DOI 10.2195/LJ_PROC_<S. Feldhorst>_<M. ten Hompel>_<M. Fiedler>_<2010>

Paket Royale – Dezentrale Steuerung für das Internet der Dinge

Dipl.-Inform. Sascha Feldhorst, Univ.-Prof. Dr. Michael ten Hompel
Technische Universität Dortmund
Lehrstuhl für Förder- und Lagerwesen

Dipl.-Inform. Martin Fiedler
Fraunhofer Institut für Materialfluss und Logistik IML
Abteilung für Verpackungs- und Handelslogistik

Abstract: Im vorliegenden Beitrag wird das dezentrale Steuerungskonzept *Paket Royale* vorgestellt. Konzipiert wurde *Paket Royale* nach dem Grundgedanken des *Internet der Dinge* und basiert auf einer Service-orientierten Architektur (SOA) und nutzt (Software-)Agenten als digitale Stellvertreter der zu transportierenden Güter. Die Kombination aus SOA und Agenten soll die Entwicklung von modularen Logistiksystemen erleichtern, die sich vor allem durch höhere Flexibilität und bessere Integration in die betrieblichen IT-Systeme auszeichnen.

1 Einleitung

Bei der Auslegung logistischer Systeme muss bereits zu Beginn des Projekts festgelegt werden, ob das System manuell, halbautomatisch oder vollautomatisch betrieben werden soll. Bei dieser Entscheidung gilt es gerade in Hochlohnländern wie Deutschland, die Anschaffungs- und Betriebskosten der verschiedenen Alternativen sorgfältig gegeneinander abzuwägen. Aufgrund ihrer hohen Anschaffungskosten muss für vollautomatische Systeme gewährleistet werden, dass sie für einen möglichst langen Zeitraum betrieben werden können. Die Betriebsdauer einer vollautomatischen Anlage abzuschätzen, gestaltet sich jedoch in den letzten Jahren zunehmend schwierig. Dies lässt sich darauf zurückführen, dass die Dynamik und Komplexität logistischer Problemstellungen nachhaltig gestiegen ist, die zugrundeliegenden Prozesse häufiger angepasst werden müssen und vollautomatische Systeme diesbezüglich die unflexibelste der drei Varianten darstellen. Grund dafür ist, dass Umbaumaßnahmen an automatisierten Materialflusssystemen häufig aufwändig und teuer sind, da viele Funktionen über verschiedene Systemebenen verteilt sind [FNS10], so dass selbst augenscheinlich lokal beschränkte Änderungen Auswirkungen auf das Gesamtsystem haben können. Dies schlägt sich nicht zuletzt in umbaubedingten Stillstandszeiten nieder. Außerdem erfordern viele Umbaumaßnahmen anlagenspezifisches Wissen des Herstellers, weshalb das durchführende Unternehmen i. d. R. nicht frei am Markt gewählt werden kann [SF10].

Um dieser Problematik zu begegnen, ist die Modularisierung der Produktions- und Materialflussanlagen ein vielversprechender Ansatz [GtH10, May09, KFO10]. Die Modularisierung betrifft dabei neben den elektrischen und mechanischen Komponenten auch die Steuerungssoftware. Insbesondere die Dezentralisierung der vormals zentralen Steuerungsfunktionen, wie z. B. Routing oder Stauvermeidung, ist ein

wichtiger Bestandteil dieses Ansatzes. Während derzeit die Steuerungsfunktionen in wenigen monolithischen Programmen umgesetzt sind, die entweder in Speicherprogrammierbaren Steuerungen oder Materialflussrechnern ablaufen, erfordert eine strikte Modularisierung eine Verlagerung dieser Funktionen in die gesteuerten Gewerke. Es ist heute möglich, die einzelnen Komponenten eines Materialflusssystems mit eingebetteten Computern auszustatten. Dadurch sind Komponenten, wie z. B. Stetigförderer oder Regalbediengeräte, in der Lage miteinander zu kommunizieren und eigenständig Software auszuführen. Dies befähigt sie, aktuelle Prozessinformationen, wie ihre Betriebsbereitschaft, in Echtzeit zur Verfügung zu stellen und sich aktiv an der Steuerung einer Anlage zu beteiligen.

Im vorliegenden Beitrag wird das Steuerungskonzept *Paket Royale* vorgestellt. Bei diesem Konzept wird eine Förderanlage als Verbund interagierender Module aufgefasst, die über ein Netzwerk miteinander vernetzt sind. Die einzelnen Module kommunizieren miteinander und ermöglichen dadurch eine dezentrale Steuerung des Gesamtsystems nach dem Grundgedanken des *Internet der Dinge* [GtH10]. Neben der Modularisierung und Verteilung der Systemfunktion, soll die Integration der Feldebene in die überlagerten IT-Systeme der Unternehmen verbessert werden.

Der Beitrag gliedert sich in vier Teile: Nach anfänglicher Motivation und Einordnung des Themas wird das zugrundeliegende Konzept vorgestellt. Dabei werden insbesondere die Service-orientierte Systemsicht, der Einsatz von Agenten zur Modellierung von Aktivitäten innerhalb des Systems und die daraus resultierenden Potentiale thematisiert. Im Anschluss werden die einzelnen Systemkomponenten beschrieben. Besonderes Augenmerk liegt dabei auf der Umsetzung der dezentralen Steuerungslogik und dem Aspekt der Datensicherheit. Abschließend werden die Ergebnisse der Erprobung des Systems vorgestellt und diskutiert.

