieselben Prozessoren sein können oder auch nicht), auf einem oder mehreren Prozessoren, auf denen der Web-Client-Prozess **502** ausgeführt wird, und/oder auf einem oder mehreren separaten Prozessoren wie etwa Prozessoren auf einer Cloud-Computing-Plattform ausgeführt werden und können auf einer oder mehreren virtuellen Maschinen ausgeführt werden, die auf einem oder mehreren Prozessoren konfiguriert sind. Darüber hinaus kann jeder Aufgabenprozess **508** in einer anderen Zielumgebung ausgeführt werden, beispielsweise durch das Ausführen jedes Aufgabenprozesses **508** in einer dezidierten virtuellen Maschine, die auf der Zielumgebung konfiguriert ist. Jeder der Aufgabenprozesse **508** repräsentiert im Allgemeinen einen Teil des auszuführenden Codes – in einigen Fällen führt ein bestimmter Aufgabenprozess **508** das Zielskript bzw. den Zielcode aus, die mit einem bestimmten Block verbunden sind, während in anderen Fällen die mit einem einzelnen Block verbundenen Anweisungen in kleinere Skripts/Codesegmente unterteilt und in mehreren Aufgabenprozessen **508** ausgeführt werden.

[0295] Insbesondere wenn Anweisungen, die mit einem einzelnen Block verbunden sind, in kleine Skripts/Codesegmente aufgeteilt und von verschiedenen Aufgabenprozessen ausgeführt werden, können die verschiedenen Aufgabenprozesse zusammenarbeiten, um eine Art der Parallelisierung zu erreichen. In diesem Dokument wird „Parallelrechner“ in der Bedeutung einer allgemeinen Praxis des Aufteilens einer Aufgabe in kleinere Einheiten und deren parallele Ausführung verwendet; „Multithreading“

bedeutet die Fähigkeit eines Softwareprogramms zur Ausführung mehrerer Threads (sofern Ressourcen vorhanden sind); und „verteiltes System“ meint die Fähigkeit zur Verteilung der Verarbeitung über mehrere physikalische oder virtuelle Maschinen. Im Prinzip kann ein verteiltes System beliebig erweitert werden. Angesichts dieser Konzepte muss klar sein, dass die Ausführung von Anweisungen auf einer verteilten Plattform nicht dasselbe ist wie die Ausführung der Anweisungen in einem verteilten Modus; ohne explizite Anweisungen, mit denen eine verteilte Verarbeitung unterstützt wird, werden die Anweisungen lokal ausgeführt.

[0296] Die Fähigkeit zur Parallelisierung einer Aufgabe ist eine Eigenschaft, die in der Definition der Aufgabe selbst enthalten ist. Einige Aufgaben können problemlos parallelisiert werden, da die von der jeweiligen Aufgabe durchgeführten Berechnungen von den anderen Aufgabenprozessen unabhängig sind und der gewünschte Ergebnissatz eine einfache Kombination der Ergebnisse der jeweiligen Aufgabenprozesse ist; wir bezeichnen diese Aufgaben als „erstaunlich parallel“.

[0297] Andere Aufgaben sind um einiges schwieriger zu parallelisieren. Zu solchen „linear parallelen“ Aufgaben zählen insbesondere jene, in denen von einem Aufgabenprozess Berechnungen durchgeführt werden, die zwar unabhängig von Berechnungen sind, die von einem anderen Aufgabenprozess durchgeführt werden, das gewünschte Ergebnis aber eine lineare Kombination der Ergebnisse der jeweiligen Aufgabenprozesse darstellt. Beispielsweise kann ein „Durchschnittsberechnungsblock“ durch Anweisungen definiert sein, die den Durchschnitt eines Wertesatzes in einer verteilten Datenbank berechnen, in denen ein einzelner Datensatz verteilt auf verschiedenen physikalischen Geräten gespeichert ist. Nehmen Sie als konkreteres Beispiel an, dass der Durchschnittsberechnungsblock die Durchschnittstemperatur einer Reihe von **300** Bioreaktoren berechnet, in denen die Temperaturdaten jeweils im Speicher einer angeschlossenen Steuereinrichtung gespeichert werden. Die Ausführung des Durchschnittsberechnungsblocks kann parallelisiert werden, indem ein Satz von Temperaturwerten für die **300** Bioreaktoren jeweils einer Vielzahl von Aufgabenprozessen zugewiesen und dann der „Gesamtdurchschnitt“ als gewichteter Durchschnitt der Durchschnittswerte der Vielzahl von einzelnen Aufgabenprozessen berechnet wird.

[0298] Noch schwieriger zu parallelisieren sind Aufgaben, in denen die Daten auf sinnvolle Weise organisiert werden müssen. „Datenparallele Aufgaben“ sind solche, in denen Berechnungen durch die einzelnen Aufgabenprozesse unabhängig von allen anderen Aufgabenprozessen durchgeführt werden, so lange jeder Aufgabenprozess über den „korrekten“

Teil der Daten verfügt. Dazu kann beispielsweise die Berechnung eines unabhängigen Temperaturdurchschnitts für jeden der gleichen oben erwähnten **300** Bioreaktoren zählen (anstatt eines Durchschnitts aller **300**), vorausgesetzt, dass es keine Quereffekte zwischen den Bioreaktoren gibt. Die Aufgabe kann parallelisiert werden, wenn jeder der Aufgabenprozesse über die Temperaturdaten für einen einzelnen der Bioreaktoren verfügt.

[0299] Damit eine solche Aufgabe datenparallel ist, müssen die Daten jedoch in Abschnitte eingeteilt werden, die mit der Aufgabe übereinstimmen. Das kann der Fall sein, wenn jeder der **300** Bioreaktoren im obigen Szenario seine Temperaturdaten in einer einzelnen zugehörigen Datenbank bzw. an einem einzelnen zugehörigen Speicherort speichert. Würden jedoch die Daten für alle **300** Bioreaktoren in einer einzigen Datenbank gespeichert und innerhalb der Datenbank nicht in irgendeiner Art und Weise organisiert, müssten die Daten reorganisiert werden, bevor irgendeine Art der Analyse durchgeführt werden könnte. In dem hier beschriebenen System kann die Datenreorganisation in dem für die Durchführung der Analyse erstellten Diagramm von anderen Blöcken durchgeführt werden (z. B. durch einen Datenfilterblock, Abfrageblöcke usw.).

[0300] Das Erstellen und Verwalten der Aufgabenprozesse **508** sowie die Parallelisierung und Verteilung der Aufgabenprozesse auf die verschiedenen Prozessoren fällt in die Verantwortung des Ausführungsdiestprozesses **506**. Wenn ein Modul oder Block ausgeführt wird, wird die Konfiguration des Moduls oder Blocks unabhängig davon, ob das Modul oder der Block als Teil eines Online-Diagramms oder eines Offline-Diagramms ausgeführt wird, vom Web-Server-Prozess **504** an den Ausführungsdiestprozess **506** gesendet. In Ausführungsformen wird die Konfiguration als JSON-Datei (Java Script Object Notation) gesendet, in der Konfigurationsdatei kann jedoch ein beliebiges Datenformat verwendet werden, das für sprachenunabhängige, asynchrone Browser/Server-Kommunikation geeignet ist. Die Konfigurationsdatei beinhaltet nativen Quellcode der durch das Diagramm **602** repräsentierten Umgebung, in der die Analyse (oder ein Teil davon) durchgeführt werden soll, sowie Werte der Umgebungseigenschaften und Variable, die für die Ausführung erforderlich sind.

[0301] Da in den Blockdefinitionen nicht festgelegt ist, in welcher bestimmten Umgebung die Blöcke ausgeführt werden müssen, und die Zielumgebung bei der Konfiguration des Blocks (oder des Moduls, das den Block beinhaltet) ausgewählt wird, können neue Zielumgebungen zu den im Analysedienst verfügbaren hinzugefügt werden, ohne dass die Anwendung, die Blockdefinitionen oder gar die gespeicherten und ausgeführten Blöcke und/oder Module neu geschrieben werden müssen.

[0302] Auf ähnliche Weise können jederzeit neue Blockdefinitionen **255** zur Blockdefinitionsbibliothek **252** hinzugefügt werden, ohne dass sich das auf die zuvor erstellten Module auswirkt oder sich der Betrieb des Analysedienstes **500** als Ganzes oder der Anwendungseinheit **514** im Besonderen ändert. Mit erneutem Bezug auf **Fig. 5B** wird das Erstellen und das Hinzufügen von neuen Blockdefinitionen in Ausführungsformen durch eine Algorithmusdefinitionsebene **525** erreicht. In der Algorithmusdefinitionsebene **525** kann der Benutzer oder Entwickler einen Definitionsgenerator **527** verwenden, um durch Entwicklung eines Algorithmus eine Definition zu erstellen. Der Definitionsgenerator **527** generiert Definitionen **255'** und speichert die Definitionen **255'** in der Blockdefinitionsbibliothek **252**, beispielsweise in einer Datenbank **529**. Wie in einem späteren Abschnitt der Beschreibung ersichtlich werden wird, kann jede Blockdefinition **255** ein zugehöriges Daten-Plug-in **523** haben, und die Daten-Plug-ins **523** können an einem Speicherort (z. B. einer Datenbank **531**) gespeichert werden, der dem Analysedienst **500** bekannt ist. Die Datendiensteinheit **520** kann die Blockdefinitionen **255** und Daten-Plug-ins **523** abrufen, wenn die Anwendungseinheit **514** initialisiert bzw. instanziert wird, und kann sie der Anwendungseinheit **514** bereitstellen.

[0303] Alternativ dazu kann in einigen Ausführungsformen eine benutzerdefinierte Blockdefinition, anstatt neu erstellt zu werden, keinen zugehörigen Algorithmus haben, sondern eine Eigenschaft beinhalten, in die der Benutzer eigenen Code platzieren kann. Das heißt, dass eine der Blockdefinitionen **255** einem Benutzer erlaubt, seinen eigenen Algorithmus als Eigenschaft in den Block einzugeben.

[0304] In diesem Dokument bezieht sich der Begriff „Parameter“ auf eine Eigenschaft bzw. auf Eigenschaften des Blocks bzw. der Blöcke und/oder des Moduls bzw. der Module, die in der Konfigurationsdatei enthalten sind und mittels der Konfigurationsdatei als Parameter an den Ausführungsdiest weitergegeben werden. Eigenschaften, die in der Konfigurationsdatei als Parameter weitergegeben werden können beispielsweise Dateipfade, Namen von Datensätzen, Spalten zur Auswahl, Konfidenzniveaus und Anzahl der in Ergebnissen von PCA-Analysen erwünschten Komponenten oder jede andere Information beinhalten, die von den Kompilierungseinheiten **526** benötigt werden, damit ausführbarer Code kompiliert werden kann, und mit deren Hilfe die Aufgabenprozesse **508** die kompilierten Anweisungen ausführen können. Das bedeutet, dass nicht alle Eigenschaften eines Blocks oder Moduls für die Ausführung erforderlich sind – einige Eigenschaften eines Blocks oder Moduls können beispielsweise festlegen oder sich auf andere Weise darauf beziehen, wie dieser Block oder dieses Modul im Datenanalysestudio angezeigt wird, und haben keine Auswirkung

darauf, wie die Daten gesammelt, manipuliert oder verarbeitet bzw. analysiert werden.

[0305] In einer Ausführungsform beinhaltet die Konfigurationsdatei die Definition des Blocks und/oder Moduls sowie sämtliche Eigenschaften, die für das Kompilieren und Ausführen des Blocks bzw. Moduls erforderlich sind. In einer alternativen Ausführungsform beinhaltet die Konfigurationsdatei nur eine Identifikation des Blocks und der erforderlichen Konfigurationsparameter, und die Blockdefinition wird aus dem Speicher abgerufen (z. B. aus der Blockdefinitionsbibliothek **252**). Trotzdem können sich die Konfigurationsparameter entsprechend der Blockdefinition ändern. Einige Blöcke können keine Konfigurationsparameter haben, während andere einen, zwei oder viele Konfigurationsparameter haben können. Darüber hinaus können die Konfigurationsparameter verpflichtend oder optional sein. Beispielsweise kann ein Datenladeblock (ein Block, der einen Datensatz lädt) oder ein Abfrageblock (ein Block, der in einem Datensatz nach bestimmten Daten sucht) einen Datenpfad erfordern, in dem der Speicherort der zu ladenden oder abzufragenden Daten angegeben ist. Ein Spaltenauswahlblock kann jedoch als Standardauswahl „alle Spalten“ haben und benötigt daher keine bestimmte, proaktive Konfiguration einer Auswahl eines Subsatzes von Spalten.

[0306] Andere Parameter in einer Konfigurationsdatei können Umgebungsparameter beinhalten. Die Konfigurationsdatei kann beispielsweise und ohne Einschränkung Folgendes beinhalten: einen Ausführungsmodus, der beispielsweise angibt, ob der Workflow bzw. ein Teil davon offline oder online ausgeführt werden soll; eine Zielausführungsumgebung (z. B. Python, PySpark, Native Spark usw.); ob die Ausführung lokal oder verteilt erfolgen soll; die Konfiguration der verteilten Umgebung; eine Angabe des Streaming-Datenquelle, an die der Workflow oder der Teil davon gebunden ist; eine Option zum Erstellen einer neuen Aufgabe oder zum Binden und Fortführen der Ausführung einer bestehenden Aufgabe; eine Option zum Zwischenspeichern von Daten über bestehende Aufgaben; eine Option zum Inkludieren einer Validierungsvariablen zur Evaluierung der Genauigkeit des Ausführungsmodells in die Ausführung eines Online-Diagramms; oder andere Umgebungsvariable.

[0307] Die Konfiguration des Blocks, Moduls und/oder der Umgebungsparameter kann auf eine aus einer Vielzahl von Arten erfolgen. Wie oben beschrieben werden beispielsweise viele Parameter entsprechend den im zugehörigen Block oder Modul festgelegten Eigenschaften an die Konfigurationsdatei übergeben. Während der Erstellung des Diagramms ist es oftmals (jedoch nicht immer) erforderlich, dass ein Benutzer einen bestimmten Block konfiguriert und ausführt, bevor ein nachfolgender Block konfiguriert und/oder ausgeführt werden kann. Das ist der Fall,

wenn der Benutzer einen ersten Block verwendet, um Daten zu laden, und einen zweiten, nachfolgenden Block, um Spalten aus den Daten auszuwählen – der Ladeblock muss konfiguriert werden (z. B. durch einen Pfad zum Ort, von dem die Daten geladen werden sollen, eine Auswahl der zu ladenden Daten usw.), bevor der Ladeblock ausgeführt werden kann, und muss ausgeführt werden, bevor die verfügbaren Spalten in den Daten dem zweiten Block zur Verfügung stehen, damit der Benutzer die Spalten auswählen kann.

[0308] Alternativ oder zusätzlich können ein oder mehrere Parameter zur Konfigurationsdatei hinzugefügt werden, wenn die Ausführung des Blocks oder Moduls angefordert wird. Das kann der Fall sein, wenn ein Benutzer beispielsweise keine der Eigenschaften des Blocks oder Moduls angibt. Es kann auch der Fall sein, wenn eine oder mehrere Eigenschaften so programmiert sind, dass sie erst bei einer Ausführungsanforderung angegeben werden müssen. In Ausführungsformen fordert das Datenanalysestudio beispielsweise einen Benutzer auf, bei der Auswahl eines Blocks oder Moduls zur Ausführung eine Zielumgebung für die Ausführung und/oder einen Zielprozessor oder eine Prozessorbank anzugeben, auf dem bzw. der der Block oder das Modul ausgeführt werden soll. Darüber hinaus können in einigen Ausführungsformen ein oder mehrere Parameter in der Konfigurationsdatei von der Umgebung des Datenanalysestudios selbst übergeben werden. Parameter wie die bevorzugte Ausführungsumgebung, Beschränkungen hinsichtlich physikalischer Orte, an denen die Verarbeitung durchgeführt werden kann bzw. muss (z. B. Verhindern oder Erfordern von Cloud-Ausführung), und andere können in der Anwendung selbst für alle Blöcke/Module, die aus der Anwendung heraus ausgeführt werden, festgelegt werden. Diese Parameter können an die Konfigurationsdatei übergeben werden, wenn die Ausführung eines bestimmten Blocks oder Moduls angefordert wird.

[0309] In jedem Fall, und nach wie vor mit Bezug auf **Fig. 5A**, erhält eine Aufgaben-Listener-Einheit **522** eine Ausführungsanforderung mit der Konfigurationsdatei für den Block oder das Modul bzw. die Blöcke oder Module, die ausgeführt werden sollen, und erstellt eine oder mehrere Aufgabenverwaltungseinheiten **524**, die dem Block und/oder Modul bzw. den Blöcken und/oder Modulen entsprechen, für die eine Ausführung angefordert wurde. Die Aufgabenverwaltungseinheiten **524** erhalten die Konfigurationsdatei und senden die Konfigurationsdatei entsprechend der angegebenen Zielausführungsumgebung an eine oder mehrere der Diagrammkompilierungseinheiten **526**, damit sie in Anweisungen kompiliert wird, die vom Prozessor in der angegebenen Zielausführungsumgebung ausgeführt werden können. Die Diagrammkompilierungseinheiten **526** können so kon-

zipiert sein, dass die jeweilige Diagrammkompilierungseinheit **526** so betrieben werden kann, dass für jede der verschiedenen Zielumgebungen eine Konfigurationsdatei kompiliert werden kann, oder sie können so konzipiert sein, dass die jeweilige Diagrammkompilierungseinheit **526** so betrieben werden kann, dass Konfigurationsdateien kompiliert werden können, die nur eine bestimmte Zielumgebung angeben. In jedem Fall besteht die Ausgabe einer Aufgabenkomplierungsdatei **526** aus computerlesbaren Anweisungen, die von einem Prozessor in der angegebenen Zielumgebung ausgeführt werden können.

[0310] Die Aufgabenverwaltungseinheiten **524**, die die Konfigurationsdatei an die Diagrammkompilierungseinheit **526** gesendet hat, sendet die kompilierte Ausgabe aus der Diagrammkompilierungseinheit **526** an eine Aufgabenausführungseinheit **528**. Die Aufgabenausführungseinheit **528** bestimmt, welche Ressource(n) (z. B. Prozessorressourcen) zur Ausführung kompilierter Ausgabewerte gemäß den angegebenen Konfigurationsparametern und mit Bezug auf die Miteinbeziehung der Zielumgebung für die Aufgabe verfügbar sind, ob die Aufgabe verteilt werden soll oder nicht, ob die Aufgabe lokal oder Remote sein soll und etwaige/oder andere Variable inklusive, jedoch nicht beschränkt auf die Quelle(n) der Eingangsdaten und den Bestimmungsort bzw. die -orte für die Ausgabedaten, und erstellt dann mithilfe der identifizierten Ressourcen einen Aufgabenprozess **508**. Jeder Aufgabenprozess **508** kompiliert Ausgabedaten in einer bestimmten Zielumgebung, um einen bestimmten Zielalgorithmus zu implementieren (d. h. angegeben durch die kompilierten Anweisungen). In Ausführungsformen können einige oder alle Aufgabenprozesse **508** durch Big Data-Analysatoren **170** ausgeführt werden, die in verteilten Daten-Engines **150** integriert sind.

[0311] Somit ist in dieser flexiblen Architektur zumindest jedes der folgenden Szenarien möglich: ein als multiple Aufgabenprozesse **508** in verteilter Weise über mehrere Prozessoren ausgeführter einzelner Block bzw. einzelnes Modul; ein als multiple Aufgabenprozesse **508** seriell auf einem einzelnen Prozessor ausgeführter einzelner Block bzw. einzelnes Modul; ein als einzelner Aufgabenprozess **508** auf einem einzelnen Prozessor ausgeführter Block bzw. Modul; multiple als entsprechende multiple Aufgabenprozesse **508** auf verteilte Weise über mehrere Prozessoren ausgeführter Blöcke usw. Darüber hinaus ist es nicht erforderlich, dass alle Prozesse für unterschiedliche Blöcke in einem Modul oder unterschiedliche Module in einem Projekt in einer einzelnen Zielumgebung betrieben werden, obwohl im Allgemeinen davon ausgegangen wird, dass zu einem einzelnen Block oder Modul gehörende Aufgabenprozesse **508** in derselben Zielumgebung (z. B. Python, PySpark usw.) ausgeführt werden. In einigen Fällen kann beispielsweise eine bestimmte Zielumgebung besser geeignet sein,

um den Zielalgorithmus zu erreichen, der mit einem Block oder Modul verbunden ist, während andere Blöcke oder Module möglicherweise in einer anderen Zielumgebung effizienter ausgeführt werden, und so können die Blöcke so konfiguriert werden, dass in den entsprechenden Eigenschaften unterschiedliche Zielumgebungen angegeben werden, was im Endeffekt dazu führt, dass die mit den unterschiedlichen Blöcken verbundenen Aufgabenprozesse **508** in unterschiedlichen Zielumgebungen ausgeführt werden.

[0312] Ferner wird auch erwartet, dass die mit einem bestimmten Modul oder Projekt verbundenen Blöcke in vielen Fällen in Aufgabenprozessen **508** ausgeführt werden, die alle verteilt bzw. alle lokal sind. Es sollte jedoch in Bezug auf die obige Beschreibung auf der Hand liegen, dass die Effizienz gesteigert werden kann, indem ein oder mehrere Blöcke eines Moduls oder Projekts lokal ausgeführt werden, während andere Blöcke eines Moduls oder Projekts auf einem oder mehreren Remote-Prozessoren ausgeführt werden (z. B. in einer Cloud-Computing-Umgebung). Nehmen wir als Beispiel ein Diagramm, in dem ein erster Block eine Abfrage nach bestimmten Daten aus einer Datenbank (oder einem anderen Datenspeicher) durchführt, die sich entfernt vom Analysedienst **500** befindet, und ein zweiter Block eine Analyse der aufgrund der Abfrage erhaltenen Daten durchführt. Es ist möglicherweise effizienter, einen Aufgabenprozess **508** auf einem Prozessor zu erstellen, der sich lokal zur Datenbank befindet (z. B. in einem integrierten DDE in der Datenbank), um die Datenabfrage lokal zum integrierten DDE durchzuführen, anstatt die Abfrage auf einem Prozessor durchzuführen, der sich entfernt von der Datenbank befindet. Das kann insbesondere dann der Fall sein, wenn sich die Datenbank in einer anderen geographischen Region befindet, etwa wenn der Benutzer Daten für eine Prozesssteuerungsumgebung abfragt, die sich von der unterscheidet, in der sich der Benutzer gerade befindet. Die Abfrageergebnisse können dann entsprechend dem zweiten Block entweder im selben Aufgabenprozess (und Prozessor), in einem anderen Aufgabenprozess im selben Prozessor oder in einem anderen Aufgabenprozess in einem anderen Prozessor (z. B. in einem lokalen Prozessor) analysiert werden.

[0313] Die Aufgabenausführungseinheit **528** protokolliert ferner insbesondere in verteilten Verarbeitungsanwendungen die Aufgabenprozesse **508** und protokolliert die Datenabhängigkeiten. Daher werden nach Abschluss jedes Aufgabenprozesses **508** die Ausgabedaten an die Aufgabenausführungseinheiten **528** zurückgegeben. Die Aufgabenausführer **528** bestimmen, ob einige oder alle Daten an einen anderen Aufgabenprozess **508** weitergeleitet und/oder an den Web-Server-Prozess **504** zur Anzeige in der Anwendungseinheit **514** durch den Web-Client-Prozess **502** zurückgegeben werden. Während der Ausführung der Aufgabenprozesse **508** können

die Aufgabenausführungseinheiten **528** Statusinformationen empfangen, die zur Anzeige in der Anwendungseinheit **514** an den Web-Server-Prozess **504** zurückgegeben werden können. Die Aufgabenprozesse **508** können auf Vorhersagedaten an die Aufgabenausführungseinheiten **528** zurückgeben, insbesondere wenn ein Aufgabenprozess **508** ein Online-Diagramm (oder einen Teil eines Online-Diagramms) ausführt, das solange ausgeführt wird, bis es explizit gestoppt wird. Die Vorhersagedaten können an die Aufgabenausführungseinheiten **528** zurückgegeben werden, während die Vorhersagedaten generiert werden, und können zur Anzeige in der Anwendungseinheit **514** an den Web-Server-Prozess **504** weitergeleitet werden; dabei ist zu beachten, dass die Vorhersagedaten gleichzeitig als Funktion eines der Blöcke im Online-Diagramm in einen Speicherort in der Prozesssteuerungsumgebung (z. B. einen DDE-Speicherort, eine Steuerung, die eine Steuerfunktion ausführt usw.) zurück geschrieben werden können. In Ausführungsformen werden Status, Zustand und/oder Vorhersagedaten zwischen den Aufgabenprozessen **508** und den Aufgabenausführungseinheiten **528** mithilfe von Rückkanal-Streaming über Kommunikationskanäle übergeben, die während der Aufgabenerstellung eingerichtet wurden, damit die Aufgabenprozesse **508** miteinander und/oder mit den Aufgabenausführungseinheiten **528** kommunizieren können.

[0314] Anhand der nachfolgenden Beispiele wird klar, dass der Datenfluss durch den Analysedienst **500** in gewisser Weise von zumindest den konfigurierten präzisen Blöcken und/oder Modulen, den für jeden Block bzw. jedes Modul konfigurierten Eigenschaften sowie davon abhängt, ob das auszuführende Diagramm ein Offline-Diagramm oder ein Online-Diagramm ist.

Beispiel 1 – Erstellen eines Offline-Diagramms

[0315] Fig. 5D ist ein Diagramm, in dem ein Beispielhintergrund **600** abgebildet ist, auf dem ein Offline-Diagramm **602** konfiguriert ist, das die Blöcke **604a–604g** und die Drähte **606a–606h** beinhaltet. Mit Bezug zurück auf Fig. 5A würde der Hintergrund **600** durch den Web-Client-Prozess **502** und insbesondere durch die Anzeigeeinheit **510** in Zusammenarbeit mit der Anwendungseinheit **514** angezeigt werden. Die Anwendungseinheit **514** würde die Daten (inklusive Blockdefinitionsbibliothek **252**) über die Datendiensteinheit **516** erhalten, die über die Serverkommunikationseinheiten **518** in Kommunikation mit dem Web-Server-Prozess **540** stehen würde.

[0316] Allgemein gibt es einige Methoden zur Konstruktion des Offline-Diagramms **602** gemäß Fig. 5D. Zuerst könnte ein Benutzer den Blockdefinitionsumschalter **248m** umschalten, um die Blockdefinitionsbibliothek **252** zu öffnen (siehe Fig. 4D). Aus der Block-

definitionsbibliothek **252** könnte der Benutzer die Blöcke **604a–604g** jeweils einzeln auswählen und die Blöcke **604a–604g** auf dem Hintergrund **600** platzieren. Als nächstes könnte der Benutzer die Drähte **606a–606h** anbringen, um die verschiedenen Blöcke miteinander zu verbinden: Draht **606a** zwischen einem Ausgang **608a** von Block **604a** und einen Eingang **610b** von Block **604b** anbringen; Draht **606b** zwischen einem Ausgang **608b** von Block **604b** und einen Eingang **610b** von Block **604c** anbringen; Draht **606c** zwischen einem Ausgang **608b** von Block **604b** und einen Eingang **610d** von Block **604d** anbringen; Draht **606d** zwischen einem Ausgang **608b** von Block **604b** und einen Eingang **610e** von Block **604e** anbringen; Draht **606e** zwischen einem Ausgang **608c** von Block **604c** und einen Eingang **610f2** von Block **604f** anbringen; Draht **606f** zwischen einem Ausgang **608d** von Block **604d** und einen Eingang **610g** von Block **604g** anbringen; Draht **606g** zwischen einem Ausgang **608d** von Block **604d** und einen Eingang **610f1** von Block **604f** anbringen und Draht **606h** zwischen einem Ausgang **608f1** von Block **604f** und einen Eingang **610g** von Block **604g** anbringen. Nachdem die Blöcke **604a–604g** und die Drähte **606a–606h** auf diese Weise platziert wurden, würde das Offline-Diagramm **602** unkonfiguriert bleiben.

[0317] Alternativ könnte der Benutzer die Blöcke **604a–604g** jeweils einzeln auswählen und platzieren und die Eigenschaften jedes Blocks **604a–604g** beim Platzieren des jeweiligen Blocks auf dem Hintergrund **600** konfigurieren, optional (und in den meisten Fällen) wird jeder Block **604a–604g** nach der Konfiguration der Eigenschaften des Blocks ausgeführt.

[0318] Der Benutzer könnte jeden Block **604a–604G** jeweils einzeln konfigurieren (unabhängig davon, ob die Blöcke **604a–604g** vor der Konfiguration der Eigenschaften platziert und verdrahtet werden, oder ob sie jeweils einzeln platziert und die Eigenschaften konfiguriert werden, bevor der nächste Block platziert wird), indem er auf den Block klickt und in einem Blockeigenschaftsfenster **612** die dem Block zugewiesenen Eigenschaften angibt. In der Fig. 5D wird der Block **604a** – ein Abfrageblock – ausgewählt und das Blockeigenschaftsfenster **612** für den Abfrageblock **604a** wird angezeigt. Im Beispielhintergrund **600**, der in Fig. 5D abgebildet ist, beinhaltet das Blockeigenschaftsfenster **612** für den Abfrageblock **604a** einen Abfrageeigenschaftseingabebereich **614**, in den der Benutzer eine Abfrage eingeben kann. Während eine detaillierte Beschreibung der Abfragesprache in einem späteren Abschnitt der Spezifikation erfolgt, ist es für den aktuellen Zweck ausreichend, wenn angenommen wird, dass die in den Abfrageeigenschaftseingabebereich **614** eingegebene Abfrage bei Ausführung des Abfrageblocks **604a** einen Speicherort der abzufragenden Daten angeben kann. Die Blockeigenschaften werden als Parameter in einer Konfigurationsdatei gespeichert, die dem Block zu-

geordnet ist. Die Konfigurationsdatei kann sich auf dem Web-Client, der den Web-Client-Prozess **502** ausführt, oder auf dem Web-Server, der den Web-Server-Prozess **504** ausführt, befinden.

[0319] Nachdem er die Eigenschaften des Abfrageblocks **604a** konfiguriert hat, würde der Benutzer im Normalfall den Block **604a** ausführen, bevor er den nächsten, nachfolgenden Block platziert und/oder mit der Konfiguration der Eigenschaften des nächsten, nachfolgenden Blocks beginnt (in diesem Fall Block **604b**). Um den Block **604a** auszuführen, würde der Benutzer die Evaluierungsbenuersteuerung **248i** aktivieren. Dadurch würde die dem Block **604a** zugeordnete Konfigurationsdatei über die Datensteinheit **520** des Web-Server-Prozesses **504** an den Ausführungsserverprozess **506** übertragen werden. Die Konfigurationsdatei würde beispielsweise als JSON-Datei durch die Ausführung des Serverprozesses **506** empfangen werden und die Aufgaben-Listener-Einheit **522** würde eine Aufgabenverwaltungseinheit **524** erstellen, mit der die Ausführungsanfrage behandelt würde. Die Aufgabenverwaltungseinheit **524** würde die entsprechende Diagrammkompilierungseinheit **526** aktivieren, um die Konfigurationsdatei in ausführbare Anweisungen (d. h. kompilierten Abfragecode) zu kompilieren, die für die in den Parametern in der Konfigurationsdatei angegebene Zielumgebung geeignet sind. Die Diagrammkompilierungseinheit **526** würde eine Ausgabedatei zurückgeben, die kompilierten Code umfasst und könnte ferner Metadaten zurückgeben, die die Zielumgebung festlegen, für die sich der kompilierte Code eignet. (Alternativ kann die Aufgabenverwaltungseinheit **524** während des Kompilierungsprozesses Kenntnis über die Zielumgebung erlangen.) Die Aufgabenverwaltungseinheit **524** leitet die Ausgabedatei an eine Aufgabenausführungseinheit **528** weiter, die gemäß den Parametern der Konfigurationsdatei für die Zielumgebung bestimmt, ob der Code lokal oder in der Ferne ausgeführt werden soll, ob der Code in einem verteilten System ausgeführt werden soll usw., und Prozessor und Speicherressourcen auswählt und sichert, an denen die in der Ausgabedatei enthaltenen Anweisungen ausgeführt werden sollen. Die Aufgabenausführungseinheit **528** erstellt dann den Aufgabenprozess bzw. die Aufgabenprozesse **508**, mit denen die Ausgabedatei ausgeführt wird.

[0320] Wenn die Ausführung der Ausgabedatei abgeschlossen ist (im Fall von Offline-Blocks oder -Modulen), oder sobald Ergebnisse vorliegen (für Online-Module), werden die Ergebnisse an die Aufgabenausführungseinheit **528** zurückgegeben. Da in diesem Fall keine weitere Ausführung schwebend war (da das Diagramm ein Offline-Diagramm ist und der soeben ausgeführte Block – der Abfrageblock **604a** – der einzige konfigurierte Block ist), werden die Ergebnisse über die Serverkommunikationseinheit **508**

an den Web-Server-Prozess **504** und schlussendlich an die Anwendungseinheit **514** auf dem Web-Client-Prozess **502** zurückgegeben. Die Ausgabedaten können alle von der Abfrage zurückgegebenen Daten beinhalten, können jedoch auch verschiedenen statistischen Daten (Metadaten) über die Abfragedaten enthalten, etwa (ohne Einschränkung) Mittelwerte, Standardabweichungen und Medianwerte für jede Datenspalte, die von der Abfrage zurückgegeben wird. Die Metadaten und die Ergebnisdaten werden, in Ausführungsformen, getrennt voneinander gespeichert. Dies erlaubt dem Benutzer günstigerweise, die Informationen über die Ergebnisdaten zu überprüfen, ohne dazu notwendigerweise die gesamten Ergebnisdaten laden zu müssen, was zeit- und/oder ressourcenintensiv (d. h. Computerspeicher) sein könnte. Der Benutzer kann sich die zurückgegebenen Daten und/oder Metadaten über die zurückgegebenen Daten ansehen, indem er auf das Anzeigeblockergebnisselement **212a** (siehe **Fig. 4A**) auf dem Block klickt. Die Standardansicht kann Metadaten für die Ergebnisse der jeweiligen in den Ergebnissen inkludierten Variablen beinhalten und kann dem Benutzer erlauben, einen oder mehrere ausgewählte Bereiche der Ergebnisdaten zum Laden und, falls gewünscht, Anzeigen (z. B. in Kurven, Diagrammen, Tabellen usw.) auszuwählen, ohne dazu alle Daten laden zu müssen. Darüber hinaus ermöglicht die Rückgabe der Ergebnisse an die Anwendungseinheit **514** die Konfiguration der Eigenschaften des nächsten, nachfolgenden Blocks (in diesem Fall Block **604b**), für den die Daten und/oder Metadaten mithilfe des Drahtes **606a** verfügbar gemacht werden.

[0321] Es sollte beachtet werden, dass die Ausführung eines Blocks (z. B. von Block **604a**) nicht in allen Fällen erforderlich ist, um den nächsten, nachfolgenden Block (z. B. Block **604b**) zu konfigurieren. Ob ein Block konfiguriert und/oder ausgeführt werden muss, bevor ein nachfolgender Block konfiguriert und/oder ausgeführt werden kann, hängt gemäß dem Verständnis der Gesamtheit der vorliegenden Offenbarung von den Datenabhängigkeiten ab.

[0322] Es sollte auch beachtet werden, dass die Aufgabenprozesse **508** während der Ausführung mit den verschiedenen verteilten Daten-Engines **150** kommunizieren können, um im Big Data-Speicher **155** gespeicherte Daten zu empfangen oder darauf zuzugreifen. In Fällen, in denen ein Aufgabenprozess **508** in einem integrierten Big Data-Analysator **170** ausgeführt wird, kann der Big Data-Analysator **170** kommunikativ – direkt oder über einen oder mehrere Datenforderungsdienste **165** – an den Big Data-Speicher **155** gekoppelt sein. Alternativ kann der Aufgabenprozess **508** in Fällen, in denen ein Aufgabenprozess **508** in einem Prozessor ausgeführt wird, der nicht Teil eines integrierten Big Data-Analysators **170** ist, über ein Netzwerk kommunikativ an die DDEs

gekoppelt sein und Daten über die Big Data-Anforderungsdienste **165** anfordern.

[0323] Die Konfiguration der Eigenschaften und die Ausführung der jeweiligen Blöcke **604b–604g** im Offline-Diagramm **602** wird auf gleiche Weise fortgesetzt und muss daher nicht im Detail besprochen werden.

A. Verwenden von Offline-Diagrammen zum Erkunden und Modellieren

[0324] Ein Benutzer der Anwendung der DDE-Benutzeroberfläche, der mit einem Offline-Diagramm arbeitet, versucht im Allgemeinen ein oder beide der folgenden Dinge zu tun: (1) verschiedene Datensätze erkunden, um nach Beziehungen Ausschau zu halten und diese zwischen verschiedenen Datensätzen und/oder Ereignissen zu entdecken; und (2) Modelle dieser Beziehungen zu erstellen, um eine beständig Vorhersagemöglichkeit in Echtzeit zu implementieren. Für das Erreichen des letzteren ist in der Regel einiger Zeitaufwand für das erstere erforderlich. Das heißt, dass ein Benutzer in vielen (wenn auch nicht in allen) Fällen Datensätze erkunden wird, um mithilfe verschiedener Analysewerkzeuge (Hauptkomponentenanalyse, Fourier-Analyse, Regressionsanalyse usw.) Beziehungen zwischen den Daten zu finden, und wenn eine Beziehung gefunden wird, die einen Vorhersagewert zu haben scheint, der Benutzer diese Analyse als Modell implementieren und in Echtzeit für eine fortlaufend Analyse des Prozesses einsetzen wird.

[0325] In Ausführungsformen kann der Analyse-dienst **500** Werkzeuge beinhalten, die eine automatische Analyse von Daten durchführen, sei es in Echtzeit oder mit historisierten Daten oder beides, um Anomalien, abnormale Bedingungen in der Prozessumgebung, Beziehungen zwischen Daten und/oder Funktionen von bestimmten Datensätzen zu finden. In einer Ausführungsform beinhaltet die Blockdefinitionsbibliothek **252** einen Block **255**, der Daten untersucht und Anomalien wie Spitzen oder Dellen in Daten, die im Kontext des untersuchten Datensatzes atypisch sind, entdeckt und/oder kennzeichnet. Ein solcher Block zur Identifizierung von Anomalien in einem Offline-Diagramm könnte als Eingang historisierte Daten erhalten, um in diesen historisierten Daten nach Funktionen zu suchen, die für die Erforschung einer weiteren Verwendung interessant sein könnten, beispielsweise einige der oben beschriebenen Technologien inklusive PLS, PCA und andere Arten der Analyse. Alternativ oder zusätzlich könnte ein Block zur Identifizierung von Anomalien in einem Online-Diagramm sowohl historisierte als auch Echtzeitdaten erhalten und die historisierten Daten mit den Echtzeitdaten vergleichen, um nach Anomalien in den Echtzeitdaten in Bezug auf die historisierten Daten zu suchen. In beiden Fällen könnte die Ausgabe des Blocks zur Identifizierung von Anomalien eine

oder mehrere Zeiten, in denen das abnormale Ereignis aufgetreten ist, die Quelle bzw. Quellen der abnormalen Daten (d. h. welche Messgröße und/oder welches Gerät ist mit der Anomalie verbunden) usw., beinhalten.

[0326] In einer Ausführungsform beinhaltet die Blockdefinitionsbibliothek **252** einen Block **255**, der autonom Beziehungen zwischen verschiedenen Daten erkennt. Ein Beziehungserkennungsblock kann zufällig oder halbzufällig Datensätze auswählen (zeitlich gereiht) und mehrere verschiedene Analysen der Datensätze durchführen, um nach Beziehungen zu suchen, die für einen Benutzer von Interesse sein könnten, insbesondere kausale Beziehungen oder prädiktive Beziehungen. Natürlich wäre es bei rein zufälliger Auswahl von Datensätzen unwahrscheinlich, viele verwendbare Ergebnisse zu erhalten. Die Auswahl der Datensätze könnte jedoch beispielsweise über den physikalischen Ort, die Anlagenhierarchie oder andere Kennzeichen gesteuert werden, die auf eine bestehende Beziehung zwischen Daten hinweisen könnten.

[0327] In einigen Ausführungsformen kann der Beziehungserkennungsblock mit dem Block zur Identifizierung von Anomalien zusammenarbeiten, um Beziehungen zwischen Daten zu finden. Beispielsweise könnte der Block zur Identifizierung von Anomalien die Zeiten ausgeben, zu denen verschiedene Anomalien in verschiedenen Datensätzen aufgetreten sind. In einer Ausführungsform speichert der Block zur Identifizierung von Anomalien die Zeiten der abnormalen Ereignisse, und wenn abnormale Ereignisse in verschiedenen Datensätzen zur gleichen Zeit oder nur kurz hintereinander auftreten, kann eine Ausgabe des Blocks zur Identifizierung von Anomalien aus einer Liste von Datenquellen bestehen, die den Datensätzen entsprechen, in denen abnormale Ereignisse in kurzem zeitlichem Abstand voneinander aufgetreten sind, und eine andere Ausgabe des Blocks zur Identifizierung von Anomalien könnte aus einer Liste von Zeiten bestehen, zu denen die Anomalien aufgetreten sind. Der Beziehungserkennungsblock kann die Ausgaben des Blocks zur Identifizierung von Anomalien als Eingabe erhalten und die entsprechenden Datensätze für die in Frage kommenden Zeitspannen laden, möglicherweise gemeinsam mit anderen Datensätzen und/oder größeren Zeitspannen (z. B. Ausweitung auf eine Zeit weiter vor und nach der Zeit, zu der die Anomalie aufgetreten ist), um verschiedene Arten der Analyse auf die Daten anzuwenden und kausale oder zumindest prädiktive Beziehungen zwischen den Daten zu suchen.

[0328] Alternativ können eine oder mehrere Eigenschaften eines Beziehungserkennungsblocks auf einen bestimmten Datensatz hinweisen (z. B. Werte für eine bestimmte Prozessvariable), für den der Benutzer Beziehungen in den Daten finden möch-

te. Der Beziehungserkennungsblock kann autonom nach Beziehungen zwischen anderen Daten – in Bezug auf den physikalischen Ort, die Anlagenhierarchie, lokale Beziehung usw. – und dem angegebenen Datensatz suchen, indem er unterschiedliche Analysen der Daten durchführt, bis eine oder mehrere kausale oder prädiktive Beziehungen gefunden werden, die den Kriterien entsprechen (z. B. 95 % prädiktive Sicherheit, 0,75 % Korrelationswert usw.).

[0329] Natürlich wird jeder automatische Block wie etwa der Beziehungserkennungsblock oder der Block zur Identifizierung von Anomalien als in einem Offline-Diagramm oder einem Online-Diagramm ausführbar betrachtet.

