

STANDARDSOFTWARE

Die 6 größten Mythen im Überblick



1. STANDARDSOFTWARE IM BLICK	2
2. SECHS MYTHEN ÜBER STANDARDSOFTWARE	3
Mythos #1 - Standardsoftware ist <i>Plug and Play</i>	3
Mythos #2 - Standardsoftware ist einfacher in der Entwicklung	4
Mythos #3 - In der Logistik gibt es keine Standardsoftware	6
Strukturelle Folgen von Standardsoftware in der Logistik	7
Projektbezogene Organisation	7
Agile Softwareentwicklung	7
Mythos #4 - Standardsoftware ist fertig entwickelt	9
Mythos #5 - Das Standard-WMS ist bei der Implementierung fehleranfällig	11
Mythos #6 - Standardsoftware ist billiger als Individualsoftware	14
3. FAZIT	15

1. STANDARDSOFTWARE IM BLICK

Jede Neuanschaffung von Anwendungssoftware bringt eine Vielzahl von komplexen Entscheidungen mit sich. Oft unberücksichtigt, aber ganz zentral für die spätere Ausführung, ist dabei die Frage, ob man auf Standardsoftware oder eine Individuallösung setzen möchte. Gerade bei scheinbar einzigartigen Vorgängen, wie beispielsweise produktspezifische Kommissioniervorgänge im Lager, neigen viele Interessenten dazu, dem individualisierten Programm den Vorzug zu geben. Denn, so meint man, das eigene Lager hat ganz eigene Voraussetzungen und ist einzigartig. Tatsächlich führt der Schluss, dass die Individualsoftware ‚besser‘ zu den Anforderungen des eigenen Lagers passt, oftmals zu vermeidbaren Mehrkosten und höherem Risiko bei der Implementierung.

In der CIM sind wir schon seit langem der Überzeugung, dass man mit Standardsoftware in der Lagerlogistik viele Probleme vermeiden kann, die bei Individualsoftware oder bei individualisierter

Standardsoftware auftreten können.

Das vorliegende Whitepaper möchte die zentralen Fragestellungen zum Thema Standardsoftware analysieren: Was ist Standardsoftware? Welche Bereiche umfasst sie und inwiefern unterscheidet sich ein standardisiertes Programm von einer Individuallösung? Was sind die Hintergründe und die Strukturen, die bei der Entwicklung von Standardsoftware nötig sind? Wie wird Logistiksoftware als Standardsoftware umgesetzt und implementiert? Welche Vorteile bietet Standardsoftware letztlich für den Kunden?



2. SECHS MYTHEN ÜBER STANDARDSOFTWARE

Standardsoftware ist im Bereich der Lagerverwaltung bereits seit Anfang der 2000er ein gängiges Schlagwort. Ausnahmslos jeder Hersteller von WMS verweist auf die Standardisierbarkeit seines Produkts und preist die Vorteile an, die damit einhergehen.¹ In jüngerer Zeit gibt es jedoch Stimmen, die Standardsoftware höhere Kosten und geringere Individualisierbarkeit nachsagen.² Obwohl Standardsoftware im Logistikbereich einen Marktanteil nahezu 80% ausmacht, herrscht offenbar eine große Verunsicherung rund um das Für- und Wider von Individual- und Standardsoftware. Das Marketing verschiedener WMS-Hersteller hantiert aus diesem Grund gerne mit Stichworten wie ‚flexibler‘ oder ‚passgenauer‘ Standardsoftware, um dem Interessenten den Eindruck zu vermitteln, man kombiniere das Beste aus zwei Welten. Hinter solchen Manövern steht eine Debatte, die beherrscht wird von Mythen über explodierende Kosten, unnötigen Funktionen und hoher Komplexität, die mit der Anschaffung von Standardsoftware einhergehen. Trotz jahrelanger Produktion von Standardsoftware bleibt bei einigen Interessenten und Branchenbeobachtern die Überzeugung bestehen, dass Standardsoftware sich nicht an den jeweiligen Prozessen orientiere und daher hohe Anpassungskosten nach sich ziehe. Der eigentliche Kern einer solchen Unsicherheit liegt jedoch an den sechs Mythen, die die Debatte rund um Standardsoftware seit langem prägen und hartnäckig bestehen bleiben.

MYTHOS #1 – STANDARDSOFTWARE IST PLUG AND PLAY

Unter Standardsoftware versteht man für gewöhnlich marktreife und vorgefertigte Softwarelösungen, die ein definiertes Aufgabengebiet mit einem bestimmten Funktionsumfang abdecken. Sie zeichnet sich dadurch aus, dass sie unter gegebenen Bedingungen und definierten Parametern ohne weitere Anpassungen zuverlässig funktioniert. Zudem ist Standardsoftware „sofort“ verfügbar, sofern die Implementierung ohne Schulungen, Fachpersonal und Customizing durchgeführt werden kann. Die bekanntesten Beispiele für Standardsoftware sind Textverarbeitungsprogramme oder Lohn- und Buchhaltungssoftware, die auf einen breiten Markt zielen und eine geringe Parametrierbarkeit erfordern.³ So lässt sich die Standardsoftware ‚Word‘ von Microsoft ohne jegliche Einstellungen auf nahezu allen gängigen Betriebssystemen installieren.

