# Horizontale Dimension der DG

*Data Governance*

*Thilo Riegel, München*

## Zusammenfassung

Data-Governance-Serie, Teil 4: Im Teil 3 haben wir die sog. vertikale Dimension der Data Governance als Teil eines gesamthaften Ansatzes zur Lösung der klassischen Probleme beschrieben. Nun gehen wir auf die sog. horizontale Dimension ein.

## Grundidee

Zunächst einmal: Was meinen wir mit der horizontalen Dimension? Es wird klar, wenn wir uns noch einmal kurz vor Augen führen, in welchem Zusammenhang Data Governance typischerweise nötig ist und am meisten gebraucht wird: Im Umfeld von BI und DWH.

### Anker: DWH-Architektur

Nun gibt es, auf abstrahierter Ebene, einen typischen System- und Prozess-Aufbau, den man in diesem Umfeld immer wieder vorfindet und der häufig auch grafisch so dargestellt wird: Ein DWH, bei dem die Daten verschiedene Stages von links nach rechts durchlaufen, und ganz rechts kommen die Reports heraus. Stark vereinfacht und sehr high-Level ausgedrückt, hat ein DWH also eine horizontale Datenrichtung von links nach rechts: Links Daten rein, Daten durch die Stages durch, rechts Daten raus (Abbildung 1).

### Unterstützung der Kernaufgaben des Data Governors

Damit ist schon einmal grob umrissen, worauf wir mit der horizontalen Dimension hinaus wollen. Nun ist es für einen Data Governor natürlich nicht damit getan, zu wissen, dass „links Daten rein, rechts Daten raus“ gehen. Bei dieser Erkenntnis fängt sein Job überhaupt erst an.

**These**

Ein Data Governor muss

- seinen Abnehmern (dem Management) gegenüber nachweisen können, welche Reports bestellt wurden und aus welchen Daten in welchen Quellsystemen sie letztlich hervorgehen,
- die Arbeit der verschiedenen Stakeholder unterstützen / DG-technisch koordinieren,
- jederzeit auskunftsfähig darüber sein, welche Rechenprozesse mit welchen Ausgangsdaten durchlaufen wurden, um die Ergebnisdaten zu produzieren,
- diese Rechenprozesse verstehen bzw. fachlich fundiert begründen können. Ggf. muss er sie auch teilweise anpassen können, ohne dass dafür gleich ein halbes Jahr Forschungsarbeit ins Land geht,
- die Prozesse, die zur Fertigstellung der Reports nötig sind, im Griff haben und optimieren,

- jederzeit in der Lage sein, neben der Basisarbeit der Erstellung regelmäßiger Reports auch Individual-Reports auf Ad-Hoc-Anfrage hin zu produzieren.

Es sollte sich für viele Leser von selbst verstehen, aber ich erwähne es trotzdem an dieser Stelle: das alles ist nicht nur als Teil einer „ordentlichen“ und professionellen Arbeits-Organisation zu sehen, mit der sich ein Data Governor ein „gut gemacht, weiter so!“ vom Management verdienen kann, sondern es ist absolut unverzichtbar – die Gründe dafür sind:

- Weil sein Team sonst – allein schon aufgrund der schieren Menge der zu verwaltenden Datenmodelle, Reports, Felder, Überleitungen, KPIs usw. – seine Arbeit nicht machen kann. **These**
- Weil sein Team sonst die Dokumentation und die Metadaten nicht aktuell halten kann **These**

