

# Pilot Moodle-DWH

**Was kann ein Data Warehouse und wie kann ein solches gebaut werden?**

Technische Umsetzung vom Bucket bis zum Dashboard

Dr. Andrea Kennel, fhnw  
Lea Boesch, fhnw

29. April 2025



# Dr. Andrea Kennel



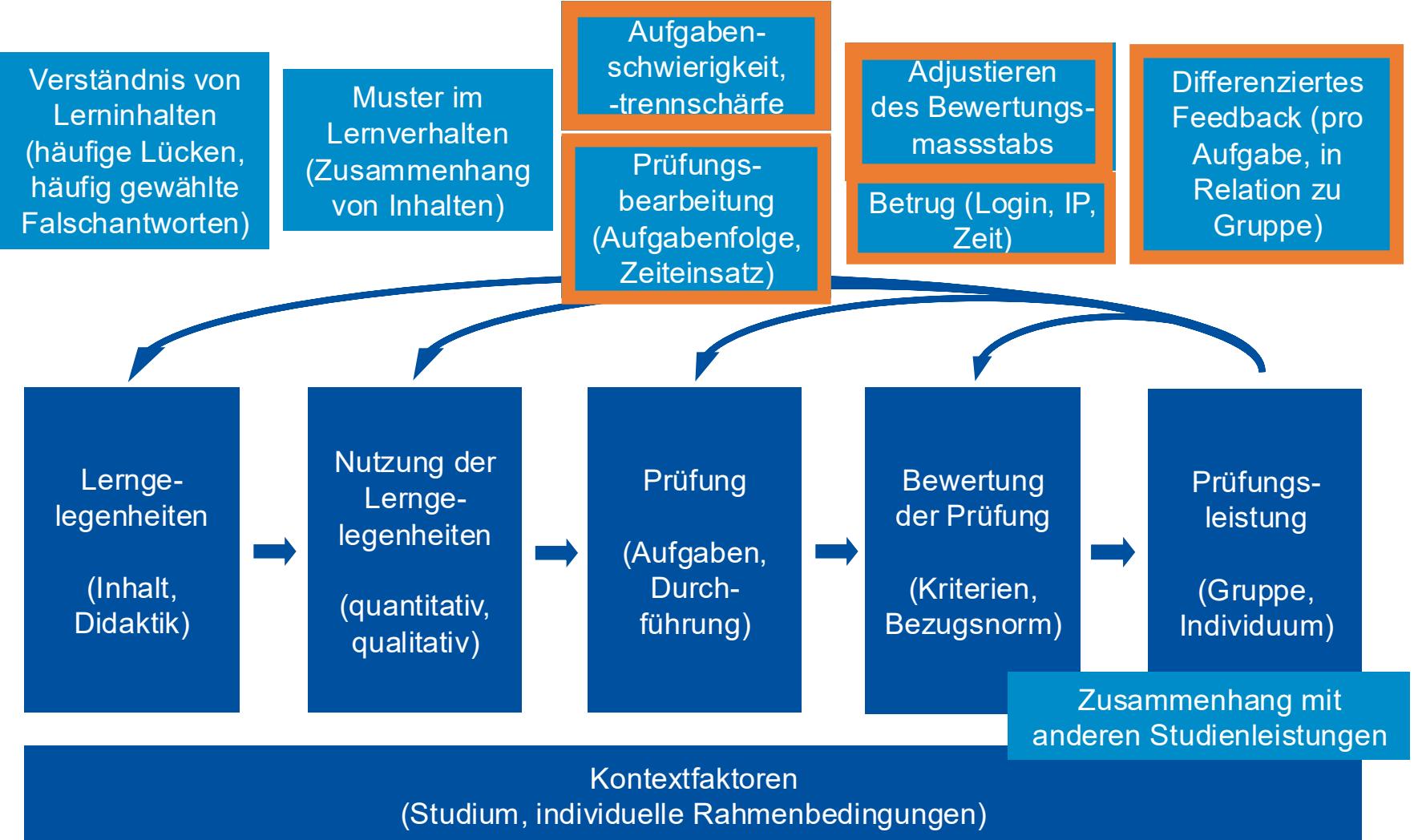
Dozentin am i4DS

# Lea Boesch



Assistentin am i4DS

# Grundlage



Aus: Scheidig, F. & Schweinberger, K. (2022). Assessment Analytics – Daten digitaler Prüfungen auswerten. In B. Berendt, A. Fleischmann, G. Salmhofer, N. Schaper, B. Szczyrba, M. Wiemer & J. Wildt (Hrsg.), Neues Handbuch Hochschullehre. Ergänzungsband Nr. 108 (H 3.9). Berlin: DUZ

# Moodle: Zeitanalyse

Antworten-Rückblick			Was	Wann	Zeit
Schritt	Zeit	Aktion			
1	3. März 2025, 13:23:04	<b>Antworten-Rückblick</b>	<b>Start</b>		<b>13:23:04</b>
2	3. März 2025, 13:25:25		<b>Weg von</b>	1	13:25:25
					00:02:21
1	3. März 2025, 13:23:04		<b>Weg von</b>	2	13:27:12
					00:01:47
2	3. März 2025, 13:27:43		<b>Weg von</b>	1	13:27:43
					00:00:31
3	3. März 2025, 13:34:53		<b>Weg von</b>	3	13:30:47
					00:03:04
				1	00:02:52
				2	00:01:47
				3	00:03:04

# Moodle DWH: Zeitanalyse

Moodle-DWH

andrea.kennel@fhnw.ch

Home Bearbeitungsdauer MC, MTF, Prime

Prüfung wählen

MSP FS23

Hier klicken für übersichtlichere Chart-Darstellung

Durchschnittliche Bearbeitungszeit pro Frage

Alle anzeigen Balkendiagramm Box-Plot

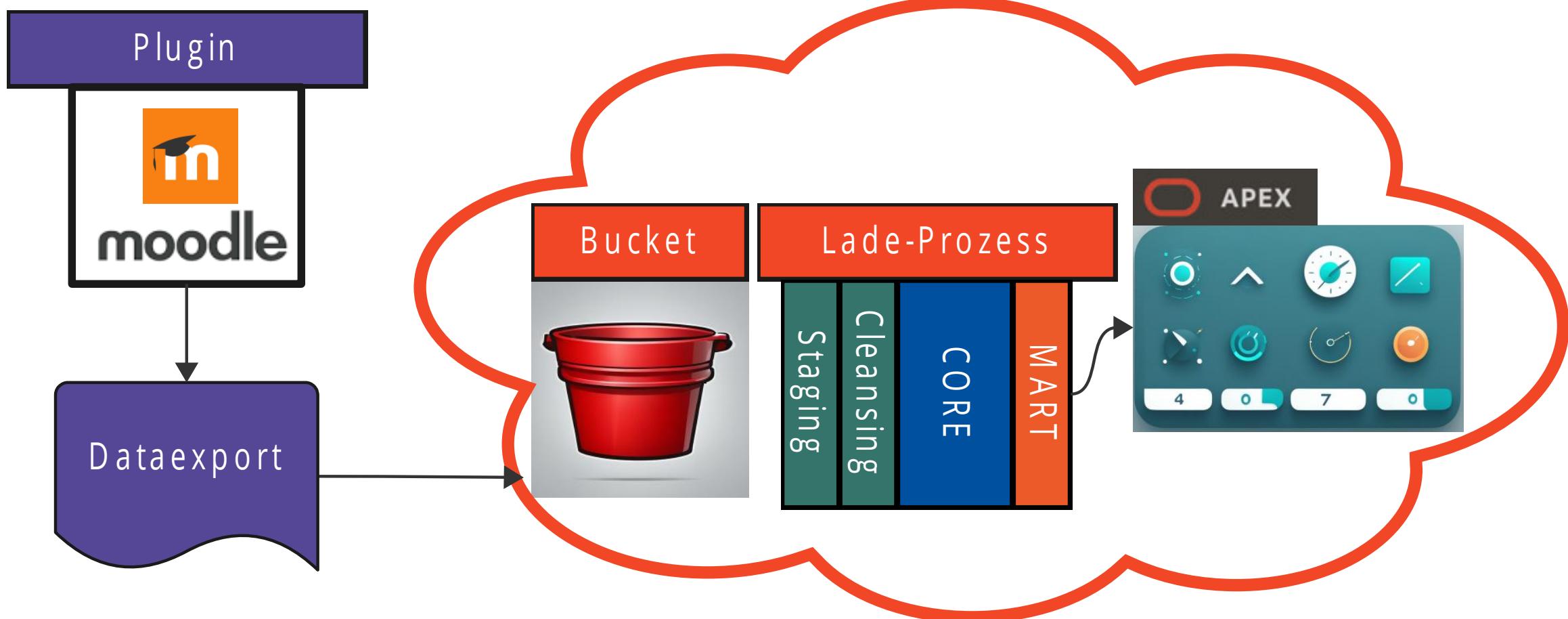
Box-Plot

The box plot displays the distribution of completion times for five tasks. The y-axis lists the tasks: 01 Aufgabe eines DBMS 2, 02 Aufgaben eines DBMS 1, 03 ERD erstellen, 04 GROUP - Variante 1, and 05 GROUP - Variante 2. Each task has a light blue box representing the interquartile range (IQR), a vertical line inside the box representing the median, and whiskers extending to the minimum and maximum values. Individual data points are shown as blue dots. Task 03 shows the widest distribution with a large IQR and whiskers extending to the top of the chart area.

