m[ethod] 2 data

Die Wunder des Data Mart – das Dimensionale auf dem Relationalen

13. Tagung der DDVUG, 9. November 2023, Basel

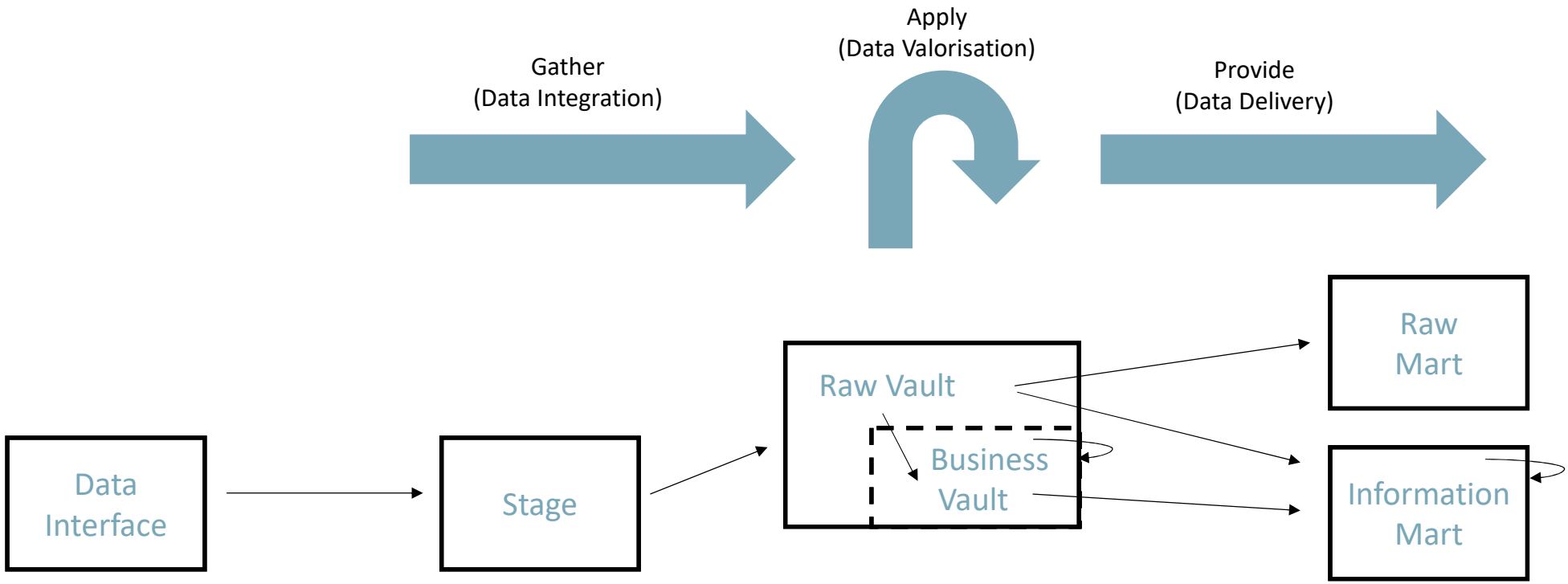
Warum?

I have written extensively on the steps required to administer conformed dimensions and conformed facts. I have never seen a comparable set of specific guidelines for the normalized EDW approach.

Kimball, Ross, "The Kimball Group Reader", Wiley 2015, S. 200

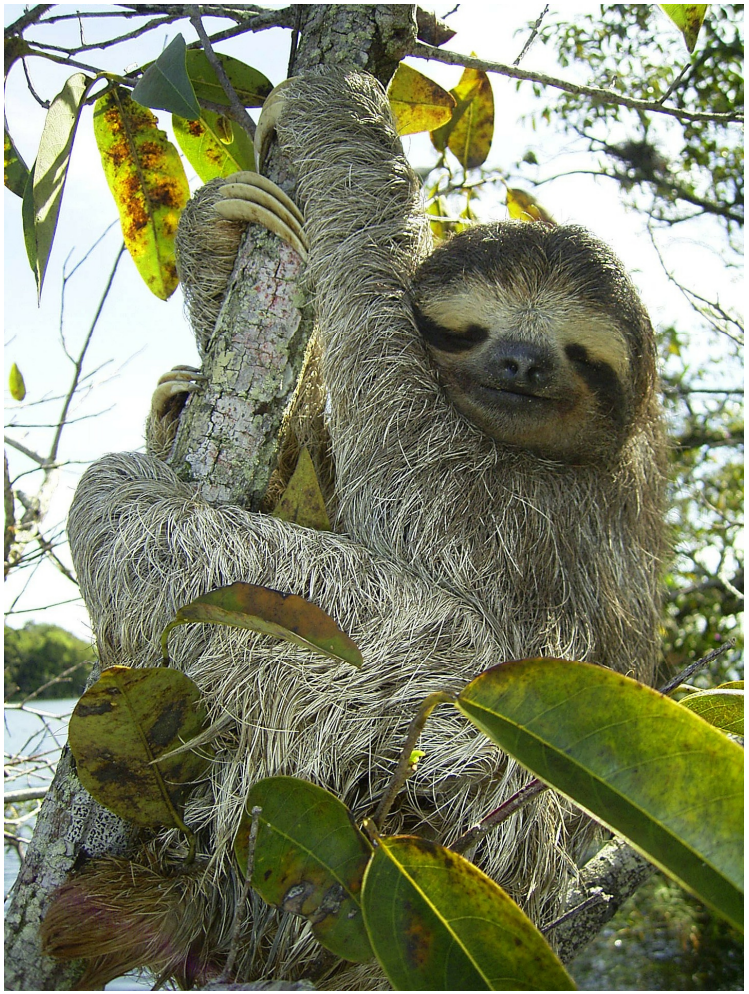
- Es gibt viele Ansätze, um einen Data Mart auf einem Data Warehouse zu beschreiben, jedoch gibt es keinen Standard und die meisten von ihnen sind lückenhaft.
- Es gibt nur wenige Muster, und sie sind nicht durchgängig miteinander verbunden.
- Oft gibt es in den Programmen, die Data Marts erstellen, redundante Logik.
- Die Erfassung der Anforderungen ist hochgradig individuell.
- Erklärungen zum Data Mart sind ohne IT-Hintergrund oft schwer zu verstehen (SCD 5!).
- Das Fachwissen für eine bestimmte Kennzahl ist im Code gespeichert.
- Es gibt keinen Standard für die Dokumentation des Data Mart.
- Metadaten und Muster sind nicht für die Automatisierung geeignet.

Architektur Data Vault



Data Interface	Stage	Raw Vault	Business Vault	Mart

Kennzahlen so einfach wie möglich machen.



- Die Berechnung der Kennzahl möglichst nur an einer Stelle.
- Kennzahlen zwischen den verschiedenen Data Marts gleich halten.
- Entspricht
  ‚Schadensfälle nach Verträgen‘
  aus dem Data Mart ‚Schadensfälle‘
  wirklich
  ‚Verträge mit Schadensfall‘
  aus dem Data Mart ‚Verträge‘?