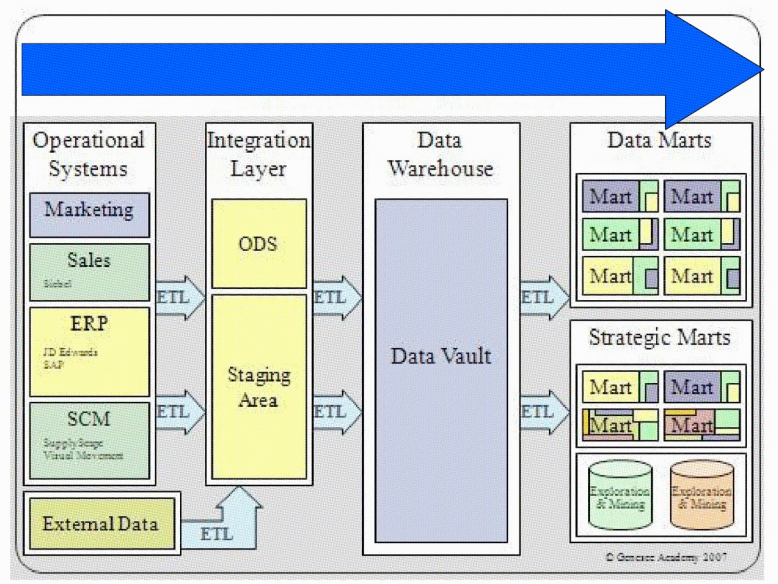


*Abbildung 1: Grundidee der horizontalen Dimension: Datenfluss durch BI/DWH-Architektur*

und damit immer wieder „von vorne“ anfangen muss – mittel- bis langfristige *unweigerlich* alles sehr teuer wird.

- Für das Umfeld Banken (in dem der Autor lange gearbeitet hat) kommt noch verschärfend hinzu: Es ist regulatorisch gefordert.

Das o.g. sind also keine „schöner wohnen“-Anforderungen, sondern pure Notwendigkeit.

## Data Lineage: Unverzichtbar

Was alles eine DG-Organisation ausmacht und was alles nötig ist, um o.g. sicher zu stellen, darüber sind ganze Bücher geschrieben worden. Hier möchten wir auf einen besonderen Aspekt hinaus: Die sog. Data Lineage.[1] Damit ist gemeint, dass wir zu jedem Daten-Element in den Quellsystemen sagen können, in welche Reports es letztlich eingeht (und welche Verarbeitungs- und Rechenprozesse da-

1 Manchmal wird auch der Begriff „Data Chain“ verwendet.

zwischen liegen, und ob es direkt oder indirekt eingeht), und ebenso umgekehrt: dass wir zu jedem Daten-Element in den Reports sagen können, aus welchen Daten-Elementen in den Quellsystemen die Information letztlich hervorgeht, und welche Verarbeitungs- und Rechenschritte dazwischen liegen. Und weil das System für den Data Governor keine Black Box sein darf, muss er das natürlich auch nach jedem wesentlichen Verarbeitungsschritt (also auf jedem DWH-Stage) ebenso sagen können.

Und nun kommt die ganz harte Anforderung:

Es genügt nicht, über all das *ungefähr* und nach jeweils einer Woche „Forschungsarbeit“ Auskunft geben zu können, sondern das DG-Team muss diese Informationen **These**

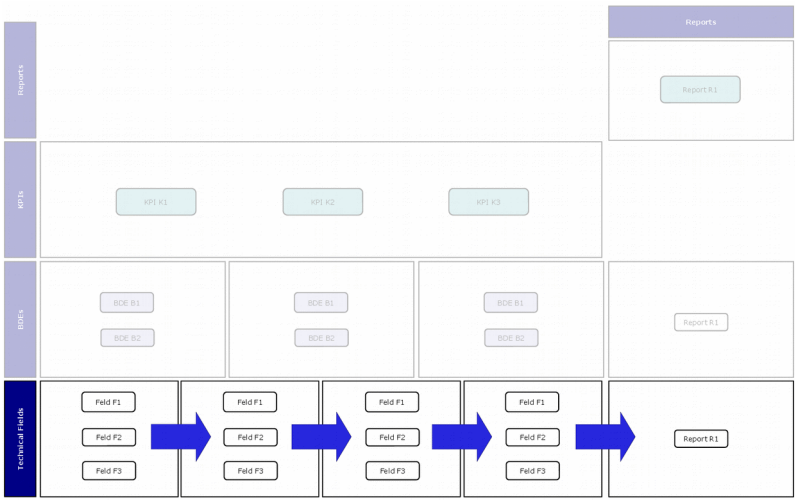


*Abbildung 2: Data Lineage (grob) auf technischer Ebene*

- *jederzeit,*
- *tages- oder zumindest wochenaktuell,*
- *voll-automatisiert auf Knopfdruck*
- *und genau (d.h. auf Einzelfeld-Ebene)*

liefern können.

