

Entwicklung, Umsetzung und Evaluierung eines dezentralen Lagerverwaltungssystems

Development, implementation and evaluation of a decentralized warehouse management system

Matthias Jung
Trung Thanh Le
Thomas Atz
Willibald A. Günthner

*fml – Lehrstuhl für Fördertechnik Materialfluss Logistik,
Technische Universität München*

Im Rahmen eines von der Deutschen Forschungsgemeinschaft (DFG) geförderten Projekts wird ein detailliertes Konzept, bestehend aus einer Softwarearchitektur und Steuerungsalgorithmen, für ein dezentrales Lagerverwaltungssystem (LVS) vorgestellt und evaluiert.

[Schlüsselwörter: dezentrale Lagerverwaltung, Bestandsverwaltung, verteilte Systeme, Softwarearchitektur]

In the context of a research project funded by the German Research Foundation (DFG) we designed a detailed concept for developing decentralized warehouse management systems. This concept is described and evaluated in this paper.

[Keywords: decentralized warehouse management, inventory control, distributed systems, software architecture]

1 EINLEITUNG

1.1 MOTIVATION FÜR DEZENTRALE LAGERVERWALTUNGSSYSTEME

Heute befinden sich produzierende Unternehmen, Logistikdienstleister und Handelsunternehmen in einem volatilen Marktumfeld, das u. a. durch hohen Konkurrenzdruck, schwer vorhersagbare Bedarfe und starke Konjunkturschwankungen gekennzeichnet ist. Um wirtschaftlich überleben zu können, müssen Unternehmen sich flexibel an neue Gegebenheiten anpassen.

Die Forschung hat sich diesen Herausforderungen gestellt und Lösungsansätze dazu entwickelt. Als erfolgversprechende Ansätze gelten modulare und dezentrale Systeme. Diese Systeme setzen sich aus vielen autonomen Subsystemen zusammen, die in Abwesenheit einer schwerfälligen, zentralen Steuereinheit selbstständig Ent-

scheidungen treffen und sich so neuen Gegebenheiten schnell anpassen können.

Wissenschaftliche Arbeiten haben diesen Ansatz vor allem im Kontext des Materialflusses untersucht (siehe [Lib11, Kuz10, May09]), im Bereich der Lagerverwaltung aber nur am Rande behandelt [Mir14]. Um diese Forschungslücke zu verkleinern legt [Mir14] durch systematische Untersuchung vorhandener Systeme zur Lagerverwaltung sowie Analyse des Potenzials einer dezentralen Lagerverwaltung einen theoretischen Unterbau.

Aufbauend auf [Mir14] werden eine Softwarearchitektur sowie Steuerungsalgorithmen für ein dezentrales LVS entwickelt. Sie dienen dazu Funktionen wie Ein- und Auslagern oder die Suche nach Waren und Lagerplätzen ohne eine zentrale Steuereinheit zu ermöglichen.

1.2 VORGEHENSWEISE

In Kapitel 2 wird der Stand der Technik zu dezentralen Systemen im Allgemeinen und Ansätzen dezentraler Lagerverwaltung im Besonderen vorgestellt. Die Anforderungen an die technische Umsetzung eines dezentralen LVS werden in Kapitel 3 aufgestellt. Darauf folgend wird ein allgemeines Konzept zur dezentralen Lagerverwaltung beschrieben. Hier wird auf den logischen Aufbau des Systems, die Nachrichtenübertragung und die systemrelevanten Funktionen eingegangen. In Kapitel 5 werden die Komponenten zur Konzeptumsetzung beschrieben. Des Weiteren wird auf die funktionalen Abläufe eingegangen, die nötig sind, um die systemrelevanten Funktionen zu realisieren. In Kapitel 6 findet auf Basis der Anforderungen und zu erfüllenden Funktionen eine Auswahl geeigneter Komponenten zur Konzeptumsetzung statt. Kapitel 7 beschreibt die Umsetzung und Evaluierung des Konzeptes. Der Beitrag schließt mit Zusammenfassung und Ausblick ab.

2 STAND DER TECHNIK

2.1 DEZENTRALE SYSTEME

Ein verteiltes System besteht aus mehreren unabhängigen Recheneinheiten, das den menschlichen Benutzern und externen Softwareprogrammen wie ein einzelnes System erscheint. Ein solches System ermöglicht u. a. auf verteilte Ressourcen leichter zuzugreifen, unterschiedliche Komponenten in das System zu integrieren, neue Komponenten hinzuzufügen oder vorhandene zu entfernen. Als Modell für ein solches verteiltes System wird üblicherweise ein Graph, bestehend aus Knoten und Kanten verwendet. Die Knoten des Graphen (engl. nodes) repräsentieren in der praktischen Umsetzung häufig Recheneinheiten und die Kanten (engl. edges) die Kommunikationsverbindungen dazwischen. In den 70er Jahren begannen Arbeiten zu dezentralen Systemen. Grundlegende Problemstellungen z. B. Synchronisation [Lam78], Bestimmung von globalen Zuständen in dezentralen Systemen [Lam82] sowie Datenkonsistenz [Lam74] wurden bereits früh erkannt. Heute unterscheidet man bei verteilten Systemen (engl. distributed systems) drei unterschiedliche Arten verteilter Systeme (distributed computing systems, distributed information systems und distributed pervasive systems). [Tan06]

Als Beispiel für ein dezentrales System gilt das Internet. Angefangen Ende der 1960er Jahre als ein Projekt des US-Verteidigungsministeriums entwickelte sich das – damals noch als ARPANET bezeichnete – Internet zu einem weltumspannenden Netzwerk von Rechnern. Mit diesem Netzwerk kann sich praktisch jeder Computer auf der Erde verbinden und Dienste wie E-Mail, Datenübertragung und Telefonie verwenden. [Mei04] Mit seinen verteilten Computersystemen ist das Internet zudem Voraussetzung für das Cloud Computing – ein Dienst, der das bedarfsgerechte Mieten von Software, Plattform und Infrastruktur ermöglicht. (vgl. [Jad12]) Die Nutzer eines solchen Dienstes bezahlen je nach Abrechnungsmodell z. B. nur für die tatsächlich verwendete Rechenleistung sowie den benötigten Speicherplatz und können zusätzliche Kapazitäten nachträglich buchen bzw. überflüssige Kapazitäten wieder kündigen. Ein Anbieter dieses Dienstes ist Amazon Web Services (AWS). Die beiden wichtigsten Komponenten von AWS heißen Amazon Elastic Compute Cloud (Amazon EC2) für die nutzerindividuelle Rechenleistung und Amazon Simple Storage Service (Amazon S3) für die bedarfsgerechte Datenspeicherung. [AWS14]

Google hat mit Spanner ein Datenbanksystem entwickelt, das als erstes verteiltes Informationssystem Daten im globalen Maßstab verteilt und extern einheitliche Transaktionen unterstützt. Darüber hinaus erlaubt es Mehrplatzbetrieb und repliziert synchron. Die Replikation erfolgt dabei selbst über Kontinente hinweg zuverlässig, so dass Spanner besonders in Fällen eingesetzt wird, die

eine hohe Datenverfügbarkeit erfordern, z. B. in großräumigen Katastrophengebieten. [Cor12]

Besteht das System (auch) aus mobilen Einheiten, handelt es sich um ein verteiltes, eingebettetes System (distributed pervasive system). Dieses Prinzip findet sich im Konzept der Smart Factory aus dem Forschungsprogramm „Industrie 4.0“ wieder. (vgl. [Bmb14]) Dort lässt sich beispielsweise das von der Daimler AG entwickelte Production 2000+ (P2000+) einordnen, welches im Bereich der Automobilherstellung für hochvolumige und flexible Produktion eingesetzt wird. Zu verarbeitende Werkstücke bieten Aufträge an, für die geeignete Werkzeugmaschinen wiederum Gebote abgeben können. Diejenige Maschine mit dem höchsten Gebot bekommt den Zuschlag für die Bearbeitung des Werkstücks. Dabei beeinflusst die momentane Arbeitsbelastung die Gebotsabgabe. Bei einer hohen Arbeitslast gibt die Maschine ein geringeres Gebot ab – ein vorgefertigter Produktionsplan wird nicht mehr benötigt. Stattdessen entsteht der Produktionsplan während des Betriebs und passt sich selbstständig und flexibel den Gegebenheiten an. [Bus01]

2.2 SYSTEME ZUR DEZENTRALEN LAGERVERWALTUNG

Ein zuverlässiges LVS zeichnet sich durch seine Datenkonsistenz und -verfügbarkeit aus. Bei einem dezentralen LVS müssen, aufgrund der verteilten Datenhaltung, diese Eigenschaften durch eine bestimmte Softwarearchitektur sowie Steuerungsmechanismen gewährleistet werden (vgl. [Mir14]; siehe auch Kapitel 4). Dazu zählen u. a. die Synchronisierung von verteilt liegenden Daten und die Reservierung von Ressourcen, um einen Zugriff durch eine andere verteilte Einheit zu verhindern. Die Beschreibung der technischen Abläufe, der Architektur sowie der Steuerungsmechanismen eines dezentralen Systems zur Lagerverwaltung sind in der Literatur nicht ausreichend vorhanden. Die gegenwärtig verfolgten Ansätze zur dezentralen Lagerverwaltung, werden in [Mir14] diskutiert:

- Das Projekt **WISHIS** basiert auf Sensorknoten, die an den Lagerplätzen angebracht sind. Auf diesen Knoten sind die Informationen über die eingelagerten Einheiten gespeichert. Der Zugriff auf diese Information erfolgt über ein Sensornetz (WSN). [Wie07]
- Das dezentrale Lagerverwaltungssystem von Arkadius **Schier** basiert auf Sensorknoten, die an den Ladehilfsmitteln (LHM) angebracht sind. Die Knoten dienen zur Ortung und Speicherung der Wareninformationen. Beide Systeme kommen aufgrund der verteilten Datenspeicherung ohne eine zentrale Datenbank aus. [Sch08]
- Ein **iBin** besteht aus einem Einsteckmodul mit integrierter Kamera und RFID. Dieses Modul wird an VDA-Behältern befestigt und dient der

bedarfsgerechten Auslösung von Bestellungen. [Wür13] Der **inBin** ist ein mit Grafikdisplay, Rechen- und Funkeinheit ausgestatteter Behälter. Nach Aussage der Entwickler dient er dazu Logistikprozesse zu kontrollieren und kann in der Kommissionierung eingesetzt werden. [IML13]

- **Smart Shelf** ist ein Regal mit integrierter RFID-Antenne. Es registriert, ob eine Ware aus dem Regal entnommen wurde und wird eingesetzt, um das Kundenverhalten zu studieren. (vgl. [Kat13, RFI09, Dec10]) Mit **Identplus** wird die Position der Ware indirekt über das Transportfahrzeug identifiziert. Dazu wird festgestellt wo sich das Fahrzeug befindet und ob es eine Ware geladen hat oder abstellt. Beim Abstellen der Ware wird aus der momentanen Fahrzeugposition und den Wareninformationen die Position der Ware bestimmt und im System abgelegt. [Kil13, Ide13]

Wichtige technische Beschreibungen dieser Forschungs- bzw. Entwicklungsarbeiten wie z. B. die eingesetzten Technologien, die Softwarearchitektur, die informationsverarbeitenden Abläufe usw. stehen nur teilweise oder gar nicht in der öffentlichen Literatur zur Verfügung. Zum einen wird aufgrund organisatorischer Abläufe (laufende Patentprüfung, Geschäftsgeheimnis, Wettbewerbsvorteilen usw.) von einer Veröffentlichung gänzlich abgesehen, zum anderen wird bei den bisherigen Veröffentlichungen der dezentrale Ansatz in den Projekten nicht durchgängig verfolgt. (vgl. u. a. [Mir14, Web12])

Tabelle 1 zeigt auf welche technischen Aspekte die einzelnen Projekte eingegangen sind. Grau unterlegte Spalten markieren dabei diejenigen Projekte, die eine dezentrale Bestandsverwaltung ermöglichen. Von diesen geht nur das Projekt von Schier auf Steuerungsmechanismen ein. Doch selbst bei Schiers dezentralem LVS werden nicht alle für ein LVS essentiellen Funktionen

unterstützt, z. B. fehlt die automatische Suche und die Reservierung von Lagerplätzen. [Sch08]

Die vorhandenen Projekte zeigen einen Mangel sowohl in der Darstellung informationsverarbeitender Abläufe als auch in der Beschreibung der eingesetzten Technologien dezentraler Systeme. Die vorliegende Arbeit geht daher auf diese Forschungslücke ein und beschreibt ein allgemeines Konzept für ein dezentrales LVS (siehe Kapitel 4 und 5), die Auswahl der nötigen Technologien (siehe Kapitel 6), die Implementierung eines konkreten und die Evaluierung (siehe Kapitel 7) eines allgemeinen dezentralen LVS.

3 ANFORDERUNGSANALYSE

3.1 ZIELGRUPPEN UND ANWENDUNGSFÄLLE FÜR DEZENTRALE LVS

Zusätzlich zu den allgemeinen Anforderungen, denen ein Lagerverwaltungssystem grundsätzlich genügen muss, ergeben sich im Detail je nach Anwendungsfall Unterschiede z. B. durch zusätzliche oder eingeschränkte Anforderungen und bei der Behandlung gegensätzlicher Ziele. In Betrieben, die aktuell erfolgreich und ohne große Einschränkungen ein zentrales Lagerverwaltungssystem einsetzen, besteht für eine Umrüstung auf eine dezentrale Lagerverwaltung zunächst kein Bedarf. Aus diesem Grund werden im Folgenden die Anforderungen solcher Unternehmen betrachtet, die prinzipiell besonders von einem dezentralen Lagerverwaltungssystem profitieren können – dies sind kleine Betriebe, die momentan kein LVS einsetzen und daher mit geringen Einstiegskosten für ein dezentrales Lagerverwaltungssystem von einer automatischen Lagerverwaltung profitieren können. Des Weiteren profitieren Unternehmen deren Lagerbestände starken Schwankungen unterliegen. Dadurch müssen entsprechende Änderungen bzw. Anpassungen (Customizing) am LVS vorgenommen werden, die mit gegenwärtigen Systemen zur Lagerverwaltung noch aufwendig sind (vgl. [Log08]).

Tabelle 1 Technische Funktionen dezentraler Lagerverwaltungsansätze

	WISHIS	Arkadius Schier	iBin	inBin	Smart Shelf	identplus
Konzeptbeschreibung	x	x	x		x	x
Technologienbeschreibung	x	x	(x)	(x)	x	x
Beschreibung der Softwarearchitektur		x				
Beschreibung der Steuerungsmechanismen		(x)			x	x
Entwicklung	x	x	x	x	x	x
Evaluierungsergebnisse		x				

3.2 ANFORDERUNGEN AN EIN DEZENTRALES LVS

3.2.1 GRUNDLEGENDE ANFORDERUNGEN

Mirlach definiert grundsätzliche Anforderungen an ein Lagerverwaltungssystem und unterscheidet dabei zwischen funktionalen und nicht-funktionalen Anforderungen [Mir14]. Die funktionalen Anforderungen sind grundsätzliche Anforderungen, die jedes Lagerverwaltungssystem erfüllen muss, um die vom System erwartete Funktionalität erzielen zu können. Die nicht-funktionalen Anforderungen müssen zwar prinzipiell auch erfüllt werden, allerdings besteht hier ein gewisser Spielraum bei der konkreten Umsetzung. So werden z. B. in Unternehmen die bisher kein Lagerverwaltungssystem verwendet haben, längere Antwortzeiten für eine Suche eher akzeptiert, da eine manuelle Suche in einem Aktenordner noch länger dauert. [Mir14]

Wichtige nicht-funktionale Anforderungen sind nach Mirlach et al. Konsistenz und Verfügbarkeit des Datenbestandes, Nachvollziehbarkeit der Bestände und Arbeitsabläufe sowie eine akzeptable Antwortzeit. Diese Anforderungen müssen in dezentralen Systemen auch bei parallelem Zugriff durch mehrere Akteure/Prozesse sichergestellt werden.

Aus den oben genannten Anforderungen lassen sich weitere, indirekte Anforderungen ableiten. Zur Gewährleistung von Konsistenz und Verfügbarkeit muss das dezentrale LVS auch gegenüber möglichen Systemstörungen robust sein. Systemstörungen können zufällig auftretende und versehentlich ausgelöste (Teil-)ausfälle oder gezielt herbeigeführte Angriffe sein. Die Sicherheitsaspekte in verteilten Systemen sind nach wie vor Gegenstand aktueller Forschung [Lya14, Him14, Tan14]. Die zur Abhärtung eines dezentralen LVS gegen diverse Angriffsarten (Sniffing, Spoofing, Replay-Attacken, Man-in-the-Middle-Attacken, Cloning & Emulation, Denial of Service (DoS), Tracking, Relay-Angriffe, Blocken) notwendigen Maßnahmen können im Rahmen dieser Veröffentlichung nicht umfassend beantwortet werden. Es sollte daher in Zukunft untersucht werden, welche Maßnahmen, die für den

Schutz verteilter Systeme entwickelt wurden, sich auf dezentrale LVS übertragen lassen.

3.2.2 ERGÄNZENDE ANFORDERUNGEN

Tabelle 2 beschreibt die zusätzlichen Anforderungen, die ergänzend zu den grundsätzlichen Anforderungen für die Umsetzung des Demonstrators getroffen wurden.

3.2.3 ABGESCHWÄCHTE ANFORDERUNGEN

Für jedes dezentrale System gilt zusätzlich das CAP-Theorem [Bre00], nach dem nur zwei der folgenden drei Anforderungen in verteilten Systemen garantiert werden können:

- 1) Konsistenz (Consistency): die Sicht aller Knoten auf alle Daten ist zum selben Zeitpunkt gleich.
- 2) Verfügbarkeit (Availability): Das System liefert immer eine Antwort.
- 3) Partitionstoleranz (Partition tolerance): Gehen Nachrichten verloren oder zerfällt das Netz in mehrere, nicht verbundene Teilnetze, arbeitet das Gesamtsystem dennoch (korrekt) weiter.