Aufgabe	Median (approx.)	Q1 (approx.)	Q3 (approx.)	Min (approx.)	Max (approx.)
01 Aufgabe eines DBMS 2	10	5	15	5	25
02 Aufgaben eines DBMS 1	10	5	15	5	35
03 ERD erstellen	15	10	20	5	45
04 GROUP - Variante 1	15	10	20	5	35
05 GROUP - Variante 2	15	10	20	5	35

<https://qdbb147251565c3-moodykea.adb.eu-frankfurt-1.oraclecloudapps.com/ords/#p4-box-chart>

# Übersicht



# Datenschutz

User ist doppelt anonymisiert

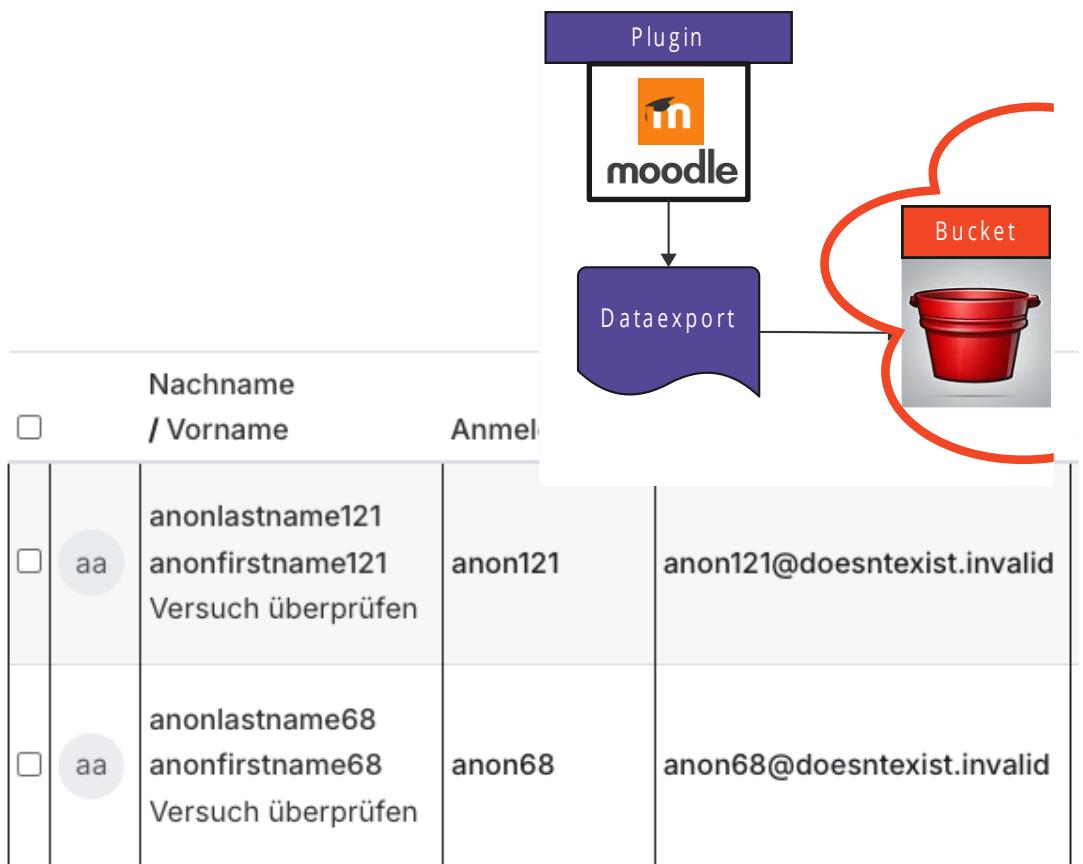
1. In Moodle User anonymisieren

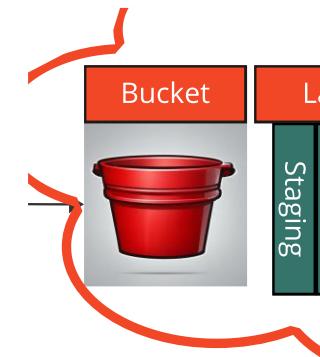
2. Nur Hash exportieren

SELECT

```
...  
  (SELECT MD5(username) FROM mdl_user WHERE id = qzat.userid) AS "qzatuserid",  
...
```

3. Daten „flachgeklopft“ als csv ins Bucket





# Vom Bucket in die Stage mit DBMS\_CLOUD.COPY\_DATA

```
"C##CLOUD$SERVICE"."DBMS_CLOUD"."COPY_DATA"  
( TABLE_NAME      => l_TABLE_NAME  
,CREDENTIAL_NAME => l_CREDENTIAL_NAME  
,FILE_URI_LIST   => l_FILE_URI_LIST  
,FIELD_LIST       => l_FIELD_LIST  
,FORMAT          => l_FORMAT  
,OPERATION_ID    => l_OPERATION_ID  
);
```

```
v_full_path VARCHAR2(2000) := p_bucket_path || p_file_name;  
  
-- DECLARE  
l_TABLE_NAME      DBMS_QUOTED_ID := '"DATA_VERSION_BACH"';  
l_CREDENTIAL_NAME DBMS_QUOTED_ID := '"OBJ_STORE_CRED"';  
l_FILE_URI_LIST   CLOB := v_full_path;  
l_FIELD_LIST      CLOB :=  
q'[  
  "SOURCESYSTEM"           CHAR(4000)  
  , "AUTHORIZEDUSERS"       CHAR(4000)  
  , "QUESTIONID"             CHAR|
```