Halten wir hier einmal kurz inne und lassen uns das auf der Zunge zergehen: Das ist eine ziemlich weit gehende Forderung, die in der Praxis alles andere als einfach zu erfüllen ist. Es bedeutet, dass wir für zehntausende von Feldern jeweils über Verweise systemgestützt und in Echtzeit über die Stages vor und zurück navigieren können, und dass hinter jedem Verweis Verarbeitungs- und Rechenvorschriften nachgesehen werden können.

## Detailgrad: Maximal

Es genügt also nicht, Bildlich gesprochen, das zu dokumentieren, was man in Abbildung 2 sieht, sondern man braucht das, was man in Abbildung 3 sieht.

Während Abbildung 2 noch Trivialitäten dokumentiert, geht es in Abbildung 3 schon ans Eingemachte. Leider sieht man in der Praxis als sog. „Dokumentation“ nur etwas, was im Detailgrad bestenfalls irgendwo zwischen beiden Abbildungen liegt – und das noch nicht einmal aktuell.[2]

2 Womit wir übrigens nicht sagen möchten, dass Aggregierte Sichten, z.B. die Darstellung auf Ebene der Tabellen anstatt auf Ebene der Felder, nicht ebenfalls ihre Berechtigung hätten. Nur genügt das eben nicht.

## Soll vs. Ist: Unterstützung der Planungs- und Umsetzungsprozesse

Und als ob das nicht genug wäre, kommt noch verschärfend hinzu:

**These**

Nicht nur der *Ist-Stand* muss auf diese Art dokumentiert und auswertbar sein (das liefern einige Systeme vom Markt halbwegs gut), sondern auch der *Soll-Stand*, und dieser auch auf allen Abstraktions-Ebenen der vertikalen Dimension.[3]

Warum? Weil die Welt nie still steht und der Umsetzungsstand nur selten dem Planstand entspricht, und weil man ohne ein klares, detailliertes Zielbild sowie ohne die genauen Unterschiede zwischen Soll und Ist nicht wirklich gut planen und tracken kann; insofern ist der Data Governor wie der Chef-Controller einer Firma – nur eben nicht für finanzielle Daten, sondern für BI/DWH-Daten.

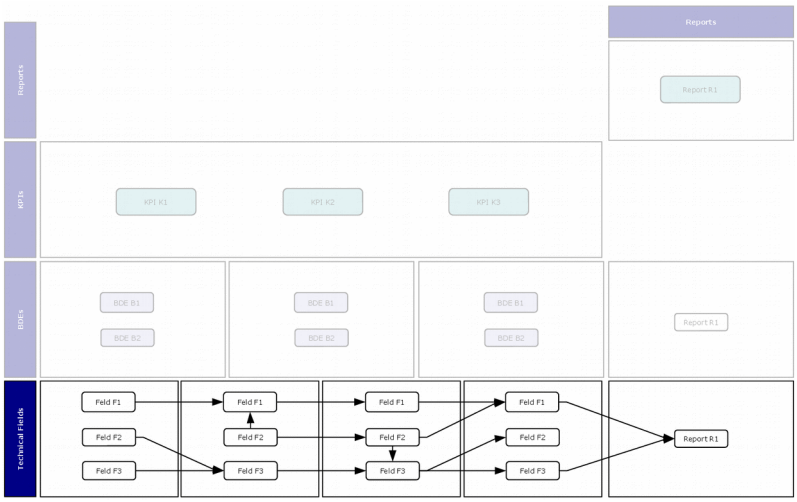


*Abbildung 3: Data Lineage (fein) auf technischer Ebene*

Ein Beispiel aus der Praxis: Wir beide waren einmal beide in einem großen BI-Projekt für das Metadaten-Management zuständig. Kernanforderungen waren u.a.:

- Vollständiges und konsistentes Design der Datenüberleitungen über die Stages hinweg (Detaildokumentationstauglich).
- High-Level-Sichten und HL-Tracking.
- Bereits zusammengetragene Datenanforderungen und deren Umsetzung durch Überleitungen tracken (low-level).
- Künftige Datenanforderungen / Überleitungen über mehrere Release-Zyklen tracken (low-level; hierbei wichtig: klare Trennung von Soll- und Ist-Stand der jew. Release-Stände).
- Nicht nur das Tracking im Umsetzungsprojekt zu unterstützen, sondern auch die künftige Linienarbeit des Data Governors (mit seinem völlig neu auszurichtenden DG-Bereich).