Vor diesem Hintergrund ist mit Einschränkungen bezüglich mindestens eines der genannten drei Kriterien in einem dezentralen LVS zu rechnen. Eine Implementierung eines dezentralen LVS sollte also versuchen die Auswirkungen der nicht vermeidbaren Einschränkungen in Bezug auf die in der Praxis relevanten Fälle so gering wie möglich zu halten.

Anhand von Beispielen lässt sich relativ leicht zeigen, dass Einschränkungen bezüglich dieser Kriterien nicht notwendigerweise zu großen Problemen in der Lagerverwaltung führen müssen:

- 1) Falls die Konsistenz der Stammdaten in einem dezentralen LVS nicht (immer) gewährleistet werden kann, ergeben sich daraus nur praxisrelevante Konsequenzen, wenn die nicht aktuellen Daten auch wirklich verwendet werden. Des Weiteren werden z. B. Stammdaten relativ selten und dann

Tabelle 2 Zusätzliche Anforderungen an die Umsetzung des Demonstrators

	Nebenbedingung	Begründung
Hardware	Energieverbrauch	Gering, damit ggfs. Stromversorgung per Batterie möglich
	Drahtlose Kommunikation	Keine Verkabelung notwendig, geringerer Infrastrukturaufwand, Nutzung auf Transportfahrzeugen möglich
	Anbindung von RFID-Readern möglich	Demonstration der Kombination dezentrales LVS inkl. automatischer Datenerfassung
Software	Plattformunabhängigkeit	Andere HW-Plattformen später nutz- und integrierbar, nur eine Software für unterschiedliche Geräte (heterogenes System aufwandsarm möglich)
Demonstrator	Präsentation/Datenausgabe an den Knoten	Darstellung/Demonstration der Vorgänge
	Einsatz auf Transportfahrzeugen	Funktionsintegration mit Staplerterminal
	Einsatz in einer Lagerhalle	Demonstration der Ergebnisse, Durchführung praxisnaher Tests

auch nur vereinzelt geändert – ein Knoten, der eine alte Version kürzlich aktualisierter Stammdaten verwendet macht also in der Regel einen relativ kleinen Fehler. Die daraufhin durchgeführten Aktionen wären nur kurze Zeit zuvor korrekt gewesen.

- 2) Kann die Verfügbarkeit des Gesamtsystems nicht garantiert werden, kann dennoch ein eingeschränkter Betrieb möglich sein, der zumindest bis zur Behebung des Problems häufig benötigte Funktionen bereitstellt. Fällt z. B. ein Knoten aus und ist dessen Datenbestand (vorübergehend) nicht verfügbar, können die übrigen Knoten viele Funktionen (z. B. Suche auf dem reduzierten Datenbestand, Auslagerung aus bzw. Einlagerung in Bereiche die von dem ausgefallenen Knoten nicht verwaltet werden usw.) weiterhin bereitstellen, obwohl das Gesamtsystem im Sinne des CAP-Theorems nicht (vollständig) verfügbar ist.
- 3) Zerfällt das dezentrale System in zwei oder mehr Partitionen, kann möglicherweise dennoch ein Teil der Partitionen weiterarbeiten (analog zu 2.). Gleichwohl muss zum Nachweis der Korrektheit des dezentralen LVS gezeigt werden, dass nur die vorausgesetzten Einschränkungen gelten, da sich sonst der Nutzen des Ansatzes nicht abschätzen lässt. Manche Funktionen, wie z. B. die Inventur, verlangen zwingend Konsistenz und Partitions-toleranz und können damit in keinem dezentralen LVS zu jedem beliebigen Zeitpunkt durchgeführt werden (Verletzung der Verfügbarkeit). Es ist daher vor Durchführung oder zumindest vor Abschluss derartiger Operationen die Verfügbarkeit zu prüfen.

4 KONZEPTBESCHREIBUNG

Dieses Kapitel widmet sich der Skizzierung des grundlegenden Konzeptes für ein dezentrales Lagerverwaltungssystem. Dabei werden vor allem notwendige informationstechnische Systeme und deren Fähigkeiten, Funktionen und Kommunikation untereinander zur Realisierung der Anwendung beschrieben. Dies lässt möglichst großen Freiraum bei der Umsetzung des Konzeptes – so kann die Topologie, die Hardwareauswahl und das Kommunikationsprotokoll abhängig von den eigenen Rahmenbedingungen ausgewählt und gestaltet werden.

4.1 ENTITÄTEN, FUNKTIONEN UND RELATIONEN

Entitäten repräsentieren in dem hier vorgestellten Konzept informationstechnische Systeme. Es gibt mehrere funktionale Ausprägungen dieser Systeme:

- Lagerbereichsknoten und I/O Knoten werden unmittelbar für die Realisierung des dezentralen LVS benötigt. Die Lagerbereichsknoten sind an unterschiedlichen Orten im Lager angebracht und werden zur Erfüllung von Datenverarbeitungsaufgaben für einen klar abgegrenzten Bereich eingesetzt. I/O Knoten sind in ihrer Funktion keinem festen Ort zugewiesen, sondern können prinzipiell an jedem geeigneten Ort für die Kommunikation mit dem gesamten dezentralen LVS eingesetzt werden. Zusätzlich zur ihrer Ein-/Ausgabefunktion können Sie dieselben Funktionen bereitstellen, die auch Lagerbereichsknoten anbieten.
- Datenerfassungsknoten werden hingegen dafür verwendet, Eingaben aus (teil-) automatischen Datenerfassungstechnologien an das LVS anzubinden (z. B. RFID-Gates oder Barcode-Reader). Eine Analyse geeigneter Technologien für den kombinierten Einsatz mit dezentralen LVS wird in [Mir14] vorgestellt. Diese Knoten übernehmen keine Verwaltungsfunktionen. Es sind sowohl mobile Versionen als auch stationäre Einheiten denkbar. Die Ausprägung orientiert sich an der Mobilität des jeweils verwendeten Datenerfassungsgerätes.

Tabelle 3 liefert eine grobe Abschätzung welche Entität wie oft in einem dezentralen LVS benötigt wird.

4.2 AUFGABEN

Die Kommunikation bzw. der Nachrichtenaustausch der Entitäten untereinander zur Umsetzung der LVS-typischen Funktionen wird in Form von Beziehungen (Relationen) definiert. Diese Beziehungen können in benutzerbeeinflusste Aufgaben und systeminterne Aufgaben eingeteilt werden. Alle Aufgaben in einem dezentralen LVS müssen mindestens einer Entität zugeordnet sein, d. h. durch diese ausgeführt werden können. Die benutzerbeeinflussten Aufgaben entsprechen den Aufgaben, die auch ein zentrales LVS bereitstellen muss, also denjenigen Funktionen die vom LVS nach außen bereitgestellt werden. Systeminterne Aufgaben sind hingegen solche Funktionen, die zur Erfüllung der

Tabelle 3 Ungefähr benötigte Anzahl nach Art der Entität

Knotenart	Einflussgrößen auf die Anzahl der Knoten		Benötigte Anzahl (Größenordnung)
	stationär	mobil	
Lagerbereichsknoten	Anzahl Lagerbereiche	-	10 - 100
I/O Knoten	1	Anzahl Flurförderzeuge + Anzahl Kommissionierer	1 - 40
Datenerfassungsknoten (optional)	Anzahl I-Punkte	Anzahl mobile Erfassungsgeräte	2+

oben genannten Funktionen notwendig, aber für den Nutzer nicht direkt sichtbar sind. (vgl. Tabelle 4).

4.3 NACHRICHTEN

Die Nachrichten (Relationen, s.o.), die zwischen den Entitäten ausgetauscht werden, können analog zu den zu verrichtenden Aufgaben in benutzerbeeinflusste Nachrichten und systeminterne Nachrichten unterteilt werden.

Benutzerbeeinflusste Nachrichten dienen zur Erfüllung benutzerbeeinflusster Aufgaben und enthalten Anfragen oder Antworten zu Stammdaten, Bewegungsdaten oder Bestandsdaten [Mir14].

Systeminterne Nachrichten werden zur Erfüllung systeminterner Aufgaben benötigt und erlauben u. a. den reibungslosen Netzbetrieb, die Bestimmung der verfügbaren Kommunikationspartner und die Synchronisierung von Daten zwischen Knoten.

4.4 DATEN

Wie bereits oben beschrieben gibt es unterschiedliche Arten von Daten: Stammdaten, Bewegungsdaten und Bestandsdaten.

Die Stammdaten bestehen aus Material- und Lagerstammdaten. Materialstammdaten beschreiben das Sortiment der im Unternehmen verwendeten Artikel bzw. Materialien und können beispielsweise Abmessungen und Gewicht, besondere Lageranforderungen (Tiefkühlware, Gefahrgut), zugewiesene Lagerplätze, Melde-

bestand usw. umfassen. Optionale, spezielle Kennzeichen weisen zusätzliche Eigenschaften oder semantische Informationen zur Ware aus. Die Lagerstammdaten sind das logische Abbild der in einem Betrieb vorhandenen Lager- bzw. Stellplätze. Sie umfassen Informationen über die verschiedenen Lagerbereiche, Lagertypen sowie die Anzahl und Anordnung der Stellplätze. Zusätzlich können weitere Eigenschaften hinterlegt werden. [Mir14]

Bestandsdaten umfassen die Anzahl und Art der im Verwaltungsbereich des LVS vorhandenen Güter. Die Art der Güter wird meist über die Sachnummer als Referenz auf den entsprechenden Stammdatensatz abgebildet. Die Mengenangabe kann je nach Materialart als Stückzahl, Gewichts- oder Volumenangabe realisiert sein. Zusätzlich müssen auch Informationen über den jeweiligen Lagerort verwaltet werden, also eine Positionsangabe für jede Lagereinheit, die den Lagermitarbeitern und ggf. den automatisierten Lagereinrichtungen das Wiederauffinden der Güter ermöglicht. [Mir14]

Bewegungsdaten repräsentieren Auftragsdaten, die zur Abwicklung der Geschäftsprozesse notwendig sind. Externe Auftragsdaten sind Aufträge, die außerhalb des eigentlichen Lagerbetriebs entstehen und Prozessabläufe im Lager anstoßen (z. B. Kundenbestellungen). Auftragsdaten sind intern, wenn sie im Zuge der lagerinternen Prozessabläufe generiert werden (z. B. ein Kommissionierauftrag) und der Koordination der tatsächlichen Arbeitsabläufe im Lager dienen.

Tabelle 4 Kommunikationsteilnehmer bei Ausführung der operativen und systeminternen Funktionen

Systeminterne Funktionen	Teilnehmer an der Kommunikation
Abwicklung der Kommunikation zwischen den Entitäten (Übermittlung von Nachrichten, Empfangsbestätigungen usw.)	
Verwaltung der Liste aller Entitäten und Zuordnung spezieller Ansprechpartner	
Bereitstellung gemeinsam genutzter Daten	
Synchronisierung der Daten	
Operative Funktionen	Teilnehmer an der Kommunikation
Einlagern	
Auslagern	
Reservieren	
Suche	
Inventur	
Auftrag erteilen	
Auftrag löschen	
Auftrag sperren	
Auftrag annehmen	

Prinzipiell gibt es noch weitere, systeminterne Daten, die vom Netzwerk genutzt werden, um bestimmte Anforderungen zu erfüllen. Das Vorhandensein und die Ausprägung dieser Daten ist aber stark von der Implementierung abhängig und kann daher im Konzept nur als „sonstige Daten“ beschrieben werden.

4.5 VOR- UND NACHTEILE

Die Vor- und Nachteile, die das Konzept gegenüber zentralen LVS bietet, sind in Tabelle 5 aufgelistet.

Tabelle 5 Vor- und Nachteile des vorgeschlagenen Konzeptes

Vorteile	Nachteile
<ul style="list-style-type: none"> • Skalierbarkeit: Zusätzliche Lagerbereiche werden durch zusätzliche Knoten verwaltet. • Robustheit: Ausfall eines Teilsystems führt nicht zwangsläufig zu vollständigem Ausfall des Gesamtsystems. • Vereinfachte Lagerorganisation: Verschiebung eines Lagerbereiches wird im LVS durch Verschiebung des zuständigen Knotens erreicht. • Ware und Information sind lokal gespeichert und verfügbar. 	<ul style="list-style-type: none"> • Zusätzlicher Kommunikationsaufwand • Weitere Nachteile können abhängig von der konkreten Umsetzung entstehen.

5 KONZEPTENTWICKLUNG

In diesem Kapitel geht es um die Entwicklung der Details des in Kapitel 4 beschriebenen Konzeptes. Es werden die Randbedingungen festgelegt und konkrete Abläufe der generellen Operationen definiert. Die Hardware und das zugrunde liegende Kommunikationsmedium sind davon allerdings noch unabhängig und werden daher für den Demonstrator erst in späteren Kapiteln spezifiziert.

5.1 ROLLEN

Jede Instanz der in Kapitel 4 beschriebenen Knotentypen (Lagerbereichsknoten, I/O Knoten und Datenerfassungsknoten) wird als eigenständige Recheneinheit realisiert. Die zur Erfüllung der Funktion des jeweiligen Knotens notwendigen logischen Programme werden auf den Recheneinheiten selbst ausgeführt.

Alle Knoten werden durch eine eindeutige ID gekennzeichnet, welche den Knoten im Netzwerk identifiziert. Ein Verzeichnisdienst (siehe Abschnitt 5.2) stellt die Liste der IDs aller aktuell im Netzwerk aktiven Knoten bereit und erfüllt damit u. a. eine ähnliche Funktion wie ein Directory Facilitator in einem JADE-basierten Multiagentensystem [Bel07]. Somit kann durch Anfrage an diesen Dienst festgestellt werden, welche Knoten aktiv bzw. hinzugekommen oder nicht mehr verfügbar sind. Zusätzlich sorgt der Verzeichnisdienst dafür, dass er stets die aktuellste Version der Stammdaten besitzt, damit alle übrigen anfragenden Knoten prüfen können, ob ihre lokale Kopie der Stammdaten noch aktuell ist. Tabelle 6 gibt einen groben Überblick, welche Daten auf welchen Entitäten abgespeichert werden.

Tabelle 6 Speicherorte der einzelnen Daten

Daten	Speicherort	Zuständigkeit
Stammdaten	Alle Knoten	Verzeichnisdienst
Auftragsdaten	Alle I/O Knoten	Verzeichnisdienst
Bestandsdaten	Zuständiger Lagerbereichsknoten	Zuständiger Lagerbereichsknoten
Netzdaten	Ein Knoten	Verzeichnisdienst

5.1.1 LAGERBEREICHSKNOTEN

Die Lagerbereichsknoten sind im unmittelbaren Umfeld des von ihnen verwalteten Lagerbereichs und der darin enthaltenen Waren und Lagerplätze angebracht. Die wesentliche Funktion dieser Knotenart ist die Verwaltung der Bestandsdaten aus dem Lagerbereich und die Beantwortung von Anfragen zu diesen Bestandsdaten. Jeder Lagerbereichsknoten ist somit für einen bestimmten, räumlich begrenzten und konkret festgelegten Bereich zuständig. Diese Zuständigkeit muss bei jedem Lagertyp hinterlegt werden, beispielsweise durch Angabe der Position eines Lagerfachs in einem Hochregallager oder durch Definition eines Lagerbereiches in einem Blocklager. Diese Knoten bilden somit das Rückgrat des dezentralen Lagerverwaltungssystems.

Der Lagerbereichsknoten kann folgende Informationen liefern:

- welche Ware ist an welchem belegten Lagerplatz eingelagert,
- welche freien Lagerplätze stehen zur Verfügung und welche Attribute haben diese Lagerplätze bzw. eignen sie sich für die Einlagerung einer Ware gemäß deren Anforderungen.
- Ist der Knoten für den angefragten Lagerplatz zuständig (bestätigen oder verneinen)?

Eine sehr häufig vorkommende Operation im LVS ist die Suche nach Waren (entweder zur Bestimmung

des Warenbestands oder zur Vorbereitung weiterer Operationen zur Warenbewegung (z. B. Ein- und Auslagerung). Ein Lagerbereichsknoten muss daher verschiedene Optionen zur Suche nach Waren oder Lagerplätzen unterstützen, die eine möglichst flexible und effiziente Suche ermöglichen:

- Zu einer flexiblen Suche nach Waren gehört eine Suche nach allen Warenattributen (Stammdaten und Inventardaten). Die Verwendung von Wildcards oder regulären Ausdrücken [ope14b] erleichtert die Suche nach nicht genau bekannten Attributwerten und erlaubt gezielt einen gewissen Suchbereich zu definieren. Für die Suche nach geeigneten Lagerplätzen sind stattdessen die Attribute der Lagerstammdaten relevant, ansonsten gelten die gleichen Anforderungen wie für die Suche nach Waren.
- Eine Verknüpfung mehrerer Suchkriterien mittels UND bzw. ODER gestattet eine eingeschränkte bzw. gröbere Definition der gewünschten Ergebnismenge.
- Zwecks Reduktion der Ergebnismenge ist es sinnvoll die Zahl der Ergebnisse je Knoten optional einschränken zu können. Wird die Ergebnismenge durch jeden Knoten nach in der Suchanfrage definierten Kriterien sortiert übermittelt, kann sichergestellt werden, dass sich die relevanten Ergebnisse stets in der reduzierten Menge befinden. Eine geeignete Datenbanksoftware kann auf dem Lagerbereichsknoten selbst komplexe Suchvorgänge, aufgrund der geringeren Anzahl Datensätze je Sensorknoten, schnell beantworten. Die von allen Knoten bereits vorsortierte Ergebnismenge kann der anfragende Knoten effizient mittels Minimum-Comparison-Merging [Knu97] zu einer sortierten Gesamtmenge zusammensetzen.
- Falls bei der Kommunikation Paketverluste auftreten können, muss ein Lagerbereichsknoten eine ihn betreffende Anfrage grundsätzlich auch dann beantworten (= ablehnen), wenn die Ergebnismenge leer ist, da semantisch zwischen einer leeren Antwort und einer nicht erteilten bzw. verloren gegangenen Antwort unterschieden werden muss.