¹ Vgl. WMS Marktreport 2020. Hg.: Fraunhofer IML

² So auch Werner Rieche im Jahr 2018, Geschäftsführer der Software AG Deutschland. In: <https://www.industry-of-things.de/ist-standardsoftware-noch-zeitgemaess-a-744278/>. Zuletzt aufgerufen am: 15.6.2021

³ Vgl. Schwarzer, Bettina; Krcmar, Helmut: Wirtschaftsinformatik. Grundlagen betrieblicher Informationssysteme (Stuttgart, 2010). S. 126f.

Hier liegt der Kern des ersten Mythos über Standardsoftware verborgen. Der hohe Bekanntheitsgrad von leicht zu bedienenden Alltagsprogrammen verführt dazu, die Eigenschaften dieser Programme zu verallgemeinern. Dabei ist es essentiell die jeweiligen Voraussetzungen der Branche bei der Verwendung des Begriffs ‚Standardsoftware‘ zu beachten: Textverarbeitungssoftware ist im Grunde eine Alltagssoftware mit überschaubarer Komplexität in Anwendung und Umsetzung. Die geringe Parametrierbarkeit ermöglicht die flexible Einsetzbarkeit des Systems – Plug and Play ist das Stichwort, mit dem hier geworben wird. Eingewöhnung und Funktionsweise kann dabei allein durch Manuals und Trial-and-Error des Benutzers ohne größeren Schaden vorgenommen werden.

Eine solche Vorgehensweise ist in vielen anderen Branchen – insbesondere der Logistik- und Fertigungsbranche – bei einer verantwortungsvollen Softwareentwicklung nicht möglich. Ein Plug-and-Play Vorgehen kann bei Lagerverwaltungssoftware schnell zum finanziellen Ruin des Logistikunternehmens führen. Wie weiter unten noch ausgeführt, erfordert die Logistikbranche nicht nur eine hohe Testbereitschaft der Softwareentwickler, sondern darüber hinaus zahlreiche Parameter, die zuvor von Fachpersonal definiert werden müssen. Erfolgt dies nicht mit gegebener Sorgfalt, können unerfreuliche Kollateralschäden entstehen.

MYTHOS #2 - STANDARDSOFTWARE IST EINFACHER IN DER ENTWICKLUNG

Die Idee, dass Standardsoftware einfacher in der Entwicklung ist, entspringt der Vorstellung, dass standardisierte Programme nicht an die spezifischen Bedürfnisse des Benutzers angepasst sind. Dabei ist genau das Gegenteil der Fall – Standardsoftware ist an die Bedürfnisse einer Vielzahl von Benutzern angepasst.

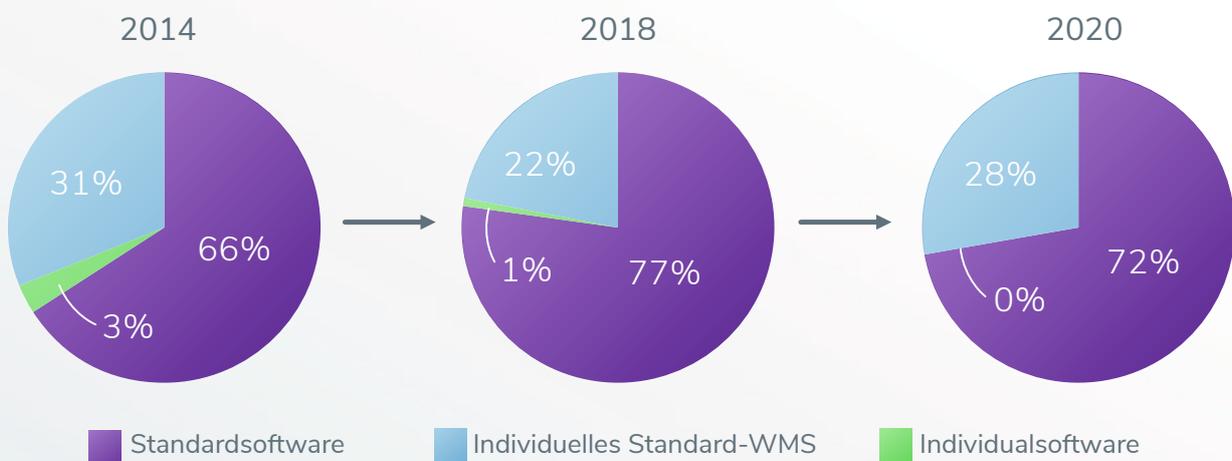
Soll ein informationsverarbeitendes Programm als Standard etabliert werden, müssen äußere Einflussfaktoren entweder integriert oder eliminiert werden. Dahinter steht die Erkenntnis, dass Standardsoftware - anders als Individualsoftware - nicht abgeschottet auf einem und demselben System läuft. Sondern sie ist mit unterschiedlichen Hardware- und Softwarebedingungen konfrontiert. Ziel der Entwickler ist es somit, ein Programm zu entwerfen, das möglichst wenig mit anderen Programmen und Betriebssystemen in Konflikt gerät. Bei einem Individualprogramm genügt es, die Anforderungen an das Programm dem bestehenden Hard- und Softwareangebot eines Interessenten anzupassen. Standardsoftware hingegen ist von vornherein darauf ausgelegt, mit unterschiedlichsten Systemen verschiedener potentieller und faktischer Interessenten und Nutzer zusammenzuarbeiten. Für Logistiksysteme bedeutet diese Anforderung einerseits, eine Plattform zu verwenden, die möglichst stabil und ohne Komplikationen läuft.

