Fachhochschule Nordwestschweiz

Pilot Moodle-DWH

Behind the scenes of a DWH in the Oracle Cloud

Technical implementation from the bucket to the dashboard

Dr. Andrea Kennel, fhnw

29. Mai 2025

Fachhochschule Nordwestschweiz

Dr. Andrea Kennel

Consultant

Lecturer for Databases

Coach for Project Management

University of Applied Sciences Northwestern Switzerland

Brugg/Windisch, Switzerland

andrea.kennel@fhnw.ch
andrea@infokennel.ch
www.infokennel.ch

Motivation

- Students write exams in Moodle
- We want to improve the quality of examinations and examination questions
- The analysis options in Moodel are limited

Moodle: Time analysis

Antworten-Rückblick

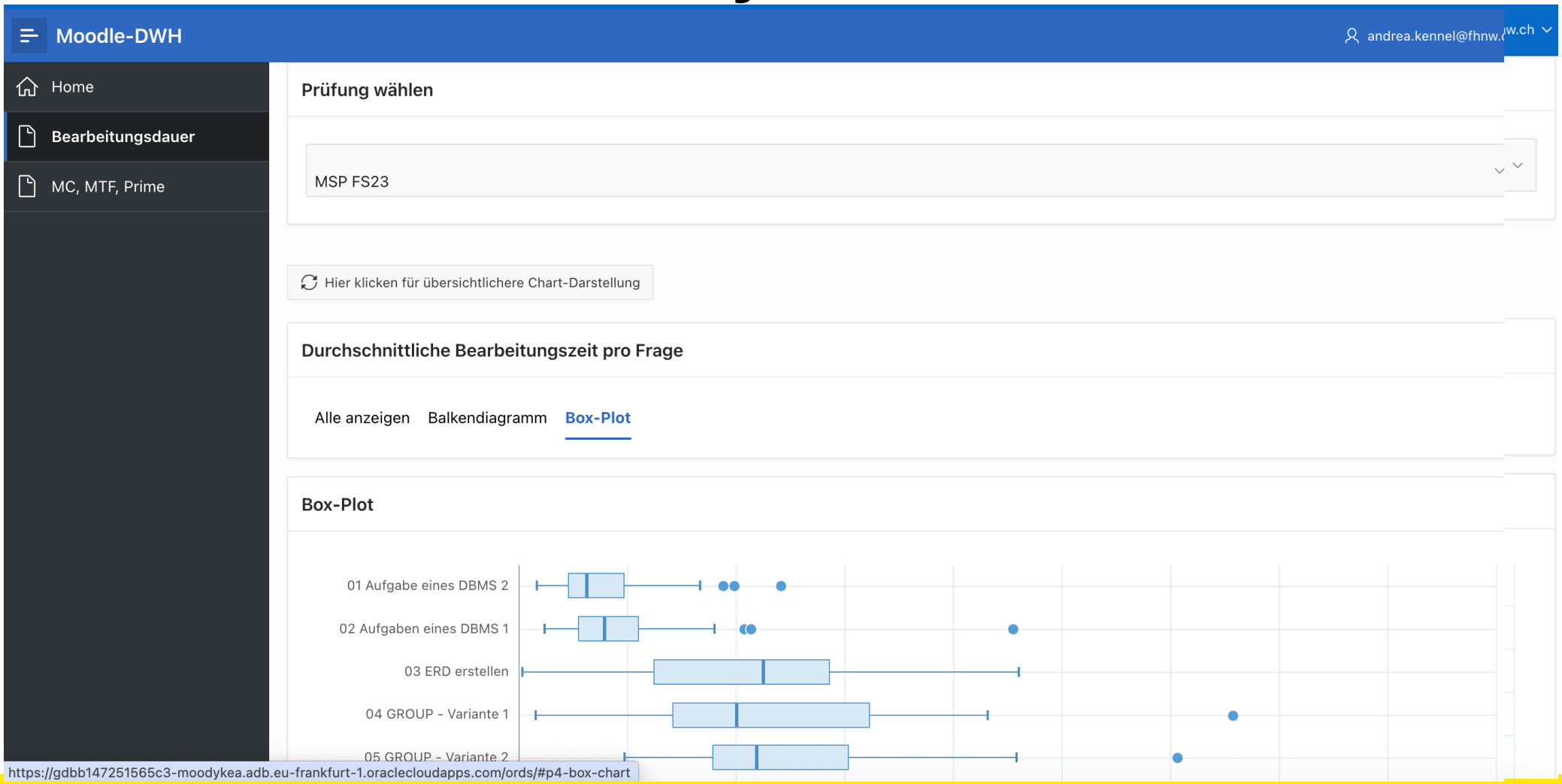
Schritt	Zeit	Aktion
1	3. März 2025, 13:23:04	
2	3. März 2025, 13:25:25	
3	3. März 2025, 13:27:43	
4	**3. März 2025, 13:34:53**	