Damit der Datenbestand stets der Realität entspricht, aktualisieren die Lagerbereichsknoten nach jeder gemeldeten Zustandsänderung die von der Änderung betroffenen Datensätze. Veränderungen können sich durch verschiedene Funktionen des LVS ergeben:

- Ein- und Auslagerungen, haben zur Folge, dass sich der Belegungs- oder Reservierungs-

zustand der betroffenen Lagerplätze verändert und entsprechend aktualisiert werden muss. Neu eingelagerte Waren werden dem Warenbestand hinzugefügt, ausgelagerte Waren aus dem Warenbestand entfernt. Reservierungen durch I/O Knoten für Lagerplätze und Waren muss der Lagerbereichsknoten ebenfalls laufend aktualisieren und zwar entweder auf Anforderung des reservierenden Knotens hin oder selbstständig nach Ablauf der zeitlichen Befristung der Reservierung. Reservierungen sind notwendig um die Konkurrenz mehrerer Anfragen bezüglich ein und derselben Ressource (Ware oder Lagerplatz) auflösen zu können. Den konkreten Ablauf der Reservierung liefert Abschnitt 5.3.2.

- Falls sich die Lagerstruktur (vorübergehend) ändert, müssen auch die Verwaltungsinformationen geändert werden. Eine Sperrung oder Löschung bestehender Lagerplätze aus dem Bestand muss daher möglich sein. Ein Lagerplatz darf nur gesperrt oder gelöscht werden, wenn dieser gerade keine Waren enthält. Für eine Sperre muss der Lagerbereichsknoten also auch Aufträge generieren können, die die Umlagerung bestehender Ware in andere Lagerplätze oder -bereiche erlauben. Umgekehrt ist das Hinzufügen zusätzlicher Lagerplätze vorzusehen für den Fall, dass ein Lagerbereich erweitert wird. Die Auflösung eines kompletten Lagerbereichs kann dann als Auflösung sämtlicher seiner Lagerplätze modelliert werden. Ein Lagerbereich wird angelegt, indem ein Lagerbereichsknoten, auf dem die zu verwaltenden Lagerbereiche inkl. der Lagerplätze vorab definiert sind, in das Kommunikationsnetz integriert wird.

Die Speicherung aller auf den Lagerbereichsknoten abgelegten Datensätze des LVS muss auf einem nichtflüchtigen Datenspeicher möglichst ohne flüchtigen Puffer erfolgen, damit die Daten einen freiwilligen oder unfreiwilligen Neustart der Recheneinheit überdauern.

5.1.2 I/O KNOTEN

Die I/O Knoten werden zur Dateneingabe und -präsentation genutzt, können also Nutzereingaben in Anweisungen für das LVS umsetzen und Ergebnisse von Anfragen (z. B. einer Suche) oder Anweisungen an Mitarbeiter (z. B. Staplerfahrer/Kommissionierer) grafisch darstellen. Letztlich lassen sich prinzipiell alle operativen Funktionen von einem I/O Knoten aus nutzen. Damit diese Knoten „unterwegs“ von Kommissionierern bzw. Fahrern von Flurförderzeugen (Gabelstapler, Routenzug usw.) genutzt werden können, müssen sie mobil sein und damit über eine autarke Stromver-

sorgung und eine drahtlose Kommunikationsverbindung zu den anderen Knoten verfügen. Zur Anbindung eines ERP-Systems an das dezentrale LVS wird ebenfalls ein (stationärer) I/O Knoten benötigt. Typische Anwendungsfälle für I/O Knoten sind:

- Mobile Recheneinheit zur Unterstützung der Kommissionierung mit einer digitalen Pickliste.
- Terminal am Flurförderzeug mit Arbeitsanweisungen für den Fahrer.
- Recheneinheit zur Erzeugung von Aufträgen, Durchführung von Inventuren, Konfiguration der übrigen Knoten, Visualisierung von statistischen Daten und der Anbindung des LVS an andere IT-Systeme. Für diese Anwendung ist nicht unbedingt ein eigenständiger Rechner erforderlich, stattdessen können auch ohnehin benötigte Mitarbeiter-PCs mit der für einen I/O Knoten benötigten Software ausgestattet und somit in das dezentrale LVS integriert werden.

Alle I/O Knoten übernehmen zusätzlich Aufgaben der Auftragsverwaltung. Dazu ist insbesondere die Möglichkeit zur Erzeugung, Stornierung, Annahme und Bestätigung von Aufträgen relevant.

5.1.3 DATENERFASSUNGSKNOTEN

Datenerfassungsknoten sind ebenfalls Teil des dezentralen Systems und werden zur automatischen Identifizierung auf Basis der RFID-UHF Technologie verwendet. Sie bestehen aus einem RFID Lese-Schreibgerät, das an einen anderen Lagerbereichs- oder I/O Knoten angeschlossen ist. Für diese Recheneinheiten müssen die benötigten Gerätetreiber für das Lese-Schreibgerät verfügbar sein. Prinzipiell sind beliebige, zum Reader kompatible Recheneinheiten denkbar. Die RFID-Transponder, die vom RFID Lese-Schreibgerät erfasst werden können, werden wahlweise an den Waren oder den Ladungsträgern angebracht, je nachdem wie feingranular die Datenerfassung erwünscht ist. Unter Zuhilfenahme der RFID-UHF Technologie kann die Identität der Ware bei der Auftragsbearbeitung überprüft werden und somit die Transparenz und Qualität der Prozessabläufe gesteigert werden [Gün09].

Die Erfassung der RFID-Tags erfolgt wie bei Einsatz zusammen mit einem zentralen LVS, eine Umstellung ist nicht notwendig. Die vom RFID-Reader erfassten Lese-Events werden vom Datenerfassungsknoten an den Knoten weitergeleitet, für den das Event bestimmt ist – dies ist der Lagerbereichsknoten, der aktuell für den Lagerbereich zuständig ist, in dem das Event erfasst wurde. Folgende Szenarien sind vorzusehen:

1. Ein RFID-Gate erfasst alle Ein- und Auslagerungen in einem Lagerbereich. In diesem Fall ist die Zuordnung Datenerfassungsknoten zu Lagerbereichsknoten statisch und kann bei Bedarf fest konfiguriert werden. Ansonsten ist die Zuordnung dynamisch zu bestimmen (s. u.). Alternativ kann ein Datenerfassungsknoten auch selbst als (zuständiger) Lagerbereichsknoten fungieren.
2. Bei einem RFID-Gate wird der Lesevorgang manuell durch einen mobilen I/O Knoten gestartet und der Datenerfassungsknoten diesem I/O Knoten vorübergehend zugeordnet. Ein möglicher Anwendungsfall ist der Wareneingang: ein Mitarbeiter übernimmt mit einem Transportmittel (z. B. Handgabelhubwagen) eine angelieferte Ware, liest alle RFID-Tags aus und lagert die Ware nach einer Qualitätsprüfung ein.
3. Ein am Gabelstapler angebrachter oder vom Kommissionierer transportierter RFID-Reader bewegt sich durch das Lager und erfasst die Ein- und Auslagerungen an beliebigen Orten. In diesem Fall wechselt entweder die Zuständigkeit ortsabhängig und es muss dynamisch die Zuständigkeit unter Angabe der Ortsinformationen bestimmt werden, oder der Datenerfassungsknoten ist dem I/O Knoten auf dem Transportmittel fest zugeordnet.

Ein Datenerfassungsknoten kann die Zuständigkeit des Lagerbereichsknotens auf mehrere Arten bestimmen. Aufgrund der räumlichen Nähe der Lagerbereichsknoten zu ihrem zuständigen Bereich genügt bei einer physischen Änderung (z. B. Ein-/Auslagerung) in diesem Bereich eine Anfrage an alle direkt erreichbaren Lagerbereichsknoten nach der Zuständigkeit, um den richtigen Knoten zu finden. Die benötigten Ortsinformationen liefern entweder der Auftrag (bzw. bei einem RFID-Gate implizit dessen Position) oder sie müssen durch weitere Informationsquellen (z. B. Barcode oder RFID-Tag am Lagerfach) oder Sensorik zur Lokalisierung [fml14] bestimmt werden.

5.2 DATENHALTUNG

Der einfacheren Darstellung halber beschränkt sich das Konzept auf die für den Betrieb eines dezentralen LVS unbedingt notwendigen Daten. In der Praxis wird ein LVS für einen sinnvollen Einsatz im Unternehmen zusätzliche Daten benötigen, daher kann das Datenmodell ohne Einschränkungen um weitere Datensätze und Datenfelder erweitert werden, wie sie in der Praxis auch in zentralen LVS genutzt werden.

5.2.1 STAMMDATEN

Ein Lagerbereichsknoten wird durch seine ID, die Netzwerkadresse, identifiziert und verfügt zusätzlich über die Lagerstammdaten und die Materialstammdaten. Die Lagerstammdaten bestehen aus dem Lagertyp, dem Ort des Lagers in der Ebene (x, y-Koordinaten), der Anzahl und Dimension der Lagerplätze (x, y, z-Koordinaten) und einer Negativliste von Waren, die im verwalteten Lagerbereich nicht eingelagert werden dürfen.

Die Materialstammdaten enthalten für jede Ware die Waren-ID (Produktnummer), einen Bezeichner für die Art der Ware und die Abmessungen der Ware (Länge, Höhe, Breite).

Waren werden durch ihre Produktinformationen, also den Stammdatensatz zum Produkt sowie den Preis des Produktes gekennzeichnet.

Geschäftsinformationen, also die Informationen zu den Geschäftspartnern (Kunde, Lieferant) bestehen aus einer internen ID, die den Geschäftspartner innerhalb des Unternehmens oder einer Unternehmensgruppe eindeutig kennzeichnet und Kontaktinformationen sowie sonstigen Information zum Geschäftspartner.

5.2.2 AUFTRÄGE UND AUFTRAGSDATEN

Ein Auftrag entspricht einer Arbeitsanweisung für Mensch oder Maschine und enthält die Auftragsdaten, wie Auftragsnummer, Informationen zur gelieferten/zu transportierenden Ware und Informationen zum Liefer- bzw. Transportprozess. Ein Produkt-Auftrag ordnet einer Produktinstanz einen Auftrag und eine eindeutige ID, die auf dem zugehörigen RFID-Chip gespeichert ist, zu. Die Auftragsdaten werden vom Verzeichnisdienst verwaltet und zusätzlich auf allen I/O Knoten gespeichert. D. h. jede Änderung an den Auftragsdaten muss auf die aktuell erreichbaren I/O Knoten propagiert werden, um die Daten bei Ausfall des Verzeichnisdienstes rekonstruieren zu können. Um den jeweils aktuellen Stand eines Auftrags erkennen zu können benötigt jeder Auftrag zusätzlich den Zeitstempel der letzten Statusänderung. Für die Auswahl eines neuen Auftrags kann ein I/O Knoten daher seinen eigenen Bestand als Grundlage verwenden und sich dann an den Verzeichnisdienst wenden, um einen bestimmten Auftrag anzunehmen. Ist dieser

zwischenzeitlich vergeben, lehnt der Verzeichnisdienst die verspätete Annahme zu Vermeidung von Doppelbearbeitung ab. Von besonderer Bedeutung ist daher der aktuelle Zustand eines Auftrags, der Auftragsstatus. Er kann einen der vier Werte aus Tabelle 7 annehmen.

5.2.3 INVENTARDATEN

Die (dynamischen) Inventardaten, im Wesentlichen sind das die eingelagerten Warendatensätze und die Zuordnung der Ware zu den entsprechenden Lagerortsinformationen (Lagerposition innerhalb des Lagerbereichs in x, y, z-Koordinaten), verwaltet der jeweils zuständige Lagerbereichsknoten. Anfragen nach geeigneten Lagerplätzen und bereits eingelagerten Waren können dementsprechend, bezogen auf den verwalteten Lagerbereich, durch diesen Knoten beantwortet werden.

5.3 FUNKTIONEN DES DEZENTRALEN LVS

Die von einem dezentralen LVS bereitzustellenden Funktionen lassen sich in zwei Kategorien einordnen: operative Funktionen und systeminterne Funktionen (vgl. Abschnitt 4.2). Soweit möglich sollte sich die Schnittstelle der operativen Funktionen bei einem zentralen gegenüber einem dezentralen LVS nicht unterscheiden. Wenn für Außenstehende die Realisierung entsprechend gekapselt ist, ist eine Anbindung eines dezentralen LVS an die übrigen IT-Systeme im Unternehmen mit dem gleichen Aufwand verbunden wie bei einem zentralen LVS. Für die systeminternen Aufgaben ist dies hingegen nicht relevant, da diese nur innerhalb des Systems genutzt werden.

5.3.1 SYSTEMINTERNE PROZESSABLÄUFE IM DEZENTRALEN LVS

Das dezentrale System besteht aus funktional gleichartigen Knoten, von denen jeder dieser Knoten prinzipiell jede Aufgabe, die innerhalb des Systems zu erledigen ist, ausführen kann. In Bezug auf die zur Verfügung gestellten Dienste eines Knotens unterscheiden sich je zwei Knoten also lediglich durch ihren aktuellen Zustand bzw. ihre Konfiguration. Dies hat im Wesentlichen den Vorteil, dass Ausfälle einzelner Knoten, die Verwaltungsaufgaben erfüllen,

Tabelle 7 Mögliche Zustände eines Auftrags

Status	Bedeutung	Konsequenz
Ausstehend	Auftrag noch nicht in Bearbeitung	Initialzustand für jeden neuen Auftrag: der Auftrag wird allen I/O Knoten zur Annahme angeboten
Angenommen	Auftrag ist einem Bearbeiter zugewiesen	Ein I/O Knoten hat den Auftrag akzeptiert und wird ihn zeitnah bearbeiten
Erfolg	Auftrag wurde erfolgreich ausgeführt	Auftrag ist abgeschlossen
Fehler	Ausführung des Auftrags war nicht erfolgreich	Auftrag ist noch nicht abgeschlossen, eine (manuelle) Fehlerbehandlung ist notwendig

leichter kompensiert werden können, da genügend Ersatzknoten zur Verfügung stehen. Als Modell für das zugrunde liegende Kommunikationsnetz wird ein zusammenhängender, ungerichteter Graph angenommen. Eine hinreichend genaue Synchronisation der Uhrzeit auf allen Knoten wird vorausgesetzt und kann in einem dezentralen System z. B. mit dem von Ebner et al. beschriebenen Verfahren [Ebn04] umgesetzt werden.

Im Folgenden werden die zum Betrieb des Netzes notwendigen Aufgaben (systeminterne Funktionen) im Detail erläutert.

5.3.1.1 VERWALTUNG VON NETZINFORMATIONEN

Folgende Informationen werden zur Verwaltung des Netzes benötigt:

- 1) Eine Liste der aktiven und der bekannten aber gegenwärtig inaktiven Knoten. Diese besteht im Wesentlichen aus einer Tabelle, in der für alle bekannten Knoten die wichtigsten Statusinformationen gespeichert sind. Dies sind u. a.:
 - a) der Knoten ist aktiv/inaktiv,
 - b) die Netzwerkadresse jedes aktuell erreichbaren Knotens,
 - c) der Typ des Knotens, der seine aktuellen zur Verfügung gestellten Dienste beschreibt und
 - d) die Aktualität des Listeneintrags in Form eines Zeitstempels.
- 2) Die aktuellste Version der Stammdaten.
- 3) Eine Liste aktuell vergebener und verfügbarer Aufträge.

Um nicht unnötig einen single point of failure [Doo02, MIL79] zu schaffen, wird die Aufgabe des Verzeichnisdienstes von einem beliebigen Knoten übernommen und nicht an einen zentralen Dienst delegiert. Da prinzipiell jeder Knoten als Verzeichnisdienst fungieren kann, aber gleichzeitig aus Gründen der Konsistenz nur ein Knoten zur gleichen Zeit Verzeichnisdienst sein soll, muss sichergestellt werden, dass nur ein Verzeichnisdienst aktiv ist und nur bei dessen Ausfall oder wenn der Dienst nicht erreichbar ist, ein anderer Knoten seine Funktion übernimmt.

Theoretisch wäre es auch möglich die Verzeichnisisinformationen bei Bedarf durch jeden Knoten neu generieren oder aktualisieren zu lassen, dies wird jedoch aufgrund der erhöhten Netzwerklast durch die mehrfach durchzuführenden Broadcastanfragen verworfen. Der Verzeichnisdienst führt stattdessen die notwendigen Anfragen durch und stellt die Informationen zur Verfügung. Zur Abfrage des aktuellen Netzstatus sind daher nur die sehr viel effizienteren Unicast-Abfragen notwendig [Dal11].

Folgende Szenarien sind in der Umsetzung berücksichtigt:

- Das Auffinden eines Verzeichnisdienstes.
- Die Erzeugung eines neuen Verzeichnisdienstes, wenn noch kein Verzeichnisdienst existiert oder der ursprüngliche Verzeichnisdienst ausfällt und kein Ersatz gefunden werden kann.
- Die Auflösung eines redundanten Verzeichnisdienstes bei Verbindung zweier Netzwerkbereiche (z. B. nach Ausfall und späterer Instandsetzung eines wichtigen Verbindungsknotens).