Allein das High-Level-Tracking des Planungs- und Umsetzungsstandes auf Projektleiter-Ebene war dabei – aufgrund der schieren Größe und Komplexität des Vorhabens – nicht ganz einfach (wegen des Riesen-Gaps zwischen den fachlichen Anforderungen des Gesamt-Projekts und bis dato verwalteten Low-Level-Anforderungen auf technischer Ebene bei den umsetzenden Teams). Aber die Sicherstel-

3 Vgl. dazu den letzten Teil dieser Artikel-Serie.

lung der Korrektheit, Vollständigkeit und Konsistenz – und v.a. der konsistenten und tagesaktuellen Sicht über alles – war eine Herausforderung. Man bekommt das mit konventionellen Mitteln für ein paar Hundert Felder noch irgendwie hin (wenn auch umständlich), aber für ein paar 10.000 Felder (mit ein paar weiteren 100.000 in der Pipeline!) sieht das einfach vollkommen anders aus; der Versuch, so etwas nicht über ein integriertes Metadaten-Repository zu machen, wäre zum Scheitern verurteilt gewesen.

Zurück zum Grundgedanken: Diese Anforderungen liest man so etwas nicht / nur selten / nicht deutlich genug in den Büchern und Artikeln zum Thema. In der Praxis sind sie nach unserer Ansicht

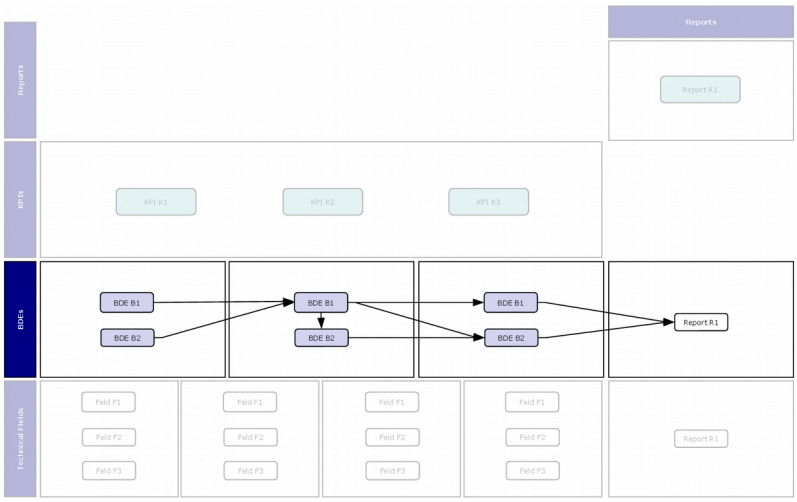


*Abbildung 4: Data Lineage auf BDE-Ebene*

sowieso selten bis gar nicht erfüllt.

## Kommunikation mit allen Stakeholdern

Nun könnte man einwenden, die vollkommen detaillierte Darstellung auf Feldebene sei ja für die Planung und das Tracking in Wirklichkeit gar nicht nötig bzw. sogar hinderlich – ein gröberer Überblick sei nötig, denn schließlich sei der DG ja schließlich selbst in einer Management-Position und nicht Programmierer.