⁷ Vgl. WMS MARKTREPORT KOMPAKT 2020 - Trends und Entwicklungen auf dem Markt für Warehouse Management Systeme, Seite 19

Der Standard bei PROLAG®World ist somit die browserbasierte Steuerung, die auf nahezu jedem System ressourcenschonend funktioniert. Störfaktoren, die bei einer maschinen-näheren Programmierung unter Umständen auftreten, können somit von vornherein ausgeschlossen werden.

Ein weiterer Faktor, den es bei der Konzeption von Standardsoftware zu berücksichtigen gilt, ist die Implementierung von Schnittstellen. Damit es zu keinen Konflikten mit bestehenden Systemen kommt, gilt es die im Markt vorherrschenden Systeme – beispielsweise Warenwirtschaftssysteme – zu erforschen und mittels einer entsprechenden Schnittstelle einzubinden. Weder darf das Programm von seiner Umgebung gestört sein noch Störungen auslösen. Für die Entwickler hat die Herstellung einer Standardsoftware daher tiefgreifende Folgen, die sich in nahezu allen Bereichen des IT-Unternehmens niederschlagen.

Auslastung im Bereich Standardsoftware



Der Anteil der WMS-Anbieter, die ihr System selbst als Individualsoftware bezeichnen, verringert sich seit 2014 stetig. Es zeichnet sich die Tendenz zur Standardsoftware ab, wobei der Anteil der kundenspezifischen Anpassungen variiert. Einer projektspezifischen Programmierung kann durch die Parametrisierung und die Weiterentwicklung von Standardfunktionalitäten seitens der WMS-Anbieter vorgegriffen werden.

MYTHOS #3 – IN DER LOGISTIK GIBT ES KEINE STANDARDSOFTWARE

Die vielen zusätzlichen Parameter und Einstellungsmöglichkeiten, die eine Standardsoftware für die reibungslose Funktion abbilden muss, schlagen sich in der Organisation des Entwicklerteams und des Unternehmens nieder. Je nach Komplexität der Anwendungssoftware, gibt es unterschiedliche Strukturen, die in den jeweiligen IT-Unternehmen anzutreffen sind. Im Bereich der WMS sind die Marktteilnehmer mit der besonderen Situation konfrontiert, dass nach wie vor ein hoher Bedarf nach Individualisierung besteht.⁹

Im Diagramm des WMS Marktreports 2020 des Fraunhofer Instituts für Materialfluss und Logistik ist ersichtlich, dass der Anteil an individualisierter Standardsoftware in den vergangenen sechs Jahren trotz erkennbarem Trend kaum gesunken ist. Bis vor wenigen Jahren waren sogar noch Hersteller von Individualsoftware im Markt vertreten. Diese marktspezifischen Bedingungen resultieren aus den Bedürfnissen und Vorstellungen von Interessenten, sowie den oftmals fehlenden Kenntnissen über die spezifischen Prozesse im eigenen Lager.⁴ Hierin offenbart sich der größte Unterschied zwischen den Voraussetzungen für Standardsoftware in der Logistik und anderen Bereichen, in denen sich Standardsoftware vollständig durchgesetzt hat. Prozesse der Lagerverwaltung sind ungleich komplexer und erfordern ein weitaus größeres Fach- und Detailwissen, das je nach Art und Zustand der Lagerware und des Lagers variieren kann. Bereits die Beschreibung eines WMS im Allgemeinen benötigt über 1000 Einzelfunktionen, die wieder anwendungsspezifisch angeordnet werden müssen. Nur die Anordnung für einfachste logistische Grundfunktionen wie etwa Vereinnahmung, Lagerung, Bereitstellung oder Kommissionierung ergibt millionenfache Variationsmöglichkeiten.⁵ Je präziser also ein Prozess und sein Eintreffen von vornherein definiert ist, desto größer ist folglich auch die Menge an Daten, die es zu verarbeiten gilt. Allein die Verfügbarkeit eines Artikels innerhalb einer Kommissionierzone erfordert die Bestandspflege, die Reservierung des angeforderten Artikels, die Führung verschiedener Ressourcen bis hin zu der Arbeitskraft, die den Artikel schließlich entnimmt. Mit berücksichtigt werden müssen zahlreiche Fehlerstrategien, falls technische Elemente ausfallen oder vorausgeplante Ereignisse nicht eintreten.⁶ Das hohe Maß an einzelnen Parametern ist vergleichbar mit der Komplexität eines Flugzeugs (ca. 2 Millionen verzahnte Einzelteile). Es entsteht folglich der Eindruck, dass die Standardlösung dem eigenen Lager nicht gerecht wird, sondern Individualisierungen nötig sind.

⁴ Vgl. WMS Marktreport, S. 56

⁵ Vgl. ten Hompel, Michael: Intralogistik - Auf dem Weg vom Prozess zum Service. In Baumgarten, Helmut (Hg.): Das Beste der Logistik. Innovationen, Strategien, Umsetzung (Berlin, 2008). S. 105-110; S. 105.

⁶ Ebd. S: 105.