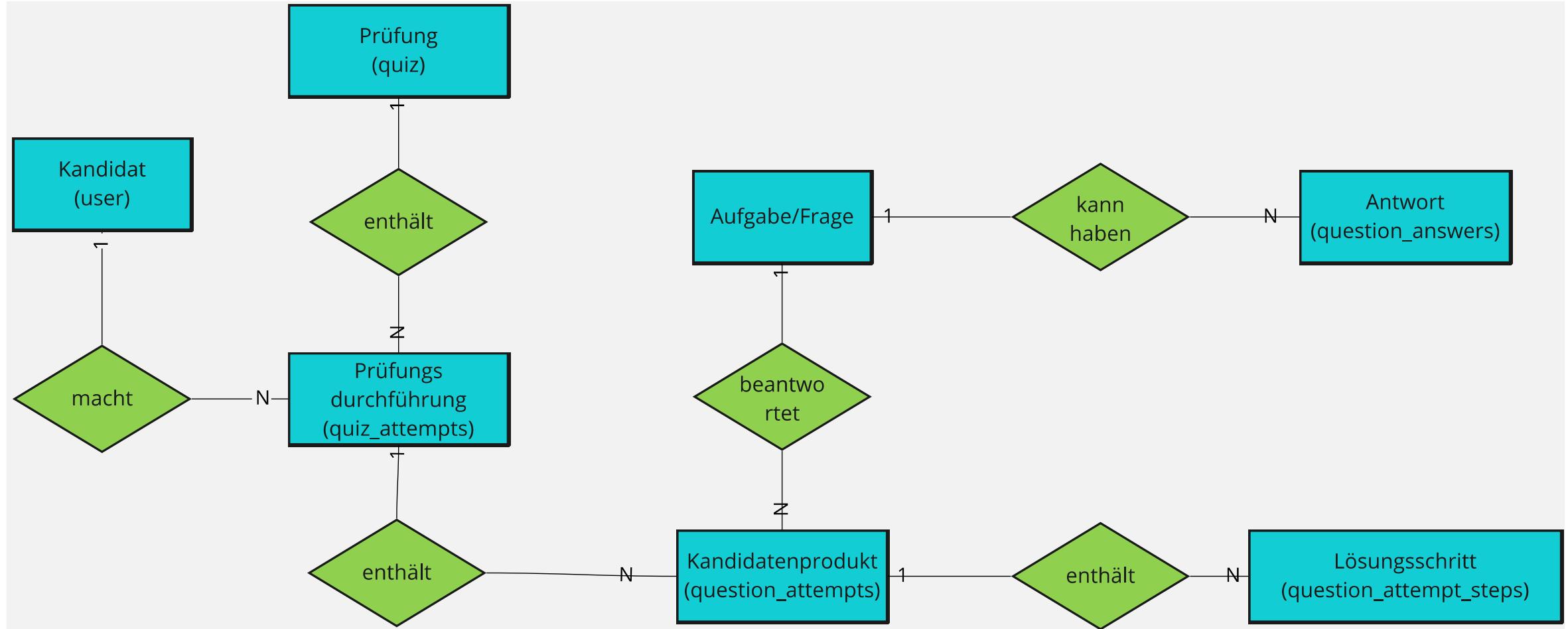
Daten aus CSV 1:1 in eine Tabelle

# Das Datenmodell der Quelle

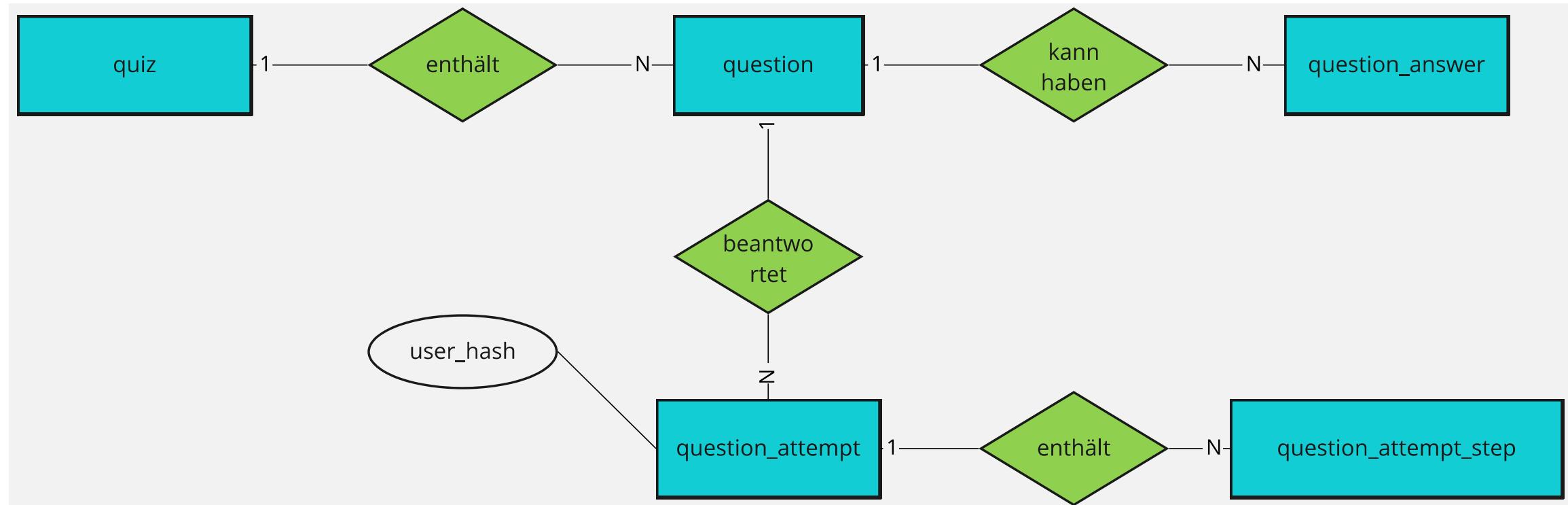
Das Datenmodell von Moodle ist öffentlich zugänglich, aber umfangreich

<https://www.examulator.com/er/4.0/tables/quiz.html>

# Das Datenmodell der Quelle (Ausschnitt)



# Das Modell in der Cleansing

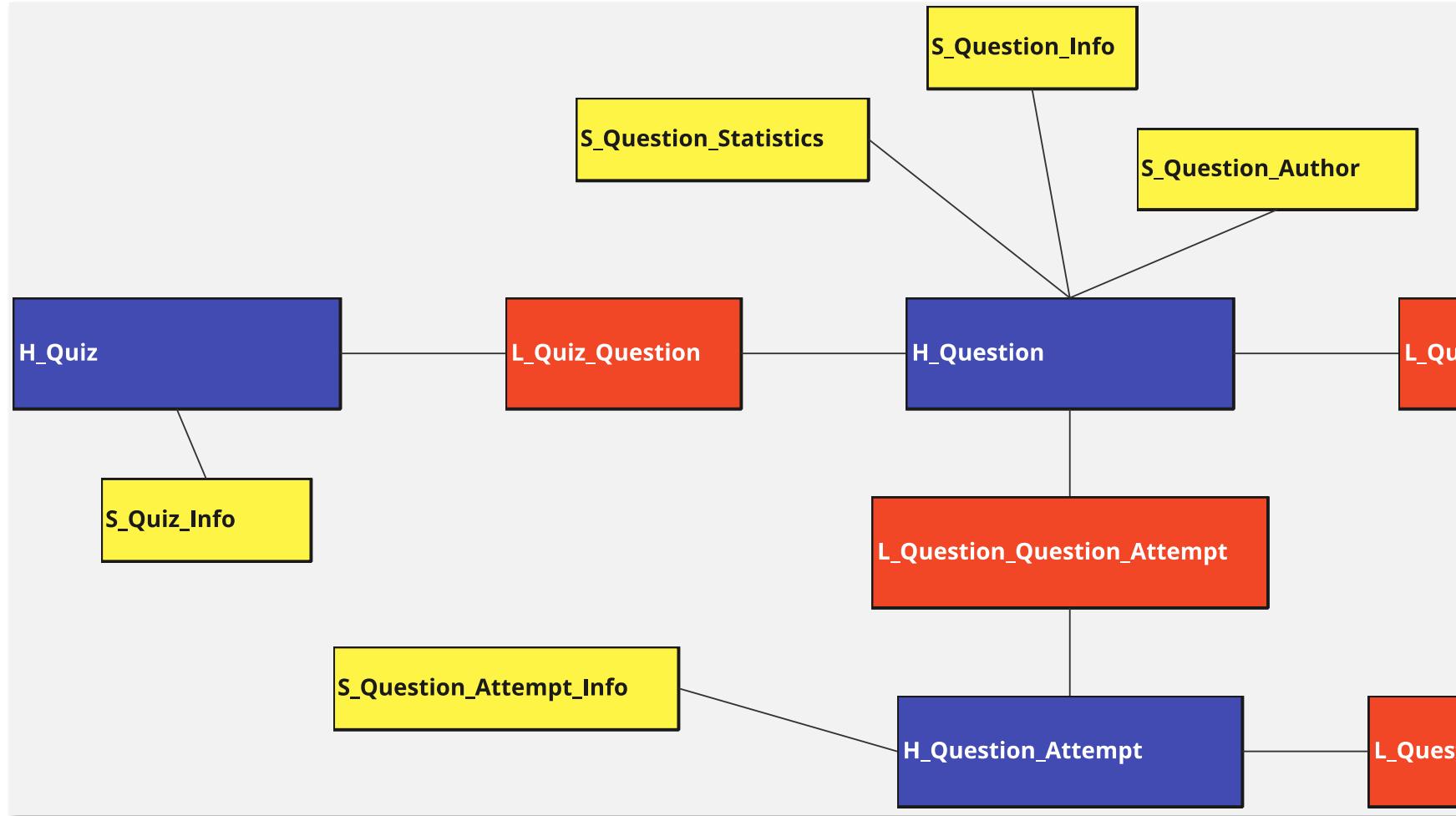


# Von der Stage in die Cleansing

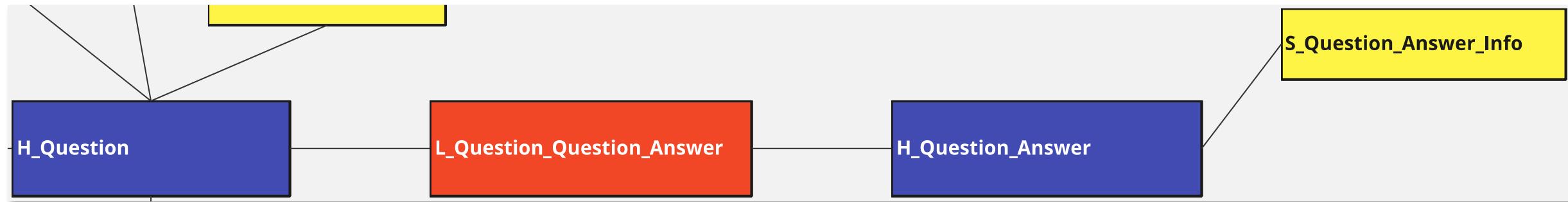
```
INSERT INTO MD_CLEANSING.quiz (
    load_id,
    sourcesystem,
    quiz_id,
    authorizedusers,
    quiz_name,
    quiz_grade)
SELECT DISTINCT
    p_load_id load_id,
    sourcesystem,
    quizid quiz_id,
    authorizedusers,
    quizname quiz_name,
    quizgrade quiz_grade
FROM MD_STAGING.data_view d;
```

```
INSERT INTO MD_CLEANSING.question_attempt_step_data (
    load_id,
    sourcesystem,
    authorizedusers,
    question_attempt_step_data_id,
    question_attempt_step_id,
    step_data_name,
    step_data_value)
SELECT DISTINCT
    p_load_id load_id,
    sourcesystem sourcesystem,
    authorizedusers authorizedusers,
    questionattemptstepdataid question_attempt_step_data_id,
    qastepid question_attempt_step_id,
    questionattemptstepdataname step_data_name,
    questionattemptstepdatavalue step_data_value
FROM MD_STAGING.data_view
WHERE questionattemptstepdataid IS NOT NULL;
```