2 Hintergrund

Paket Royale basiert auf einer Dezentralisierung der vormals zentralen Steuerungsfunktionen als einen wichtigen Bestandteil der Modularisierung der Gesamtanlage. Dazu verwendet *Paket Royale* eine Service-orientierte Architektur (SOA) und nutzt (Software-)Agenten zur Steuerung. Im Folgenden wird die vorgestellte Arbeit zunächst in aktuelle Forschungsaktivitäten eingeordnet und anschließend wird herausgestellt, warum eine Kombination von Service- und Agentenorientierung zur Steuerung von Materialflüssen sinnvoll ist.

2.1 Einordnung

Seit einigen Jahren untersuchen verschiedene Vertreter aus Forschung und Industrie neue Konzepte und Technologien aus dem Gebiet der Software-Technik und verteilter Systeme hinsichtlich ihrer Anwendbarkeit zur Entwicklung flexibler und modularer Automatisierungssysteme [GtH10, S.29]. Ein früher Ansatz um die Wiederverwendbarkeit, Kapselung und Modularität in der Entwicklung von industriellen Steuerungssystemen zu verbessern, stellt der Standard IEC 61499 dar [Lew01]. Dieser Stan-

dard beschreibt die Entwicklung von verteilten Steuerungssystemen aus miteinander verbundenen Funktionsblöcken – sogenannten Funktionsblocknetzwerken.

Der verfolgte Ansatz profitiert von dem durchgängigen Einsatz von Service-Orientierung innerhalb des Steuerungssystems. Das Fundament hierfür wurde innerhalb des internationalen Forschungsprojekts SIRENA gelegt, das als erstes die Anwendbarkeit von Service-Orientierung innerhalb der industriellen Domäne gezeigt hat. Die Forschungsprojekte SOCRADES und SODA haben die Arbeiten aus SIRENA fortgesetzt und weitere Aspekte Service-orientierter Architekturen (SOA) im industriellen Umfeld beleuchtet. Im Zuge des Projekts SOCRADES wurde u. a. eine Middleware-Plattform, ein Integrationsansatz für bestehende Systeme und ein SOA-basiertes Steuerungssystem mithilfe von Petri-Netzen entwickelt [MLR+10].

Weitere wichtige Grundlagen für *Paket Royale* wurden während des BMBF geförderten Forschungsprojekts *Internet der Dinge* von 2006 bis 2010 erarbeitet. Bereits seit 2002 werden in Dortmund verschiedene Aspekte der dezentralen Steuerung und auch des *Internet der Dinge* in der Intralogistik erforscht. Mithilfe verschiedener Demonstratoren wurde u. a. die Modularisierung, Autokonfiguration, Standardisierung der Datenschnittstellen und die Entwicklung von dezentraler Steuerungssoftware untersucht. Dabei stellte sich heraus, dass eine strikte Modularisierung eine deutlich bessere Übertragbarkeit der erarbeiteten Lösungen auf andere Systeme und Anlagen zulässt. Als Folge entstand die Idee, dass für dezentrale Systeme Modulbaukästen entstehen müssen, welche nicht horizontal, d. h. entlang der Gewerke, sondern vertikal aufgebaut sind [GtH10]. Diese Idee verfolgt auch Mayer mit dem sogenannten Flex-Förderer [May09]. Dabei handelt es sich um ein ortsgebundenes, dezentral gesteuertes Stetigfördermodul, das für den Aufbau von topologieflexiblen Anlagen für reine Beförderungsaufgaben konzipiert wurde. Eine weitere Lösung, welche auf diesem Grundgedanken aufbaut, wurde von Overmeyer et al. in Form von koppelbaren, kleinskaligen Fördermodulen entwickelt [KFO10]. Diese Module sind kleiner als die beförderten Güter und werden für den Transport eines Gutes zusammengeschaltet. Die Steuerung erfolgt dezentral auf Basis von zellularen Automaten. Obwohl sich *Paket Royale* nicht mit der Entwicklung von Fördertechnik beschäftigt hat, wurde die Idee vertikaler Module adaptiert, so dass die resultierenden Steuerungssysteme Anlagen verschiedener Modularisierungsgrade unterstützen.

2.2 SOA und Agenten

Bei *Paket Royale* werden die Systemfunktionen, die klassisch in wenigen großen Software-Blöcken umgesetzt sind, mithilfe des SOA-Ansatzes auf viele kleinere Module, sogenannte Dienste, verteilt. Gemäß dem SOA-Paradigma versteht man unter einem Dienst eine Software, die eine fachliche Funktion kapselt und als Element in größeren Verarbeitungsabläufen genutzt werden kann. Die Nutzung eines Dienstes erfolgt über eine festdefinierte Schnittstelle, die über ein Netzwerk aufgerufen werden kann. Die Umsetzung von Steuerungsfunktionen dieser bietet den Vorteil, dass per Definition nur eine abgegrenzte Teilaufgabe eines Software-Systems umsetzen und

sich deshalb leichter entwickeln, wiederverwenden und warten lassen. Außerdem ist es möglich, durch die Komposition verschiedener Basisdienste höherwertige Dienste in Form von Hierarchien aufzubauen, die sich aufwandsarm an Änderungen anpassen lassen. Während das SOA-Konzept bisher hauptsächlich auf den höheren Ebenen der Unternehmens-IT zum Einsatz kommt, können zukünftig auch logistische Automatisierungssysteme aus lose gekoppelten Diensten zusammengestellt und nahtlos in die Diensthierarchie der Unternehmen integriert werden. Dafür ist es jedoch notwendig, dass auch die technischen Gewerke, wie z. B. Förderer oder Regalbediengeräte, ihre Funktionen als Dienste anbieten [SF10]. *Paket Royale* setzt zu diesem Zweck auf eine SOA, die speziell für den Einsatz auf der Geräteebene entwickelt worden ist – eine sogenannte *SOA für Geräte*. Die Anwendbarkeit dieses Konzepts wurde in verschiedenen internationalen Forschungsprojekten evaluiert [CBC+09].