Strukturelle Folgen von Standardsoftware in der Logistik

Projektbezogene Organisation

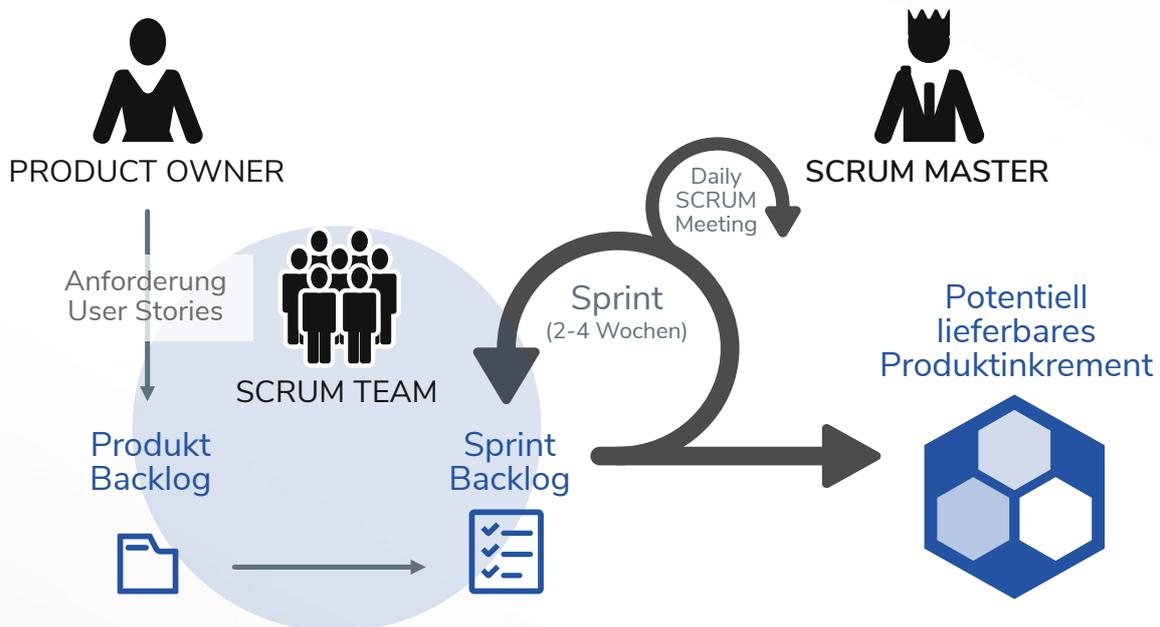
Aus diesem Grund hat sich in Entwicklungsmannschaften vieler WMS-Hersteller eine projektbezogene Implementierung eingebürgert, die die besonderen Ansprüche der Branche nach Individualisierung berücksichtigt. Bei einer solchen Organisationsstruktur leitet das Projektmanagement ein Team aus Entwicklern und Testern an, die die bereits vorhandene Standardsoftware an die spezifischen Bedürfnisse des Kunden anpassen. Geschäftsprozesse, die nicht im Standard enthalten sind, werden dann speziell für den Benutzer eingepflegt. Neue Funktionen werden hingegen in die bisherige Standardsoftware integriert. Die jeweiligen Projektteams müssen folglich Rücksprache halten, um auf demselben Standard arbeiten zu können, da sich die Systeme sonst voneinander unterscheiden.

Die projektbezogene Organisationsstruktur bringt jedoch entschiedene Nachteile mit sich. Unter anderem ist ein sehr hoher Grad an Kommunikation zwischen den Teams erforderlich, um den Standard trotz kundenspezifischer Individualisierung auf aktuellem Stand zu halten. Die Pflege des eigentlichen Standardsystems wird so zu einem zeit- und kostenintensiven Aufwand, der in der Praxis oftmals mehr als Belastung denn als Gewinn gesehen wird. Gerade durch die ohnehin hohe Arbeitsbelastung der einzelnen Projektteams während der Projektphasen gehen oftmals technische Neuerungen und Errungenschaften der einzelnen Teams unter. Die Partikularisierung der Standardsoftware, die dann von Team zu Team variiert, ist somit die Konsequenz aus der hohen strukturellen Konzentration auf individualisierte Standardsoftware.

Auch für den Kunden, der vermeintlich von der Individualisierung profitiert, erhöht sich durch projektbezogene Programmierung das Risiko für höhere Kosten sowie Systemfehler- und Bugs. Denn jede individuelle Anpassung wird mit Blick auf den jeweiligen Kunden entworfen, eingepflegt und getestet. Neben dem höheren Zeitaufwand entstehen so größere Risiken für Ausfälle, da die Tester nur die Daten dieses einen Kunden berücksichtigen können.

Agile Softwareentwicklung

Um diesen Problemen zu entgegnen, strukturiert die CIM seit einigen Jahren ihre Entwicklung nach der sogenannten Scrum Methode. Angelehnt an die Prinzipien agiler Softwareentwicklung organisiert sich die Entwicklungsmannschaft nicht mehr projektbezogen, sondern arbeitet nur noch zentral an der Standardsoftware. Diese bleibt somit immer auf dem neuesten Stand und ist das Referenzobjekt für alle laufenden Projekte.



Nach dem Organisationsprinzip der Scrum Methode teilen die jeweiligen Projektleiter die Wünsche und Bedürfnisse des Kunden, die außerhalb des Standards liegen, nicht mehr den Entwickler*innen mit, sondern dem sogenannten Product Owner.

Der/die Mitarbeiter*in, die diese Funktion ausfüllt, vergleicht die Anforderungen mit dem Standard, überprüft die Prozesse und gibt dann gegebenenfalls die erforderlichen Änderungen an die Entwicklung und Qualitätssicherung – das Scrum-Team – weiter.