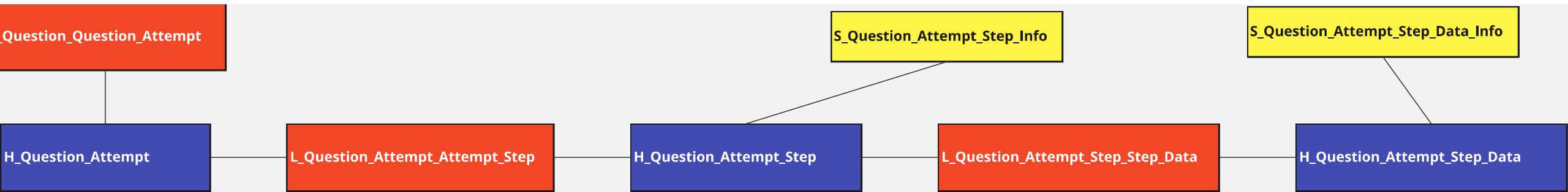
# Das Modell im Core



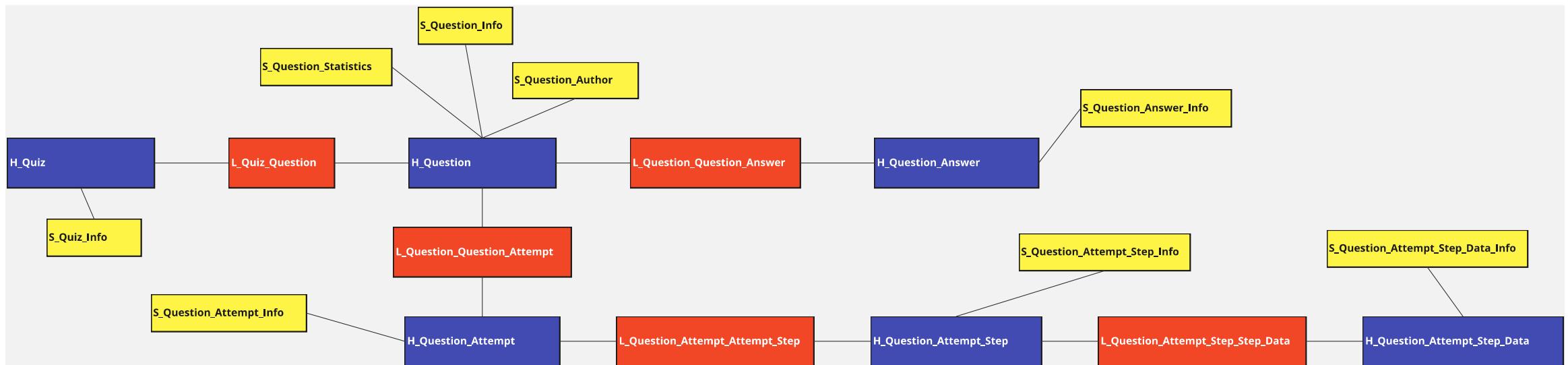
# Das Modell im Core

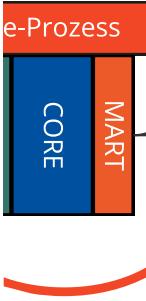


# Das Modell im Core



# Das Modell im Core

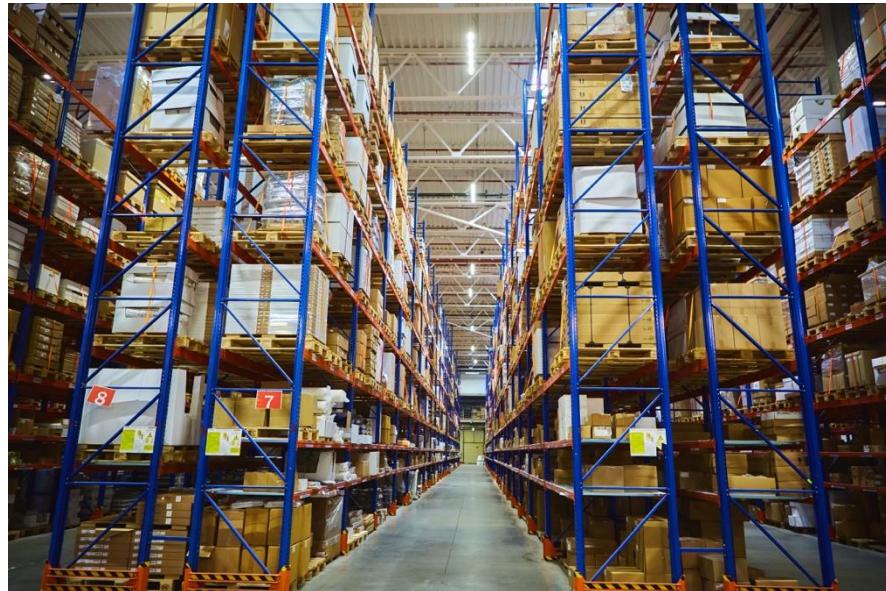




# MART & Interface für End User

# MART

CORE



VS.

MART



Bilder: Adobe Stock

# MART-Schicht



Zweck des Marts: Daten für gewünschte Analysen bereitstellen

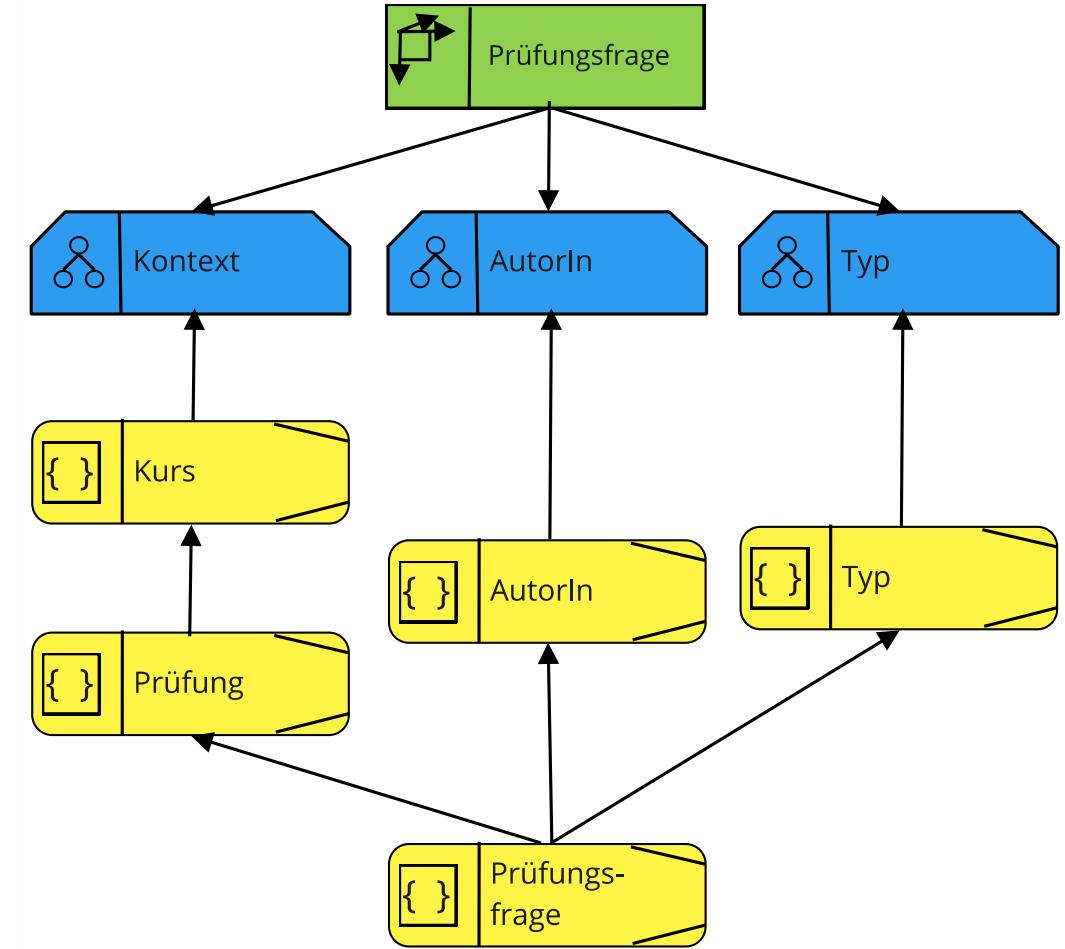
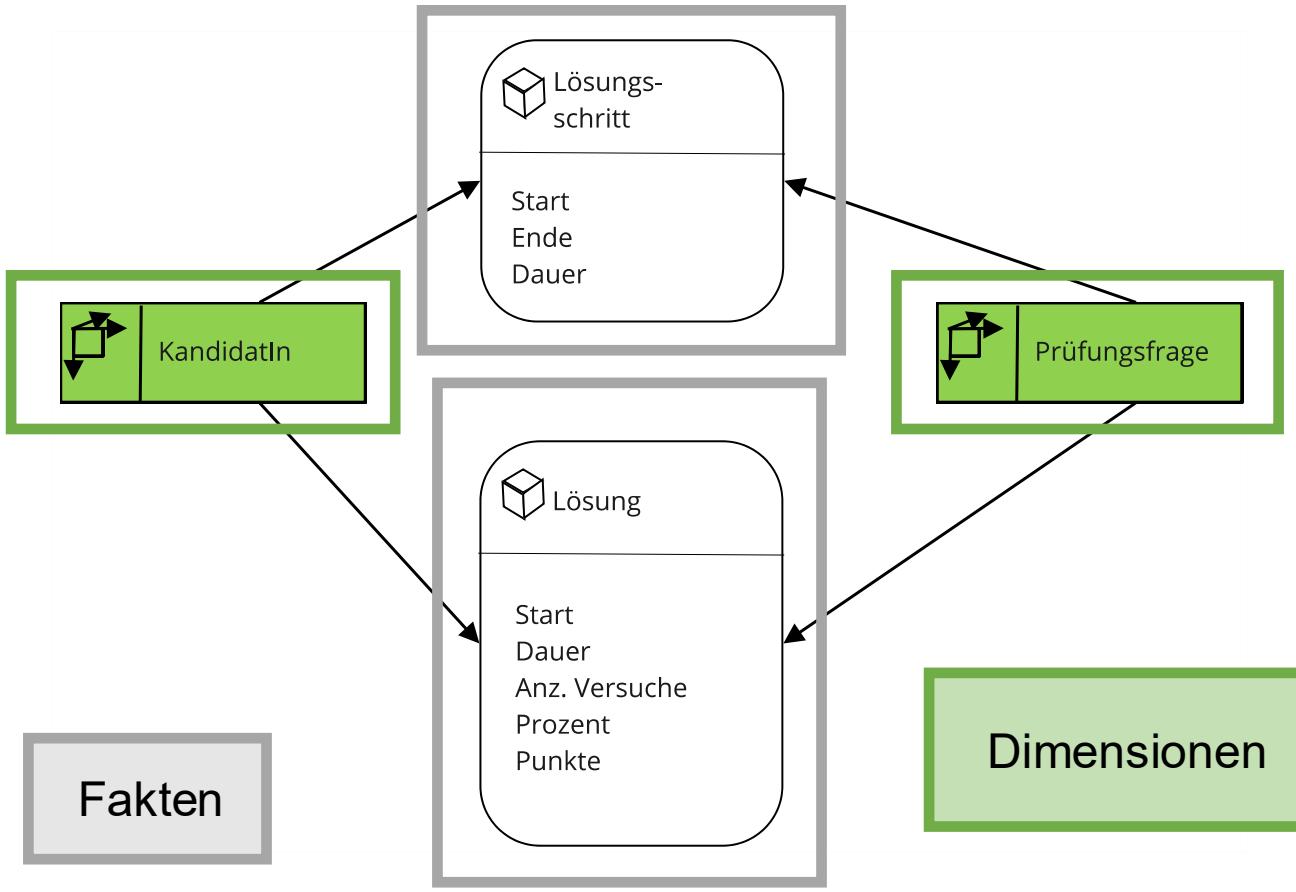
- Anderer Ansatz als im Core: Auf die Bedürfnisse der End User zugeschnitten.
- Andere Konzepte und Konventionen für die Modellierung