Neben einer Service-orientierten Systemsicht verwendet *Paket Royale* Software-Agenten. Ein Agent wird (ähnlich wie ein Dienst) als kleine, abgeschlossene Software betrachtet, die ein vorgegebenes Ziel kapselt und eigenständig in einer Laufzeitumgebung abläuft. Zur Zielerreichung können Agenten miteinander kommunizieren und, falls sinnvoll, dazu auch die Ausführungsumgebung wechseln. Im Gegensatz zu einem Dienst arbeitet ein Agent proaktiv, d. h. während sich ein Dienst passiv verhält und nur auf die Aufrufe der Dienstnutzer oder externe Ereignisse reagiert, agiert der Agent und versucht fortlaufend, das gegebene Ziel zu erreichen. Ein Agent ist zur Umsetzung von kontinuierlichen Steuerungsaufgaben innerhalb eines Materialflusssystems deshalb besser geeignet als ein Dienst [SF10]. Darum werden in *Paket Royale* die Funktionen der Anlage als Dienste und die verteilte Steuerungslogik in Form von Agenten realisiert, um so die Stärken aus beiden Ansätzen für die Systementwicklung nutzbar zu machen.

3 Paket Royale

Im Zuge des Projekts *Paket Royale* ist ein neues Steuerungskonzept für Materialflussanlagen entstanden, das auf einer vollständig dezentralen Steuerungslogik beruht. Dies ermöglicht die Steuerung einer logistischen Anlage ohne übergeordnete Koordinierungskomponente, wie z. B. einen Materialflussrechner. *Paket Royale* stellt ein Rahmenwerk für die Entwicklung von dezentralen Materialflusssteuerungen bereit. Zur Evaluierung des Rahmenwerks ist ein Steuerungssystem für eine bestehende Stetigförderanlage realisiert worden. Das zugrunde liegende Konzept und die einzelnen Systemkomponenten werden im Folgenden vorgestellt.

3.1 Konzept

Um die Entwicklung von Anlagen aus vertikalen Modulen zu unterstützen, wird innerhalb von *Paket Royale* jedes Gewerk als Dienstleister gegenüber der zu transportierenden Güter angesehen. Die angebotenen Dienste können dabei sowohl die abstrakten Funktionen des Gewerks, wie z. B. Transport, Verzweigung oder Pufferung,

als auch die direkte Ansteuerung der verbauten Aktorik und Abfrage der Sensorik zur Verfügung stellen. Damit wird der einzelne Förderer zumindest aus Sicht der Software zum mechatronischen Gerät. Für die Kommunikation mit diesen Geräten wird auf standardisierte IT-Technologien, wie Ethernet und *Web Services* zurückgegriffen. Insbesondere der Einsatz von *Web Services* stellt sicher, dass die Schnittstelle in maschinenlesbarer Form vorliegt und über die gängigen Programmiersprachen, wie Java oder .NET, angesprochen werden kann. Dies erleichtert u. a. die Integration der Anlagendienste in überlagerte WMS, Leit- oder ERP-Systeme.

Wie bereits zuvor erwähnt, ist eine Dienstlandschaft grundsätzlich reaktiv. Zur Umsetzung einer Steuerung auf Basis von Anlagendiensten stehen daher zwei Ansätze zur Verfügung: Zum einen können die höheren Steuerungsaufgaben in eine überlagerte Software integriert werden, welche die Dienste gemäß den Steuerungszielen aufruft. Zum anderen können die Steuerungsaufgaben direkt in die Dienste integriert werden, welche dann ausschließlich auf Ereignisse aus dem technischen Prozess oder angrenzenden Systemen reagieren. Da eine rein ereignisorientierte Sichtweise bei der Steuerungsentwicklung weniger verbreitet ist und von den Entwicklern ein Umdenken bei der Programmierung erfordert, realisiert *Paket Royale* die Steuerungslogik in Form eines überlagerten Multiagentensystems (vgl. Abbildung 1).

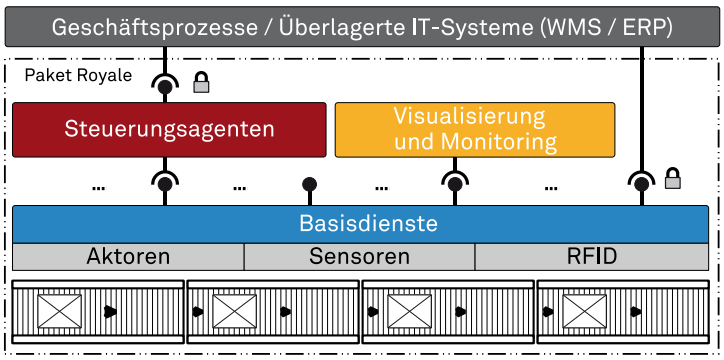


Abbildung 1: Systemüberblick *Paket Royale* [SF10]

Wie man der Abbildung 1 entnehmen kann, setzt sich *Paket Royale* neben der gesteuerten Anlage aus drei Hauptkomponenten zusammen: den Basisdiensten, den Steuerungsagenten sowie einer Visualisierungs- und Monitoring-Komponente.