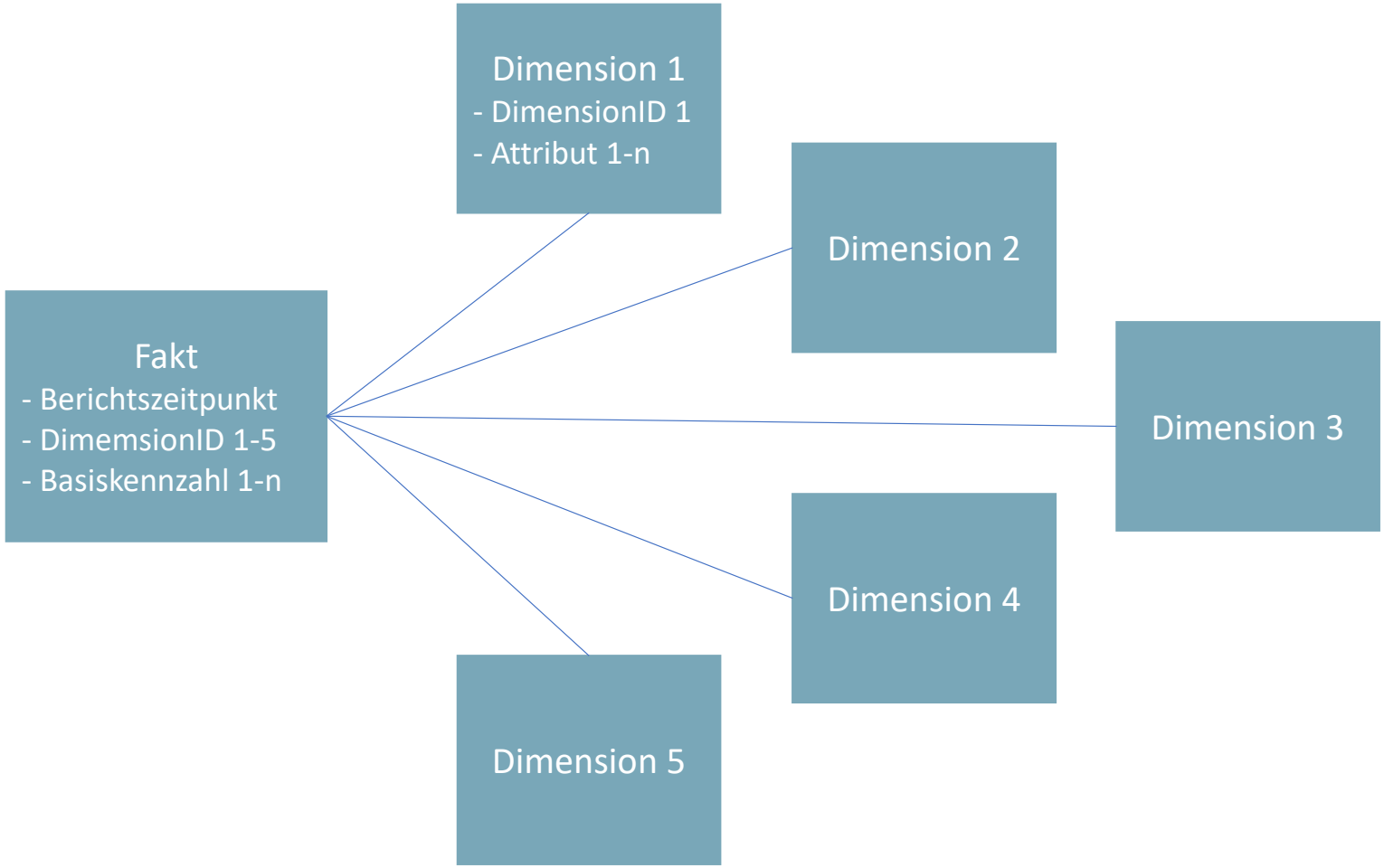
Klassifizierung der Kennzahlen nach ihrer Entstehung

- **Basis-Kennzahlen**
  werden aus dem Core Warehouse (Raw und Business Vault) ausgewählt:
  - Einnahmen
  - Anzahl der Verträge

- **Abgeleitete Kennzahlen**
  werden im Data Mart aus Kennzahlen und Dimensionen berechnet
  - Einnahmen nach Zahlungseingang, Einnahmen nach Auftragseingang
  - Anzahl der Verträge mit Personen unter 25 Jahren, mit Rentnern

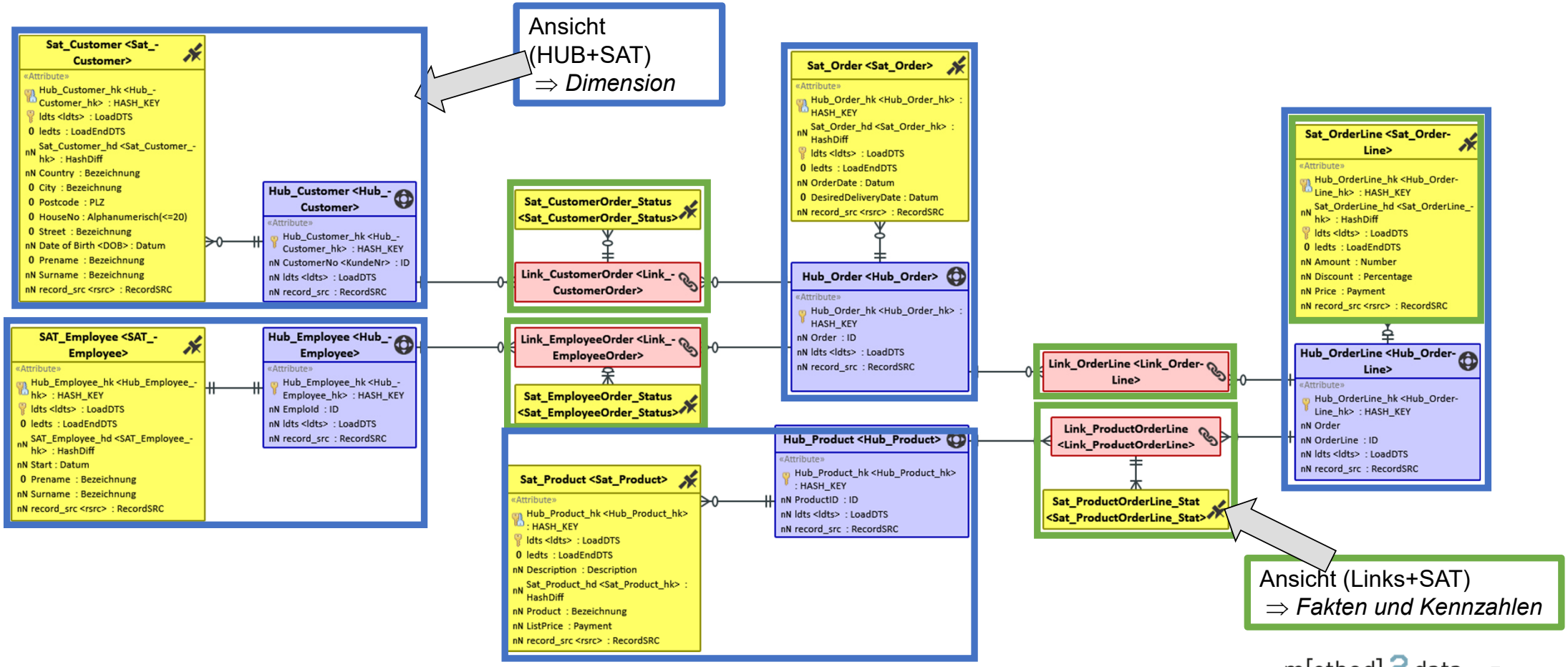
- **Berechnete Kennzahlen**
  können nur im Frontend berechnet werden, da die Berechnung erst nach der Auswahl der Dimensionen erfolgen kann
  - insbesondere Quoten
  - Schadenfreie Verträge in %

Änderungen und Zeit im dimensionalen Modell



- Der Berichtszeitpunkt als fixer Zeitpunkt für die Fakten
- Fakt und Dimension sind fix verbunden
- Attribute in Dimensionen können sich ändern
- Die SCD-Typen 0-7 wurden geschaffen, um Änderungen in den Dimensionen zu behandeln

Data Vault ermöglicht eine schnelle Übergang zu einem Star-Schema.