Bilder: Adobe Stock

# Fragestellung 1

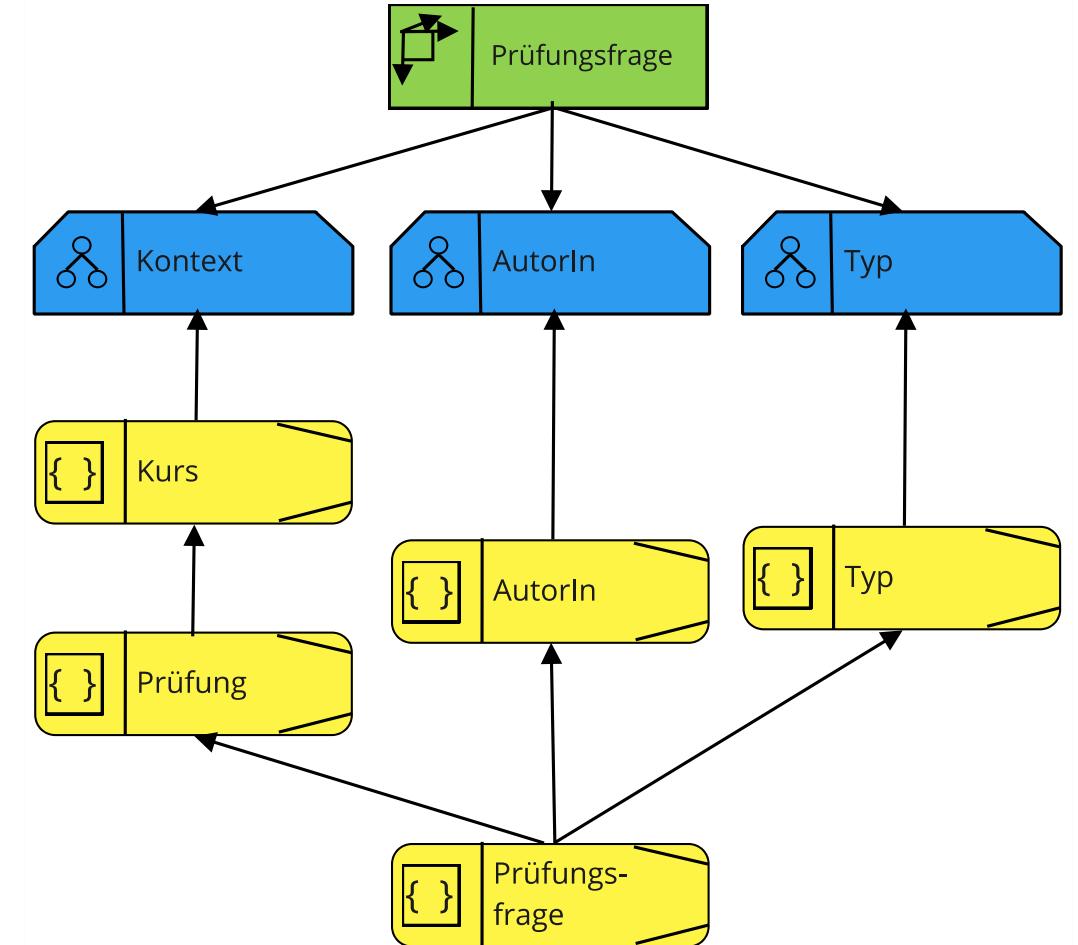
Wie viel Zeit wurde für die verschiedenen  
Prüfungsfragen eingesetzt?

