

Standardsoftware vs. Individualentwicklungen im Bankenbereich

Der richtige Mix entscheidet. Der Einsatz von Standardsoftware insbesondere im Bereich der sogenannten Kernbankensysteme ist in deutschsprachigen Finanzinstituten allgemein ein positiv belegtes Thema. Individualsoftware wird im Gegensatz dazu häufig mit veralteten Legacy-Systemen gleichgesetzt, die i.d.R. noch auf Großrechnersystemen laufen.



Autor: Arne Schultz,
Leiter Development &
Integration Services bei
der innobis AG

In seiner Studie „IT Finanzarchitektur – Zufallsprodukt oder gezielte Weiterentwicklung“ sieht PriceWaterhouseCoopers sogenannte „individuelle Datenverarbeitungen“ in erster Linie als Risikopotenzial – eine Einschätzung, die offensichtlich auch durch die nur spärlich verfügbaren Management-Informationen bzw. die hohe Abhängigkeit von wenigen Key-Playern („Kopfmonopole“) entsteht. Dazu passt auch der Trend im deutschsprachigen Raum, dass die Anzahl der auf Standardsoftware setzenden Banken zunimmt. Eine Studie von Ernst & Young spricht zum Beispiel im Bereich des Schweizer Private Banking von einem Anteil von zwei Dritteln. Aktuelle prominente Beispiele in Deutschland sind u.a. die Deutsche Bank (Projekt Magellan) oder die KfW (Einführung SAP Finance).

Im Bereich der Sparkassen- und Genossenschaftsverbände steigt in den letzten Jahren durch die Zentralisierung der IT-Dienstleistungen in gemeinsamen Servicegesellschaften der Verbreitungsgrad zumindest gemeinsam genutzter Software, auch der Großteil der deutschen Förderbanken setzt weitestgehend auf Standardsoftware. Gleichzeitig lässt sich erkennen, dass der Anteil der Legacy-Kernbankensysteme weiterhin hoch ist. Noch bedeutsamer ist aber die Tatsache, dass offenbar mit dem Einsatz von Standardsoftware nicht zwangsläufig



Das Neben- und Miteinander von Standardsoftware und Individualentwicklung ist bei Geldinstituten üblich

der Verzicht auf – durchaus umfängliche – Individualentwicklung einhergeht. So gaben 54 Prozent der befragten Banken in der oben genannten PWC-Studie an, an der Nutzung von individuellen Datenverarbeitungen festhalten zu wollen. Wie passen diese augenscheinlich gegenläufigen Aussagen zusammen?

Erwartungshaltung von Kunden

Um sich dieser Frage zu nähern, ist es wichtig, sich zunächst einmal mit den Begrifflichkeiten auseinander zu setzen. In Wikipedia sind Standardsoftware „Softwaresysteme [...], die einen klar definierten Anwendungsbereich abdecken und als vorgefertigte Produkte erworben werden können“.

Die Erwartungshaltung von Finanzinstituten an Standardsoftwarehersteller geht

deutlich über diese Definition hinaus; gegenüber einem vollständigen Individualansatz bzw. einem „Best-of-Breed“-Ansatz werden insbesondere folgende Parameter gefordert:

- Kosteneinsparungen,
- Sicherheit,
- Zukunftsfähigkeit,
- Flexibilität,
- Vereinfachung von Standardprozessen,
- fortlaufende Weiterentwicklung (z.B. mit Blick auf gesetzliche Änderungen).

In der Realität werden aber nicht immer alle Erwartungen erfüllt, insbesondere führt die Einführung von Standardsoftware nicht zwangsläufig dazu, dass die Ausgaben für die Erstellung und Pflege von Eigenentwicklungen deutlich zurück gehen. So ergibt eine aktuelle Forrester-Umfrage, dass der Kostenanteil für Standardprodukte und Eigenentwicklung bei Anwenderunternehmen etwa gleich hoch ist.