3.2 Basisdienste

Bei den Basisdiensten handelt es sich um eine verteilte, modulbasierte Integrationssoftware, welche eine Dienstumgebung für die Interaktion mit dem technischen Prozess bereitstellt und somit als Bindeglied zwischen der SOA- und der Automatisierungswelt fungiert [FLtH+09]. Dazu kapseln die Basisdienste die Funktionen der Fördertechnik in Form von wiederverwendbaren Diensten. Diese Dienste ermöglichen es, aus entfernten Programmen direkt auf die Aktoren und Sensoren einer Anlage zuzugreifen. Auf diese Weise können z. B. die Steuerungsagenten Motoren ansteuern oder sich über Sensorereignisse informieren lassen.

Innerhalb der Basisdienste wird jedes Gewerk durch ein eigenes Software-Modul repräsentiert. Ein solches Software-Modul, das auch als Gerätefassade bezeichnet wird, bietet unterschiedliche Dienste für die Ansteuerung und Überwachung eines Gewerks an. Welche Dienste eine Gerätefassade anbietet, ist freikonfigurierbar und hängt von der Aufgabe des Gewerks innerhalb der Anlage ab. Auf der Versuchsanlage werden die Gerätefassaden auf sieben Industrie-PCs (IPCs) ausgeführt, welche direkt mit der Fördertechnik verbunden sind (vgl. Abbildung 2).

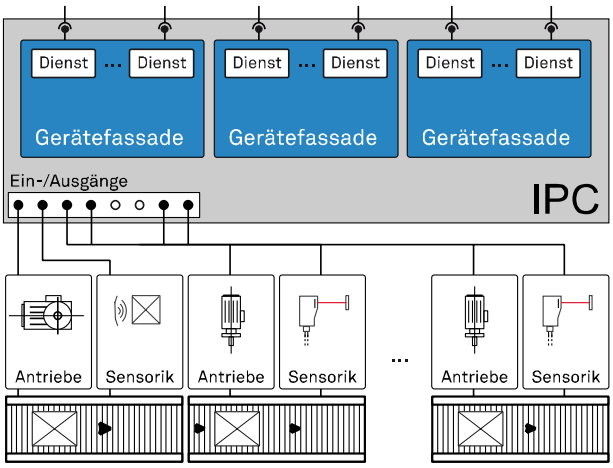


Abbildung 2: Aufbau der Basisdienste

3.3 Steuerungsagenten

Im Gegensatz zu klassischen Materialflusssteuerungen wird bei *Paket Royale* die Steuerungslogik mithilfe eines Multiagentensystems realisiert. Darin übernehmen die Agenten atomare Steuerungsaufgaben und agieren dezentral ohne eine gemeinsame Anlaufstelle. Zu diesem Zweck wird eine eigene Agentenlaufzeitumgebung angeboten, die verschiedene Strategien für die Steuerung von stetigen Materialflussanlagen implementiert und zusätzlich die Entwicklung neuer Strategien erlaubt. Die derzeit integrierten Strategien dienen der Nachfolger- und Pufferüberwachung (NPÜ) sowie der dezentralen Sammel- und Verzweigungssteuerung. Die NPÜ ist notwendig, da es keine zentrale Entscheidungsstelle gibt, die festlegt, ob ein Förderer zu einem bestimmten Zeitpunkt fördern darf. Deshalb darf ein Förderer nur starten, wenn vorgegebene Regeln erfüllt sind, z. B. wenn einer der Nachfolger gestartet ist. Die Einhaltung dieser Regeln zu prüfen ist Aufgabe der NPÜ. Um Kollisionen im Konfliktbereich von Zusammenführungen zu vermeiden, werden verschiedene Sammelstrategien angeboten. Im einfachsten Fall wird die nachrangige Strecke überwacht und ankommende Güter dürfen nur einfahren, sofern kein Gut auf der vorrangigen Strecke beeinflusst wird. Wenn zusätzlich die vorrangige Strecke eine Überwachung des Konfliktbereichs durchführt, kann auch eine relative Vorfahrt oder sogar eine gleichberechtigte Abfertigung realisiert werden. Es gilt zu beachten, dass die Strategien nur auf Sensoren der beteiligten Förderer zugreifen, um so die konsequente Modularisierung sicherzustellen. Neben den Strategien zur operativen Steuerung der Anlage unterstützt die Plattform außerdem eine verteilte Auftragsverwaltung.

Zur Umsetzung der Steuerungsaufgaben werden verschiedene Agententypen definiert, die sowohl stationär als auch mobil arbeiten. Dazu gehören der Fördereragent, der Paketagent und der Auftragsagent. Der Fördereragent vereint in sich alle Steuerungsfunktionen, die den Zustand der Fördertechnik verändern (NPÜ, Sammel- und Verzweigungssteuerung). Damit nicht mehrere Agenten den Zustand desselben Fördermoduls verändern, wird jedem Förderer genau ein Fördereragent zugeordnet. Auch innerhalb des Agenten wird diese Logik beibehalten und jedem Aktor des Förderers eine eigene Controllerkomponente zugewiesen (vgl. Abbildung 3b).