# ADAPT-Modellierung

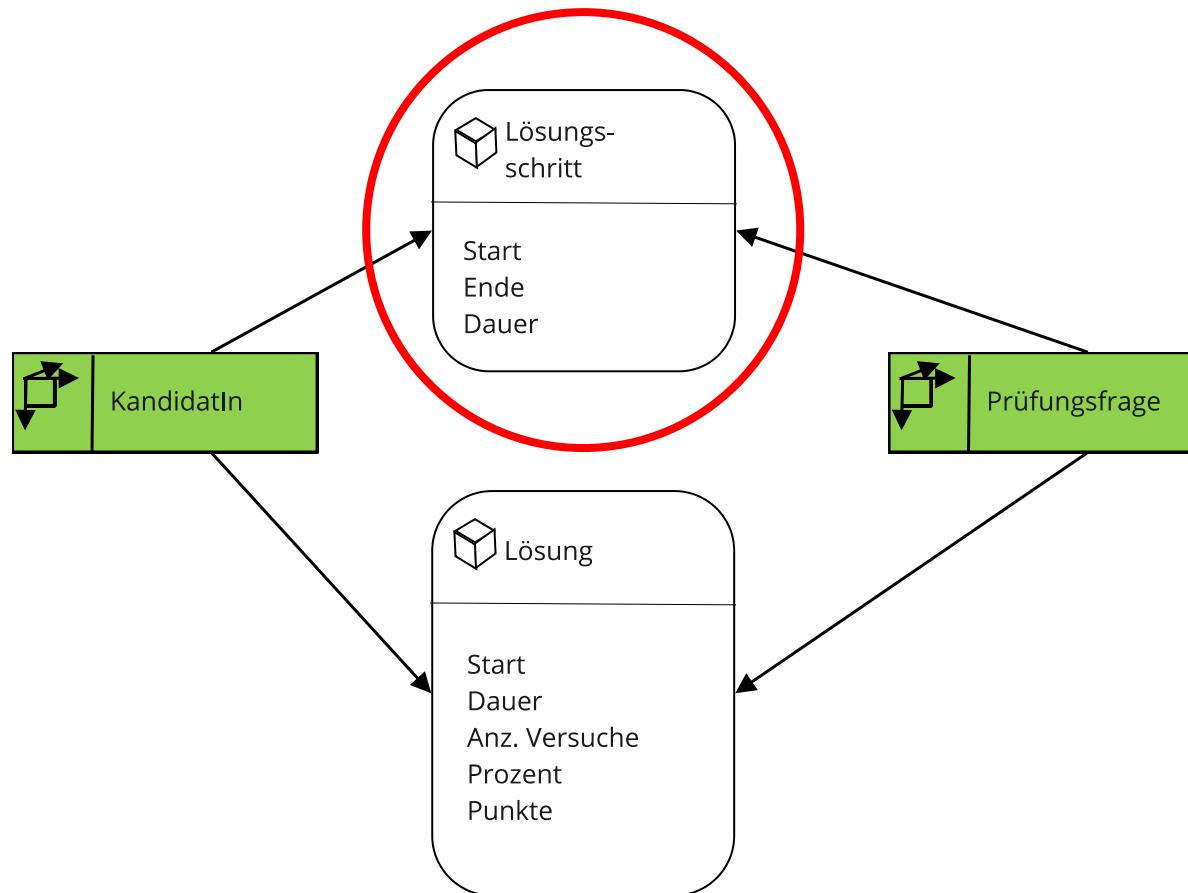


# Vom Core in die Marts – Prüfungsfrage

```
INSERT INTO MD_MART.DM_D_Question
SELECT
    seq_md_id_question.nextval MD_ID_Question,
    q.question_id QU_MD_BK_question_id,
    q.load_id QZ_MD_BK_load_id,
    q.question_name QU_question_name,
    q.question_text QU_question_text,
    ...
    q.question_type TY_question_type
FROM MD_CORE.v_question q
WHERE q.load_id = this_load_id;
```



# Vom Core in die Marts – Lösungsschritt



# Moodle: Zeitanalyse

Antworten-Rückblick			Was	Wann	Zeit
Schritt	Zeit	Aktion			
1	3. März 2025, 13:23:04	<b>Antworten-Rückblick</b>	<b>Start</b>	<b>13:23:04</b>	
2	3. März 2025, 13:25:25		<b>Weg von</b>	<b>1</b>	<b>13:25:25</b> <b>00:02:21</b>
			<b>Weg von</b>	<b>2</b>	<b>13:27:12</b> <b>00:01:47</b>
3	3. März 2025, 13:27:43		<b>Weg von</b>	<b>1</b>	<b>13:27:43</b> <b>00:00:31</b>
			<b>Weg von</b>	<b>3</b>	<b>13:30:47</b> <b>00:03:04</b>
4	3. März 2025, 13:34:53			<b>1</b>	<b>00:02:52</b>
				<b>2</b>	<b>00:01:47</b>
				<b>3</b>	<b>00:03:04</b>

# Vom Core in die Marts – Lösungsschritt

MD_ID_QUESTION	MD_ID_CANDIDATE	END_TIME	START_TIME
1847	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:03:52	2023-10-03 12:03:52
1847	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:05:01	2023-10-03 12:05:01
1848	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:07:17	2023-10-03 12:07:17
1849	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:12:26	2023-10-03 12:07:17
1848	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:13:20	2023-10-03 12:12:26
1849	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:17:47	2023-10-03 12:13:20
1848	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:18:52	2023-10-03 12:17:47
			2023-10-03 12:18:52

# Vom Core in die Marts – Lösungsschritt

MD_ID_QUESTION	MD_ID_CANDIDATE	START_TIME	END_TIME	DURATION
1847	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:03:52	2023-10-03 12:05:01	69
1848	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:05:01	2023-10-03 12:07:17	136
1849	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:07:17	2023-10-03 12:12:26	309
1848	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:12:26	2023-10-03 12:13:20	54
1849	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:13:20	2023-10-03 12:17:47	267
1848	ba6ec0d000fbc7360dfee3e555bb4b90	2023-10-03 12:17:47	2023-10-03 12:18:52	65

# Vom Core in die Marts – Lösungsschritt

```
INSERT INTO MD_MART.DM_F_Question_Submission_Time_Segment
SELECT
    MD_ID_Question,
    quiz_attempt_user_hash MD_ID_Candidate,
    time_prev_start_time,
    createdunixtime end_time, time_spent duration, load_id
FROM (
    SELECT
        d_q.MD_ID_Question,
        qa.quiz_attempt_user_hash, ...
        LAG(step.createdunixtime, 1, 0) OVER (PARTITION BY qa.QUIZ_ATTEMPT_USER_HASH
            ORDER BY step.createdunixtime) AS time_prev,
        step.createdunixtime -
        LAG(step.createdunixtime, 1, 0) OVER (PARTITION BY qa.QUIZ_ATTEMPT_USER_HASH
            ORDER BY step.createdunixtime) AS time_spent
    FROM md_core.V_QUESTION_ATTEMPT qa JOIN
        MD_CORE.V_QUESTION_ATTEMPT_STEP step
        ON qa.H_QUESTION_ATTEMPT_SID = step.H_QUESTION_ATTEMPT_SID JOIN
        ...
        MD_MART.DM_D_Question d_q ON d_q.QU_MD_BK_question_id = q.question_id AND d_q.QZ_MD_BK_load_id = q.load_id
    WHERE qa.load_id = this_load_id)
```

# Vom Core in die Marts – Lösungsschritt

```
LAG(step.createdunixtime, 1, 0)
```

```
OVER (
```

**LAG(expr, offset, default)**

```
PARTITION BY qa.QUIZ_ATTEMPT_USER_HASH  
ORDER BY step.createdunixtime)
```

```
AS time_prev
```

**OVER (PARTITION BY ... ORDER BY ...)**

# Vom Core in die Marts – Lösungsschritt

Wichtig hier: Testdaten!

Eigens erstellte und mit Timer ausgefüllte Moodle-Prüfung  
→ Protokoll



# Interface für End User: Oracle APEX

# Proprietäre Anwendung von Oracle

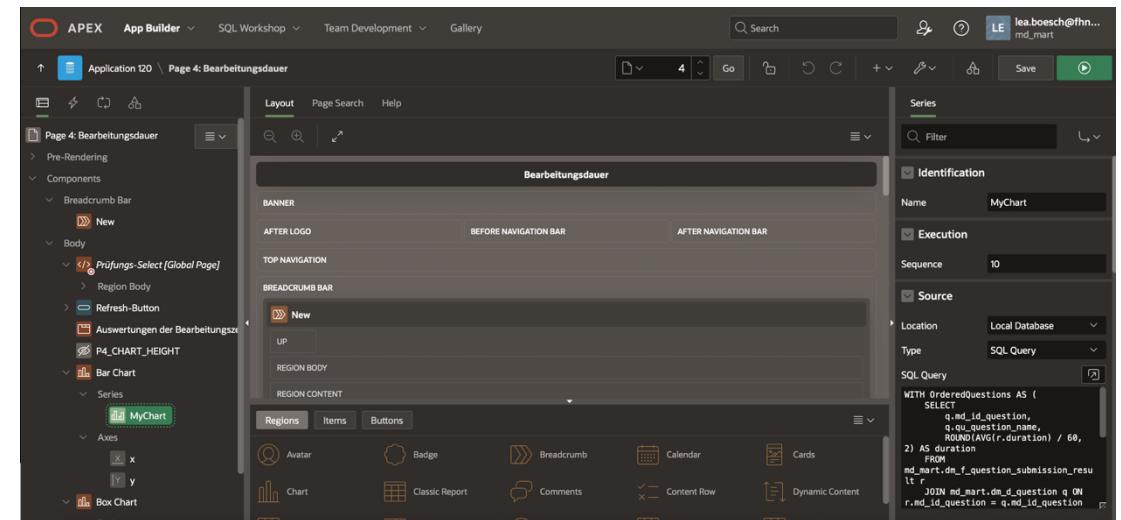
→ Auch in der kostenlosen Oracle-Cloud-Lizenz verfügbar.