Drei Arten von Faktentabellen

- **Transaction Grain**
  fester Punkt in Raum und Zeit

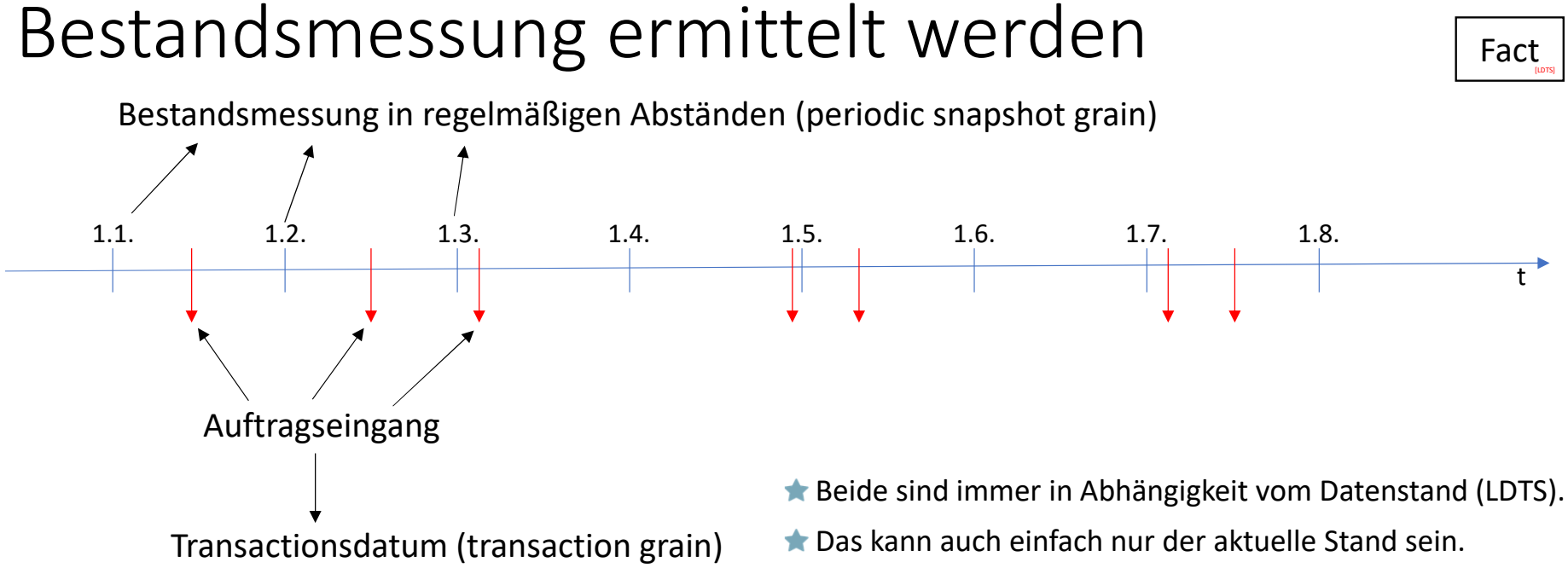
- **Periodic Snapshot Grain**
  regelmäßig wiederkehrende Messung

- **Cumulating Snapshot Grain**
  Entwicklung z.B. eines Prozesses über mehrere Faktentabellen hinweg, berechnet auf der Grundlage anderer Faktentabellen. In der Regel für Prozessketten wie Auftragseingang, Auftragsbestätigung, Lieferung, Rechnungsstellung, Zahlung, ...

- Bei einem historischen Data Warehouse kann das Cumulating Snapshot Grain auch direkt aus dem Core Warehouse erstellt werden, da die Historie die gleichen Ergebnisse wie in anderen Faktentabellen gewährleistet.
- Letzendlich eine Frage des geringsten Aufwands.

Quelle für die 3 Arten von Fakten: Kimball, Ross: The Kimball Group Reader, Wiley, Seite 284

Fakten können per Transactionsdatum oder per Bestandsmessung ermittelt werden



- Beide sind immer in Abhängigkeit vom Datenstand (LDTS).
- Das kann auch einfach nur der aktuelle Stand sein.

Der ***Berichtszeitpunkt*** ist der Zeitpunkt, an dem die Daten laut Datenstand des Data Warehouse gültig sind. Der Berichtszeitpunkt besteht aus einem fachlichen Zeitpunkt – ein Transactionsdatum oder ein periodic snapshot date – und einem technischen Datenstand.

Periodic snapshot date ist ein Zeitpunkt aus einer regelmäßigen Zeitreihe geprüft gegen entweder eine fachliche oder eine technische Historie.

Unterschiede zwischen Transactions- und periodischem Snapshot-Grain

★ Events

1. März: Auftragseingang

25. April: Änderung der Auftragspositionen
(zwei neue, eine wird gestrichen)

9. Juni: Bestellung wird geliefert

★ Transaction Grain (Auftragseingang)

Faktentabelle März: Bestellung

Faktentabelle April: Delta auf Bestellung

Faktentabelle Juni: Lieferung

Spezielle Aggregationsfunktionen für die Zeit:

- Bisheriges Jahr
- Erster Wert
- Letzter Wert

★ Periodic snapshot Grain (Auftragsbuch)

Faktentabelle März: Zufluss
Bestellung

Faktentabelle April: Bestellung

Faktentabelle Mai: Bestellung

Faktentabelle Juni: Abfluss

Nur die Zu- und Abflüsse können über die Zeit aggregiert werden. Normalerweise werden alle Daten als Zeitreihen ohne zeitliche Aggregation dargestellt.

Zeit in Dimensionen oder ‚Wie man Historie in Dimensionen speichert'.

überhaupt keine Veränderung

nur den aktuellen Wert

Historische Speicherung von Werten

Vermeidung der Aktualisierung der Faktentabelle

SCD Type	Dimension Table Action	Impact on Fact Analysis
Type 0	No change to attribute value	Facts associated with attribute's original value
Type 1	Overwrite attribute value	Facts associated with attribute's current value
Type 2	Add new dimension row for profile with new attribute value	Facts associated with attribute value in effect when fact occurred
Type 3	Add new column to preserve attribute's current and prior values	Facts associated with both current and prior attribute alternative values
Type 4	Add mini-dimension table containing rapidly changing attributes	Facts associated with rapidly changing attributes in effect when fact occurred
Type 5	Add type 4 mini-dimension, along with overwritten type 1 mini-dimension key in base dimension	Facts associated with rapidly changing attributes in effect when fact occurred, plus current rapidly changing attribute values
Type 6	Add type 1 overwritten attributes to type 2 dimension row, and overwrite all prior dimension rows	Facts associated with attribute value in effect when fact occurred, plus current values
Type 7	Add type 2 dimension row with new attribute value, plus view limited to current rows and/or attribute values	Facts associated with attribute value in effect when fact occurred, plus current values

Data Vault ist immer Typ 2

Anpassung der Faktentabelle notwendig, da mehr als ein Wert pro ID in der Dimension gültig ist.

Quelle: http://www.kimballgroup.com/2013/02/design-tip-152-slowly-changing-dimension-types-0-4-5-6-7/

Klassifizierung der Dimensionen nach ihrer Geschichte

- **as-is:**
  Alle Fakten beziehen sich auf den aktuellen Dimensionswert

  Basierend auf HubID

- **as-was:**
  Alle Fakten beziehen sich auf den Dimensionswert, der zum Zeitpunkt der Erfassung des Fakts gültig war (Berichtszeitpunkt).

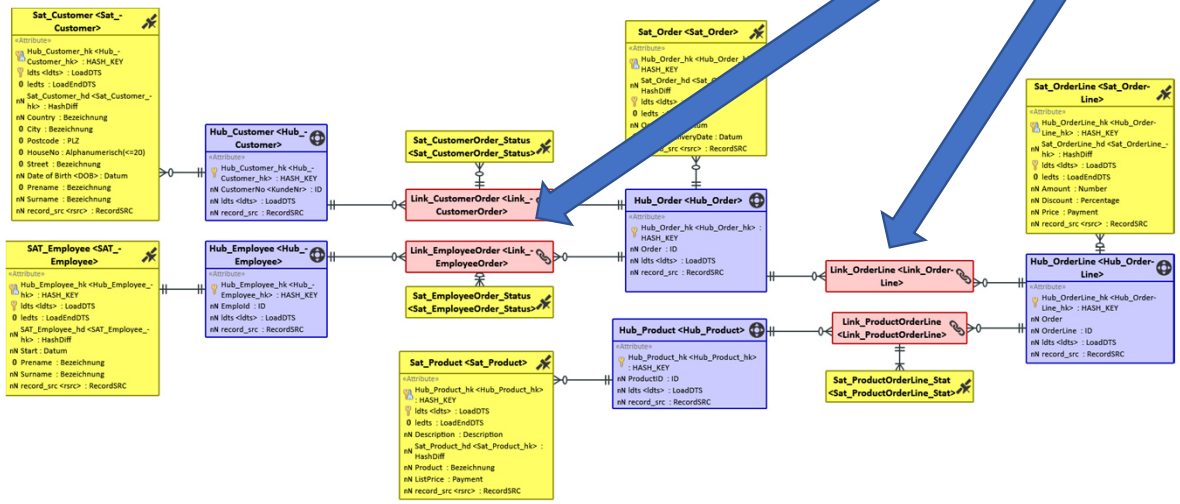
  Basierend auf HubID und LDTS

- **as-of:**
  Alle Fakten beziehen sich auf den Dimensionswert, der zu einem bestimmten, frei wählbaren Zeitpunkt galt