Das stimmt, und es stimmt nicht. Natürlich muss der DG nicht höchstpersönlich jedes einzelne DB-Feld und jede ETL-Strecke beim Vornamen kennen, wie man so sagt. Er muss den Überblick haben, das stimmt schon. Aber seine Leute müssen es können. Und er muss dafür sorgen, dass sie dafür die richtigen Werkzeuge und die richtige Arbeitsumgebung haben. Außerdem hat der DG eine koordinierende Funktion; viele Stakeholder im Hause (sowohl innerhalb als auch außerhalb seiner DG-Organisation) benötigen diese Informationen, und letztlich müssen *alle* Sichten auf *allen* Detail-Ebenen integriert und redundanzfrei unterstützt werden. Die DG-Organisation muss z.B. mit folgenden Stakeholdern professionell und informiert reden können:

- Mit dem externen Spezialisten-Programmierer, der für drei Wochen reinkommt und eine Spezialstrecke implementiert
- Mit den Spezialisten von den Fachabteilungen, die über Jahre hinweg auf leicht abstrahierter Ebene (vielleicht IT-affin, aber ohne Detailkenntnisse des technischen Modells) die Berechnungen modellieren

- Mit dem Top-Management, das natürlich nicht sagt: „gebt mir Feld x aus Tabelle y“, sondern „ich brauche monatlich einen Report, der diese fünf KPIs zusammenfasst – Ist versus Plan und Forecast bis zum Jahresende.“

## …und das Gleiche in den anderen Ebenen

Erinnert Sie die Aufzählung von oben an etwas – Programmierer, Fachspezialist, Top-Management? Richtig, da kommt die vertikale Dimension hinein, die wir schon im letzten Teil behandelt hatten. Wir haben also auf jeder Ebene aus der vertikalen Dimension zusätzlich die horizontalen Infor-

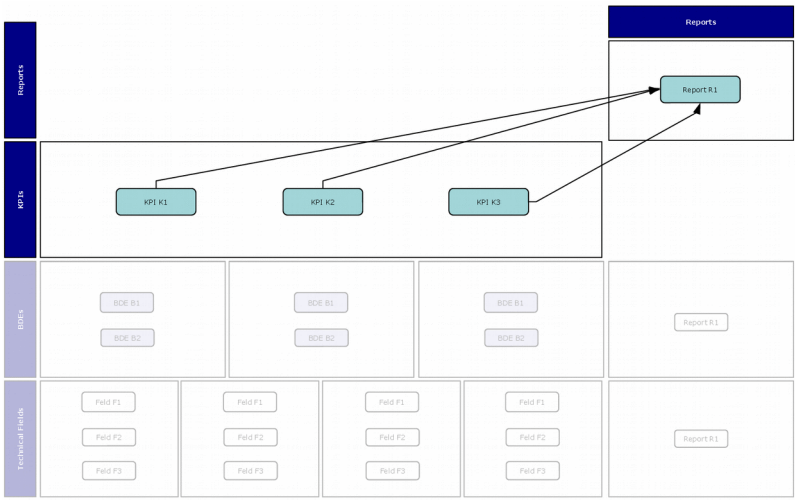


*Abbildung 5: Data Lineage auf BDO/KPI/Report-Ebene*

mationen (Abbildung 4).

Beachten Sie dabei bitte auch, dass auf der Ebene der BDEs in der Grafik nur drei Stages abgebildet sind (ohne Reports) anstatt vier wie auf der Ebene der technischen Felder.[4] Das muss nicht immer so sein, kommt aber durchaus mal vor; so kommt es in der Praxis immer wieder vor, dass aus technischen oder organisatorischen Gründen ein zusätzlicher Stage eingezogen wird, der fachlich gesehen nur eine redundante Kopie des vorhergehenden ist. Das ist auch vollkommen in Ordnung im Sinne der fachlichen Abstraktion – die fachlichen Spezialisten müssen und sollen nichts über die technichen und organisatorischen Details wissen, die keine Relevanz für ihre Arbeit haben. Dennoch denken jedoch auch Fachspezialisten in Stages (als Verarbeitungs-Schritten), und dies sollten sie in einem gewissen, vorgegeben n Rahmen tun. Wir werden im nächsten Teil noch einmal auf diesen Aspekt zurück kommen.

## Höchste Ebene: Wie bei der vertikalen Dimension

Werfen wir nun noch abschließend keinen kurzen Blick auf die höchste Stufe in der horizontalen Dimension (Abbildung 5).

