

Von der Daten- zur Informationsmigration

Stefan Berner
Diso AG
Gümligen, Schweiz

Schlüsselworte

Datenmigration, Informationsmigration, Best Practice

Einleitung

Datenbestände sind etwas vom Wertvollsten in der IT. Dabei geht es weniger um Daten selber, als um die Informationen (Wissen, Know-how) die sie repräsentieren.

Mit jeder Migration in eine neue Umgebung oder Applikation steht man vor dem Problem, was und wie man migrieren soll. Übernehme ich alles, habe ich immer noch ein legacy System, dessen Komplexität mit jeder Übernahme steigt. Übernehme ich zu wenig, verliere ich Know-how.

Informationsmigration setzt die Idee um, Informationen d.h. Wissen aus der alten Umgebung zu extrahieren, von den Altlasten zu trennen und in die neue Umgebung überzuführen. Im Vortrag werden Lösungsideen und Vorgehensweisen aufgezeigt, die sich in der Praxis bewährt haben.

Problemstellung

Erinnern Sie sich an Ihren letzten Umzug? An die Kisten im Keller oder auf dem Dachboden. Kisten deren Inhalt man die letzten Jahre nicht benutzt hat, die möglicherweise vom vorletzten Umzug her mit "Keller" oder "Dachboden" angeschrieben sind. Was tun mit diesen Kisten?

In den Datenbeständen der bestehenden Applikationen steckt viel Kapital. Einerseits in Form von Wissen (Kundenstamm, Produktions- oder Verkaufsstatistiken, Dokumentation) andererseits in Form von mission-critical Daten (Rechnungsdaten, Kontostände, Mitgliederbestand etc.). Deren Verlust kann den Betrieb massiv behindern und zu riesigen Kosten führen.

Was tun mit diesem Wissen, *diesen Kisten*?

Beim Wechsel auf eine neue Applikation oder Umgebung müssen diese Datenbestände übernommen d.h. migriert werden.

Migrationen hängen von vielen Kriterien ab:

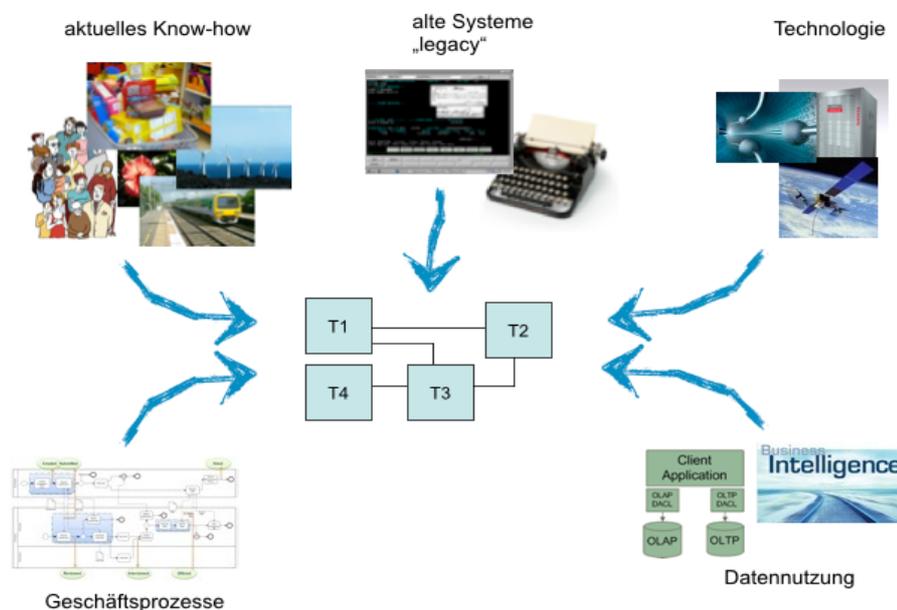
- ▶ Betriebliche Anforderungen an Qualität und Vollständigkeit der Übernahme
- ▶ Verfügbare Zeit und Ressourcen
- ▶ Technische Möglichkeiten und Restriktionen
- ▶ Datenmengen
- ▶ Wartungsfenster, Übergangszeiten
- ▶ Strukturelle, inhaltliche Änderungen

All diese Kriterien spielen für die Qualität der Übernahme eine Rolle. Viele sind technische Herausforderungen, die durch gut organisierte und durchgeführte handwerkliche Arbeit sauber gelöst werden können.