5.3.1.2 KONTAKTIEREN EINES VERZEICHNISDIENSTES

- 1) Ist aus der Vergangenheit ein Verzeichnisdienst bekannt, wird zunächst angenommen, dass der entsprechende Knoten diese Funktion immer noch wahrnimmt und der bekannte Knoten befragt ob er diese Funktion noch ausübt. Übt der befragte Knoten diese Funktion noch aus, ist ein Verzeichnisdienst gefunden und entsprechende Anfragen können an diesen gerichtet werden.
- 2) Antwortet der Knoten jedoch innerhalb einer definierten Zeitspanne nicht oder ist vorab kein Verzeichnisdienst bekannt, muss der aktuelle Verzeichnisdienst gefunden werden. Zu diesem Zweck wird eine Broadcastanfrage nach einem Verzeichnisdienst gesendet, die alle im selben Netz erreichbaren Knoten an den Absender direkt beantworten (1->N Anfrage, N->1 Antwort). Die Antwort enthält die Adresse des Verzeichnisdienstes, sowie einen Zeitstempel zu dem die Information aktuell war.
 - a) Meldet sich ein Verzeichnisdienst auf diese Anfrage (erkennbar daran, dass Absender und Adresse des Verzeichnisdienstes übereinstimmen) ist ein Verzeichnisdienst gefunden.
 - b) Die übrigen Antworten der Knoten werden nach dem Zeitstempel absteigend sortiert und dann, bei der aktuellsten Information beginnend, die genannten Verzeichnisdienste kontaktiert bis ein aktiver Verzeichnisdienst gefunden wird.
 - c) Erhält der suchende Knoten keine Antwort oder sind alle ermittelten Verzeichnisdienste nicht erreichbar, wird der Knoten selbst zum Verzeichnisdienst und teilt dies per Broadcast allen Knoten mit, die daraufhin ihren Verzeichnisdiensteintrag aktualisieren und sich durch Antwort beim Verzeichnisdienst anmelden, so dass der Verzeichnisdienst seine Liste aktiver Knoten erhält. Erreicht diese Mitteilung einen anderen Verzeichnisdienst, wird dieser den Vorgang „Redundanz bei Verzeichnisdienst auflösen“ starten. Nun kann

der suchende Knoten die Anfrage an sich selbst stellen.

5.3.1.3 AUFBAU DER LISTE AKTIVER KNOTEN

siehe oben (c)

5.3.1.4 AKTUALISIERUNG DER LISTE AKTIVER KNOTEN

Um die Liste aller aktiven Knoten aktuell zu halten, werden alle bereits als aktiv bekannten Knoten in regelmäßigen Abständen per Unicast kontaktiert und anhand der Antwort geprüft, ob der Knoten noch aktiv ist. Neu hinzugekommene Knoten oder wieder erreichbare Knoten werden anschließend wie beim Aufbau der Liste kontaktiert. Auf diese Anfrage antworten jedoch nur diejenigen Knoten, die beim ersten Schritt noch nicht geantwortet haben. Alle aktualisieren aber den gültigen Verzeichnisdienst. Erstere Knoten haben bereits die aktuelle Version der Stammdaten erhalten, für letztere Knoten muss der Verzeichnisdienst jeweils separat noch „Stammdaten synchronisieren“ ausführen.

5.3.1.5 STAMMDATEN SYNCHRONISIEREN

Zur Prüfung der Aktualität und eindeutiger Identifizierung von Datensätzen in den Stammdaten, ist jeder Datensatz mit einer ID und einem Zeitstempel der letzten Änderung ausgestattet. Zusätzlich wird der Zeitstempel der letzten Änderung an den Gesamtdaten gespeichert. Die Synchronisation der Stammdaten wird in zwei Schritten ausgeführt. Zunächst überprüfen die beiden Knoten anhand des Zeitstempels zum Gesamtdatensatz, ob der gesamte Datensatz übereinstimmt. Nur wenn dies nicht der Fall ist, wird für jeden Datensatz, dessen Zeitstempel neuer ist als der ältere Zeitstempel der beiden ganzen Datensätze, die ID und der zum Datensatz gehörige Zeitstempel übertragen – dies sind alle potentiell hinzugekommenen bzw. geänderten Datensätze. Existiert die ID auf einem Knoten nicht, wird der Datensatz des anderen Knoten übernommen, ansonsten verwenden beide Knoten den Datensatz mit dem neueren Zeitstempel und sind nach dieser Prozedur auf dem gleichen Stand.

5.3.1.6 STAMMDATEN ÄNDERN

Stammdaten können von I/O Knoten geändert werden. Dazu ändert der Knoten seinen lokalen Datensatz und verwendet dann die Funktion „Stammdaten synchronisieren“, um diese Änderung dem Verzeichnisdienst mitzuteilen.

5.3.1.7 REDUNDANZ BEI VERZEICHNISDIENST AUFLÖSEN

Diese Funktion wird durch einen Verzeichnisdienst immer dann ausgeführt, wenn er erfährt, dass

noch ein anderer Verzeichnisdienst existiert. Sofern er diesen Verzeichnisdienst kontaktieren kann, ziehen beide Verzeichnisdienste eine Zufallszahl, der Dienst mit der höheren Zahl bleibt erhalten, der andere wird beendet (bei Gleichstand gewinnt der Dienst, der den Wettbewerb initiiert hat). Zusätzlich synchronisieren beide Verzeichnisdienste noch ihre Stammdaten. Bevor der unterlegene Dienst sich beendet übermittelt er jedoch noch die Liste der ihm bekannten Knoten, der überlegene Dienst aktualisiert daraufhin seine Liste aktiver Knoten und synchronisiert die Stammdaten mit allen Knoten.

5.3.1.8 VERWALTUNG DER AUFTRAGSDATEN

Die Auftragsdaten werden beim Verzeichnisdienst abgelegt und zusätzlich zwischen allen I/O Knoten synchronisiert, damit bei Ausfall des Verzeichnisdienstes stets eine Liste aller Auftragsdaten rekonstruierbar ist. Aufgrund der geringeren Anzahl an I/O Knoten ist diese Synchronisierung mit verhältnismäßig geringem Aufwand verbunden, da lediglich alle I/O Knoten die Informationen vom Verzeichnisdienst mitgeteilt bekommen müssen. Der Verzeichnisdienst kann dann als Blackboard (Master) [Bus96] fungieren, Aufträge entgegennehmen und an die I/O Knoten (Slaves) verteilen. Die Synchronisierung der Auftragsdaten nach einem Ausfall des Verzeichnisdienstes kann analog zur Synchronisierung der Stammdaten erfolgen (zunächst synchronisiert sich der (neue) Verzeichnisdienst mit allen anderen I/O Knoten um alle Daten zu beziehen, führt diese dann zusammen und synchronisiert den zusammengefassten Datensatz erneut mit allen anderen I/O Knoten. Diese Synchronisation kann (vorübergehend) fehlschlagen, z. B. wenn der Verzeichnisdienst zu einem I/O Knoten keine Verbindung mehr hat. Es besteht dann die Gefahr, dass Aufträge fälschlicherweise doppelt ausgeführt werden, z. B. wenn der Verzeichnisdienst einen Auftrag vergibt und ein I/O Knoten die Verbindung zum Verzeichnisdienst kurz vorher verliert, dadurch selbst zum Verzeichnisdienst wird und den Auftrag anderweitig vergibt. Dieses Problem lässt sich auf zwei Arten lösen:

- 1) Der Verzeichnisdienst erlaubt keine Annahme, Rückgabe oder Löschung von Aufträgen, die er nicht vollständig synchronisieren kann, dann werden diese (möglicherweise zeitkritischen) Aufträge nicht ausgeführt, bis die Synchronisierung vollständig abgeschlossen ist.
- 2) Der vom Verzeichnisdienst getrennte I/O Knoten bemerkt diesen Umstand (z. B. durch eine Prüfung der Verbindung zum Verzeichnisdienst vor jeder Auftragsänderung) und erlaubt keine Annahme, Rückgabe oder Löschung von Aufträgen, die er zum Zeitpunkt des Verbindungsverlustes bereits kannte. In diesem Fall können die Aufträge durch

mit dem Verzeichnisdienst verbundene I/O Knoten normal verarbeitet werden, eine spätere Synchronisierung entfernt die anderweitig bereits erledigten Aufträge aus der Liste des betroffenen I/O Knotens.

Neu hinzugekommene Aufträge können sowohl durch den Verzeichnisdienst als auch den getrennten I/O Knoten ganz normal verwaltet und synchronisiert werden.

5.3.1.9 PRÜFUNG VOR JEDER OPERATIVEN FUNKTION

Vor jedem operativen Prozessablauf muss der Knoten zur Aktualisierung seiner Netzdaten den Verzeichnisdienst kontaktieren und die Liste der aktiven Knoten sowie die Stammdaten auf Aktualität prüfen und ggf. synchronisieren.

5.3.1.10 KONTAKT ZU MEHREREN KNOTEN HERSTELLEN

Grundsätzlich ist es nicht empfehlenswert an beliebige viele Lagerbereichsknoten gleichzeitig eine Anfrage zu senden, da dies auf von Knoten gemeinsam genutzten Verbindungsstrecken zu Kollisionen von Nachrichten und damit Informationsverlusten führen kann. Ein geeignetes Kommunikationsprotokoll kann diese Verluste zwar erkennen und durch erneute Übertragung beheben, dies reduziert aber die Übertragungsgeschwindigkeit und führt zu einer zeitlichen Verzögerung. Daher wird eine konstante Zahl definiert, welche die Zahl der durch einen beliebigen Knoten maximal gleichzeitig kontaktierten Knoten repräsentiert. Erst nach Abschluss der Kommunikation mit der vorhergehenden Knotenmenge wird die nächste Knotenmenge kontaktiert bis alle gewünschten Knoten erreicht wurden.

5.3.2 OPERATIVE PROZESSABLÄUFE IM DEZENTRALEN LVS

Die operativen Funktionen sind bereits aus zentralen LVS bekannt und müssen in einem dezentralen LVS unter Nutzung der in 5.3.1 beschriebenen, systeminternen Prozessabläufe umgesetzt werden. Das Verhalten gegenüber dem Benutzer soll sich dabei nicht von einem zentralen LVS unterscheiden (vgl. Abschnitt 5.3), daher wird folgend nur das interne Verhalten der jeweiligen Prozessabläufe beschrieben.

5.3.2.1 SUCHE (WARE ODER LAGERPLATZ)

Eine Suche nach Waren oder Lagerplätzen richten I/O Knoten anhand von Suchkriterien an alle Lagerbereichsknoten. Da vorab nicht bekannt ist, welche Ware bzw. welcher Lagerplatz von welchem Lagerbereichsknoten verwaltet wird, wird die Anfrage

zunächst an alle erreichbaren Lagerbereichsknoten gesendet. Die Lagerbereichsknoten gleichen die Suchanfrage mit ihrem Datenbestand ab und bestimmen so jeweils für sich ihre Antwortmenge. Diese wird dann, auch wenn sie leer ist, an den anfragenden I/O Knoten übertragen, damit dieser in jedem Fall eine Antwort erhält und einen Abriss der Kommunikation zu dem angefragten Knoten erkennen kann. Der I/O Knoten sammelt alle Antworten und führt diese zu einem Suchergebnis zusammen.

5.3.2.2 RESERVIERUNG

Eine Reservierung sichert den exklusiven Zugang zu einer Ressource bei deren gemeinsamer Nutzung. Ein aktuell freier Lagerplatz steht nur für eine einzige Einlagerung zur Verfügung. Möchten mehrere FFZ diesen Platz zur gleichen Zeit für eine Einlagerung verwenden, so würde ohne Reservierung der erste Bewerber eine Einlagerung durchführen und alle folgenden Bewerber den Lagerplatz belegt vorfinden, obwohl dieser zum Zeitpunkt der Anfrage nach einem Lagerplatz noch frei war. Analog gilt dies für die Entnahme einer Ware. Der einfachen Formulierung halber wird zur Beschreibung der Reservierung sowohl für Waren als auch Lagerplätze der Begriff Ressource verwendet.

Ein I/O Knoten, der eine Ressource gesucht und gefunden hat und verwenden möchte, muss diesen Wunsch zunächst anmelden, also eine Reservierung vornehmen, die der Lagerplatz mit der ID des I/O Knoten verknüpft. Konkurrieren mehrere Knoten um die gleiche Ressource wird nach First-Come-First-Served [Bla14] entschieden, wer die Ressource erhält. Die Reservierung erfolgt nach dem Drei-Wege-Handshake [Ste11], damit sich Lagerbereichsknoten und I/O Knoten die Reservierung gegenseitig sicher bestätigen und hat eine zeitlich begrenzte Gültigkeit, damit eine einmal erstellte, aber (z. B. durch Ausfall des reservierenden Knotens) nicht genutzte Reservierung, die Ressource nicht permanent sperrt. Der konkrete Ablauf des Reservierungsvorgangs wird in Abbildung 1 dargestellt.

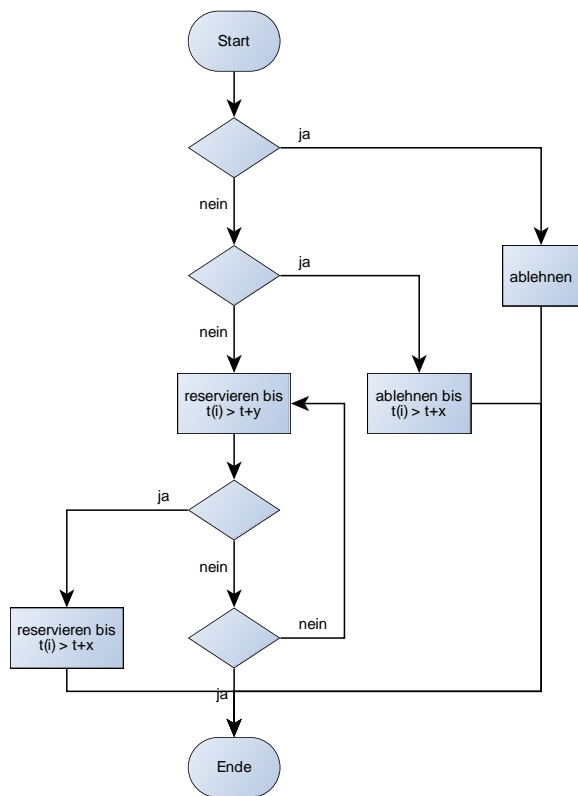


Abbildung 1 Ablaufdiagramm zur Reservierung (t =Eingang der Reservierungsanfrage, x =Reservierungsdauer, y =vorläufige Reservierungsdauer bis zur Bestätigung)

Prinzipiell empfiehlt es sich eine Möglichkeit vorzusehen eine Reservierung zu verlängern. Jeder I/O Knoten könnte seine benötigte Reservierung dann in regelmäßigen Abständen durch Anfrage am Lagerbereichsknoten auffrischen. Damit lässt sich verhindern, dass eine Reservierung aufgrund einer längeren Fahrzeit zur Ressource unnötig verfällt. Gleichzeitig kann der Timeout für einen Verfall der Reservierung so möglichst kurz gehalten werden.

5.3.2.3 WARENEINLAGERUNG

Bevor eine Einlagerung stattfinden kann wird eine Suche nach einem geeigneten Lagerplatz durchgeführt und dieser reserviert. Erst wenn die Reservierung bestätigt und damit gültig ist, kann mit dem Einlagerungsprozess begonnen werden.

Bei Ankunft am Lagerbereich kommunizieren der I/O Knoten A auf dem Transportfahrzeug, das die Einlagerung durchführen möchte, und der für den Lagerplatz zuständige Lagerbereichsknoten B wie folgt (vgl. Abbildung 2):

- 1) A kündigt B die Einlagerung an.
 - a) B prüft, ob der Lagerplatz (noch) für A reserviert ist und erteilt A die Erlaubnis zur Einlagerung, die B bestätigt und darauf die

Einlagerung durchführt. B aktualisiert seine Inventardaten, in dem er die von A übermittelten Informationen zur eingelagerten Ware auf dem verwendeten Lagerplatz verbucht.

- b) Wenn A die Einlagerung hingegen abbricht, sendet er B eine Absage und dieser löscht die Reservierung und markiert den Lagerplatz wieder als frei.
- c) Liegt keine Reservierung für A (mehr) vor, so verweigert B die Einlagerung und A muss einen (anderen) Lagerplatz für die Einlagerung suchen und reservieren.

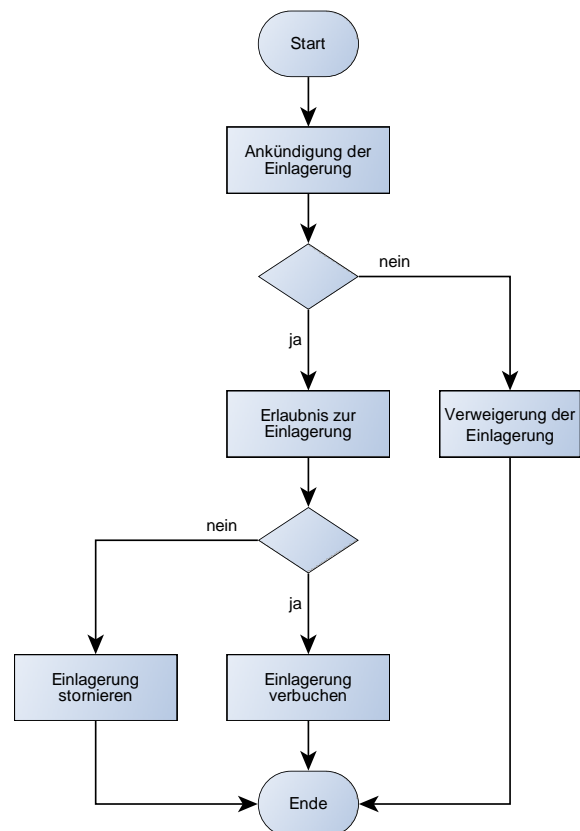


Abbildung 2 Ablaufdiagramm zur Einlagerung

5.3.2.4 WARENAUSLAGERUNG

Eine Warenauslagerung unterscheidet sich von einer Wareneinlagerung in der Art der reservierten Ressource. Das grundsätzliche Vorgehen ist aber analog (siehe Abbildung 2). Es wird jedoch physisch eine Auslagerung durchgeführt und die Inventardaten des Lagerbereichsknotens um die ausgelagerte Ware reduziert sowie der entsprechende Lagerplatz als frei verbucht. Kommt eine Ident-Technologie (wie z. B. RFID) zum Einsatz, so prüft der I/O Knoten zusätzlich, ob die ausgelagerte Ware dem Soll entspricht, bevor er die Auslagerung bestätigt.