Die Funktion des Scrum-Masters wird hingegen ausgefüllt von einer Mitarbeiterin oder einem Mitarbeiter, der/die als Außenstehende/r die Arbeit des Scrum-Teams unterstützt. Ein Scrum-Master ist eine dienende Führungskraft, die in täglichen Scrum-Meetings die Arbeit des Entwicklerteams Probleme und Schwierigkeiten bespricht, sowie Konflikte erkennt und deren Auflösung unterstützt. Im Zusammenspiel mit Product Owner, Scrum Master und Scrum Team werden schließlich mithilfe von festgelegten Prozessen die Ziele definiert und die Anpassungswünsche des Kunden direkt in den Standard integriert. Durch die enge Verzahnung von Entwicklung und Qualitätssicherung und der Arbeit an einer zentralen Standardsoftware ist bei dieser Organisationsstruktur das Risiko für zusätzliche Kosten durch doppelte oder unnötige Anpassungen reduziert. Zeitgleich kann der/die Projektleiter*in ein deutlich stabileres System garantieren, da die gesamten Testkapazitäten nicht auf mehrere ‚Projektsysteme‘ gleichzeitig aufgeteilt werden müssen, sondern immer nur auf ein einziges System – nämlich die Standardsoftware – konzentriert sind.

Für die Kunden als Stakeholder und die Softwarehersteller resultiert bei dem Organisationsprinzip nach der Scrum-Methode folglich eine Win-Win-Situation. Für den Softwarehersteller entfällt die kostspielige und zeitintensive versteckte Mehrarbeit projektbasierter

Entwicklung, da jede Anpassung an das System nur einmal gemacht werden muss. Der Kunde hingegen profitiert von einem günstigeren Gesamtprodukt sowie einem sichereren System. Nicht nur sind die Testkapazitäten im Scrum deutlich höher, sondern funktionale Neuerungen werden bei jedem Releasewechsel in die bestehenden Systeme der anderen Kunden integriert. Der Mehrgewinn an Test- und Praxisdaten führt unmittelbar zu einem stabileren System, da Bugs und Fehler schneller entdeckt werden können.

Die projektunabhängige Organisation führt zu einem weiteren Nebeneffekt, der gerne übersehen wird. Es hat nicht mehr nur eine Person – der Projektleiter – Kenntnisse über die Vorgänge und Prozesse der jeweiligen Endkunden. Zwar ist ein Projektmanagement zur Kommunikation, Vorbereitung und Implementierung nach wie vor nötig. Serviceleistungen und Fehlerbehebungen fallen jedoch in den Aufgabenbereich eines allgemeinen Serviceteams und sind nicht mehr ein Teil des Projekts. Die Konsequenz daraus ist eine umfangreiche Dokumentation, sowohl in der Programmierung als auch in den Funktionen der Standardsoftware. Der Benutzer ist letztlich unabhängig von Einzelpersonen und kann so auf eine strukturelle Verlässlichkeit bei Problemen und Serviceanfragen zählen.

MYTHOS #4 – STANDARDSOFTWARE IST FERTIG ENTWICKELT

Aus den bisher behandelten Hintergründen dürfte deutlich werden, dass bis zum Release(-wechsel) bei Standardsoftware zahlreiche Faktoren berücksichtigt werden müssen. Durch die Funktionsfülle des Softwarepakets sind die Testerfordernisse um ein Vielfaches höher. Dies ist unter anderem darin begründet, dass die Gefahr von Kollateralschäden für den Entwickler deutlich größer ist: Bugs betreffen bei Standardsoftware selten nur einen Kunden, sondern meist gleich zahlreiche.

Ein System, das sich selbst testet

Die automatisierten Tests werden nicht etwa einmalig vor dem Release durchgeführt, sondern das System ist dazu ausgelegt, diese im täglichen Ablauf der Entwicklung selbstständig durchzuführen. Die wichtigsten Logiken sind in einem Testframework hinterlegt, das automatisierte Tests ermöglicht. Die Software ist dabei so konfiguriert, dass auch die untertags eingeführten Änderungen abseits der Arbeitszeit überprüft werden. Zu Beginn des nächsten Arbeitstages erhalten Entwickler und Qualitätsmanagement die Testergebnisse, die darüber informieren, ob alle Funktionen noch wie erwartet vorhanden sind oder ob ein Kollateralschaden entstanden ist.

Es ist somit im Interesse des Softwareentwicklers, diese schon vor dem Release ausfindig zu machen und zu fixen. Um die Prozesse zu erleichtern, implementiert man in jede Standardsoftware automatisierte Tests, die mit jedem Update an die neuen Gegebenheiten angepasst werden. Eine eigene Testmannschaft sorgt bereits im Vorfeld für eine sichere Umgebung.

Ein Kritikpunkt, der Entwicklern von Standardsoftware gerne vorgeworfen wird, ist die Konzentration auf Releasezyklen und die damit einhergehenden Releasewechsel. Kritiker adressieren damit die Sorge, dass die individuellen Bedürfnisse des Benutzers zugunsten nutzloser Software-Updates mit neuen, aber unbrauchbaren Funktionen zurückgestellt werden. Tatsächlich handelt es sich hierbei um das irrige Versprechen, dass Individualsoftware weniger anfällig für Bugs, Fehler und Sicherheitslücken sei. Fakt ist jedoch, dass Softwareupdates und Funktionserweiterungen bei Individualsoftware deutlich aufwendiger sind und schon allein aufgrund der fehlenden Testdaten und Testkapazitäten niemals die Stabilität von Standardsoftware erreichen. Individualsysteme sind starr, Updates sind nicht die Regel und stellen immer einen Eingriff in die Entwicklungsumgebung dar.