Ein entscheidender Punkt für jede erfolgreiche Migration ist das Wissen, das in Datenbeständen steckt. Dieses Wissen steckt teilweise in den Attributwerten, teilweise in den Strukturen, in denen diese Datenwerte abgelegt sind.

In einem Datenmodell, das über die Jahre gewachsen ist, stecken verschiedene Arten von Semantik (Wissen):

- ▶ Aktuell benötigtes Wissen
- ▶ Obsoletes Wissen
- ▶ Veraltete Methoden der Strukturierung (hierarchische oder andere vor-relationale Datenbanksysteme, Codesysteme etc.)
- ▶ Steuerung von bestehenden Programmen, Abläufen und Geschäftsprozessen
- ▶ Technologieaspekte (Medienbrüche, Restriktionen der Datenbank- und Client-Software, etc.)
- ▶ Frühere Migrationslösungen
- ▶ Flickwerk, Umgehungslösungen durch Erweiterungen



Diese Punkte überlappen sich häufig oder sind ineinander verschachtelt. Sie lassen sich nicht einfach voneinander trennen. Genau diese Komplexität und ihre Nichtbeherrschung bzw. der Respekt vor den unabsehbaren Folgen führen dazu, dass viele den Aufwand scheuen und lieber alles “as-is” übernehmen.

Dieser Aufsatz konzentriert sich auf den Aspekt der Migration der Strukturen und damit des Know-Hows.

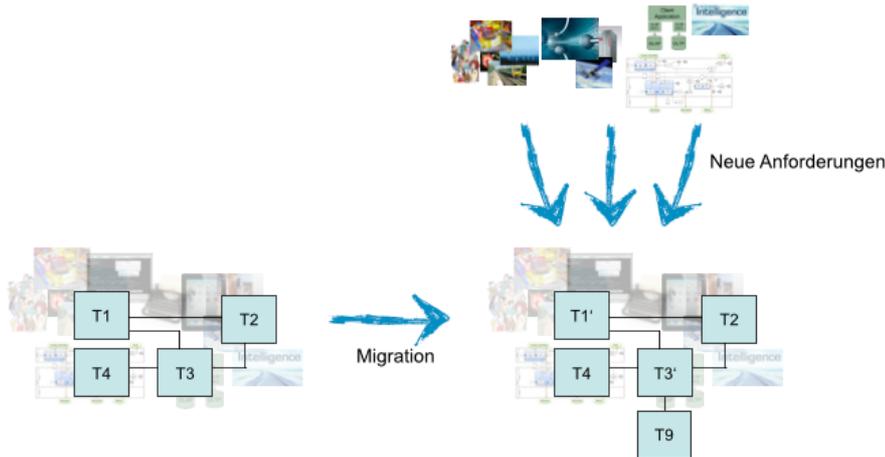
Häufig gewählte Lösungsansätze

Am einfachsten ist es, jede Kiste ungesehen an den neuen Ort mitzunehmen. Möglicherweise mit dem guten Vorsatz sie dann am neuen Ort auszusortieren. Man weiss genau, dass es am Besten wäre, sämtliche Kisten auszumisten und nur die noch brauchbaren Dinge mitzunehmen. Aber man scheut den Aufwand. Die Möbelpacker zu bezahlen und am neuen Ort Platz zu schaffen sind organisatorische, rasch erledigte Aufgaben. Alles anschauen und entscheiden, ob man es jemals noch brauchen wird, ist aufwendiger.

Zu häufig wird eine Datenübernahme als technische Aufgabe verstanden und durchgeführt. Bei einer Migration müssen nebst den Werten auch ihre Zusammenhänge und Strukturen - das Wissen eben - transformiert werden. Dies wird oft erst bemerkt wenn die Datenübernahme zu lange dauert und die Qualität (das Resultat) nicht zufriedenstellend ist.

Eigenentwicklung

Häufig scheut man den Aufwand, die bestehende, historisch gewachsene Datenstruktur zu analysieren und in eine neue Form zu bringen. Zu viel Komplexität, zu viel verstecktes (Struktur-)Wissen, zu wenig Zeit verleiten dazu die alten Datenstrukturen möglichst unverändert in die neue Applikation zu übernehmen. Man ergänzt sie um die neuen Anforderungen und baut die Benutzer-Module neu.



Kritik: Es werden alle Altlasten (Flickwerk, Fehler, Überflüssiges, Veraltetes) übernommen und die neue Lösung wird strukturell aussehen wie die alte. Der vermeintliche Vorteil einer raschen Übernahme ohne Umlernen wird teuer bezahlt, wenn man versucht neue, der heutigen Realität entsprechende Module zu schreiben. Sämtliche Unzulänglichkeiten der alten Strukturen müssen mit hohem Aufwand in der Benutzerschnittstelle ausgebügelt werden.