Außerdem gelingt es selten, den vollen Funktionsumfang der teilweise über mehrere Jahrzehnte gewachsenen 'Alt'-Systeme in einem Schritt zu übernehmen, was häufig ein großes Problem für die Akzeptanz bei den Anwendern darstellt.

Erfolgskriterien für den Einsatz von Standardsoftware

Standardsoftware allein ist daher kein 'Allheilmittel'; der erfolgreiche Einsatz hängt stark von der Berücksichtigung einiger grundlegender Rahmenbedingungen ab, die über den reinen Abgleich des benötigten Funktionsumfangs hinausgehen:

1. Realistische Erwartungshaltung. Standardsoftware entsteht als Synthese aus Anforderungen (möglichst vieler) initialer Projektpartner und in der Folge über Weiterentwicklungen, die i.d.R. immer über neue Software-Kunden finanziert werden müssen. Ihre meist hohen Erwartungen an die Weiterentwicklung / funktionale Erweiterung sind daher nur dann realistisch, wenn das Marktpotenzial nicht bereits ausgeschöpft ist und der Softwarehersteller für neue Funktionen Projektpartner als Inputgeber gewinnen kann.
2. Gezielter Einsatz. Standardprodukte haben ihren höchsten Nutzen in für Banken differenzierungsunkritischen Bereichen, die im Normalfall institutsübergreifend ähnlich abgebildet werden können. In Bereichen, die für die Differenzierung gegenüber dem Wettbewerb kritisch sind, können Standardlösungen und die damit verbundene Anpassung von Prozessen an die Software sogar schädlich sein.
3. Technologieplattform mit einbeziehen. Die reine funktionale Abdeckung ist nicht der einzige Nutzen, den Standardsoftware verspricht. Bei Kernbankensystemen erwirbt der Käufer i.d.R. ein komplettes Ökosystem aus Technologieplattform, Komponenten für Querschnittsfunktionen wie z.B. den Datenaustausch mit fremden Systemen und nicht zuletzt eine Programmiersprache.
4. Grenzen akzeptieren. Jedes Institut verfügt über Prozesse und Berechnungsvorschriften bei denen ausgefeilte Konfigurationsmöglichkeiten von Standardprodukten nicht weiterhelfen. Obwohl naheliegend, ist es hier häufig sinnvoller, auf ein 'Verbiegen' des Standards zu verzichten. Auch wenn dies auf den ersten Blick kostengünstiger erscheint, wird der kurzfristige Vorteil vielfach mit langfristigen Problemen erkauft (Seiteneffekte der Anpassungen auf viele durch den Standard abdeckbare Produkte).

Geeignete Einsatzbereiche für Standardsoftware im Bankenbereich

Wenn diese Aspekte berücksichtigt werden, verspricht der Einsatz von Standardsoftware einen spürbaren Nutzen gegenüber vollständigen Individualansätzen. Standardsoftware hat überall dort gute

Chancen, wo viele Banken sich eine gleichartige Motivation und ähnliche Anforderungen teilen. Im Bankenwesen trifft dieser Sachverhalt insbesondere auf die folgenden Bereiche zu:

- Kernbankensysteme – wobei der Trend hier klar in die Richtung geht, 'Servicebaukästen' anzubieten, die dann von weiteren Systemen (insbesondere Front-Ends) verwendet werden können.
- regulatorische Themen – zu denen vornehmlich das Meldewesen und Risikocontrolling, gehören.
- Rechenkerne – etwa solche, die von gesetzlichen Auflagen beeinflusst sind (z.B. für die Berechnung von Effektivzinsen oder Vorfälligkeitsentschädigungen).
- Technologieplattformen – die gerade für die Themen Laufzeitumgebung Interoperabilität zwischen Systemen, Sicherheit o.ä. Lösungsansätze bieten.