5.3.2.5 INVENTUR

Ein I/O Knoten, der eine Inventur durchführt, startet eine Suche, nach allen Waren, die von allen Lagerbereichsknoten verwaltet werden. Das grundsätzliche Vorgehen ist wie bei einer Suche (vgl. 5.3.2.1), allerdings kann eine Inventur erst dann erfolgreich abgeschlossen werden, wenn alle Lagerbereichsknoten, die gegenwärtig verwendete Lagerbereiche verwalten, auf die Suchanfrage geantwortet haben. Eine Liste dieser Lagerbereichsknoten muss dann entweder durch einen Mitarbeiter oder durch eine Anfrage an den Verzeichnisdienst auf Korrektheit überprüft werden. Eine automatische Prüfung durch den Verzeichnisdienst setzt voraus, dass dieser über eine korrekte, statische Liste aller genutzten Knoten im Lager verfügt – die ohnehin vorhandene Liste aktiver Knoten genügt nicht, da ein Knoten ausfallen kann, dessen Lagerbereich bei einer Inventur zwingend berücksichtigt werden muss. Umgekehrt wäre auch eine automatisch erstellte Liste aller jemals aktiven gewesen Knoten nicht zielführend, da Knoten auch außer Betrieb genommen oder getauscht werden können. Für den Fall, dass es nur aktive Lagerbereichsknoten gibt, die aktuell auch wirklich einen Lagerbereich verwalten, kann statt der Liste der Knoten auch die Zahl der Knoten, die auf eine Suchanfrage geantwortet haben, geprüft werden.

5.3.2.6 AUFTRÄGE ERSTELLEN UND STORNIEREN

Die Auftragsdaten werden primär vom Verzeichnisdienst verwaltet, welcher als Blackboard für die Auftragsverwaltung fungiert (vgl. 5.3.1.8). Auf diesem Blackboard können Aufträge als „ausstehend“ erstellt und storniert werden, indem sie auf dem Blackboard durch einen I/O Knoten angelegt bzw. gelöscht werden. Eine Stornierung ist der Einfachheit halber nur für „ausstehende“ Aufträge vorgesehen. Für die Stornierung bereits angenommener Aufträge müssten die bisher durchgeführten Arbeitsschritte rückgängig gemacht werden. Dazu kann z. B. der zu stornierende Auftrag zu Ende ausgeführt und ein neuer Auftrag mit umgekehrtem Start und Ziel erzeugt und ausgeführt werden. Solche Storno-Aufträge dürfen selbst nicht storniert werden.

5.3.2.7 AUFTRAG ANNEHMEN UND DESSEN AUSFÜHRUNG BESTÄTIGEN

Ein Auftrag wird angenommen, indem ein I/O Knoten den Auftrag reserviert (analog zur Reservierung eines Lagerplatzes) und anschließend bearbeitet. Ist der Auftrag für den I/O Knoten reserviert, so kann er vom Verzeichnisdienst die Bearbeitung des Auftrages anfordern, bekommt dies gestattet und darf mit der Ausführung beginnen. Der Verzeichnisdienst setzt den Status des Auftrags in

diesem Fall auf „angenommen“. Nach Abschluss des Auftrags bestätigt der I/O Knoten die Ausführung dem Verzeichnisdienst, der den Auftrag daraufhin als erfolgreich abgeschlossen markiert und aus der Auftragsliste entfernt. In allen anderen Fällen wird die Ausführung des Auftrags verweigert. Nach Ablauf der Reservierung, steht ein vorläufig reservierter Auftrag wieder allen I/O Knoten zur Verfügung. Angenommene, aber aus gleich welchen Gründen nicht erfolgreich abgeschlossene Aufträge, benötigen eine Sonderbehandlung wie bei einem zentralen LVS.

5.4 MÖGLICHE SYSTEMSTÖRUNGEN

Störungen im System können sich auf die Einsatzfähigkeit und Verfügbarkeit des Systems auswirken. In Tabelle 8 wird aufgelistet, welche Störungen zu welchen Konsequenzen für das Gesamtsystem führen und welche Funktionalität in einem solchen Fall noch gewährleistet werden kann. Der Ausfall eines einzelnen Lagerbereichsknoten führt dazu, dass der Datenbestand des Knotens nicht mehr verfügbar ist. Alle Operationen, welche diesen Datenbestand nicht benötigen, funktionieren normal weiter. Fallen hingegen mehrere Lagerbereichsknoten aus, ist auch die Kommunikation noch betriebsbereiter Knoten betroffen, falls einer der ausgefallenen Knoten als einziger zwei separate Netzteile miteinander verbindet. Ist ein I/O Knoten nicht betriebsbereit, kann auch das zugehörige Transportmittel nicht mehr arbeiten. Ein Ausfall aller Knoten oder des gesamten Kommunikationsnetzes führt zu einer Störung des Gesamtsystems. In allen Fällen kann möglichen Einschränkungen durch zusätzliche Redundanz der Funktion begegnet werden. Dies ist aber u. U. sehr aufwändig zu realisieren.

5.5 SICHERHEIT

Ein Lagerverwaltungssystem verwaltet für Unternehmen sensible Daten, die Unbefugten Einblick in Prozesse und den Zustand des Unternehmens erlauben. Bei zentralen Systemen werden diese Daten üblicherweise vor unbefugtem Zugriff durch Unterbringung des IT Systems in einem abgeschlossenen Serverraum vor Ort oder im Rechenzentrum bei einem IT Dienstleister, zu dem nur ein stark eingeschränkter Personenkreis Zugang hat, geschützt.

Da sich die Daten laut Konzept auf den Sensorknoten im Lager befinden und dort für jede Person mit Zutritt zum Lager prinzipiell zugänglich sind, sollten weitere Maßnahmen zur Informationssicherheit ergriffen werden. In dem vorliegenden Konzept werden diese Aspekte nicht weiter betrachtet. Prinzipiell ist es aber möglich sowohl die Daten auf den Sensorknoten als auch die Kommunikation zwischen den Sensorknoten zu verschlüsseln. Ein

Tabelle 8 Systemstörungen beim dezentralen LVS und ihre Konsequenzen

Systemstörung	Konsequenz	Betroffene Operationen	Auswirkung auf Gesamtnetz	Erkennbar	Maßnahme
Ein Lagerbereichsknoten fällt aus	Knoten und zugehörige Daten nicht verfügbar	<ul style="list-style-type: none"> • Suche (unvollständig) • Inventur • Ein-/Auslagerung (lokal) • FIFO Prinzip kann nicht immer gewährleistet werden 	<ul style="list-style-type: none"> • Kommunikation ggfs. eingeschränkt, falls Netz in Teilnetze zerfällt. • ERP-System kann nicht jedes Teilsystem erreichen 	Ja, durch Verzeichnisdienst	Redundanz (Nachbarknoten übernehmen Aufgabe)
Mehrere Lagerbereichsknoten fallen aus	Analog zu Ausfall eines Knotens, Einschränkungen betreffen mehrere Lagerbereiche				Redundanz bei Kommunikationswegen durch zusätzliche Knoten, so dass Graph der Kommunikationsverbindungen auch bei Verlust von Knoten zusammenhängend bleibt
Ausfall Kommunikation	Betrieb gestört	Alle	Nichts geht mehr	Ja, Anbindung an ERP System nicht verfügbar	Redundantes Kommunikationsnetz (nur bedingt sinnvoll)
Ein I/O Knoten fällt aus	Betroffenes Transportfahrzeug nicht nutzbar	Ein-/Auslagerung des aktuellen Vorgangs	Reduzierte Transportkapazität	Ja, durch Verzeichnisdienst	Transportfahrzeug darf Bearbeitung eines Auftrags nur beginnen, wenn Verzeichnisdienst erreichbar
Alle I/O Knoten fallen aus	Kein Transportfahrzeug nutzbar, Liste der nicht ausgeführten Aufträge verloren	Ein-/Auslagerung	Keine	Ja, durch Verzeichnisdienst	Nicht sinnvoll, da Funktionalität bereits mehrfach redundant (in anderen I/O Knoten)
ERP-Anbindung fällt aus	Keine neuen Aufträge	Keine	Keine	Ja, durch ERP System	Redundante Anbindung an ERP System
Legende	Weiß: kein Unterschied zum zentralem System Orange: mittlere Einschränkungen Rot: Betrieb nur im dezentralen Fall gestört Grün: keine oder sehr geringe Einschränkungen Gelb: geringfügige Einschränkungen				

Schutz gegenüber manipulierten Knoten ist ebenfalls denkbar [TPM14]. Auch eine Anbringung an schwierig zugänglichen Orten kann die Hürde für einen Datendiebstahl in der Praxis erhöhen.

6 AUSWAHL DER EINGESETZTEN TECHNOLOGIEN

Für eine Umsetzung des im vorherigen Kapitel beschriebenen Konzeptes werden Hardwareeinheiten als Grundlage für die beschriebenen Knotentypen und eine Kommunikationstechnologie für die Verbindung der Knoten untereinander benötigt und beides in diesem Kapitel ausgewählt. Maßgeblich für die Anforderungen an die Technologien und damit deren

Auswahl sind die Einsatzbedingungen in einem dezentralen LVS. Für den Fall, dass keine geeignete Technologie existiert, die alle gestellten Anforderungen gleichzeitig erfüllen kann, erfolgt eine Priorisierung der Anforderungskriterien.

Als Hardwareeinheit wird für jede Knotenart aus den in [Mir14] beschriebenen, prinzipiell geeigneten Rechnerklassen (Server, Tablet-PC, SBC, Sensor-knoten), eine besonders geeignete Klasse ausgewählt. Ein Repräsentant dieser Klasse kann dann für die Umsetzung verwendet werden. Prinzipiell eignen sich auch viele andere Instanzen dieser Klasse dafür – der Grad der Eignung einer konkreten Hardwareplattform hängt allerdings von den eigenen Entwicklungszielen

und aktuell am Markt verfügbaren Komponenten ab. Eine detaillierte Bewertung ist daher erst bei der eigenen Implementierung sinnvoll.

Für eine aufwandsarme Umsetzung des Konzeptes ist die Verwendung standardisierter und etablierter Kommunikationsprotokolle notwendig, da nur dann auf umfangreiche Bibliotheken für die Nutzung dieser Protokolle zurückgegriffen werden kann. Es werden daher nur von gängigen Betriebssystemen unterstützte und häufig verwendete Protokolle in Betracht gezogen.

Zunächst wird eine geeignete Hardwarekomponente für jeden Knotentyp ausgewählt und im Anschluss kompatible Kommunikationsprotokolle bezüglich ihrer Eignung analysiert.

6.1 RAHMENBEDINGUNGEN

Die Rahmenbedingungen in Tabelle 9 leiten sich aus den Anforderungen (Abschnitt 3.2) sowie den Funktionen und Eigenschaften der Entitäten des Konzeptes (Kapitel 5) ab.

6.2 PRIORISIERUNG DER ANFORDERUNGSKRITERIEN

Zur Priorisierung der Anforderungskriterien eignet sich die zu diesem Zweck im Projektmanagement häufig eingesetzte MoSCoW-Methode besonders gut, da sie sowohl die technische Notwendigkeit als auch die zeitliche Dringlichkeit einer Anforderung berücksichtigt und klar nachvollziehbare Ergebnisse

liefert [IIB09]. Eine Aussage wie gut die jeweiligen Anforderungen erfüllt werden oder eine Rangfolge der Anforderungen innerhalb der selben Priorität kann durch Anwendung dieser Methode nicht bestimmt werden.

Die MoSCoW-Methode sieht dazu folgende Prioritäten vor:

- 1) (M)ust-Anforderungen haben die höchste Priorität und sind für das Projekt unabdingbar.
- 2) (S)hould-Anforderungen haben die zweithöchste Priorität und sollten erfüllt sein, wenn sie nicht mit den Must-Anforderungen im Widerspruch stehen. Hier werden die in Abschnitt 3.2 definierten Nebenbedingungen eingeordnet.
- 3) (C)ould-Anforderungen haben die nächst niedrigere Priorität und können erfüllt sein – müssen jedoch nicht. Ihre Nichterfüllung ist für den Grunderfolg des Projektes nicht entscheidend.
- 4) (W)on't-Anforderungen haben die geringste Priorität und sind technisch notwendig, aber zeitlich nicht kritisch.

6.2.1 RECHENEINHEITEN

Das Konzept sieht Lagerbereichsknoten, I/O Knoten und Datenerfassungsknoten mit jeweils unterschiedlichem Einsatzzweck vor. Für jede Knotenart gibt es unterschiedliche Anforderungen, daher sind die Kriterien je Knoten separat zu betrachten.

Tabelle 9 Klassifizierung und Begründung der Rahmenbedingungen für ein dezentrales LVS

Klassifizierung	Bedingung	Begründung
Umgebungsrelevant (unmittelbares Einsatzfeld)	Größe der Lagerfläche	Als Referenzszenario wird ein kleiner industrieller Betrieb angenommen. Die Lagerfläche wird abhängig von Branche und Produktpalette bestimmt. Da Branche und Produktpaletten mannigfaltig sind, wird eine obere Grenze für die Lagerfläche von 10.000m ² angenommen.
	Unterschiedliche Lagerarten	Auf dem selben Werksgelände müssen z. B. Regallager, Kleinteilelager, Blocklager, Tiefkühlager mit unterschiedlichen Eigenschaften verwaltet werden.
Finanziell	Geringe Kosten	Die Kosten sollten nicht über den üblichen Kosten eines (zentralen) LVS von 50.000€ bis 300.000€ liegen [Wol13]
Technisch	Flexibilität, Wandlungsfähigkeit	Siehe Kapitel 4
	Geringer Installations- und Wartungsaufwand, Schlanke Infrastruktur	Die durchschnittliche Einführungszeit von drei bis neun Monaten bei herkömmlichen LVS soll zur Vermeidung der Akzeptanz nicht überschritten werden. [Wol13]
	Einsatz etablierter Technologien	Einfache Implementierung und indirekt geringere Kosten
	Grafische Benutzeroberfläche	Mensch-Maschine-Interface zur Interaktion mit den Mitarbeitern
	Mobilität	Falls die Knotenart nicht stationär betrieben werden soll

6.2.1.1 LAGERBEREICHSKNOTEN

Die Anzahl der Lagerbereichsknoten wird im Wesentlichen durch die Zahl der Lagerbereiche bestimmt und ist im Vergleich zu den I/O Knoten und Datenerfassungsknoten groß. Daher müssen die Lagerbereichsknoten in erster Linie vergleichsweise kostengünstig sein, um den Kostenrahmen eines LVS nicht zu sprengen. Da die Datenhaltung und Informationsverarbeitung (auch) auf den Knoten selbst stattfindet, müssen die Rechenleistung und der Speicherplatz für derartige Funktionen ausreichend dimensioniert sein.

Die Recheneinheiten werden, anders als in einem zentralen LVS, in einem industriellen Umfeld eingesetzt und sollten daher robust sein und gegen Staub- und Flüssigkeitseinwirkung geschützt werden. Eine drahtlose Kommunikation ist ebenso sinnvoll, um durch Vermeidung von Datenkabeln die Infrastruktur schlank zu halten.

Zusätzlich können die Stromkabel zu den Lagerbereichsknoten eingespart werden, wenn diese durch einen Akku mit Strom versorgt werden. Eine flexible Anbringung der Knoten kann durch ein kompaktes Gehäuse mit geringen Abmessungen erreicht werden. Sinnvoll aber aufwändig zu realisieren wäre eine vollständig autarke Stromversorgung, z. B. durch Solarzellen an den Knoten.

Tabelle 10 MoSCoW Kriterien für die Auswahl der Hardware (Lagerbereichsknoten)

M	S	C	W
Ausreichend Speicherplatz	Etablierte und ausgereifte Technologie	Batteriebetrieb und lange Akkulaufzeit	Energieversorgung autark durch Energy-Harvesting
Genügend Rechenleistung für Informationsverarbeitung	Robust und für industrielle Umgebung ausgelegt	Geringe Abmessungen	
Kostengünstig	Kommunikation über Funktechnologie		

6.2.1.2 I/O KNOTEN

Im Gegensatz zu den Lagerbereichsknoten sind die (meisten) I/O Knoten mobil und müssen von Mitarbeitern oder auf Flurförderzeugen während des Gebrauchs transportiert werden. Eingabe- und Bedienmöglichkeiten in Form einer grafischen Benutzeroberfläche (GUI), sowie eine drahtlose

Kommunikation und eine eigene, für die gesamte Arbeitszeit (z. B. eine Schicht) ausreichende Stromversorgung sind daher für den Praxiseinsatz zwingend notwendig. Aufgrund der geringeren Anzahl benötigter Einheiten dieser Klasse, sind die Kosten für eine solche Einheit nicht ganz so wichtig wie bei den Lagerbereichsknoten. Zusätzlich sollten die von Mitarbeitern mitgeführten I/O Knoten aus ergonomischen Gründen ein geringes Gewicht aufweisen und einfach zu bedienen sein. Für die Integration von Ident-Technologien sollten externe Anschlüsse für Lesegeräte einer Ident-Technologie vorhanden oder, im Idealfall, die entsprechende Technologie bereits integriert sein. Bei Einsatz in rauer Umgebung kann die Lebensdauer eines I/O Knotens analog zu den Lagerbereichsknoten durch den Einsatz robuster Hardware gesteigert werden.