[0330] In jedem Fall würde ein Benutzer auf dem Hintergrund **245** im Normalfall ein Offline-Diagramm wie das Diagramm **602** auf dem Hintergrund **600** erstellen. Das Erstellen des Diagramms würde im Allgemeinen wie oben beschrieben vor sich gehen, wobei ein oder mehrere Blöcke **604** auf dem Hintergrund **600** platziert werden, und die Eigenschaften eines jeden Blocks **604** jeweils einzeln nacheinander konfiguriert werden und jeder Block **604** ausgeführt wird, bevor die Eigenschaften des nächsten konfiguriert werden. Der Benutzer kann sich die Daten an jedem Punkt des Diagramms kritisch ansehen, indem er auf das Benutzersteuerelement einer Datenzusammenfassung für den Block **604** klickt, um verschiedenen Statistiken (z. B. Mittelwerte, Mediane, Standardabweichungen usw.) über die Daten im Block anzuzeigen, oder er kann sich mit dem Ausgang **608** des Blocks **604** (über den Draht **606**) verbinden, um die Daten zu untersuchen (z. B. Block **604e** in Fig. 5D). Angenommen, der Benutzer hat die Blöcke im Offline-Diagramm bis inklusive dem Block, mit dem der Untersuchungsblock verbunden ist, ausgeführt, dann hat er die Möglichkeit, den Untersuchungsblock zum Anzeigen der Daten in verschiedenen Kurven anzuzeigen, Metadaten mit Bezug zu den Daten und ähnliches anzuzeigen. Beispielsweise erlaubt im Offline-Diagramm **602** der Untersuchungsblock **604e** dem Benutzer, die Daten anzuzeigen, nachdem die Daten vom Block **604b** verarbeitet wurden. Auf ähnliche Weise erhält Block **604g** die Daten vom PLS-Block **604f** und vom Spaltenblock **604d**. Im letzteren Fall kann der Untersuchungsblock **604g** dem Benutzer erlauben, die Ausgabe des PLS-Modells zu visualisieren (dazu kann auch ein vorhergesagter Wert einer Variablen gehören) und die Ausgabe mit den tatsächlichen Werten dieser Variable (aus dem Block **604d**) zu vergleichen, die verwendet wurden, um das PLS-Modell zu erstellen. Der Benutzer kann die Möglichkeit haben, durch Ansehen der Kurven der Daten zu bestimmen, ob das PLS-Modell den Prozess exakt abbildet, so dass es im Prozess einen prädiktiven Wert hat.

[0331] In Ausführungsformen ist das Benutzersteuerelement für die Datenzusammenfassung mit einem Block **255** verknüpft und spezifisch für Typ und Funktion des Blocks. Ein Block, der Daten ladet (z. B. ein Abfrageblock oder ein Ladeblock) kann beispielsweise so programmiert werden, dass das Benutzersteuerelement Datenzusammenfassung bei Aktivierung die Anzeige verschiedener statistischer Daten veranlasst, durch die die geladenen oder von der Abfrage zurückgegebenen Daten charakterisiert sind. Die Daten können in Ausführungsformen einen Mittelwert, einen Medianwert, eine Standardabweichung, einen Maximalwert und einen Minimalwert für jede in den Daten inkludierte Datenquelle haben. Im Gegensatz dazu kann ein Block, der eine Analyse von Daten durchführt (z. B. eine PLS- oder PCA-Analyse), andere Daten anzeigen, wenn das Benutzersteuerelement Datenzusammenfassung aktiviert wird. Der Analyseblock kann ein oder mehrere Bestimmtheitsmaße, Koeffizienten für PLS- und/oder PCA-Analyse, Varianzwerte, Beobachtungszählwerte (z. B. wie viele Zeitreihenwerte für einen bestimmten Datenquelle inkludiert waren) und anpassbare Kurven anzeigen, mit denen der Benutzer auswählen kann, welche Daten (d. h. aus welchen Datenquellen) er anzeigen möchte. Der Untersuchungsblock zeigt ferner abhängig von den Datenausgängen, mit denen die Dateneingänge verbunden sind, in Ausführungsformen ein unterschiedliches Verhalten. Das heißt, dass die Formate und Typen der angezeigten und/oder für die Anzeige verfügbaren Daten in einem Untersuchungsblock von den Typen der Blöcke abhängig sein können, in denen die Eingangswerte für den Untersuchungsblock generiert werden.

[0332] Mit Bezug auf Fig. 5D wird nun die Funktion des Offline-Diagramms **602** allgemein beschrieben, das im Wesentlichen eine Diagrammkonfiguration aus vielen Möglichkeiten darstellt. Das Offline-Diagramm **602** beginnt mit dem Abfrageblock **604a**, dessen Zweck es ist, einen bestimmten Datensatz zu finden und für die Analyse zu laden. Der spezifische Datensatz kann, um nur ein Beispiel zu nennen, aus historisierten Prozesssteuerungsdaten bestehen, die mit einer bestimmten Prozessvariable verknüpft sind, die zu drei bestimmten Zeiten eine abnormale Abweichung gezeigt hat. Im Allgemeinen kann der Abfrageblock **604a** jedoch beliebige gespeicherte Daten abfragen, darunter, ohne Einschränkung, beliebige Daten im Big Data-Speicher **155**, beliebige in einer oder mehreren der Datenquellen **168** gespeicherte oder im Cache zwischengespeicherte Daten, in externen Datenquellen wie etwa Wetterdatenquellen, Datenquellen für Lieferketten, Lieferungsnachverfolgung usw. gespeicherte Daten. In Ausführungsformen speichert eine Datenbank oder ein Dokumentenspeicher (nicht abgebildet) Dokumente (z. B. JSON-Dokumente), die mit dem Analysedienst verbundene Online- und/oder Offline-Diagramme beschreiben; in diesem Fall kann der Abfrageblock **604a** so konfiguriert werden, dass

diese Datenbank oder dieser Dokumentenspeicher in Bezug auf die Arten der durchgeführten Berechnungen, die Quelle bzw. Quellen der in den Berechnungen verwendeten Daten, die Qualität der Ergebnisse usw. abgefragt wird. Eine Abfrage des letzteren Typs kann günstigerweise einen Verlauf oder eine Wissensdatenbank der zu erstellenden Diagramme ermöglichen. Der Abfrageblock **604a** kann so konfiguriert werden, dass Werte dieser Prozessvariablen und eine Anzahl anderer Prozessvariablen ausgewählt werden, und kann sogar so konfiguriert werden, dass die Abtastrate und/oder die Werte der spezifischen Prozessvariablen und der anderen Prozessvariablen beispielsweise nur innerhalb einer Zeitspanne ausgewählt werden, die jeweils einer Stunde vor bis eine Stunde nach den drei Ereignissen entspricht. Das heißt, die Abfrage kann Zeitspannen auswählen, die disjunkt sind, und Daten finden, die während dieser Zeitspannen für eine beliebige Anzahl von Variablen erzeugt werden, von denen zumindest eine die ist, die der Benutzer vorherzusagen wünscht.

[0333] Nach der Ausführung können die vom Abfrageblock **604a** abgerufenen Daten vom Füllblock **604b** verwendet werden. Im Beispieldiagramm **602** kann der Füllblock **604b** Daten für eine Variable ausfüllen, die den Zeiten entspricht, zu denen die Variable keinen Wert aufgewiesen hat, eine der anderen Variablen jedoch schon. Das heißt, wenn einer der abgefragten Werte mit einer häufigeren Rate abgetastet wird als ein anderer, kann der Füllblock **604b** Werte für die weniger häufig abgetastete Variable einsetzen, um die Häufigkeit der öfter abgetasteten Variable zu erreichen. In Ausführungsformen kann der Füllblock **604b** den Wert der weniger häufig abgetasteten Variablen extrapolieren. Wenn beispielsweise vier Werte der häufiger abgetasteten Variable zwischen den Werten der seltener abgetasteten Variable auftreten, kann der Block **604b** die Differenz zwischen den beiden aufeinanderfolgenden Werten der seltener abgetasteten Variable berechnen (z. B. 1, 0 und 6,0), durch vier dividieren und die „fehlenden“ vier Werte mit aufeinanderfolgenden höheren oder niedrigeren Werten einsetzen (z. B. 2,0; 3,0; 4,0 und 5,0), so dass für jeden Wert der häufiger abgetasteten Variable ein entsprechender Wert der seltener abgetasteten Variable zur Verfügung steht. (Selbstverständlich wäre das in einer Echtzeitanalyse keine plausible Methode zur Datenbefüllung, da die späteren Werte noch nicht bekannt wären.) In anderen Ausführungsformen kann der Füllblock **604b** die „fehlenden“ Werte einfach mit den aktuellsten Abtastwerten für den seltener abgetasteten Wert befüllen. Im oben verwendeten Beispiel würden die fehlenden Werte jeweils durch den Wert 1,0 ersetzt werden.

[0334] Nachdem die Daten gefunden und die Werte eingefügt wurden, um einen Datensatz ohne fehlende Werte zu erhalten, wird der Ausgang des Füllblocks **604b** drei Blöcken zur Verfügung gestellt: dem Prüf-

block **604e**, einem ersten Spaltenblock **604c** und einem zweiten Spaltenblock **604d**. Der Prüfblock **604e** wird vorstehend beschrieben. Der erste und der zweite Spaltenblock **604c** und **604d** funktionieren jeweils dahingehend, dass sie Variablen (Spalten) aus den Daten ziehen. Die Daten werden in Tabellen gespeichert, bei denen jede Spalte für eine Variable steht; und die Zeilen in jeder Spalte stehen für Werte der jeweiligen Variable zu unterschiedlichen Zeitpunkten (d.h. Zeitreihendaten). Der erste Spaltenblock **604c** kann beispielsweise alle Spalten auswählen (d.h. alle Variablen), in denen die Daten für die Variablen enthalten sind, bei denen es sich nicht um die Prozessvariable handelt, die eine abnormale Variation aufwies, wohingegen der zweite Spaltenblock **604d** beispielsweise die Spalte auswählen kann, in der die Daten für die Prozessvariable enthalten sind, die die abnormale Variation aufwies.

[0335] Der Ausgang jedes Spaltenblocks **604c** und **604d** wird an den PLS-Block **604f** gesendet. Der Eingang **610f2** des Blocks **604f** kann bei einer Ausführungsform so konfiguriert werden, dass er die Werte von unabhängigen, erklärenden oder Eingangsvariablen aufnimmt, wohingegen der Eingang **610f1** des Blocks **604f** so konfiguriert werden kann, dass er die Werte von abhängigen, erklären oder Ausgangsvariablen aufnimmt. Während eine konkrete Anordnung und/oder Funktion der Eingänge zum Modellblock (z. B. der PLS-Block **604f**, im Offline-Diagramm **602**) in Diagramm **602** veranschaulicht wird, können die Anordnung und die Funktion der Eingänge in Abhängigkeit von der Funktion des Blocks, der Eingangstypen, der Anzahl an Eingängen usw. unterschiedlich sein.

[0336] Der Ausgang **608f1** des Blocks **604f** ist mit dem Prüfblock **604g** verbunden. Der Prüfblock **604f** empfängt dementsprechend den vom PLS-Block **604f** ausgegebenen Wert und die Werte der Variable, die die abnormale Variation zeigte. Der Benutzer, welcher den Prüfblock **604g** verwendet, ist in der Lage, zu sehen, wie sich der Ausgang des Blocks **604f** im Vergleich mit der abhängigen Variable zu einem beliebigen Zeitpunkt darstellt, und den Vorhersagewert zu bestimmen. Der Ausgang **608f2** des Blocks **604f**, wenngleich dieser in Fig. 5D nicht mit einem anderen Block verbunden ist, veranschaulicht eine Eigenschaft einiger der Blöcke und Leitungen. Insbesondere handelt es sich bei Ausgang **608f2** um einen Objektausgang. Der Objektausgang ist zur Ausgabe von Code steuerbar; in diesem Fall gibt der Objektausgang beispielsweise das Sensibilitätsmodell entsprechend der Programmierung aus, einschließlich des Modells, der Daten und des Ausgangs. Natürlich bedeutet dies, dass eine mit dem Ausgang **608f2** verbundene Leitung zwangsläufig in der Lage wäre, das Modell durch die Leitung zu transportieren. Der Anschluss der Leitung an den Eingang eines neuen Blocks könnte beispielsweise den Block mit dem gesamten davor geschalteten Diagramm zum Erstellen

len eines Blocks programmieren, der den gesamten Funktionsumfang und alle Daten enthält.

[0337] Dementsprechend, wie jetzt augenscheinlich sein sollte, kann der Benutzer das Offline-Diagramm **602** iterativ überarbeiten und die Ergebnisse prüfen, um ein Diagramm zu erstellen, welches den gewünschten Vorhersagewert für eine bestimmte Variable enthält. Dabei kann der Benutzer unterschiedliche Datensätze, unterschiedliche Teilmengen desselben Datensatzes, unterschiedliche Modellierungs-/Analysetechniken und dergleichen verwenden. Anders ausgedrückt, kann der Benutzer unterschiedliche Datensätze abfragen oder laden (z. B. durch Austauschen oder Modifizieren von Block **604a** des Offline-Diagramms oder durch Hinzufügen zum Block **604a** des Offline-Diagramms **602**), die Daten unterschiedlich segmentieren, indem er versucht, unterschiedliche Variablen vorherzusagen (z. B. durch Ändern der in jedem der Blöcke **604c** und **604d** ausgewählten Spalten), unterschiedliche Arten der Auswertung (PCA, PLS usw.) und/oder unterschiedliche Eigenschaften für eine konkrete Analyse ausprobieren (z. B. durch Austauschen oder Modifizieren des Blocks **604f**) usw.

[0338] In Anbetracht des vorstehenden Beispiels ist klar, dass die DDE-Benutzerschnittstellenanwendung und die dazugehörigen Analysedienste bei vielen Systemen einen vorteilhaften Nutzen aufweisen, bei denen große Mengen an Daten erfasst werden. Im Hinblick auf Prozesssteuerungssysteme haben Prozesssteuerungsumgebungen relativ unlängst die gesammelte und gespeicherte Menge an Daten ausgeweitet, die in einigen Fällen alle in der Umgebung generierten Daten umfasst. Wenngleich ältere System signifikante Daten generierten, verwendeten diese Systeme lediglich einen geringen Teil der Daten zur Prozesssteuerung, wobei die Daten in Prozesssteuerungsalgorithmen eingegeben und einige der Daten beispielsweise an Bedienerarbeitsplätzen angezeigt wurden, wurde eine relativ geringe Menge der Daten für eine spätere Auswertung und/oder Verwendung gespeichert. Der Wert einer Variable, die einmal pro Sekunde abgefragt wurde, konnte beispielsweise lediglich einmal pro Minute gespeichert werden, um den Bedarf an einer nachträglichen Auswertung mit den Grenzen hinsichtlich der Speicher- und Netzwerkkapazitäten im Gleichgewicht zu halten. Ferner wurden gespeicherte Daten oftmals komprimiert, was sich negativ auf die Belastbarkeit der Daten auswirkte.

[0339] Im Gegensatz dazu speichern aktuelle Systeme einen wesentlich größeren Teil der in der Anlage generierten Daten. Bei Ausführungsformen speichern solche Systeme mittlerweile jeden Wert einer Variablen jedes Mal, wenn eine Probe aufgezeichnet wird, und können ferner Daten speichern, die vorher noch nie erfasst wurden, wie beispielswei-

se Eingaben durch den Bediener, Nachrichten zwischen zwei Bedienern, Videos usw. Die hierin beschriebenen DDE-Benutzerschnittstellenanwendung und Analysedienste arbeiten zusammen, um die Prüfung aller vorstehend genannten Daten, zusätzlich zu den anderen Daten (z. B. von anderen Anlagen, Wetterdaten usw.), und ein Erkennen von Beziehungen zu ermöglichen, die vorher unter Umständen nicht erkannt wurden oder nicht erkennbar waren. Durch das Erkennen der Beziehungen zwischen unterschiedlichen Prozesswerten und der Interaktionen von und zwischen Prozesswerten können Anlagentechniker, Bediener und das Wartungspersonal das Design verbessern und die Prozessanlagen bauen, bedienen und instandhalten, was wiederum zu Prozessanlagen führt, die kostengünstiger, effizienter, bedienungs- und wartungsfreundlicher sind, ein besseres Produkt herstellen, weniger umweltschädlich und für das Personal und die umgebenden Gemeinden sicherer sind.

B. Erstellen eines Online-Diagramms

[0340] Natürlich liegt ein Weg der Nutzung der erkannten Beziehungen hinsichtlich des verbesserten Ablaufs des Prozesses darin, die erkannten Beziehungen zu verwenden, um eine kontinuierliche Echtzeitvorhersage vorzunehmen. Insbesondere wenn in historischen Daten des Prozesses eine oder mehrere Beziehungen zwischen Prozesswerten oder anderen Daten erkannt wurden, wie beispielsweise, dass ein Satz Werte verwendet werden kann, um einen anderen Wert oder andere Werte vorherzusagen (oder das Auftreten einiger Ereignisse), können die erkannten Beziehungen verwendet werden, um Echtzeitdaten aus dem Prozess zum Vorhersagen desselben Wertes oder derselben Werte heranzuziehen (oder das Auftreten derselben Ereignisse). Die DDE-Benutzerschnittstellenanwendung und Analysedienste erleichtern die Verwendung der erkannten Beziehungen im Hinblick auf die Durchführung einer vorhersagenden Analyse, siehe Beschreibung unten.

[0341] Die DDE-Benutzerschnittstellenanwendung umfasst Funktionen, die eine Umwandlung eines Offline-Diagramms (wie beispielsweise das Offline-Diagramm **602**) in ein Online-Diagramm ermöglichen (d.h. ein Diagramm, bei dem mindestens ein Echtzeitwert verwendet wird, um einen Aspekt des Anlagenbetriebs vorherzusagen). Wie vorstehend beschrieben, unterscheidet sich ein Online-Diagramm dahingehend von den Offline-Diagrammen, dass es an mindestens eine Quelle für Echtzeitdaten gebunden ist (anstelle von rein historischen Daten) und eine kontinuierliche Echtzeitausgabe bereitstellt, die aufgerufen, gespeichert und/oder in einem Steueralgorithmus verwendet werden kann, um einen Alarm und/oder Warnungen auszulösen und/oder Veränderungen im Betrieb der Prozessanlage hervorzurufen.

[0342] Solange ein Offline-Diagramm mindestens einen modellgenerierenden Block umfasst, kann der Benutzer der DDE-Benutzerschnittstellenanwendung die Online-Umschaltbenutzersteuerung **248f** aktivieren, wodurch das Offline-Diagramm automatisch in ein Online-Diagramm umgewandelt und auf der Arbeitsfläche **245** angezeigt wird. Unter erneuter Bezugnahme auf **Fig. 5A** wird die Umwandlung des Offline-Diagramms in ein Online-Diagramm bei Ausführungsformen durch eine Diagrammumwandlungseinheit **530** im Webserverprozess **504** erreicht.

[0343] In ihrer einfachsten Ausführungsform sucht die Diagrammumwandlungseinheit **530** den modellgenerierenden Block (z. B. den PLS-Block **604f** im Offline-Diagramm **602**), entfernt sie jedwede Ausgabeleitungen (z. B. die Leitung **606h**), fügt sie einen Schreibblock hinzu und verbindet sie eine Leitung zwischen dem Ausgang des modellgenerierenden Blocks und dem Eingang des Schreibblocks. Der Schreibblock schreibt im Allgemeinen die Werte, die durch das Modell ausgegeben werden, in einen Datenspeicherort, einen Prozesssteuerungsalgorithmus (z. B. ausführend einen Steuerungsalgorithmus in einer Steuerung oder in einem Funktionsblock in einer Prozessleitvorrichtung), und/oder in die Übersichtsseite. Die Diagrammumwandlungseinheit **530** ersetzt ferner den Offline-Datenladeblock durch einen entsprechenden Online-Datenladeblock, der, anstatt eine Reihe historischer Daten zu laden, an mindestens eine Echtzeitdatenquelle aus der Prozesssteuerungsumgebung gebunden ist.

[0344] Bei verschiedenen anderen Ausführungsformen führt die Diagrammumwandlungseinheit **530** jedoch mehr aus als den einfachen Austausch von zwei Blöcken im Offline-Diagramm durch. Bei einigen Ausführungsformen entfernt die Diagrammumwandlungseinheit **530** aktiv Blöcke aus dem Offline-Diagramm, um das Online-Diagramm zu erstellen. Benötigt ein Modellblock beispielsweise zwei Eingänge, um ein Modell zu erstellen (z. B. eine Reihe von X-Variablen und eine Y-Variable), würden die beiden Eingänge an unterschiedlichen Blöcken erstellt. Weist das Modell bei der Umsetzung des erstellten Modells jedoch nur einen Eingang auf (d.h. im Online-Diagramm), ist der Block, der vorher dem anderen Eingang Daten geliefert hat, nicht mehr erforderlich und kann entfernt werden. In einem weiteren Beispiel kann die Diagrammumwandlungseinheit **530** Prüfblöcke entfernen (z. B. den Prüfblock **604e** in **Fig. 5D**), wenn sie das Online-Diagramm erstellt.

[0345] Bei einigen Ausführungsformen kann sich der Diagrammwandler **530** auf eine Schemabibliothek beziehen (nicht abgebildet), die sowohl Offline- als auch Online-Schemata für jeden Blocktyp liefert, wobei die Schemata festlegen, welche Eingänge und welche Ausgänge mit der Online- und Offline-Version des Blocks assoziiert sind, auf welche Datenquel-

le(n) zugegriffen wird und dergleichen. Bei alternativen Ausführungsformen legt jede Blockdefinition **255** in der Blockdefinitionsbibliothek **252** sowohl die Online- als auch die Offline-Schemata für den Block fest.

[0346] Bei einigen Ausführungsformen kann die Diagrammumwandlungseinheit **530** so konfiguriert sein, dass sie während der Umwandlung eines Offline-Diagramms in ein Online-Diagramm optionale Abläufe durchführt oder nicht. Beispielsweise können einige der optionalen Abläufe unter anderem Folgendes umfassen: Markieren des vorhergesagten Wertes, so dass dieser als Eingang für einen Prozesssteuerungsalgorithmus verwendet werden kann, Ausgeben einer kontinuierlichen Kurve des vorhergesagten Wertes auf dem Übersichtsbildschirm und Schreiben des vorhergesagten Wertes in die verteilte Datenengine. In einigen Fällen kann die mindestens kleine Menge aus Ergebnissen, die durch die Diagrammumwandlungseinheit **530** erzeugt wurden, zu Analysezwecken und/oder zur Visualisierung des Steuerungssystems verwendet werden (wie beispielsweise Faceplates, Trendicons, usw.).

[0347] Die Echtzeitquellen, an die die Datenladeblöcke (Laden, Abfragen usw.) gebunden sind, der Ort/ die Orte, an den/die der Ausgang des Modellblocks berichtet, und/oder die virtuellen Markierungen, die mit den ausgegebenen Daten assoziiert sind, sind bei Ausführungsformen jeweils ebenfalls in dem Maße konfigurierbar, dass eine bestimmte Ausführungsform jede Option umsetzt. Bei Ausführungsformen ist jede in der Umgebung der Anwendungseinheit **514** konfiguriert, die sich im Webclientprozess **502** beispielsweise als Moduleigenschaft und/oder als Blockeigenschaft befindet. Beispielsweise können die Moduleigenschaften für ein Modul einen Ort, an dem Daten eingetragen werden, und einen Ort umfassen, an dem Echtzeitdaten gebunden werden, wie im Allgemeinen in **Fig. 5F** veranschaulicht. In **Fig. 5F** ist ein Moduleigenschaftendialog **630** für das Modul veranschaulicht, wenn ein Benutzer eine Moduleigenschaftensteuerung **632** aktiviert. Der Moduleigenschaftendialog **630** umfasst ein Modulbezeichnungsfeld **634**, in das der Benutzer einen Namen für das Modul eingegeben kann. Der Moduleigenschaftendialog **630** umfasst ferner ein Post-Feld **636**, durch das ein Benutzer vorgeben kann, wo die Daten, die während der Ausführung durch das Online-Diagramm ausgegeben werden, hingeschrieben werden sollen (d.h. eingetragen). Bei dem Ort, an dem die Daten eingetragen werden, kann es sich um einen Speicherort, an dem die Daten historisch geordnet werden, ein Steuerungsmodul in der Prozesssteuerungsumgebung, welches die eingetragenen Werte als Eingaben für einen Steuerungsalgorithmus verwendet, um den Betrieb der Prozessanlage zu steuern, oder beides handeln. Eine Benutzersteuerung **636a** kann ein Explorer-Fenster öffnen, in dem der Benutzer den Ort auswählen kann, an dem die Daten eingetragen

werden, indem der Benutzer mögliche Orte in einem grafischen Format angezeigt bekommt (z. B. in einem Baumformat, in einer Verzeichnisstruktur, in einer Liste optionaler Orte usw.). Gleichermaßen kann der Benutzer während der Ausführung des Online-Diagramms durch ein Feld **638** einen Datenpfad, an den eine Bindung erfolgen soll, als eine Quelle für Online-Daten vorgeben. Eine Benutzersteuerung **638a** öffnet gleichermaßen ein Explorer-Fenster, über das der Benutzer den Ort auswählen kann, von dem die Daten empfangen werden.

[0348] Alternativ kann der Offline-Block eine Eigenschaft in Verbindung mit der Offline-Version des Blocks enthalten und kann die Online-Version des Blocks eine ähnliche Eigenschaft für die Online-Version des Blocks enthalten. Beispielsweise kann ein Ladeblock eine „Quellpfad“-Eigenschaft (z. B. \DDE_path\device_tag) in der Offline-Version des Blocks aufweisen, in der Online-Version des Blocks jedoch eine „Vorrichtungsmarkierung“-Eigenschaft (z. B. \unit1\device_tag) aufweisen, wodurch der Ladeblock Echtzeitdaten aus der Quelle auslesen kann, die den im Quellpfad des Online-Blocks gespeicherten Daten entsprechen. Gleichermaßen kann ein Schreibblock im Online-Diagramm eine Eigenschaft aufweisen, die den Ort vorgibt, in dem die ausgegebenen Daten geschrieben werden sollten.

[0349] Bei noch einer anderen Alternative, diesmal unter Bezugnahme auf **Fig. 5G**, können Blöcke in einem Offline-Diagramm **640** jeweils für den Offline- und Online-Betrieb konfiguriert sein. Beispielsweise kann ein Ladeblock **642** ausgewählt werden und einen assoziierten Blockeigenschaftendialog **644** aufweisen. Der Blockeigenschaftendialog **644** umfasst ein Blockbezeichnungsfeld **646**, in das der Benutzer einen Namen für den Block eingeben kann. Der Blockeigenschaftendialog **644** umfasst ferner ein Offline-Quellenfeld **648**, in dem ein Benutzer vorgeben kann, wo die Daten, die in das Offline-Diagramm geladen werden sollen, zu finden sind. Bei dem Ort, an dem die zu ladenden Daten gespeichert sind, kann es sich beispielsweise um einen Speicherort handeln, an dem die Daten historisch geordnet werden. Eine Benutzersteuerung **648a** kann ein Explorer-Fenster öffnen, in dem der Benutzer den Ort auswählen kann, von dem die Daten geladen werden, indem er die verfügbaren Orte im grafischen Format angezeigt bekommt (z. B. in einem Baumformat, in einer Verzeichnisstruktur, in einer Liste mit optionalen Orten usw.). Gleichermaßen kann der Benutzer durch ein Feld **649** einen Datenpfad, an den die Bindung erfolgen soll, während der Ausführung des Online-Diagramms als Quelle für Online-Daten festlegen. Eine Benutzersteuerung **649a** öffnet gleichermaßen ein Explorer-Fenster, in dem der Benutzer den Ort auswählen kann, von dem die Daten empfangen werden. Natürlich kann es sich bei den Steuerungen **648**, **648a**, **649**, **649a** um beliebige Steuerungen handeln,

die in der Blockdefinition vorgegeben sind und die Ausführung entsprechender Plugins hervorrufen können, siehe vorstehende Beschreibung.

[0350] Bei Ausführungsformen werden die Beziehungen zwischen den Pfaden zu gespeicherten, historisch geordneten Daten, von denen ein Offline-Block Daten abfragen würde, auf der einen Seite, und einer entsprechenden Quelle für Echtzeitdaten, an die ein Online-Block gebunden werden könnte, auf der anderen Seite, durch ein Datenquellenverwaltungsmodul **532** verwaltet. Das Datenquellenverwaltungsmodul **532** kann als ein Teil des Webserververprozesses **502** integriert sein, als ein Teil der Diagrammwandlungseinheit **530** enthalten sein oder einfach eine Nachschlagetabelle sein, auf die sich die Diagrammwandlungseinheit **530** bezieht, wenn diese ein Offline-Diagramm in ein Online-Diagramm umwandelt. Bei mindestens einigen Ausführungsformen arbeitet das Datenquellenverwaltungsmodul **532** mit der Großdatenfrageverwaltungseinheit **165** zusammen, um einen Echtzeitdatenstrom entsprechend der Anforderungen eines Online-Blocks einzurichten.

[0351] **Fig. 5E** veranschaulicht ein Online-Diagramm **622**, welches dem Offline-Diagramm **602** in **Fig. 5D** entspricht. Das Online-Diagramm **622** umfasst die Blöcke **624a**, **624b**, **624c**, **624f** und **624h** und die Leitungen **626a**, **626b**, **626e** und **626j**. Wenn die Blöcke und/oder Leitungen denen im Offline-Diagramm **602** entsprechen, sind die mit den Blöcken assoziierten Buchstaben gleich.

[0352] Betrieb und Ausführung des Online-Diagramms **622** erfolgen im Allgemeinen identisch mit dem Offline-Diagramm **602**, siehe Beschreibung oben. Das Online-Diagramm **622** kann, sobald alle Eigenschaften konfiguriert sind (und die assoziierten Parameter in der Konfigurationsdatei/in den Konfigurationsdateien gespeichert sind), „eingesetzt“ werden, indem die Einsatzbenutzerschnittstellensteuerung **248j** aktiviert wird. Durch die Aktivierung der Steuerung **248j** wird die Konfigurationsdatei an den Ausführungsdienstprozess **506** gesendet, wo sie entsprechend der Parameter in der Konfigurationsdatei übersetzt wird. Bei Ausführung als ein oder mehrere Arbeitsprozesse **508** empfangen die Arbeitsprozesse **508** jedoch, anstatt Daten aus dem Großdatenspeicher **155** über die Analysenetzwerkschnittstelle **175** herunterzuladen oder abzurufen, Echtzeitdaten beispielsweise von Großdatenempfängern oder direkt durch einen Strom, der durch eine der Großdatenfrageverwaltungseinheiten **165** eingerichtet wird. Anstatt Ausgabedaten zurück zu den Auftragsausführungseinheiten **528** zu senden, um lediglich an die Anwendungseinheit **514** über die Serverkommunikationseinheit **518** des Webserverprozesses **504** zur Anzeige auf dem Übersichtsbildschirm zu senden, können die Arbeitsprozesse **508** Ausgabedaten an

den Großdatenspeicher **155** und/oder an eine oder mehrere der Datenquellen (z. B. Steuerungen, andere DDE usw.) zurückschicken.

ABFRAGESPRACHE FÜR INDUSTRIELLE LEISTUNGSÜBERWACHUNG/-ANALYSE

[0353] Die mit einem Prozesssteuerungssystem oder einer Prozesssteuerungsumgebung, auf dem/der das industrielle Leistungsüberwachungs-/analysesystem **100** arbeitet, assoziierten Datenquellen bieten in der Regel Zeitreihendaten, wenngleich andere Datentypen verwendet werden können (z. B. Querschnittsdaten von einer Vielzahl an Chargen, die in einer oder mehreren Prozessanlagen **5** separat ausgeführt wurden). Zeitreihendaten umfassen unterschiedliche Typen an Datenmessungen unter Verwendung unterschiedlicher Arten an Messgeräten in der Prozessanlage **5**, einschließlich der Feldgeräte **15–22** und **40–46**. Die Datenquellen können hinsichtlich ihres Formats innerhalb einer breiten Palette von allgemein bekannten bis hin zu firmeneigenen Formaten variieren, z. B. OSIsoft PI, DeltaV Historian, SEEQ, FF3 und/oder manuell erfasste Formate in Excel-Tabellen. Einige Datenquellen können relationale Datenbanken enthalten, während andere Datenquellen nicht-relationale (NoSQL) Datenbanken enthalten können. Bei noch anderen Datenquellen handelt es sich eventuell nicht um Datenbanken, sondern es werden anstelle dessen Dateiverzeichnisse oder Text in einem Dokument verwendet (z. B. ein XML-Dokument), um Daten abzuspeichern. Zusätzlich zu den Unterschieden hinsichtlich der Abfragesyntax kann die Vielzahl an Datenquellen aufgrund der Unterschiede hinsichtlich der Art und Weise, wie Daten gespeichert werden, grundlegend unterschiedliche Abfragestrukturen erforderlich machen. Beispielsweise speichern dokumentenorientierte nicht-relationale Datenbanken, wie beispielsweise Mongo, die Daten auf der Grundlage von Dokumenten, anstatt in Tabellen, die über SQL-Abfragen in relationalen Datenbanken verfügbar sind, wie beispielsweise MySQL. Dementsprechend setzen Abfragen nach Daten, die in unterschiedlichen Typen von Datenquellen abgespeichert sind, unterschiedliche Strukturen und Regeln ein, zusätzlich zu den Unterschieden hinsichtlich Formatierung und Syntax. Anstelle der Verwendung jedes natürlichen Abfragemechanismus der unterschiedlichen Datenquellen, um auf die darin gespeicherten Daten zuzugreifen, verwenden die hierin beschriebenen industriellen Leistungsüberwachungs-/analysesysteme und -techniken (und insbesondere die DDE-Benutzerschnittstellenanwendung) eine standardisierte Abfrage zum Herstellen einer Schnittstelle mit jeder der Datenquellen **702a–702f**, wie in **Fig. 6A** veranschaulicht.

[0354] **Fig. 6A** veranschaulicht ein Blockdiagramm mit verschiedenen Datenquellen **702a–702f** und Datenbereitstellungseinheiten **704a–704d**, die kommunikativ mit einem Abfrageblock **708** der DDE-Benutzerschnittstellenanwendung verbunden sind, über einen Abfrageausführungsdienst **706**. Bei jeder der Datenquellen **702a–702f** handelt es sich um eine elektronische Datenquelle, die Daten in einem elektronisch lesbaren Format speichert. Einige Datenquellen **702e** und **702f** bilden ohne eine dazwischengeschaltete Datenbereitstellungseinheit **704** eine Schnittstelle mit dem Abfrageausführungsdienst **706**, wie beispielsweise durch eine interne oder Netzwerksdatenverbindung. Andere Datenquellen **702a–702d** bilden über eine oder mehrere Datenquellenbereitstellungseinheiten **704a–704d** eine Schnittstelle mit dem Abfrageausführungsdienst **706**. Die Datenbereitstellungseinheiten **704a–704d** können so konfiguriert sein, dass sie auf Daten zugreifen, Daten suchen, Daten sortieren, Daten lesen und/oder Daten schreiben, jeweils in die oder von den entsprechenden Datenquellen **702a–702d**. Bei einigen Ausführungsformen können die Datenbereitstellungseinheiten **704a–704d** datenquellspezifische Abfragen empfangen, in deren Rahmen die jeweiligen datenquellspezifischen Abfrageformate der Datenquellen **702a–702d** verwendet werden. Bei anderen Ausführungsformen können die Datenquellenbereitstellungseinheiten **704a–704d** so konfiguriert sein, dass sie eine standardisierte Abfrage **709**, die eine individuell angepasste Abfragesprache verwendet, vom Abfrageblock **708** empfangen und die standardisierte Abfrage in ein entsprechendes datenquellspezifisches Abfrageformat umwandeln und dazu die konkreten Abfragemechanismen der Zieldatenquellen **702a–702d** verwenden. Die Datenbereitstellungseinheiten **704a–704d** oder die Datenquellen **702e–702f** können eine Ausführungsengine (nicht abgebildet) enthalten, die datenquellspezifischen Abfragen für eine spezifische Datenquelle **702** verarbeitet und ausführt. Die Ausführungsengine kann ein Bestandteil der Datenquelle **702** selbst oder ein Bestandteil der mit der Datenquelle **702** assoziierten Datenbereitstellungseinheit **704** sein (z. B. die Datenbereitstellungseinheit **704b**, die mit der Datenquelle **702b** assoziiert ist). Bei jeder Datenquelle **702** kann es sich um eine Datenbank oder einen Dokumentenspeicher handeln, wie beispielsweise ein Dokumentenspeicher, der eine Vielzahl von JavaScript-Object-Notation-Dateien (JSON) enthält, die wiederum Daten enthalten. Bei einigen Ausführungsformen können die Datenquellen **702a–702f** sogar Live-Datenquellen umfassen, wie beispielsweise Datenströme, die durch Feldgeräte **15–22** und **40–46** der Prozessanlage **5** erzeugt wurden. Solche Live-Datenquellen können dadurch abgefragt werden, dass abgerufene Daten analysiert werden und lediglich ein Teil der konkret angefragten Daten zurückbehalten oder extrahiert wird.

[0355] Die Datenquellen **702a–702f** können so konfiguriert sein, dass sie Daten in einem beliebigen bekannten oder hiernach entwickelten Format spei-

chern. Die Datenquellen **702a**, **702b** und **702e** sind als Datenbanken veranschaulicht, um die Daten darzustellen, die in relationalen Datenbanken gespeichert sind. Beispiele für solche relationalen Datenbanken sind unter anderem MySQL oder andere Datenbanken, bei denen die Daten in Tabellen abgelegt werden und eine SQL-formatierte Abfrage der Daten zur Anwendung kommt. Die Datenquellen **702c**, **702d** und **702f** sind als Sammlungen von Dateien oder Dateneinträgen veranschaulicht, um nicht-relationale Datenspeicher darzustellen, wie beispielsweise NoSQL-Datenbanken oder Datenquellen, bei denen es sich nicht um eine Datenbank handelt. Beispiele für nicht-relationale Datenbanken sind unter anderem dokumentenorientierte Datenbanken, wie beispielsweise MongoDB oder CouchDB, bei denen die Daten auf der Grundlage von Dokumenten und nicht auf der Grundlage von Tabellen gespeichert werden. Da das Speichern oder Suchen von extrem umfangreichen Daten oder komplexen Datensätzen unter Verwendung nicht-relationaler Datenbanken effizienter sein kann, werden solche Datenbanken häufig zur Auswertung großer Datenmengen herangezogen. Die Abfrage solcher nicht-relationalen Datenbanken erfordert jedoch unterschiedliche Techniken und eine unterschiedliche Syntax, da die Daten in solchen nicht-relationalen Datenbanken nicht allgemein in tabellarischer Form angeordnet sind (welches die Grundlage der SQL-Datenbanken bildet). Da die SQL-Abfrage weit verbreitet verwendet wird und hingänglich bekannt ist, wird sie auch häufig zum Speichern und Auswerten von Daten verwendet. Durch die Verwendung eines standardisierten Abfrageformats für die standardisierte Abfrage **709**, welche in eine datenquellspezifische Abfrage umgewandelt wird, die ein datenquellspezifisches Format verwendet, kann ein Benutzer durch die hierin offen gelegte Erfindung auf Daten in SQL- oder NoSQL-Datenbanken zugreifen und dabei dasselbe Abfrageformat verwenden. Das standardisierte Abfrageformat ermöglicht ferner die Kombination von Daten aus mehreren unterschiedlichen Typen von Datenquellen **702** in einem Datensatz mit einer konsistenten Datenstruktur und einem konsistenten Datenformat. Dementsprechend kann der Abfrageblock **708** als ein Superverbinder fungieren, der Datenquellen **702** mit unterschiedlichen Datenstrukturen, Datenformaten und Abfragemechanismen unter Verwendung standardisierter Abfragen **709** miteinander verbindet.

[0356] Bei Abfrageblock **708** kann es sich um einen Block handeln, der innerhalb des Datenanalysestudios umgesetzt ist, wie vorstehend erörtert, um Daten vorzugeben, die aus den Datenquellen **702a–702f** entnommen werden sollen. Beispielsweise kann es sich beim Abfrageblock **708** um eine Konfigurationsdatei mit einer oder mehreren Blockeigenschaften handeln, einschließlich einer Eigenschaft, die die standardisierte Abfrage **709** vorgibt. Bei der standardisierten Abfrage **709** kann es sich um eine sepa-

rate Datei handeln (wie beispielsweise eine JavaScript-Object-Notation-Datei), auf die sich der Abfrageblock **708** bezieht. Bei Abfrageblock **708** kann es sich alternativ um ein beliebiges Objekt handeln, welches die standardisierte Abfrage **709** enthält oder anzeigen, wie beispielsweise ein Prozess oder eine Routine, der/die innerhalb einer beliebigen der verteilten Datenengines **150** arbeitet. Der Abfrageblock **708** empfängt die standardisierte Abfrage **709** (wie beispielsweise durch eine Auswahl oder Eingabe des Benutzers), und die standardisierte Abfrage **709** verwendet ein standardisiertes Abfrageformat. Durch die Verwendung eines standardisierten Abfrageformats können die Datenquellen **702a–702f** abgefragt werden, ohne dass der Benutzer, Techniker oder die datenabfragende Einheit Kenntnis von den besonderen Typen oder Strukturen der Daten in den Datenquellen **702a–702f** haben muss. Das standardisierte Abfrageformat kann entweder ein vorhandenes Abfrageformat sein, welches durch eine Datenquelle verwendet wird, oder ein anderes Abfrageformat, welches nicht direkt durch irgendwelche Datenquellen verwendet wird. Im letzteren Fall sind standardisierte Abfragen, in deren Rahmen die Syntax des standardisierten Abfrageformats verwendet wird, nicht direkt dahingehend ausführbar oder umsetzbar, um Daten von den Datenquellen **702a–702f** abzurufen. Die standardisierte Abfrage kann eine oder mehrere der Datenquellen **702a–702f** vorgeben (oder eine oder mehrere der Datenbereitstellungseinheiten **704a–704d**), von denen die Daten abgerufen werden sollen. Alternativ kann der Abfrageblock **708** eine Eigenschaft aufweisen, die eine oder mehrere der Datenquellen **702a–702f** vorgibt.

[0357] Der Abfrageausführungsdienst **706** empfängt eine standardisierte Abfrage **709** vom Abfrageblock **708** und veranlasst die Abfrage einer oder mehrerer der Datenquellen **702a–702f**. Der Empfang der standardisierten Abfrage **709** vom Abfrageblock **708** kann den Empfang einer Datei umfassen, die Abfrageparameter enthält, aus denen die Daten hervorgehen, die von den Datenquellen **702a–702f** abgerufen werden sollen. Die Veranlassung der Abfrage der Datenquellen **702a–702f** kann die Extraktion der Abfrageparameter und die Erstellung einer oder mehrerer datenquellspezifischer Abfragen auf der Grundlage der Abfrageparameter umfassen. Jede datenquellspezifische Abfrage kann durch den Ausführungsdienst **706** ausgeführt oder zur Ausführung an die Datenquellen **702e–702f** oder die Datenbereitstellungseinheiten **704a–704d** gesendet werden, um die in den Abfrageparametern geforderten Daten abzurufen. Solche Abfrageparameter können spezifische Daten vorgeben, die von einer oder mehreren Datenquellen **702** abgerufen werden sollen, bei denen es sich um Messdaten (wie beispielsweise Messungen von Feldgeräten **15–22** und **40–46**), rechnerisch ermittelte Maße oder Maße, die anderweitig von Messdaten abgeleitet wurden, und/oder Metada-

ten hinsichtlich der Datenquellen **702** oder der darin gespeicherten Daten handeln kann. Beispielsweise können solche Metadaten Hinweise auf Typen, Quellen oder die Qualität von Daten enthalten, die in einer Datenquelle **702** gespeichert sind, einschließlich der Arten der an den Daten durchgeführten Berechnungen. Solche Metadaten können bei der Erstellung von Prozessmodellen oder einer Wissensbasis aus den Datenquellen **702** des Systems nützlich sein.