Hier fällt eine leichte Asymmetrie auf: erstens schon rein grafisch, weil der Report auf einer anderen Ebene liegt als die KPIs, und zweitens, weil dies eigentlich dasselbe Teilbild ist wie das höchste aus dem letzten Teil, in dem wir die vertikale Dimension vorgestellt haben: Auch da sahen wir nur noch die Zuordnung der KPIs zu den Reports – *vertikal* dargestellt.

4 Die Anzahl der Stages an sich ist hier übrigens natürlich nicht wörtlich zu verstehen, sondern rein illustrativ.

Was haben wir denn jetzt da? Ist das nun eine vertikale oder eine horizontale Beziehung? Da stimmt doch was nicht im Schema? Auf den ersten Blick ja, aber das ist nur scheinbar ein Widerspruch. Wir werden im nächsten Teil noch etwas genauer auf diesen Punkt zurückkommen.

An dieser Stelle, nach den ganzen abstrakten Einführungen mit Datentypen und Dimensionen, müssen wir uns natürlich fragen, wozu das gut ist. Für sich allein genommen wirkt das alles noch noch ein wenig wie *l'art pour l'art.* Soweit ja, zugegeben. Aber zusammen genommen, richtig modelliert und v.a. befüllt, entfaltet das alles eine ungeheure Kraft für den DG! Mehr dazu im nächsten Teil.

## Falscher Pragmatismus

Häufiger haben wir erlebt, dass solche Anforderungen als „im Prinzip richtig, jedoch (jetzt erst einmal) unrealistisch“ bewertet werden. Es wird ein „pragmatischer“ Ansatz gefordert. Das Problem mit solchen Einschätzungen ist, dass man natürlich einerseits nichts gegen Pragmatismus haben kann (sofern er denn echt und berechtigt ist) und man dieser Aussage daher nicht allgemein widersprechen kann. Ferner ist es natürlich auch klar, dass man größere Sachen dieser Art nicht „eben schnell mal“ aufbaut, sondern sich über Zwischenziele dort hin iterieren muss. Allerdings warnen wir vor falschem Pragmatismus, d.h. der leider weit verbreiteten Jaja-schon-gut-aber-jetzt-mal-realistisch-Einstellung. Diese ist natürlich nicht generell falsch, aber in diesem Zusammenhang sehr gefährlich, weil langfristig teuer!

Um das zu begründen, möchten wir noch einmal auf das o.g. BI-Projekt verweisen: Verkürzt könnte man sagen: „Entweder man macht es, oder man macht es nicht“. Wenn jetzt bereits, sagen wir, 20.000 Felder und 50 Reports nicht nur einmalig realisiert, sondern langfristig unterstützt und gepflegt werden müssen, sondern man darüber hinaus auch weiß, dass die 5- bis 10-fache Menge aus anderen Unternehmensbereichen in der Pipeline ist,[5] dann hat das Konsequenzen. Wenn der Kunde dann das dazu nötige Metadaten-Management nicht wirklich betreiben will, dann mutet der Verweis auf „Pragmatismus“ oder Budget-Mangel ungefähr so an, als würde man vorschlagen, die dispositiven Daten seien zunächst noch in „kostengünstig“ und „pragmatisch“ in Excel-Sheets und auf Notizzetteln zu speichern, und hinterher, in 3 Jahren, könne man ja immer noch auf eine „zugegebenermaßen ideale, aber leider zu teure“ Datenbank-basierte Lösung gehen. Es ist erstaunlich, bei wie vielen Kunden wir derartigen ernst gemeinten Unsinn schon hören mussten.[6]

# Bildernachweis

- Abbildung 1: http://commons.wikimedia.org/wiki/File:Data_warehouse_overview.JPG (leicht bearbeitet).
- Alle anderen Abbildungen vom Autor.

5 Wohlgemerkt: Nicht optional, sondern konkret und verbindlich geplant!

6 Man verzeihe uns an dieser Stelle die scharfen Worte und den Sarkasmus. Tatsächlich wünschen wir unseren Kunden – ernsthaft und ehrlich – auch künftig alles Gute; aber sie werden sehr wahrscheinlich jetzt mit großen Problemen kämpfen, die sie hätten vermeiden können.