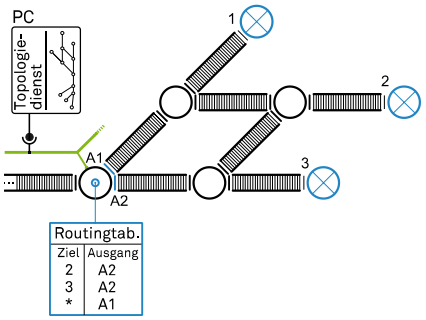
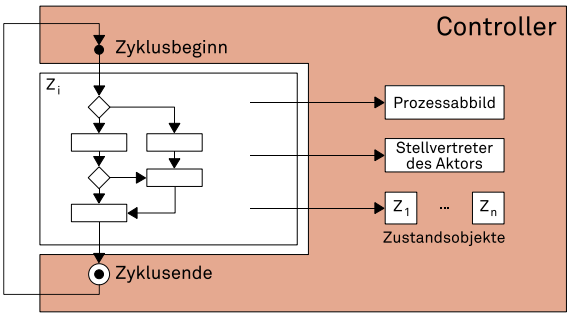


Abbildung 3a: Routing-Beispiel



3b: Aufbau eines Controllers für einen Aktor

Ein Fördereragent enthält also einen oder mehrere Controller, in denen die Steuerungsstrategien in Form von Zustandsmaschinen umgesetzt werden. Jede Zustandsmaschine ist für einen Aktor des Förderers zuständig und besteht aus einer Menge von Zustandsobjekten, welche die zugehörige Logik enthalten. Welche Zustandsobjekte in den Agenten geladen werden, ist flexibel konfigurierbar und kann während der Laufzeit geändert werden. So können sowohl allgemeingültige als auch spezifische Arbeitsabläufe in einem Fördereragenten hinterlegt werden. Aktuell werden verschiedene Zustandsmaschinen angeboten, z. B. für eine absolute Vorfahrt an Zusammenführungen oder eine zielabhängige Verzweigung an Weichen.

Das Routing innerhalb eines dezentralen Systems erfordert eine Topologie-Erkennung und eine anschließende Berechnung der Routing-Tabellen. Dies erfolgt mithilfe von sogenannten Topologiediensten. Die Topologiedienste verwalten jeweils einen Teil einer Anlage, kennen dessen Topologie und berechnen die Routing-Tabellen der zugehörigen Weichen (vgl. Abbildung 3a). Zur Bewältigung dieser Aufgaben können in den Diensten verschiedene Strategien realisiert werden. Aktuell werden Teiltopologien aus speziell formatierten CAD-Zeichnungen extrahiert und in die Topologiedienste geladen. Diese tauschen ihre Teiltopologien aus und berechnen daraus eine globale Topologie, auf der mithilfe des Algorithmus von Dijkstra die Routing-Tabellen bestimmt werden. Dieser Umsetzung liegt der Gedanke zugrunde, dass jeder Umbau der Anlage auch mit gewissen Planungstätigkeiten verbunden ist und deshalb eine CAD-Zeichnung eine sinnvolle, da immer verfügbare, Ausgangsbasis für die Topologie-Erkennung darstellt. Ändert sich die Topologie wird dies automatisch unter den Diensten verteilt, die Routing-Tabellen aktualisiert und die Agenten per Ereignisnachricht darüber informiert.

Die verteilte Auftragsverwaltung wird durch zwei weitere Agententypen umgesetzt: Zum einen durch stationäre Auftragsagenten, die an den Wareneingängen Transportaufträge von übergeordneten Systemen entgegennehmen. Zum anderen durch mobile Paketagenten, die als digitale Stellvertreter zusammen mit dem beförderten Gut über die Anlage migrieren.

Ein Paketagent ist mobil, damit er stets dort ausgeführt wird, wo sich das zugehörige Gut befindet. Nur so kann ein Paketagent die Entscheidungen der Fördereragenten beeinflussen. An einer Weiche legt der mobile Paketagent fest, welchen Weg das Gut einschlagen soll und migriert anschließend zum nächsten Entscheidungspunkt. Der Lebenszyklus eines Paketagenten ist in Abbildung 4 dargestellt. Durch das Zusammenwirken der Agenten wird die Versuchsanlage auch ohne übergeordnete Steuerungsinstanz komplett dezentral gesteuert und bei gleichem Funktionsumfang eine höhere Flexibilität erzielt. Dies ist darauf zurückzuführen, dass die Agenten eigenständige Software-Einheiten sind, die eine spezielle Strategie kapseln und zur Laufzeit separat ausgetauscht werden können, ohne dabei das gesamte Steuerungssystem zu beeinflussen. Durch dieses Konzept ist es daher möglich, bestimmte Teile einer Anlage zu verändern und wieder in Betrieb zu nehmen.