[0358] Bei dem Abfrageausführungsdienst **706** kann es sich um den vorstehend erörterten Ausführungsdienstprozess **506** handeln. Der Abfrageausführungsdienst **706** kann beliebige der Bestandteile des Analysedienstes **500** enthalten, einschließlich der Datendienste **520**, der Aufgabenempfangseinheit **522**, der Auftragsverwaltungseinheiten **524**, die Auftragsausführungseinheiten **528**, oder die Datenquellenverwaltungseinheit **532**. Beispielsweise kann der Abfrageblock **708** durch die Anwendung **514** mit einer Eingabe von einem Benutzer erstellt werden, einschließlich der standardisierten Abfrage **709**, die in einer JSON-Datei gespeichert ist. Nach dem Empfang einer Benutzerabfrage oder einem auslösenden Ereignis kann die Anwendung **514** die Kommunikation der standardisierten Abfrage **709** von dem oder durch den Webserverprozess **504** zur Aufgabenempfangseinheit **522** des Ausführungsdienstprozesses **506** veranlassen (wobei der Ausführungsdienstprozess **506** als Ausführungsdienst **706** dienen kann). Die Aufgabenempfangseinheit **522** kann den Empfang der standardisierten Abfrage **709** durch eine oder mehrere Auftragsverwaltungseinheiten **524** veranlassen, die ferner eine oder mehrere Auftragsausführungseinheiten **528** dazu veranlassen können, die standardisierte Abfrage **709** in eine oder mehrere datenquellspezifische Abfragen umzuwandeln, die mit einem oder mehreren Arbeitsprozessen **508** assoziiert sind. Die Arbeitsprozesse **508** können anschließend die eine oder mehreren Datenquellen **702** (d.h. die Datenquellen **168** oder den Großdatenspeicher **155**) dazu veranlassen, unter Verwendung der datenquellspezifischen Abfragen abgefragt zu werden. Die durch das Abfragen der einen oder mehreren Datenquellen **702** abgerufenen Daten können anschließend von den Arbeitsprozessen **508**, dem Ausführungsdienstprozess **506**, den Datendiensten **520** und/oder der Anwendung **514** empfangen werden.

[0359] Bei einigen Ausführungsformen wandelt der Abfrageausführungsdienst **706** die standardisierte Abfrage **709** in datenquellspezifische Abfragen um, welche datenquellspezifische Formate verwenden, die den Datenquellen **702e** oder **702f** heimisch sind. Bei alternativen Ausführungsformen kann der Abfrageausführungsdienst **706** eine oder mehrere Datenquellen **702a–702f** zur Abfrage bestimmen und einer oder mehreren Datenbereitstellungeinheiten **704a–704d** die standardisierte Abfrage

709 bereitstellen, damit diese in datenquellspezifische Abfragen umgewandelt wird. Die Datenbereitstellungseinheiten **704** können separat von den Datenquellen **702** oder mit den Datenquellen **702** kombiniert sein. Die Datenbereitstellungseinheiten **704a** und **704d** werden dahingehend veranschaulicht, dass sie kommunikativ mit den Datenquellen **702a** bzw. **702d** verbunden sind, aber dennoch separat sind. Im Gegensatz dazu sind die Datenbereitstellungseinheiten **704b** und **704c** dahingehend veranschaulicht, dass sie die Datenquellen **702b** und **702c** enthalten bzw. mit diesen kombiniert sind. Beispielsweise kann es sich bei der Datenbereitstellungseinheit **704b** um einen Server (wie beispielsweise einen Datenhistoriker) oder ein Datenbankschnittstellenprogramm handeln, in dem die Datenquelle **720b** als Datenbank gespeichert ist. In einem weiteren Beispiel kann es sich bei der Datenbereitstellungseinheit **704a** gleichermaßen um einen Server oder ein Datenbankschnittstellenprogramm handeln, welcher/s mit einer externen Datenquelle **702a** verbunden ist, wie beispielsweise eine Datenbank, die in einer externen Speichervorrichtung kommunikativ mit dem Server verbunden ist. In noch einem weiteren Beispiel kann es sich bei den Datenbereitstellungseinheiten **704a–704d** um Arbeitsprozesse **508** handeln, die Algorithmen enthalten, um die standardisierte Abfrage **709** in datenquellspezifische Abfragen umzuwandeln, wenn sie die standardisierte Abfrage **709** oder die Abfrageparameter von der standardisierten Abfrage **709** vom Abfrageausführungsdienst **706** empfangen.

[0360] Um die Verwendung der standardisierten Abfragen im Rahmen der Prozesssteuerung und Prozessanalyse zu veranschaulichen, veranschaulicht **Fig. 6B** ein Blockdiagramm des Abfrageblocks **708** im Datenanalysestudio. Der Abfrageblock **708** ist mit der standardisierten Abfrage **709** assoziiert, die Prozessdaten von einer oder mehreren Datenquellen **702a–702f** abruft. Beispielsweise kann ein Benutzer die standardisierte Abfrage **709** im standardisierten Abfrageformat in einem Feld des Abfrageblocks **708** definieren. Die standardisierte Abfrage **709** kann zusammen mit dem Abfrageblock **708** oder in einer separaten Datei gespeichert werden. Beispielsweise kann es sich bei dem Abfrageblock **708** um eine Konfigurationsdatei handeln, die eine Abfrageeigenschaft enthält, die eine Kennung einer JSON-formatierten Datei speichert, die die standardisierte Abfrage **709** enthält. Nach dem Auftreten eines Ereignisses (wie beispielsweise das Auftreten einer Zeit, eines Prozesssteuerungszustandes oder einer Auswahl durch den Benutzer), kann die standardisierte Abfrage **709** durch den Abfrageblock **708** erkannt und an den Abfrageausführungsdienst **706** gesendet werden, um die angeforderten Daten von einer oder mehreren der Datenquellen **702a–702f** abzurufen. Die im Rahmen der standardisierten Abfrage **709** geforderten Daten oder Informationen im Zusammenhang mit solchen

Daten (z. B. Zusammenfassungsinformationen, Metadaten zur Validierung usw.) können an den Abfrageblock **708** oder eine andere Datenempfangseinheit zurückgeschickt werden. Eine solche andere Datenempfangseinheit kann einen anderen Block im Datenanalysestudio, einen Arbeitsprozess **508**, den Abfrageausführungsdienst **706**, Datendienste **520**, die Anwendung **514**, einen Programmspeicher oder einen beliebigen anderen Dienst, eine beliebige andere Routine, einen beliebigen anderen Prozess oder eine beliebige andere Vorrichtung umfassen, der/die Daten in Verbindung mit der Prozessanlage **5** weiter auswerten, darstellen oder speichern kann. Wie vorstehend erörtert, können die Prozessdaten bei einigen Ausführungsformen über eine oder mehrere Datenbereitstellungseinheiten **704** abgerufen werden. Die abgerufenen Daten können am Abfrageblock **708**, am Abfrageausführungsdienst **706** oder an einer anderen Datenempfangseinheit im gewünschten Format empfangen oder durch den Abfrageblock **708**, den Abfrageausführungsdienst **706** oder eine andere Datenverarbeitungseinheit formatiert werden. Beispielsweise kann der Abfrageausführungsdienst **706** einen Datenrahmen erstellen, indem er die Umwandlung der standardisierten Abfrage **709** in eine oder mehrere datenquellspezifische Abfragen und die Formatierung der Ergebnisse solcher quellspezifischen Abfragen in der gewünschten Art und Weise veranlasst. Solche formatierten Datenrahmen können bei einigen Ausführungsformen an den Abfrageblock **708** zurückgegeben werden. Die von der einen oder den mehreren Datenquellen **702** abgerufenen Daten können eine Vielzahl an Datenpunkten umfassen, die unter Verwendung der datenquellspezifischen Abfrage abgerufen wurden, wobei die Datenpunkte Prozessvariablen entsprechen können, wie beispielsweise Messungen in der Prozessanlage **5** oder Werte, die von solchen Messungen abgeleitet wurden. Solche Datenpunkte können Punkte in einer Zeitreihe darstellen, wobei jedem Punkt ein Zeitstempel zugeordnet ist. Alternativ können solche Daten Querschnittsdaten darstellen, die mit einem Ort, einer Prozesscharge oder anderen Erkennungsmerkmalen assoziiert sein können. Die durch die Ausführung der standardisierten Abfrage abgerufenen Daten können als Datenrahmen bezeichnet werden, um deren variable Struktur zu kennzeichnen, bei der ein tabellarisches Format verwendet werden kann oder nicht.

[0361] Der Abfrageblock **708** kann den unter Verwendung der standardisierten Abfrage erhaltenen Datenrahmen dem Füllblock **710** zur Verfügung stellen, der leere oder nicht numerische (NaN) Einträge im Datensatz entsprechend der mit dem Füllblock **710** assoziierten Regeln befüllt. Dies kann erfolgen, um die Daten zu bereinigen oder eine Überprüfung auf fehlerhafte Daten oder Hinweise auf Fehler im Datenrahmen durchzuführen, sowie um Datenpunkte hinzuzufügen, die benötigt werden, um die gewünschte Abtastrate zu erreichen, wie andernorts

hierin beschrieben. Der Datenrahmen kann anschließend für eine weitere Auswertung und/oder Prozesssteuerung verwendet werden. Wie veranschaulicht, können ein oder mehrere Datensätze aus dem Datenrahmen durch den unabhängigen Variablenblock **712** und den abhängigen Variablenblock **714** ausgewählt werden. Beispielsweise kann der abhängige Variablenblock **712** einen oder mehrere Datentypen aus dem Datenrahmen auswählen, den er vom Füllblock **710** erhalten hat, die Daten entsprechen, die mit vorgegebenen Parametern oder Eigenschaften übereinstimmen (z. B. Druck- oder Temperaturmessungen), oder Datenspalten (wenn der Datenrahmen einer Datentabelle entspricht). Der abhängige Variablenblock **714** kann gleichermaßen verwendet werden, um einen oder mehrere Datentypen aus dem Datenrahmen auszuwählen, den er vom Füllblock **710** erhalten hat. Wenn gleich zwei Blöcke **712** und **714** veranschaulicht sind, kann eine beliebige Anzahl von ähnlichen Blöcken Anwendung finden. Der Analyseblock **716** kann anschließend die Daten empfangen, die durch den unabhängigen Variablenblock **712** und den abhängigen Variablenblock **714** ausgewählt wurden, wie durch die Verbinde zwischen den Blöcken veranschaulicht. Der Analyseblock **716** kann eine beliebige Analyse durchführen, für die er konfiguriert ist. Beispielsweise kann der Analyseblock **716** eine Analyse nach der Methode der partiellen kleinsten Quadrate (PLS) durchführen, um die Auswirkungen der Daten des unabhängigen Variablenblocks **712** auf die Daten des abhängigen Variablenblocks **714** zu bestimmen. Ein Prüfblock **718** kann mit dem Analyseblock **716** und dem abhängigen Variablenblock **714** verbunden sein, um die ordnungsgemäße Konfiguration des Abfrageblocks **708** und der anderen Blöcke und Verbindungen zu überprüfen. Dieser Prüfblock **718** kann grafische, tabelliforme oder Textausgaben generieren, die der Benutzer speichern oder aufrufen kann. Wenn gleich die vorstehende Beschreibung die Blöcke **708-718** dahingehend erörtert, dass diese bestimmte Handlungen vornehmen, wird vorausgesetzt, dass diese Blöcke anstelle dessen den Webserverprozess **504**, den Ausführungsdienstprozess **506** und/oder die Arbeitsprozesse **508** des Analysedienstes **500** zur Durchführung dieser Handlungen veranlassen können, wie anderswo hierin erörtert.

[0362] Um die Verwendung standardisierter Abfragen **709** weiter zu veranschaulichen, veranschaulicht Fig. 6C eine beispielhafte Abfrage, die ein beispielhaftes standardisiertes Abfrageformat verwendet. Die beispielhafte Abfrage wird dahingehend dargestellt, dass sie ein JSON-konformes Format verwendet, um die betriebssystemübergreifende Kompatibilität zu maximieren, wobei jedoch ein beliebiges anderes Format verwendet werden kann (z. B. ein XML-basiertes Format, ein CSV-basiertes Format usw.). Die beispielhafte Abfrage beginnt mit einer Abfragebezeichnung, gefolgt von einem Hinweis auf das

Format, welches zu verwenden ist, um den Datenrahmen (welches als JSON vorgegeben ist) zurückzugeben, und von einem Hinweis auf eine Version des Abfrageformates. Das Datenfeld „timeSelector“ zeigt eine Vielzahl von Zeitrahmen, die Zeitbereiche, in denen die Daten zurückzugeben sind, durch die Objekte „startTime“ und „endTime“ angeben. Bei der beispielhaften Abfrage werden die Start- und Endzeiten im Zusammenhang mit Kalenderdaten und Uhrzeiten in der koordinierten Weltzeit vorgegeben, getrennt durch den Buchstaben „T“. Wie durch die beispielhafte Abfrage angedeutet, kann jeder der Zeitrahmen durch ausgeschlossene Zeiträume getrennt werden, für die keine Daten abgefragt werden und die den Zeiträumen zwischen der Endzeit eines des Zeitrahmens und der Startzeit eines anderen Zeitrahmens entsprechen. Im Anschluss an das Datenfeld „timeSelector“ gibt das Objekt „sampleRateSecs“ eine Abtastrate in Sekunden vor, wobei vorgegeben wird, wie viele Datenpunkte pro Zeiteinheit abgerufen werden sollen. Der finale Anteil der beispielhaften Abfrage besteht aus einem Datenfeld „Spalten“, aus dem die Parameter der aus der Datenquelle **702** abzurufenden Daten hervorgehen (d.h. die abzurufenden Prozessvariablen). Die beispielhafte Abfrage gibt vier Arten oder Spalten von Daten vor, die zurückzugeben sind. Jede der vier Spalten wird durch ein „tag“-Objekt kenntlich gemacht, welches eine Datenquelle und/oder eine Prozessvariable einer Datenquelle vorgibt (d.h. „FT630B/DENS.CV“, „PT 615/WIRED_PSIA.CV“, „TT6079/INPUT_1.CV“ und „630.molefrac.c5“). Jede der vier Spalten umfasst ein „Alias“-Objekt, um die zurückgegebenen Daten innerhalb des Datenrahmens zu kennzeichnen, ein „dataType“-Objekt, um den Datentyp der Daten vorzugeben (z. B. Fließkommawert, langer ganzzahliger Wert, Textfolge usw.), ein „renderType“-Objekt (z. B. Werte, Zählwerte usw.), und ein „Format“-Objekt, welches das Format der zurückzugebenden Daten vorgibt (d.h. „0.###“ deutet auf ein Format hin, welches aus einem Fließkommawert mit drei Nachkommastellen besteht).

[0363] Die mit den Spalten assoziierten „tag“-Objekte können eine Datenquelle **702** kennzeichnen, von der die Daten abgerufen werden sollen. Alternativ, wenn die beispielhafte Abfrage nicht ausdrücklich die Datenquelle **702** vorgibt, kann eine solche Datenquelle oder können solche Datenquellen **702a–702f** durch andere Eigenschaften des Blocks **708** vorgegeben werden. Bei ähnlichen Abfragen, bei denen das standardisierte Abfrageformat verwendet wird, kann die Datenquelle oder können die Datenquellen **702** ausdrücklich durch ein Objekt in der Abfrage vorgegeben werden. Eine solche Vorgabe kann einen Pfad zum Quellenobjekt oder einen Wert umfassen, der den Typ der Datenquelle (z. B. MongoDB, CouchDB, SQL usw.) angibt. Die Vorgabe der Datenquelle **702** kann ferner ggf. eine assozierte Datenbereitstellungseinheit **704** vorgeben. Alternativ kann die Da-

tenquellenverwaltungseinheit **532** einen Hinweis auf die Datenbereitstellungseinheit **704** oder die Struktur oder das Format geben, die/das durch die Datenquelle **702** verwendet wird. Bei einigen Ausführungsformen kann die Vorgabe der Quelle ferner eine separate Vorgabe der Datenquelle **702** für jeden Parameter oder für jede Spalte der abzurufenden Daten umfassen. Dadurch können Abfragen, welche das standardisierte Abfrageformat verwenden, Daten von mehreren unterschiedlichen Datenquellen **702** abrufen.

[0364] Wie vorstehend erwähnt, werden standardisierte Abfragen **709** in datenquellspezifische Abfragen umgewandelt, die für jede Zieldatenquelle **702** ausgeführt werden können. Eine solche Umwandlung kann durch den Abfrageausführungsdienst **706** oder durch jede Zieldatenbereitstellungseinheit **704a–704d** ausgeführt werden. **Fig. 6D** veranschaulicht ein beispielhaftes Abfrageverfahren **740**, bei dem der Abfrageausführungsdienst **706** die standardisierte Abfrage **709** in eine oder mehrere datenquellspezifische Abfragen umwandelt; und **Fig. 6E** veranschaulicht ein beispielhaftes Abfrageverfahren **760**, bei dem die Datenbereitstellungseinheit **704** die standardisierte Abfrage **709** in eine quellspezifische Abfrage umwandelt.

[0365] Das Verfahren **740** kann mit dem Empfang einer standardisierten Abfrage unter Verwendung eines standardisierten Abfrageformates (Block **742**) beginnen. Die standardisierte Abfrage kann durch den Abfrageblock **708**, den Abfrageausführungsdienst **706** oder eine Datenbereitstellungseinheit **704a–704d** empfangen werden. Die Abfrage kann ein beliebiges der vorstehend beschriebenen Merkmale oder Elemente umfassen. Die Abfrage kann von einem Benutzer über eine direkte Eingabe oder von einem Programmblöck oder einer Programmroutine empfangen werden, der/die standardisierte Abfragen auf der Grundlage von Eingaben des Benutzers generieren und/oder speichern kann. Die standardisierte Abfrage kann in einer Datei oder in einem Format erstellt werden, die/das mit JavaScript-Object-Notation (JSON) konform ist, oder unter Verwendung irgendeiner anderen geeigneten Sprache oder Syntax. Die standardisierte Abfrage kann eine Syntax verwenden, die von einigen Datenquellen **702a–702f** ausgeführt werden kann, um Daten abzurufen (wie beispielsweise SQL), oder die standardisierte Abfrage kann eine Syntax verwenden, die nicht direkt durch die Datenquellen **702a–702f** ausgeführt werden kann, um Daten abzurufen (wie beispielsweise die in **Fig. 6C** veranschaulichte beispielhafte Abfrage). Die standardisierte Abfrage kann die abzurufenden Daten (z. B. Zeitrahmen und Parameter der Daten), die Datenquellen **702**, von denen die Daten abgerufen werden sollen (z. B. Datenquellen **702a–702f** oder Datenbereitstellungseinheiten **704a–704d**), und/oder das Format vorgeben, in dem die Daten bereitzustellen sind. Wird die stan-

dardisierte Abfrage **709** am Abfrageblock **708** empfangen, kann sie anschließend nach dem Auftreten eines auslösenden Ereignisses an den Abfrageausführungsdienst **706** gesendet werden, wie beispielsweise die Auswahl des Benutzers einer Option zum Ausführen der Abfrage oder eine Anfrage nach den Abfragedaten durch ein anderes Objekt, eine andere Routine, einen anderen Block, einen anderen Prozess, einen anderen Dienst oder eine andere Funktion innerhalb des Analysedienstes **500**.

[0366] Im Anschluss an den Empfang der standardisierten Abfrage **709** kann der Abfrageausführungsdienst **706** (oder die Datenbereitstellungseinheit **704**) eine oder mehrere Datenquellen **702** bestimmen, von denen die Daten abgerufen werden sollen (Block **744**). Bei einigen Ausführungsformen kann die Bestimmung das Empfangen oder Erkennen einer Vorgabe hinsichtlich einer oder mehrerer Datenquellen **702** umfassen. Die Vorgabe kann für jede Dateneigenschaft spezifisch sein (z. B. Prozessvariable oder Spalte) oder kann für die gesamte Abfrage allgemein gehalten sein. Eine solche Vorgabe oder solche Vorgaben können als Objekte oder Markierungen in der standardisierten Abfrage **709** enthalten oder durch eine zusätzliche Eigenschaft des Abfrageblocks **708** vorgegeben sein. Beispielsweise kann eine Vorgabe hinsichtlich der Datenquelle **702** für eine Datenspalte oder einen Datentyp dem Objekt oder der Markierung vorangestellt werden, welches/welche die Datenspalte oder den Datentyp vorgibt. Alternativ kann der Abfrageblock **708** oder die Datenbereitstellungseinheit **704** anderweitig mit einer oder mehreren Datenquellen **702** assoziiert sein. Diese Assoziation kann dadurch herbeigeführt werden, dass eine Vorgabe hinsichtlich einer oder mehrerer Datenquellen **702** getrennt von der standardisierten Abfrage **709** empfangen wird. Beispielsweise kann der Abfrageblock **708** über eine Datenquelleneigenschaft mit einer oder mehreren Datenquellen **702** (oder mit einer oder mehreren Datenbereitstellungseinheiten **704**) assoziiert sein, wenn der Abfrageblock **708** durch das Empfangen einer Auswahl der Datenquellen **702** (oder Datenbereitstellungseinheiten **704**) erstellt wird. Gleichermaßen kann eine Datenbereitstellungseinheit **704a–704d** mit einer oder mehreren Datenquellen **702a–702d** assoziiert sein, wenn die Datenbereitstellungseinheit **704a–704d** konfiguriert ist, entweder inhärent oder durch Assoziationen, die während der Konfiguration erstellt wurden.

[0367] Unter Anwendung der Informationen hinsichtlich der einen oder mehreren Datenquellen und der standardisierten Abfrage **709** werden datenquellspezifische Abfragen erzeugt, um Daten von der einen oder den mehreren Datenquellen **702** (Block **746**) abzurufen. Die datenquellspezifischen Abfragen können durch den Abfrageausführungsdienst **706** erstellt und an jede Datenquelle **702a–702f** gesendet werden, die abgefragt werden soll (di-

rekt oder über eine Datenquellenbereitstellungseinheit **704a–704d**), basierend auf den Datenquellen, welche bei Block **744** vorgegeben wurden. Alternativ kann der Abfrageausführungsdienst **706** die standardisierte Abfrage **709**, die von jeder der einen oder mehreren Datenbereitstellungseinheiten **704a–704d**, welche mit den Datenquellen **702a–702d** assoziiert sind, von denen die Daten abgerufen werden sollen, in eine datenquellspezifische Abfrage umgewandelt werden soll, auf der Grundlage der bei Block **744** vorgegebenen Datenquellen erstellen. Beispielsweise kann der Abfrageausführungsdienst **706** einen oder mehrere Arbeitsprozesse **508** dazu veranlassen, die standardisierte Abfrage **709** als Datenbereitstellungseinheiten **704** umzuwandeln. Ungeachtet ob der Abfrageausführungsdienst **706** oder die Datenbereitstellungseinheiten **704** die datenquellspezifische Abfrage aus der standardisierten Abfrage **709** erzeugt, muss die datenquellspezifische Abfrage ein datenquellspezifisches Format verwenden, welches durch eine Datenbereitstellungseinheit **704** oder eine Datenquelle **702** ausgeführt werden kann, um die in der Datenquelle **702** gespeicherten Daten abzurufen. Bei dem datenquellspezifischen Abfrageformat kann es sich um ein beliebiges bekanntes oder hiernach entwickeltes Format oder eine beliebige bekannte oder hiernach entwickelte Syntax zum Abrufen oder anderweitigen Zugreifen auf Daten handeln, die in einer elektronischen Datenquelle gespeichert sind, wie beispielsweise SQL, MongoDB, CouchDB usw.

[0368] Das Erzeugen der datenquellspezifischen Abfrage kann das Erkennen von Abfrageparametern in der standardisierten Abfrage **709** umfassen. Die Abfrageparameter können Parameter umfassen, die mit einem Zeitrahmen oder anderen Eigenschaften der Daten assoziiert sind, wie beispielsweise der durch das Datenfeld „timeSelector“ vorgegebene Zeitrahmen, die durch das Objekt „sampleRateSecs“ vorgegebene Abtastrate und die durch das Datenfeld „columns“ vorgegebenen Datenparameter (insbesondere die „tag“-Objekte) in der vorstehend erörterten beispielhaften standardisierten Abfrage. Diese Abfrageparameter können Hinweise auf Prozessvariablen umfassen, die mit Typen von Messungen, Typen von Messgeräten oder konkreten Messgeräten assoziiert sind (wie beispielsweise die Feldgeräte **15–22** und **40–46**). Ein oder mehrere Zeitrahmen können für jede abzufragende Datenquelle **702** vorgegeben sein, und unterschiedliche Datenquellen oder Daten mit unterschiedlichen Parametern (z. B. unterschiedliche Spalten, unterschiedliche Prozessvariablen usw.) innerhalb einer Datenquelle können unterschiedliche damit assoziierte Zeitrahmen aufweisen. Die bestimmten Abfrageparameter können durch den Abfrageausführungsdienst **706** oder die Datenbereitstellungseinheit **704** aus der standardisierten Abfrage **709** extrahiert und in Aspekte der datenquellspezifischen Abfrage umgewandelt werden. Die erzeug-

te datenquellspezifische Abfrage umfasst dementsprechend eine Abfragesprache, welche die wesentlichen Parameter darstellt, die in der standardisierten Abfrage **709** vorgegeben sind, wobei jedoch solche wesentlichen Parameter in der datenquellspezifischen Abfrage im datenquellspezifischen Abfrageformat enthalten sind.

[0369] sBei Ausführungsformen, bei denen ein oder mehrere Zeitrahmen und Abtastraten vorgegeben wurden, kann die datenquellspezifische Abfrage erzeugt werden, um zu vorgegebenen Zeitpunkten innerhalb des Zeitraums, der durch jeden Zeitrahmen vorgegeben ist, Daten bereitzustellen, die mit einer Vielzahl von Datenpunkten assoziiert sind. Jeder Datenpunkt kann einen Zeitstempel aufweisen, der eine vorgegebene Zeit anzeigt, die mit dem Datenpunkt assoziiert ist, wobei es sich um ein ganzzahliges Vielfaches des Zeitraums der Abtastrate zur oder im Anschluss an die Startzeit handelt, die mit dem Zeitrahmen assoziiert ist, und dies zur oder vor der Endzeit erfolgt, die mit dem Zeitrahmen assoziiert ist. Mehrere Zeitrahmen können durch die standardisierte Abfrage vorgegeben werden, siehe auch **Fig. 6C**. In solchen Fällen kann die datenquellspezifische Abfrage erzeugt werden, um Daten bei Datenpunkten abzurufen, die über einen Zeitstempel verfügen, welcher der Abtastrate oder den Abtastraten innerhalb jedes der mehreren Zeitrahmen entspricht. In einigen solchen Fällen können separate datenquellspezifische Abfragen für einige oder alle der mehreren Zeitrahmen erzeugt werden. Wenn ein oder mehrere Zeitrahmen ohne eine Abtastrate vorgegeben ist/sind, kann die Vielzahl an Datenpunkten mit Zeitstempel abgerufen werden, wobei die Zeitstempel jeden beliebigen Wert innerhalb des Zeitrahmens annehmen können. Dementsprechend zeigt der Zeitstempel Zeitpunkte an, an denen jeder Datenpunkt gemessen wurde (oder an denen die zugrundeliegenden Prozesswerte gemessen wurden).

[0370] Der Abfrageausführungsdienst **706** oder die Datenbereitstellungseinheit **704** können anschließend die Ausführung der einen oder mehreren datenquellspezifischen Abfragen veranlassen, um die bestimmten Datenquellen **702** abzufragen (Block **748**). Dies kann die Übertragung der datenquellspezifischen Abfrage vom Abfrageausführungsdienst **706** an eine oder mehrere Datenbereitstellungseinheiten **707a-704d** und/oder an eine oder mehrere Datenquellen **702a-702f** umfassen. Gleichermassen können die Datenbereitstellungseinheiten **704** bei einigen Ausführungsformen die datenquellspezifische Abfrage an die Datenquellen übertragen, um die angeforderten Daten abzurufen. Bei weiteren Ausführungsformen kann die datenquellspezifische Abfrage durch eine Ausführungsengine der empfangenen Datenquelle **702** ausgeführt werden. Bei anderen Ausführungsformen können die Datenbereitstellungseinheiten **704** eine oder mehrere der

datenquellspezifischen Abfragen dadurch ausführen, dass sie auf die in den Datenquellen **702** gespeicherten Daten zugreifen und diese manipulieren. Die nach der Ausführung der einen oder mehreren datenquellspezifischen Abfragen von der einen oder den mehreren Datenquellen zurückgegebenen Daten können anschließend an eine Datenempfangseinheit gesendet oder von einer Datenempfangseinheit empfangen werden, einschließlich der Datenbereitstellungseinheit **704**, des Abfrageausführungsdienstes **706** oder des Abfrageblocks **708**. Dementsprechend erhält die Datenempfangseinheit die angefragten Daten von der einen oder den mehreren Datenquellen **702** (Block **750**).

[0371] Bei Ausführungsformen, bei denen eine Abtastrate vorgegeben ist, kann das Abrufen der Daten die Sicherstellung umfassen, dass die Daten Datenpunkte mit Zeitstempeln umfassen, die mit jedem der Abtastpunkte übereinstimmen, die durch den Zeitrahmen und die Abtastrate vorgegeben sind. Dies kann für jeden Zeitrahmen und für jede Abtastrate erfolgen. Wie vorstehend erwähnt, entsprechen diese Abtastpunkte ganzzahligen Vielfachen des Zeitraumes der Abtastrate zur oder im Anschluss an die Startzeit des Zeitrahmens und erfolgen diese zur oder vor der Endzeit des Zeitrahmens. Wenn die in der Datenquelle **702** gespeicherten Datenpunkte keinen Datenpunkt für einen oder mehrere der mit den Abtastpunkten assoziierten Zeitpunkte umfassen, können zusätzliche Datenpunkte mit Zeitstempeln hinzugefügt werden, die den Abtastpunkten entsprechen. Den zusätzlichen Datenpunkten können Werte zugeordnet werden, die mit Werten der Dateneinträge in der Datenquelle **702** assoziiert sind, die zeitlich am nächsten vor dem Zeitstempel liegen. Beispielsweise kann eine Datenquelle **702** Datenpunkte für die Zeitpunkte 7:01:5500 (mit dem Wert V1), 7:02:0500 (mit dem Wert V2) und 7:02:5500 (mit dem Wert V3) umfassen, aber der in der standardisierten Abfrage vorgegebene Zeitrahmen kann 7:02:0000 (Startzeit) bis 7:03:0000 (Endzeit) lauten, und der Zeitraum der Abtastrate kann 0:00:5000 (entsprechend einer Abtastrate von einer Probe pro halber Sekunde) sein. In einem solchen Beispiel weisen die abgerufenen Datenpunkte Zeitstempel von 7:02:0000, 7:02:5000 bzw. 7:03:0000 auf, welche die Werte V1, V2 bzw. V3 aufweisen. Dementsprechend wird der aktuellste vorherige Wert V2 (gemessen bei 7:02:0500) für den Datenpunkt mit dem Zeitstempel 7:02:5000 verwendet, selbst wenn ein anderer Wert (V3 zum Zeitpunkt 7:02:5500) zeitlich näher dran ist, aber nach dem Zeitstempel liegt. Zusätzlich können überschüssige Datenpunkte, die zu Zeitpunkten zwischen den gewünschten Abtastzeiten auftreten, aus den abgerufenen Daten entfernt oder gelöscht werden. Dementsprechend hat jeder Datenpunkt in den abgerufenen Daten (im Anschluss an solche Anpassungen oder Harmonisierungen) einen Zeitstempel, der mit einem ganzzahligen Vielfachen des Zeitraums der Abtastrate zur oder im An-

schluss an die Startzeit innerhalb jedes Zeitrahmens assoziiert ist, der durch die standardisierte Abfrage vorgegeben wird. Andere Mittel zum Harmonisieren von Datenpunkten innerhalb der abgerufenen Daten können gleichermaßen Anwendung finden.

[0372] Werden in ein und derselben standardisierten Abfrage mehrere Datenquellen **702** vorgegeben, kann jede der Datenquellen **702** Datenpunkte mit unterschiedlichen Zeitstempeln aufweisen und können die mit solchen Zeitstempeln assoziierten Messungen bei anderen Abtastraten erfolgt sein. Beispielsweise können eine erste Datenquelle **702a** und eine zweite Datenquelle **702d** in der standardisierten Abfrage vorgegeben sein. Bei der Datenquelle **702a** kann es sich um eine erste Datenquelle handeln, die Datenpunkte mit einer ersten Abtastrate speichert (z. B. eine Messung pro Sekunde), und bei der Datenquelle **702d** kann es sich um eine zweite Datenquelle handeln, die Datenpunkte mit einer zweiten Abtastrate speichert (z. B. vier Messungen pro Sekunde). Die erhaltenen Daten können entsprechend der in der standardisierten Abfrage vorgegebenen Parameter harmonisiert werden, indem Datenpunkte zu Zeitpunkten hinzugefügt oder entfernt werden, bei denen es sich um ganzzahlige Vielfache des Zeitraumes der Abtastrate handelt, die durch die standardisierte Abfrage vorgegeben wurde, siehe vorstehende Erörterung.

[0373] Wenn die Datenempfangseinheit die Daten von den einen oder den mehreren Datenquellen **702** abruft, kann die Datenempfangseinheit ferner die Daten entsprechend Formatierungsparametern oder Vorgaben formatieren, die mit der standardisierten Abfrage **709** (Block **752**) assoziiert sind. Solche Formatierungsparameter oder Vorgaben können in der standardisierten Abfrage **709** enthalten sein, mit dem standardisierten Abfrageformat assoziiert sein oder können in einer separaten Eigenschaft des Abfrageblocks **708** vorgegeben sein. Bei einigen Ausführungsformen kann dies ferner das Erzeugen eines Datenrahmens aus den abgerufenen Daten dadurch umfassen, dass das Format der abgerufenen Daten so angepasst wird, dass es dem gewünschten Format entspricht. Wenn eine Vielzahl von Datenquellen **702** abgefragt wurde, kann das Erzeugen des Datenrahmens ferner das Kombinieren der von jeder der Vielzahl von Datenquellen **702** abgerufenen Daten umfassen, um einen Gesamtdatenrahmen zu erzeugen. Wenn beispielsweise ein erster Datensatz von einer ersten Datenquelle **702a** und ein zweiter Datensatz von einer zweiten Datenquelle **702d** abgerufen werden, kann ein Gesamtdatenrahmen erzeugt werden, in dem der erste und der zweite Datensatz miteinander kombiniert werden.

[0374] Das Verfahren **760** kann mit dem Empfang einer standardisierten Abfrage **709** von einer Datenabfrageeinheit (Block **762**) beginnen. Eine Datenabfra-

geeinheit, wie beispielsweise der Abfrageblock **708** oder der Abfrageausführungsdienst **706**, kann Daten von einer oder mehreren Datenquellen **702** dadurch anfordern, dass eine standardisierte Abfrage **709** an eine Datenbereitstellungseinheit **704** gesendet wird. Die Datenbereitstellungseinheit **704** kann mit einer oder mehreren Datenquellen **702** assoziiert oder so konfiguriert sein, dass sie Daten von verschiedenen Datenquellen **702** über eine Netzwerkverbindung abruft. Alternativ kann die Datenabfrageeinheit die standardisierte Abfrage **709** von einem anderen Datenblock, einer anderen Softwareroutine, einem anderen Prozess oder einem anderen Dienst im Datenanalysesystem **100** empfangen. Zusätzlich kann eine Datenempfangseinheit (bei der es sich gleichermaßen um den Abfrageblock **708**, den Abfrageausführungsdienst **706**, einen Arbeitsprozess **508**, Datendienste **520**, die Anwendung **514**, einen Programmspeicher, oder einen beliebigen anderen Dienst, eine beliebige andere Routine, einen beliebigen anderen Prozess oder eine beliebige andere Vorrichtung handeln kann, die/der die mit der Prozessanlage **5** assoziierten Daten weiter analysieren, darstellen oder speichern kann) durch die standardisierte Abfrage **709** oder anderweitig vorgegeben werden. Bei einer solchen Datenempfangseinheit kann es sich um die selbe Einheit handeln wie die Datenabfrageeinheit in einigen Fällen, oder die Datenabfrageeinheit kann separat von der Datenempfangseinheit vorliegen. Zum Zwecke der Übersichtlichkeit erfolgt die folgende Erörterung des beispielhaften Verfahrens **760** unter der Annahme, dass die Datenbereitstellungseinheit **704a** eine standardisierte Abfrage **709** vom Abfrageausführungsdienst **706** als Datenabfrageeinheit und Datenempfangseinheit empfängt, wobei die standardisierte Abfrage **709** Daten von der Datenquelle **702a** abfragt. Dies erfolgt zur besseren Veranschaulichung der Hauptmerkmale des Verfahrens und soll den Geltungsbereich der Offenlegung nicht einschränken. Der Fachmann wird verstehen, dass zahlreiche alternative Konfigurationen mit lediglich geringfügigen und gewöhnlichen Anpassungen des hierin erörterten Verfahrens einfach hergestellt werden können.

[0375] Wenn die standardisierte Abfrage an der Datenbereitstellungseinheit **704a** empfangen wird, extrahiert die Datenbereitstellungseinheit **704a** Abfrageparameter aus der standardisierten Abfrage **709** (Block **764**). Die Abfrageparameter können mit der Zeit (z. B. ein Zeitrahmen, in dem die Daten abgerufen werden, eine Abtastrate usw.), mit Datentypen oder Dateneigenschaften (z. B. Prozessvariablen, Spalten in einer Tabelle, Messungen, berechnete Werte aus Messungen usw.) oder mit der Datenquelle **702a**, von der die Daten abgerufen werden sollen (z. B. eine Vorgabe hinsichtlich einer Datenbank, eines Pfades zu dieser oder einer Tabelle in dieser), assoziierte Parameter umfassen. Das Extrahieren der Abfrageparameter kann das Bestimmen

eines oder mehrerer Parameter auf der Grundlage von Objekten, Datenfeldern oder Elementen in der standardisierten Abfrage **709** umfassen. Bei einigen Ausführungsformen kann die Datenbereitstellungseinheit **704a** ferner Parameter extrahieren, die vorgeben, wie die Daten an den Abfrageausführungsdienst **706** zurückgegeben werden sollen (d.h. Formatierung, Struktur, Timing oder Protokoll, das zu verwenden ist, wenn die angefragten Daten der Datenabfrageeinheit zur Verfügung gestellt werden). Die Datenbereitstellungseinheit **704a** kann die extrahierten Abfrageparameter in einem flüchtigen oder nichtflüchtigen Speicher abspeichern, um diese beim Erzeugen einer oder mehrerer datenquellspezifischer Abfragen und/oder dem Formatieren der abgerufenen Daten, die der Datenabfrageeinheit zur Verfügung gestellt werden sollen, zu verwenden.

[0376] Die Datenbereitstellungseinheit **704a** kann anschließend eine datenquellspezifische Abfrage auf der Grundlage der extrahierten Abfrageparameter (Block **766**) erzeugen, wobei die datenquellspezifische Abfrage ein datenquellspezifisches Abfrageformat verwendet, welches mit der Datenquelle **702a** assoziiert ist. Beispielsweise kann es sich bei der Datenquelle **702a** um eine nicht-relationale Datenbank handeln, die eine MongoDB-Datenstruktur oder ein MongoDB-Datenformat verwendet, wobei die Datenbereitstellungseinheit **704a** in diesem Fall eine datenquellspezifische Abfrage unter Verwendung der Abfragesyntax von MongoDB erzeugt, um die durch die Abfrageparameter vorgegebenen Daten abzurufen, die aus der standardisierten Abfrage **709** extrahiert wurden. In einem alternativen Beispiel kann es sich bei der Datenquelle **702a** um eine relationale Datenbank handeln, die MySQL verwendet, wobei die Datenbereitstellungseinheit **704a** in diesem Fall eine datenquellspezifische Abfrage unter Verwendung einer SQL-Abfragesyntax erzeugt, um die durch die Abfrageparameter vorgegebenen Daten abzurufen, die aus der standardisierten Abfrage **709** extrahiert wurden. Zum Erzeugen der datenquellspezifischen Abfrage kann die Datenbereitstellungseinheit **704a** eine Zuordnung zwischen dem standardisierten Abfrageformat und dem datenquellspezifischen Abfrageformat anwenden. Eine solche Zuordnung kann Anpassungen hinsichtlich der Syntax oder des Formates umfassen, durch die die in der standardisierten Abfrage **709** ausgedrückten Abfrageparameter in im Wesentlichen äquivalente Parameter im datenquellspezifischen Abfrageformat umgewandelt werden. Bei einigen Ausführungsformen können das Extrahieren der Abfrageparameter und das Erzeugen der datenquellspezifischen Abfrage miteinander kombiniert werden, so dass die Datenbereitstellungseinheit **704a** die durch die standardisierte Abfrage vorgegebenen Abfrageparameter direkt Parametern oder Elementen der datenquellspezifischen Abfrage zuordnet. Wie vorstehend erörtert, kann die datenquellspezifischen Abfrage

so erzeugt werden, dass sie Datenpunkte mit Zeitsstempeln zurückgibt, die ganzzahligen Vielfachen des Zeitraums der Abtastrate zur oder im Anschluss an die Startzeit innerhalb jedes Zeitrahmens entsprechen, der durch die standardisierte Abfrage **709** vorgegeben ist.

[0377] Sobald die datenquellspezifische Abfrage erzeugt wurde, kann die Datenbereitstellungseinheit **704a** die datenquellspezifische Abfrage ausführen, um die angefragten Daten von der Datenquelle **702a** abzurufen (Block **768**). Bei einigen Ausführungsformen kann die Datenbereitstellungseinheit **704a** eine Anfrage hinsichtlich der Ausführung der datenquellspezifischen Abfrage an die Datenquelle **702a** senden und kann die Datenquelle **702a** die datenquellspezifische Abfrage ausführen und die entsprechenden Daten an die Datenbereitstellungseinheit **704a** zurückgeben. Alternativ kann die Datenbereitstellungseinheit **704a** als eine Schnittstelle oder Ausführungsengine für die Datenquelle **702a** fungieren, wobei die Datenbereitstellungseinheit **704a** in diesem Fall die datenquellspezifische Abfrage entsprechend der Regeln des datenquellspezifischen Abfrageformates ausführen kann, um auf die in der Datenquelle **702a** gespeicherten Daten zuzugreifen, diese zu analysieren und diese auszuwählen. Unabhängig davon, ob die datenquellspezifische Abfrage durch die Datenbereitstellungseinheit **704a** oder die Datenquelle **702a** ausgeführt wird, werden die durch die Ausführung der datenquellspezifischen Abfrage erhaltenen Daten durch die Datenbereitstellungseinheit **704a** abgerufen.

[0378] Bei einigen Ausführungsformen kann die Datenbereitstellungseinheit **704a** die durch die Ausführung der Abfrage abgerufenen Daten auf der Grundlage der standardisierten Abfrage **709** formatieren (Block **770**). Die Datenbereitstellungseinheit **704a** kann die abgerufenen Daten dementsprechend so verarbeiten, dass die abgerufenen Daten so formatiert sind, dass sie den Anforderungen an die Daten entsprechen, die in der standardisierten Abfrage **709** enthalten sind. Dies kann das Erzeugen eines Datenrahmens für die abgerufenen Daten umfassen, der die abgerufenen Daten in einem Format enthält, welches durch den Abfrageausführungsdienst **706** einfach verwendet werden kann, einschließlich der Anwendung von Datenformatregeln und alternativen Datenbezeichnungen auf die abgerufenen Daten. Beispielsweise kann die standardisierte Abfrage **709** vorgeben, dass die Daten unter Verwendung einer JSON-Datei in einem Dokument bereitgestellt werden. Gleichermassen kann die standardisierte Abfrage vorgeben, dass die Daten in einem tabellarenen Format an die Datenabfrageeinheit zurückgegeben werden oder dass die Datenwerte als Fließkommaziffern mit drei Nachkommastellen zu formatieren sind. Wenn gleich diese Formatierungsanforderungen in der standardisierten Abfrage **709** enthal-

ten sein können, können einige oder alle der Formatierungsanforderungen durch das standardisierte Abfrageformat (oder eine Version davon) vorgegeben werden. So werden die standardisierten Abfragen, die das standardisierte Abfrageformat (oder eine Version davon) verwenden, die Daten immer in einem konsistenten Format zurückgeben. Wie vorstehend erörtert, kann das Formatieren der abgerufenen Daten das Anpassen oder Harmonisieren der Zeitpunkte der Datenpunkte umfassen, so dass diese Zeitstempel aufweisen, die ganzzahligen Vielfachen des Zeitraums der Abtastrate zur oder im Anschluss an die Startzeit in jedem Zeitrahmen entsprechen, der durch die standardisierte Abfrage **709** vorgegeben ist. Dies kann ferner das Hinzufügen von Datenpunkten mit entsprechenden Zeitstempeln oder das Entfernen überschüssiger Datenpunkte umfassen, die an Zeitpunkten auftreten, die zwischen den gewünschten Abtastzeiten liegen.