  Basierend auf HubID und LDTS

Änderungen der Dimension?
-> Neuberechnung des LDTS-Teils in der Faktentabelle

Ergänzen der Faktentabelle um die HubIDs der Dimensionen

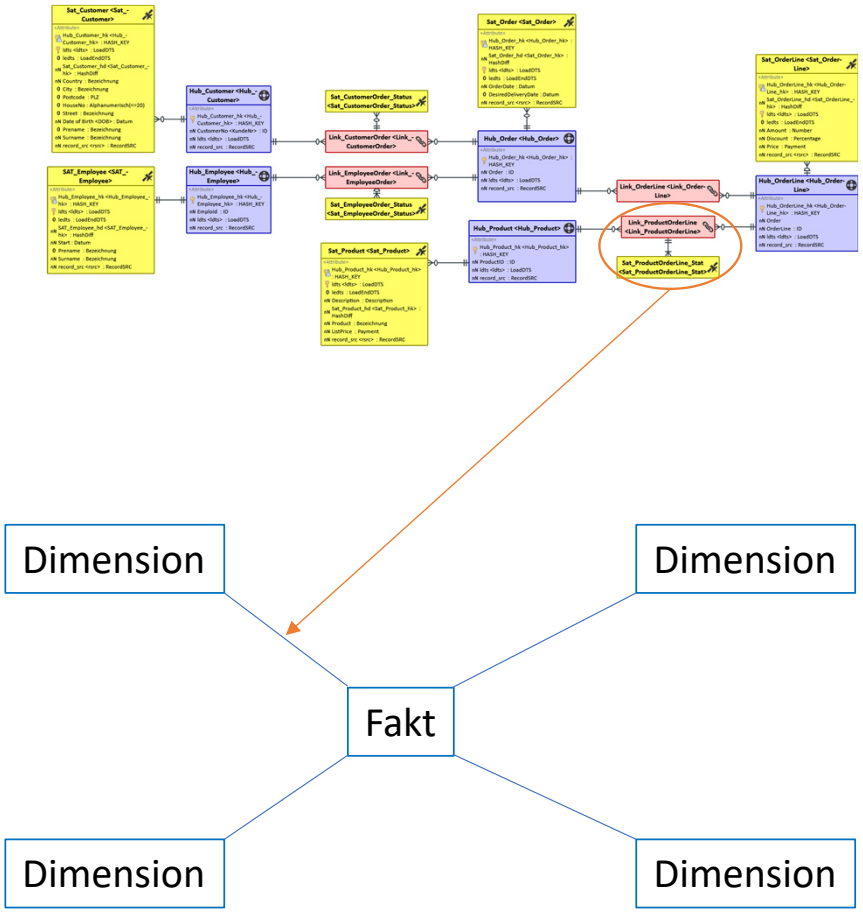


Hinzufügen von HubIDs durch Zugriff auf die Links und deren Status-Satelliten

- normalerweise zum Berichtszeitpunkt
- in seltenen Fällen kann dies auch mit einer anderen Zeit geschehen, um eine Dimension zu einem bestimmten Zeitpunkt zu erhalten, wie:
  - Beschäftigungsstatus bei Beginn des Versicherungsvertrags
  - Familienstand bei der Registrierung auf der Website
  - ...
- Mit oder ohne Auflösung eines möglichen same-as-link zu diesem Hub

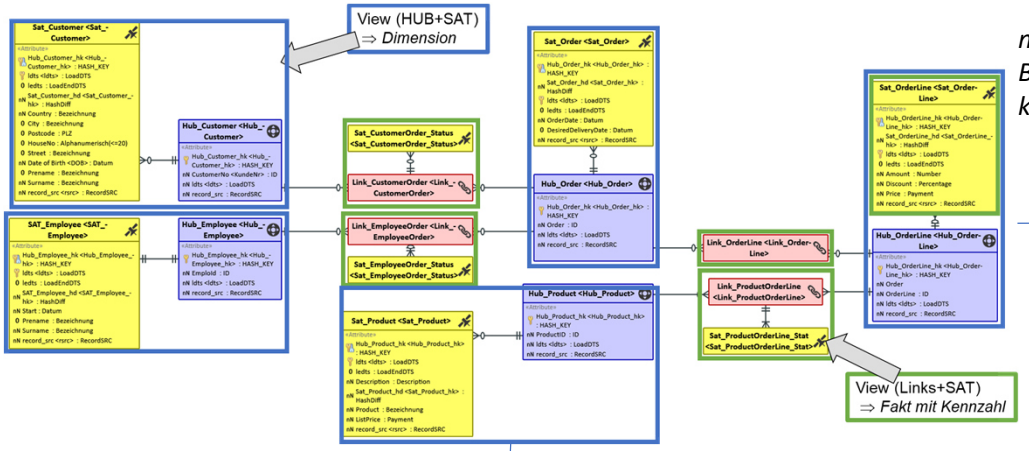
Dies ist unabhängig von der Dimension Zeit (as-*)

Der richtige Dimensionsschlüssel (HubID) für eine Faktentabelle



- Hier geht es um die Frage, welches Objekt für die Faktentabelle ausgewählt werden soll.
- Das ist etwas anderes als die Frage, welcher Datenstand der Dimension zu wählen ist.
- Verwendung einer anderen Zeit als dem Berichtszeitpunkt, mit der durch die Links/Linksatelliten zwischen Faktentabelle und der Dimension navigiert wird.
- Es kann mehr als einen Weg vom Satelliten mit der Basiskennzahl zu dieser HubID geben
- Dies ist die Beziehung zwischen Fakt und Dimension (FDR - Fakt-Dimension-Beziehung)

Core Warehouse liefert alle notwendigen Informationen - in 3 separaten Schritten / Zeiten



nur eine Auswahl von Basis-Kennzahlen, keine Logik im Spiel

1 Berichtszeitpunkt

3 Zeit für die Bestimmung der Dimension

Basis-Faktentabelle

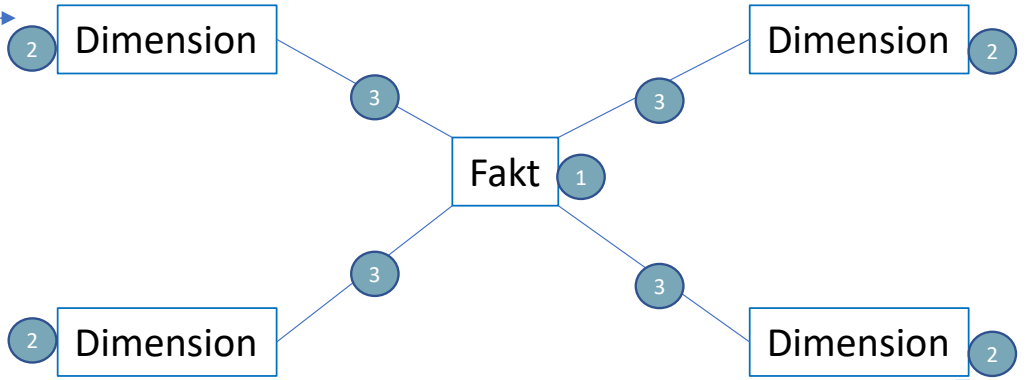
- für eine bestimmte Transaction
oder
- für einen periodischen snapshot

-> auf dem kleinsten Grain

2 Dimensionen

2 Gültigkeit(en) der Dimensionswerte

- aktuelle Werte (as-is)
oder
- Werte zum Berichtszeitpunkt (as-was)
oder
- Werte zu jedem anderen Zeitpunkt (as-of)



Alle Anforderungen für einen Data Mart

- Kennzahlen
  - Basiskennzahlen
  - Abgeleitete Kennzahlen
  - Berechnete Kennzahlen
- Fakten
  - Transaction Grain
  - Periodic snapshot Grain
- Berichtszeitpunkt
- Dimensionen
  - as-is
  - as-was
  - as-of
- Fakt-Dimension-Relation (FDR)
  - Pfad
  - Zeit

Alle Anforderungen für einen Data Mart

- Basiskennzahln
  - Basiskennzahlen
  - Abgeleitete Kennzahlen
  - Berechnete Kennzahlen
- Fakten
  - Transaction Grain
  - Periodic snapshot Grain
- Berichtszeitpunkt
- Dimensionen
  - as-is
  - as-was
  - as-of
- Fakt-Dimension-Relation (FDR)
  - Pfad
  - Zeit

Gute Fragen für die Anforderungsaufnahme

Erläutern Sie Kennzahlen und Dimensionen.

Welche Basiskennzahlen? Dekonstruieren Sie und fragen Sie, ob das richtig ist. Machen Sie Fehler.