Tabelle 11 MoSCoW Kriterien für die Auswahl der Hardware (I/O Knoten)

M	S	C	W
Grafische Benutzeroberfläche	Etablierte und ausgereifte Technologie	Robust und für industrielle Umgebung ausgelegt	Energieversorgung autark durch Energy-Harvesting
Dateneingabemöglichkeit	Kostengünstig		Integrierte ID Technologie (Kamera für Barcode)
Batteriebetrieb und lange Akkulaufzeit	Anbindung für Lesegerät /Schreibgerät oder integrierte ID Technologie		
Kommunikation über Funktechnologie	Leicht, ergonomische Bauweise		

6.2.1.3 DATENERFASSUNGSKNOTEN

Bei stationären Datenerfassungsknoten muss für die Erfassung von Daten ein Lese-/Schreibgerät der gewünschten Ident-Technologie entweder angebunden werden oder bereits integriert sein. Das Gerät sollte aufgrund seines permanenten Aufenthaltes in industrieller Einsatzumgebung entsprechend robust sein und gegen Staub- und Flüssigkeitseinwirkung geschützt werden.

Wird die Funktionalität zur Datenerfassung in einen I/O Knoten integriert, sind diese Anforderungen zusätzlich zu denen des I/O Knotens zu berücksichtigen. Mobile Datenerfassungsknoten sollten möglichst in einen I/O Knoten integriert werden, um

ausreichenden Benutzerkomfort zu gewährleisten. Bezüglich Aufbau und Handhabung werden in diesem Fall ähnliche Anforderungen gestellt, wie bei I/O Knoten.

Tabelle 12 MoSCoW Kriterien für die Auswahl der Hardware (Datenerfassungsknoten)

M	S	C	W
Anbindung für Lese-/Schreibgerät oder integrierte ID Technologie (z. B. Kamera für Barcode)	Robust und für industrielle Umgebung ausgelegt	Batteriebetrieb und lange Akkulaufzeit (mobil)	Energieversorgung autark durch Energy-Harvesting (mobil)
		Leicht, ergonomische Bauweise (mobil)	
		Kommunikation über Funktechnologie (mobil)	

6.2.2 KOMMUNIKATIONSPROTOKOLLE

Für die Übertragung von Informationen über Kommunikationsprotokolle wurde das ISO/OSI-Schichtenmodell entwickelt [Zim80], das die Kommunikation in 7 Schichten unterteilt. Jede dieser Schichten muss durch ein Übertragungsprotokoll abgedeckt sein. Diese Auswahl beschränkt sich nur auf die Protokolle der ISO/OSI-Schichten 1-6. Das Protokoll für die Anwendungsschicht (Schicht 7) wird durch das Konzept (in Kapitel 5) definiert.

Laut Rahmenbedingungen (vgl. Tabelle 9) muss das System flexibel und wandlungsfähig sein. Das Kommunikationsprotokoll muss also fähig sein sich neuen Gegebenheiten anzupassen. Ausgefallene Knoten sollen automatisch aus dem Netz entfernt bzw. neu hinzugekommene Knoten in das dezentrale Netz integriert werden.

Die Informationsübertragung zwischen zwei Knoten muss fehlerfrei und zuverlässig sein. Dazu eignen sich verbindungsorientierte Protokolle, die über eine ausreichend Fehlererkennung/-korrektur verfügen. Zur Kontaktierung aller Knoten (z. B. beim Aufbau der Liste aktiver Knoten, 5.3.1.3) werden zusätzlich Broadcast-Nachrichten benötigt, die sich mit verbindungslosen Protokollen erzeugen lassen (vgl. z. B. RFC 922).

Die zugrundeliegende, physikalische Übertragungstechnik (ISO/OSI Schicht 1) muss drahtlos

sein, da mobile I/O Knoten entsprechend angebunden werden müssen (siehe 6.2.1.2). Wird aus diesem Grund eine drahtlose Infrastruktur aufgebaut, sollte diese auch von Knoten verwendet werden, für die eine drahtlose Kommunikation nicht zwingend Voraussetzung ist. Bei Verwendung einer drahtlosen Übertragungstechnologie ergibt sich potentiell eine schlanke Infrastruktur, da das Verlegen von Datenkabeln (teilweise) entfallen kann. Für eine Praxisanwendung des Konzeptes wären zusätzlich die in Abschnitt 5.5 definierten Sicherheitsziele relevant.

Tabelle 13 MoSCoW Kriterien für die Auswahl der Kommunikationsprotokolle

M	S	C	W
Etablierte und ausgereifte Technologie	Geringer Installations- und Wartungsaufwand, Schlanke Infrastruktur		Sicherheit (Vertraulichkeit, Integrität, Verfügbarkeit) [BSI08]
Flexibilität, Wandlungsfähigkeit	Robust gegen Störungen der Kommunikation		
Verbindungsorientiert für Verbindungsaufbau und Kommunikation	Kostengünstig		
Verbindungslos für Broadcast			
Erkennung / Korrektur von Übertragungsfehlern			
Funk/kabellos			

6.3 AUSWAHL

Anhand der oben definierten Prioritätenliste (vgl. Abschnitt 6.2) für Hardware und Kommunikation kann eine Auswahl geeigneter Recheneinheiten aus der Kategorie Server, Tablet-PC, SBC und Sensorknoten vorgenommen werden. Die Vorgehensweise ist dabei immer gleich:

- 1) Beginne bei der höchsten Priorität.
- 2) Wähle denjenigen Kandidaten, welcher die meisten Anforderungen der aktuellen Priorität erfüllt, dieser ist das Ergebnis. Erfüllen mehrere verbliebene Kandidaten die gleiche Anzahl an Kriterien, behalte diese und entferne die übrigen Kandidaten.
- 3) Wechsle zur nächst niedrigeren Priorität und wiederhole 2.-3. bis keine niedrigere Priorität mehr existiert.
- 4) Wähle aus den verbliebenen Lösungen den (notfalls subjektiv) geeignetsten Kandidaten. Der

für die endgültige Auswahl notwendige Vergleich wird in diesem Fall anhand einer Bewertung der Eignung bezüglich der jeweiligen Anforderungen geführt. Tabelle 14 liefert hierzu eine Einschätzung auf einer 3-stufigen Skala:

- Kein Eintrag bedeutet, durch Einsatz der genannten Gerätekategorie ist die Anforderung nicht sinnvoll erfüllbar (diese Information verwendet indirekt bereits die MoSCoW-Methode).
- Eine Lösung, deren Eintrag mit einem „++“ gekennzeichnet ist, eignet sich besonders gut oder die Umsetzung ist besonders einfach möglich.
- Alle übrigen Einträge werden mit einem „+“ markiert (prinzipiell geeignet).

Folgend wird nach der MoSCoW-Methode ein geeigneter Kandidat für jeden Knotentyp ausgewählt.

6.3.1 LAGERBEREICHSKNOTEN

- 1) Must: Tablet-PC und SBC erfüllen alle Kriterien während Sensorknoten nicht genug Speicher bereitstellen, um die Datenhaltung eines Lagerbereichs übernehmen zu können und Server zu teuer sind.
- 2) Should/Could: Sowohl Tablet-PC als auch SBC können alle Kriterien erfüllen.
- 3) Won't: Der Aufwand für eine Energieversorgung mittels Energy-Harvesting ist in beiden Fällen gleich.

Somit sind Tablet-PC und SBC für den Einsatz als Lagerbereichsknoten prinzipiell geeignet. Wesentliche Vorteile eines SBC gegenüber einem Tablet-PC sind der geringere Preis und die Erweiterbarkeit um Funktionsmodule. Daher stellt ein SBC aus Sicht der Autoren die bessere Alternative dar.

Es existiert mittlerweile eine große Auswahl unterschiedlicher SBC, die sich u. a. in Abmessungen, Speicherplatz und Prozessorleistung unterscheiden. Für die Lagerbereichsknoten wird der Raspberry Pi (RPi) als Hardwareplattform vorgeschlagen, da es sich um einen sehr preisgünstigen SBC handelt, auf dem ein vollwertiges Betriebssystem und dafür übliche Programme gestartet werden können. Dies vereinfacht die Programmierung gegenüber Mikrokontroller-basierten SBC deutlich.

6.3.2 I/O KNOTEN

- 1) Must: Nur ein Tablet-PC erfüllt direkt alle Kriterien. Ein SBC könnte prinzipiell mit entsprechenden Modulen erweitert werden, damit eine Dateneingabemöglichkeit und Batteriebetrieb

möglich werden. Das Ergebnis wäre dann aber lediglich ein mit mehr Aufwand erreichter, ähnlicher Funktionsumfang und wird daher nicht berücksichtigt. Ein Server ist für den Kommissionierer zu schwer und kann aufgrund des hohen Stromverbrauchs nicht über einen längeren Zeitraum durch eine Batterie mit Strom versorgt werden. Ebenso wie ein Sensorknoten verfügt der Server zunächst über kein geeignetes GUI.

Ein handelsüblicher Tablet-PC wird daher als I/O Knoten genutzt. Der für die Anwendung relevante Funktionsumfang ist bei diesen Geräten sehr ähnlich, so dass Hersteller und Modell nur eine untergeordnete Rolle spielen.

6.3.3 DATENERFASSUNGSKNOTEN

- 1) Must: Alle Kandidaten außer der Sensorknoten können an ID-Technologien angebunden werden.
- 2) Should: Alle verbliebenen Kandidaten sind in einer für den industriellen Einsatz geeigneten Form verfügbar.
- 3) Could/Won't: Diese Anforderungen sind nur für die mobile Version definiert.

Für die ortsgebundenen Datenerfassungsknoten eignen sich daher prinzipiell Server, Tablet-PC und SBC. Verwendet werden zu diesem Zweck ebenfalls SBC, da damit die Funktion der Lagerbereichsknoten und ortsgebundener Datenerfassungsknoten von der selben Recheneinheit ausgeführt werden können und somit eine kostensparende Funktionsintegration der beiden Knotentypen erfolgen kann. Für eine Funktionsintegration mit einem mobilen Datenverarbeitungsknoten sollte ein Tablet-PC mit Android als Betriebssystem verwendet werden, da Peripheriegeräte wie RFID-Reader dort einfach angebunden werden können [mti14]. Tablet-PC verfügen zudem üblicherweise über eine integrierte Kamera, welche zum Auslesen von Barcodes genutzt werden kann.

6.3.4 KOMMUNIKATIONSPROTOKOLLE

Entsprechend der in 6.2.2 aufgelisteten Kriterien werden Protokolle für die im ISO/OSI-Modell definierten Schichten ausgewählt. Anders als bei den Recheneinheiten wird keine vorab fest vorgegebene Klassenmenge definiert. Es existiert eine Vielzahl an Protokollen auf den jeweiligen Schichten des ISO/OSI-Modells, die prinzipiell nahezu beliebig miteinander kombinierbar sind. Für die Demonstration der Eignung eines dezentralen LVS spielt die optimale Wahl einer solchen Protokollkombination keine Rolle. Daher genügt es eine Kombination zu wählen, die alle dafür aufgestellten Must- und Should-Kriterien erfüllt und

Tabelle 14 Durch die Geräteklassen erfüllbare Kriterien

Kriterium	Server	Tablet-PC	SBC	Sensorknoten
Umgebungsrelevant				
Klein		+	+	++
Leicht, ergonomisch		++	+	++
Robust für industrielle Anwendung	++	+	+	++
Technisch				
Rechenleistung	++	+	+	
Speicherplatz	++	+	+	
Funkkommunikation	+	+	+	+
Batteriebetrieb		+	+	++
Energy-Harvesting		+	+	+
Ausgereifte Hardware	++	++	+	++
Anbindung ID Technologie	++	+	+	
Integrierte ID Technologie		+		
Grafische Benutzeroberfläche		++		
Dateneingabe	+	+		
Finanziell				
Kostengünstig		+	++	++
+ sinnvoll erfüllbar, ++ besonders gut erfüllbar/besonders einfach möglich				

theoretisch um Sicherheitsfunktionen erweitert werden kann, sofern diese gewünscht sind (vgl. Tabelle 13). Das Vorgehen ist analog zur (notfalls subjektiven) Wahl einer Klasse bei mehreren geeigneten Kandidaten.

Tabelle 15 Gewählte Protokolle in den ISO/OSI-Schichten

ISO/OSI Schicht	Protokoll	Begründung
1	IEEE802.11 [IEEE802.11]	Drahtlose Kommunikation, sehr weit verbreitet, kostengünstige Hardwaremodule für alle Plattformen verfügbar
2	IEEE802.11s [IEEE802.11s]	Flexibilität, Wandlungsfähigkeit
3	IPv4/IPv6 [RFC793]	Etablierte und ausgereifte Technologie, sehr umfangreicher Einsatz (Internet)
4	TCP [RFC793] + UDP [RFC768, RFC2460] (Broadcasts)	Siehe IP, Fehlererkennung, verbindungsorientiert (TCP), verbindungslos (UDP)
5	-	Nicht notwendig
6	-	Java-Implementierung. Zwischen Java Clients ist die Datenrepräsentation stets gleich, diese Schicht wird daher für die Demonstration nicht benötigt.
7	Eigenes Protokoll	Kein geeignetes Standardprotokoll verfügbar

Die Kombination IEEE802.11s und TCP/IP für verbindungsorientierte Kommunikation bzw. UDP/IP

für Broadcasts ist zur Erfüllung der Anforderungen geeignet (siehe Tabelle 15). IEEE 802.11s ist ein flexibles und robustes Kommunikationsprotokoll und deckt die ISO/OSI Schichten 1 und 2 ab. Die Kommunikationsteilnehmer (Knoten) bilden untereinander selbstständig ein Netzwerk, indem jeder Mesh-Knoten als Router zu allen nächstgelegenen Knoten dient. Knoten können dem Netzwerk automatisch beitreten und es auch wieder verlassen, so dass sich die Zahl der Teilnehmer und die Reichweite des Gesamtnetzes vergrößert bzw. verringert. [Fac06] Diese Kombination bietet eine Alternative zur normalerweise in Lagerhallen zusätzlich benötigten, drahtlosen Netzwerkversorgung. TCP/IP und UDP/IP haben sich durch den mittlerweile Jahrzehnte dauernden Einsatz als zentrale Protokolle der Internet Architektur bewährt und sind als Implementierung für jedes internetfähige Gerät verfügbar.

6.3.5 ZUSAMMENSPIEL VON HARDWARE UND KOMMUNIKATIONSProtokoll

Die gewählte Kombination von Hardware und Kommunikationsprotokollen ist umsetzbar und kann daher verwendet werden. Einzig IEEE802.11s (drahtloses Mesh-Netzwerk) ist ein recht neuer Standard und wird noch nicht von allen Betriebssystemen zueinander kompatibel unterstützt. Für die Integration von Knoten ohne Mesh-Netzwerk Unterstützung muss daher ein Vermittler existieren, der mit einem Netzwerkmodul einen herkömmlichen IEEE802.11 Zugangspunkt aufbaut, mit einem zweiten

Modul Teilnehmer eines IEEE802.11s basierten Mesh-Netzwerk ist und dadurch beide Netzwerke als Router miteinander verbinden kann. Ansonsten wird die dezentrale Struktur des Lagerverwaltungssystems durch die Protokolle geeignet abgebildet.

Die theoretisch maximale Funkreichweite eines WLAN-Moduls beträgt ca. 100 m – in der Praxis je nach Anbringungsort bei direkter Sichtverbindung etwa die Hälfte davon. Die Abstände zwischen den RaspberryPi sollten aber deutlich geringer als die maximale Funkreichweite sein, um eine gute Signalqualität und damit stabile Informationsübertragung zwischen zwei Knoten zu erreichen. Bei einem Mesh-Netzwerk sollten sich zudem die Sendebereiche der Knoten für eine gute Netzabdeckung im Lager und möglichst viele redundante Verbindungen zur Kompensation des Ausfalls einzelner Knoten deutlich überlappen. Auf einer Fläche von 100 m x 100 m sollten sich durchschnittlich neun RaspberryPi mit einem maximalen Abstand zu einem Nachbarknoten von ca. 25 m befinden (siehe Abbildung 3). Damit stets eine Sichtverbindung¹ zwischen den Knoten besteht, sollten diese möglichst hoch über dem Boden angebracht werden (z. B. in der obersten Ebene eines Hochregallagers). Es ist darauf zu achten, dass die Netzabdeckung auch am Boden ausreichend ist, damit sich I/O Knoten mit den Lagerbereichsknoten verbinden können. Grundsätzlich ist das Vorgehen ähnlich zur Planung eines drahtlosen Netzwerks für Lagerhallen [Eis07].

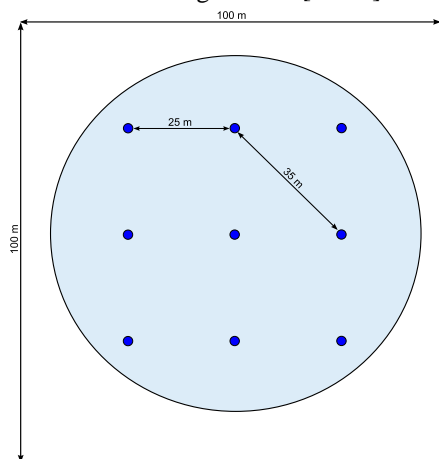


Abbildung 3 Vorschlag zur Anordnung der Knoten (blau) für eine geeignete Netzabdeckung (Kreis entspricht der Netzabdeckung allein durch den Knoten im Zentrum)

¹ Eine Sichtverbindung ist bei IEEE802.11 grundsätzlich nicht notwendig, die Reichweite ist bei Bestehen einer solchen Sichtverbindung jedoch in der Regel größer und leichter einzuschätzen.

7 EVALUIERUNG

7.1 EXEMPLARISCHER VERSUCHSAUFBAU

Anhand des in Kapitel 5 beschriebenen Konzeptes und unter Nutzung der in Kapitel 6 gewählten Hardware und Kommunikationsschnittstellen folgt ein exemplarischer Aufbau der verschiedenen Knoten für ein dezentrales LVS in einer Versuchshalle. Dieser Aufbau ist nicht repräsentativ für ein in der Praxis häufig vorkommendes Lagerlayout, sondern soll zeigen, dass das Konzept für viele verschiedene Lagertypen und übliche Transportmittel unter realistischen Umgebungsbedingungen eingesetzt werden kann.