Zudem hat sich das Verständnis der eigenen Informationswelt nicht verbessert: *Alter Wein in neuen Schläuchen*. Die neue Lösung ist von Anfang an als *legacy system* aufgebaut.

Fremdsoftware

Beim Kauf einer Fremdsoftware bleibt Ihnen nichts anderes übrig, als die Daten in die neue, eingekaufte Struktur zu überführen. Bei zu grossen strukturellen Differenzen zwischen dem alten und dem neuen Modell kann dies zu einer fast unlösbaren Aufgabe werden. Nicht selten macht man aus Kostengründen einen Schnitt und beginnt in der neuen Welt auf der grünen Wiese. D.h. viel Wissen und Kapital gehen verloren. Oft wird dann ein altes System noch jahrelang im Lesemodus betrieben um den Verlust etwas abzufedern.

Sollte der Anbieter von sog. Standardsoftware vorschlagen, in seinem Produkt die alte Struktur des Käufers nachzubilden, ergeben sich ähnliche Probleme wie im Fall der Eigenentwicklung. Ergänzt um die Problematik, dass die alte Struktur in einer vorgegebenen neuen Umgebung simuliert werden soll.

Kritik: Viel Wissen und Kapital gehen verloren. Man beginnt in einer neuen Welt, die ganze Firma muss umlernen und ggf. ihre Prozesse ändern. Nach einer aufwendigen Schulungsphase wird in einer neuen Welt gearbeitet, die nicht wirklich verstanden wird.

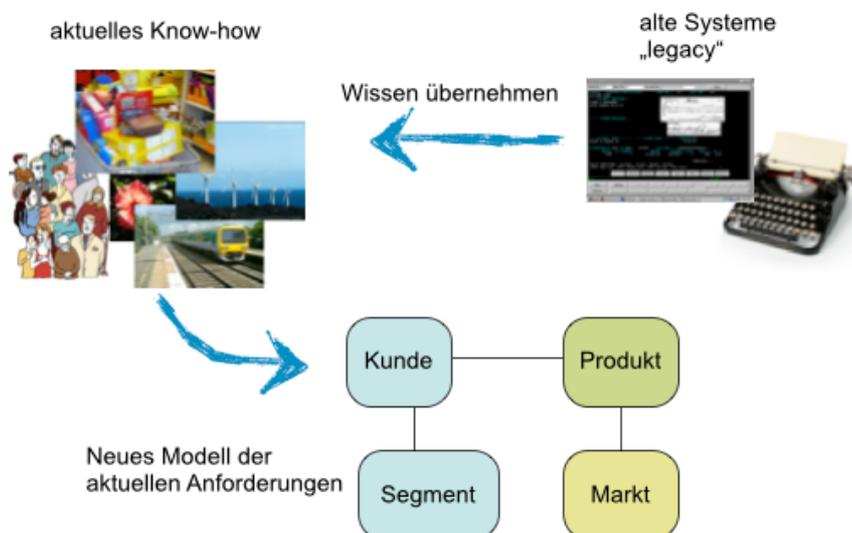
Erfolg versprechender Lösungsansatz

Das einzige, was wirklich in die neue Lösung übernommen werden soll, ist das heute noch gültige und benötigte Wissen. Genau betrachtet möchte man gar nicht die Daten migrieren. Diese sind eine Form der früheren, technischen Lösung. Wirklich wertvoll sind die Informationen die in diesen Datenbeständen stecken. Diese Informationen repräsentieren das Wissen über die eigene Welt. Diese Informationen möchte man in eine neue, adäquate, moderne Lösung überführen. **Der wirkliche Gewinn liegt in der Informationsmigration.**

Der einzige längerfristig sinnvolle und erfolgreiche Weg besteht darin, die angestrebten Informationsbedürfnisse der neuen Lösung zu modellieren, die alte Lösung zu analysieren, in ihre semantischen Aspekte aufzutrennen und danach eine Migration der alten Daten in die neue Struktur vorzunehmen.