Insbesondere beim Thema Cyber-Security ist der Glaube, dass keine Weiterentwicklung mehr vonnöten ist, nicht mehr für zeitgemäß halten. Die Logistik-Branche stellt heute eine kritische Infrastruktur dar und muss nicht nur in den Krisenzeiten einer Pandemie auf einem stabilen Fundament stehen, sondern auch gegen terroristische und schadhafte Angriffe auf die IT-Systeme gewappnet sein. Die Konzentration auf die Releasefähigkeit des Benutzersystems ist somit in einem größeren Kontext zu betrachten.

Releasewechsel-Vertrag

Die Releasewechsel von Standardsoftware werden bei CIM mit einem Releasewechselvertrag festgeschrieben, der die erforderlichen Serviceleistungen genau beschreibt. So bleibt PROLAG®World in puncto Sicherheit und Aktualität immer auf dem neuesten Stand – große Systemumbrüche, die einer Neuinstallation gleichkommen, gehören mit regelmäßigen Releasewechseln der Vergangenheit an. CIM erweitert permanent den Funktionsumfang durch zusätzliche Sicherheitsfeatures und optimierte Kompatibilität mit den aktuellen Technologien (z. B. Android für PROLAG®Go). Der Benutzer erhält eine Aktualisierung der jeweiligen Dokumentation und der Testumgebung, Mitarbeiter*innen werden geschult und auf Wunsch wird auch die Inbetriebnahme vor Ort von CIM Mitarbeiter*innen durchgeführt.

Der alleinige Blick auf den Funktionsumfang des Systems genügt hier nicht, um den Nutzen des Systems zu beschreiben. Der Einsatz von Standardsoftware entspricht durch die beständige Systembetreuung und Schulung den Prinzipien nachhaltiger und zukunftsorientierter Planung. Vor diesem Hintergrund wirkt der Mythos, dass Standardsoftware keiner Entwicklung mehr bedürfe, geradezu irrsinnig.

Die Weiterentwicklung und Pflege einer standardisierten Anwendung ist nicht nur im Interesse des Kunden, der von regelmäßigen Softwareupdates und einer ausführlichen Dokumentation profitiert, sondern auch im Interesse des Herstellers – und oftmals sogar ein wesentlicher Teil des Geschäftsmodells. Die Gepflogenheiten der jeweiligen Märkte bestimmen dabei die gängigen Abwicklungsmodelle. Anders als Benutzer häufig denken, ist auch bei sogenannten Softwaregiganten die Wartung und der Service zahlungspflichtig. Haben früher regelmäßig neue Releases die älteren Versionen abgelöst, hat sich in den vergangenen Jahren der Trend zu Abonnements durchgesetzt. Selbst bei komplett kostenfreien Softwarelösungen steht meist ein Profit im Hintergrund, der für den Benutzer nicht sichtbar ist. Im Zeitalter von Big Data haben sich die Methoden mit vermeintlich kostengünstiger Software Geld zu verdienen potenziert.

Die beständige Weiterentwicklung des Softwaresystems ist daher die größte Stärke, die Standardsoftware zu bieten hat. Während Individualsoftware stets auf den Ist-Zustand gerichtet ist, hält Standardsoftware bereits die Werkzeuge für eine zukünftige Bedarfsänderung bereit. Sie ermöglicht so ein hohes Maß an Investitionssicherheit.

MYTHOS #5 – DAS STANDARD-WMS IST BEI DER IMPLEMENTIERUNG FEHLERANFÄLLIG

Bislang unberücksichtigt ist die Frage, wie die Implementierung und das Customizing trotz der komplexen Voraussetzungen im Logistikbereich vonstattengehen. Standardsoftware in der Logistik basiert auf der Grundannahme vom Einsatz vorgedachter Geschäftsprozesse. Dabei gilt, dass die meisten Geschäftsprozesse standardisiert sind und in Wiederholungen ablaufen.

Wie die Praxis gezeigt hat, gibt es so viele Gemeinsamkeiten zwischen den Geschäftsprozessen verschiedener Unternehmen, dass es möglich ist, diese Logik in einer konfigurierbaren Software alleine durch Einstellungen zu implementieren. Um dies zu leisten, hat die Standardsoftware verschiedene Bedingungen zu erfüllen. Die wichtigste ist, dass die Standardsoftware über Modelle von Geschäftsprozessen verfügen muss, die den tatsächlichen Strukturen und Abläufen des Unternehmens ähnlich sein müssen. Dies geschieht nachvollziehbar und transparent mit Hilfe von Lagerdaten und entsprechenden Analysen der realen Geschäftsprozesse.

Zur Analyse muss die Standardsoftware Dokumente und Demonstrationsbeispiele liefern, die mit den vorhandenen Prozessen verglichen werden und eventuelle Lücken ans Tageslicht bringen. Entsprechende Testpläne oder Simulationen sind notwendig, um die Einsetzbarkeit vor einer Inbetriebnahme nachweisen zu können.