[0379] Sobald die abgerufenen Daten formatiert wurden, liefert die Datenbereitstellungseinheit **704a** die formatierten Daten an die Datenempfangseinheit (Block **772**). Die Datenbereitstellungseinheit **704a** kann zur weiteren Analyse einen Datenrahmen im geforderten Format oder in einem beliebigen bekannten Format an den Abfrageausführungsdienst **706** übertragen. Bei weiteren Ausführungsformen kann die gesamte oder ein Teil der Formatierung der abgerufenen Daten anstelle dessen durch den Abfrageausführungsdienst **706** durchgeführt werden, nachdem die Datenbereitstellungseinheit **704a** die unformatierten oder teilweise formatierten Daten, die durch die Ausführung der datenquellspezifischen Abfrage erhalten wurden, an den Abfrageausführungsdienst **706** gesendet hat. Wenn die Daten von der Datenabfrageeinheit empfangen und formatiert wurden, können die formatierten Daten im Rahmen der Prozessanlagensteuerung oder Analyse verwendet werden, wie anderswo hierin erörtert. Bei einigen Ausführungsformen kann der Abfrageausführungsdienst **706** den Datenrahmen ferner einer anderen Datenempfangseinheit zur Verfügung stellen, wie beispielsweise dem Abfrageblock **708** oder einem Arbeitsprozess **508**.

[0380] Im Allgemeinen können alle Teile der Datenquellen **702a-f**, der Datenquellenbereitstellungseinheiten **704a-d** und des Abfrageblocks **708**, sowie die Abfragesprache, die verwendet wird, um die Daten von den unterschiedlich formatierten Datenquellen **702a-f** zu erhalten, in Verbindung mit allen Teilen der **Fig. 1**, **Fig. 2**, **Fig. 3**, **Fig. 4A–Fig. 4Q** und/or **Fig. 5A–Fig. 5G** und/oder mit einer beliebigen Anzahl an Merkmalen und/oder Techniken arbeiten, die in den Abschnitten der vorliegenden Offenlegung beschrieben werden, die den **Fig. 1**, **Fig. 2**, **Fig. 3**, **Fig. 4A–Fig. 4Q** und/oder **Fig. 5A–Fig. 5G** entsprechen.

FREQUENZANALYSETECHNIK FÜR EINE FRÜHZEITIGE FEHLERERKENNUNG

[0381] Eine neue Datenanalysetechnik oder -funktion (z. B. die, die vom verteilten Industrieprozessleistungsüberwachungs-/analysesystem oder DAS **100** bereitgestellt werden kann) ist eine Frequenzanalysetechnik oder -funktion zur frühzeitigen Fehlererkennung in Prozesssteuerungssystemen oder Prozessanlagen, wie beispielsweise die Prozessanlage **5**. Die neue Frequenzanalysetechnik oder -funktion wird hierin als eine „rollende schnelle Fourier-Transformation“ oder „rollende FFT“ bezeichnet und kann in Verbindung mit jedem der Systeme, jeder der Architekturen, jedem der Verfahren und jeder der Techniken verwendet werden, die hierin beschrieben werden. Beispielsweise kann eine Datenblockdefinition für die rollende FFT in der Datenblockdefinitionsbibliothek **252** gespeichert und Benutzern zugänglich gemacht werden, um in Datenmodulen verwendet zu werden, die Offline und/oder Online ausgeführt werden. Zusätzlich oder alternativ können eine oder mehrere verteilte Datenengines **102x** eine rollende FFT an Daten ausführen, die an dem einen oder den mehreren DDE **102x** abgerufen wurden. Eine rollende FFT ist jedoch nicht darauf beschränkt, nur in Verbindung mit den hierin beschriebenen Systemen, Architekturen, Verfahren und Techniken verwendet zu werden, und kann in Verbindung mit einem beliebigen System, einer beliebigen Architektur, einem beliebigen Verfahren und/oder einer beliebigen Technik verwendet werden, das/die Zeitreihendaten erzeugt. Wie unten veranschaulicht, handelt es sich bei der Analysetechnik oder -funktion der rollenden FFT um eine Kombination aus beschreibenden und vorhersagenden Analysen.

[0382] Grundsätzlich können Auffälligkeiten, Fehler, Leistungsabfälle und/oder andere unerwünschte oder ungewollte Zustände in einem Prozessleitsystem oder einer Anlage verhindert werden (oder deren Auswirkungen minimiert werden), wenn Prozessdaten mit Leitindikatoren zum zukünftigen Verhalten der verfahrenstechnischen Anlage erkannt werden können, vorzugsweise in einem Zeitrahmen, der vorbeugende oder abschwächende Maßnahmen zulässt. Zu diesen Prozessdaten gehören unter anderem Messdaten, beispielsweise des Drucks, der Temperaturen und Fließraten von Stoffen, die durch die Anlage fließen, sowie vergleichbare Informationen zu Ausrüstungsgegenständen. Zu diesen Prozessdaten können des Weiteren die chemische Zusammensetzung von Massenströmen und der An-/Aus-Zustand von Geräten gehören. Im Allgemeinen kann eine rollende FFT auf alle Online- und Offline-Zeitreihendaten angewendet werden, die als Ergebnis der Steuerung eines Prozesses innerhalb einer verfahrenstechnischen Anlage generiert und auf geeignete Weise erfasst werden, beispielsweise durch Abtastung, Empfang eines Datenstroms, Auslesen einer Datenbank,

Datendatei, Datenquelle (z.B. Seeq) oder Historians, Datenabruf usw.

[0383] Ein Verfahren zur Bestimmung von Leitindikatoren in einer verfahrenstechnischen Anlage beinhaltet die Verhaltensanalyse der Prozessdaten über einen bestimmten Zeitraum. Das Verhalten kann sich aus verschiedenen Gründen ändern, in einigen Fällen kann die Änderung mit Prozessstörungen im Zusammenhang stehen, die zu Auffälligkeiten, Fehlern, Leistungsabfällen und/oder anderen Zuständen führen können und die als solche als Leitindikatoren für den entsprechenden Zustand betrachtet werden können. Das in diesem Dokument beschriebene rollende FFT-Verfahren wandelt einen ersten Satz von Zeitreihendaten entsprechend den Leitindikatoren in einen Frequenzbereich um und generiert in der Folge einen zweiten Satz an Zeitreihendaten basierend auf den Frequenzbereichsdaten, die dann überwacht und dazu genutzt werden können, das Auftreten der Auffälligkeit, des Fehlers, des Leistungsabfalls und/oder den Zuständen der verfahrenstechnischen Anlage zu prognostizieren.

[0384] Üblicherweise können Prozessdaten von einer verfahrenstechnischen Anlage erhoben, erhalten oder anderweitig erfasst werden, um sie als Eingabe in einem Frequenzanalyseverfahren zu verwenden. Dazu können alle Daten im Zusammenhang mit den vorhandenen Prozesssignalen in der verfahrenstechnischen Anlage verwendet werden, beispielsweise Temperatur, Ströme, Druck, Zusammensetzungen und/oder andere kontinuierliche Signale, die als Ergebnis des Betriebs der verfahrenstechnischen Anlage zur Steuerung eines Prozesses generiert werden. Üblicherweise wird eine FFT an den erfassten Prozessdaten durchgeführt, um Amplituden von wichtigen, darin enthaltenen Frequenzen durch Verwendung eines festen Fensters (z.B. spezifische Anzahl von Datenpunkten) zu identifizieren, üblicherweise basierend auf einer Zweierpotenz (z.B. $2^{10} = 1024$). Moderne Berechnungsverfahren ermöglichen eine nutzerdefinierte Länge des Datenfensters, die Länge wird jedoch häufig durch die verfügbare Computerspeicherkapazität begrenzt. Die Anzahl von Abtastungen und die Abtastrate in einer FFT müssen des Weiteren das Nyquist-Kriterium von mindestens zwei Abtastungen pro schnellster, relevanter Frequenz erfüllen. Des Weiteren sollte die konventionelle FFT an mehreren Zyklen des periodischen Verhaltens des gewünschten Datensignals agieren.

[0385] Bei vielen konventionellen Anwendungen der FFT, die auf Prozessdaten angewendet werden, wird jedoch davon ausgegangen, dass sich das relevante Signal nicht im Laufe der Zeit ändert. Die rollende FFT wird von dieser Annahme jedoch nicht eingeschränkt. Die "rollende" FFT bietet den Vorteil, die Änderungen des Signals über einen bestimmten Zeitraum zu erfassen, um zu bestimmen, wann diese

Änderungen am Signal im entsprechenden Zeitraum auftreten. Die rollende FFT beinhaltet im Speziellen die Anwendung einer FFT auf ein Fenster von Daten, die durch ein relevantes Signal oder eine relevante Variable (beispielsweise eine Messung, ein erfasster Wert oder ein anderes als Ergebnis der verfahrenstechnischen Anlage generiertes Signal) generiert wurden, und zeichnet die Frequenzamplituden (z.B. Spitzenamplituden) für dieses Fenster auf. Das Fenster wird dann zeitlich um eine Abtastung nach vorn verschoben, dann wird die FFT erneut durchgeführt und die Ergebnisse aufgezeichnet oder gespeichert. Dieser Vorgang wird bis zum Ende der Zeitreihendaten wiederholt. Da die FFT für jede Abtastzeit im Datensatz durchgeführt wird (in einigen Fällen mit Ausnahme der ersten n-1 Abtastungen, wobei n für die Anzahl von Abtastungen im Fenster steht), werden ein oder mehrere Zeitreihendatensätze, inklusive der Amplituden (z.B. Spitzenamplituden), von einer oder mehreren relevanten Frequenzen erstellt oder generiert. Jede relevante Frequenz kann auf eine entsprechende neue Prozessvariable der verfahrenstechnischen Anlage reagieren, die die entsprechenden Zeitreihendaten entsprechend der Amplituden (z.B. Spitzenamplituden) der relevanten Frequenz generieren. Die von jeder neuen Prozessvariablen generierten Zeitreihendaten können gespeichert, überwacht und/oder analysiert werden, um mögliche Auffälligkeiten, Fehler oder andere Zustände der verfahrenstechnischen Anlage zu prognostizieren.

[0386] Daher kann für ein relevantes Signals oder eine relevante Variable eine oder mehrere neue Prozessvariablen entsprechend der prognostizierten, unerwünschten Zustände der verfahrenstechnischen Anlage erstellt, generiert und innerhalb der verfahrenstechnischen Anlage verwendet werden. Für ein relevantes Signal oder eine relevante Variable kommen individuelle Signale oder Messpunkte und die Kombination (z.B. eine Summe oder andere Kombination) der Signale oder Messwerte der individuellen Signale/Punkte in Frage. Eine relevante Variable kann von Nutzer definiert werden oder automatisch von einem Datenblock, Datenmodul und/oder einer Datenanalysefunktion festgelegt werden.

[0387] Wie oben bereits aufgeführt beinhaltet der Prozess zur Festlegung eines Satzes neuer Zeitreihendatensätze die Umwandlung des relevanten Signals oder der relevanten Variablen von einem Zeitbereich in einen Frequenzbereich, die Erfassung der Frequenzbereichsdaten und die Umwandlung der erfassten Frequenzbereichsdaten zur Bestimmung der Zeitreihendaten entsprechend der neuen Prozessvariablen. Die Umwandlung oder Rückwandlung in den Zeitbereich spielt dabei eine wichtige Rolle, da es die Kontrolle der neuen Zeitreihendatensätze gemeinsam mit dem ursprünglichen Prozessdatensignal und/oder anderen Zeitreihenprozessdaten ermöglicht. Die neuen Zeitreihendaten können dann

durch Verwendung der gleichen Analyseverfahren (z.B. statistische Analyse, Hauptkomponentenanalyse, Standardabweichungen usw.) ausgewertet werden, die für die Nutzung an den durch die ursprünglichen relevanten Signale oder Variablen und/oder Prozessdaten zur Verfügung stehen. Darüber hinaus ist in einigen Fällen die Analyse gemeinsam mit den ursprünglichen Prozessdaten und/oder anderen Prozessdaten möglich. Sowohl die neuen Zeitreihendaten als auch die ursprünglichen/anderen Prozessdaten können beispielsweise als Eingabe einem Transformers-Datenblock **521** zur Verfügung gestellt werden.

[0388] Als Beispiel für ein rollendes FFT-Analyseverfahren und seine Vorteile dient ein Beispieldaten, in dem die rollende FFT auf das Problem der Erkennung einer potentiellen Abfackelung in einer verfahrenstechnischen Anlage wie einer Raffinerie oder einem Chemiewerk angewendet wird. Im Allgemeinen verfügen einige verfahrenstechnische Anlagen über eine Fackelanlage, in der die überschüssigen Dämpfe aus den einzelnen Verarbeitungsschritten gesammelt und die gesammelten, überschüssigen Dämpfe verbrannt werden, um giftige Chemikalien vor der Einleitung ins Freie zu entfernen. Das Verbrennen der überschüssigen Dämpfe oder Gase wird allgemeine als "Abfackeln" oder "Abfackelung" bezeichnet. In einigen Fällen wird das Fackelgas nicht verbrannt, sondern als Rohstoff oder Heizgas komprimiert. Ist die Kapazität des Fackelgasverdichters jedoch überschritten, findet eine Abfackelung statt, indem das überschüssige Gas durch einen Brenner geführt wird. Typischerweise wird die Anzahl und/oder Frequenz der Abfackelungen durch Umweltschutzbestimmungen und/oder andere Vorgaben reguliert. Während einige geplante Abfackelungen zulässig und notwendig sind, können die Betreiber verfahrenstechnischer Anlagen oder Unternehmen mit Strafen belegt werden, wenn ungeplante Abfackelungen zu häufig auftreten. Daher ist es für den Betreiber oder Bediener einer verfahrenstechnischen Anlage wünschenswert zu prognostizieren, dass eine ungeplante Abfackelung auftritt oder wahrscheinlich auftreten wird, basierend auf den aktuellen Betriebsbedingungen, wobei die Prognose mit genügten Vorlaufzeit stattfinden sollte, damit der Bediener Maßnahmen ergreifen kann, um die ungeplante Abfackelung zu verhindern.

[0389] Dieses Problem zu lösen ist schwierig, da moderne Raffinerien und Chemiewerke komplexe Anlagen mit vielen verknüpften Systemen sind (z.B. verbundene Systeme im zwei- oder sogar dreistelligen Bereich), wobei jedes System an sich schon als große verfahrenstechnische Anlage betrachtet werden kann. Die Systeme sind in der Regel an eine normale Fackelanlage angeschlossen. Da jedes dieser Systeme eine potentielle Quelle für die durch die Abfackelanlage zu behandelnden Dämpfe sein kann, ist

es schwierig zu überwachen, welches System oder welche Systeme sich dem Zustand des Abfackelns nähern. Darüber hinaus ist es nach einer Abfackelung nicht direkt ersichtlich, welches System dafür verantwortlich ist.

[0390] Das in diesem Dokument beschriebene, rollende FFT-Verfahren kann genutzt werden, um dieser Situation Herr zu werden. **Abb. 7A** zeigt beispielhaft, durch eine Raffinerie generierte Prozessdaten, auf die die rollende FFT angewendet wurde. Die beispielhaften Prozessdaten beinhalten eine Prozessströmungsmessung oder ein -signal **1100**, die von einem System in der Abfackelanlage der Raffinerie während eines bestimmten Zeitintervalls erfasst wurden. Wie dargestellt ist das Prozessströmungssignal **1100** periodisch, wobei der Zeitabschnitt circa einen Tag umfasst (z.B. kann die Periodizität dem täglichen Heiz- und Kühlzyklus entsprechen). Zusätzlich beinhalten die beispielhaften Prozessdaten ein weiteres Signal **1102**, welches den Abfackeldruck der Fackelanlage im gleichen Zeitrahmen wiedergibt. Beim Abfackeldruck handelt es sich um eine Messung der gesamten Dämpfe in der Fackelanlage und diese Messung kann beispielsweise von einem Sensor eines Verdichters oder anderen Gefäßes, in dem sich überschüssige Gase oder Dämpfe befinden, erfasst werden. Bitte beachten Sie, dass die Signaldaten **1100** und **1102** in **Abb. 7A** zeitlich abgestimmt und so eingeteilt wurden, dass ihr Verhalten und ihre Beziehung über den Zeitrahmen leicht zu erkennen sind. In einer Ausführungsform erhalten ein oder mehrere Datenblöcke, Datenmodule und/oder ein oder mehrere DDE **102x** die Signaldaten **1100** und **1102** und führen die zeitliche Abstimmung der beiden Signale **1100** und **1102** durch, so dass die Signale **1100** und **1102** in der zeitlich angepassten Form wie in **Abb. 7A** dargestellt werden können.

[0391] **Abb. 7B** zeigt die gleichen Signale **1100** und **1102** für einen anderen Zeitintervall, während dessen eine Abfackelung **1104** in Signal **1102** aufgetreten ist (z.B. ist der Abfackeldruck entsprechend des Signals **1102** angestiegen, hat den Grenzdruck überschritten und so die Abfackelung **1104** ausgelöst). Aus **Abb. 7B** ist ersichtlich, dass sich das Verhalten des Prozessströmungssignals **1100** vor der Abfackelung **1104** verändert. Zum Beispiel ändert sich das periodische Verhalten des Prozessströmungssignals **1100** zwei bis drei Tage vor dem Eintreten der Abfackelung **1104**. Die Identifizierung der Änderung des Prozessströmungssignals **1100** als Leitindikator für die Abfackelung **1104** kann beispielsweise durch Verwendung von Datenanalyseverfahren wie Hauptkomponentenanalyse, Kreuzkorrelation, PLS-Pfadanalyse usw. bestimmt werden. Die Änderung/der Leitindikator kann erfasst, identifiziert oder festgelegt werden, ein neues Signal oder eine neue Prozessvariable entsprechend der identifizierten Änderung/des identifizierten Leitindikators kann generiert,

festgelegt oder erstellt werden und der Zeitreihenoutput der neuen Prozessvariablen kann durch Verwendung einer rollenden FFT bestimmt werden. Entsprechend kann durch Überwachung der Zeitreihendaten der neuen, durch die rollende FFT bestimmten Prozessvariablen auf die Präsenz eines Leitindikators eine bevorstehende Abfackelung bestimmt werden und eine entsprechende Warnung kann an den Nutzer (z.B. Ingenieur, Bediener usw.) ausgegeben werden, so dass vorbeugende oder abschwächende Maßnahmen ergriffen werden können. In einer Ausführungsform kann die neu festgelegte Prozessvariable in das Prozessleitsystem der Raffinerie integriert werden (beispielsweise durch Zuordnung eines Tags und/oder auf andere Weise, mit der die Prozessleitdatenbank die neue Prozessvariable erkennt) und mittels traditioneller Prozessüberwachungssysteme und -verfahren überwacht werden. Auf diese Weise können der Status und weitere Informationen entsprechend der neuen Prozessleitvariablen wie jede andere Prozessinformation mit Grenzwerten, Alarmmeldungen, Trendkurven usw. dem Nutzer zur Verfügung gestellt werden.

[0392] In einer Ausführungsform ist das Prozessströmungssignal, das zur Prognose der Abfackelung eingesetzt wird, eine Kombination (z.B. Summe, gewichtetes Mittel oder andere geeignete Kombination) aus den einzelnen Prozessströmen oder Messungen. **Abb. 7C** zeigt eine Ausführungsform, bei der ein Prozessströmungssignal, auf das die rollende FFT-Analyse angewendet werden kann, Änderungen an der Summe der einzelnen Prozessablaufsignale **1106** entspricht. Das summierte Signal **1106** kann beispielsweise durch Addition der Messungen der einzelnen Strömungssignale erstellt werden. Durch Generieren der Summe **1106** oder einer anderen, gewünschten Kombination der Signale wird die Anzahl neuer zu erstellender und zu analysierender Signale reduziert, wodurch die Gesamtanalyse vereinfacht werden kann. Allerdings gibt es keine Beschränkung, nur ein einzelnes Prozesssignal oder eine Kombination der Signale zu verwenden. **Abb. 7C** zeigt zeitgleich das Abfackeldrucksignal **1102** inklusive der Abfackelungen **1108** und **1110** während des dargestellten Zeitintervalls. Bitte beachten sie, dass die Signale **1106** und **1102** zeitlich aufeinander abgestimmt sind, so dass die Zusammenhänge leicht zu erkennen sind. Eine Änderung im Verhalten der Summe der Prozessströme **1106** genau vor Auftreten der Abfackelung **1108** ist sichtbar, für die Abfackelung **1110** tritt die Änderung in der Summe der Prozessströme **1106** jedoch erst nach der Abfackelung **1110** auf. Zusätzliche Analysen zur Beziehung zwischen den Strömen **1102** und **1106** können notwendig sein, um zu bestimmen, ob (und wenn ja, wie) das Verhalten der summierten Prozessabläufe **11006** ..., beispielsweise über einen längeren Zeitintervall durch Durchführung zusätzlicher Analysefunktionen usw.

[0393] **Abb. 7D** zeigt die Ergebnisse **1112** der Anwendung des rollenden FFT-Verfahrens auf das Signal **1106**, wobei die Summe der Signale **1106** vom ursprünglichen Zeitbereich in den Frequenzbereich und zurück in den Zeitbereich überführt wurde. Insbesondere das Signal **1112** in **Abb. 7D** entspricht den Amplituden (z.B. Zeitreihendaten) einer bestimmten relevanten Frequenz (die vierte Frequenz entspricht beispielsweise den vier Zyklen pro Tag) des Signals **1106**. Die vierte Frequenz entsprechend der vier Zyklen pro Tag wurde als neue Prozessvariable festgelegt, deren Zeitreihen-Spitzenamplitudenwerte von den entsprechenden Frequenzbereichsdaten erfasst und im Zeitbereich als Signal **1112** in Kombination mit dem Abfackeldrucksignal **1102** dargestellt wurden. Bitte beachten sie, dass die Signale **1112** und **1102** zeitlich aufeinander abgestimmt sind, so dass die Zusammenhänge leicht zu erkennen sind. Wie in **Abb. 7D** ersichtlich, steht die Spitzenamplitude **1113** des Signals **1112** im Zusammenhang mit der Abfackelung **1108**. Die Daten **1112** scheinen verrauscht zu sein und enthalten so genannte falsch positive Werte (z.B. Ereignisse **1115a**, **1115b**). Falsch positive Werte sind jedoch nicht weiter problematisch, da sie beratend verwendet können und/oder für "beinahe" Abfackelungen stehen können, die verhindert wurden.

[0394] Um die Signaldaten **1112** weiter zu verarbeiten oder aufzubereiten, können zusätzliche Datenanalyseverfahren zum Einsatz kommen. In **Abb. 7E** wurde beispielsweise eine Hauptkomponentenanalyse an den Frequenzbereichsdaten der vierten Frequenz entsprechend der vier Zyklen pro Tag durchgeführt, um die erste Hauptkomponente zu bestimmen, wobei die Zeitreihenwerte der ersten Hauptkomponente als Signal **1118** dargestellt sind. **Abb. 7E** zeigt eine vergrößerte Ansicht des Verhaltens der ersten Hauptkomponente **1118** während des Zeitintervalls um die Abfackelung **1108** des Abfackeldrucksignals **1102**. Bitte beachten sie, dass die Signale **1118** und **1102** zeitlich aufeinander abgestimmt sind, so dass die Zusammenhänge leicht zu erkennen sind. Wie in **Abb. 7E** ersichtlich, steigt der Wert der ersten Hauptkomponente **1118** weit vor der Abfackelung **1108**, danach sinkt der Wert der ersten Hauptkomponenten **1118** deutlich. Um den Anstieg zu erfassen, kann ein abklingender Filter verwendet werden, um das Signal **1118** über einen gewissen Zeitraum zurückzuhalten. In einer Ausführungsform kann der abklingende Filter durch eine Datenblock-Definition und eine entsprechende Datenblockeigenschaften definiert werden, die eine individuelle Einstellung des abklingenden Filters für jede Einstellung ermöglichen. Die konfigurierbaren Eigenschaften des abklingenden Filterblocks können beispielsweise die Anstiegs- und/oder Abstiegsrate des Signals definieren. In einigen Ausführungsformen können zusätzliche Hauptkomponenten bestimmt und zur Steigerung der Sensitivität des Verfahrens verwendet werden.

[0395] Abb. 7F zeigt die vergrößerte Ansicht eines anderen Abschnitts von Abb. 7D. Abb. 7F zeigt das Verhalten der ersten Hauptkomponente 1118 während des Zeitintervalls um die Abfackelung 1110 des Abfackeldrucksignals 1102 detailliert. In Abb. 7F ist der Wert der ersten Hauptkomponente 1118 entsprechend der Abfackelung 1110 kleiner als für die Abfackelung 1108, dennoch ist die Differenz der Werte im Vergleich zu den Ausgangswerten noch immer deutlich.

[0396] Das Verfahren der rollenden FFT-Analyse zur Feststellung von Leitindikatoren in Prozessleitdaten kann in der Offline-Analyse und für den Aufbau von Datenmodellen zum Einsatz kommen. Sobald die neuen Prozessvariablen (z.B. Frequenzen, Hauptkomponenten und/oder relevante Daten höherer Ordnung) identifiziert und festgelegt wurden, kann die rollende FFT-Analyse an den von der verfahrenstechnischen Anlage online gestreamten Echtzeitdaten durchgeführt werden. Die rollende FFT kann beispielsweise als Datenblock definiert und in der Blockdefinitionsdatenbank 252 gespeichert werden, so dass Beispiele des rollenden FFT-Datenblocks in Online-Datenmodule integriert und mit den Online-Quellen innerhalb der verfahrenstechnischen Anlage verknüpft werden können. Die live gestreamten Daten, auf die die rollende FFT-Analyse angewendet wird, können dem Nutzer oder Anlagenbedie-ner Echtzeit-Prognosen-/Warnungen eines drohen-ten Zustandes liefern. In einigen Ausführungsfor-men kann der Nutzer einen kombinierten Datenblock definieren, um einen mit anderen Analysefunktionsblöcken (z.B. Hauptkomponentenanalyse, PLS-Pfad-analyse und/oder anderen Analysefunktionsblöcke) verknüpften, rollenden FFT-Datenblock zu integrie-ren. Zusätzlich kann der Nutzer einen Datenblock definieren, der verschiedene Prozesssignale zu ei-nem einzigen Signal kombiniert (z.B. Summe, Be-rechnung des gewichteten Durchschnitts usw.), um ihn als Eingabe für den rollenden FFT-Datenblock und/oder einen Kombinationsblock inklusive des rol-lenden FFT-Datenblocks zu verwenden.

[0397] Das rollende FFT-Datenanalyseverfahren ist folglich ein Beispiel für ein deskriptives Analysever-fahren, das sowohl für Offline-Prozessdaten als auch für Online-Prozessdaten verwendet werden kann.

[0398] In einigen Ausführungsformen können die neuen Prozessvariablen entsprechend des rollenden FFT-Analyseverfahrens in die verfahrenstechnische Anlage integriert werden. Die neuen Prozessvariablen können beispielsweise in einer Prozessleitdatenbank der verfahrenstechnischen Anlage definiert, identifi-ziert (z.B. durch Zuordnung entsprechender Prozess-leittags) und gespeichert werden. In einer Ausfüh-rungsform können die von der neuen Prozessvaria-bljen generierten Zeitreihendaten als Eingabe für ei-ne Steuerfunktion oder einen Steuerblock dienen, die

der Steuerung eines Teiles der Prozesse innerhalb der verfahrenstechnischen Anlage dienen, oder als Auslöser für einen Änderung innerhalb der verfahrenstechnischen Anlage.

[0399] Neben der Feststellung potentieller Abfacke-lungen gehören zu den weiteren Situationen, in de-nen die rollende FFT zur Vermeidung unerwünsch-ter Zustände oder Ereignisse zum Einsatz kommen kann, die Warnung vor der potentiellen Auslösung ei-nes Überdruckventsils, vor potentiellen Druckanstie-gen, vor einem bevorstehenden Pumpenausfall, ei-nem bevorstehenden Ventilausfall, instabilen Ström-mungszuständen wie einem Überlaufen der Rektifi-kationskolonnen, vor Ausfällen in den Mahlwerken, vor produktionsbedingten Instabilitäten in den Öl- und Gasbohrungen usw. In einigen Fällen kann die rol-lende FFT auch zum Schutz vor unerwünschten Leis-tungsmessungen einer oder mehrerer Einheiten in-nerhalb der verfahrenstechnischen Anlage oder der verfahrenstechnischen Anlage als Ganzes angewen-det werden.

[0400] In Abb. 7G ist ein Flussdiagramm für ein bei-spielhaftes Verfahren 1200 für die frühe Fehlerer-kennung in verfahrenstechnischen Anlagen und Pro- zessleitsystemen dargestellt. Ein oder mehrere Teile des Verfahrens 1200 können beispielsweise durch ei-nen Datenlock in der Blockdefinitionsdatenbank 252 ausgeführt werden. Ein oder mehrere Teile des Ver-fahrens 1200 können durch einen oder mehrere Teile des industriellen Prozessleistungsüberwachungs/-analysesystems 100, beispielsweise durch einen oder mehrere DDE 102x, durch ein Offline-Datenmodul, ein Online-Datenmodul usw. durchgeführt wer-den. Natürlich können ein oder mehrere Teile des Verfahrens 1200 auch durch andere als die in die-sem Dokument beschriebenen Systeme, Geräte und Vorrichtungen des Datenanalysesystems 100 durch-geföhrt werden. In einigen Ausführungsformen kann das Verfahren 1200 mehr, weniger oder andere als die in diesem Dokument beschriebenen Schritte um-fassen.

[0401] In Block 1202 kann das Verfahren 1200 den Erhalt oder die Erfassung eines ersten Satzes von Prozesssignalen oder Daten beinhalten, die als Ergeb-nis der Steuerung eines Prozesses durch die verfahrenstechnischen Anlage generiert wurden. Der erste Satz von Prozesssignalen kann Zeitbereichs-daten enthalten, die von einem oder mehreren Pro- zessleitgeräten zur Steuerung eines Prozesses in der verfahrenstechnischen Anlage generiert werden, und/oder Zeitbereichsdaten, die von einer oder meh-reren anderen Komponenten, Geräten oder Einhei-ten innerhalb der verfahrenstechnischen Anlage als Ergebnis der Prozesssteuerung generiert wurden (z.B. Prozessdaten der ersten Ordnung, die von verschie-denen Datenquellen innerhalb der verfahrenstech-nischen Anlage generiert wurden). Der erste Satz

an Prozesssignalen kann beispielsweise Werte der Prozessmessungen über einen bestimmten Zeitraum enthalten, wie beispielsweise Temperaturen, Ströme, Druckwerte, Zusammensetzungen, Zustände usw. In einigen Ausführungsformen kann der erfasste erste Satz an Prozesssignalen oder Daten Prozessdaten der zweiten oder einer anderen, höheren Ordnung über einen gewissen Zeitraum beinhalten, die als Ergebnis der Prozesstreuerung durch die verfahrenstechnischen Anlage generiert wurden, wie Diagnoseergebnisse, eine Serie von Maßnahmen durch den Bediener oder Nutzer, eine verfügbare Bandbreite eines Links des Prozessübertragungsnetzwerkes, ein Ergebnis einer Datenanalyse usw. Der erfasste erste Satz an Prozesssignalen kann beispielsweise auch Offline-Prozessdaten und/oder Online-Prozessdaten beinhalten.

[0402] In Block **1204** kann das Verfahren **1200** die Festlegung eines Leitindikators für eine Auffälligkeit, einen Fehler, einen Leistungsabfall und/oder einen anderen, unerwünschten/ungewollten Zustand beinhalten, der innerhalb der verfahrenstechnischen Anlage auftritt (oder aufgetreten ist), basierend auf dem erfassten, ersten Satz an Prozesssignalen. Der Leitindikator kann beispielsweise eine Änderung im Verhalten eines oder mehrerer Prozessleitsignale sein, die vor dem Eintreten der Auffälligkeit, des Fehlers, des Leistungsabfalls und/oder eines anderen Zustandes auftritt, wie beispielsweise ein Ausschlag in der Spitzenamplitude einer bestimmten Frequenz eines bestimmten Prozessleitsignals (z.B. der Ausschlag **1104** in Abb. 7B). In einer Ausführungsform kann der Leitindikator durch Verwendung eines oder mehrerer statistischer Analyseverfahren an einem oder mehreren erfassten, ersten Sätzen an Prozesssignalen bestimmt werden, z.B. durch Hauptkomponentenanalyse, PLS-Pfadanalyse, Clusteranalyse, Kreuzkorrelation usw. Ein oder mehrere Offline- und/oder Online-Datenblöcke und/oder -Datenmodule des Datenanalysesystems **100** können in einer weiteren Ausführungsform ein oder mehrere erfasste Prozesssignale verwenden, um einen oder mehrere Leitindikatoren für eine Auffälligkeit, einen Fehler, einen Leistungsabfall und/oder einen anderen Zustand festzulegen.

[0403] In Block **1206** kann das Verfahren **1200** die Erstellung, Definition oder Generierung eines Satzes mit einer oder mehreren neuen Prozessvariablen entsprechend des Leitindikators beinhalten. In einer Ausführungsform kann eine neue Prozessvariable entsprechend einer bestimmten Änderung im Verhalten eines bestimmten Signals erstellt, definiert oder generiert werden. Eine relevante Frequenz (z.B. Frequenz des Signals **1106** in Abb. 7C), in der der Leitindikator auftreten kann, kann beispielsweise identifiziert und als neue Prozessvariable erstellt/generiert werden. In einigen Ausführungsformen beinhaltet das Generieren eines Satzes mit einer oder meh-

reren neuen Prozessvariablen (Block **1206**) die Definition/Identifizierung/Generierung einer ersten neuen Prozessvariable entsprechend des Leitindikators (z.B. Änderung des Signals **1106** in Abb. 7C) und die anschließende Anwendung eines oder mehrerer Analyseverfahren an den durch die erste neue Prozessvariable generierten Zeitreihendaten (entweder allein oder in Kombination mit anderen, durch andere Prozessvariable generierte Zeitreihendaten) zur Festlegung einer anderen, neuen, mit dem Leitindikator verknüpften Prozessvariablen. Bezuglich Abb. 7D wurde eine Hauptkomponentenanalyse an den Frequenzbereichsdaten entsprechend des Signals **1106** aus Abb. 7C zur Bestimmung einer ersten Hauptkomponente durchgeführt, wobei die erste Hauptkomponente als eine weitere, neue Prozessvariable identifiziert/festgelegt wurde, deren Werte über einen bestimmten Zeitraum durch das Signal **1112** in Abb. 7C dargestellt sind.

[0404] In einigen Ausführungsformen des Blocks **1206** können ein oder mehrere Offline- und/oder Online-Datenblöcke und/oder -Datenmodule des Datenanalysesystems **100** dazu verwendet werden, eine oder mehrere neue Prozessvariablen entsprechend des Leitindikators zu identifizieren und zu erstellen/festlegen/generieren. Die eine oder mehreren neuen Prozessvariablen können in einigen Fällen getagt und/oder innerhalb der verfahrenstechnischen Anlage oder dem Prozessleitsystem gespeichert werden.

[0405] In Block **1208** kann das Verfahren **1200** die Erfassung eines nachfolgenden Satzes mit Prozesssignalen (z.B. in einer mit Block **1202** vergleichbaren Weise) und die Anwendung einer rollenden FFT auf diesen Satz zur Bestimmung von Zeitreihendaten entsprechend der einen oder mehreren neuen Prozessvariablen beinhalten. Der nachfolgende Satz mit Prozesssignalen kann beispielsweise Offline- und/oder Online-Signale beinhalten. Zur Anwendung einer rollenden FFT auf diesen Satz kann die FFT an einem ersten Datenfenster des nachfolgenden Satzes mit Prozesssignalen angewendet werden und die Werte aus den Spitzenamplituden der darin enthaltenen Frequenzen können als Zeitreihendaten gespeichert werden, eine FFT kann an einem zweiten Datenfenster durchgeführt werden und die Werte des Spitzenamplituden der darin enthaltenen Frequenzen können als Zeitreihendaten gespeichert werden und so weiter, wobei entsprechende Zeitreihendaten für eine oder mehrere relevante, im nachfolgenden Satz mit Prozesssignalen enthaltene Frequenzen generiert werden. Wenn eine bestimmte relevante Frequenz eines bestimmten Prozesssignals beispielsweise einer neuen Prozessvariablen entspricht, kann eine rollende FFT auf den nachfolgenden Satz mit Prozesssignalen angewendet werden, um das Verhalten einer bestimmten relevanten Frequenz innerhalb des nachfolgenden Satzes mit Prozesssignalen über einen bestimmten Zeitraum zu erfassen, z.B. um

Zeitreihendaten zu erfassen, in denen aus der neuen Prozessvariablen über einen bestimmten Zeitraum generierte Spitzenamplitudenwerte beinhaltet sind. In Situationen, in denen Analyseverfahren zur Identifizierung neuer Prozessvariablen durchgeführt wurden (z.B. entspricht die neue Prozessvariable der ersten Hauptkomponente der Frequenzbereichsdaten entsprechend Signal **1106**, wie zuvor beschrieben), so können das eine oder die mehreren Analyseverfahren auch auf den Block **1208** angewendet werden, um die Zeitreihendaten dieser neuen Prozessvariablen zu bestimmen. In einer Ausführungsform kann der Block **1208** von einem oder mehreren Teilen des Datenanalysesystems **100** durchgeführt werden, beispielsweise durch einen oder mehrere Online- und/oder Offline-Datenblöcke und/oder – Datenmodule.

[0406] In Block **1210** kann das Verfahren **1200** die Überwachung der durch eine oder mehrere neue Prozessvariablen (z.B. wie in Block **1208** erfasst) generierten Zeitreihendaten auf Präsenz des Leitindikators enthalten. Wenn die Spitzenamplitude einer bestimmten relevanten Frequenz beispielsweise einen Grenzwert für eine Größe und/oder eine Zeitspanne überschreitet, kann die Präsenz des Leitindikators festgestellt werden. In einer Ausführungsform wird die neue Prozessvariable durch ein entsprechendes Tag oder einen anderen Indikator identifiziert und in die verfahrenstechnische Anlage integriert, wobei die Überwachung der Zeitreihendatenwerte (Block **1215**) durch die Fehlerkennung, das Alarmsystem und/oder andere Überwachungseinrichtungen der verfahrenstechnischen Anlage durchgeführt werden kann. Die Überwachung der Zeitreihendaten (Block **1215**) zur Feststellung der Präsenz des Leitindikators kann zusätzlich oder alternativ durch das Datenanalysesystem **100** ausgeführt werden.

[0407] In Block **1212** kann das Verfahren **1200** die Generierung eines Indikators dafür umfassen, dass das Eintreten eines Fehlers, einer Auffälligkeit, eines Ereignisses, eines Leistungsabfalls, eines unerwünschten Zustandes und/oder eines erwünschten Zustandes usw. basierend auf der während der Überwachung des Blocks **1210** erkannten Präsenz des Leitindikators prognostiziert wurde. In einigen Fällen enthält der Block **1212** die Generierung eines Indikators für einen Zeitrahmen, während dessen das Auftreten des besagten Zustandes prognostiziert wird. Beispielsweise kann eine Trendkurve, ein Alarm, eine Warnsignal und/oder ein anderer visueller oder akustischer Indikator generiert und auf einer Benutzeroberfläche ausgegeben werden, um den Nutzer oder Bediener vor bevorstehenden Auffälligkeiten, Fehlern, Leistungsabfällen und/oder anderen Zuständen in der verfahrenstechnischen Anlage zu warnen, basierend auf der Präsenz des Leitindikators in den erfassten Zeitreihendaten.

[0408] In einigen Ausführungsformen (in Abb. 7G nicht dargestellt) kann das Verfahren **1200** beinhalten, dass ein Signal (beispielsweise ein Steuersignal oder anderes Signal, das eine Änderung anzeigen) generiert und der angeschlossenen verfahrenstechnischen Anlage bereitgestellt wird, basierend auf der erkannten Präsenz des Leitindikators. Das Datenanalysesystem **100** kann beispielsweise ein oder mehrere Steuersignale basierend auf der erfassten Präsenz des Leitindikators in den überwachten Daten (Block **1210**) generieren und das eine oder die mehreren Steuersignale automatisch einer oder mehreren Steuereinheiten zur Verfügung stellen, um das Verhalten mindestens eines Teils der verfahrenstechnischen Anlage zu ändern. Zu den weiteren Signalen, die der angeschlossenen verfahrenstechnischen Anlage zur Auslösung einer Änderung zur Verfügung gestellt werden können, gehören Steuerimpulse oder Signale, die eine Änderung eines Parameters, eines Wertes, einer Konfiguration und/oder eines Zustandes (z.B. eines Ausrüstungsgegenstandes, eines Gerätes, einer Routine oder Anwendung usw.) oder die Änderung an einer innerhalb der verfahrenstechnischen Anlage befindlichen oder gemeinsam mit dieser betriebenen Anwendung anzeigen.

[0409] In einigen Ausführungsformen (in Abb. 7G nicht dargestellt) kann das Verfahren **1200** beinhalten, dass ein oder mehrere Signale generiert und dem Datenanalysesystem **100** zur Verfügung gestellt werden. Das Verfahren **1200** kann beispielsweise beinhalten, dass dem Datenanalysesystem **100** ein Indikator für die neuen Prozessvariablen und deren entsprechende Identifikatoren, ein Indikator für die festgelegten Leitindikatoren, die Identitäten und Sequenzen der verschiedenen, an den Zeitreihendaten (und höher geordneten, daraus generierten Daten wie den Ergebnissen der verschiedenen, daran ausgeführten Analyseverfahren) durchgeführten Analyseverfahren zur Festlegung zusätzlicher, neuer Prozessvariablen, durch die neuen Prozessvariablen generierte, überwachte Zeitreihendaten sowie die Präsenz der Leitindikatoren usw. zur Verfügung gestellt werden. Im Allgemeinen kann das Verfahren **1200** umfassen, dass alle durch Anwendung des Verfahrens **1200** generierten Daten dem Datenanalysesystem **100** zur Verfügung gestellt werden. In einer Ausführungsform können das eine oder die mehreren Signale, die dem Datenanalysesystem **100** zur Verfügung gestellt werden, in Form von gestreamten Daten auftreten.

[0410] Das Verfahren **1200** ist natürlich nicht auf die Blöcke **1202–1212** beschränkt. In einigen Ausführungsformen können zusätzliche Blöcke durch das Verfahren **1200** ausgeführt werden und/oder einige der Blöcke **1202–1212** können bei dem Verfahren **1200** ausgelassen werden. Des Weiteren können Ausführungsformen des Verfahrens **1200** in Verbindung mit einem oder allen Teilen aus den Abb. 1,

Fig. 2, Fig. 3, Fig. 4A–Fig. 4Q, Fig. 5A–Fig. 5G und Fig. 6A–Fig. 6E und/oder einer beliebigen Anzahl an Funktionen und/oder Verfahren verwendet werden, die in den anderen Abschnitten dieses Dokumentes beschrieben sind.