Je nachdem, wie gut strukturiert die alte Lösung ist, kann der Aufwand gross werden. Dem grossen Aufwand steht ein grosser Gewinn gegenüber. Die möglichen Vorteile die man sich erarbeitet entfalten ihre Wirkung häufig erst längerfristig. Kurzfristige Gewinne sind kaum zu erzielen. Sie sind es aber wert, genau betrachtet und langfristig bewertet zu werden:

- ▶ Man hat reinen Tisch, d.h. eine saubere, konsistente Informationsbasis für die Zukunft.
- ▶ Es fallen Altlasten weg.
- ▶ Erwerb von technologieunabhängigem Know-How über die eigene (Informations-)Welt.
- ▶ Verbesserung der Datenqualität.
- ▶ Möglichkeit, seine Prozesse und Module auf Basis einer aktuellen, entschlackten Struktur zu modernisieren.
- ▶ Leute die mit der neuen Software arbeiten, werden ihre alten Wissensstrukturen in der neuen Umgebung wieder erkennen.



Praktische Möglichkeiten

Für jeden erfolgreichen Wechsel einer bestehenden Lösung ist es zwingend, die Anforderungen der neuen Lösung aus fachlicher Sicht, auf Ebene der Informationen (d.h. als konzeptionelles Datenmodell, ohne technische Aspekte) zu modellieren. Je nach dem, wie gut das in älteren Lösungen gemacht wurde, ist dies ein kleiner Schritt oder aber man muss die Unterlassungen der Vergangenheit ausbügeln und mehr investieren.

Mit einer Beschreibung dessen, was wirklich gebraucht wird, kann der Migrationsprozess in Angriff genommen werden.

Eigenentwicklung

Es empfiehlt sich in fast allen Fällen, die neue Lösung gemäss dem neuen Modell zu bauen. Durch die Analyse der bestehenden Daten(-strukturen) ergeben sich Ähnlichkeiten und Differenzen. Daraus kann der Migrationspfad entworfen werden.

In mehreren Projekten hat es sich bewährt, die Datenmigration parallel mit der Entwicklung der neuen Software als eigenes Teilprojekt anzugehen. Sobald die erste stabile Version des neuen Informationsmodells vorliegt, sollte mit der Datenmigration begonnen werden. Die frühzeitige Beschäftigung mit

der Migration der Daten unterstützt die Transformation der Informationen, bietet Hinweise auf Schwachstellen in der neuen Lösung und erhöht damit die Qualität des neuen Systems. Man lernt seine Informationswelt besser kennen durch:

- ▶ frühzeitige Überlegungen, welche Daten (nicht) übernommen werden.
- ▶ Entscheide, wie Altlasten behandelt werden.
- ▶ die Bereinigung von inkonsistenten Beständen.
- ▶ die Entflechtung unterschiedlicher Semantik-Aspekte in den Daten.

Man kann die Modelle und Strukturen der gekauften Software mit den fachlichen Anforderungen vergleichen. Bietet sich die Gelegenheit, die Analyse des eigenen Informationsbedürfnisses vor der Evaluation der Fremdsoftware durchzuführen, können diese Anforderungen in die Evaluation einbezogen werden. Dies eröffnet ganz neue Möglichkeiten in der Kommunikation mit Anbietern. Ihre Versprechen können verifiziert werden.

Die Differenzen zwischen den eigenen Bedürfnissen und der Struktur der Lösung bestimmen Art und Umfang des Migrationspfades. Selbst wenn die Migration nicht 100% möglich ist, Differenzen und ihre Auswirkungen werden besser erkannt. Entscheide zum Umgang mit den Differenzen sind besser fundiert. Kenntnis der Differenzen hilft auch, Leute welche mit der Software arbeiten werden gezielter zu schulen.

Schlusswort

Es bleibt anzumerken, dass viele technische und organisatorische Probleme nur entstehen bzw. entstanden sind, weil inhaltliche Aspekte inadäquat behandelt werden oder wurden. Viele IT-Lösungen sind komplexer als es die Aufgabenstellung verlangen würde. Diese Komplexität (besser Kompliziertheit) entsteht, wenn eine Lösung auf falschen Strukturen aufbaut. Auf einem schlechten Fundament kann kein gutes Haus entstehen.

Der Aufwand welcher in eine fachlich-inhaltlich saubere Modellierung gesteckt wird, erspart längerfristig einen viel höheren Migrations-, Reparatur-, Lern- und Betriebsaufwand.

Sind die Kisten ausgemistet, fühlt man sich meist erleichtert. Man hat die Übersicht wieder. Und man nimmt sich ganz fest vor, es nie wieder so weit kommen zu lassen...

Kontaktadresse:

Stefan Berner
Diso AG
Morgenstrasse 1
CH-3073 Gümligen

Telefon: +41 (0) 31 958 90 90
E-Mail: sberner@diso.ch
Internet: www.diso.ch