Dazu werden verschiedene Lagerarten mit jeweils einem Raspberry Pi als Lagerbereichsknoten ausgestattet:

- 1 Palettenregallager
- 1 Bodenblocklager
- 1 Manuelles Kleinteilelager
- 1 Manuell bedientes Paternosterregallager
- 1 Automatisches Kleinteilelager
- 1 Kühllager

Die Mitarbeiter nutzen unterschiedliche Transportmittel (Kommissionierwagen, Handgabelhubwagen, Gabelstapler) zur Warenbewegung und verwenden dazu jeweils einen Tablet-PC als mobilen I/O Knoten. Der kombinierte Warenein- und Warenausgang ist mit einem UHF-RFID Gate und einem Datenerfassungsknoten ausgestattet. Ein Verwaltungsarbeitsplatz, bestehend aus einem handelsüblichen Desktop-PC mit der Funktionalität eines I/O Knotens, sorgt zusätzlich für die ERP-Anbindung des dezentralen LVS. Ein weiterer Raspberry Pi dient als Access-Point zur Anbindung der Tablet-PCs, da diese Geräte aktuell ohne eine Modifikation der Firmware nicht direkt am Mesh-Netzwerk teilnehmen können. Die räumliche Anordnung der Lagerbereiche und der zugehörigen Knoten kann Abbildung 4 entnommen werden. Grundsätzlich ist zu beobachten, dass ein stabiles, vielfach redundantes Kommunikationsnetz zwischen den Lagerbereichsknoten aufgebaut wird. Die Netzabdeckung umfasst den gesamten Versuchsbereich. Ein simulierter Ausfall eines beliebigen Lagerbereichsknotens ändert daran nichts.

Ein praxisnaher Test des oben beschriebenen Aufbaus bzw. einer beliebigen Instanz ist nur begrenzt repräsentativ, da sich daraus nicht ohne Weiteres allgemeine Aussagen für beliebige andere Szenarien ableiten lassen. Ein dezentrales LVS soll jedoch ein hohes Maß an Flexibilität mitbringen und sich auf nahezu beliebige Szenarien anwenden lassen. Daher

wird für die Evaluierung ein theoretischer Ansatz gewählt, der von der Anzahl der Knoten weitgehend unabhängig ist. Abseits der Funktionalität ist die Latenz der Ausführung eines interaktiven Programms bzw. die sich für Prozess oder Nutzer daraus ergebenden Verzögerungen ein wichtiges Bewertungskriterium (vgl. 3.2.1). Als Maß für die Latenz einer Operation kann die Antwortzeit des Systems beginnend ab einer entsprechende Anfrage dienen. Der jeweilige Lagerprozess darf für einen reibungslosen Ablauf durch diese Antwortzeit nicht signifikant verzögert werden.

7.2 THEORETISCHE BERECHNUNG

Zunächst wird die Antwortzeit im günstigsten Fall (best case scenario) betrachtet, also der Fall, in dem

- keine Fehler auftreten (Paketverluste, defekte Knoten usw.),
- alle angefragten Knoten betriebsbereit sind, auf Anfragen antworten,
- schon eine Kommunikationsverbindung mit den zu kontaktierenden Knoten besteht und
- bereits ein Verzeichnisdienst gefunden wurde.

Die Wahrscheinlichkeit für den Ausfall eines der Knoten im dezentralen LVS steigt zwar mit der Anzahl der Knoten an, ist aber generell als gering zu bewerten. Auch das Funknetz ist die meiste Zeit über betriebsbereit, ansonsten ist eine vernünftige Lagerhaltung nicht möglich. Eine Kommunikationsverbindung wird beim ersten Kontaktversuch zwischen zwei Knoten aufgebaut und danach dauerhaft verwendet, solange der Knoten erreichbar ist. Overhead durch den Verbindungsaufbau fällt also nur sehr selten an. Grundlegende Einflussgrößen für die Betrachtung des günstigsten Falles sind:

- der Mittelwert der durchschnittlichen Signallaufzeit zwischen allen Knotenpaaren t_{Signal} (Signallaufzeit vom ersten Knoten zum Zweiten und zurück),
- die mittlere Prozesszeit zur Bestimmung der für die Antwort relevanten Daten $t_{\text{Verarbeitung}}$ und
- die mittlere Übertragungsdauer der Antwortdaten $t_{\text{Transport}}$.

Folgende Annahmen werden zur Vereinfachung der Berechnung getroffen:

- Jeder Knoten verwaltet nur einen relativ kleinen Datenbestand, $t_{\text{Verarbeitung}}$ ist daher im Vergleich zu t_{Signal} sehr gering ($t_{\text{Verarbeitung}} \ll t_{\text{Signal}}$) und kann vernachlässigt werden.

- $t_{\text{Transport}}$ hängt hauptsächlich von t_{Signal} ab, wenn wenig Daten übertragen werden müssen. Bei vielen Daten hingegen überwiegen die Bandbreite der Verbindung und die zu übertragenden Datenmenge ($t_{\text{Signal}} \approx \text{Datenmenge/Bandbreite}$). Da von jedem Knoten nur ein relativ kleiner Datenbestand verwaltet wird zu dem Antworten geliefert werden können, ist auch die zu übertragende Datenmenge je Knoten gering und wird daher durch eine zusätzliche Signallaufzeit approximiert ($t_{\text{Transport}} \approx t_{\text{Signal}}$). Ist der Umfang der Daten bei einer Operation stets so gering, dass er in der unmittelbaren Reaktion auf die Anfrage (also in einem gemeinsamen Paket) kodiert werden kann, ist $t_{\text{Transport}} = 0$.
- Für den Nutzer ist lediglich die Antwortzeit der operativen Prozessabläufe und die dazu unmittelbar benötigten systeminternen Funktionen relevant. Sehr selten notwendige oder regelmäßig im Hintergrund durchgeführte Operationen ohne direkten Bezug zu operativen Prozessabläufen haben keinen nennenswerten Einfluss auf die für den Nutzer sichtbare Antwortzeit des Systems und können daher vernachlässigt werden.
- Die Inventur ist normalerweise zeitlich unkritisch, da sie selten und dann in einem großen Zeitfenster durchgeführt werden kann.

Der mittlere Wert von t_{Signal} kann über die sogenannte Round-Trip-Time (RTT) der Kommunikation zwischen je zwei Knoten bestimmt werden (vgl. [RFC792, Ste11]): t_{Signal} entspricht dem Mittelwert aller zwischen je zwei Knoten im Netz bestimmten RTT-Werte.

Zusätzlich ist eine Betrachtung der Kosten bezüglich der Wartezeit für gängige Fehler aufschlussreich, um deren Einfluss auf die Antwortzeit abschätzen zu können. Diese Kosten bestehen hauptsächlich aus Wartezeiten auf Timeouts. Die Antwortzeit bei Eintritt eines oder mehrerer Fehler kann mit der Zusammenstellung aus Tabelle 17 grob abgeschätzt werden.

Häufig von Interesse ist auch der durchschnittliche Fall (average case scenario). Um diesen Fall zu betrachten wäre es nötig eine Sequenz üblicher Anfragen, dazugehöriger Antworten und Ausfälle zu definieren und deren Ausführung für ein konkretes dezentrales LVS zu simulieren. Schon allein die Definition einer solchen Sequenz ist problematisch. Eine Simulation dieser Sequenz unter Berücksichtigung diverser Ausfälle (inkl. deren Eintrittswahrscheinlichkeit) liefert zudem lediglich Aussagen für den konkret simulierten Aufbau. Bei geringer Fehlerwahrscheinlichkeit ist die

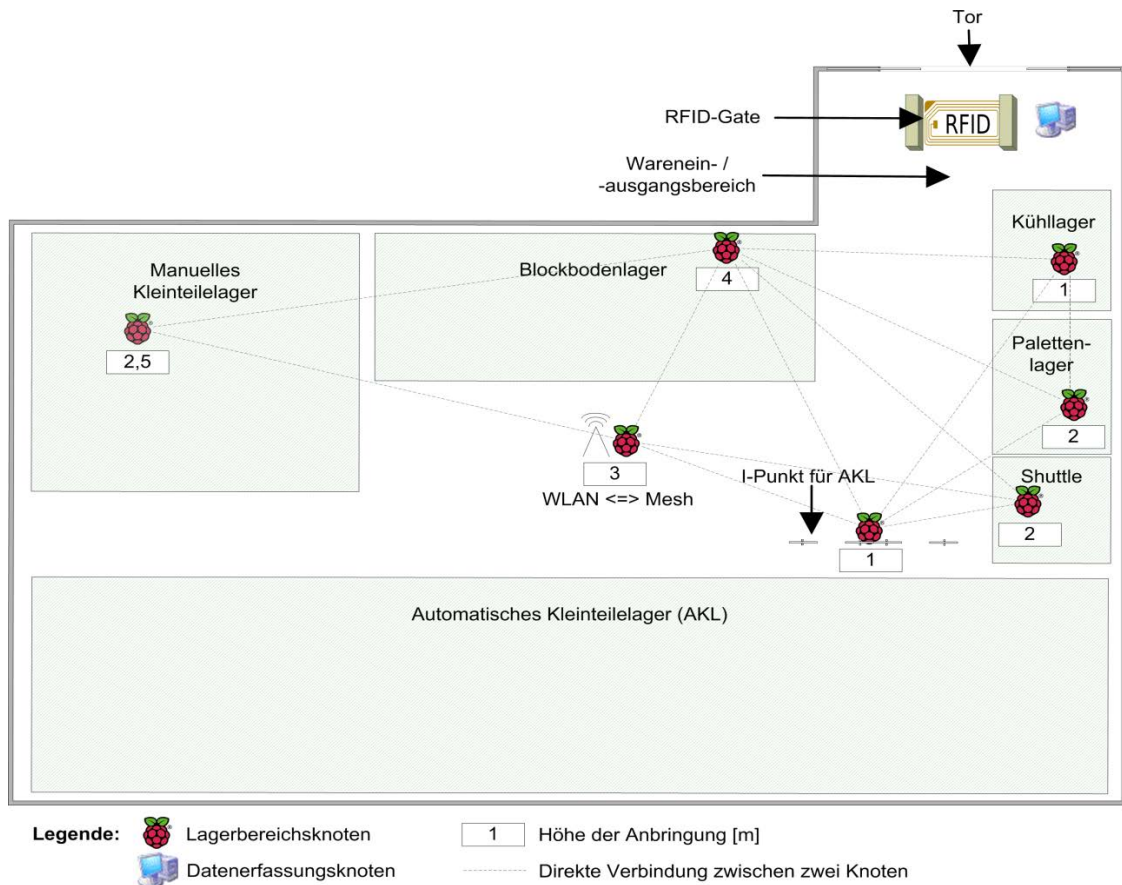


Abbildung 4 Lagerlayout des exemplarischen Aufbaus

Tabelle 16 Analyse der Ausführungszeit operativer und systeminterner Funktionen (best case)

Systeminterne Funktionen	Best case	Beispielwert ²	Beispielwert ³
Prüfung vor jeder operativen Funktion (5.3.1.9)	$t_{\text{signal}} + t_{\text{transport}} = 2 t_{\text{signal}}$	+0,2 s	+0,4 s
Operative Funktionen			
Suche (5.3.2.1)	$\left\lceil \frac{n}{m} \right\rceil * (t_{\text{signal}} + t_{\text{transport}}) = 2 * \left\lceil \frac{n}{m} \right\rceil * t_{\text{signal}}$	1,2 s	2,0 s
Reservierung (5.3.2.2)	$t_{\text{signal}} + t_{\text{signal}} = 2 t_{\text{signal}}$	0,2 s	0,4 s
Wareneinlagerung (5.3.2.3)	t_{signal}	0,1 s	0,2 s
Warenauslagerung (5.3.2.4)	t_{signal}	0,1 s	0,2 s
Inventur (5.3.2.5)	$2 * \left\lceil \frac{n}{m} \right\rceil * t_{\text{signal}}$ (analog Suche)	1,2 s	2,0 s
Auftrag erstellen (5.3.2.6)	$\left(\left\lceil \frac{i}{m} \right\rceil + 1\right) * t_{\text{signal}}$	0,2 s	0,6 s
Auftrag annehmen (5.3.2.7)	$2 t_{\text{signal}} + \left\lceil \frac{i}{m} \right\rceil * t_{\text{signal}} + \left\lceil \frac{i}{m} \right\rceil * t_{\text{signal}} + t_{\text{signal}}$ $= (2 * \left\lceil \frac{i}{m} \right\rceil + 3) * t_{\text{signal}}$	0,5 s	1,4 s
Ausführung bestätigen (5.3.2.7)	$t_{\text{signal}} + \left\lceil \frac{i}{m} \right\rceil * t_{\text{signal}} = \left(\left\lceil \frac{i}{m} \right\rceil + 1\right) * t_{\text{signal}}$	0,2 s	0,6 s

i : Anzahl I/O Knoten, n : Anzahl aktiver Knoten, m : maximale Anzahl gleichzeitig kontaktierter Knoten

² Antwortzeit für $t_{\text{signal}} = 100\text{ms}$, $n = 30$, $m = 10$, $i \leq m$

³ Antwortzeit für $t_{\text{signal}} = 200\text{ms}$, $n = 100$, $m = 20$, $m < i \leq 2m$

Tabelle 17 Zusätzliche Wartezeit bei bestimmten Fehlern

Fehler	Zusätzliche Wartezeit	Empfehlung	Eintrittswahrscheinlichkeit	Auswirkung
Paketverlust	TCP-Retransmission Time [RFC2988, RFC1122]	Dynamische Berechnung (siehe RFC 2988)	häufig, ggf. mehrfach	geringfügig
Aktiver Knoten antwortet nicht	Timeout für Knotenantwort	$\approx 10 * t_{signal}$	selten	mittel
Verzeichnisdienst antwortet nicht	Durch Suche nach Verzeichnisdienst (vgl. 5.3.1.2)	Kein eigener Timeout	selten	stark

Auswirkung der Fehler im Mittel gering und weist damit ein geringes Gewicht in der durchschnittlichen Antwortzeit auf. Diese liegt also im durchschnittlichen Fall nahe der Antwortzeit im günstigsten Fall. Der durchschnittliche Fall wird daher hier nicht betrachtet.

7.3 AUSWERTUNG

Die technisch minimal mögliche Antwortzeit beträgt t_{signal} . Normalerweise zu erwartende Antwortzeiten für Funktionen, die lediglich den Kontakt zu einem speziellen, vorab bekannten Knoten, benötigen (Reservierung, Ein-/Auslagerung) sind mit maximal $2t_{signal}$ sehr niedrig und stellen in einem dezentralen LVS kein Problem dar. Eine spürbare Verbesserung ist nicht möglich – solange eine Bestätigung zurückgesendet werden muss, entspricht für Anfrage und Antwort t_{signal} der minimal realisierbaren Antwortzeit. Wesentlichen Einfluss auf die Wartezeit bei Operationen zur Auftragsverwaltung hat die Synchronisierung der Auftragsdaten zwischen allen I/O Knoten. Da üblicherweise jedoch nur eine geringe Zahl dieser Knoten eingesetzt wird (je Mitarbeiter und je Flurförderzeug ein Knoten), ist auch die Wartezeit in diesen Fällen gering. Am aufwändigsten sind die an viele Knoten gerichteten Anfragen, also die Suche und Inventur. Doch selbst bei höherer Knotenzahl ist eine Antwortzeit im niedrigen einstelligen Sekundenbereich zu erwarten. Für die Akzeptanz eines dezentralen LVS stellt die Antwortzeit einer im Hintergrund automatisch ausgeführten Lagerplatzsuche oder einer deutlich seltener durchgeführten, manuellen Suche nach Ware kein nennenswertes Hindernis dar. Für eine Inventur ist der theoretisch bestimmte Wert tendenziell zu niedrig, da bei einer Suche nach allen Daten auf allen Knoten die Approximation der Übertragungsdauer nicht mehr genau genug ist. Allerdings ist auch eine deutlich höhere Antwortzeit für diese Operation aufgrund der seltenen Durchführung kein Problem für die Akzeptanz des dezentralen Ansatzes.

Auftretende Fehler werden in diesem Konzept durch eine über einen gewissen Zeitraum ausbleibende Antwort des Kommunikationsteilnehmers erkannt (Timeout). Die Antwortzeit verlängert sich im Fehlerfall etwa um die Länge dieses Timeouts. Ein einfacher Paketverlust führt nur zu einer geringfügigen

Wartezeit. Antwortet jedoch ein Knoten aus anderen Gründen (z. B. über einen längeren Zeitraum) nicht, so muss der vormals aktive Knoten als nicht mehr aktiv erkannt werden. Der Timeout dafür wird sinnvollerweise deutlich länger gewählt, da sonst bereits wenige Paketverluste den Knoten als nicht erreichbar kennzeichnen würden. Deutlich länger dauert es, wenn der bisherige Verzeichnisdienst nicht mehr erreichbar ist und ein Neuer gesucht werden muss. Beide Fälle treten jedoch nur selten auf. Der zeitgleiche Eintritt mehrerer Fehler verlängert die Antwortzeit der Operation unter Umständen noch einmal. Nicht alle Fehler wirken sich jedoch zwingend kumulativ aus. Antworten z. B. mehrere Knoten, die zur gleichen Zeit angefragt werden nicht, läuft der Timeout parallel aus und erhöht die Antwortzeit nur einmal. Sind allerdings zwei zu unterschiedlichen Zeitpunkten kontaktierte Knoten nicht erreichbar, addieren sich die Verzögerungen. Eine signifikante Verzögerung der teilweise mehrere Minuten dauernden Lagerprozesse ist nur in seltenen Ausnahmefällen bei gleichzeitigem Eintritt mehrerer Fehler zu erwarten. Diese Fehler können vorübergehend zusätzlich zu einer Einschränkung des Betriebs führen (vgl. Abschnitt 5.4). Um das Auftreten kritischer Fehler zu minimieren, sollte eine Monitoringfunktion zur Erkennung, vereinzelter Fehler in ein dezentrales LVS integriert werden. Treten vereinzelt Fehler gehäuft auf, kann dies ein Hinweis auf einen bevorstehenden Ausfall sein, der sich, wenn frühzeitig erkannt, in der Regel vorher beheben und dadurch vermeiden lässt.