[0411] Angesichts des Vorangegangen ist für jeden, der mit der Entwicklung von Prozessleitsystemen von zentralen Prozessleitsystemen, in denen die Steuerung und andere Verarbeitungsschritte mehr oder weniger zentral ausgeführt wurden, zu verteilten Steuersystemen (DCS) der nächsten Generation, in denen die Steuerung und andere Verarbeitungsschritte zwischen verschiedenen Steuereinheiten innerhalb der Anlage aufgeteilt sind, ersichtlich, dass das neue verteilte Analysesystem (DAS) **100** und die damit verbundenen, in diesem Dokument beschriebenen, industriellen Prozessleistungsüberwachungs- und -analyseverfahren, -geräte, -vorrichtungen, -komponenten und -verfahren einen vergleichbaren Sprung hin zu einer neuen Generation an Prozessleistungsüberwachungs- und -Analysesystemen darstellt. Anstelle einer gezielten, eng begrenzten Analyse an einem Bedienpult eines Prozessleitsystems, einer Offline-Analyseeinrichtung oder anstelle der Verwendung rudimentärer Analysegeräte ermöglichen das verteilte Analysesystem **100** und die in diesem Dokument beschriebenen, verbundenen Verfahren, Geräte, Vorrichtungen, Komponenten und Verfahren beispielsweise eine umfassende Überwachung und Echtzeitanalyse, die in der gesamten Anlage und sogar auf Unternehmens- und/oder Konzernebene verteilt sind, vergleichbar mit verteilten Steuersystemen (siehe z.B. **Abb. 2B**), so dass Beschreibungen, Prognosen und Verordnungen im Zusammenhang mit dem aktuellen Betrieb der verfahrenstechnischen Anlage in Echtzeit und zu jedem Zeitpunkt zur Verfügung stehen. Zusätzlich ermöglicht es die modulare, verteilte Architektur des DAS **100**, dass Leistungs- und Analysefunktionen so nah oder so fern wie möglich zu den Datenquellen der verfahrenstechnischen Anlage und/oder dem Prozessleitsystem integriert werden, zusätzlich ist es möglich, das DAS **100** bei Bedarf durch zusätzliche DDE **102x** zu ergänzen (z.B. für zusätzliche lokale Verarbeitungsleistung, zur Bereitstellung von Analysen in einer bestimmten Region oder einem bestimmten Bereich der Anlage **5**, bei Expansion der Anlage **5** usw.). Des Weiteren, wie bereits zuvor beschrieben, ermöglicht es das DAS **100**, dass Design, Definition, Nutzung und Überwachung der Datenmodule von den Bereichen und Plattformen abgeteilt oder getrennt werden, in denen sie zum Einsatz kommen, wodurch einem Dateningenieur oder einem Nutzer die Pflicht abgenommen wird, die Codierung und Zusammenstellung von Analyse- und Datenmodulen/-modellen auszuführen. Des Weiteren ermöglicht diese Trennung einer größere Flexibilität und nahtlose Migration in verschiedene Zielumfelder.

[0412] Bei Implementierung in eine Software kann jede der in diesem Dokument beschriebenen Anwendungen, Leistungen und Engines in jedem nichtflüchtigen, nichttransistorischen Speichermedium wie einer Magnetplatte, einer Laserdisk, einer Festplatte, einem Molekularspeicher oder anderem Speichermedium im RAM oder ROM eines Computers oder Prozessors usw. gespeichert werden. Obwohl die in diesem Dokument beschriebenen Systembeispiele sich inklusive aller auf einer Hardware ausgeführter Software und/oder Firmware, neben anderen Komponenten, verstehen, sind die aufgeführten Systeme nur zur Veranschaulichung und keinesfalls als Beschränkung zu betrachten. Es ist beispielsweise vorgesehen, dass jede dieser Hardware-, Software- und Firmwarekomponenten exklusiv in Hardware, exklusive in Software oder in jeglicher Kombination aus Hardware und Software ausgeführt werden kann. Entsprechend ist es für Personen mit durchschnittlichen Kenntnissen zum Thema offensichtlich, dass die in diesem Dokument beschriebenen Systembeispiele, obwohl in einer Software implementiert, die auf dem Prozessor eines oder mehrerer Computergeräte ausgeführt ist, nicht die einzigen Möglichkeiten darstellen, ein solches System zu implementieren.

[0413] Obwohl die vorliegende Erfindung mit Bezug auf spezifische Beispiele beschrieben wurde, die ausschließlich zur Veranschaulichung und nicht zur Beschränkung der Erfindung dienen, ist es für Personen mit durchschnittlichen Fachkenntnissen offensichtlich, dass Änderungen, Erweiterungen oder Lösungen an den beschriebenen Ausführungsformen möglich sind, ohne vom Grundgedanken der Erfindung abzuweichen.

[0414] Die spezifischen Merkmale, Strukturen und/oder Eigenschaften der jeweiligen Ausführungsformen können auf jede geeignete Art und Weise und/oder durch jede geeignete Kombination mit einer und/oder mehreren anderen Ausführungsformen kombiniert werden, inklusive der Verwendung von ausgewählten Merkmalen mit oder ohne entsprechender Verwendung anderer Merkmale. Zusätzlich können zahlreiche Änderungen vorgenommen werden, um eine bestimmte Anwendung, Situation und/oder ein bestimmtes Material an den Grundgedanken der vorliegenden Erfindung anzupassen. Es versteht sich, dass andere Variationen und/oder Änderungen an den Ausführungsformen der in diesem Dokument beschriebenen und/oder dargestellten Erfindung angesichts der in diesem Dokument aufgeführten Erkenntnisse möglich sind und als Teil des Grundgedankens der vorliegenden Erfindung zu verstehen sind. Bestimmte Aspekte der Erfindung sind in diesem Dokument als beispielhafte Aspekte beschrieben.

[0415] Die folgenden Aspekte werden als Beispiel, nicht aber als Beschränkung angegeben.

A1. Ein Verfahren zum Abrufen von Zeitreihendaten aus einer elektronischen Datenquelle, umfassend:

Empfangen einer Angabe zu der elektronischen Datenquelle, aus der die Daten abgerufen werden sollen;

Empfangen einer Angabe über einen Zeitbereich, in dem die Daten abgerufen werden sollen, wobei die Angabe in einem standardisierten Abfrageformat ausgedrückt wird, das unabhängig ist von der elektronischen Datenquelle;

Empfangen einer Angabe über ein Merkmal der Daten, die abgerufen werden sollen, wobei die Angabe in dem standardisierten Abfrageformat ausgedrückt wird, das unabhängig ist von der elektronischen Datenquelle;

Bestimmen eines datenquellspezifischen Abfrageformats, das mit der elektronischen Datenquelle assoziiert ist;

Generieren einer datenquellspezifischen Abfrage, die mit dem datenquellspezifischen Format kompatibel ist, auf Basis der Angaben zum Zeitbereich und dem Merkmal der Daten, die abgerufen werden sollen;

Bewirken, dass die datenquellspezifische Abfrage ausgeführt wird, um Daten aus der elektronischen Datenquelle auszuwählen; und/oder Empfangen der Daten von der elektronischen Datenquelle.

A2. Das Verfahren nach dem Aspekt A1, wobei die Datenquelle eine relationale Datenquelle oder eine nicht-relationale Datenquelle ist.

A3. Das Verfahren nach einem der Aspekte A1 oder A2, wobei die Angabe zum Zeitbereich und/oder die Angabe zum Merkmal in einer standardisierten Abfrage empfangen werden, die das standardisierte Abfrageformat nutzt.

A4. Das Verfahren nach dem Aspekt A3, wobei die Angabe zu der elektronischen Datenquelle auch in der standardisierten Abfrage empfangen wird.

A5. Das Verfahren nach einem der Aspekte A3 oder A4, wobei die standardisierte Abfrage ferner ein Format für die Daten angibt.

A6. Das Verfahren nach einem der Aspekte A3–A5, wobei die standardisierte Abfrage eine JavaScript Object Notation(JSON)-Datei ist.

A7. Das Verfahren nach einem der Aspekte A1–A6, ferner das Empfangen einer Angabe zu einem Format umfassend, in dem die Daten aus der elektronischen Datenquelle empfangen werden sollen, und/oder wobei die Daten aus der elektronischen Datenquelle in dem angegebenen Format empfangen werden.

A8. Das computerimplementierte Verfahren nach einem der Aspekte A1–A6, ferner umfassend: Empfangen einer Angabe zu einem Format, in dem die Daten abgerufen werden sollen, und/oder Umwandeln der aus der elektronischen Datenquelle empfangenen Daten in das angegebene Format.

A9. Das Verfahren nach einem der Aspekte A1–A8, wobei das standardisierte Abfrageformat eine Syntax verwendet, die sich von einer nativen Syntax, die von der elektronischen Datenquelle genutzt wird, unterscheidet.

A10. Das Verfahren nach einem der Aspekte A1–A9, wobei das standardisierte Abfrageformat eine Syntax verwendet, die nicht direkt ausführbar ist, um die Daten aus der elektronischen Datenquelle abzurufen.

A11. Das Verfahren nach einem der Aspekte A1–A10, wobei das Generieren der datenquellspezifischen Abfrage das Abbilden des angegebenen Zeitbereichs und/oder der angegebenen Merkmale der Daten, die abgerufen werden sollen, auf das bestimmte datenquellspezifische Abfrageformat beinhaltet.

A12. Das Verfahren nach einem der Aspekte A1–A11, wobei das datenquellspezifische Abfrageformat auf Basis eines Typs der elektronischen Datenquelle bestimmt wird.

A13. Das Verfahren nach einem der Aspekte A1–A12, wobei das Bewirken, dass die datenquellspezifische Abfrage ausgeführt wird, das Senden der datenquellspezifischen Abfrage zu einer Execution Engine der elektronischen Datenquelle beinhaltet, wobei die Execution Engine die elektronische Datenquelle unter Verwendung der datenquellspezifischen Abfrage befragt.

A14. Das Verfahren nach einem der Aspekte A1–A13, wobei die Angabe des Zeitbereichs mehrere Startzeiten und/oder mehrere Endzeiten beinhaltet.

A15. Das Verfahren nach einem der Aspekte A1–A14, wobei die Angabe des Zeitbereichs eine Sampling-Rate für die Daten, die abgerufen werden sollen, beinhaltet.

A16. Das Verfahren nach einem der Aspekte A1–A15, ferner das Generieren eines Datenrahmens umfassend, der die von der elektronischen Datenquelle empfangenen Daten beinhaltet, wobei der Datenrahmen mehrere Datenpunkte beinhaltet, wobei jeder Datenpunkt mit einer Zeit innerhalb des Zeitrahmens assoziiert ist.

A17. Das Verfahren nach dem Aspekt A16, wobei jeder Datenpunkt mit einer Zeit innerhalb des Zeitbereichs assoziiert ist, die mit einem ganzzahligen Vielfachen einer Sampling-Periode, die auf eine Startzeit folgt, abgeglichen ist, und/oder die Startzeit ist.

A18. Das Verfahren nach einem der Aspekte A1–A17, wobei die Angabe zu den Merkmale der Daten, die abgerufen werden sollen, eine Angabe zu einem oder mehreren Messungstypen in Bezug auf den Betrieb einer Prozessanlage beinhaltet.

A19. Das Verfahren nach dem Aspekt A18, wobei der eine oder die mehreren Messungstypen Messungen von einer oder mehreren Vorrichtungen vor Ort, die innerhalb der Prozessanlage angeordnet sind, beinhaltet bzw. beinhalten.

A20. Das Verfahren nach einem der Aspekte A18 oder A19, wobei die Angabe zu dem einen oder den mehreren Messungstypen ein oder mehrere Tags, Aliases und/oder Datentypen beinhaltet, die mit den Daten assoziiert sind.

B1. Verfahren zum Bereitstellen von Daten aus einer elektronischen Datenquelle, umfassend:

Empfangen einer standardisierten Abfrage in einem standardisierten Abfrageformat von einer Datenabfrageinstanz, wobei das standardisierte Abfrageformat von einem datenquellspezifischen Abfrageformat, das von der elektronischen Datenquelle genutzt wird, verschieden ist;

Extrahieren von Abfrageparametern aus der standardisierten Abfrage;

Generieren einer datenquellspezifischen Abfrage in dem datenquellspezifischen Abfrageformat auf Basis der extrahierten Abfrageparameter; Ausführen der datenquellspezifischen Abfrage, um die Daten aus der elektronischen Datenquelle abzurufen; und/oder

Bereitstellen der abgerufenen Daten an einer Datenempfangsinstanz.

B2. Das Verfahren nach dem Aspekt B1, wobei die Datenabfrageinstanz einen Abfrageblock innerhalb eines Analyseprogramms umfasst.

B3. Das Verfahren nach einem der Aspekte B1 oder B2, wobei die elektronische Datenquelle eine nicht-relationale Datenbank ist.

B4. Das Verfahren nach einem der Aspekte B1–B3, wobei die standardisierte Abfrage eine Syntax verwendet, die nicht direkt ausführbar ist, um die Daten aus der elektronischen Datenquelle abzurufen.

B5. Das Verfahren nach einem der Aspekte B1–B4, wobei das Extrahieren der Abfrageparameter beinhaltet: Bestimmen einer Zeitspanne und/oder eines Datenparameters der Daten, wobei die Zeitspanne eine Startzeit und/oder eine Endzeit und/oder den Datenparameter aufweist, der einen Typ der Daten angibt, die aus der elektronischen Datenquelle abgerufen werden sollen.

B6. Das Verfahren nach dem Aspekt B5, wobei das Extrahieren der Abfrageparameter ferner das Bestimmen einer Sampling-Rate beinhaltet.

B7. Das Verfahren nach dem Aspekt B6, wobei das Bereitstellen der abgerufenen Daten an der Abfrageinstanz das Antworten mit einem Satz von mit der Zeitspanne assoziierten Datenpunkten beinhaltet, wobei jeder Datenpunkt einen Zeitstempel beinhaltet, der die Startzeit und/oder ein ganzzahliges Vielfaches der Spanne der Sampling-Rate im Anschluss an die Startzeit ist.

B8. Das Verfahren nach dem Aspekt B7, wobei ein oder mehrere Datenpunkte mit Zeitstempeln, die ganzzahlige Vielfache der Zeitspanne der Sampling-Rate nach der Startzeit angeben, unter Verwendung von Werten von Dateneinträgen in die elektronische Datenquelle, die mit Zeiten assoziiert sind, die den einzelnen Zeitstempeln des ei-

nen oder der mehreren Datenpunkte mit dem kürzesten Abstand vorangehen, zu dem Datensatz hinzugefügt werden.

B9. Das Verfahren nach einem der Aspekte B5–B8, wobei der Satz von Datenpunkten auf Basis eines Formats formatiert wird, das in der standardisierten Abfrage angegeben ist.

B10. Das Verfahren nach einem der Aspekte B5–B9, wobei der Typ, der von dem Datenparameter angegeben wird, ein Messungstyp und/oder eine Messung von einem Messvorrichtungstyp ist.

B11. Das Verfahren nach dem Aspekt B10, wobei der Typ ferner eine bestimmte Messvorrichtung angibt.

B12. Das Verfahren nach einem der Aspekte B11 oder B12, wobei die Messvorrichtung eine Vorrichtung vor Ort ist, die innerhalb einer Prozessanlage angeordnet ist.

B13. Das Verfahren nach einem der Aspekte B5–B12, wobei der Datenparameter ferner ein Tag der Daten, die abgerufen werden sollen, ein Alias für die Daten, die an der Datenabfrageinstanz bereitgestellt werden sollen, und/oder einen Datenformattyp für die Daten, die an der Datenabfrageinstanz bereitgestellt werden sollen, angibt.

B14. Das Verfahren nach einem der Aspekte B1–B13, wobei das Bereitstellen der Daten an der Datenempfangsinstanz das Senden eines Datenrahmens, der die abgerufenen Daten enthält, an die Datenempfangsinstanz umfasst.

B15. Das Verfahren nach einem der Aspekte B1–B14, wobei die standardisierte Abfrage eine Angabe eines Formats für den Datenrahmen beinhaltet, und/oder wobei der Datenrahmen gemäß der Angabe des Formats für den Datenrahmen formatiert wird.

B16. Das Verfahren nach einem der Aspekte B1–B15, wobei die standardisierte Abfrage angibt, dass der Datenrahmen als JavaScript Object Notation(JSON)-Datei bereitgestellt werden soll.

B17. Das Verfahren nach einem der Aspekte B1–B15, wobei die Datenempfangsinstanz dieselbe Instanz ist wie die Datenabfrageinstanz.

C1. Ein Verfahren zum Zugreifen auf Prozessanlagendaten aus einer Mehrzahl von elektronischen Datenquellen, umfassend:

Empfangen einer standardisierten Abfrage, die ein standardisiertes Abfrageformat nutzt;

Generieren einer ersten datenquellspezifischen Abfrage auf Basis der standardisierten Abfrage, wobei die erste datenquellspezifische Abfrage ein erstes Abfrageformat nutzt, das mit einer ersten elektronischen Datenquelle assoziiert ist;

Generieren einer zweiten datenquellspezifischen Abfrage auf Basis der standardisierten Abfrage, wobei die zweite datenquellspezifische Abfrage ein zweites Abfrageformat nutzt, das mit einer zweiten elektronischen Datenquelle assoziiert ist;

Bewirken, dass die erste datenquellspezifische Abfrage ausgeführt wird, um einen ersten Datensatz aus der ersten elektronischen Datenquelle abzurufen;

Bewirken, dass die zweite datenquellspezifische Abfrage ausgeführt wird, um einen zweiten Datensatz aus der zweiten elektronischen Datenquelle abzurufen; und/oder

Generieren eines kombinierten Datenrahmens, der den ersten und den zweiten Datensatz beinhaltet.

C2. Das Verfahren nach dem Aspekt C1, wobei die standardisierte Abfrage eine erste Angabe zu der ersten elektronischen Datenquelle und/oder eine zweite Angabe zu der zweiten elektronischen Datenquelle beinhaltet.

C3. Das Verfahren nach dem Aspekt C2, wobei die erste Angabe das erste Abfrageformat benennt und/oder die zweite Angabe das zweite Abfrageformat benennt.

C4. Das Verfahren nach einem der Aspekte C2 oder C3, wobei die erste und/oder die zweite Angabe Angaben zu Datenspalten vorangestellt werden.

C5. Das Verfahren nach einem der Aspekte C1–C4, wobei das standardisierte Abfrageformat eine Abfragesyntax nutzt, die sich von einer Abfragesyntax des ersten Abfrageformats und/oder einer Abfragesyntax des zweiten Abfrageformats unterscheidet.

C6. Das Verfahren nach einem der Aspekte C5, wobei die Abfragesyntax des standardisierten Abfrageformats nicht direkt ausführbar ist, um den ersten Datensatz von der ersten elektronischen Datenquelle und/oder den zweiten Datensatz von der zweiten elektronischen Datenquelle abzurufen.

C7. Das Verfahren nach einem der Aspekte C1–C6, wobei die standardisierte Abfrage in einer JavaScript Object Notation(JSON)-Datei enthalten ist.

C8. Das Verfahren nach einem der Aspekte C1–C7, wobei die standardisierte Abfrage eine Angabe eines Zeitbereichs beinhaltet, der eine Zeitspanne benennt, die Dateneinträgen im ersten Datensatz und/oder im zweiten Datensatz entspricht.

C9. Das Verfahren nach dem Aspekt C8, wobei der Zeitbereich eine Mehrzahl von Zeitspannen beinhaltet.

C10. Das Verfahren nach dem Aspekt C9, wobei mindestens zwei von den mehreren Zeitspannen durch Ausschlusszeitspannen getrennt sind, in denen keine Daten abgefragt werden.

C11. Das Verfahren nach einem der Aspekte C9–C10, wobei der Zeitbereich eine Mehrzahl von Zeitspannen beinhaltet, und wobei der Zeitbereich eine erste, mit dem ersten Datensatz assoziierte Zeitspanne und/oder eine zweite, mit dem zweiten Datensatz assoziierte Zeitspanne benennt.

C12. Das Verfahren nach einem der Aspekte C1–C11, wobei die erste elektronische Datenquelle eine relationale Datenquelle ist und/oder die zweite elektronische Datenquelle eine nicht-relationale Datenquelle ist.

C13. Das Verfahren nach einem der Aspekte C1–C12, wobei die standardisierte Abfrage eine Angabe zu einer Sampling-Rate für den kombinierten Datenrahmen beinhaltet.

C14. Das Verfahren nach einem der Aspekte C1–C13, wobei:

der erste Datensatz Daten mit einer ersten Abtastrate beinhaltet,

der zweite Datensatz Daten mit einer zweiten Abtastrate beinhaltet, und/oder

das Generieren des kombinierten Datenrahmens das Abgleichen der ersten und der zweiten Abtastrate beinhaltet.

C15. Das Verfahren nach dem Aspekt C14, wobei das Abgleichen der ersten und der zweiten Sampling-Rate das Hinzufügen von Datenpunkten beinhaltet, die Zeiten ohne Abtastung entsprechen, die nötig sind, um eine gewünschte Sampling-Rate zu erreichen. C16. Das Verfahren nach dem Aspekt C15, wobei die hinzugefügten Datenpunkte gegebene Werte der abgetasteten Daten sind, die den hinzugefügten Datenpunkten zeitlich direkt vorangehen.

C17. Das Verfahren nach dem Aspekt C14, wobei das Abgleichen der ersten und der zweiten Sampling-Rate das Entfernen von abgetasteten Datenpunkten beinhaltet.

C18. Das Verfahren nach einem der Aspekte C1–C15, wobei der erste und der zweite Datensatz jeweils mehrere Datenpunkte beinhalten, die Information enthalten, die von einer oder mehreren Vorrichtungen vor Ort innerhalb einer Prozessanlage gemessen werden.

D1. Ein System zum Abrufen von Zeitreihendaten, die mit dem Betrieb einer Prozessanlage assoziiert sind, umfassend:

eine elektronische Datenquelle, die ein materielles, nichtflüchtiges Speichermedium umfasst, das die Daten speichert, die mit dem Betrieb der Prozessanlage assoziiert sind;

einen Prozessor, der kommunikationstechnisch mit der elektronischen Datenquelle gekoppelt ist; einen Programmspeicher, der kommunikationstechnisch mit dem Prozessor verbunden ist und der ein materielles, nichtflüchtiges Speichermedium umfasst, das Anweisungen speichert, die, wenn sie vom Prozessor ausgeführt werden, den Prozessor zu folgendem veranlassen:

Empfangen einer Angabe zu der elektronischen Datenquelle, aus der die Daten abgerufen werden sollen; Empfangen einer Angabe über einen Zeitbereich, in dem die Daten abgerufen werden sollen, wobei die Angabe in einem standardisierten Abfragefor-

mat ausgedrückt wird, das unabhängig ist von der elektronischen Datenquelle;

Empfangen einer Angabe über ein Merkmal der Daten, die abgerufen werden sollen, wobei die Angabe in dem standardisierten Abfrageformat ausgedrückt wird, das unabhängig ist von der elektronischen Datenquelle;

Bestimmen eines datenquellspezifischen Abfrageformats, das mit der elektronischen Datenquelle assoziiert ist;

Generieren einer datenquellspezifischen Abfrage, die mit dem datenquellspezifischen Format kompatibel ist, auf Basis der Angaben zum Zeitbereich und dem Merkmal der Daten, die abgerufen werden sollen;

Bewirken, dass die datenquellspezifische Abfrage ausgeführt wird, um Daten aus der elektronischen Datenquelle auszuwählen; und/oder Empfangen der Daten von der elektronischen Datenquelle.

D2. Das System nach dem Aspekt D1, wobei die Datenquelle eine relationale Datenquelle oder eine nicht-relationale Datenquelle ist.

D3. Das System nach einem der Aspekte D1 oder D2, wobei die Angabe zum Zeitbereich und/oder die Angabe zum Merkmal in einer standardisierten Abfrage empfangen werden, die das standardisierte Abfrageformat nutzt.

D4. Das System nach dem Aspekt D3, wobei die Angabe zu der elektronischen Datenquelle auch in der standardisierten Abfrage empfangen wird.

D5. Das System nach einem der Aspekte D3 oder D4, wobei die standardisierte Abfrage ferner ein Format für die Daten angibt.

D6. Das System nach einem der Aspekte D3–D5, wobei die standardisierte Abfrage eine JavaScript Object Notation(JSON)-Datei ist.

D7. Das System nach einem der Aspekte D1–D6, wobei der Programmspeicher ferner Anweisungen speichert, die bewirken, dass der Prozessor eine Angabe zu einem Format empfängt, in dem die Daten aus der elektronischen Datenquelle empfangen werden sollen, und/oder wobei die Daten in dem angegebenen Format aus der elektronischen Datenquelle empfangen werden.

D8. Das System nach einem der Aspekte D1–D6, wobei der Programmspeicher ferner Anweisungen speichert, die bewirken, dass der Prozessor:

- eine Angabe zu einem Format empfängt, in dem die Daten abgerufen werden sollen, und/oder die aus der elektronischen Datenquelle abgerufenen Daten in das angegebene Format umwandelt.

D9. Das System nach einem der Aspekte D1–D8, wobei das standardisierte Abfrageformat eine Syntax verwendet, die sich von einer nativen Syntax, die von der elektronischen Datenquelle genutzt wird, unterscheidet.

D10. Das System nach einem der Aspekte D1–D9, wobei das standardisierte Abfrageformat eine

Syntax verwendet, die nicht direkt ausführbar ist, um die Daten aus der elektronischen Datenquelle abzurufen.

D11. Das System nach einem der Aspekte D1–D10, wobei die Anweisungen, die bewirken, dass der Prozessor die datenquellspezifische Abfrage generiert, bewirken, dass der Prozessor den angegebenen Zeitbereich und/oder die angegebenen Merkmale der Daten, die abgerufen werden sollen, auf das bestimmte datenquellspezifische Abfrageformat abbildet.

D12. Das System nach einem der Aspekte D1–D11, wobei das datenquellspezifische Abfrageformat auf Basis eines Typs der elektronischen Datenquelle bestimmt wird.

D13. Das System nach einem der Aspekte D1–D12, wobei die Anweisungen, die bewirken, dass der Prozessor bewirkt, dass die datenquellspezifische Abfrage ausgeführt wird, bewirken, dass der Prozessor die datenquellspezifische Abfrage zu einer Execution Engine der elektronischen Datenquelle schickt, wobei die Execution Engine die elektronische Datenquelle unter Verwendung der datenquellspezifischen Abfrage befragt.

D14. Das System nach einem der Aspekte D1–D13, wobei die Angabe des Zeitbereichs mehrere Startzeiten und/oder mehrere Endzeiten beinhaltet.

D15. Das System nach einem der Aspekte D1–D14, wobei die Angabe des Zeitbereichs eine Sampling-Rate für die Daten, die abgerufen werden sollen, beinhaltet.

D16. Das System nach einem der Aspekte D1–D15, wobei der Programmspeicher ferner Anweisungen speichert, die bewirken, dass der Prozessor einen Datenrahmen generiert, der die Daten beinhaltet, die aus der elektronischen Datenquelle empfangen werden, wobei der Datenrahmen mehrere Datenpunkte beinhaltet, wobei jeder Datenpunkt mit einer Zeit innerhalb des Zeitrahmens assoziiert ist.

D17. Das System nach dem Aspekt D16, wobei jeder Datenpunkt mit einer Zeit innerhalb des Zeitbereichs assoziiert ist, die mit einem ganzzahligen Vielfachen einer Sampling-Periode, die auf einen Startzeitpunkt folgt, abgeglichen ist, und/oder die Startzeit ist.

D18. Das System nach einem der Aspekte D1–D17, wobei die Angabe zu den Merkmalen der Daten, die abgerufen werden sollen, eine Angabe zu einem oder mehreren Messungstypen in Bezug auf den Betrieb der Prozessanlage beinhaltet.

D19. Das System nach dem Aspekt D18, ferner eine oder mehrere Vorrichtungen vor Ort umfassend, die innerhalb der Prozessanlage angeordnet ist bzw. sind, wobei der eine oder die mehreren Messungstypen Messungen von der einen oder von den mehreren Vorrichtung vor Ort beinhaltet bzw. beinhalten.

D20. Das System nach einem der Aspekte D18 oder D19, wobei die Angabe zu dem einen oder den mehreren Messungstypen ein oder mehrere Tags, Aliases und/oder Datentypen beinhaltet, die mit den Daten assoziiert sind.

D21. Das System nach einem der Aspekte D1–D20, wobei die elektronische Datenquelle ein Massendatenspeicherknoten ist.

E1. Ein System zum Bereitstellen von Daten, die mit dem Betrieb einer Prozessanlage assoziiert sind, umfassend:

eine elektronische Datenquelle, die ein materielles, nichtflüchtiges Speichermedium umfasst, das die Daten speichert, die mit dem Betrieb der Prozessanlage assoziiert sind;

einen Prozessor, der kommunikationstechnisch mit der elektronischen Datenquelle gekoppelt ist; einen Programmspeicher, der kommunikationstechnisch mit dem Prozessor gekoppelt ist und der ein materielles, nichtflüchtiges Speichermedium umfasst, das Anweisungen speichert, die, wenn sie vom Prozessor ausgeführt werden, den Prozessor zu folgendem veranlassen:

Empfangen einer standardisierten Abfrage in einem standardisierten Abfrageformat von einer Datenabfrageinstanz, wobei das standardisierte Abfrageformat von einem datenquellspezifischen Abfrageformat, das von der elektronischen Datenquelle genutzt wird, verschieden ist;

Extrahieren von Abfrageparametern aus der standardisierten Abfrage;

Generieren einer datenquellspezifischen Abfrage in dem datenquellspezifischen Abfrageformat auf Basis der extrahierten Abfrageparameter; Ausführen der datenquellspezifischen Abfrage, um die Daten aus der elektronischen Datenquelle abzurufen; und/oder

Bereitstellen der abgerufenen Daten an einer Datenempfangsinstanz.

E2. Das System nach dem Aspekt D1, wobei die Datenabfrageinstanz einen Abfrageblock innerhalb eines Analyseprogramms umfasst.

E3. Das System nach einem der Aspekte E1 oder E2, wobei die elektronische Datenquelle eine nicht-relationale Datenquelle ist.

E4. Das System nach einem der Aspekte D1–E3, wobei die standardisierte Abfrage eine Syntax verwendet, die nicht direkt ausführbar ist, um die Daten aus der elektronischen Datenquelle abzurufen.

E5. Das System nach einem der Aspekte E1–E4, wobei die Angaben, die bewirken, dass der Prozessor die Abfrageparameter extrahiert, bewirken, dass der Prozessor eine Zeitspanne und/oder einen Datenparameter der Daten bestimmt, wobei die Zeitspanne eine Startzeit und/oder eine Endzeit und/oder den Datenparameter beinhaltet, der einen Typ von Daten angibt, die aus der elektronischen Datenquelle abgerufen werden sollen.

E6. Das System nach dem Aspekt E5, wobei das Extrahieren der Abfrageparameter ferner das Bestimmen einer Sampling-Rate beinhaltet.

E7. Das System nach dem Aspekt E6, wobei die Anweisungen, die bewirken, dass der Prozessor abgerufene Daten an der Abfrageinstanz bereitstellt, bewirken, dass der Prozessor mit einem Satz von Datenpunkten, die mit der Zeitspanne assoziiert sind, antwortet, wobei jeder Datenpunkt einen Zeitstempel beinhaltet, der die Startzeit oder ein ganzzahliges Vielfaches der Spanne der Sampling-Rate im Anschluss an die Startzeit ist.

E8. Das System nach dem Aspekt E7, wobei ein oder mehrere Datenpunkte mit Zeitstempeln, die ganzzahlige Vielfache der Zeitspanne der Sampling-Rate nach der Startzeit angeben, unter Verwendung von Werten von Dateneinträgen in die elektronische Datenquelle, die mit Zeiten assoziiert sind, die den einzelnen Zeitstempeln des einen oder der mehreren Datenpunkte mit dem kürzesten Abstand vorangehen, zu dem Datensatz hinzugefügt werden.

E9. Das System nach einem der Aspekte E5–E8, wobei der Satz von Datenpunkten auf Basis eines Formats formatiert wird, das in der standardisierten Abfrage angegeben ist.

E10. Das System nach einem der Aspekte E4–E9, wobei der Typ, der von dem Datenparameter angegeben wird, ein Messungstyp und/oder eine Messung von einem Messvorrichtungstyp ist.

E11. Das System nach dem Aspekt E10, wobei der Typ ferner eine bestimmte Messvorrichtung angibt.

E12. Das System nach einem der Aspekte E11 oder E12, wobei die Messvorrichtung eine Vorrichtung vor Ort ist, die innerhalb der Prozessanlage angeordnet ist.

E13. Das System nach einem der Aspekte E5–E12, wobei der Datenparameter ferner ein Tag der Daten, die abgerufen werden sollen, ein Alias für die Daten, die an der Datenabfrageinstanz bereitgestellt werden sollen, und/oder einen Datenformattyp für die Daten, die an der Datenabfrageinstanz bereitgestellt werden sollen, angibt.

E14. Das System nach einem der Aspekte E1–E13, wobei die Angaben, die bewirken, dass der Prozessor die Daten an der Datenempfangsinstanz bereitstellt, bewirken, dass der Prozessor einen Datenrahmen, der die abgerufenen Daten enthält, an die Datenempfangsinstanz schickt.

E15. Das System nach einem der Aspekte E1–E14, wobei die standardisierte Abfrage eine Angabe eines Formats für den Datenrahmen beinhaltet, und/oder wobei der Datenrahmen gemäß der Angabe des Formats für den Datenrahmen formatiert wird.

E16. Das System nach einem der Aspekte E1–E15, wobei die standardisierte Abfrage angibt, dass der Datenrahmen als JavaScript Object Notation(JSON)-Datei bereitgestellt werden soll.

E17. Das System nach einem der Aspekte E1–E15, wobei die Datenempfangsinstanz dieselbe Instanz ist wie die Datenabfrageinstanz.

F1. System zum Zugreifen auf Prozessanlagen-daten, die mit dem Betrieb einer Prozessanlage assoziiert sind, aus einer Mehrzahl elektronischer Datenquellen, umfassend:

eine erste elektronische Datenquelle und eine zweite elektronische Datenquelle, die jeweils ein materielles, nichtflüchtiges Speichermedium umfasst, das die Prozessanlagendaten speichert, die mit dem Betrieb der Prozessanlage assoziiert sind;

einen Prozessor, der kommunikationstechnisch mit der ersten und der zweiten elektronischen Datenquelle gekoppelt ist;

einen Programmspeicher, der kommunikations-technisch mit dem Prozessor gekoppelt ist und der ein materielles, nichtflüchtiges Speichermedium umfasst, das Anweisungen speichert, die, wenn sie vom Prozessor ausgeführt werden, den Pro- zessor zu folgendem veranlassen:

Empfangen einer standardisierten Abfrage, die ein standardisiertes Abfrageformat nutzt;

Generieren einer ersten datenquellenspezifischen Abfrage auf Basis der standardisierten Abfrage, wobei die erste datenquellenspezifische Abfrage ein erstes Abfrageformat nutzt, das mit der ersten elektronischen Datenquelle assoziiert ist;

Generieren einer zweiten datenquellenspezifi- schen Abfrage auf Basis der standardisierten Ab- frage, wobei die zweite datenquellenspezifische Abfrage ein zweites Abfrageformat nutzt, das mit der zweiten elektronischen Datenquelle assoziiert ist;

Bewirken, dass die erste datenquellenspezifische Abfrage ausgeführt wird, um einen ersten Daten- satz aus der ersten elektronischen Datenquelle abzurufen;

Bewirken, dass die zweite datenquellenspezifi- sche Abfrage ausgeführt wird, um einen zweiten Datensatz aus der zweiten elektronischen Daten- quelle abzurufen; und/oder

Generieren eines kombinierten Datenrahmens, der den ersten und den zweiten Datensatz beinhaltet.

F2. Das System nach dem Aspekt F1, wobei die standardisierte Abfrage eine erste Angabe zu der ersten elektronischen Datenquelle und/oder eine zweite Angabe zu der zweiten elektronischen Da- tenquelle beinhaltet.

F3. Das System nach dem Aspekt F2, wobei die erste Angabe das erste Abfrageformat benennt und/oder die zweite Angabe das zweite Abfrage- format benennt.

F4. Das System nach einem der Aspekte F2 oder F3, wobei die erste und/oder die zweite Angabe Angaben zu Datenspalten vorangestellt werden.

F5. Das System nach einem der Aspekte F1–F4, wobei das standardisierte Abfrageformat eine Ab-

fragesyntax nutzt, die sich von einer Abfragesyntax des ersten Abfrageformats und/oder einer Ab- fragesyntax des zweiten Abfrageformats unter- scheidet.

F6. Das System nach einem der Aspekte F5, wo- bei die Abfragesyntax des standardisierten Abfrageformats nicht direkt aus- führbar ist, um den ersten Datensatz von der ersten elektronischen Datenquelle und/oder den zweiten Datensatz von der zweiten elektronischen Datenquelle abzurufen.

F7. Das System nach einem der Aspekte F1–F6, wobei die standardisierte Abfrage in einer JavaS- cript Object Notation(JSON)-Datei enthalten ist.

F8. Das System nach einem der Aspekte F1–F7, wobei die standardisierte Abfrage eine Angabe ei- nes Zeitbereichs beinhaltet, der eine Zeitspanne benennt, die Dateneinträgen im ersten Datensatz und/oder im zweiten Datensatz entspricht.

F9. Das System nach dem Aspekt F8, wobei der Zeitbereich eine Mehrzahl von Zeitspannen be-inhaltet.

F10. Das System nach dem Aspekt F9, wobei mindestens zwei von den mehreren Zeitspannen durch Ausschlusszeitspannen getrennt sind, in denen keine Daten abgefragt werden.

F11. Das System nach einem der Aspekte F9– F10, wobei der Zeitbereich eine erste, mit dem ersten Datensatz assoziierte Zeitspanne und/oder eine zweite, mit dem zweiten Datensatz assoziierte Zeitspanne benennt.

F12. Das System nach einem der Aspekte F1– F11, wobei die erste elektronische Datenquelle ei- ne relationale Datenquelle ist und/oder die zwei- te elektronische Datenquelle eine nicht-relational- e Datenquelle ist.

F13. Das System nach einem der Aspekte F1– F12, wobei die standardisierte Abfrage eine Anga- be zu einer Sampling-Rate für den kombinierten Datenrahmen beinhaltet.

F14. Das System nach einem der Aspekte F1– F13, wobei:

der erste Datensatz Daten mit einer ersten Sam- pling-Rate beinhaltet,

der zweite Datensatz Daten mit einer zweiten Sampling-Rate beinhaltet, und/oder

das Generieren des kombinierten Datenrahmens das Abgleichen der ersten und der zweiten Sam- pling-Rate beinhaltet.

F15. Das System nach dem Aspekt F14, wobei das Abgleichen der ersten und der zweiten Sam- pling-Rate das Hinzufügen von Datenpunkten be-inhaltet, die Zeiten ohne Abtastung entsprechen, die nötig sind, um eine gewünschte Sampling-Ra- te zu erreichen.

F16. Das System nach dem Aspekt F15, wobei die hinzugefügten Datenpunkte gegebene Werte der abgetasteten Daten sind, die den hinzugefügten Datenpunkten zeitlich direkt vorangehen.

F17. Das System nach dem Aspekt F14, wobei das Abgleichen der ersten und der zweiten Sampling-Rate das Entfernen von abgetasteten Datenpunkten beinhaltet.

F18. Das System nach einem der Aspekte F1–F15, wobei der erste und der zweite Datensatz jeweils mehrere Datenpunkte beinhalten, die Information enthalten, die von einer oder mehreren Vorrichtungen vor Ort innerhalb einer Prozessanlage gemessen werden.

G1. Materielles, nichtflüchtiges computerlesbares Medium, das Anweisungen speichert, um aus einer elektronischen Datenquelle Zeitreihendaten abzurufen, die mit dem Betrieb einer Prozessanlage assoziiert sind und die, wenn sie von einem Prozessor eines Computersystems ausgeführt werden, bewirken, dass das Computersystem:

eine Angabe zu der elektronischen Datenquelle empfängt, aus der die Daten abgerufen werden sollen;

eine Angabe über einen Zeitbereich empfängt, in dem die Daten abgerufen werden sollen, wobei die Angabe in einem standardisierten Abfrageformat ausgedrückt wird, das unabhängig ist von der elektronischen Datenquelle;

eine Angabe über ein Merkmal der Daten empfängt, die abgerufen werden sollen, wobei die Angabe in dem standardisierten Abfrageformat ausgedrückt wird, das unabhängig ist von der elektronischen Datenquelle;

ein datenquellspezifisches Abfrageformat bestimmt, das mit der elektronischen Datenquelle assoziiert ist;

eine datenquellspezifische Abfrage, die mit dem datenquellspezifischen Format kompatibel ist, auf Basis der Angaben zum Zeitbereich und dem Merkmal der Daten, die abgerufen werden sollen, generiert;

bewirkt, dass die datenquellspezifische Abfrage ausgeführt wird, um Daten aus der elektronischen Datenquelle auszuwählen; und/oder die Daten von der elektronischen Datenquelle empfängt.

G2. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt G1, wobei die Datenquelle eine relationale Datenquelle oder eine nicht-relationale Datenquelle ist.

G3. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1 oder G2, wobei die Angabe zum Zeitbereich und/oder die Angabe zum Merkmal in einer standardisierten Abfrage empfangen werden, die das standardisierte Abfrageformat nutzt.

G4. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt G3, wobei die Angabe zu der elektronischen Datenquelle auch in der standardisierten Abfrage empfangen wird.

G5. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G3 oder G4,

wobei die standardisierte Abfrage ferner ein Format für die Daten angibt.

G6. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G3–G5, wobei die standardisierte Abfrage eine JavaScript Object Notation(JSON)-Datei ist.

G7. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G6, das ferner Anweisungen speichert, die bewirken, dass der Prozessor eine Angabe zu einem Format empfängt, in dem die Daten aus der elektronischen Datenquelle empfangen werden sollen, und/oder wobei die Daten in dem angegebenen Format aus der elektronischen Datenquelle empfangen werden.

G8. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G6, das ferner Anweisungen speichert, die den Prozessor dazu veranlassen:

eine Angabe zu einem Format zu empfangen, in dem die Daten abgerufen werden sollen, und/oder die aus der elektronischen Datenquelle abgerufenen Daten in das angegebene Format umzuwandeln.

G9. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G8, wobei das standardisierte Abfrageformat eine Syntax verwendet, die sich von einer nativen Syntax, die von der elektronischen Datenquelle genutzt wird, unterscheidet.

G10. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G9, wobei das standardisierte Abfrageformat eine Syntax verwendet, die nicht direkt ausführbar ist, um die Daten aus der elektronischen Datenquelle abzurufen.

G11. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G10, wobei die Anweisungen, die bewirken, dass der Prozessor die datenquellspezifische Abfrage generiert, bewirken, dass der Prozessor den angegebenen Zeitbereich und/oder die angegebenen Merkmale der Daten, die abgerufen werden sollen, auf das bestimmte datenquellspezifische Abfrageformat abbildet.

G12. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G11, wobei das datenquellspezifische Abfrageformat auf Basis eines Typs der elektronischen Datenquelle bestimmt wird.

G13. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G12, wobei die Anweisungen, die bewirken, dass der Prozessor bewirkt, dass die datenquellspezifische Abfrage ausgeführt wird, bewirken, dass der Prozessor die datenquellspezifische Abfrage zu einer Execution Engine der elektronischen Datenquelle schickt, wobei die Execution Engine die elektronische Datenquelle unter Verwendung der datenquellspezifischen Abfrage befragt.

G14. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G13, wobei die Angabe des Zeitbereichs mehrere Startzeiten und/oder mehrere Endzeiten beinhaltet.

G15. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G14, wobei die Angabe des Zeitbereichs eine Sampling-Rate für die Daten, die abgerufen werden sollen, beinhaltet.

G16. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G15, das ferner Anweisungen speichert, die bewirken, dass der Prozessor einen Datenrahmen generiert, der die Daten beinhaltet, die von der elektronischen Datenquelle empfangen werden, wobei der Datenrahmen mehrere Datenpunkte beinhaltet, wobei jeder Datenpunkt mit einer Zeit innerhalb des Zeitrahmens assoziiert ist.

G17. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt G16, wobei jeder Datenpunkt mit einer Zeit innerhalb des Zeitbereichs assoziiert ist, die mit einem ganzzahligen Vielfachen einer Sampling-Periode, die auf einen Startzeitpunkt folgt, abgeglichen ist, und/oder der Startzeitpunkt ist.

G18. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G17, wobei die Angabe zu den Merkmalen der Daten, die abgerufen werden sollen, eine Angabe zu einem oder mehreren Messungstypen in Bezug auf den Betrieb der Prozessanlage beinhaltet.