Antworten-Rückbli

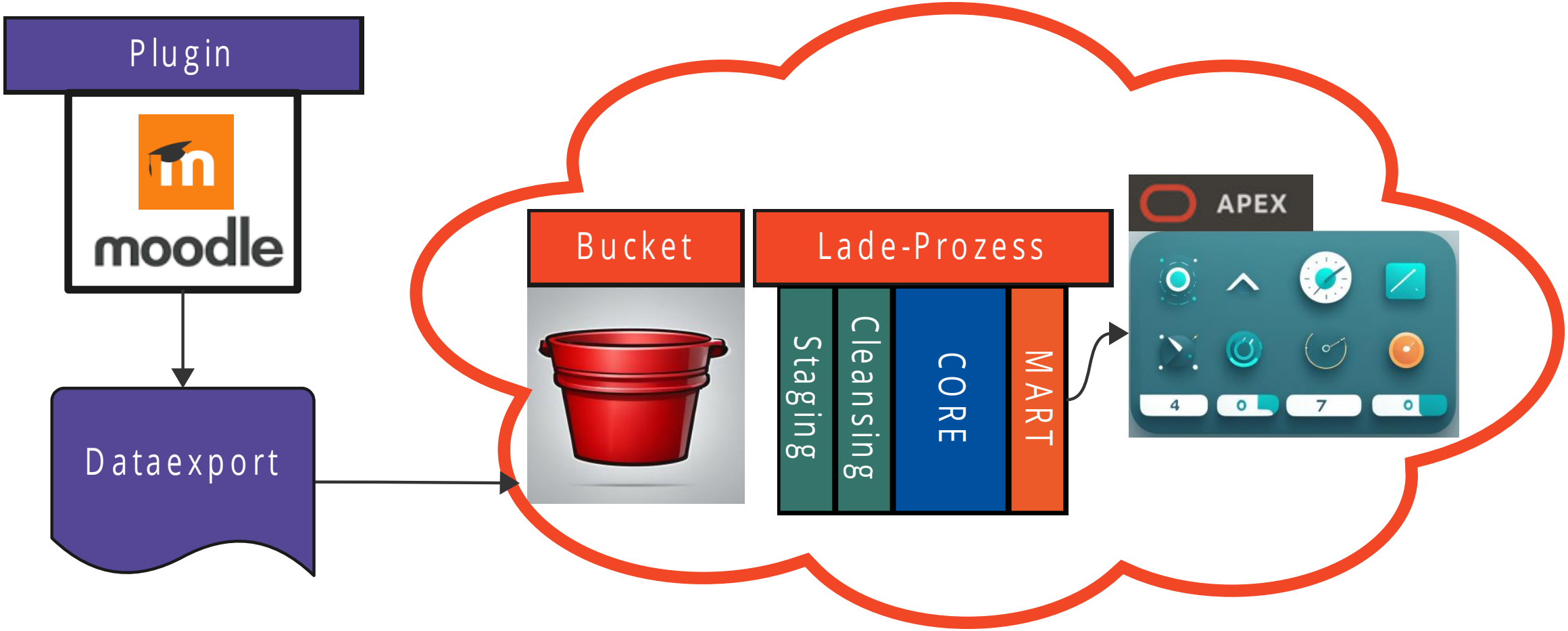
Schritt	Zeit
1	3. März 2025, 13:23:04
2	3. März 2025, 13:27:12
3	**3. März 2025, 13:34:53**

Was		Wann	Zeit
Start		13:23:04	
Weg von	1	13:25:25	00:02:21
Weg von	2	13:27:12	00:01:47
Weg von	1	13:27:43	00:00:31
Weg von	3	13:30:47	00:03:04
	1	00:02:52	
	2	00:01:47	
	3	00:03:04	

