



---

## Standardsoftware versus Individualsoftware: Vor- und Nachteile im Vergleich

---



Whitepaper und technische Dokumentation

---

### Informationen zu diesem Dokument

**Autor:** Tobias Eichner, [tobias@starenterprise.com](mailto:tobias@starenterprise.com)

**Datum der Erstveröffentlichung:** Februar 2013 | **Datum der letzten Überarbeitung:** 12. Juni 2015

© STAR ENTERPRISE - [www.starenterprise.com](http://www.starenterprise.com) | Lizenzbedingungen: [www.starenterprise.com/de/bedingungen/](http://www.starenterprise.com/de/bedingungen/)

## Standardsoftware versus Individualsoftware: Vor- und Nachteile im Vergleich

Für jeden IT-Entscheider stellt sich irgendwann einmal die Frage, ob für die Realisierung eines bestimmten Projekts auf Standardsoftware zurückgegriffen oder die Entwicklung einer maßgeschneiderten Lösung in Form von Individualsoftware in Auftrag gegeben werden soll.

Um die Antwort aber gleich vorweg zu nehmen: Diese Frage kann pauschal nicht mit einem simplen „Ja“ oder „Nein“ beantwortet werden. Leider.

Das Weiterlesen lohnt sich aber dennoch...

Die Wahl einer bestimmte Methodik ist von einer Vielzahl Faktoren abhängig, die angesichts der Projektdetails unterschiedlich gewichtet werden müssen und sich zudem gegenseitig beeinflussen können.

Gedacht als kleine Entscheidungshilfe möchte ich im folgenden einige grundlegende Unterschiede der beiden Softwaretypen näher beleuchten.

### [Der Faktor Zeit]

Zunächst einmal spielt der für die Realisierung benötigte Zeitaufwand eine nicht ganz unwichtige Rolle. Zeit ist schließlich Geld, auch und gerade in der IT.

Standardsoftware steht naturgemäß sofort zur Verfügung. Oft können während der Realisierungsphase mehrere Produkte parallel auf ihre Eignung hin getestet werden. Bis zum Produktiveinsatz ist der Weg dann nur noch ein kurzer.

Bei Individualsoftware hingegen ist eine intensive Mitarbeit des späteren Anwenders (also des Auftraggebers) vonnöten:

Dies beginnt bereits in der Planungsphase, um eine möglichst vollständige Bedarfsanalyse durchführen zu können, setzt sich dann bei wiederkehrenden Previews während der Entwicklung fort und endet schließlich in der gemeinsam vorzunehmenden Implementierung.

Nachlässigkeiten bei der Planung führen oft zu zeitaufwändigen Anpassungen während der Realisierung. Mit unabsehbaren Folgen für die Einführung des fertigen Produkts. Dies kann bei zeitkritischen Projekten ein hohes Risiko in sich bergen und schlimmstenfalls das Scheitern des Projekts bedeuten.

### [Die Kostenfrage - eine Milchmädchenrechnung ?]

Nicht ganz unwichtig dürfte für viele Projekte auch die Einhaltung des verfügbaren Budgetrahmens sein. So weit, so gut.

Allerdings sind die reinen Anschaffungskosten (Lizenzgebühren bei Standardsoftware oder Entwicklungskosten für Individualsoftware) zumindest bei größeren Projekten immer nur ein Teilaspekt:

Insbesondere für unternehmenskritische Anwendungen (also solche, die essentielle betriebliche Abläufe unterstützen) ist die Gewährleistung von Support und die Verfügbarkeit regelmäßiger Aktualisierungen durch den Softwareanbieter essentiell.

Hierfür stehen im Bereich der Standardsoftware für gewöhnlich langfristig ausgelegte Support- und Update-Vereinbarungen zur Verfügung.

### **[Die Kostenfrage - eine Milchmädchenrechnung ?] (Fortsetzung)**

Bei Individualsoftware ist dies über einen Wartungsvertrag geregelt, welcher bestimmte Leistungen des Entwicklers dem Kunden in einem streng definierten Rahmen garantiert.

Im Fall von Standardsoftware kann meist aus einer Reihe vordefinierter Leistungspakete gewählt werden.

### **[Zukunftssicherheit = Investitionssicherheit]**

Die zu erwartenden Gesamtkosten tangieren daneben eine andere, ebenfalls nicht zu vernachlässigende Frage: Wie lässt sich langfristig die Sicherheit der getätigten Investitionen bewahren ?