G19. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt G18, ferner eine oder mehrere Vorrichtungen vor Ort umfassend, die innerhalb der Prozessanlage angeordnet ist bzw. sind, wobei der eine oder die mehreren Messungstypen Messungen von der einen oder von den mehreren Vorrichtung vor Ort beinhalten.

G20. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G18 oder G19, wobei die Angabe zu dem einen oder den mehreren Messungstypen ein oder mehrere Tags, Aliases und/oder Datentypen beinhaltet, die mit den Daten assoziiert sind.

G21. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte G1–G20, wobei die elektronische Datenquelle ein Massendatenspeicherknoten ist.

H1. Ein materielles, nichtflüchtiges computerlesbares Medium zur Bereitstellung von Daten, die mit dem Betrieb einer Prozessanlage assoziiert sind, aus einer elektronischen Datenquelle, die, wenn sie von einem Prozessor eines Computersystems ausgeführt werden, bewirken, dass das Computersystem:

eine standardisierte Abfrage in einem standardisierten Abfrageformat von einer Datenabfrageinstanz empfängt, wobei das standardisierte Abfrageformat von einem datenquellspezifischen

Abfrageformat, das von der elektronischen Datenquelle genutzt wird, verschieden ist; Abfrageparameter aus der standardisierten Abfrage extrahiert; eine datenquellspezifische Abfrage in dem datenquellspezifischen Abfrageformat auf Basis der extrahierten Abfrageparameter generiert; die datenquellspezifische Abfrage ausführt, um die Daten aus der elektronischen Datenquelle abzurufen; und/oder die abgerufenen Daten an einer Datenempfangsinstanz bereitstellt.

H2. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt H1, wobei die Datenabfrageinstanz einen Abfrageblock innerhalb eines Analyseprogramms umfasst.

H3. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H1 oder H2, wobei die elektronische Datenquelle eine nicht-relationale Datenquelle ist.

H4. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H1–H3, wobei die standardisierte Abfrage eine Syntax verwendet, die nicht direkt ausführbar ist, um die Daten aus der elektronischen Datenquelle abzurufen.

H5. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H1–H4, wobei die Angaben, die bewirken, dass der Prozessor die Abfrageparameter extrahiert, bewirken, dass der Prozessor eine Zeitspanne und/oder einen Datenparameter der Daten bestimmt, wobei die Zeitspanne eine Startzeit und/oder eine Endzeit und/oder den Datenparameter beinhaltet, der einen Typ von Daten angibt, die aus der elektronischen Datenquelle abgerufen werden sollen.

H6. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt H5, wobei das Extrahieren der Abfrageparameter ferner das Bestimmen einer Sampling-Rate beinhaltet.

H7. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt H6, wobei die Anweisungen, die bewirken, dass der Prozessor abgerufene Daten an der Abfrageinstanz bereitstellt, bewirken, dass der Prozessor mit einem Satz von Datenpunkten, die mit der Zeitspanne assoziiert sind, antwortet, wobei jeder Datenpunkt einen Zeitstempel beinhaltet, der die Startzeit und/oder ein ganzzahliges Vielfaches der Spanne der Sampling-Rate im Anschluss an die Startzeit ist.

H8. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt H7, wobei ein oder mehrere Datenpunkte mit Zeitstempeln, die ganzzahlige Vielfache der Zeitspanne der Sampling-Rate nach der Startzeit angeben, unter Verwendung von Werten von Dateneinträgen in die elektronische Datenquelle, die mit Zeiten assoziiert sind, die den einzelnen Zeitstempeln des einen oder der mehreren Datenpunkte mit dem kürzesten Abstand vorangehen, zu dem Datensatz hinzugefügt werden.

H9. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H5–H8, wobei der Satz von Datenpunkten auf Basis eines Formats formatiert wird, das in der standardisierten Abfrage angegeben ist.

H10. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H5–H9, wobei der Typ, der von dem Datenparameter angegeben wird, ein Messungstyp und/oder eine Messung von einem Messvorrichtungstyp ist.

H11. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt H10, wobei der Typ ferner eine bestimmte Messvorrichtung angibt.

H12. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H11 oder H12, wobei die Messvorrichtung eine Vorrichtung vor Ort ist, die innerhalb der Prozessanlage angeordnet ist.

H13. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H5–H12, wobei der Datenparameter ferner ein Tag der Daten, die abgerufen werden sollen, ein Alias für die Daten, die an der Datenabfrageinstanz bereitgestellt werden sollen, und/oder einen Datenformatotyp für die Daten, die an der Datenabfrageinstanz bereitgestellt werden sollen, angibt.

H14. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H1–H13, wobei die Angaben, die bewirken, dass der Prozessor die Daten an der Datenempfangsinstanz bereitstellt, bewirken, dass der Prozessor einen Datenrahmen, der die abgerufenen Daten enthält, an die Datenempfangsinstanz schickt.

H15. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H1–H14, wobei die standardisierte Abfrage eine Angabe eines Formats für den Datenrahmen beinhaltet, und/oder wobei der Datenrahmen gemäß der Angabe des Formats für den Datenrahmen formatiert wird.

H16. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H1–H15, wobei die standardisierte Abfrage angibt, dass der Datenrahmen als JavaScript Object Notation (JSON)-Datei bereitgestellt werden soll.

H17. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte H1–H15, wobei die Datenempfangsinstanz dieselbe Instanz ist wie die Datenabfrageinstanz.

I1. Ein materielles, nichtflüchtiges computerlesbares Medium zum Zugreifen auf Prozessanlagendaten, die mit dem Betrieb einer Prozessanlage assoziiert sind und die, wenn sie von einem Prozessor eines Computersystems ausgeführt werden, bewirken, dass das Computersystem:
eine standardisierte Abfrage empfängt, die ein standardisiertes Abfrageformat nutzt;
eine erste datenquellspezifische Abfrage auf Basis der standardisierten Abfrage generiert, wo-

bei die erste datenquellspezifische Abfrage ein erstes Abfrageformat nutzt, das mit einer ersten elektronischen Datenquelle assoziiert ist;

eine zweite datenquellspezifische Abfrage auf Basis der standardisierten Abfrage generiert, wobei die zweite datenquellspezifische Abfrage ein zweites Abfrageformat nutzt, das mit einer zweiten elektronischen Datenquelle assoziiert ist; bewirkt, dass die erste datenquellspezifische Abfrage ausgeführt wird, um einen ersten Datensatz von der ersten elektronischen Datenquelle abzurufen;

bewirkt, dass die zweite datenquellspezifische Abfrage ausgeführt wird, um einen zweiten Datensatz von der zweiten elektronischen Datenquelle abzurufen; und/oder

einen kombinierten Datenrahmen generiert, der den ersten und den zweiten Datensatz beinhaltet.

I2. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt I1, wobei die standardisierte Abfrage eine erste Angabe zu der ersten elektronischen Datenquelle und/oder eine zweite Angabe zu der zweiten elektronischen Datenquelle beinhaltet.

I3. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt I2, wobei die erste Angabe das erste Abfrageformat benennt und/oder die zweite Angabe das zweite Abfrageformat benennt.

I4. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I2 oder I3, wobei die erste und/oder die zweite Angabe Angaben zu Datenspalten vorangestellt werden.

I5. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I1–I4, wobei das standardisierte Abfrageformat eine Abfragesyntax nutzt, die sich von einer Abfragesyntax des ersten Abfrageformats und/oder einer Abfragesyntax des zweiten Abfrageformats unterscheidet.

I6. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I5, wobei die Abfragesyntax des standardisierten Abfrageformats nicht direkt ausführbar ist, um den ersten Datensatz von der ersten elektronischen Datenquelle und/oder den zweiten Datensatz von der zweiten elektronischen Datenquelle abzurufen.

I7. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I1–I6, wobei die standardisierte Abfrage in einer JavaScript Object Notation(JSON)-Datei enthalten ist.

I8. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I1–I7, wobei die standardisierte Abfrage eine Angabe eines Zeitbereichs beinhaltet, der eine Zeitspanne benennt, die Dateneinträgen im ersten Datensatz und/oder im zweiten Datensatz entspricht.

I9. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt I8, wobei der Zeitbereich eine Mehrzahl von Zeitspannen beinhaltet.

I10. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt I9, wobei mindestens zwei von den mehreren Zeitspannen durch Ausschlusszeitspannen getrennt sind, in denen keine Daten abgefragt werden.

I11. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I9–I10, wobei der Zeitbereich eine erste, mit dem ersten Datensatz assoziierte Zeitspanne und/oder eine zweite, mit dem zweiten Datensatz assoziierte Zeitspanne benennt.

I12. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I1–I11, wobei die erste elektronische Datenquelle eine relationale Datenquelle ist und/oder die zweite elektronische Datenquelle eine nicht-relationale Datenquelle ist.

I13. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I1–I12, wobei die standardisierte Abfrage eine Angabe zu einer Sampling-Rate für den kombinierten Datenrahmen beinhaltet.

I14. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I1–I13, wobei: der erste Datensatz Daten mit einer ersten Sampling-Rate beinhaltet,
der zweite Datensatz Daten mit einer zweiten Sampling-Rate beinhaltet, und/oder
das Generieren des kombinierten Datenrahmens
das Abgleichen der ersten und der zweiten Sampling-Rate beinhaltet.

I15. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt I14, wobei das Abgleichen der ersten und der zweiten Sampling-Rate das Hinzufügen von Datenpunkten beinhaltet, die Zeiten ohne Abtastung entsprechen, die nötig sind, um eine gewünschte Sampling-Rate zu erreichen.

I16. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt I15, wobei die hinzugefügten Datenpunkte gegebene Werte der abgetasteten Daten sind, die den hinzugefügten Datenpunkten zeitlich direkt vorangehen.

I17. Das materielle, nichtflüchtige computerlesbare Medium nach dem Aspekt I, wobei das Abgleichen der ersten und der zweiten Sampling-Rate das Entfernen von abgetasteten Datenpunkten beinhaltet.

I18. Das materielle, nichtflüchtige computerlesbare Medium nach einem der Aspekte I1–I15, wobei der erste und der zweite Datensatz jeweils mehrere Datenpunkte beinhalten, die Information enthalten, die von einer oder mehreren Vorrichtungen vor Ort innerhalb einer Prozessanlage gemessen werden.

J1. Irgendeiner der Aspekte A1–A20, B1–B17, C1–C18, D1–D21, E1–E17, F1–F18, G1–G21, H1–H17 oder I1–I18, wobei ein Teil oder alle von den Aktionen und/oder Schritte von einer oder von mehreren verteilten Data Engines (DDEs) durch-

geführt wird bzw. werden, die in einem verteilten Prozessüberwachungs- und Analysesystem für industrielle Anlagen angeordnet ist bzw. sind, wobei das verteilte Prozessüberwachungs- und Analysesystem für industrielle Anlagen umfasst: eine Mehrzahl von DDEs, die in eine Prozessanlage eingebettet sind, die dazu dient, einen Prozess zu steuern, wobei jede von den DDEs mit einer oder mehreren betreffenden Datenquellen gekoppelt ist, die als Ergebnis davon, dass der Prozess von der Prozessanlage gesteuert wird, Daten generiert, und wobei jede von den DDEs die Daten, die von der einen oder den mehreren betreffenden Datenquellen generiert werden, speichert; und/oder

ein Datenanalysenetz, das ein Streamen von Analysedaten zwischen der Mehrzahl von Distributed Data Engines und/oder eine Übertragung von Abfragen von Daten, die an der Mehrzahl von DDEs gespeichert sind, unterstützt.

J2. Irgendeiner der Aspekte A1–A20, B1–B17, C1–C18, D1–D21, E1–E17, F1–F18, G1–G21, H1–H17, I1–I18 oder J1, wobei ein Teil oder alle von den verwendeten Daten, Information und/oder Signalen aus einer oder aus mehreren verteilten Data Engines (DDEs) abgerufen wird bzw. werden, die in einem verteilten Prozessüberwachungs- und Analysesystem für industrielle Anlagen angeordnet ist bzw. sind, wobei das verteilte Prozessüberwachungs- und Analysesystem für industrielle Anlagen umfasst:

eine Mehrzahl von DDEs, die in eine Prozessanlage eingebettet sind, die dazu dient, einen Prozess zu steuern, wobei jede von den DDEs mit einer oder mehreren betreffenden Datenquellen gekoppelt ist, die als Ergebnis davon, dass der Prozess von der Prozessanlage gesteuert wird, Daten generiert, und wobei jede von den DDEs die Daten, die von der einen oder den mehreren betreffenden Datenquellen generiert werden, speichert; und/oder

ein Datenanalysenetz, das ein Streamen von Analysedaten zwischen der Mehrzahl von Distributed Data Engines und/oder eine Übertragung von Abfragen von Daten, die an der Mehrzahl von DDEs gespeichert sind, unterstützt.

J3. Der Aspekt J1 oder der Aspekt J2, wobei die eine oder die mehreren DDEs mit Prozesssteuerungen innerhalb der Prozessanlage assoziiert sind, und/oder wobei die eine oder die mehreren Datenquellen mit Vorrichtung vor Ort innerhalb der Prozessanlage assoziiert sind.

J4. Irgendeiner der vorangehenden Aspekte in Verbindung mit irgendeinem oder irgendwelchen anderen von den vorangehenden Aspekten.

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 Nicht-Patentliteratur

- IEEE 802.11 [0089]
- IEEE 802.11 [0100]

Patentansprüche

1. System zum Zugreifen auf Prozessanlagendaten, die mit dem Betrieb einer Prozessanlage assoziiert sind, aus einer Mehrzahl elektronischer Datenquellen, umfassend:

eine erste elektronische Datenquelle und eine zweite elektronische Datenquelle, die jeweils ein materielles, nichtflüchtiges Speichermedium umfassen, das die Prozessanlagendaten speichert, die mit dem Betrieb der Prozessanlage assoziiert sind;

einen Prozessor, der kommunikationstechnisch mit der ersten und der zweiten elektronischen Datenquelle gekoppelt ist;

einen Programmspeicher, der kommunikationstechnisch mit dem Prozessor verbunden ist und der ein materielles, nichtflüchtiges Speichermedium umfasst, das Anweisungen speichert, die, wenn sie vom Prozessor ausgeführt werden, den Prozessor zu folgendem veranlassen:

Empfangen einer standardisierten Abfrage, die ein standardisiertes Abfrageformat nutzt;

Generieren einer ersten datenquellspezifischen Abfrage auf Basis der standardisierten Abfrage, wobei die erste datenquellspezifische Abfrage ein erstes Abfrageformat nutzt, das mit der ersten elektronischen Datenquelle assoziiert ist;

Generieren einer zweiten datenquellspezifischen Abfrage auf Basis der standardisierten Abfrage, wobei die zweite datenquellspezifische Abfrage ein zweites Abfrageformat nutzt, das mit der zweiten elektronischen Datenquelle assoziiert ist;

Bewirken, dass die erste datenquellspezifische Abfrage ausgeführt wird, um einen ersten Datensatz von der ersten elektronischen Datenquelle abzurufen;

Bewirken, dass die zweite datenquellspezifische Abfrage ausgeführt wird, um einen zweiten Datensatz von der zweiten elektronischen Datenquelle abzurufen; und

Generieren eines kombinierten Datenrahmens, der den ersten und den zweiten Datensatz beinhaltet.

2. System nach Anspruch 1, wobei die standardisierte Abfrage eine erste Angabe zu der ersten elektronischen Datenquelle und eine zweite Angabe zu der zweiten elektronischen Datenquelle beinhaltet, wobei insbesondere die erste Angabe das erste Abfrageformat benennt und die zweite Angabe das zweite Abfrageformat benennt oder die erste und die zweite Angabe Angaben zu Datenspalten vorangestellt werden.

3. System nach Anspruch 1, wobei das standardisierte Abfrageformat eine Abfragesyntax nutzt, die sich sowohl von einer Abfragesyntax des ersten Abfrageformats als auch einer Abfragesyntax des zweiten Abfrageformats unterscheidet und insbesondere nicht direkt ausführbar ist, um den ersten Datensatz von der ersten elektronischen Datenquelle

oder den zweiten Datensatz von der zweiten elektronischen Datenquelle abzurufen.

4. System nach Anspruch 1, wobei die standardisierte Abfrage in einer JavaScript Object Notation (JSON)-Datei enthalten ist.

5. System nach Anspruch 1, wobei die standardisierte Abfrage eine Angabe eines Zeitbereichs beinhaltet, der eine Zeitspanne benennt, die Dateneinträgen im ersten Datensatz und im zweiten Datensatz entspricht,

wobei der Zeitbereich insbesondere eine Mehrzahl von Zeitspannen beinhaltet, und wobei mindestens zwei von den mehreren Zeitspannen durch Ausschlusszeitspannen getrennt sind, in denen keine Daten abgefragt werden, und wobei genauer gesagt der Zeitbereich eine erste, mit dem ersten Datensatz assoziierte Zeitspanne und eine zweite, mit dem zweiten Datensatz assoziierte Zeitspanne benennt.

6. System nach Anspruch 1, wobei die erste elektronische Datenquelle eine relationale Datenquelle ist und die zweite elektronische Datenquelle eine nicht-relationale Datenquelle ist.

7. System nach Anspruch 1, wobei die standardisierte Abfrage eine Angabe zu einer Abtastrate für den kombinierten Datenrahmen beinhaltet.

8. System nach Anspruch 1, wobei:
der erste Datensatz Daten mit einer ersten Abtastrate beinhaltet,
der zweite Datensatz Daten mit einer zweiten Abtastrate beinhaltet, und
das Generieren des kombinierten Datenrahmens das Abgleichen der ersten und der zweiten Abtastrate beinhaltet.

9. System nach Anspruch 8, wobei das Abgleichen der ersten und der zweiten Abtastrate das Hinzufügen von Datenpunkten beinhaltet, die Zeiten ohne Abtastung entsprechen, die nötig sind, um eine gewünschte Abtastrate zu erreichen, und wobei insbesondere

die hinzugefügten Datenpunkte gegebene Werte der abgetasteten Daten sind, die den hinzugefügten Datenpunkten zeitlich direkt vorangehen und/oder das Abgleichen der ersten und der zweiten Abtastrate das Entfernen von abgetasteten Datenpunkten beinhaltet.

10. System nach Anspruch 1, wobei der erste und der zweite Datensatz jeweils mehrere Datenpunkte beinhalten, die Information enthalten, die von einer oder mehreren Vorrichtungen vor Ort innerhalb einer Prozessanlage gemessen werden.

11. Materielles, nichtflüchtiges computerlesbares Medium, das Anweisungen speichert, um aus einer

elektronischen Datenquelle Zeitreihendaten abzurufen, die mit dem Betrieb einer Prozessanlage assoziiert sind und die, wenn sie von einem Prozessor eines Computersystems ausgeführt werden, bewirken, dass das Computersystem:
die Schritte nach einem der Ansprüche 1 bis 10 ausführt.

12. Verfahren zum Bereitstellen von Daten aus einer elektronischen Datenquelle, umfassend:
Empfangen einer standardisierten Abfrage in einem standardisierten Abfrageformat von einer Datenabfrageinstanz, wobei das standardisierte Abfrageformat von einem datenquellspezifischen Abfrageformat, das von der elektronischen Datenquelle genutzt wird, verschieden ist;
Extrahieren von Abfrageparametern aus der standardisierten Abfrage;
Generieren einer datenquellspezifischen Abfrage in dem datenquellspezifischen Abfrageformat auf Basis der extrahierten Abfrageparameter;
Ausführen der datenquellspezifischen Abfrage, um die Daten aus der elektronischen Datenquelle abzurufen; und
Bereitstellen der abgerufenen Daten an einer Datenempfangsinstanz.

13. Verfahren nach Anspruch 12, wobei die Datenabfrageinstanz einen Abfrageblock innerhalb eines Analyseprogramms umfasst.

14. Verfahren nach Anspruch 12, wobei die standardisierte Abfrage eine Syntax verwendet, die nicht direkt ausführbar ist, um die Daten aus der elektronischen Datenquelle abzurufen.

15. Verfahren nach Anspruch 12, wobei das Extrahieren der Abfrageparameter beinhaltet: Bestimmen (i) einer Zeitspanne und eines Datenparameters der Daten, wobei die Zeitspanne eine Startzeit und eine Endzeit des Datenparameters aufweist, der einen Typ von Daten angibt, die aus der elektronischen Datenquelle abgerufen werden sollen, und (ii) einer Abtastrate, wobei insbesondere der Typ, der von dem Datenparameter angegeben wird, ein Messungstyp oder eine Messung von einem Messvorrichtungstyp ist, wobei die Messvorrichtung eine Vor-Ort-Vorrichtung ist, die innerhalb einer Prozessanlage angeordnet ist oder der Datenparameter ferner ein Tag der Daten, die abgerufen werden sollen, ein Alias für die Daten, die an der Datenabfrageinstanz bereitgestellt werden sollen, und einen Datenformattyp für die Daten, die an der Datenabfrageinstanz bereitgestellt werden sollen, angibt.

16. Verfahren nach Anspruch 12, wobei das Bereitstellen der abgerufenen Daten an der Abfrageinstanz das Zurückschicken eines Satzes von Datenpunkten, die mit der Zeitspanne assoziiert sind, beinhaltet, wobei jeder Datenpunkt einen Zeitstempel beinhaltet,

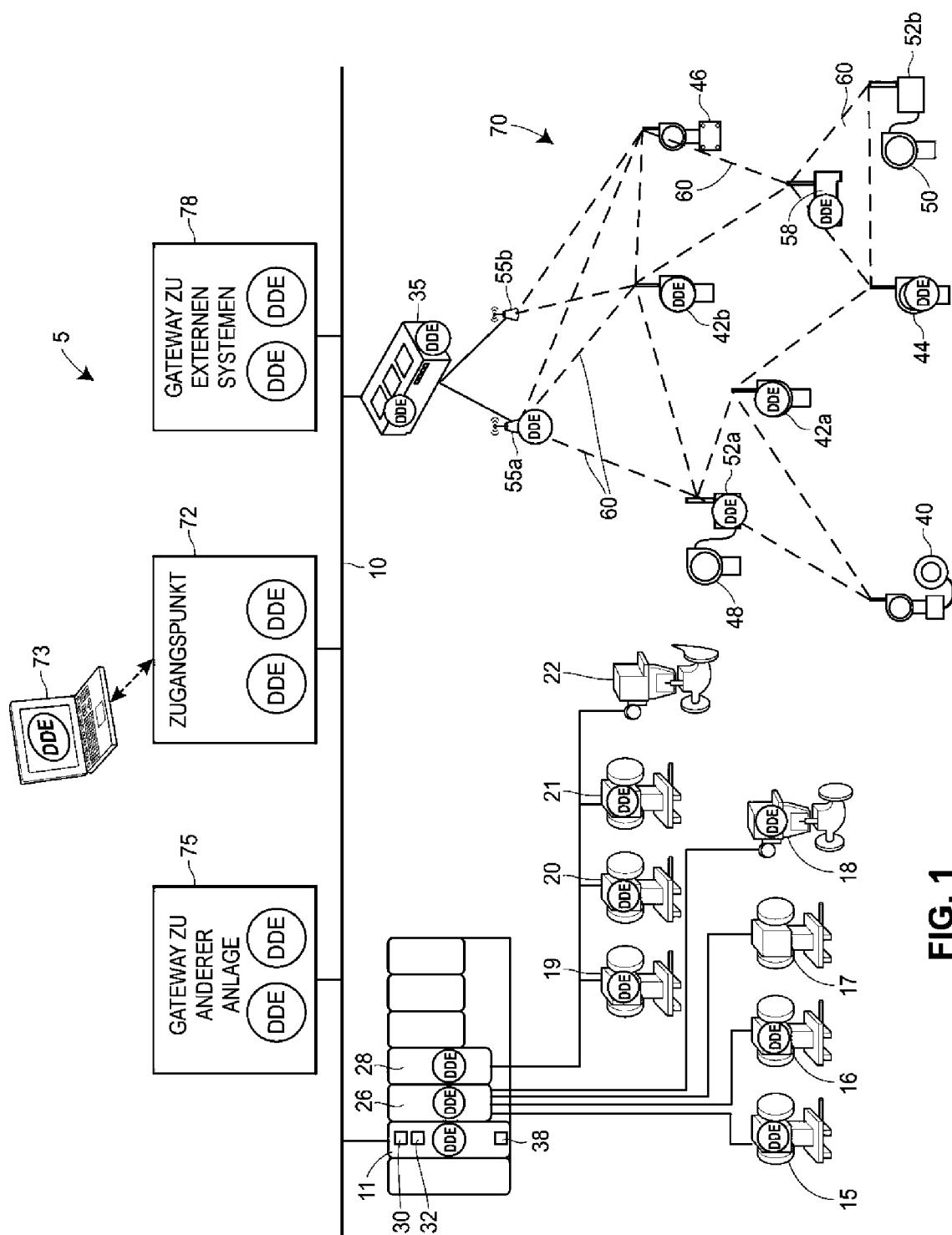
der die Startzeit oder ein ganzzahliges Vielfaches der Spanne der Abtastrate im Anschluss an die Startzeit ist, wobei insbesondere ein oder mehrere Datenpunkte mit Zeitstempeln, die ganzzahlige Vielfache der Zeitspanne der Abtastrate nach der Startzeit angeben, unter Verwendung von Werten von Dateneinträgen in die elektronische Datenquelle, die mit Zeiten assoziiert sind, die den einzelnen Zeitstempeln des einen oder der mehreren Datenpunkte mit dem kürzesten Abstand vorangehen, zu dem Datensatz hinzugefügt werden.

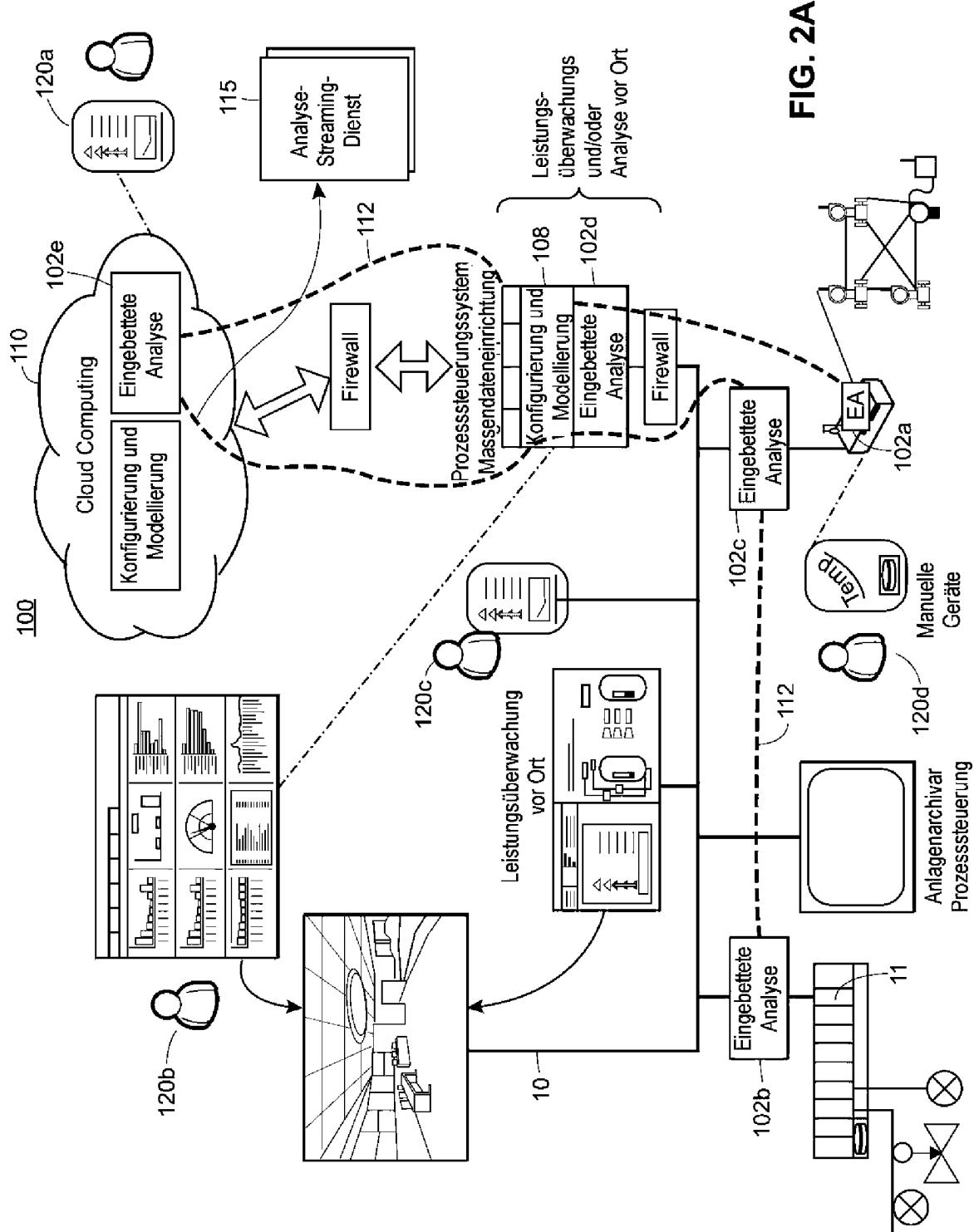
17. Verfahren nach Anspruch 12, wobei das Bereitstellen der Daten an der Datenempfangsinstanz das Senden eines Datenrahmens, der die abgerufenen Daten enthält, zur Datenempfangsinstanz umfasst, und wobei die standardisierte Abfrage eine Angabe eines Formats für den Datenrahmen beinhaltet, und wobei der Datenrahmen gemäß der Angabe des Formats für den Datenrahmen formatiert wird.

18. Verfahren nach Anspruch 12, wobei die Datenempfangsinstanz dieselbe Instanz ist wie die Datenabfrageinstanz.

Es folgen 35 Seiten Zeichnungen

Anhängende Zeichnungen

**FIG. 1**



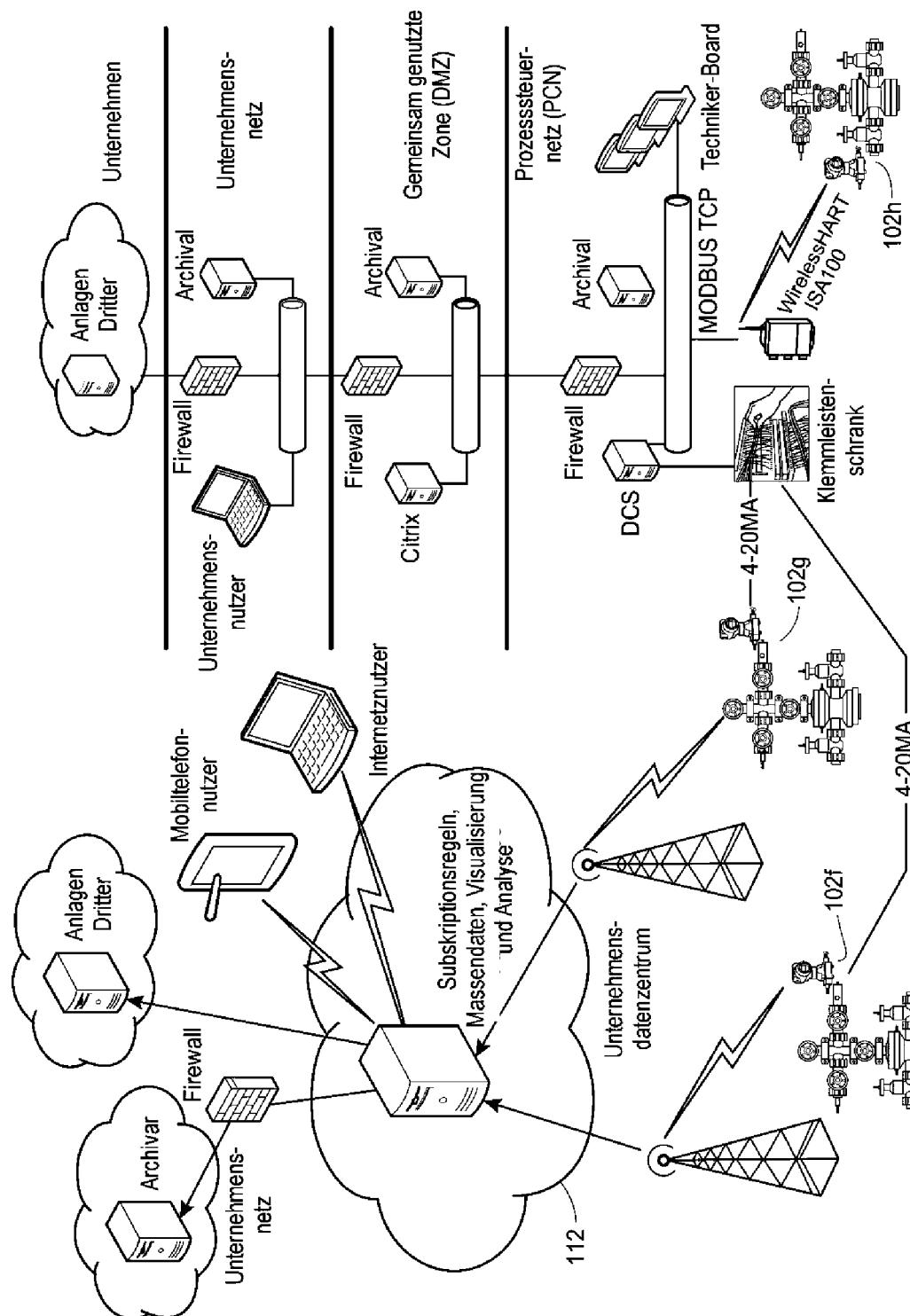


FIG. 2B

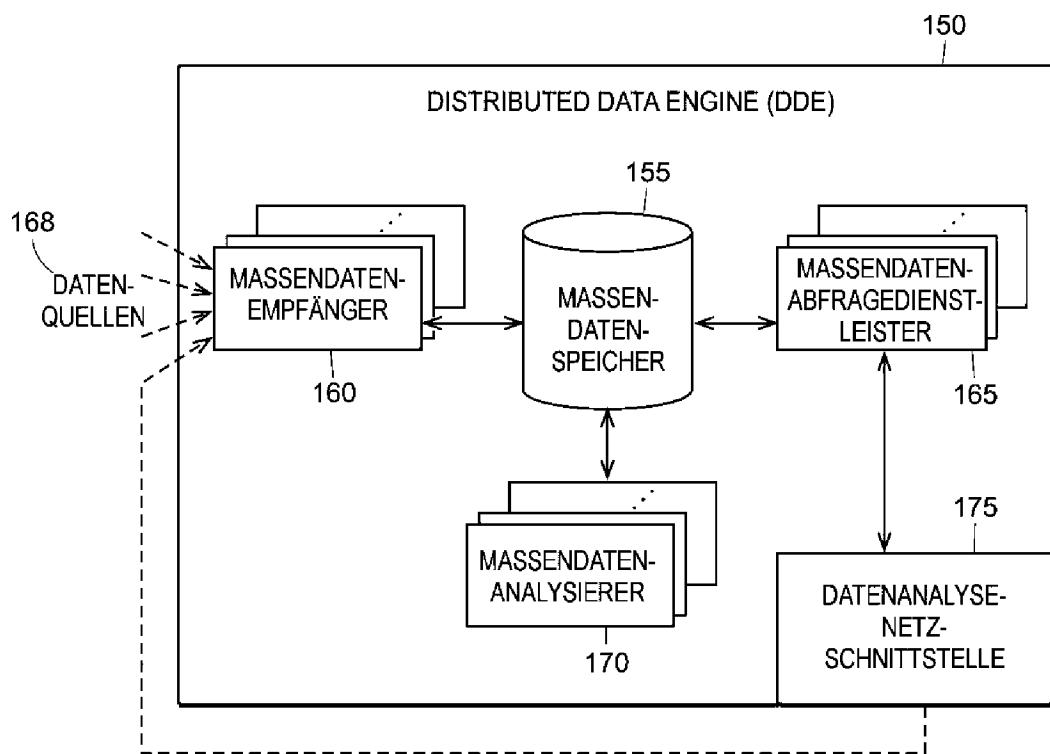


FIG. 3

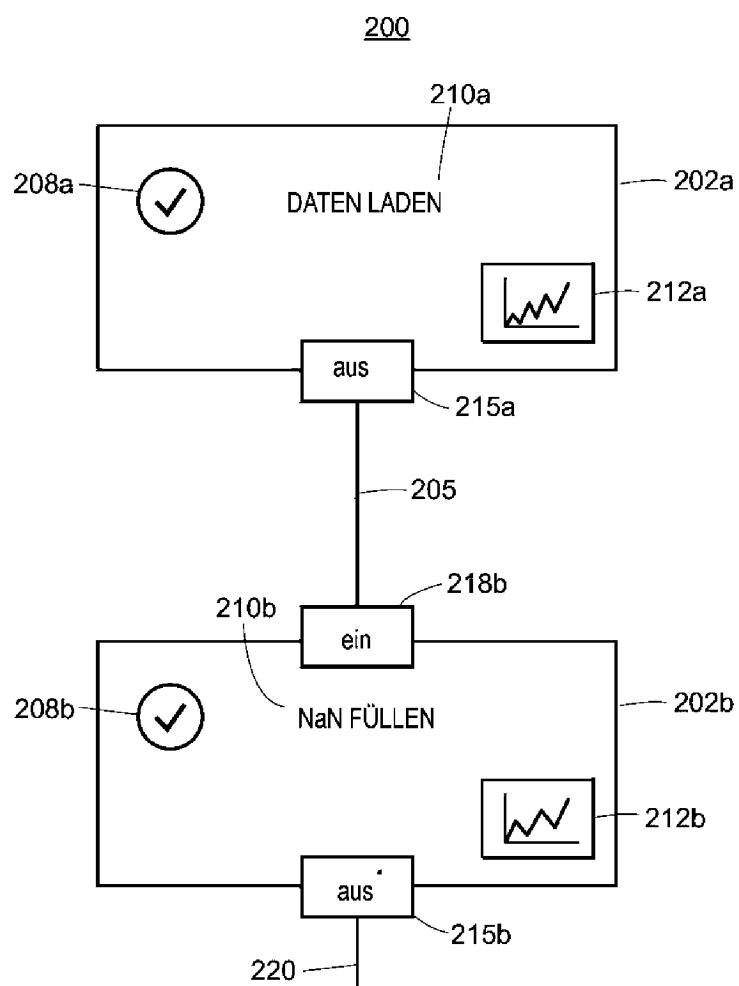


FIG. 4A

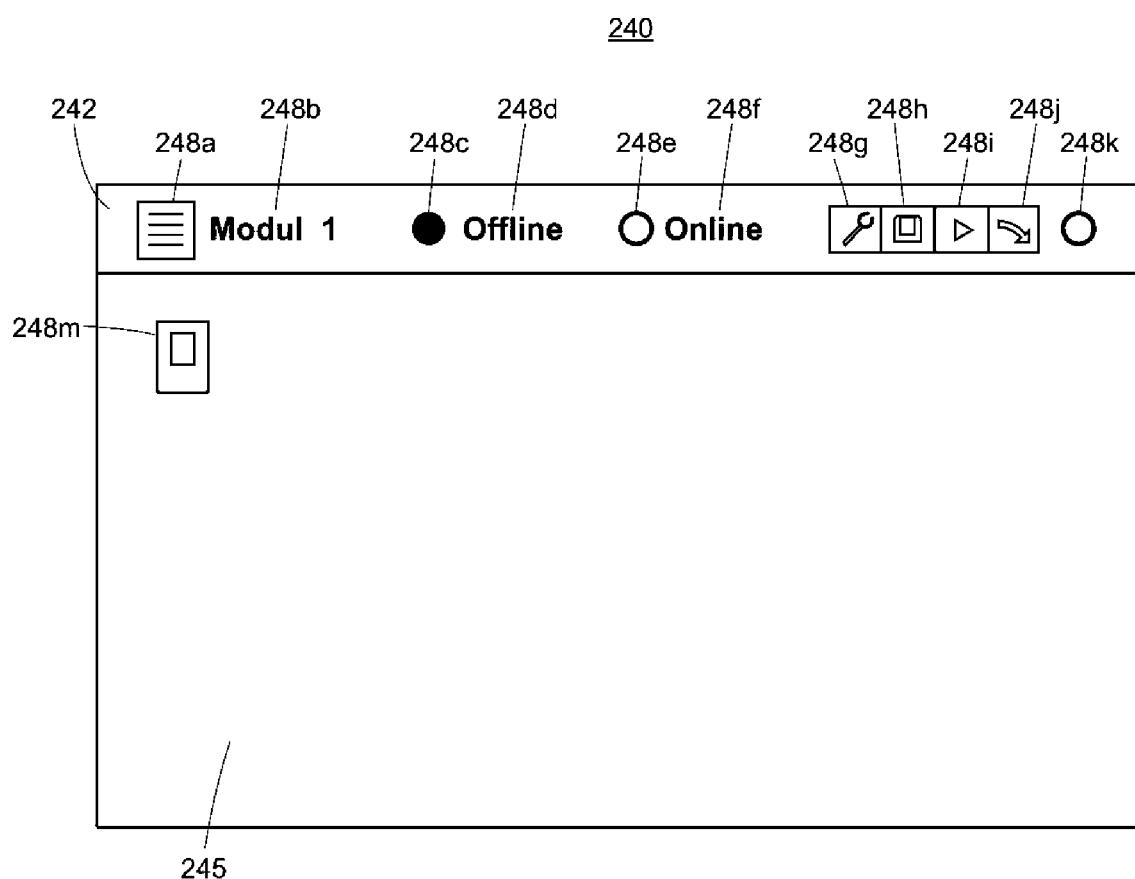
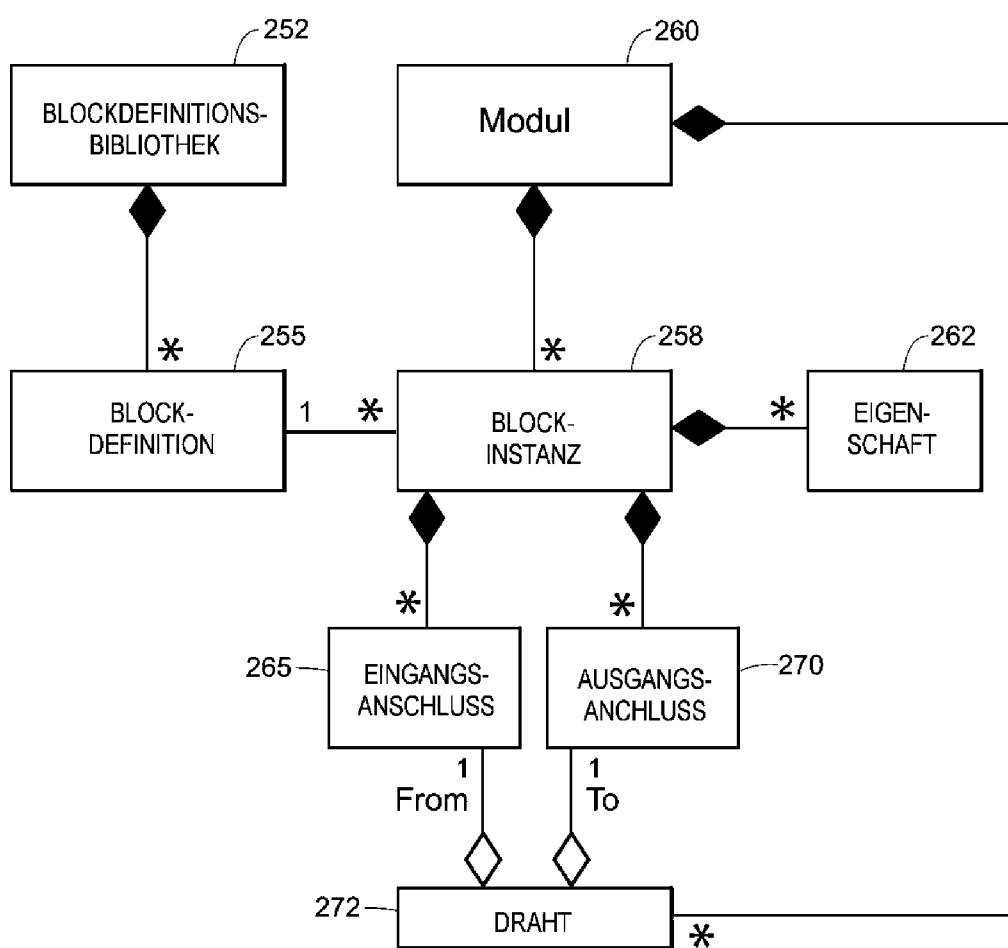