Häufig verfehlt Erwartungen

Neben erfolgreichen Beispielen gibt es aber auch Bereiche, in denen verschiedene Anwender sehr schwer unter einen Hut zu bekommen sind. Die Generalisierung von Problemfeldern ist dort derart komplex, dass prinzipiell funktionierende Lösungsansätze von anwendenden Unternehmen nicht mehr zielführend einsetzbar sind. Hierzu zählen aus Sicht des Autors z.B.:

- nischenspezifische Fachfunktionen, da hier die Fachlichkeit in den Standardprodukten aufgrund der geringen Fallzahlen nicht gut genug abgebildet werden kann,
- Workflow-Management-Systeme, insbesondere in Bereichen, in denen der Fokus nicht auf organisatorischen Abläufen mit wenigen Daten, sondern auf der integrierten und dynamischen Bearbeitung komplexer Sachverhalte liegt,
- Out-of-the-box-Lösungen für regulatorische Themen, da diese auch von den Standardsoftwareherstellern i.d.R. erst mit mehreren Partnern aufgearbeitet werden müssen und die Banken – trotz anderer Erwartungshaltung – nicht von der Notwendigkeit entbinden, eigene konzeptionelle Ansätze zu entwickeln.

Qualitätskriterien für Standardsoftware

Was aber zeichnet nun eine besonders erfolgreiche Standardsoftware aus? Wenn man ein wenig hinter die Kulissen der gerne in Analysen bemühten Dimensionen „Completeness of Vision“ und „Ability to Execute“ schaut, dann sind aus Sicht des Autors – außerhalb der Ebene 'Funktionsumfang' – die folgenden Sachverhalte relevant:

- erkennbare und stabile Releasestrategie/Plattformstrategie: Ein verlässlicher Prozess für die Erstellung und Auslieferung sowie eine stabile technologische Basis sind unverzichtbare Grundlage für den Betrieb einer Standardsoftware.

- erkennbare und stabile Methodik der Softwareentwicklung: Globale Ansätze (z.B. SOA) sollten ebenso durchgängig erkennbar und verwendet sein wie Detailspekte wie z.B. Unit-Tests; gute Ökosysteme bieten hier auf allen Ebenen Tools und Templates und entbinden Unternehmen so davon, für jeden Bereich eigene Richtlinien erstellen zu müssen.
- Erweiterbarkeit / Anpassbarkeit ohne Hersteller: Releasesichere Vorgehen zur kundenspezifischen Erweiterbarkeit inklusive eines definierten Prozesses im Rahmen von Updates sind unverzichtbare Grundlage für die hauseigene Softwareentwicklung.
- Mandantenfähigkeit: Die Separierbarkeit von Geschäftsbereichen erleichtert bei sich verändernden Unternehmensstrukturen die Systemanpassung.
- vorhandenes Ökosystem von Dienstleistern (von Betrieb bis Beratung): Eine ausreichende Auswahl von Anbietern ermöglicht sowohl Leistungsspitzen abzudecken als auch unterschiedliche Sourcing-Ansätze, ohne eine zu große Abhängigkeit entstehen zu lassen.

Erfolgskriterien für ein erfolgreiches Nebeneinander

Auch für die erfolgreiche Integration von Individualsoftware lassen sich einige Best-Practice-Ansätze ausmachen

- möglichst offene / technologisch reife Standardplattform: Die Basis für Eigenentwicklungen sollte sich im Hinblick auf notwendige Infrastruktur, Vorgehensmodelle und technologische Ansätze möglichst nicht von der der wichtigsten Standardanbieter unterscheiden.
- Eigenentwicklungen 'nach dem Modell des Herstellers': Aspekte wie Mandantenfähigkeit, Releasefestigkeit, Nutzung von Infrastrukturwerkzeugen wie z.B. Logging sollten auch bei Individualsoftware möglichst nach den Standards des Herstellers erfolgen.
- fachliches Know-how auch im Standard-Bereich nicht vernachlässigen: Auch, wenn bestimmte Prozesse vollständig über Standardsoftware abgedeckt werden, stellt der Erhalt des fachlichen Wissens in diesen Bereichen die unverzichtbare Grundlage für mögliche Erweiterungen dar.