Um die Auswirkungen durch Timeouts zu reduzieren, wird empfohlen den Wert des jeweiligen Timeouts als Kompromiss zwischen geringer Antwortzeit im Fehlerfall und geringer Wahrscheinlichkeit einer verspäteten (und daher nicht mehr berücksichtigten) Antwort bei korrekter Funktion zu wählen.

8 ZUSAMMENFASSUNG UND AUSBLICK

Das in dieser Veröffentlichung erläuterte, weitgehend technologieunabhängige Feinkonzept zeigt, dass ein aus dezentralen Einheiten bestehendes LVS umgesetzt und angewendet werden kann. Dabei sind aufgrund des verteilt gespeicherten Datenbestandes und

höherer Signallaufzeiten zusätzliche Maßnahmen für eine robuste Umsetzung der einzelnen Funktionen notwendig. Eine detaillierte Beschreibung zur Umsetzung der operativen Prozessabläufe und der im Hintergrund durchgeführten systeminternen Operationen liefert eine solide Basis für eine eigene Implementierung eines solchen Systems. Die Integration einer automatischen Datenerfassung mittels einer Ident-Technologie lässt sich ohne nennenswerten Zusatzaufwand gestalten.

Grundsätzlich gut geeignet für die Verwaltung der Daten und Einbindung von Ident-Technologien in das Lagerverwaltungssystem sind Singleboard-Computer. Diese kostengünstigen Recheneinheiten können ohne Probleme auf einem reduzierten Datenbestand die gleichen Verwaltungsfunktionen abdecken wie ein zentrales Lagerverwaltungssystem. Ein mobiles Interface zum dezentralen LVS kann mit handelsüblichen Tablet-PC realisiert werden. Als flexibles Kommunikationsmedium für die Knoten eines dezentralen LVS bietet sich der noch relativ neue Standard IEEE802.11s für ein drahtloses Mesh-Netzwerk an. Mit diesem Protokoll wird ein flexibles Netzwerk zwischen den Knoten eines dezentralen LVS aufgebaut – neue/entfernte Knoten werden automatisch integriert/gelöscht.

Im Konzept bisher nicht berücksichtigt ist die Nachvollziehbarkeit der Bestände in einem dezentralen LVS. Für diese Funktion ist es notwendig die Historie der Zustandsänderungen jeder Ware im Lager zu speichern und für Suchanfragen zur Verfügung zu stellen. Diese Funktion kann unabhängig vom bisher gewählten Konzept betrachtet werden. Problematisch ist vor allem der Ausfall einzelner Knoten, da in diesem Fall Teile der Historie verloren gehen können, die, analog zur Inventur, für die Erfüllung der Funktion aber nicht unberücksichtigt bleiben dürfen. Die Entwicklung eines Ansatzes zur Umsetzung der Historienfunktion mittels sicherer Datenspeicherung in dezentralen LVS erscheint daher sinnvoll.

Die theoretische Evaluierung des vorgeschlagenen Aufbaus zeigt im Normalfall akzeptable Antwortzeiten auf alle zeitkritischen operativen Funktionen des Lagerverwaltungssystems. Potentiellen Störfaktoren muss in einer Umsetzung besondere Aufmerksamkeit zuteil werden. Im dezentralen Fall lässt sich der sichere Betrieb eines LVS nicht bei Eintritt aller Fehlerarten gewährleisten. Die Auswirkungen der Störungen lassen sich aber oft mit vergleichsweise wenig Aufwand abmildern, so dass ein eingeschränkter Betrieb weiterhin möglich ist. Treten Fehler auf, erhöhen sich die Antwortzeiten abhängig von der Art des Fehlers und der jeweiligen Funktion. Insbesondere das zeitnahe Auftreten mehrerer Fehler kann zu starken Verzögerungen bis hin zum Ausfall des gesamten Systems führen. Um das Auftreten solcher Mehrfach-Fehler zu minimieren, sollte eine Monitoringfunktion, die

gehäuftes Auftreten einzelner Fehler erkennt und dann vor möglichen Ausfällen warnt, eingesetzt werden.

Intelligente Behälter sind bereits mit einer logischen Hardwareeinheit ausgestattet und können daher möglicherweise leicht zu einem vollwertigen Lagerbereichsknoten aufgerüstet werden und damit in ein dezentrales LVS integriert werden.

In dieser Veröffentlichung wird der Ansatz verfolgt, ein drahtloses Netz für die Kommunikation zwischen den dezentralen Komponenten zu verwenden. Ein ggf. sonst ohnehin benötigtes drahtloses Netzwerk für die Kommunikation der Flurförderzeuge mit dem Lagerverwaltungssystem ist dadurch nicht mehr erforderlich. Denkbar ist aber auch der umgekehrte Fall: bei Vorhandensein eines drahtlosen Funknetzwerkes könnten die zahlreichen, bereits installierten Zugangspunkte als Komponenten eines dezentralen LVS verwendet werden. Derartige Hardware besteht ebenfalls aus einem Rechensystem, auf das prinzipiell eine große Auswahl zusätzlicher Softwarepakete aufgespielt und damit zusätzlich für weitere Zwecke im Lager genutzt werden kann (vgl. [ope14a]). Diese Funktionsintegration ist nicht auf den Anwendungsfall eines dezentralen LVS begrenzt und bietet den Vorteil, dass keine Kosten für zusätzliche Hardware anfallen und eine Abwärtskompatibilität mit nicht Mesh-fähiger Netzwerkhardware gegeben ist.

Die bisher theoretisch untersuchten Antwortzeiten eines auf dem vorgestellten Konzept basierten dezentralen LVS sollten für mehrere, repräsentative Referenzszenarien, bestimmt und mit dem theoretischen Ansatz verglichen werden. Auf diesem Weg kann überprüft werden, in wie weit die vereinfachten Annahmen zur Berechnung ein gutes Abbild der Realität darstellen. Wichtige weitere, bisher nicht berücksichtigte Einflussgrößen können so ebenfalls identifiziert und in den theoretischen Ansatz integriert werden.

Hinweis: Der vorliegende Aufsatz ist im Rahmen des Projektes „Dezentrale Lagerverwaltung für wandlungsfähige logistische Systeme“ entstanden. Die Deutsche Forschungsgemeinschaft (DFG) förderte dieses Projekt unter dem Geschäftszeichen GU-427/13-1.

LITERATUR

- [AWS14] <http://aws.amazon.com/de/>, aufgerufen am 15.04.2014
- [Bel07] Bellifemine, F. et al.: *Developing Multi-Agent Systems with JADE*; John Wiley & Sons, Ltd, 2007, S. 72
- [Bla14] Black, P. E.: *first-in, first-out*, in *Dictionary of Algorithms and Data Structures* [online], Vreda Pieterse and Paul E. Black, eds. 17 December 2004. <http://www.nist.gov/dads/HTML/firstinfrstt.html>, aufgerufen am 26.04.2014
- [Bmb14] <http://www.bmbf.de/de/19955.php>, aufgerufen am 25.04.2014
- [Bre00] Brewer, E. A.: *Towards robust distributed systems*. (Vortrag), Principles of Distributed Computing, Portland, Oregon, July 2000
- [BSI08] BSI-Standard 100-2: *IT-Grundschatz Methodology*, 2008, S. 47f
- [Bus96] Buschmann, F. et al.: *Pattern-Oriented Software Architecture, Volume 1 - A System Of Patterns*; Wiley, Volume 1 edition, 1996, S. 71ff
- [Bus01] Busmann, S. et al.: *An Agent-based Approach to the Control of Flexible Production Systems*, 8th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA 2001), S. 169–174, Antibes Juanlespins, France, 2001
- [Cor12] Corbet, J. et al.: *Spanner: Google's Globally-Distributed Database*, Google Inc.; OSDI 2012
- [Dal11] Daldoul, Y. et al.: *The Impact of the Reliability on the Fairness Between the Unicast and the Multicast in Highly Loaded WLANs*, 11th IEEE International Workshop on Wireless Local Networks, 2011
- [Dec10] Decker, C. et al.: *K Revealing the Retail Black Box by Interaction* 23rd Int. Conf. on Distributed Computing Systems Workshops, 2003
- [Doo02] Dooley, K.: *Designing Large-scale LANs*, S. 31, O'Reilly, 2002
- [Ebn04] Ebner, A.: *Aspects of Decentralized Time Synchronization in Vehicular Ad hoc Networks*, WIT 2004 - 1st International Workshop on Intelligent Transportation, Hamburg, 2004
- [Eis07] Eisen, M.: *Designing & Deploying Wireless LAN Systems*, IEEE, 2007, http://www.ieee.li/pdf/viewgraphs/designing_deploying_wireless_lan_systems.pdf, aufgerufen am 25.04.2014
- [Fac06] Faccin, S. et al.: *Mesh WLAN Networks: Concept and System Design*, Wireless Communications, IEEE (Volume:13, Issue: 2), S. 10-17, 2006
- [fml14] <http://serverfml.fml.mw.tu-muenchen.de/Kompendium/index.php?title=Lokalisierungssystem>, aufgerufen am 04.02.2014
- [Gün09] Günthner, W. A. et al.: *Abschlussbericht an die Bayerische Forschungsstiftung über das Forschungsprojekt RFID in der Logistik – Werkzeuge zur Identifikation und Nutzung von RFID - Potenzialen*, fml-Lehrstuhl, Technische Universität München, 2009, S. 18-19
- [Him14] Himmel, M.A.: *Security on distributed systems: Cloud security versus traditional IT*, IBM Journal of Research and Development (Volume:58 , Issue: 1), 2014, S. 31-313
- [Ide13] IdentPro GmbH: *identplus – Automatisches 3D-Tracking von Waren und Behältern mit Staplern*. Produktbeschreibung, 2013
- [IEEE802.11] IEEE 802.11™: *Wireless LANs*; <http://standards.ieee.org/getieee802/download/802.11-2012.pdf>, aufgerufen am 26.04.2014

- [IEEE802.11s] IEEE Standard for Information Technology--*Telecommunications and information exchange between systems--Local and metropolitan area networks--Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking*, 2011
- [IIB09] IIBA: *A Guide to the Business Analysis Body of Knowledge*. International Institute of Business Analysis. Version 2.0; 2009, S. 102, S. 156
- [IML13] Fraunhofer-Institut für Materialfluss und Logistik: *inBin – Der intelligente Behälter*. http://www.impl.fraunhofer.de/de/themengebiete/automation_eingebettete_systeme/Produkte/IntelligenterBehaelter.html, aufgerufen am 25.10.2013
- [Jad12] Jadeja, Y.: *Cloud Computing - Concepts, Architecture and Challenges*; 2012; International Conference on Computing, Electronics and Electrical Technologies [ICCEET]
- [Kat13] Kathrein-Sachsen GmbH: *Neue KRAI RFID UHF SmartShelfAntenne*. <http://www.kathrein-rfid.de/de/node/104>, aufgerufen am 27.11.2013
- [Kil13] Kilz, S.: *Alles unter Kontrolle – Waren-verfolgung per 3D-Tracking*. In: IT&Production 5/2013, S. 68-69
- [Knu97] Knuth, D.: *The Art of Computer Programming, Volume 3: Sorting and Searching*, Third Edition. Addison-Wesley, 1997. ISBN 0-201-89685-0. Pages 158–160 of section 5.2.4: Sorting by Merging. Section 5.3.2: Minimum-Comparison Merging, S. 197–207
- [Kuz10] Kuzmany, F.: *Konzeption und Entwicklung von Modulen für das Internet der Dinge*; Dissertation, Technische Universität München, 2010
- [Lam74] Lamport, L.: *A New Solution of Dijkstra's Concurrent Programming Problem*; Communications of the ACM 17, 8; 1974; S. 453-455
- [Lam78] Lamport, L.: *Time, Clocks and the Ordering of Events in a Distributed System*; Communications of the ACM 21, 7, 1978, S. 558-565
- [Lam82] Lamport, L.: *An Assertional Correctness Proof of a Distributed Program*; Science of Computer Programming 2, 3; 1982; S. 175-206
- [Lib11] Libert, S.: *Beitrag zur agentenbasierten Gestaltung von Materialflusststeuerungen*, Dissertation, Praxiswissen Service, 2011
- [Log08] Logistik Inside, Ausgabe 03/2008; S. 50-51
- [Lya14] Lyamin, N.: *Real-Time Detection of Denial-of-Service Attacks in IEEE 802.11p Vehicular Networks*, Communications Letters, IEEE (Volume:18, Issue: 1), 2014
- [May09] Mayer, S.: *Development of a completely decentralized control system for modular continuous conveyors*; Dissertation, Wissenschaftliche Berichte des Institutes für Fördertechnik und Logistiksysteme der Universität Karlsruhe (TH) Band 73, Universitätsverlag Karlsruhe, 2009
- [Mei04] Meinel, C.: *WWW - Kommunikation, Internetworking, Web-Technologien*, Springer-Verlag Berlin Heidelberg New York, 1.Auflage, 2004
- [MIL79] MIL-STD-1574A, § 5.1.1.1; Kirkman, R.: *Evaluating Single Point Failures for Safety & Reliability*, IEEE TRANSACTIONS ON RELIABILITY, VOL.R-28, NO. 3, 1979
- [Mir14] Mirlach, M. et al: *Lagerverwaltung und Bestandsmanagement in dezentral gesteuerten Systemen*, Logistics Journal, 2014
- [mti14] <http://www.mti.com.tw/minime/>, aufgerufen am 27.04.2014

- [ope14a] <https://openwrt.org/>, aufgerufen am 26.04.2014
- [ope14b] The Open Group Base Specifications Issue 6; IEEE Std 1003.1, 2004 Edition; <http://pubs.opengroup.org/onlinepubs/009695399/toc.htm>; aufgerufen am 26.04.2014
- [RFI09] RFIDSpan Technology: *RFIDSpan Technology Introduces iReader-980 to Revolutionize Item-Level Tracking and SmartShelf Design*. Pressemitteilung 28.12.2009, <http://www.rfidspan.com/newsevents.html>, aufgerufen am 27.11.2013
- [RFC768] RFC768; <http://www.ietf.org/rfc/rfc768.txt>, aufgerufen am 26.04.2014
- [RFC791] RFC791; <http://www.ietf.org/rfc/rfc791.txt>, aufgerufen am 26.04.2014
- [RFC792] RFC792; <http://www.ietf.org/rfc/rfc792.txt>, aufgerufen am 26.04.2014
- [RFC793] RFC793, <http://www.ietf.org/rfc/rfc793.txt>, aufgerufen am 26.04.2014
- [RFC1122] RFC1122, <http://www.ietf.org/rfc/rfc1122.txt>, aufgerufen am 26.04.2014
- [RFC2460] RFC2460, <http://www.ietf.org/rfc/rfc2460.txt>, aufgerufen am 26.04.2014
- [RFC2988] RFC2988, <http://www.ietf.org/rfc/rfc2988.txt>, aufgerufen am 26.04.2014
- [Sch08] Schier, A.: *Drahtlose Sensornetzwerke in der Logistik: Einsatz drahtloser Sensornetzwerke zur dezentralen Lagerhaltung*, VDM Verlag Dr. Müller, 2008
- [Ste11] Stevens, W. R.: *TCP/IP Illustrated, Volume 1, The Protocols*, 2nd edition; Addison-Wesley Professional, 2011
- [Tan06] Tanenbaum, A.: *Distributed Systems – Principles and Paradigms*, Second Edition; Peason Prentice Hall, 2007, S. 17ff
- [Tan14] Tan, Z. et al.: *A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis*; *IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS*, VOL. 25, NO. 2, 2014
- [TPM14] TPM Main Specification Level 2 Version 1.2, Revision 116; https://www.trustedcomputinggroup.org/resources/tpm_main_specification; aufgerufen am 26.04.2014
- [Web12] Weber, R.: *Fraunhofer-Forscher zeigen den Behälter für alle Fälle*. In: *MM Logistik – Das Portal für Logistikentscheider*, Vogel Business Media, 2012
- [Wie07] Wieland, T.; Fenne, M.; Scholz, Ch.: *Drahtlose Sensornetze im industriellen Einsatz*. Projektblatt, Hochschule Coburg, 2007
- [Wol13] Wolf, O. et al: *Warehouse Management Systems – Heute und Morgen; Erkenntnisse der Internationalen Marktstudie WMS: 2000 – 2005*; http://www.warehouse-logistics.com/Download/Literatur/DE_Artikel_WMS_Heute-und-morgen_warehouse_logistics.pdf, aufgerufen am 12.03.2013
- [Wür13] Würth Industrie Service GmbH & Co. KG: *iBin – Bestände im Blick*. Pressemitteilung. Bad Mergentheim, 13.02.2013
- [Zim80] Zimmermann, H.: *OSI Reference Model-The ISO Model of Architecture for Open Systems Interconnection*, *IEEE TRANSACTIONS ON COMMUNICATIONS*, VOL. COM-28, NO.4, S. 425, 1980

Dipl.-Inf. Matthias Jung, Research Assistant at the Chair Fördertechnik Materialfluss Logistik, Technische Universität München

Dipl.-Ing. Trung Thanh Le, Research Assistant at the Chair Fördertechnik Materialfluss Logistik, Technische Universität München

Dipl.-Ing. Thomas Atz, Research Assistant at the Chair Fördertechnik Materialfluss Logistik, Technische Universität München

Prof. Dr.-Ing. Dipl.-Wi.-Ing. Willbald A. Günthner, Professor at the Chair Fördertechnik Materialfluss Logistik, Technische Universität München

Address: Chair Fördertechnik Materialfluss Logistik, Technische Universität München, Boltzmannstraße 15, 85748 Garching bei München, Germany, Phone: +49 089 289-15921, Fax: +49 089 289-15922, E-Mail: kontakt@fml.mw.tum.de