Die häufigsten Individualisierungen von Standardsoftware betreffen die Definition der vorhandenen Geschäftsprozesse im vorherrschenden Standard. Die bereits implementierten Geschäftsprozesse variieren je nach Lagerart, Art der Artikel und damit der Branche des Benutzers. So bietet beispielsweise die CIM mit PROLAG®World eine vorgefertigte Softwarelösung, deren integrierte Geschäftsprozesse an die Bedürfnisse unterschiedlicher Branchen angepasst sind. Die vorgedachten Prozesse sind im Hinblick auf die Funktionsreife des Systems und der gängigsten Lagerarten auf Effizienz, Arbeitskraft- und Materialkosten optimiert. Die Standardsoftware gibt so zu einem gewissen Grad die Geschäftsprozesse des Benutzers vor. Je nach Bedarf sind bestimmte Parameter zur Konfigurierung vorgesehen, die unter Umständen die Bedienbarkeit erleichtern und den Schulungsaufwand für Mitarbeiter*innen verringern. Bei der Bedarfsanalyse und dem Software-Design werden die vorhandenen Geschäftsprozesse somit im Vorfeld analysiert und gegebenenfalls angepasst, um optimale Ergebnisse mit der neuen Software zu erzielen.

Sollten die Geschäftsprozesse von den einstellbaren Parametern des gelieferten Standards abweichen, gilt es eine Kosten-Nutzen-Analyse durchzuführen. Denn im Normalfall erreichen komplett individualisierte Geschäftsprozesse, die zusätzlich in die gelieferte Software integriert werden müssen, nicht den Grad an Effizienz, der mit den errechneten und empfohlenen Prozessen möglich wäre.



Dagegen stehen die Kosten einer Individualisierung von Standardsoftware nicht im Verhältnis zu deren Nutzen, da mit einer Schulung der bedienenden Mitarbeiter*innen die Umstellung unkompliziert und problemlos unterstützt und gefördert werden kann. Entsprechen die Geschäftsprozesse nicht den Empfehlungen des Systems, müssen diese auch bei Updates und Funktionserweiterungen stets manuell angepasst werden, was die Fehleranfälligkeit des Systems stark erhöht. Automatisierte Tests greifen unter Umständen nicht mehr und unnötige Zusatzkosten entstehen aufgrund individueller Testerfordernisse.

Ist Standardsoftware stabiler? Eine komplexe Risikokalkulation:

Softwaresysteme bestehen in der Regel aus Klassen - im Fall PROLAG®World sind das etwa 5000 an der Zahl. In diesen Klassen sind Logiken (Methoden) implementiert, die von den verschiedenen Klassen aufgerufen werden, wenn sie benötigt werden. Im Schnitt sind in jeder Klasse ca. 10 Methoden (Logiken), die wiederum auf ca. 20 Datenfelder in der Klasse arbeiten.

Rufen Klassen Methoden auf und diese rufen erneut Methoden auf, so erzeugen diese eine Abhängigkeit. Man kann sich vorstellen, dass bei einer Menge von 5000 Klassen schon einfache Anforderungen sehr viele Klassen und Methoden aufrufen. Es entstehen also zahllose Abhängigkeiten. Wie hoch ist nun die Wahrscheinlichkeit, dass eine der Methoden und Klassen fehlerhaft ist?

Sind zwei Module voneinander abhängig, so ist die Gesamtwahrscheinlichkeit, dass die Funktion erfüllt wird, das Produkt aus den Einzelwahrscheinlichkeiten der Module. Besteht also ein System aus 10 voneinander abhängigen Modulen, die alle mit einer Wahrscheinlichkeit von 98 % einzeln ihre Funktion erfüllen, so beträgt die Gesamtwahrscheinlichkeit, dass die Gesamtheit dieser 10 Module seine Funktion erfüllt, nur noch 82 %.

Die Berechnung zeigt, dass eine neu erstellte Software einen enormen Testaufwand erfordert, der fast immer unterschätzt wird. Für die Softwarehersteller bedeuten die kumulierten Risiken einzelner Klassen die Ausarbeitung einer sehr komplexen und ausgefeilten Teststrategie. Da bei Standardsoftware viele Daten verfügbar sind und der Testaufwand für viele und nicht nur einen Kunden betrieben wird, können automatisierte Tests entwickelt werden, die eine beständige Überprüfung ermöglichen. Die Verlässlichkeit und die Stabilität von Standardsoftware ist individualisierter Software deshalb weit überlegen.

MYTHOS #6 – STANDARDSOFTWARE IST BILLIGER ALS INDIVIDUALSOFTWARE

Der Glaube, dass eine Spezialanfertigung teurer ist als ein Massenprodukt, ist tief verankert im Zeitalter industrieller Fertigung. Das Fließbandprodukt, das in großen Mengen gefertigt wird, ist notwendig günstiger als die Sonderanfertigung – die Materialkosten sind mengenrabattiert, die Arbeitsschritte vereinfacht und standardisiert. Sind diese Voraussetzungen in der Softwarebranche auch gegeben?

In der Softwarebranche hängt der Preis direkt von der Komplexität der Parametrierbarkeit des Systems ab. Im Gegensatz zu Individualsoftware hat Standardsoftware für gewöhnlich zahlreiche Parameter, um eine möglichst breite Zielgruppe anzusprechen. Der Kunde verlangt heute ein System, das im Standard seine Abläufe möglichst so abbildet, dass er sein Geschäft optimal betreiben kann. Das führt dazu, dass die Standardsysteme sehr flexibel konfigurierbar sein müssen, was wiederum zu einer Erhöhung der Komplexität und damit zu einem höheren Preis führt. Aus bereits erläuterten Gründen ist der Bedarf an Parametrierbarkeit von Lagerverwaltungssoftware branchenbedingt sehr groß.