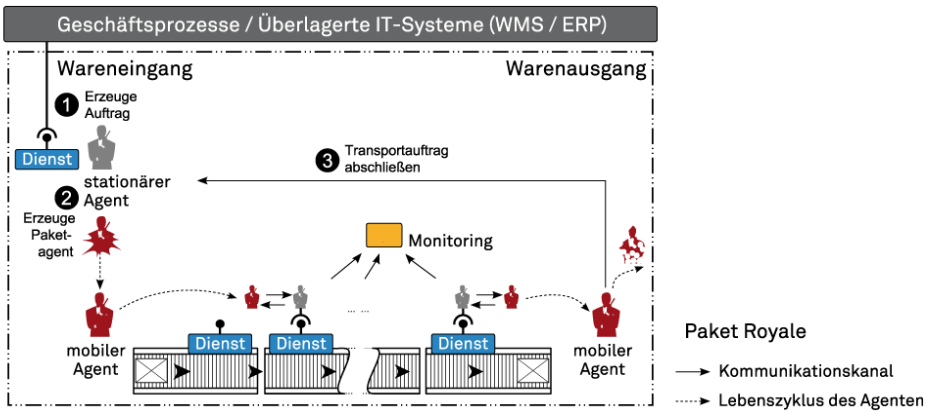


Abbildung 4: Lebenszyklus eines Paketagenten

3.4 Visualisierung und Monitoring

Die Bereitstellung der Sensorinformationen direkt aus der Feldebene ermöglicht eine sehr genaue Bestimmung des Anlagenzustands. Die so gewonnenen Daten werden bei *Paket Royale* in einer speziell entwickelten Leitstand-Software aufbereitet und in einer gerenderten 3D-Darstellung angezeigt. Dadurch wird der Leitstand zum umfassenden Überwachungswerkzeug, in dem Systemereignisse, wie z. B. Störungen, für den Bediener übersichtlich angezeigt werden. Passend zum Grundgedanken des *Internet der Dinge* stellt dabei jeder Förderer seine eigene Darstellung in Form eines 3D-Modells zur Verfügung. Sobald ein Förderer gestartet wird, kann sein Modell über das Netzwerk direkt in den Leitstand geladen werden. Außerdem kann mithilfe der Sensor- und RFID-Ereignisse innerhalb der Visualisierung auch die Bewegung der Materialflussobjekte synchron zur realen Anlage dargestellt werden [FFH+10].

3.5 Kommunikation und Datensicherheit

Die mit *Paket Royale* entwickelten dezentralen Steuerungssysteme stellen verschiedene Basisdienste zur Verfügung. Diese offene Schnittstelle für den direkten Zugriff auf die einzelnen Förderelemente erlaubt eine flexible Gestaltung aller Steuerungsabläufe und eine vereinfachte Integration neuer Komponenten. Aufgrund ihres feingranularen Aufbaus bietet diese Schnittstelle jedoch zusätzliche Möglichkeiten für ungewollte Manipulationen, da die Anlage in vollem Umfang über ein Netzwerk erreichbar ist. Die inneren Abläufe der Steuerung sind aus Sicht der Datensicherheit angreifbar. Da die Datensicherheit eines Automatisierungssystems direkte Auswirkungen auf die Betriebssicherheit der automatisierten Anlage hat, sind nähergehende Sicherheitsbetrachtungen vor diesem Hintergrund unbedingt erforderlich. Im Zuge der Sicherheitsbetrachtung gilt es die verwendeten Kommunikationsverfahren zu untersuchen und zu prüfen, wie sich Sicherheitsziele, wie Authentizität, Integrität, Vertraulichkeit oder Verfügbarkeit innerhalb des dezentralen Systems umsetzen lassen.

Die Basisdienste sind mit dem für eingebettete Systeme entwickelten *OASIS-Standard Web Services Discovery and Web Services Devices Profile (kurz: WS-DD)* realisiert worden. WS-DD nutzt das nachrichtenbasierte SOAP-Protokoll, welches der Anwendungsschicht zuzuordnen ist. SOAP basiert auf XML und kann über beliebige Kanäle wie HTTP oder direkt über UDP übertragen werden. Damit ist es möglich die Basisdienste über die Anwendungsschicht mit den Dienstnutzern zu verbinden.

Müssen zwischen Agent und Basisdienst bestimmte Sicherheitsziele umgesetzt werden, so kann je nach physikalischer bzw. netztopologischer Zuordnung von Basisdienst und Agent kein Sicherheitsdienst einer unterliegenden Schicht genutzt werden. Die Sicherheitsdienste müssen demnach auf der Anwendungsebene realisiert werden, d. h. die SOAP-Nachrichten müssen signiert und/oder verschlüsselt werden. Hierfür bietet sich der Einsatz von *WS-Security* an, ein Standard der Vorgaben zur Einbindung von digitalen Signaturen mittels *XML Digital Signature* und Verschlüsselungen mittels *XML Encryption* macht. Jeder Basisdienst von *Paket Royale* wird dazu mit einem Sicherheitsmodul ausgestattet, welches zum einen eine Sicherheitsrichtlinie für diesen Dienst definiert und zum anderen die Einhaltung dieser Richtlinie gewährleistet. In der Sicherheitsrichtlinie kann genau konfiguriert werden welche Sicherheitsziele für welche Teile eines angebotenen Dienstes zu erfüllen sind.

4 Erprobung

Zur Erprobung des hier vorgestellten Konzepts wurde mithilfe von *Paket Royale* eine Steuerung für eine reale Versuchsanlage entwickelt und anschließend unter verschiedenen Gesichtspunkten untersucht, wie z. B. der erzielbaren Reaktionszeit. Dabei wurden vor allem die Basisdienste und das Monitoring-System genauer betrachtet [FFH$^+$10, FLtH$^+$09]. Im Zuge dieses Beitrags soll ein besonderes Augenmerk auf

den Einfluss und die Anwendbarkeit von Sicherheitsmechanismen innerhalb des mit *Paket Royale* entwickelten Systems gelegt werden.