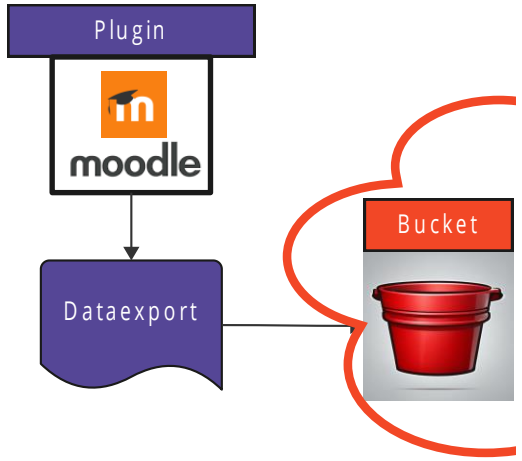
Moodle DWH: Time analysis

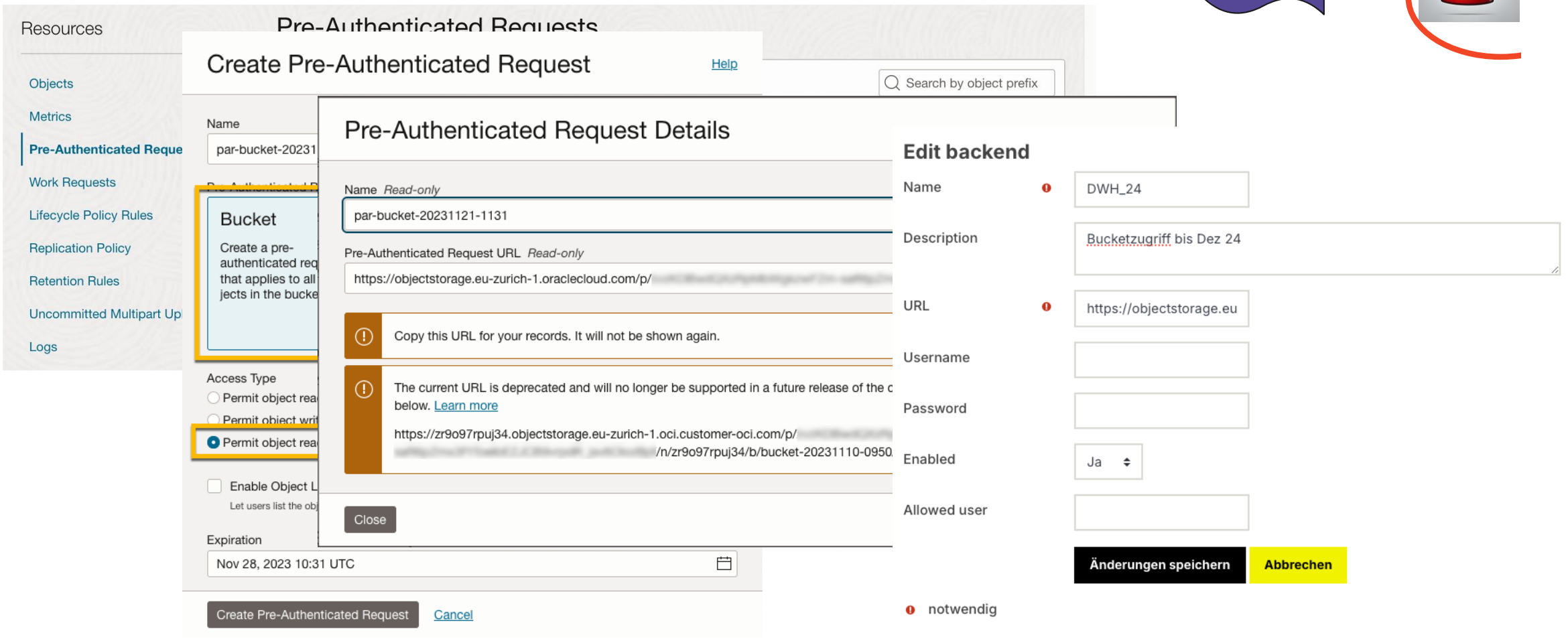


Overview

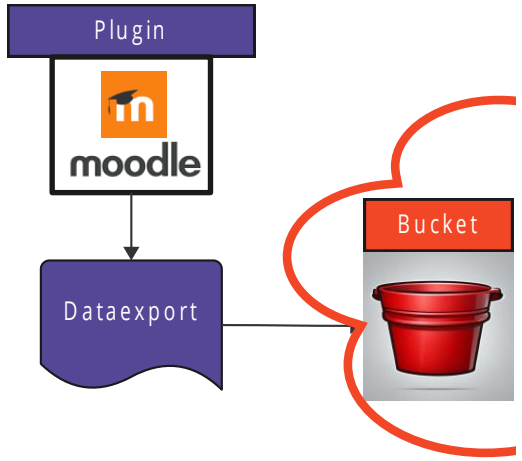

Plugin
moodle
Dataexport
Bucket
Lade-Prozess
Staging
Cleansing
CORE
MART
APEX
4
0
7
0