Sofern durchgängig berücksichtigt, erlauben es diese Ansätze, dass auch Individu-

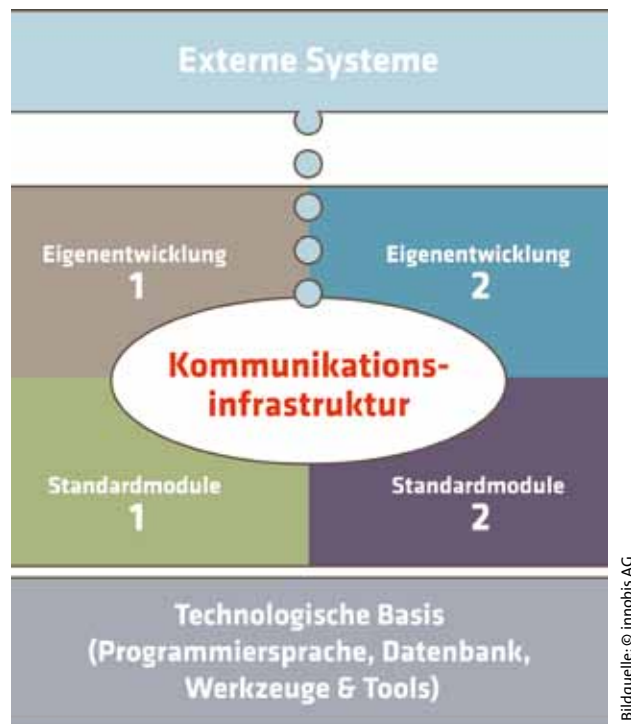
alsoftware von den technischen Innovationen von Standardplattformen profitieren. Darüber hinaus können bei einheitlichen Verfahrensweisen Schnittstellen- und Betriebskosten gesenkt und die Abhängigkeit von wenigen Spezialisten reduziert werden.

Schwarz und Weiß muss nicht Grau ergeben

Die wenigsten Banken setzen heutzutage noch auf die vollständige Eigenentwicklung von Kernbanksystemen. Auf der anderen Seite werden Standardlösungen für Kernbanksysteme i.d.R. für eine Bank keine vollständige Abdeckung aller notwendigen Prozesse bieten. Das Neben- und Miteinander von Standardsoftware und Individualentwicklung ist daher eine Tatsache und die eigentliche Herausforderung besteht darin,

„Nach unseren Erfahrungen ist die Kombination aus einer stabilen und ausgereiften Standardsoftware inklusive einer aktuellen Technologieplattform und analog der nach Herstellermethoden entwickelten Individualsoftwarekomponenten die beste Lösung für Banken.“

Arne Schultz, Leiter Development & Integration Services bei der innobis AG



Entscheidend für das erfolgreiche Nebeneinander von Standardmodulen und Individualentwicklungen ist die Nutzung einer einheitlichen technologischen Plattform sowie von Querschnittsfunktionen

die beiden Teile zu einem Ganzen mit Mehrwert und nicht zu zwei isolierten Welten werden zu lassen. Dies gelingt am besten durch:

- das Orientieren an Vorgehen und Methodik des Herstellers für Individualentwicklung,
- die soweit wie möglich einheitliche Plattform-Nutzung,
- das Zulassen enger Kopplung von Individual- an Standardsoftware dort, wo diese nicht alleine 'lebensfähig' wäre und damit Verzicht auf teure Overhead-Kosten z.B. durch SOA-Ansätze ohne erkennbaren Mehrwert,
- das Einhalten des Single-Point-of-Truth-Prinzips über die gesamte Applikationslandschaft hinweg (dieses Prinzip gilt für Daten ebenso wie für zentrale bankfachliche Berechnungen z.B. die Ermittlung von Partnerverbünden oder Obligoberechnungen).

Wenn dazu noch einheitliche Zielbilder im Sinne einer Standardarchitektur als Grundlage von Designprozessen genutzt werden, dann sind im Zeitablauf auch Kostenvorteile gegenüber getrennten Ansätzen realisierbar. ■