4.1 Versuchsanlage und Versuchsaufbau

Als Untersuchungsobjekt dient eine Versuchsanlage für die Beförderung von leichtem Stückgut. Die Streckenlänge beträgt ca. 120 m, aufgeteilt auf zwei miteinander verbundenen Ebenen (vgl. Abbildung 5). Anders als üblich, wird die Versuchsanlage nicht von einer SPS gesteuert. Stattdessen sind direkt an der Fördertechnik sieben IPCs verbaut, von denen jeder einen Teil der Anlage steuert. Dabei sind alle Sensoren und Aktoren eines Förderers stets an den gleichen IPC angeschlossen.

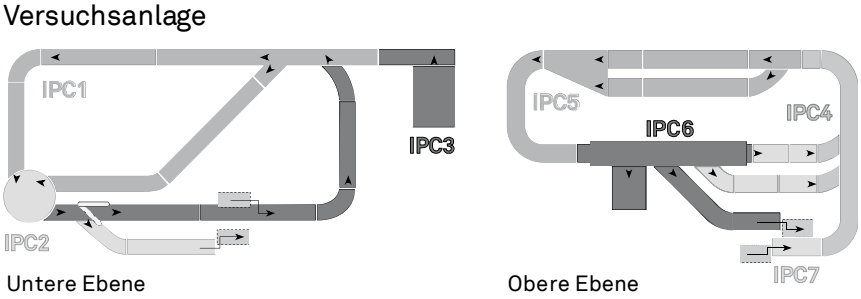


Abbildung 5: Topologie der Versuchsanlage

Als Messplattformen für die Versuche kommen die aktuell an der Anlage verbauten IPCs (IPC_A: 266 MHz Geode, 128 MB RAM) und ein neueres Modell (IPC_B: 1,4 GHz Pentium-M, 256 MB RAM) zum Einsatz. IPCs vom Typ IPC_B sollen künftig die IPC_A ersetzen. Auf den beiden IPCs wird ein Basisdienst samt Sicherheitsmodul konfiguriert und gestartet. Ein Testnotebook NB (2,8 GHz T9600, 3 GB RAM) spricht die Basisdienste per Ethernet zyklisch an. Als Laufzeitumgebung kommt auf dem IPC_B und dem Testnotebook die aktuelle *Java Runtime Environment* (JRE) zum Einsatz. Auf dem IPC_A dagegen läuft eine JRE speziell für den Einsatz auf ressourcenschwachen Geräten. Als Kryptographiebibliothek wurde *BouncyCastle* gewählt.

Um allgemeine Zeitschranken zu finden, wird in einem ersten Schritt die vollständige Dauer einer *Request-Response*-Nachricht gemessen. Dabei wird eine Eingabe vom Testnotebook an die Basisdienste auf den IPCs gesendet, welche eine passende Antwort zurückschicken. Ziel ist die Bestimmung des Einflusses von Signatur und Verschlüsselung auf die Antwortzeiten dieser Dienstoperation. Außerdem wird die Auswirkung von Signatur und Verschlüsselung auf die Nachrichtengröße betrachtet.

4.2 Ergebnisse

Zunächst wurde gemessen, wie viel Zeit die Ausführung einer Kryptooperation auf IPC_A benötigt. Als Testoperationen wurde eine RSA-Signatur mit einem 2048 Bit-X.509-Zertifikat, deren Überprüfung und die Berechnung eines SHA1-Hashwerts gewählt. Als Eingabe wurde ein SOAP-Body einer Ereignisnachricht mit einer Länge von 90 Bytes gewählt, da dies einen sehr häufigen Nachrichtentyp darstellt. Abbildung 6a zeigt das Ergebnis von 1000 Durchläufen als Boxplot (Whisker: 2,5% bzw.

97,5% Quantil). Hierfür benötigt IPC_A im Median 345 ms. Die Überprüfung einer RSA-Signatur ist hingegen deutlich schneller, gleiches gilt für die Berechnung eines SHA1-Hashwerts. Die Messungen zeigen, dass die Berechnung einer RSA-Signatur auf IPC_A im Verhältnis zur Dauer der Kommunikation (ca. 25 ms) [FLtH+09] zu zeitaufwändig ist.

Vergleichsweise wurde der neuere IPC_B untersucht. Abbildung 6b zeigt die Ergebnisse von 1000 Aufrufen des Basisdienstes. Hier wurde die komplette Umlaufzeit der Kommunikation (engl. *Roundtrip*) gemessen. In einem *Roundtrip* werden von Nutzer und Dienst je nach Variante zwei Signaturen und evtl. zwei Verschlüsselungen berechnet sowie die Überprüfungen und Entschlüsselungen ausgeführt. NB und IPC_B benötigen für einen normalen, ungesicherten Datenaustausch im Median 7 ms, bei verlangter Signatur 43 ms und bei zusätzlicher Verschlüsselung 71 ms.