Im Bereich der Standardsoftware kann es immer wieder einmal vorkommen, daß der dahinterstehende Anbieter sprichwörtlich die Segel streicht oder zumindest sich das Produkt schlichtweg in eine für den Anwender falsche Richtung entwickelt, sodaß es für den ursprünglich geplanten Einsatz zunehmend unattraktiv erscheint.

Das Ausfallrisiko des verantwortlichen Softwareanbieters trifft grundsätzlich auch auf eine individuell entwickelte Lösung zu. Mit einem Unterschied:

Der Auftraggeber erhält, eine entsprechende Vereinbarung vorausgesetzt, Zugriff auf die Sourcecodes (Quelltexte) und die technische Dokumentation. Damit wäre die weitere Betreuung der individuell entwickelten Applikation durch einen neuen Anbieter jedenfalls eine denkbare Option für den Notfall.

Neben der Sicherstellung einer möglichst langen Nutzungsdauer des erworbenen Softwareprodukts, ist auch die „Lebenszeit“ der mit der Software erzeugten Daten von großer Bedeutung.

Vor allem dann, wenn es sich um für das Unternehmen wertvolle Datenbestände handelt (z.B. Kundenprofile, Produktionsdaten) oder gesetzliche Vorgaben erfüllt werden müssen (z.B. Aufbewahrungsfristen bei digitalen Rechnungen, Buchungsdatensätze).

Aus technischer Sicht betrachtet, könnte dies durch die Offenlegung der in der Software verwendeten Datenformate und Schnittstellen geschehen oder aber durch eine möglichst universelle Exportfunktion.

Individualsoftware dürfte bei dieser Anforderung eher punkten, da die freie Wahl einer dem Einsatzzweck optimierten Lösung besteht.

Entsprechende Worst-Case-Szenarien müssen jedenfalls ausreichend berücksichtigt werden. Schließlich läßt sich mit einem „Legacy“-System nur so lange produktiv weiterarbeiten, wie sich die Rahmenbedingungen und Anforderungen nicht wesentlich verändern.

### **[Eignung und Anpassungsfähigkeit]**

Die günstigste „Out-of-the-Box“-Lösung ist sinnlos, eignet sie sich nicht oder nur bedingt für die Erledigung der an sie gestellten Aufgaben.

Es gilt also, die geforderte Funktionalität in allen Bereichen sicherzustellen. Dennoch wird Standardsoftware in vielen Fällen immer einen Kompromiss aus benötigten, verfügbaren und optionalen Features bedeuten.

### **[Eignung und Anpassungsfähigkeit] (Fortsetzung)**

Im ungünstigsten Fall könnte sogar die Anpassung bewährter betrieblicher Arbeitsabläufe an das verfügbare Softwareprodukt erforderlich sein.

Und branchenspezifische Lösungen sowie hoch spezialisierte Anwendungen sind mitunter am Softwaremarkt gar nicht erhältlich, da es schlichtweg an einem ausreichend großen Nutzerkreis mangelt, welcher die Entwicklung eines solchen Produkts lohnenswert erscheinen ließe.

Diese Disziplin könnte Individualsoftware deshalb systembedingt für sich gewinnen. Allerdings macht das die Entscheidung nicht einfacher, wie wir gleich im nächsten Abschnitt sehen werden.

### **[Technische Qualität entscheidet über Produktivität]**

Nur mit (möglichst) fehlerfreier und ergonomisch ausgereifter Software ist die Erzielung einer hohen Produktivität möglich. Das steht fest. Weniger gesichert ist allerdings, ob nun Standardsoftware oder Individualsoftware diese Kriterien am besten erfüllen kann.

Standardsoftware besitzt einen hohen Verbreitungsgrad, wird also von vielen Benutzern tagtäglich eingesetzt. Bugs lassen sich so schnell lokalisieren und durch den Entwickler beheben. Theoretisch zumindest.

In der Praxis obliegt Standardsoftware meist starren Entwicklungszyklen und nicht jeder gefundene Fehler wird auch tatsächlich durch ein Update behoben. Die Gründe sind oft technischer Natur, können aber ebenso der Produktpolitik des Herstellers geschuldet sein (Kosten-/Nutzenverhältnis, "das ist kein Bug, sondern ein Feature").

Bei Individualsoftware geschieht die Fehlerbehebung durch die mögliche enge Zusammenarbeit zwischen Auftraggeber und Entwickler wesentlich schneller und zielgerichteter. Voraussetzung ist jedoch meist ein entsprechender Wartungsvertrag.