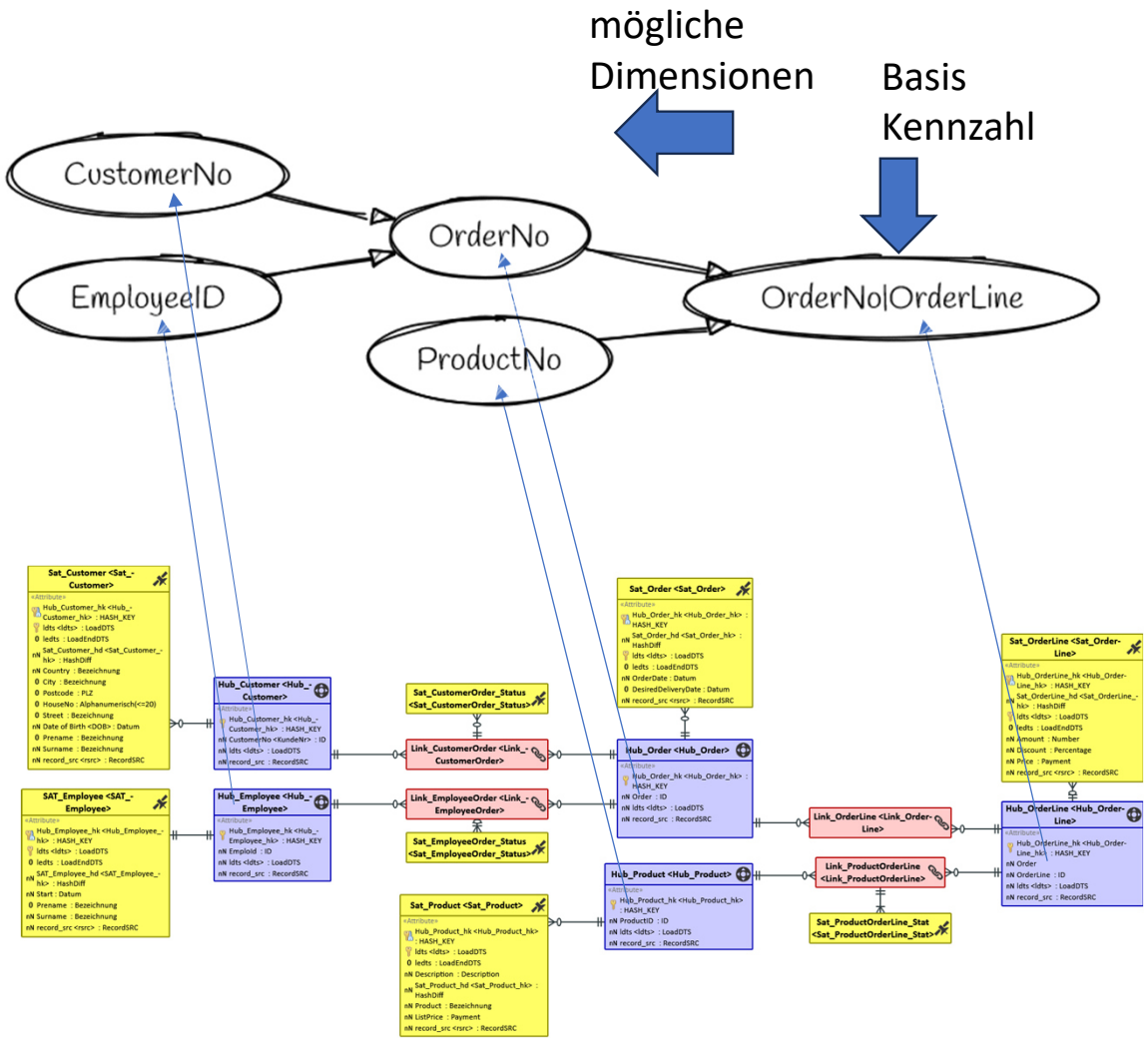
Messung von Transactionen oder Beständen?

Wann? Wie oft?

Was wollen Sie sehen, wenn sich das Attribut "x" der Dimension "z" ändert? Den neuen Wert? Oder der Wert zu diesem Zeitpunkt? Suchen Sie nach organisatorischen Dimensionen (as-of)

Welche Dimensionen? Legen Sie die Basiskennzahlen auf Ihr Geschäftsobjektmodell. Suchen Sie von dort aus nach Dimensionen.

Erstellen eines Geschäftsobjektmodells aus Ihrem Data Vault-Modell



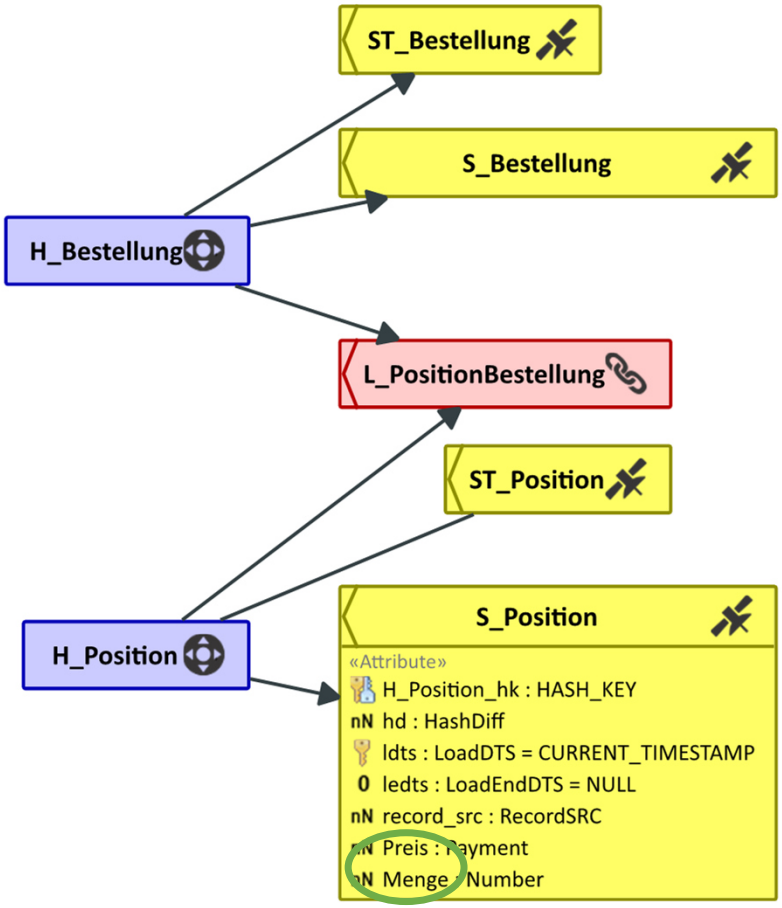
- Für jeden Hub ein Business-Objekt
- Zeichnen Sie für jeden binären Link einen Pfeil zum anderen Geschäftsobjekt (Richtung von 1:n, die Stage verrät die Kardinalität)
- n-äre LInks ist oder Links mit Satelliten (Status-Satelliten zählen hier nicht) -> ein Business-Objekt anlegen
- Zeichnen Sie eine Linie zwischen allen beteiligten Business-Objekten (Hubs) und dem neuen Business-Objekt mit dem Pfeil in Richtung des neuen Business-Objekts
- Organisieren Sie alle Geschäftsobjekte entlang der Pfeile in Leserichtung (von links nach rechts, oben-unten)

Alle Anforderungen für einen Data Mart

- Basiskennzahln
  - Basiskennzahlen ← Nicht zu automatisieren — Business Vault
  - Abgeleitete Kennzahlen
  - Berechnete Kennzahlen ← Nicht zu automatisieren — Außerhalb des Geltungsbereichs, implementiert im Report
- Fakten
  - Transaction Grain
  - Periodic snapshot Grain
- Berichtszeitpunkt ← Nicht zu automatisieren — Parameter
- Dimensionen
  - as-is
  - as-was
  - as-of
- Fakt-Dimension-Relation (FDR)
  - Pfad
  - Zeit

Was kann automatisiert werden?