# Browserbasiert

## Direkter Datenbankzugriff

## Fazit:

- Schnelle erste Resultate
  - Einfache Standard-Charts können leicht generiert werden
  - Grosse Menge an Funktionalitäten
  - Unintuitiv und unübersichtlich
  - Abweichungen von den Basic-Funktionen können aufwändig werden





# Interface für End User: Oracle APEX

Moodle-DWH

Willkommen bei Moodle-DWH

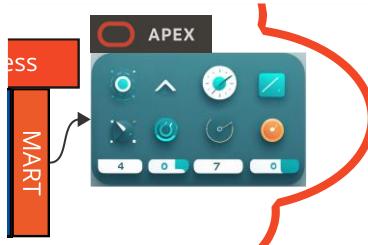
[Was ist Moodle-DWH?](#)

[Anleitung für Dozierende](#)

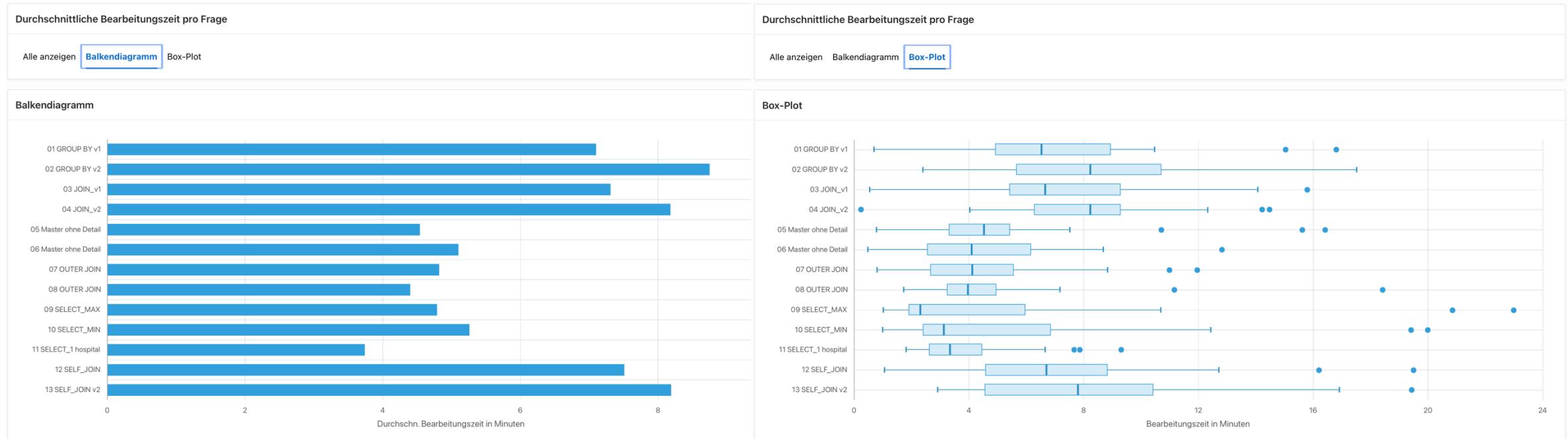
Home

Bearbeitungsdauer

MC, MTF, KPrime



# Interface für End User: Oracle APEX





## Fragestellung 2

Bei Multiple-Choice-Fragen:

Welche Antwortmöglichkeit wurde wie oft korrekt bzw. inkorrekt beantwortet?



# Vom Core in die Marts – Gewählte Antwortoptionen

Wahl der Antwortoptionen

STEP_DATA_NAME	STEP_DATA_VALUE
choice0	1
choice1	1
choice2	0
choice3	0
choice4	0
_order	680,684,681,683,682

???

ChatGPT!

Antwortoptionen

QUESTIONANSWERSID	ANSWER
683	<p dir="ltr"></p><p dir="ltr">SELECT gruppen_name</p><p dir="ltr">
681	<p dir="ltr" style="text-align: left;"></p><p dir="ltr">
684	<p></p><p dir="ltr">SELECT ig.gruppen_name</p><p dir="ltr">
682	<p dir="ltr"></p><p dir="ltr">SELECT gruppen_name</p><p dir="ltr">
680	<p dir="ltr"><span style="font-size: 0.9375rem;">SELECT i



# Vom Core in die Marts – Gewählte Antwortoption

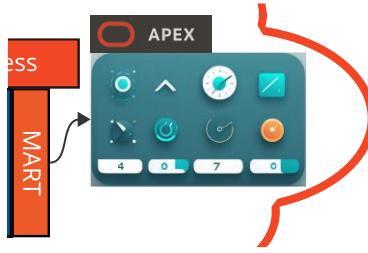
Ausgehend vom folgenden Oracle SQL Select

```
„select d.step_data_name, d.step_data_value  
from ...“
```

mit Resultat „choice0 1 choice4 0 choice3 0 choice2 0 choice1  
1 \_order 680,684,681,683,682“

möchte ich das Select so anpassen, dass die Spalte  
"step\_data\_value" ersetzt wird durch die Werte, welche bei  
der Resultat-Zeile mit step\_data\_name "\_order" erscheinen.  
Dabei soll auch die Zeile mit "\_order" im Resultat nicht mehr  
angezeigt werden.

STEP_DATA_NAME	STEP_DATA_VALUE
choice0	1
choice1	1
choice2	0
choice3	0
choice4	0
_order	680,684,681,683,682



# Vom Core in die Marts – Gewählte Antwortoption

...  
danke. choice\_data hat keine Spalte h\_question\_attempt\_sid, das  
müstest du aus v\_question\_attempt\_step oder v\_question\_attempt  
holen.

→ Datenmodell kennen

...  
das Ergebnis kommt leer zurück

...  
Ich sehe den Fehler. Die \_order-Zeile hat nicht dieselbe  
h\_question\_attempt\_step\_sid wie die "choice"-Zeilen. Du müsstest  
für die Verknüpfung auf h\_question\_attempt\_sid gehen

→ Logik der Verknüpfungen kennen



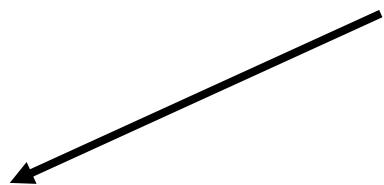
# Vom Core in die Marts – Gewählte Antwortoption

super. nun möchte ich noch die ursprünglichen Werte von step\_data\_value der Choice-Zeilen auch anzeigen, und die neue Zeile der gesplitteten \_order-Zeile zu "questionanswersid" umbenennen.

...langes verschachteltes SQL ...

STEP_DATA_NAME	ORIGINAL_STEP_DATA_VALUE	QUESTIONANSWERSID
choice0	1	680
choice1	1	684
choice2	0	681
choice3	0	683
choice4	0	682

STEP_DATA_NAME	STEP_DATA_VALUE
choice0	1
choice1	1
choice2	0
choice3	0
choice4	0
_order	680,684,681,683,682





# Vom Core in die Marts – Gewählte Antwortoption

```

WITH order_data AS (
  SELECT
    a.h_question_attempt_sid,
    d.step_data_value AS order_values,
    a.quiz_attempt_user_hash,
    q.question_id,
    q.question_type
  FROM
    MD_CORE.v_question_attempt_step_data d
  JOIN
    MD_CORE.v_question_attempt_step s ON d.h_question_attempt_step_id =
    s.h_question_attempt_step_id
  JOIN
    MD_CORE.v_question_attempt a ON s.h_question_attempt_step_id =
    a.h_question_attempt_sid
  JOIN
    md_core.v_question q on a.h_question_sid = q.h_question_sid
  WHERE
    d.step_data_name = '_order'
    AND d.load_id = this_load_id
    AND q.question_type = 'multichoice'
),
choice_data AS (
  SELECT
    d.step_data_name,
    d.step_data_value AS original_step_data_value,
    d.h_question_attempt_step_id,
    s.h_question_attempt_step_id,
    s.createdunixtime
  FROM
    MD_CORE.v_question_attempt_step_data d
  JOIN
    MD_CORE.v_question_attempt_step s ON d.h_question_attempt_step_id =
    s.h_question_attempt_step_id
  WHERE
    d.step_data_name LIKE 'choice%'
    AND d.load_id = this_load_id
),

```

```

choice_order AS (
  SELECT
    c.step_data_name,
    c.original_step_data_value,
    c.order_values,
    TO_NUMBER(REGEXP_SUBSTR(o.order_values, '[^,]+', 1, CAST(SUBSTR(c.step_data_name, 7) AS INT) + 1)) as questionanswersid,
    c.h_question_attempt_step_id,
    c.h_question_attempt_sid,
    o.quiz_attempt_user_hash,
    c.createdunixtime,
    o.question_id,
    o.question_type
  FROM
    choice_data c
  JOIN
    order_data o ON c.h_question_attempt_sid = o.h_question_attempt_sid
)
SELECT
  dm_d_question.md_id_question,
  co.quiz_attempt_user_hash MD_ID_CANDIDATE,
  dm_d_question_answer_mc.md_id_question_answer_mc MD_ID_QUESTION_ANSWER_MC,
  to_boolean(cast(co.original_step_data_value as varchar2(1))) selected, -- ist ursprünglich
  ein CLOB
  TO_DATE('1970-01-01', 'YYYY-MM-DD') + NUMTODSINTERVAL(co.createdunixtime, 'SECOND')
  saved_at,
  this_load_id load_id
  FROM choice_order co
  JOIN dm_d_question_answer_mc on co.questionanswersid =
  dm_d_question_answer_mc.QA_MD_BK_questionanswersid
  AND dm_d_question_answer_mc.QZ_MD_BK_load_id = this_load_id
  join dm_d_question on co.question_id = dm_d_question.QU_MD_QUESTION_ID
  and dm_d_question.QZ_MD_BK_load_id = this_load_id
;