Der Mythos, dass Standardsoftware günstiger sein muss als Individualsoftware beinhaltet somit nur einen geringen Wahrheitsgehalt. Zwar gibt es hier keinen höheren Materialverbrauch, aber bei steigender Komplexität der Parametrierbarkeit einen größeren Bedarf an Entwickler*innen. Dies führt notwendig zu höheren „Produktions“-Kosten und damit zu einem höheren Preis. Allerdings können die Voraussetzungen zur Entwicklung von Standardsoftware verbessert werden, wodurch die Entwicklungskosten auf einem Minimum gehalten werden. Ein Prozess, den die CIM mit der Umstellung auf agile Softwareentwicklung bereits vor Jahren angestoßen hat.

Die CIM kommt in Sachen Preisgestaltung dem Interessenten mit der Möglichkeit einer Branchenspezialisierung im Standard entgegen. Mithilfe der einzigartigen Ultraflex-Technologie geht das System auf die Anforderungen unterschiedlicher Branchen standardmäßig ein und kann auf die jeweiligen Vorgänge und Prozessoptimierungen problemlos reagieren. PROLAG®World bietet so eine hohe Konfigurierbarkeit, die mit ihren zahlreichen Einstellungsmöglichkeiten dem Benutzer einen hohen Grad von Unabhängigkeit vom Entwickler garantiert. Denn alle Einstellungen lassen sich eigenhändig durchführen. Zugleich kann mithilfe der branchenspezifischen Parameter die Komplexität der Standardsoftware möglichst gering gehalten werden. Je nach Branche werden überflüssige Parameter von vornherein ausgeschlossen - was letztlich die Investitionskosten auf das Nötigste eingrenzt.

3. FAZIT

Die sechs Mythen über Standardsoftware erfreuen sich einer großen Bekanntheit. Bei näherer Betrachtung erweisen sie sich jedoch als wenig differenziert oder sogar falsch. Dass Standardsoftware *Plug and Play* ist, lässt sich nicht verallgemeinern, sondern ist stark branchenabhängig. Gerade in der Logistikbranche wäre ein *Plug and Play* Vorgehen wenig sinnvoll und sogar verantwortungslos. Obwohl man im allgemeinen Sprachgebrauch ‚Standard‘ gerne Synonym mit ‚einfach‘ verwendet, ist Standardsoftware deutlich aufwendiger in der Entwicklung als dies bei Individualsoftware der Fall ist.

Aus diesem Grund müssen sich Hersteller von Standardsoftware intensiv mit der richtigen Herangehensweise bei der Organisation ihres Entwicklerteams beschäftigen. Dass Standardsoftware in der Logistikbranche dennoch der vorherrschende Trend ist, zeigt der WMS Marktreports 2020 vom Fraunhoferinstitut. Folgt man dieser Analyse wird der bisher noch umfangreiche Anteil individualisierter Standardsoftware auch im Logistikbereich in absehbarer Zeit verschwinden. Die Gründe hierfür sind vielfältig und liegen sowohl in der Organisation und den Strukturen des Entwicklungsteams als auch in den Ergebnissen der Kosten/Nutzen-Analyse von Standardsoftware im Vergleich zu Individualsoftware. Mit der Abkehr von projektbezogener hin zu einer agilen Softwareentwicklung verstärkt sich der Paradigmenwechsel der vergangenen Jahrzehnte. Mit der Scrum-Methode können erhebliche Vorteile in Sachen Systemstabilität, Investitionssicherheit und Kosteneffizienz sowohl für den Entwickler als auch für den Benutzer erreicht werden.

Auch die damit einhergehende Konzentration auf regelmäßige Releasewechsel lässt Standardsoftware schrittweise alternativlos in der Logistikbranche werden. Die ausgefeilte Teststrategie von Standardsoftware ermöglicht zudem eine stabilere Umgebung bei der Implementierung der Releasewechsel. Die zunehmenden Sicherheitserfordernisse, sowohl in Bezug auf Cyber-Kriminalität als auch beim Thema Arbeitssicherheit, lässt sich nur noch mit regelmäßigen Updates und standardisierten Geschäftsprozessen erfüllen. In diesem Zusammenhang ist auch mit dem letzten Mythos zu brechen: Seriöse Standardsoftware ist bereits aus strukturellen Gründen nicht günstiger als Individualsoftware. Mit steigender Komplexität der Parametrierbarkeit steigt auch der Preis – allerdings auch die Vorteile für die Benutzer. Aufgrund der zahlreichen integrierten Parameter lässt sich das WMS als Standardsoftware für die branchenspezifische und auch lagerspezifische Benutzung konfigurieren. Es bietet so ein Optimum an Usability - und zwar unabhängig von künftigen Veränderungen oder Erweiterungen. Das WMS als Standardsoftware ist somit eine Investition in zukunftssichere und nachhaltige Logistikdienstleistungen, die Benutzerfreundlichkeit und Stabilität langfristig garantieren können.

Sie haben noch Fragen? Unsere Experten sind gerne für Sie da:

CIM GmbH
Livry-Gargan-Straße 10
82256 Fürstfeldbruck

Tel.: +49 8141 5102-0
E-Mail: info@cim.de
Internet: cim-logistics.com