Die Basiskennzahlen als Teil des Raw Vault

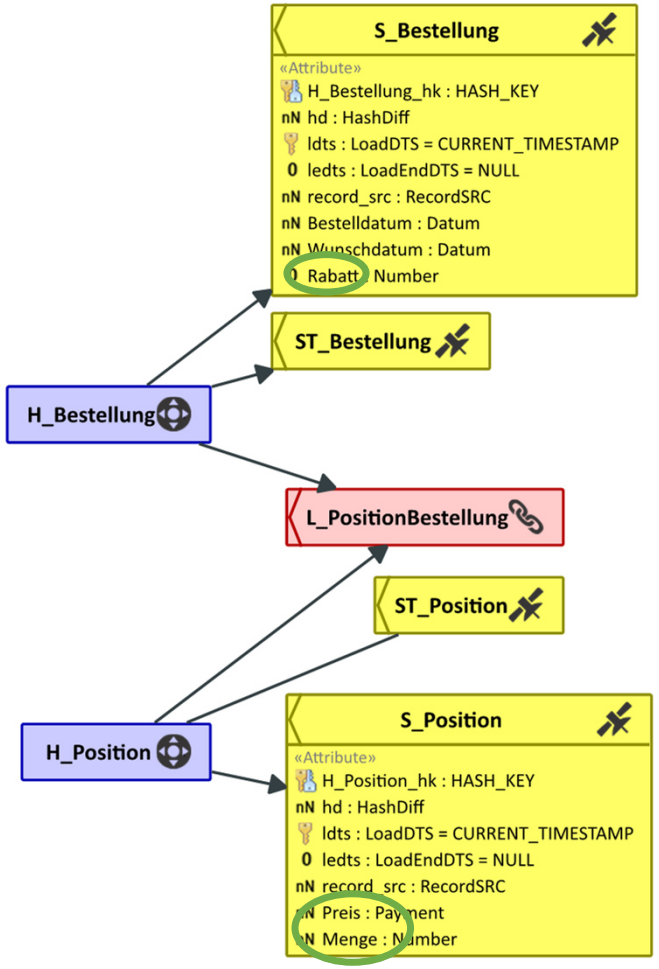


- Direkt vom Satelliten S_Position
- Alles ist bereits an einem Ort

Beginn der Faktentabelle:

- H_Position_HK (ID für kleinstes Korn)
- Preis
- Betrag
- Einnahmen (der Einfachheit halber berechnet)

Basiskennzahl 'Umsatz' benötigt 'Rabatt' von einem anderen Satelliten mit einer anderen Granularität

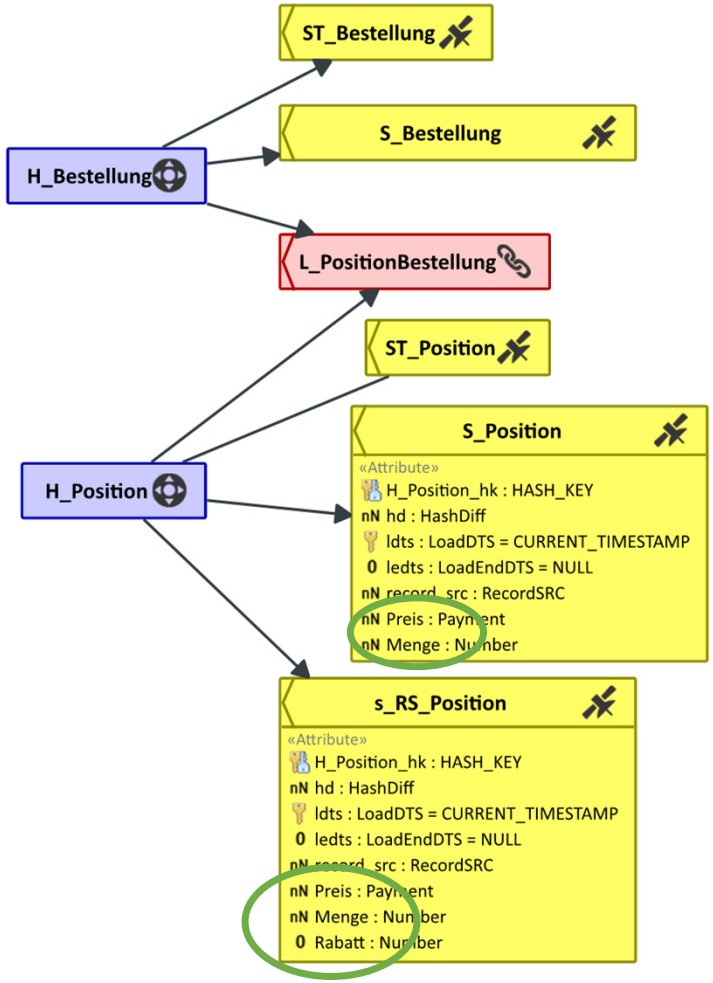


- Der Auftragsrabatt ist für die Berechnung der Einnahmen erforderlich.
- Der Betrag muss auf die einzelnen Positionen aufgeteilt werden.
- Oft hilft es, das Ergebnis in einem Business Vault Satelliten zu speichern.
- Oder als transactional link.

Beginn der Faktentabelle / Inhalt von Business Vault Satellite / Bridge:

- H_Position_HK (ID für kleinstes Korn)
- Preis
- Betrag
- Ergebnis aus Rabatt / SUMME (alle Positionen.Betrag zu einer Bestellung)
- Einnahmen (der Einfachheit halber berechnet)

Die Basis-Kennzahl wird von zwei Quellensystemen gespeist

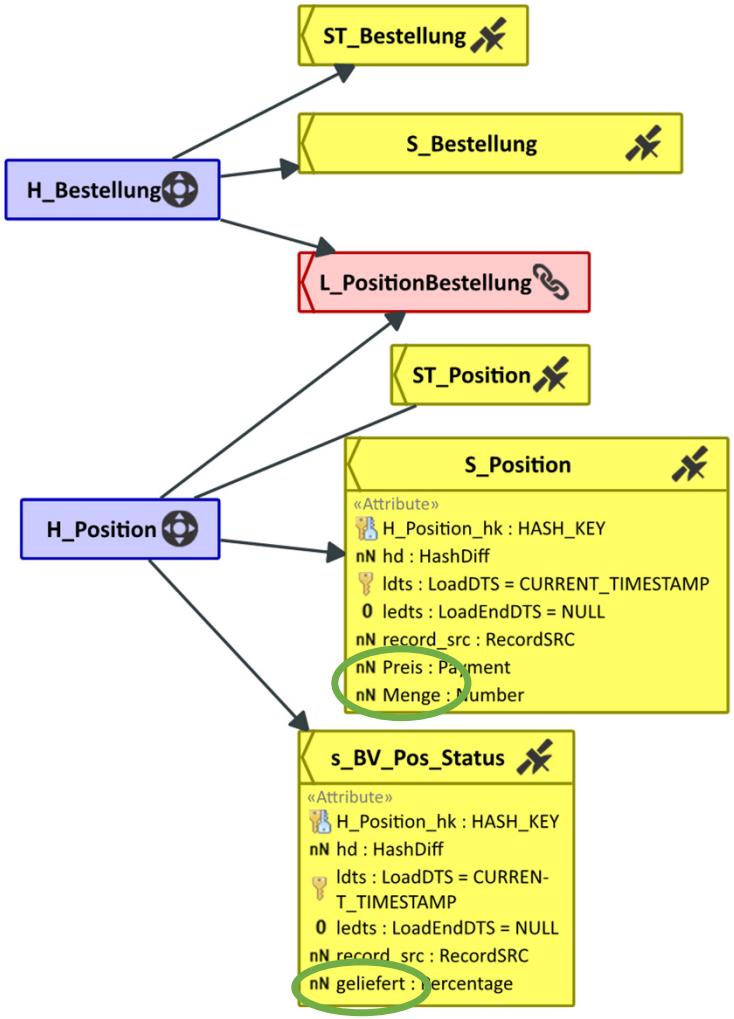


- Preis, Betrag und Rabatt stammen aus zwei Systemen.
- Auch hier wird das Ergebnis in einem Business Vault Satelliten gespeichert, falls dies hilfreich ist.
- Der Rabatt ist jetzt anders, er muss für beide geeignet sein.