```



# Vom Core in die Marts – Gewählte Antwortoption

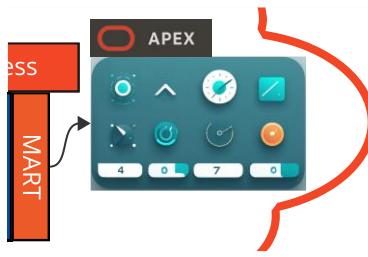
```
...  
SELECT  
    c.step_data_name,  
    c.original_step_data_value,  
  
    REGEXP_SUBSTR(o.order_value  
        , '^([^\d]*)(\d+)([^\d]*)$') AS order_value,  
    CAST(SUBSTR(c.step_data_name, 1, 1) AS NUMBER)  
        AS step_data_name_length
```

STEP_DATA_NAME	STEP_DATA_VALUE
choice0	1
choice1	1
choice2	0
choice3	0
choice4	0
_order	680,684,681,683,682

AS questionanswersid,

•

STEP_DATA_NAME	ORIGINAL_STEP_DATA_VALUE	QUESTIONANSWERSID
choice0	1	680
choice1	1	684
choice2	0	681
choice3	0	683
choice4	0	682



# Vom Core in die Marts – Gewählte Antwortoption

**In diesem Fall hilfreich bei der Arbeit mit ChatGPT:**

- Kontext über den benötigten Teil des Datenmodells mitliefern
- Genaue Formulierung (das bedingt, dass ich genau weiß, was ich möchte)
- Verständnis des Datenmodells
- Verständnis, wie die Daten zusammenhängen
- Testfälle → erhaltene Antwort überprüfbar
- Schrittweises Vorgehen (siehe verschachteltes SQL)



# Fragestellung 2 in APEX

Welche Antwort wurde wie oft korrekt/inkorrekt beantwortet?

Betrachte die folgenden Statements

1. SELECT A1, A2, ... FROM A JOIN B ON ...
2. SELECT A1, A2, ... FROM A LEFT JOIN B ON ...
3. SELECT A1, A2, ... FROM A RIGHT JOIN B ON ...
4. SELECT A1, A2, ... FROM A OUTER JOIN B ON ...

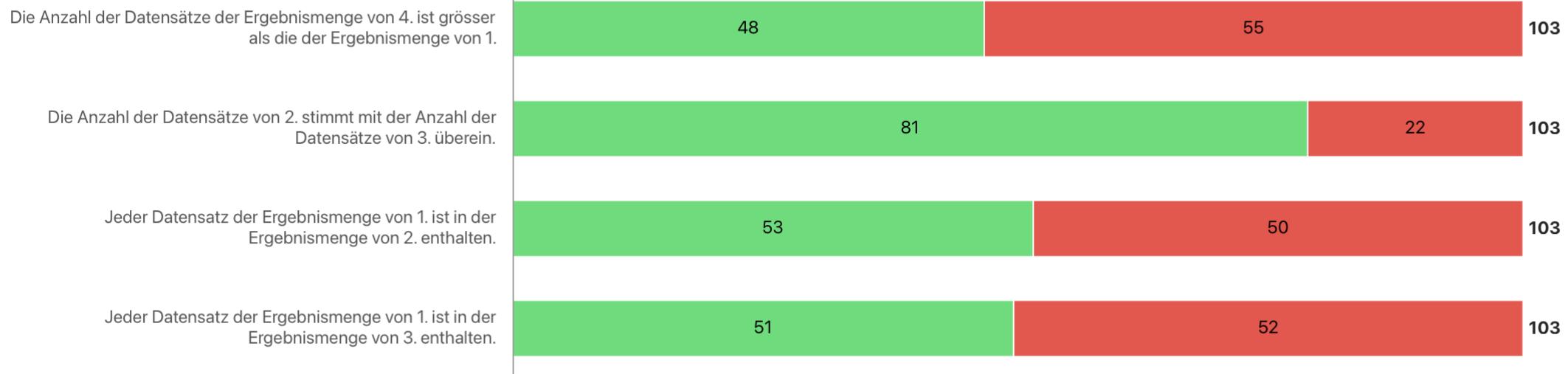
Es gilt: alle Statements haben dieselben Attribute im SELECT und dieselbe Bedingung im ON.

Welche Aussagen sind wahr für die jeweiligen Ergebnismengen:





## Fragestellung 2 in APEX



# Erste Erweiterung: kprime und mtf

Markieren Sie den gewünschten Fragetyp ×

-  Kprim (ETH)
-  Lückentext (Cloze)
-  Lückentextauswahl
-  MTF (ETH)

Wählen Sie einen Fragetyp, um  
seine Beschreibung zu sehen.

Problem:  
Benötigen andere Daten aus anderen Quelltabellen

# Erste Erweiterung: kprime und mtf

- Analyse des Datenmodells in der Quelle
- Export der Daten, Analyse im Excel
- Anpassungen in allen Schichten des DWH

# Stolpersteine

1. Mehrere Versionen von Daten
2. Staging neue Tabelle
3. Cleansing: Tabellen mit mehr Attributen
4. Core: Neue Satelliten, Hubs und Links
5. Mart: weitere Marts

Wie kann man das am einfachsten anpacken

# Stolpersteine

- 1. Mehrere Versionen von Daten
  - 2. Staging neue Tabelle
  - 3. Cleansing: Tabellen mit mehr Attributen
  - 4. Core: Neue Satelliten
  - 5. Mart: weitere Marts
- 
- 1. Attribute können NULL sein

# Stolpersteine

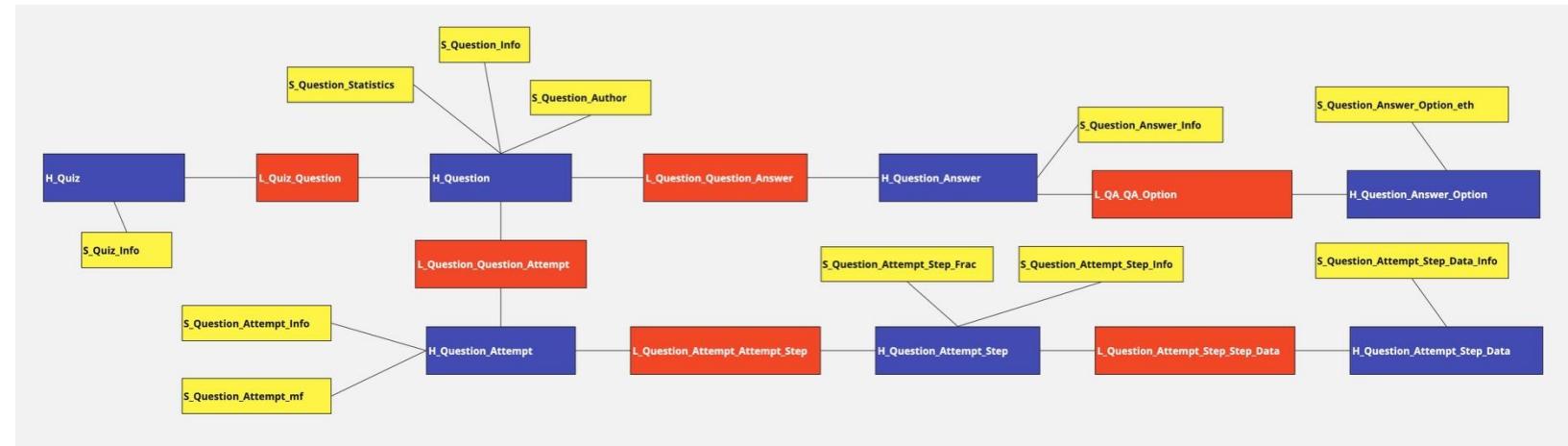
1. Mehrere Versionen von Daten
  2. Staging neue Tabelle
  3. Cleansing: Tabellen mit mehr Attributen
  4. Core: Neue Satelliten
  5. Mart: weitere Marts
2. In Staging sind immer nur Daten vom aktuellen Load
- Je Version eine Tabelle
- View, die alle Tabellen mit UNION zusammenführt
- Code Versionsabhängig

# Stolpersteine

1. Mehrere Versionen von Daten
  2. Staging neue Tabelle
  3. Cleansing: Tabellen mit mehr Attributen
  4. Core: Neue Satelliten
  5. Mart: weitere Marts
3. In Cleansing sind immer nur Daten vom aktuellen Load
  - Tabellen bekommen mehr Attribute
  - Neue Tabellen
  - Code Versionsabhängig

# Stolpersteine

1. Mehrere Versionen von Daten
2. Staging neue Tabelle
3. Cleansing: Tabellen mit mehr Attributen
4. Core erweiterbar, da Data Vault
5. Mart: weitere Marts



# Flexible Lösungen

1. View für Staging
2. Daten in Cleansing immer löschen
3. Core als Data Vault
4. Views auf Core
5. IF (NOT) EXISTS in Skripten

# Was könnte als nächstes kommen ?

1. Auswertung Multiple Choice ✓
2. Erreichte Punkte
3. Erreichte Punkte in Korrelation zu Dauer
4. Mehr als eine Prüfung exportieren
5. User Management

## Fazit

1. Pilot bringt Mehrwert und kommt an APS und HSI zum Einsatz
2. Einfach ausbaubar
3. Klare Konzepte sind flexibel

# Übersicht