FIG. 4B

250**FIG. 4C**

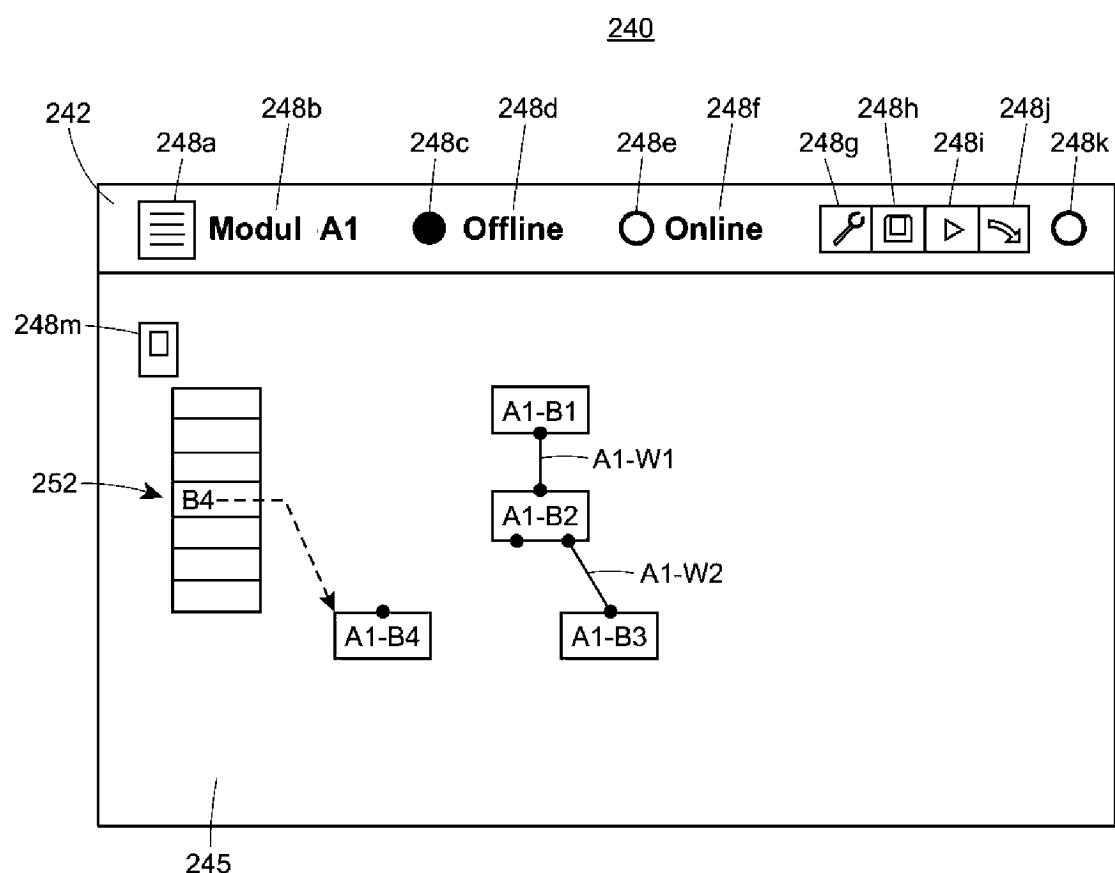


FIG. 4D

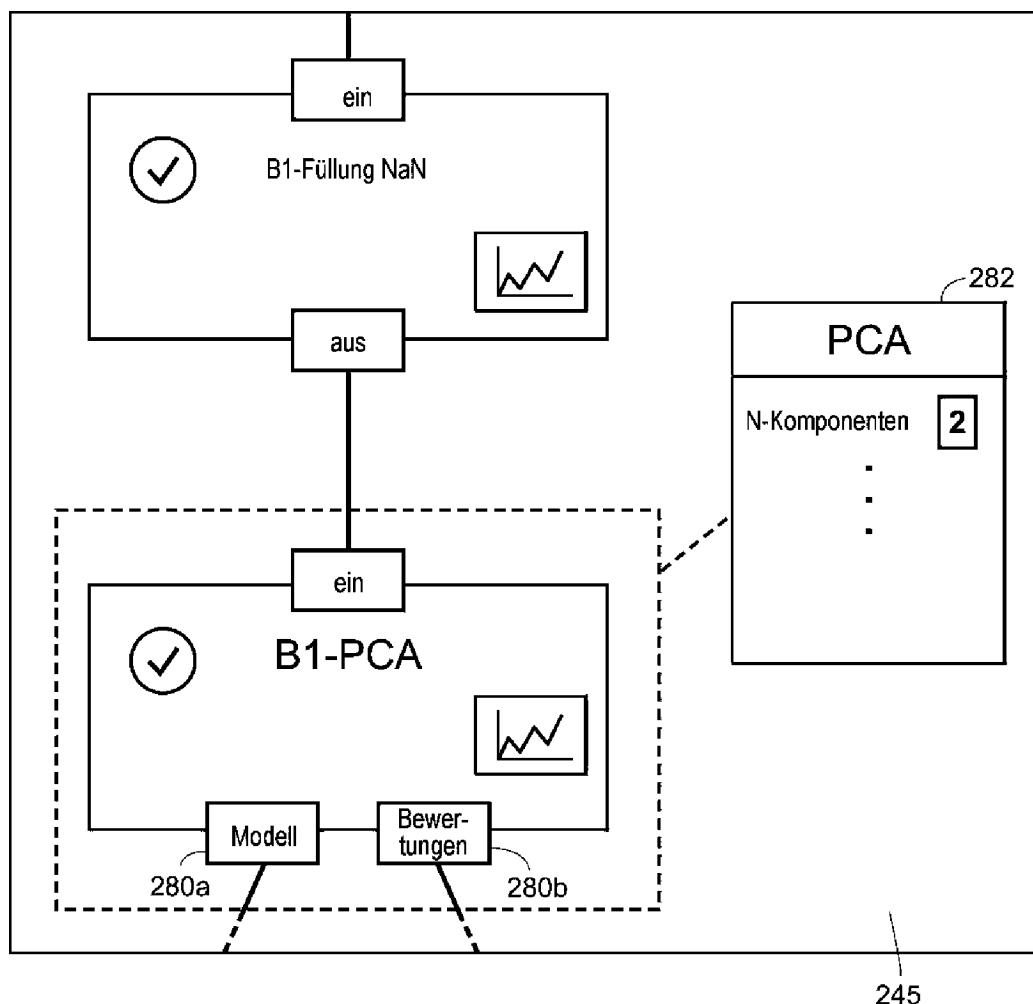


FIG. 4E

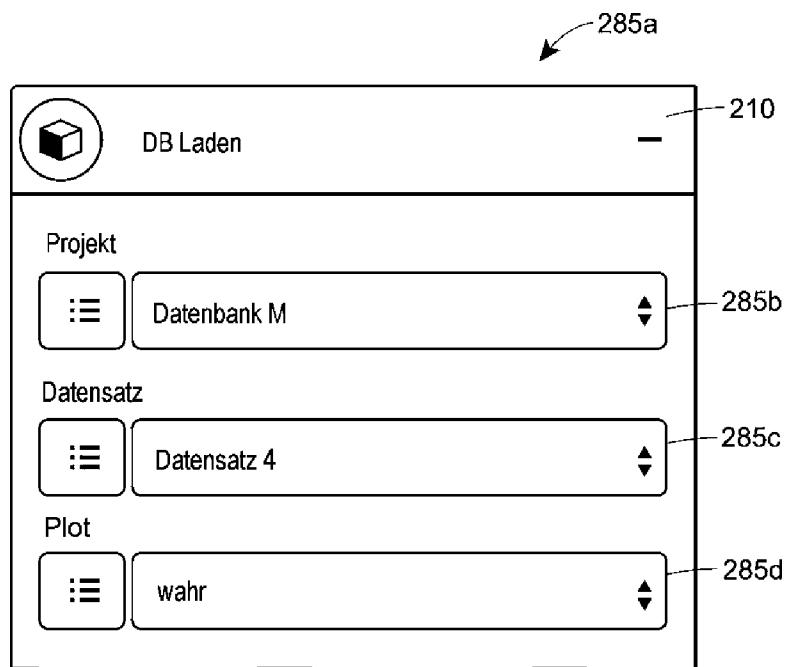


FIG. 4F

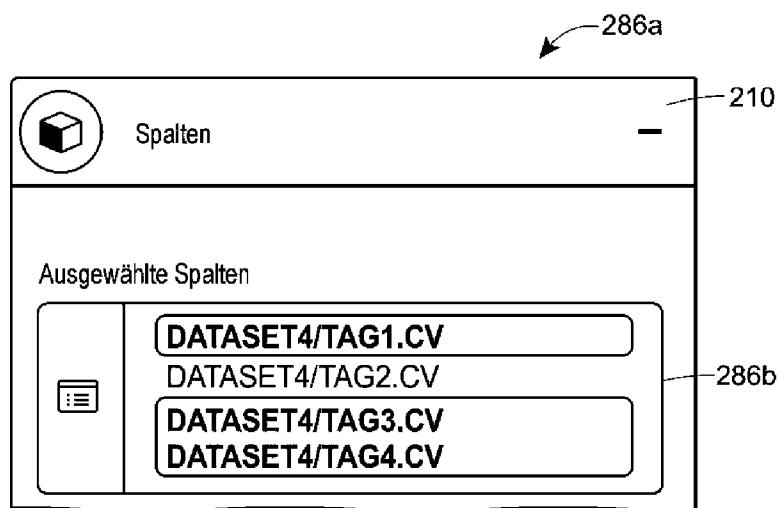


FIG. 4G

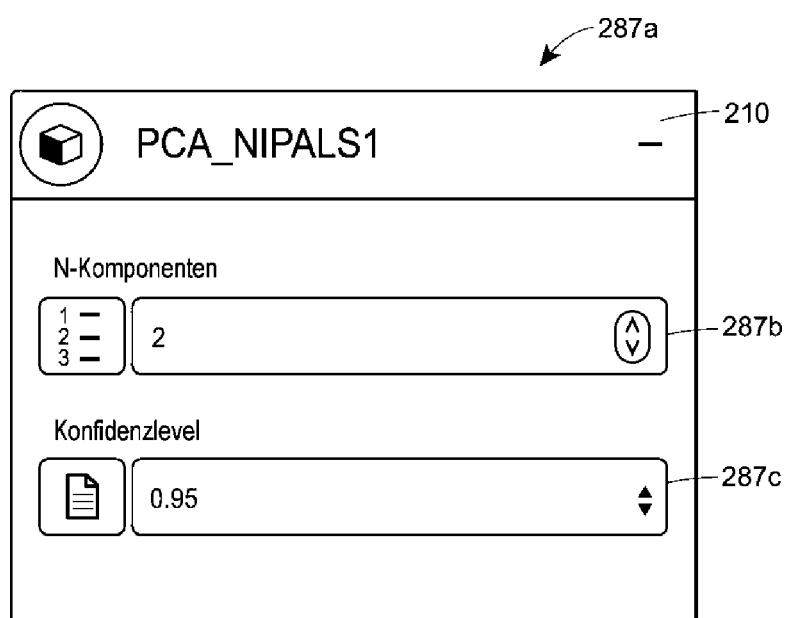


FIG. 4H

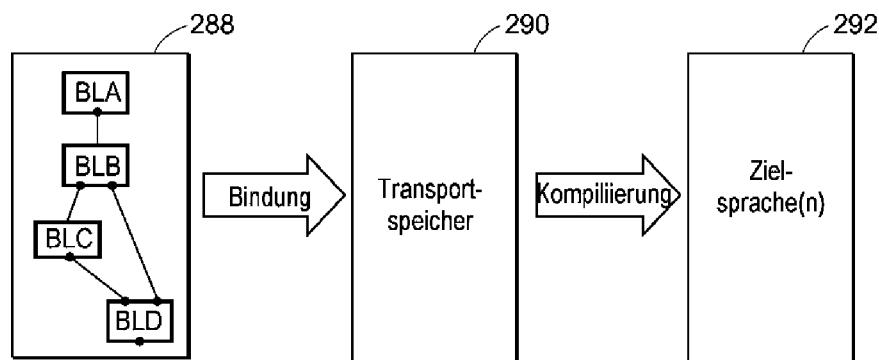


FIG. 4I

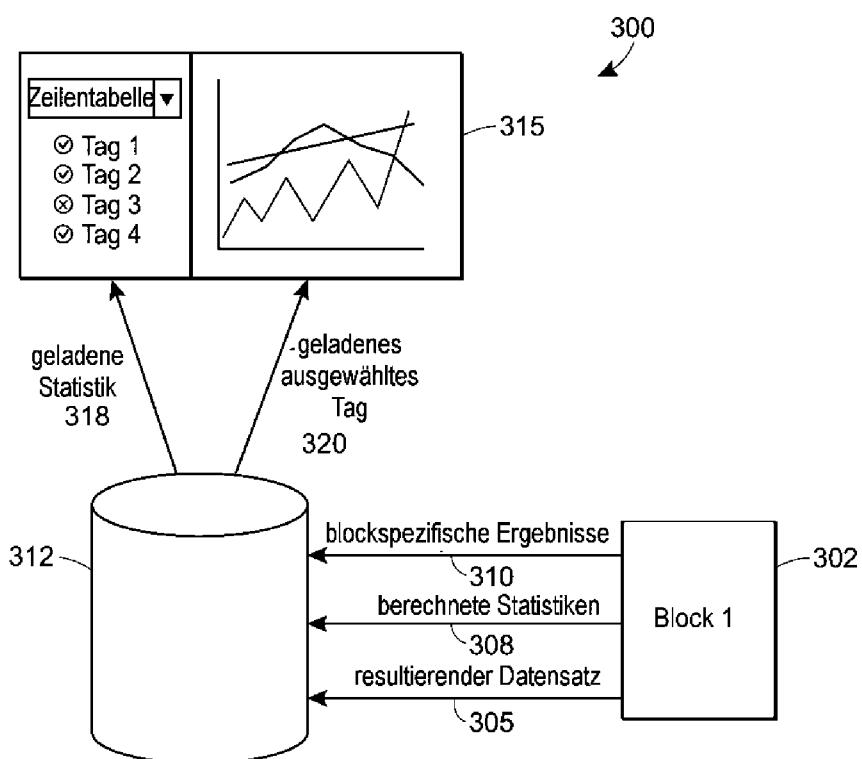


FIG. 4J

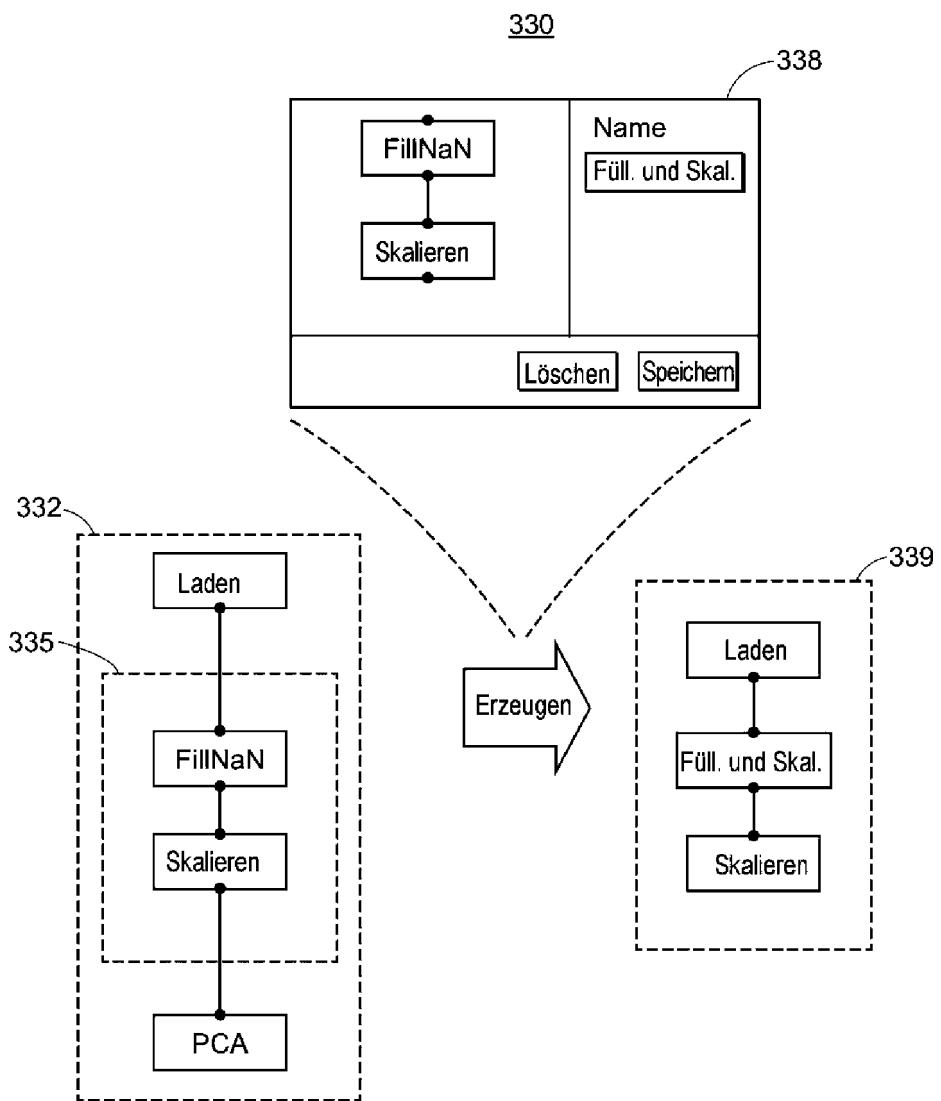


FIG. 4K

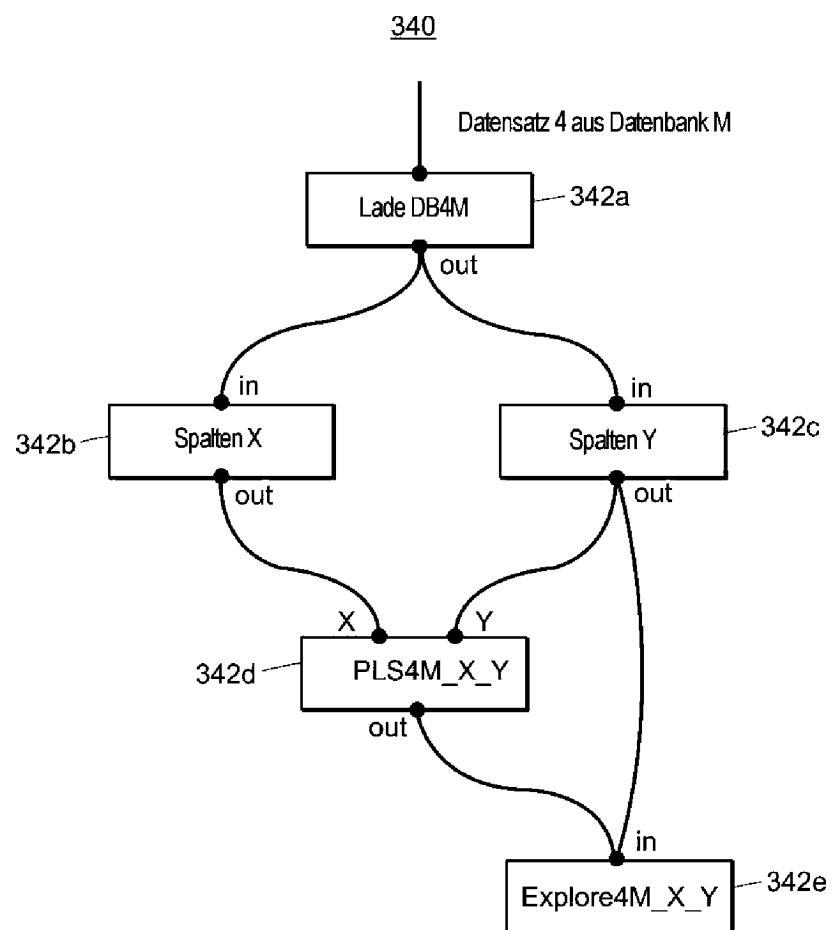


FIG. 4L

345

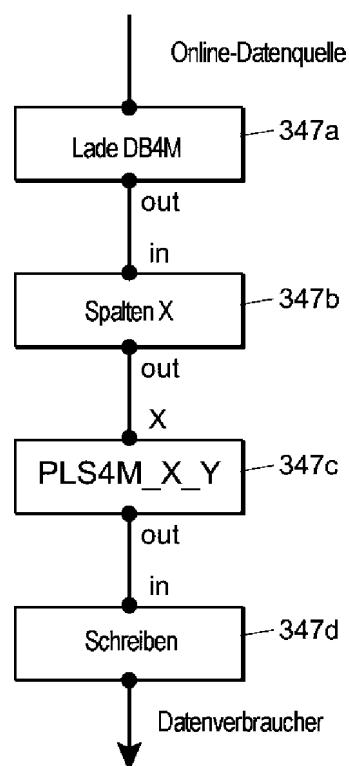


FIG. 4M

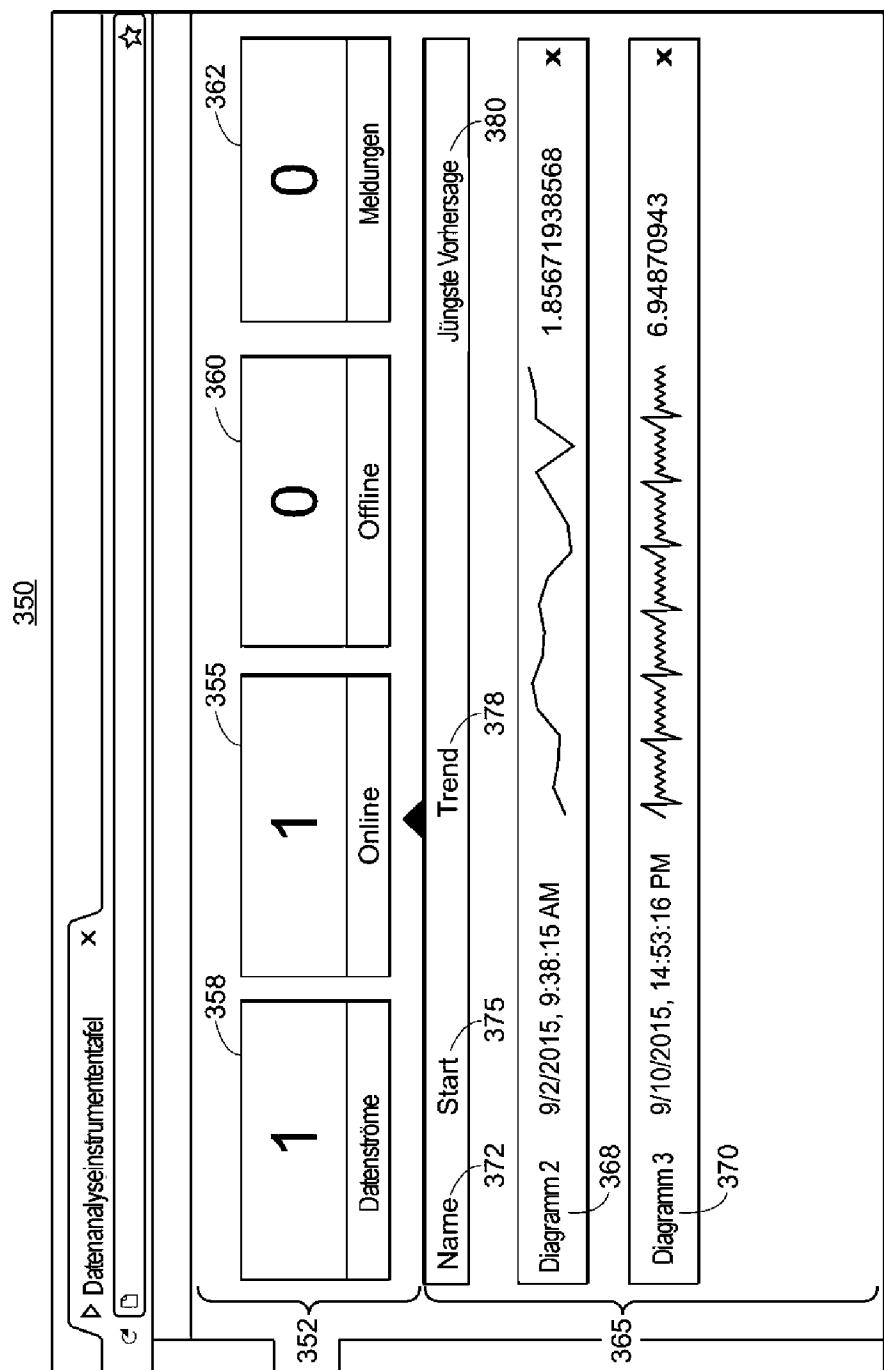


FIG. 4N-1

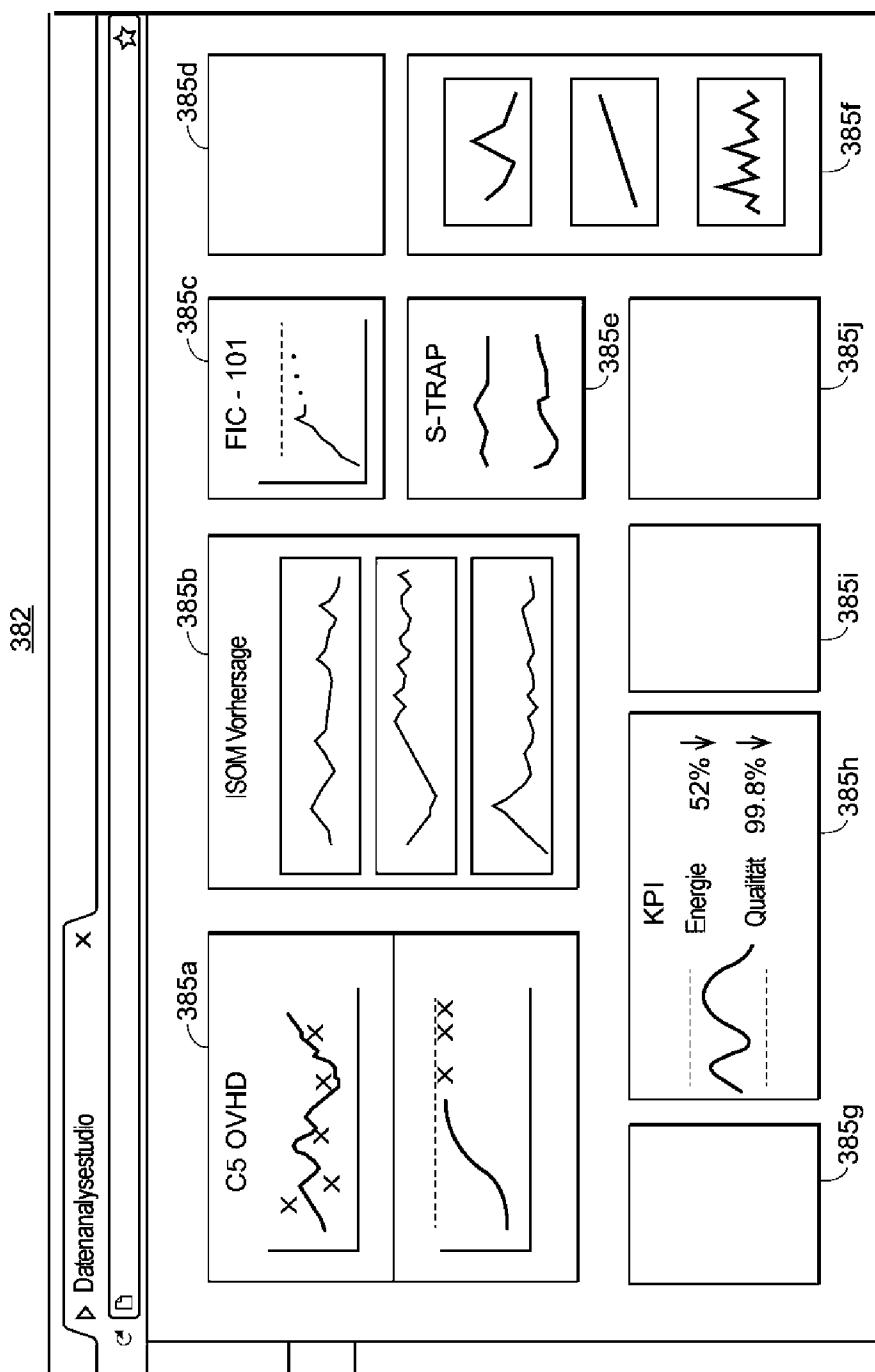


FIG. 4N-2

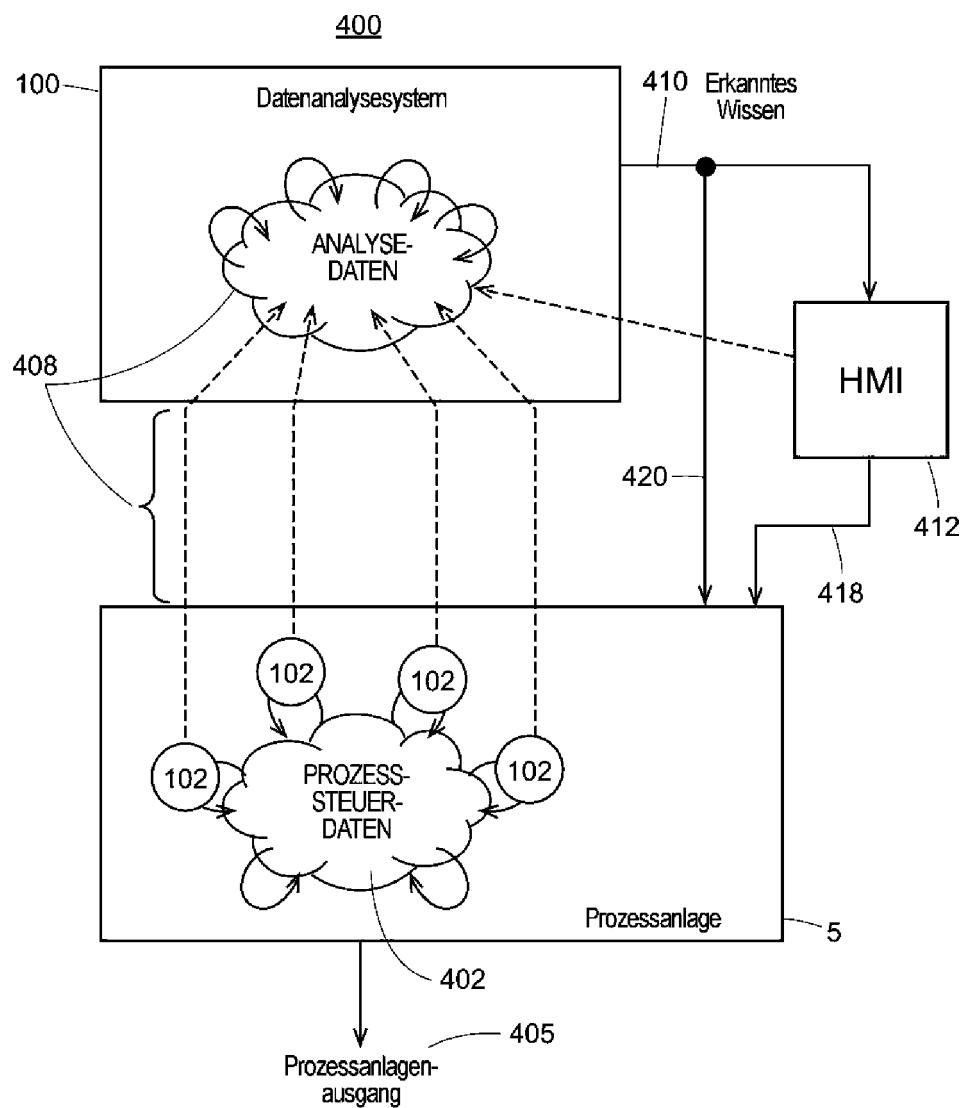
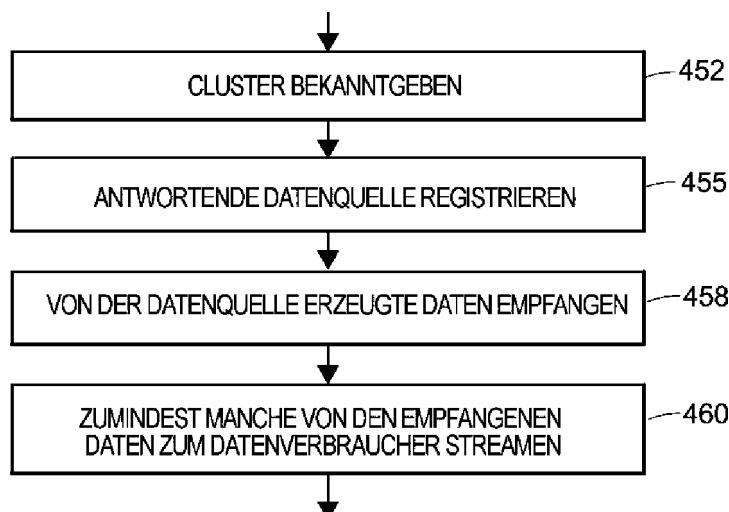
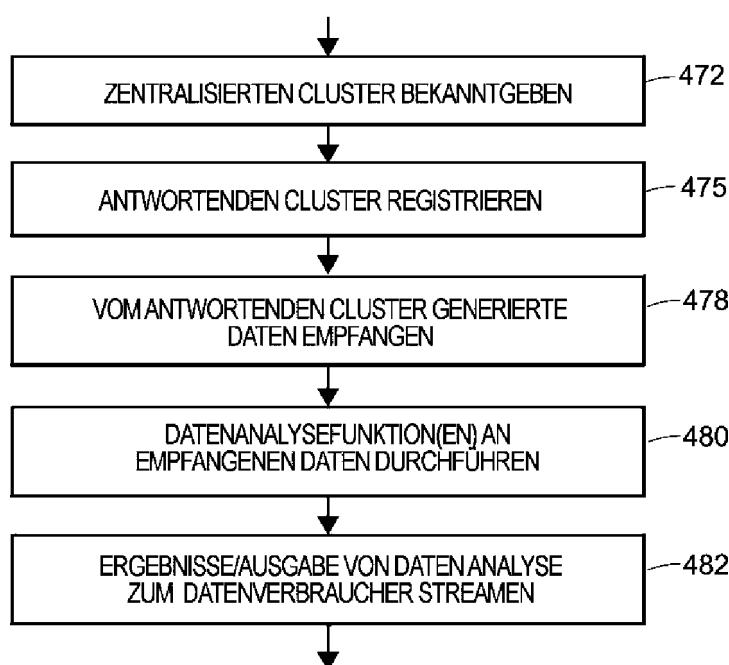
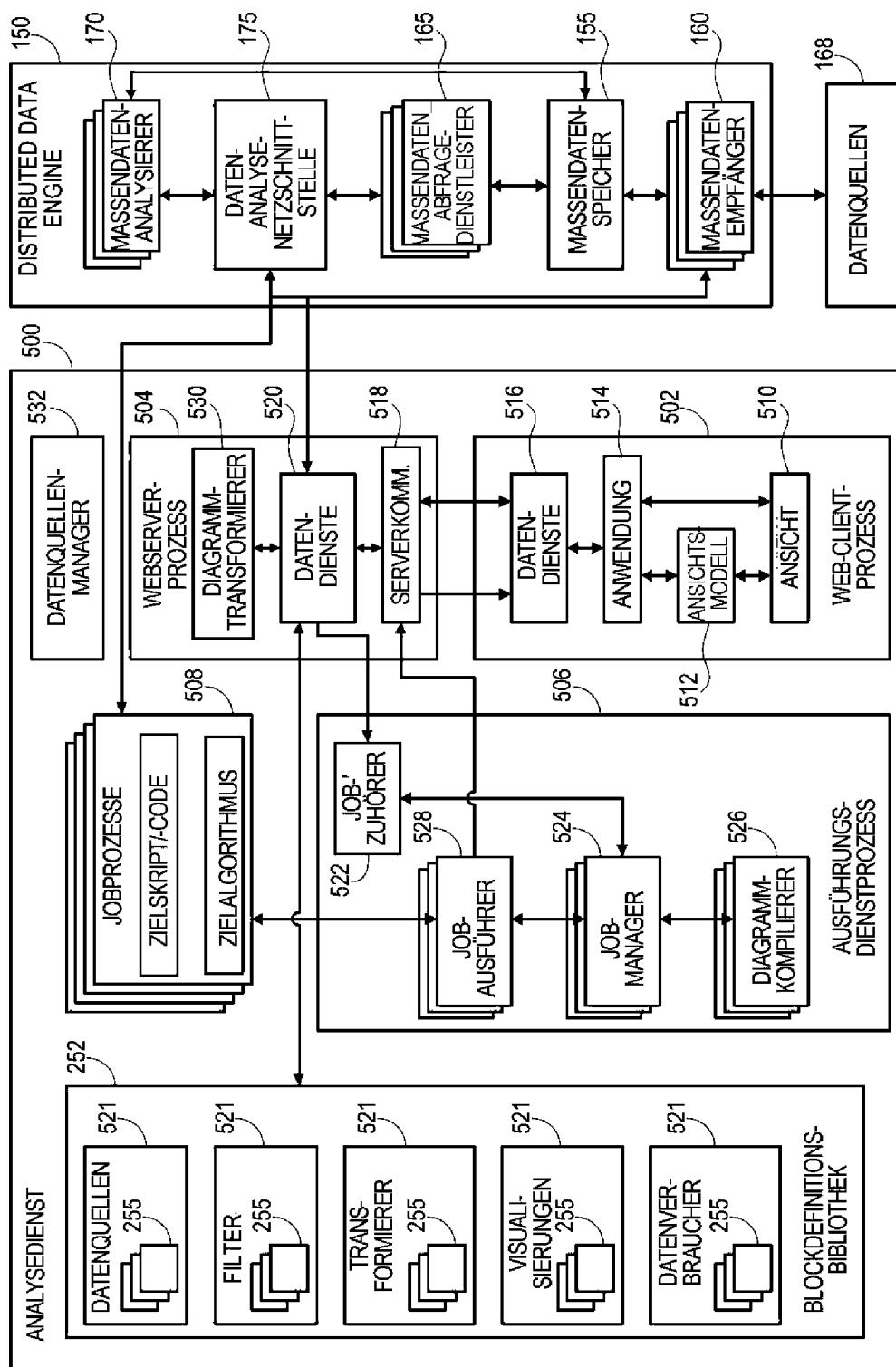