Beginn der Faktentabelle / Inhalt von Business Vault Satellite / Bridge:

- H_Position_HK (ID für kleinstes Korn)
- Preis
- Betrag
- Rabatte (auch Teil der Einnahmen)
- Einnahmen (der Einfachheit halber berechnet)
- Quelle: System

Hinzufügen des Lieferzustands als Basisschlüssel

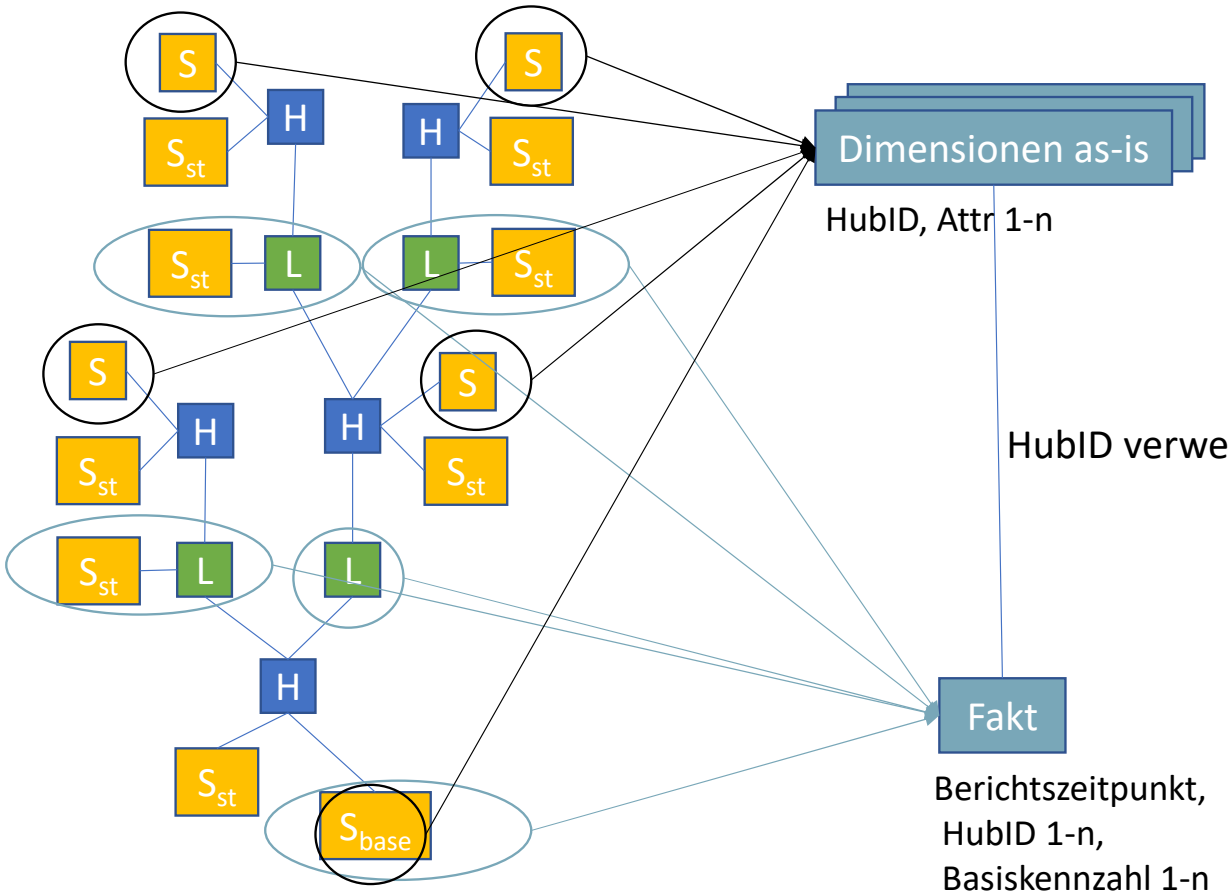


- Wenn eine Position geliefert wird, kann sie auf einer anderen Granularität gefunden werden (Lieferung)
- Der Grad der Lieferung kann für diese Granularität berechnet werden (Business Vault Satellit)
- Vorsicht beim Aggregieren dieser Kennzahl.

Beginn der Faktentabelle / Inhalt von Business Vault Satelliten / transactional Link:

- H_Position_HK (ID für kleinstes Korn)
- Preis
- Betrag
- Einnahmen (der Einfachheit halber berechnet)
- Liefergrad

Blick auf die Umsetzung – aktueller Zustand

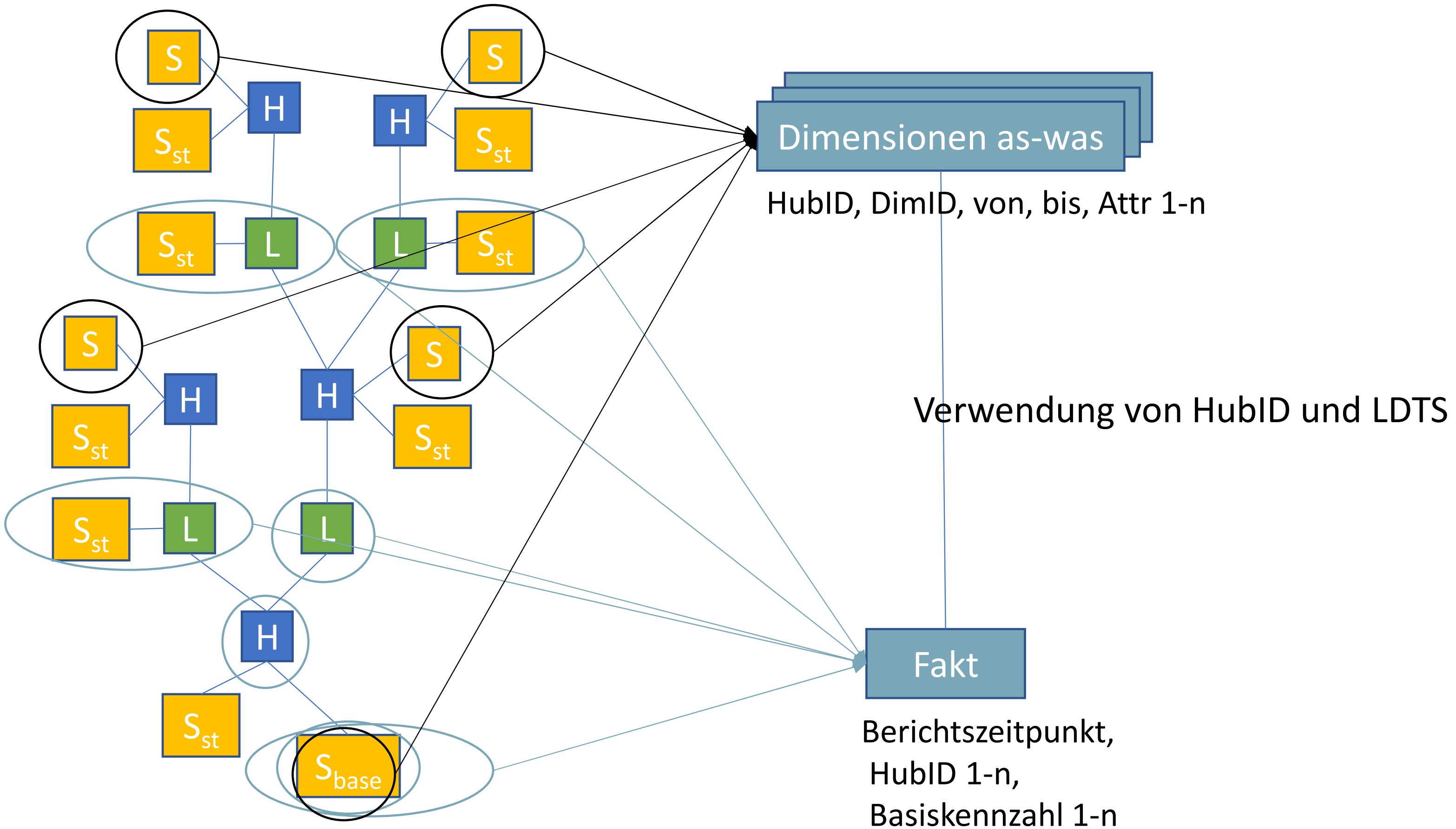


80% der Lösungen

Das Wissen der FDR nutzen, um alles für die Fakten zu sammeln

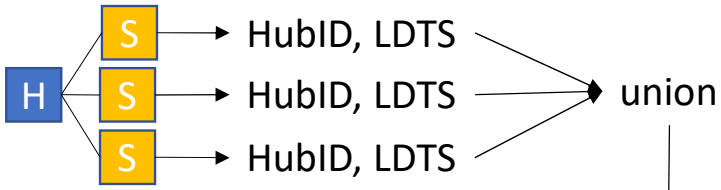
Legende
H - Hub
L - Link
S - Satellit
S_{base} - Satelit mit Basiskennzahl
S_{st} - Status Satelite für Löschungen