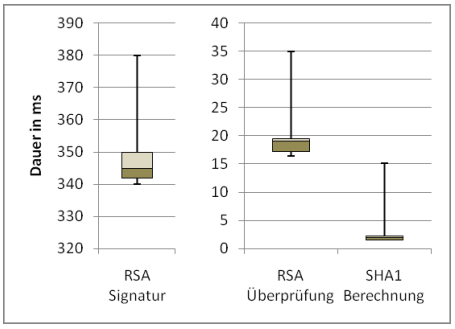


Abbildung 6a: Kryptooperationen auf IPC_A

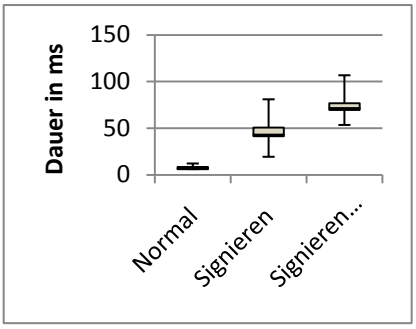


Abbildung 6b: Roundtripzeit zw. NB ↔ IPC_B

Eine Gegenüberstellung der Nachrichtengröße der Testnachricht (737 Bytes) ergibt eine mehr als 6-fach größere Nachricht (4506 Bytes) bei der Verwendung einer Signatur. Bei Verwendung von Signatur und Verschlüsselung vergrößert sich die Nachricht sogar um das 7,5-fache (5672 Bytes). Allein das angehängte X.509-Zertifikat ist mit 1681 Bytes mehr als doppelt so groß wie die Ursprungsnachricht. Es sei jedoch angemerkt, dass sich während der Tests nicht das Netzwerk, sondern die CPU als der limitierende Faktor herausstellte. Zusammengefasst haben Sicherheitsoperationen einen erheblichen Einfluss auf die Kommunikationsdauer. Für den Einsatz in dezentralen, echtzeitnahen Systemen ist die Geschwindigkeit der Kommunikation entscheidend. Folglich muss die Ausführungsdauer der Sicherheitsoperationen optimiert oder falls notwendig in einer höhergelagerten Systemebene realisiert werden.

5 Fazit

Im Zuge des Projekts *Paket Royale* wurde ein neues Steuerungskonzept entwickelt, welches eine SOA mit einem Multiagentensystem kombiniert. Durch den Einsatz von Basisdiensten aus einem standardisierten Baukasten wird eine herstellerunabhängige Schnittstelle zu den Gewerken umgesetzt. Dies lockert zum einen die enge Bindung von Anlagenbetreibern und -herstellern und zum anderen erleichtert diese modulare Sichtweise die Erweiterung einer Anlage. Die Steuerungslogik wird auf eigenständige Agenten verteilt, welche zur Laufzeit rekonfigurierbar sind. So können lange Stillstandszeiten vermieden werden, welche bei einer Modifikation eines zentralen

SPS-Programms unvermeidbar wären. Trotzdem sind für diesen Ansatz noch einige Fragen offen bzw. müssen tiefergehend beleuchtet werden. Dazu gehört u. a. die Entwicklung von durchgängigen Engineering-Methoden und Werkzeugen, die dabei helfen das volle Potenzial dezentraler, modularer Systeme auszuschöpfen. Außerdem gilt es ein Sicherheitskonzept zu erarbeiten, das bei akzeptablen Übertragungs- und Verarbeitungszeiten, das benötigte Maß an Datensicherheit gewährleistet.

Zusammenfassend lässt sich feststellen, dass der dezentrale und modulare Ansatz vielversprechend ist, da sich mit diesem hochflexible Systeme gestalten lassen. Als Folge kommen mittlerweile erste Produkte auf den Markt, welche die klassische Trennung zwischen Förder-, Automatisierungs- und Leittechnik zugunsten der Modulsicht aufgeben und somit auf dezentrale Steuerungen setzen.

Literatur

[CBC+09] Candido, G.; Barata, J.; Colombo, A. W.; Jammes, F.: SOA in reconfigurable Supply Chains: A research readmap. In: Engineering Applications of Artificial Intelligence 22 (6), p. 939-949. Elservier: 2009.

[FFH+10] Feldhorst, S.; Fiedler, M.; Heinemann, M.; ten Hompel, M.; Krumm, H.: Event-based 3D-Monitoring of Material Flow Systems in Real-Time. In: Proceedings of the 8th IEEE International Conference on Industrial Informatics (INDIN'10). Osaka, Japan: 2010.

[FLtH+09] Feldhorst, S.; Libert, S.; ten Hompel, M.; Krumm, H.: Integration of a Legacy Automation System into a SOA for Devices. In: Proceedings of the 14th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'09), Palma de Mallorca, Spain: 2009.

[GtH10] Günthner, W.; ten Hompel, M.: Internet der Dinge in der Intralogistik. Berlin, Heidelberg: Springer Verlag, 2010.

[KFO10] Krühn, T.; Falkenberg, S.; Overmeyer, L.: Decentralized Control for Small-Scaled Conveyor Modules with Cellular Automata. In: Proceedings of the 4th International Conference on Automation and Logistics (ICAL'10). Hong Kong, China: 2010.

[Lew01] Lewis, R.: Modelling control systems using IEC 61499 – Applying function blocks to distributed systems. London: IET, 2001.

[May09] Mayer, H. S.: Development of a completely decentralized control system for modular continous conveyors. Karlsruhe: Dissertation, Universität Karlsruhe (TH), 2009.

[MLR+10] Mendes, M. J.; Leitao, P.; Restivo, F.; Colombo, A.: Composition of Petri Nets Models in Service-oriented Industrial Automation. In: Proceedings of the 8th IEEE International Conference on Industrial Informatics (INDIN'10). Osaka, Japan: 2010.

[SF10] Sadowsky, V.; Feldhorst, S.: Paket Royale - Dezentrales Steuerungskonzept für das Internet der Dinge. In: Tagungsband zum 5. BVL Wissenschaftssymposium Logistik "Strukturwandel in der Logistik", S. 296-306. Darmstadt: 2010.