### **[Software ist nie fehlerfrei !]**

Davor ist weder Standard- noch Individualsoftware gefeit.

Designfehler können durch eine sorgfältige Planung vermieden (oder zumindest reduziert) werden. Syntaxfehler und technische Inkompatibilitäten lassen sich mit dem Einsatz moderner Entwicklungsumgebungen und Testwerkzeuge nahezu ausschließen.

Ein großes Problem stellen jedoch Logik- und Laufzeitfehler dar, die sich erst im laufenden Betrieb (nach längerer Nutzung) bemerkbar machen und deren Behebung mitunter äußerst zeit- und damit kostenintensiv ist.

Ein Maß zur Bestimmung der Softwarequalität ist die „Fehlerrate pro 1000 Codezeilen“, die statistisch ermittelt werden kann.

Eine Rate von null Fehlern läßt sich - besonders bei technisch anspruchsvollen Anwendungen - so gut wie nicht erreichen.

Typisch sind hier Werte zwischen 0,3 und 0,7 Fehlern. Mit steigender Komplexität der Software steigt auch unweigerlich die Fehlerrate.

Neben den Kosten für die bloße Realisierung einer Softwarelösung müssen auch die Ausgaben für deren regelmäßige Wartung und Pflege berücksichtigt werden.

## Zusammenfassung

### [Allgemeingültige Vorteile der Standardsoftware]

- Relativ günstig in der Anschaffung.
- Sofort verfügbar und einsatzbereit ("out-of-the-box").

### [Allgemeingültige Vorteile der Individualsoftware]

- Exakte und vollständige Umsetzung aller Anforderungen.
- Flexible Anpassung an zukünftige Bedürfnisse.
- Hohe Investitionssicherheit.

### [Mögliche Risiken beim Einsatz von Standardsoftware]

- Ungünstige Entwicklung der Produktpolitik des Herstellers.
- Unzureichende Support-Optionen.

### [Mögliche Risiken beim Einsatz von Individualsoftware]

- Hohe zusätzliche Kosten durch Fehler bei Planung und Realisierung.

### [Fazit]

Standardsoftware oder Individualsoftware ?

Das ist eine Frage, die sich pauschal schier unmöglich beantworten lässt.

Für jedes Projekt sollte das Für und Wider sorgfältig abgewogen und in Zusammenarbeit mit einem erfahrenen Softwareentwickler eine geeignete Lösung ausgearbeitet werden, die den Bedürfnissen des Auftraggebers gerecht wird.

Ich hoffe, ich konnte einige Denkansätze und Diskussionsgrundlagen liefern, die dabei helfen, die richtige Entscheidung zu treffen.

## Über STAR ENTERPRISE

### [Informationstechnologie ist unser Universum !]

STAR ENTERPRISE bietet als **IT- und Business-Agentur** ein breitgefächertes Angebot **professioneller Produkte und Dienstleistungen** für Firmenkunden aller Branchen und jeder Größe.

Unser Portfolio reicht von **Unternehmensberatung** über **Softwareentwicklung** bis hin zu **IT-Services**. Mit innovativen Lösungen unterstützen wir unsere Kunden dabei, die Chancen moderner Informationstechnologien effektiv und gewinnbringend zu nutzen.

**Unsere Philosophie ist einfach:** Wir möchten **langfristige Kundenbeziehungen** aufbauen. Das erreichen wir als Geschäftspartner durch **verantwortungsvolles und ergebnisorientiertes Handeln**.

## Kontaktinformationen

### [Autor]

Sie haben **themenspezifische Fragen zu diesem Whitepaper** und möchten mit dem Autor in Kontakt treten ? Gerne. Auf der Titelseite finden Sie den Namen und dessen persönliche E-Mail-Adresse.

### [STAR ENTERPRISE]

Wir freuen uns darauf, mit Ihnen gemeinsam die **Möglichkeiten für Ihr Unternehmen** zu besprechen. Unsere **individuellen Leistungen und unser freundlicher Service** werden Sie bestimmt überzeugen.

Bitte wenden Sie sich an **unseren Kundenservice**: [www.starenterprise.com/de/beratung-angebot-anfordern/](http://www.starenterprise.com/de/beratung-angebot-anfordern/)  
E-Mail-Adresse für **allgemeine Anfragen**: [info@starenterprise.com](mailto:info@starenterprise.com)

## Website

**STAR ENTERPRISE:** [www.starenterprise.com](http://www.starenterprise.com)