From Moodle to the bucket



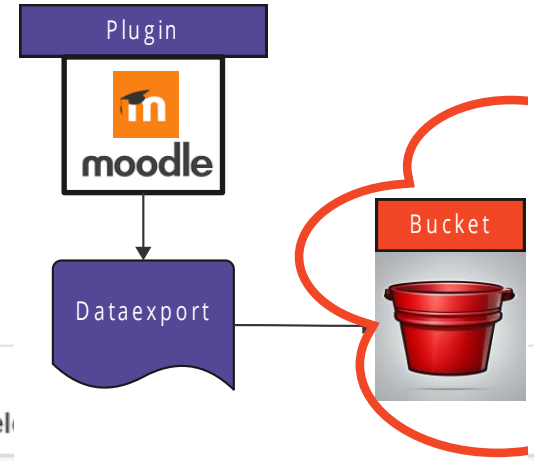


From Moodle to the bucket with PHP-function cURL_exec




```
// PUT to a Pre-Authenticated Requests enabled Oracle Object Storage Bucket.
$url = $DB->get_field('report_datawarehouse_bkends', 'url', ['id' => $backendid]);
// Initiate cURL object.
$curl = curl_init();
// Set your URL.
curl_setopt($curl, CURLOPT_URL, $url . $filename);
// Indicate your protocol.
curl_setopt($curl, CURLOPT_PROTOCOLS, CURLPROTO_HTTPS);
// Set HTTP method to PUT.
curl_setopt($curl, CURLOPT_PUT, 1);
// Indicate the file you want to upload.
curl_setopt($curl, CURLOPT_INFILE, fopen($tempfolder . '/' . $filename, 'rb‘));
. . .
// Execute.
curl_exec($curl);
```

Data protection



User is double anonymised

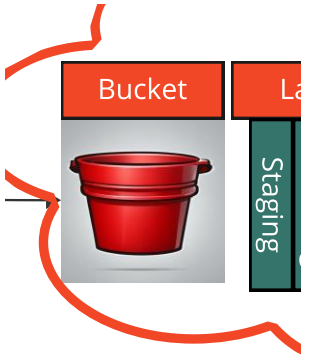
		Nachname / Vorname	Anmel	
☐	aa	anonlastname121 anonfirstname121 Versuch überprüfen	anon121	anon121@doesntexist.invalid
☐	aa	anonlastname68 anonfirstname68 Versuch überprüfen	anon68	anon68@doesntexist.invalid