Betrachtung der Umsetzung – as-was, as-of snapshot



Legende
H - Hub
L - Link
S - Satellit
S_{base} - Satelit mit Basiskennzahl
S_{st} - Status Satelite für Löschungen

Dimension, die verschiedene Attribute von mehreren Satelliten aufnimmt



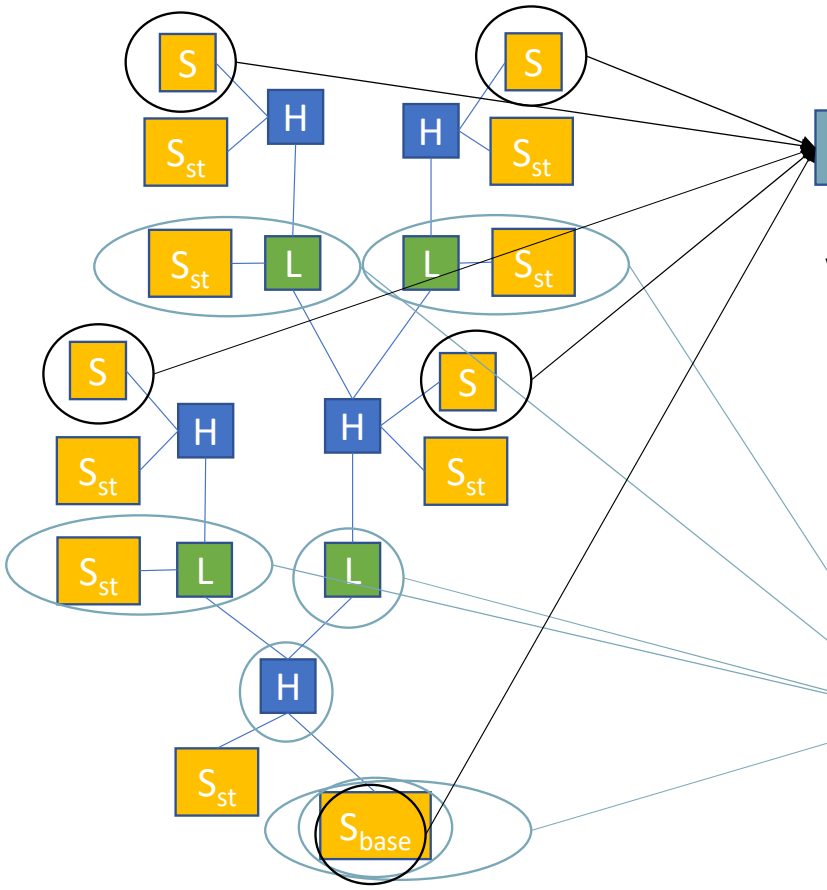
mit HubID und LDTS auswählen
auf allen drei Satelliten für jeden Eintrag
nach HubID und zwischen LDTS und LEDTS

Dimension mit
HubID, DimensionID, von, [bis,] Attribute

Beitritt zu mehreren Satellitengeschichten:

- eine Vereinigung aller HubID, LDTS erstellen
- und die ursprünglichen Satellitendaten damit abgleichen
- wenn es keine Übereinstimmung gibt, wird der ghost-record verwendet
- Erstellung einer Zeile für jedes LDTS in der union
- die DimensionID mittels row_number() erzeugen
  partitioniert nach HubID und ldts
  sortiert nach ldts
- nach Roelant Vos https://roelantvos.com/blog/creating-data-vault-point-in-time-and-dimension-tables-merging-historical-data-sources/

Betrachtet man die Umsetzung – as-was / as-of DimID



Dimensionen as-was

HubID, DimID,
von, bis, Attr 1-n

Basis-Faktor

Berichtszeitpunkt,
HubID 1-n,
Basiskennzahl 1-n

10%

Nach Berichtszeit filtern und dann einen equi-join durchführen

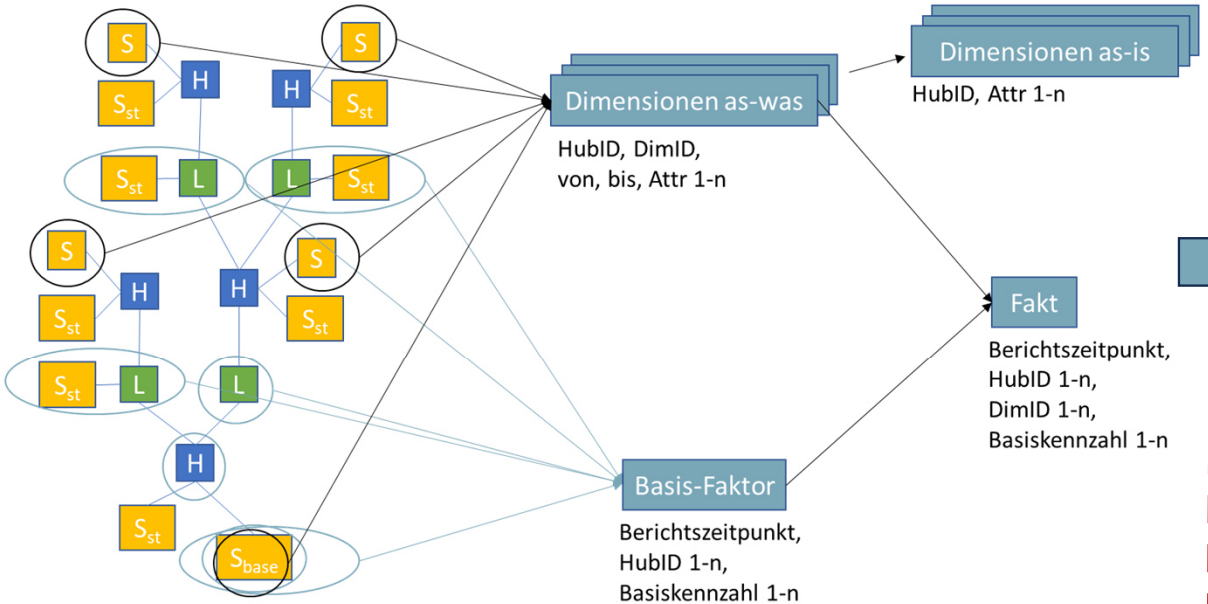
Fakt

Berichtszeitpunkt,
HubID 1-n,
DimID 1-n,
Basiskennzahl 1-n

Bei einer Änderung von Dimensionen muss nur eine DimID geändert werden.

Legende
H - Hub
L - Link
S - Satellit
S_{base} - Satelit mit Basiskennzahl
S_{st} - Status Satelite für Löschungen

Betrachtet man die Umsetzung – as-was / as-of DimID



Legende
H - Hub
L - Link
S - Satellit
S_{base} - Satelit mit Basiskennzahl
S_{st} - Status Satelite für Löschungen

-Aggregieren
-Abgeleitete Kennzahlen
-Fakten verknüpfen
-USS
-...

-> Wunschgerechte Faktentabellen ohne die Fakten neu berechnen zu müssen

Schnittstelle zum Reporting, sowie zu jedem weiteren Empfänger

Mit Anbindung an die technische Data Governance (für wen, wie oft, wie lang, wann kann es gelöscht werden.

Alle Anforderungen für einen Data Mart

Was kann automatisiert werden?

- Basiskennzahln
  - Basis-Kennzahlen
  - Abgeleitete Schlüsselzahlen ← Formel — Eine Menge Metadaten, mit eigener Grammatik / YAML
  - Berechnete Verhältnisse
- Fakten
  - Transaction Grain ← Nur eine Auswahl — Neue und geänderte Transactionen
  - Periodic snapshot Grain ← Logik der Muster — Zufluss, Abfluss, Delta
- Berichtszeitpunkt
- Dimensionen
  - as-is ← Aktuelle Sicht auf Satelliten
  - as-was ← snapshot oder DimID

    Einfache Umsetzung mit Snapshot, DimID erfordert mehr Logik
  - as-of ← Einzelfälle
- Fakt-Dimension-Relation (FDR)
  - Pfad ← erforderlich — Letztlich nur die Konfiguration dieser Pfade
  - Zeit ← Einzelfälle

Danke für Ihre Aufmerksamkeit!

Mechanik der Zeiten im Data Vault Warehouse