FIG. 4O

450**FIG. 4P**470**FIG. 4Q**

**FIG. 5A**

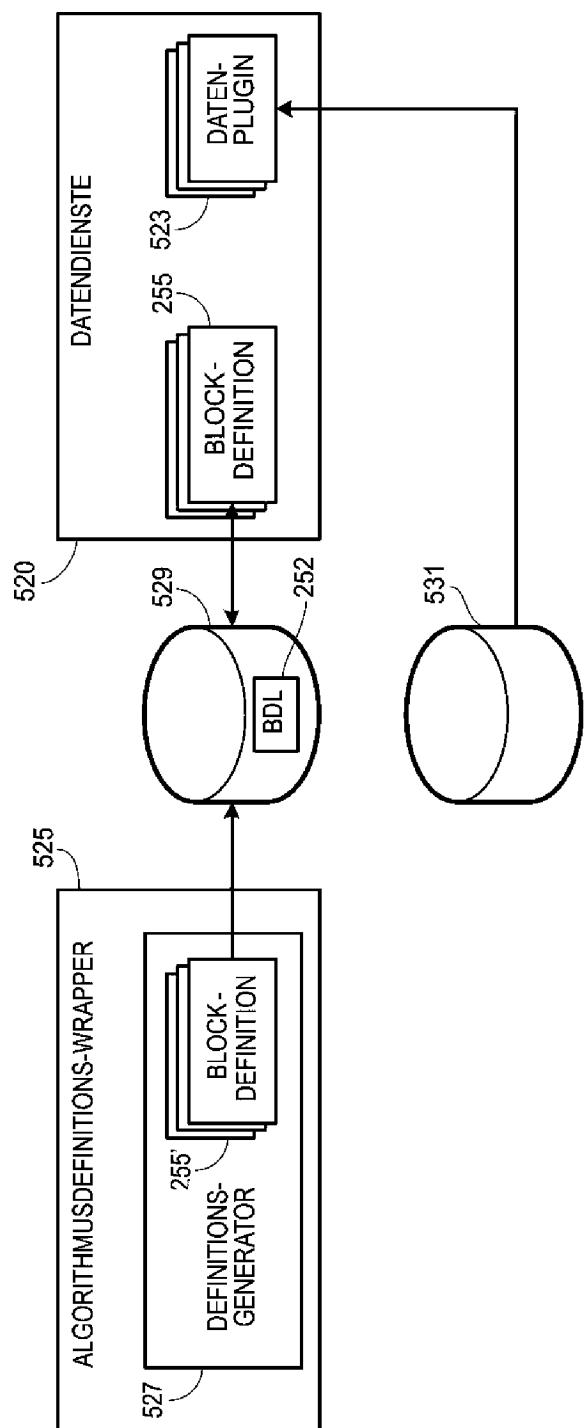
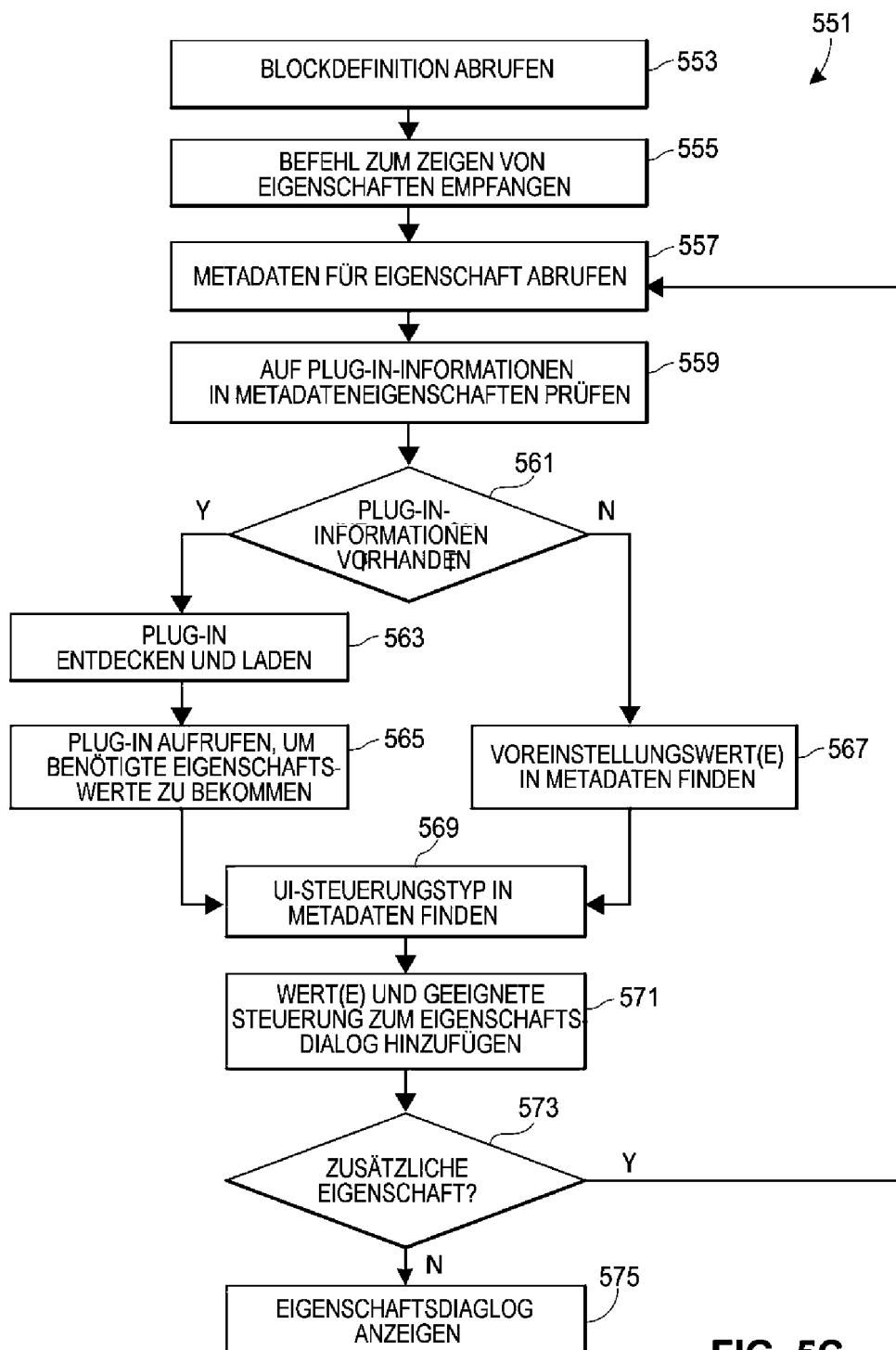
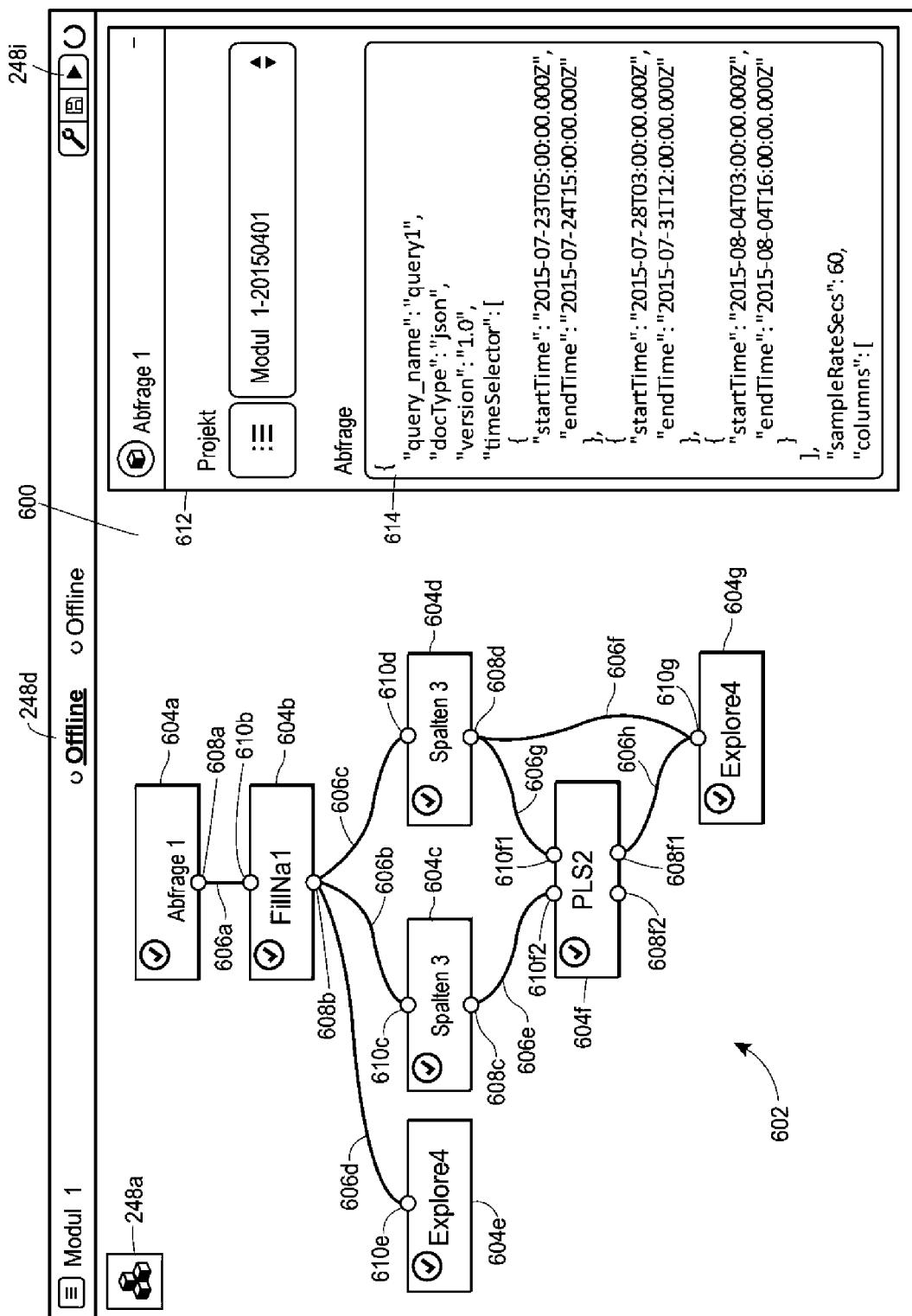
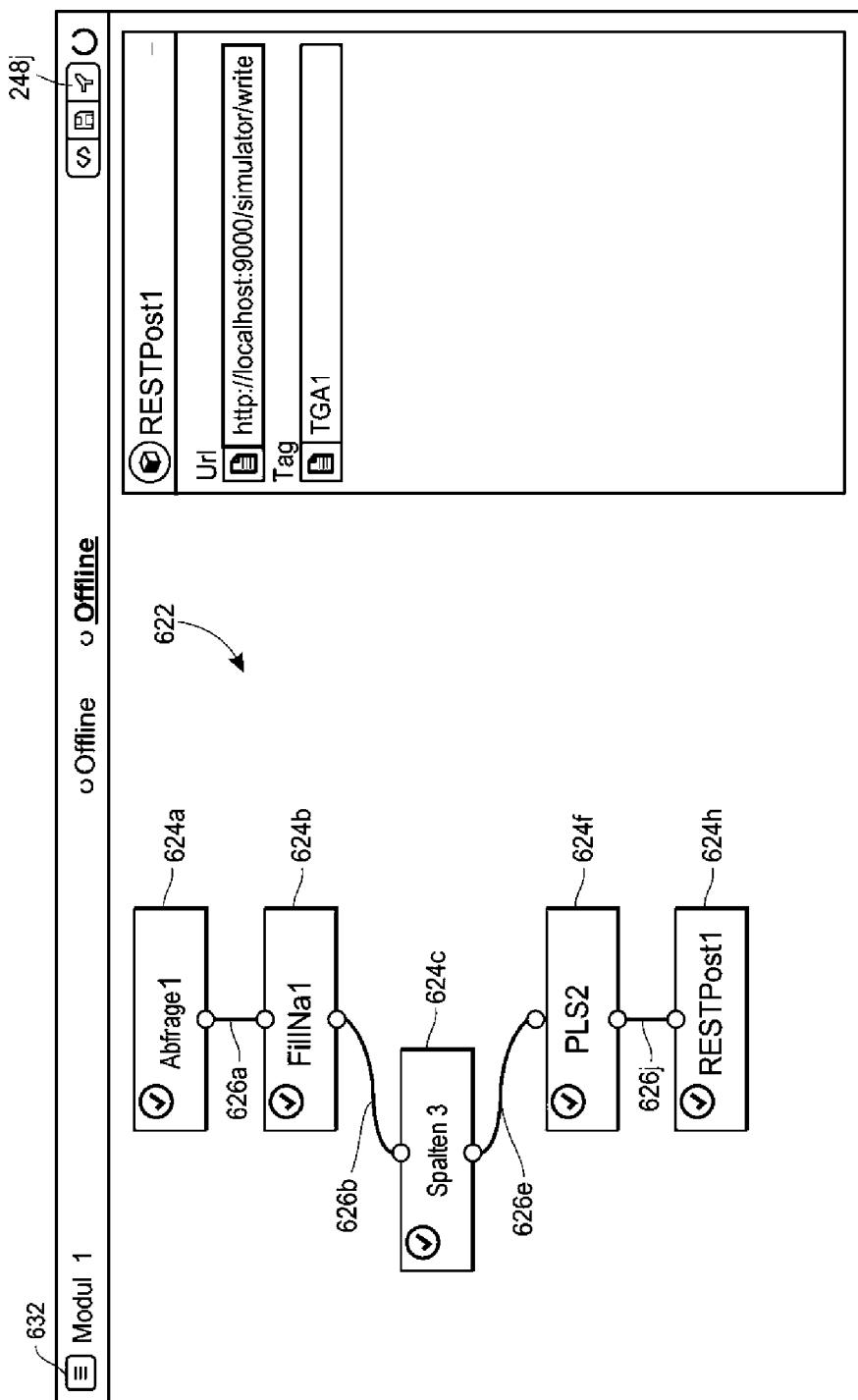


FIG. 5B

**FIG. 5C**

**FIG. 5D**

**FIG. 5E**

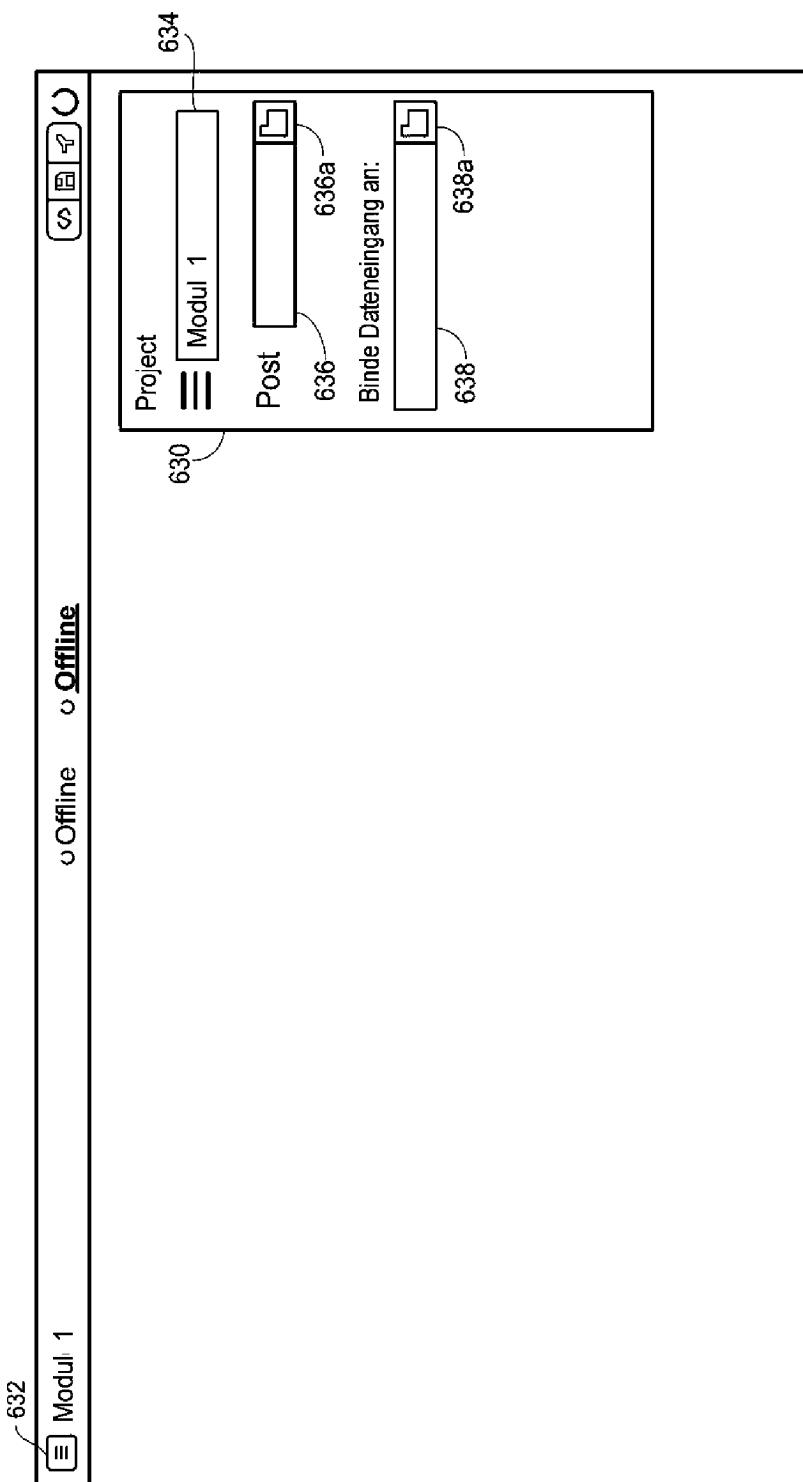
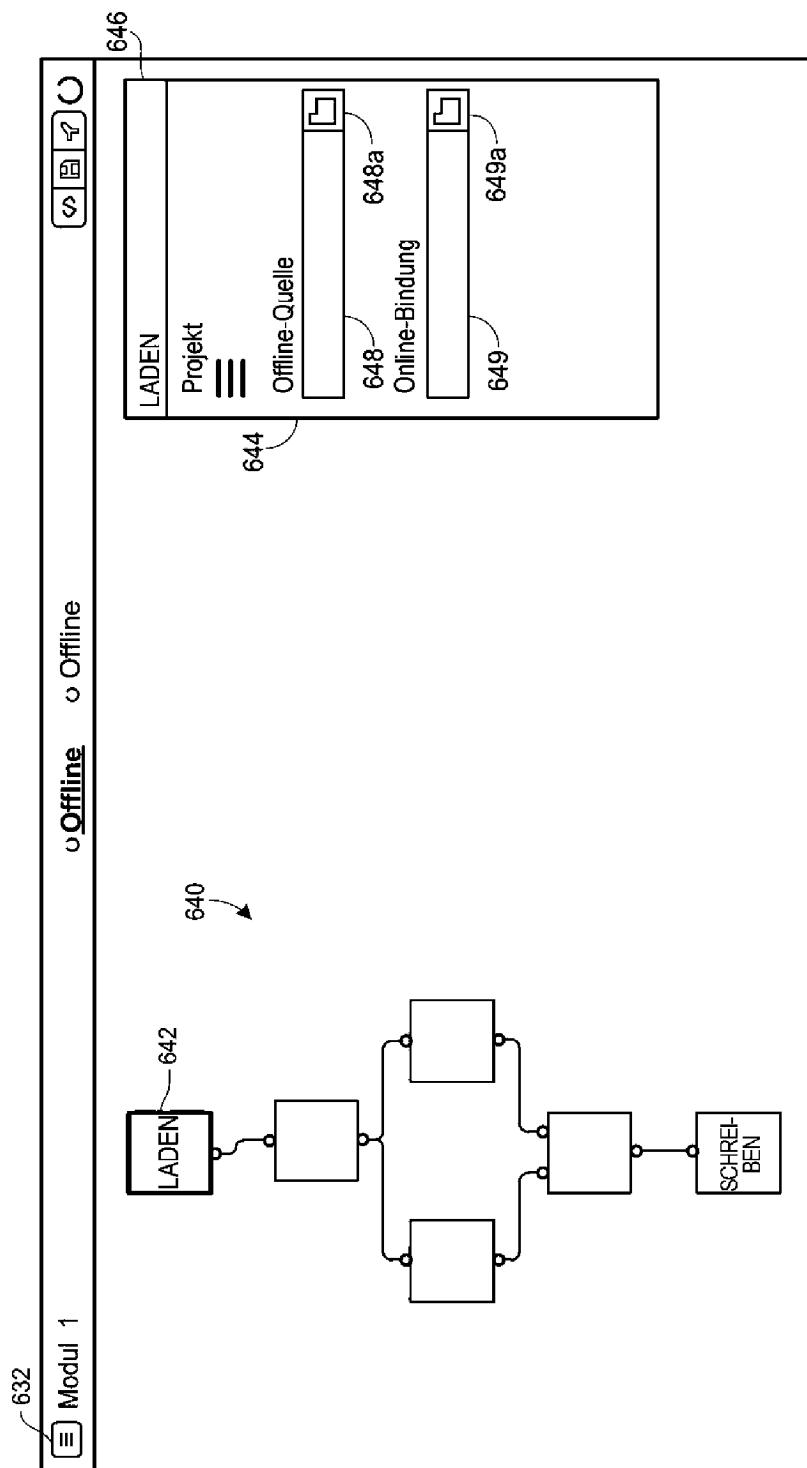


FIG. 5F

**FIG. 5G**

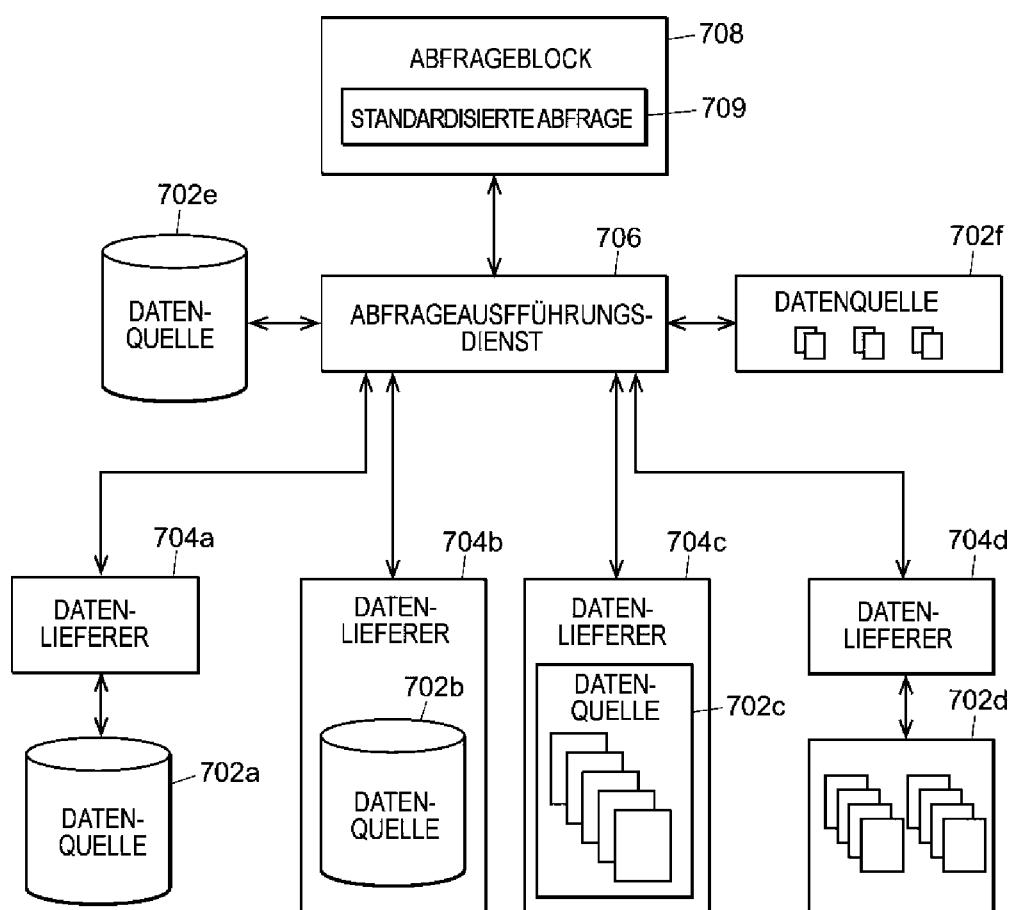


FIG. 6A

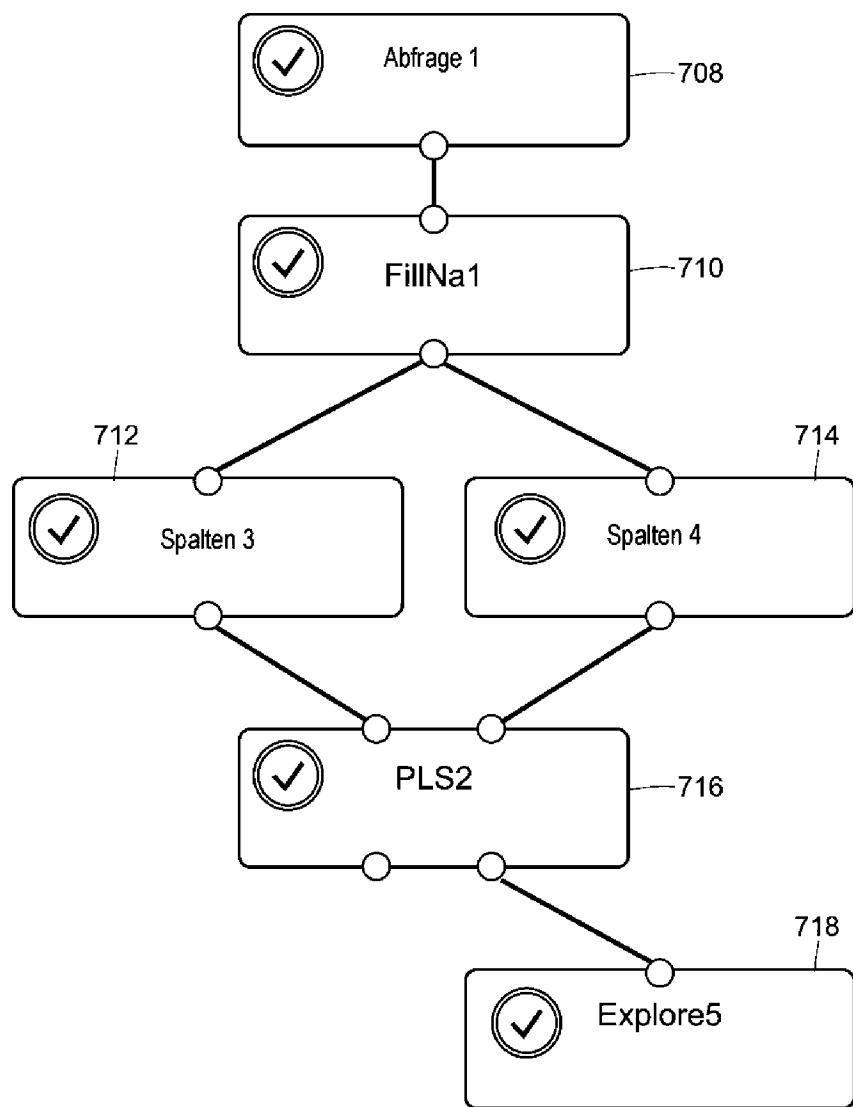


FIG. 6B

```
{
  "query_name": "query1",
  "docType": "json",
  "version": "1.0",
  "timeSelector": [
    {
      "start": "2015-07-23T05:00:00.000Z",
      "end": "2015-07-24T15:00:00.000Z"
    },
    {
      "start": "2015-07-28T03:00:00.000Z",
      "end": "2015-07-31T12:00:00.000Z"
    },
    {
      "start": "2015-08-04T03:00:00.000Z",
      "end": "2015-08-04T16:00:00.000Z"
    }
  ],
  "sampleRateSecs": 60,
  "columns": [
    {
      "tag": "FT630B/DENS.CV",
      "alias": "FT630B/DENS.CV",
      "dataType": "Float",
      "renderType": "VALUE",
      "format": "0.###"
    },
    {
      "tag": "PT615/WIRED_PSIA.CV",
      "alias": "PT615/WIRED_PSIA.CV",
      "dataType": "Float",
      "renderType": "VALUE",
      "format": "0.###"
    },
    {
      "tag": "TT6079/INPUT_1.CV",
      "alias": "TT6079/INPUT_1.CV",
      "dataType": "Float",
      "renderType": "VALUE",
      "format": "0.###"
    },
    {
      "tag": "630.molefrac.c5",
      "alias": "630.molefrac.c5",
      "dataType": "Float",
      "renderType": "VALUE",
      "format": "0.###"
    }
  ]
}
```

FIG. 6C

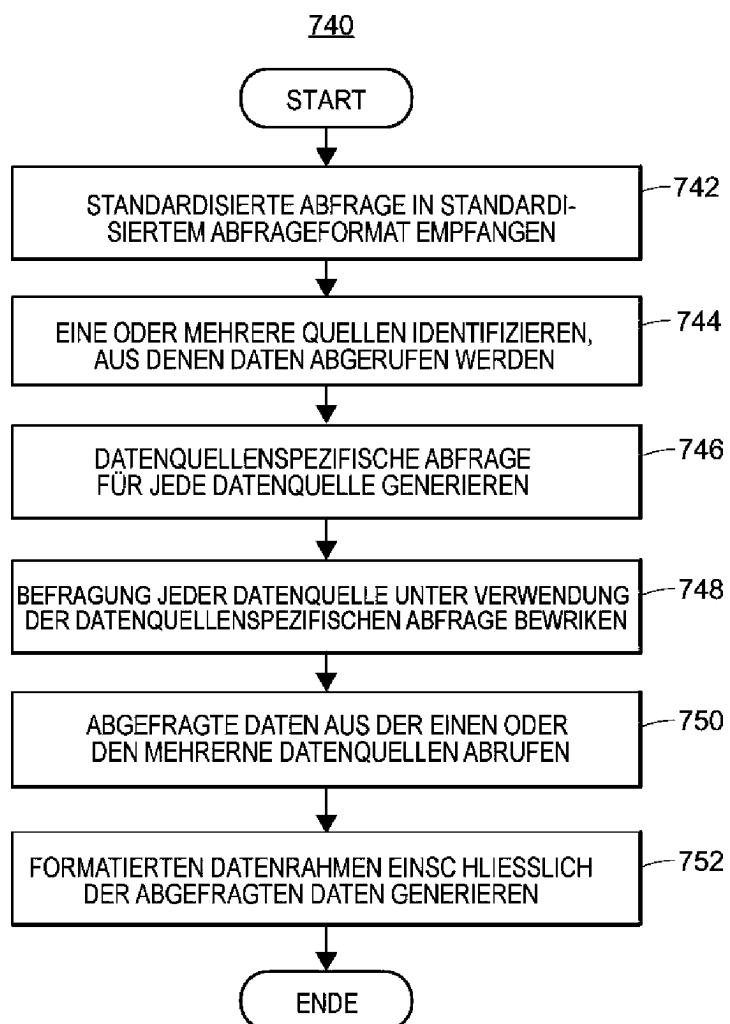


FIG. 6D

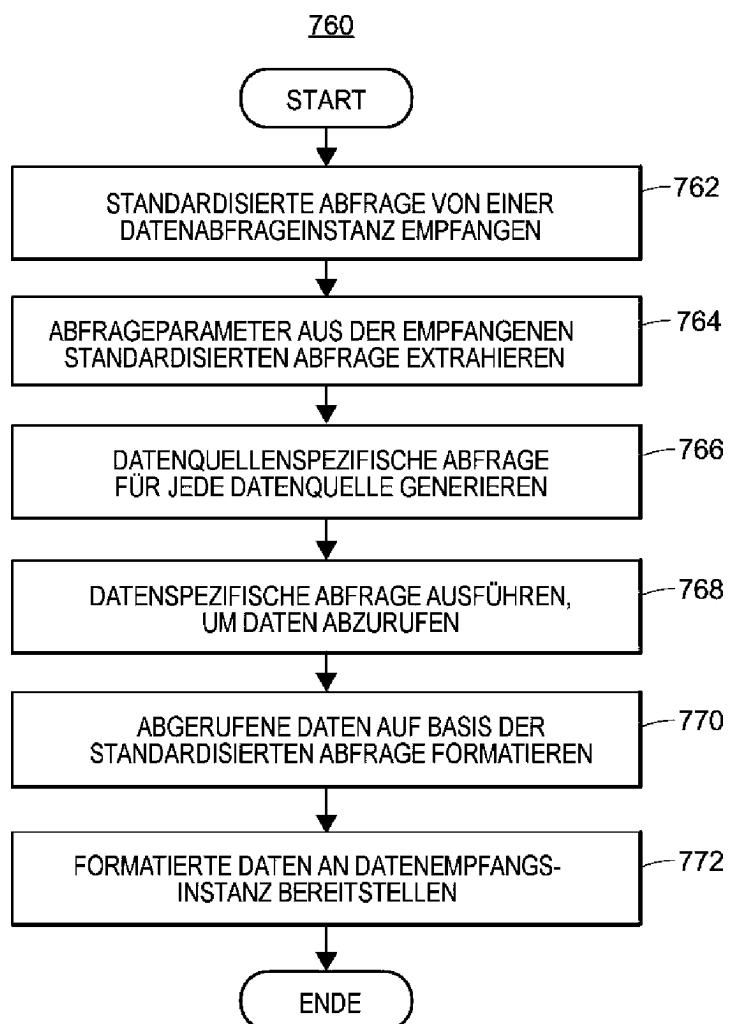
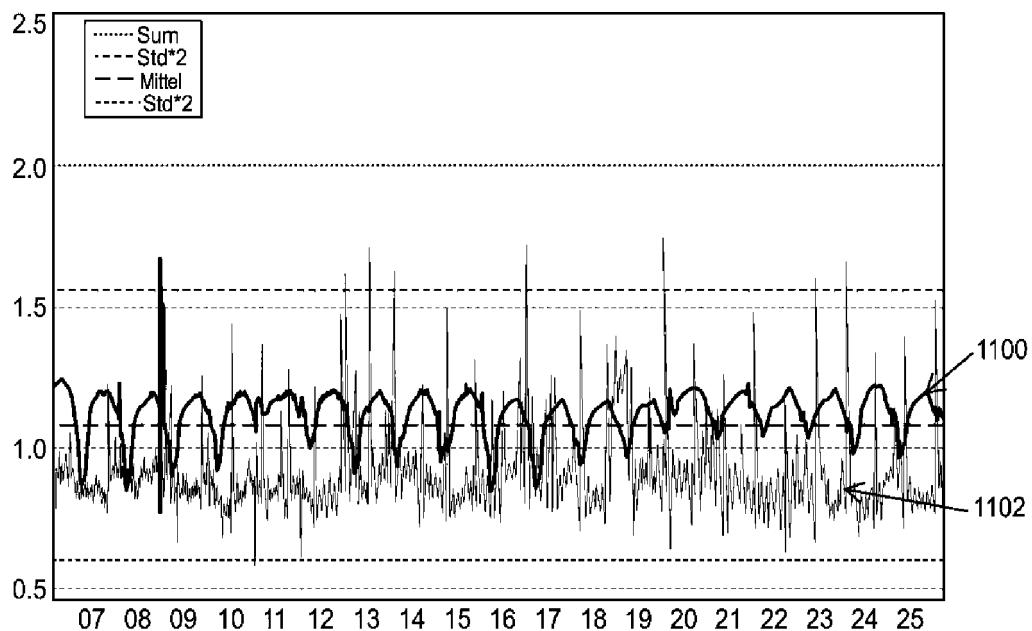
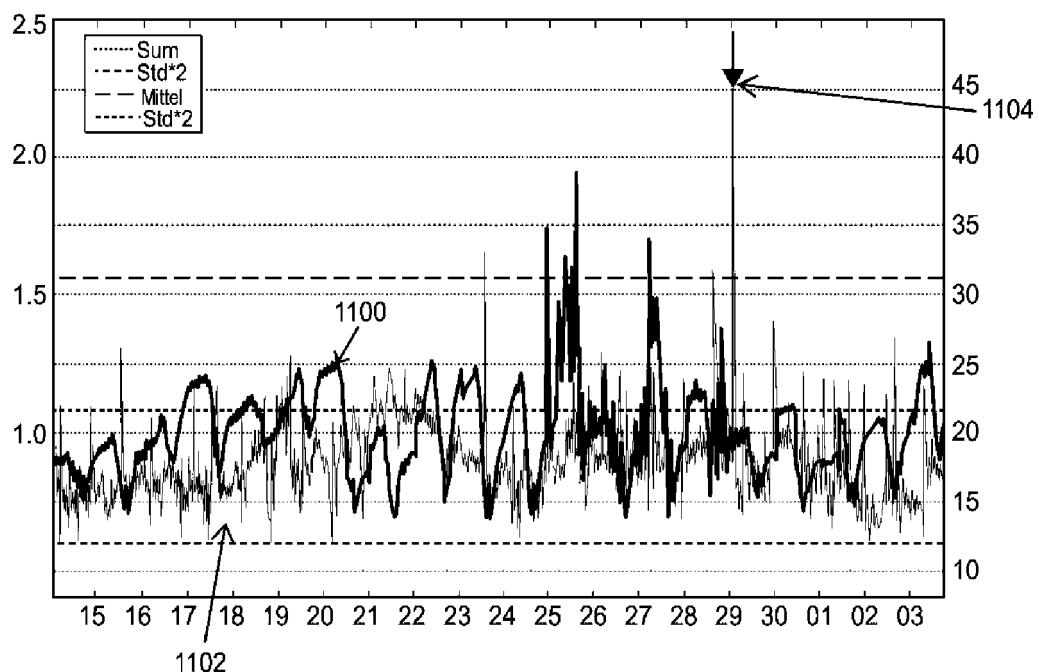


FIG. 6E

**FIG. 7A****FIG. 7B**

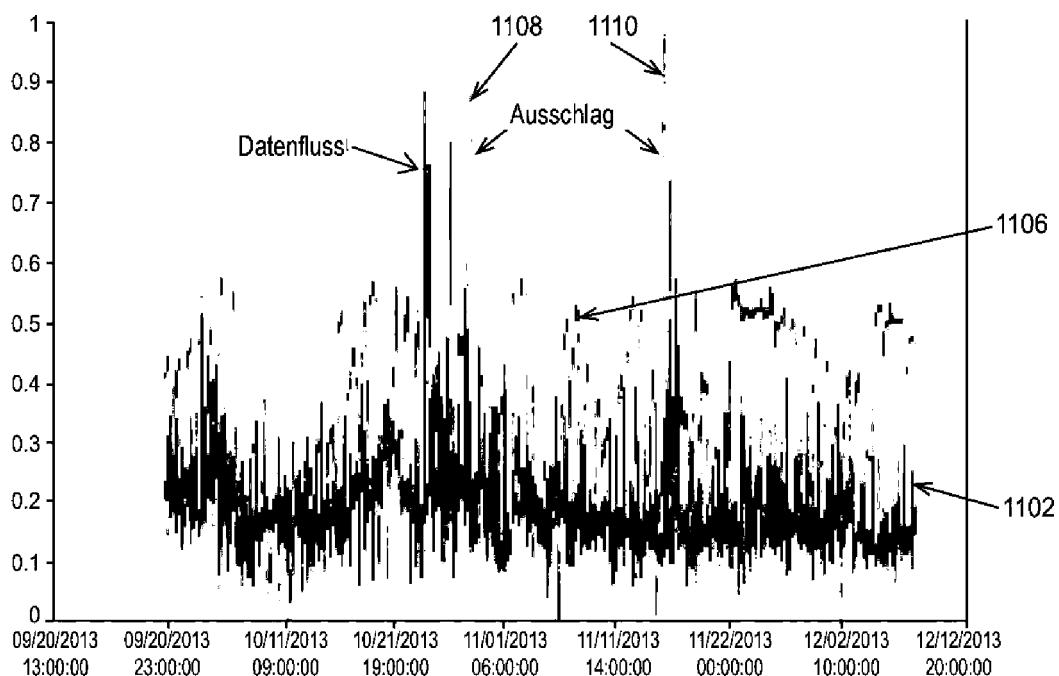


FIG. 7C

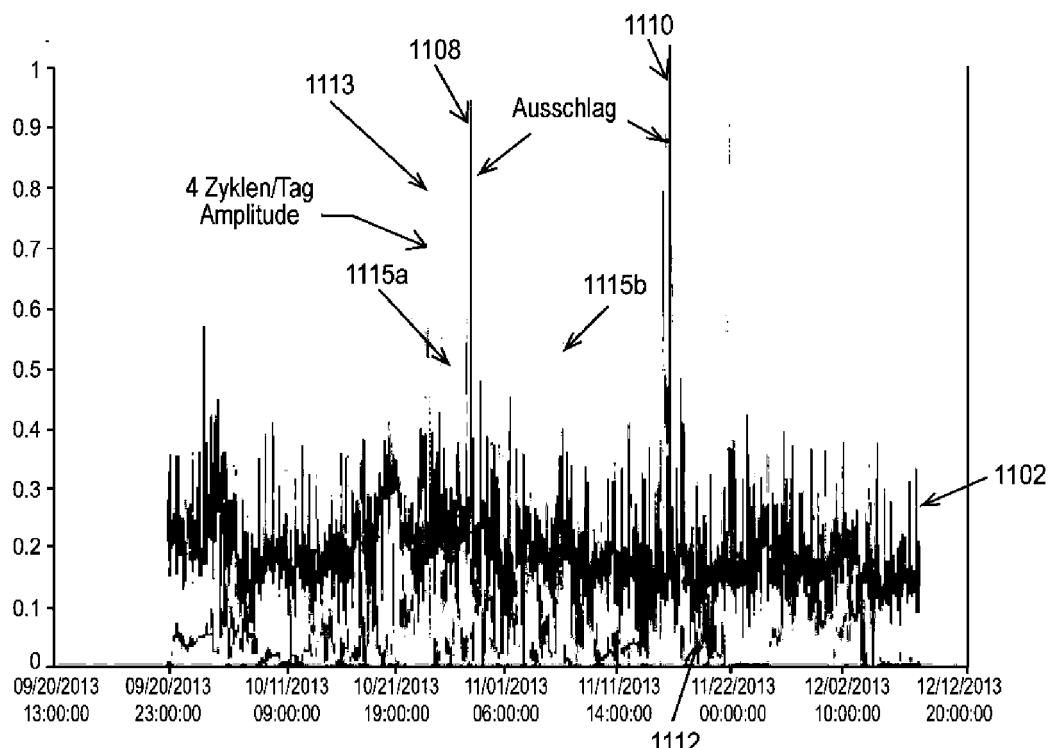


FIG. 7D

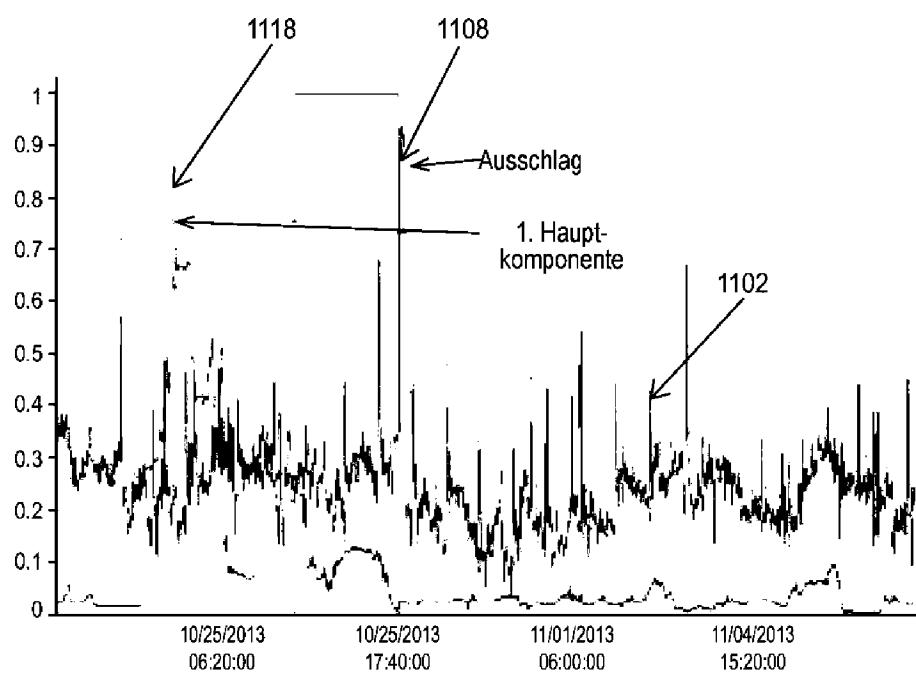


FIG. 7E

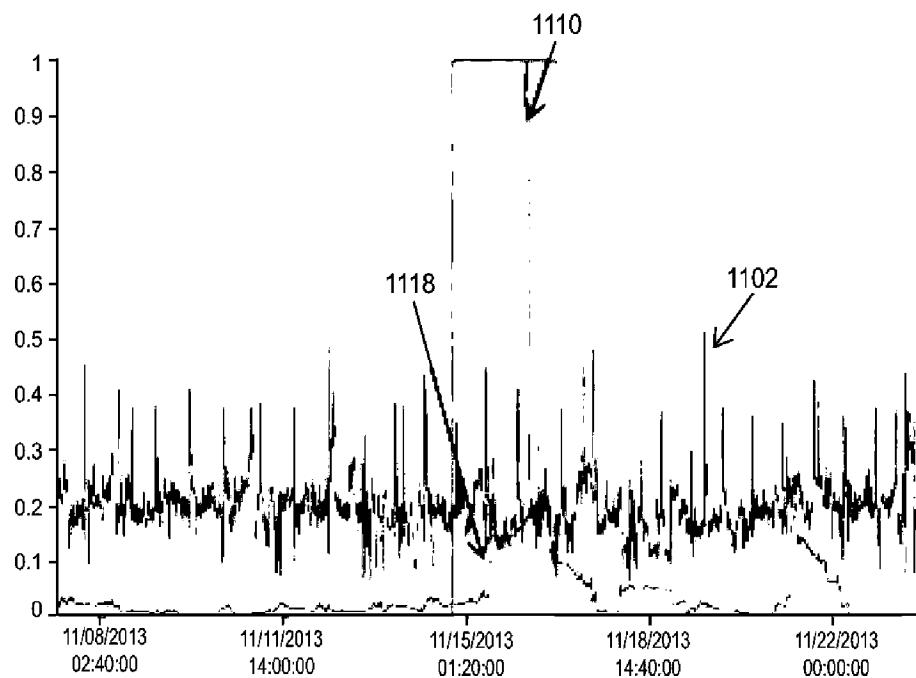
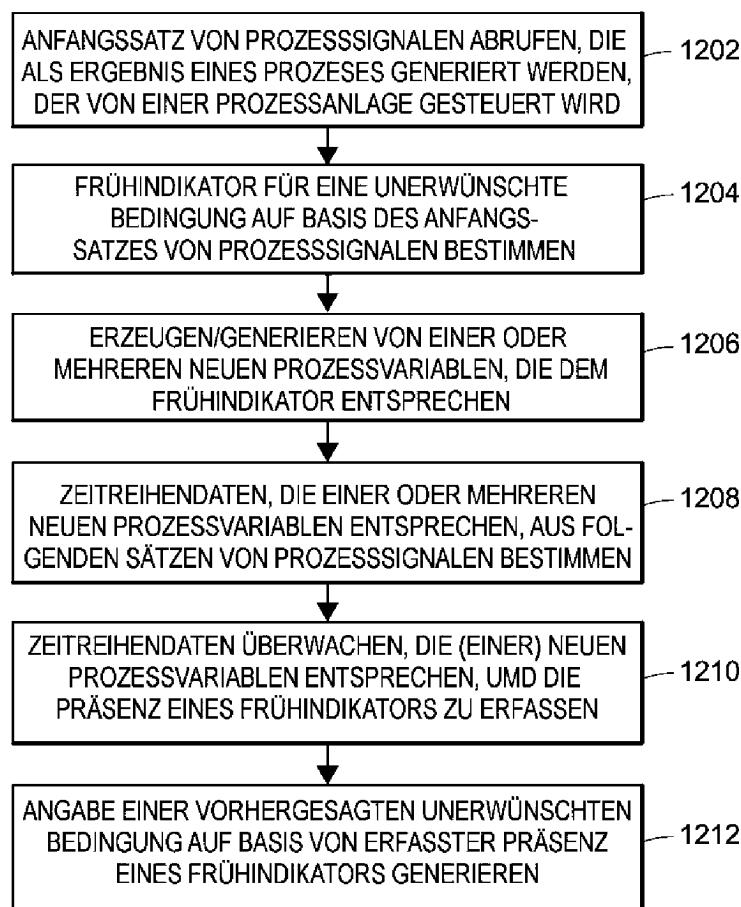


FIG. 7F

1200**FIG. 7G**