1. Anonymise users in Moodle

2. Export hash only

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

3. Data ‘knocked flat’ as csv into the bucket



From the bucket to the staging with 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
```

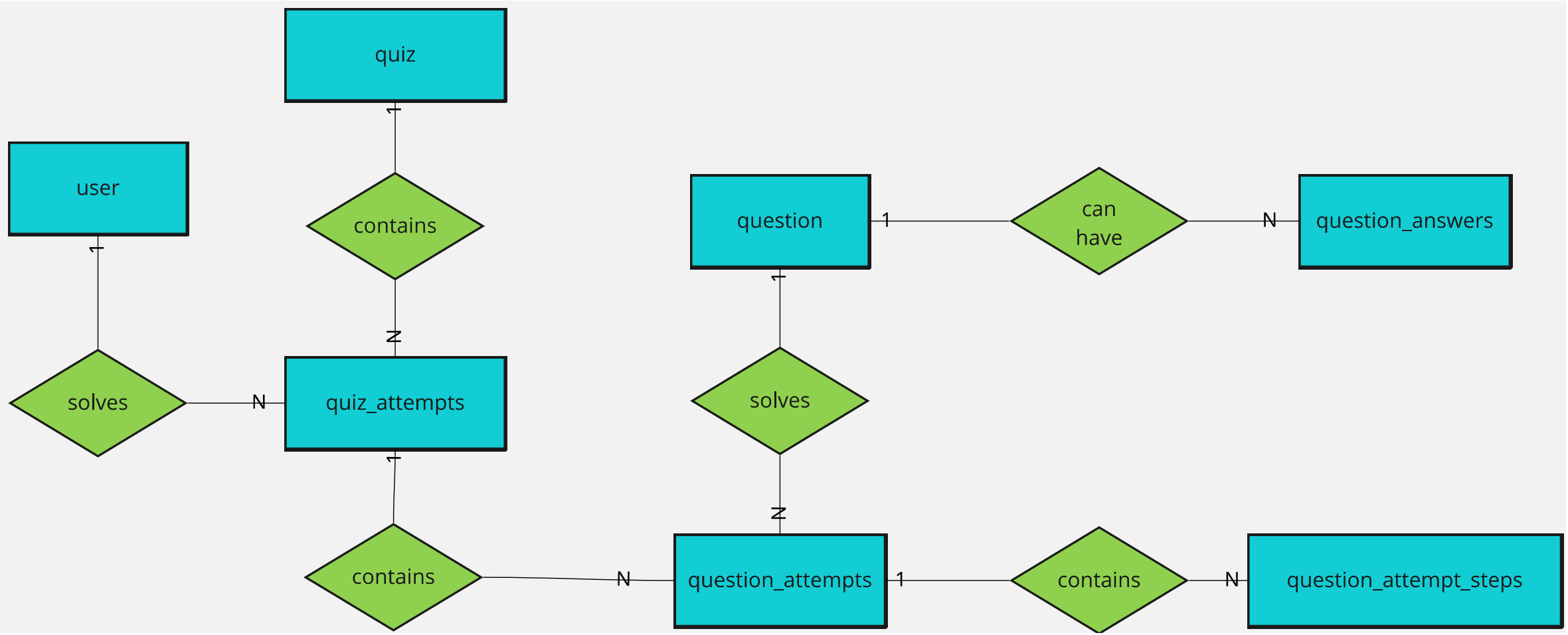
Data from CSV 1:1 into a table

The data model of the source

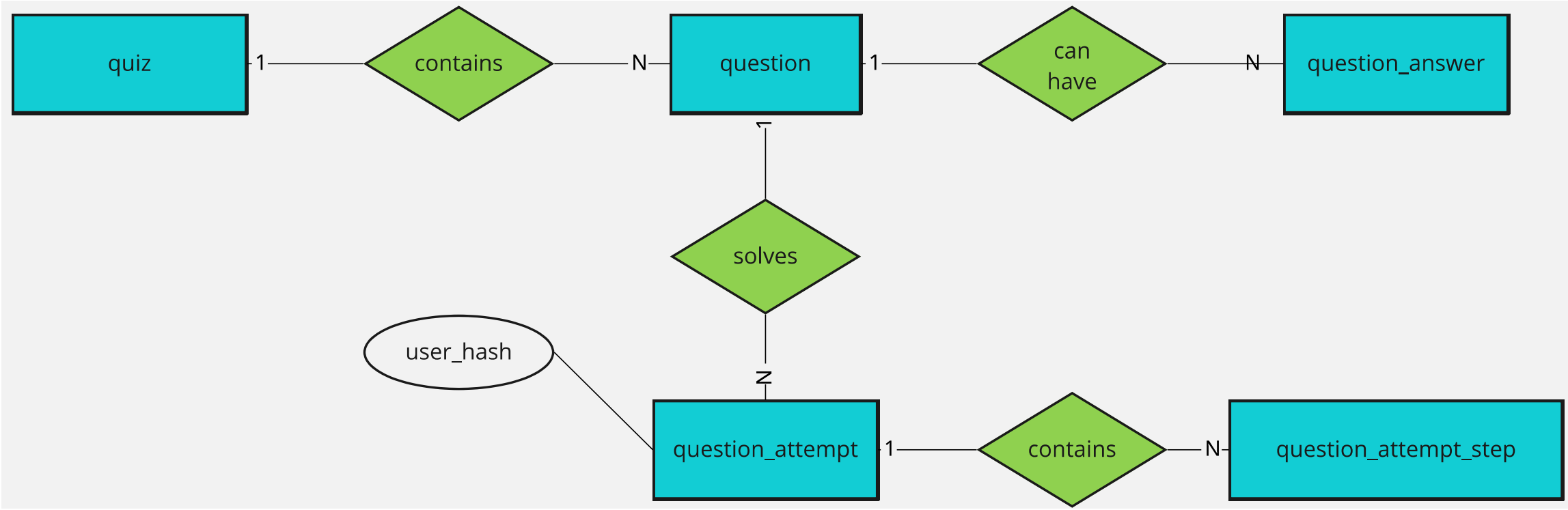
The Moodle data model is open to the public, but complex and large

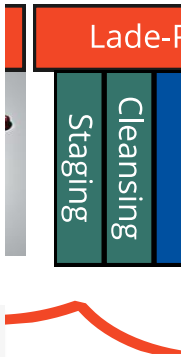
https://www.examulator.com/er/4.0/tables/quiz.html

The source data model (extract)



The model in the cleansing



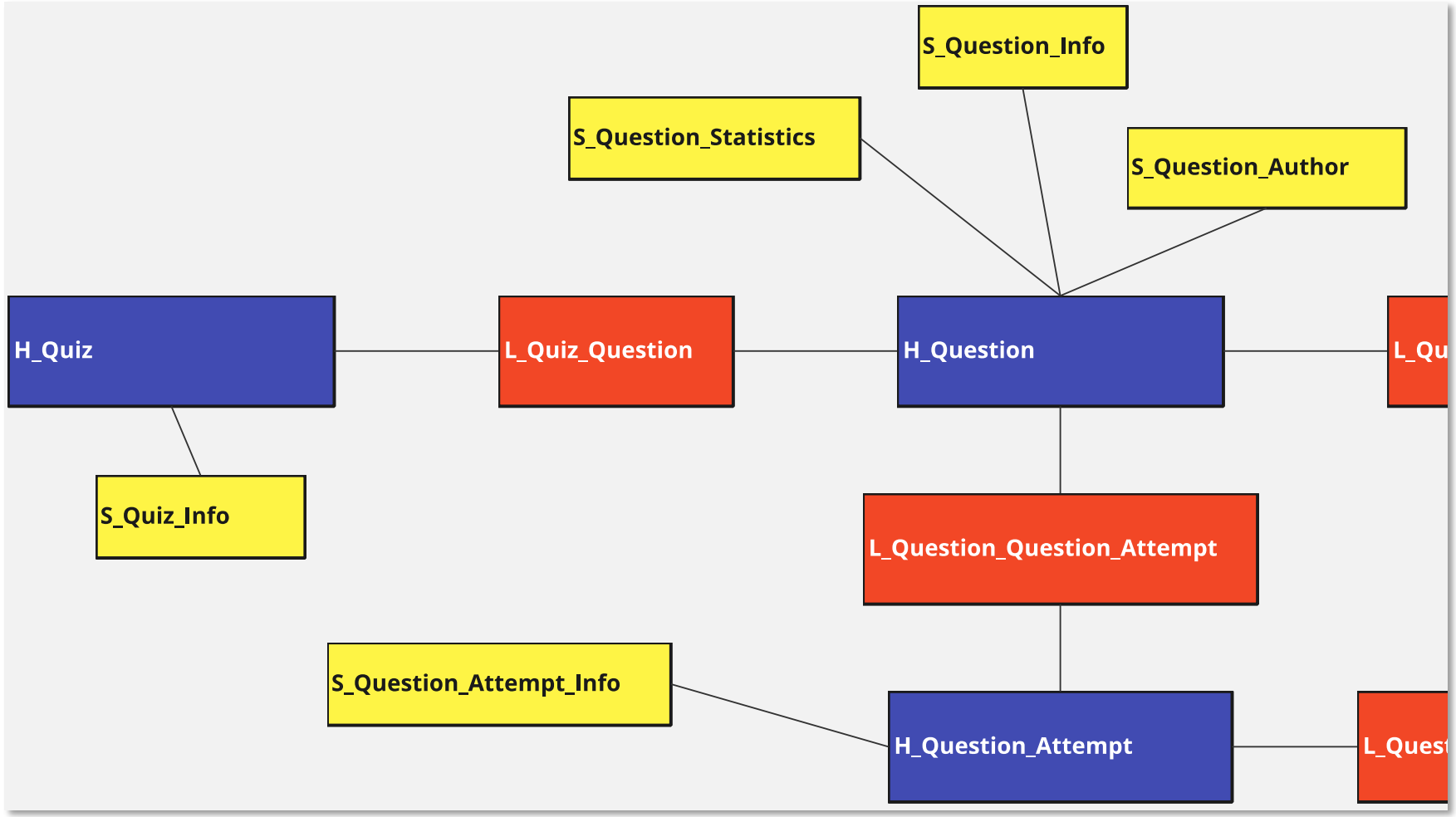


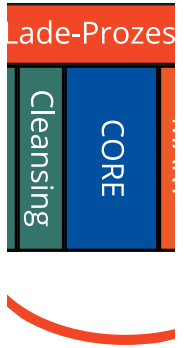
From the staging to the cleansing

```
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;
```

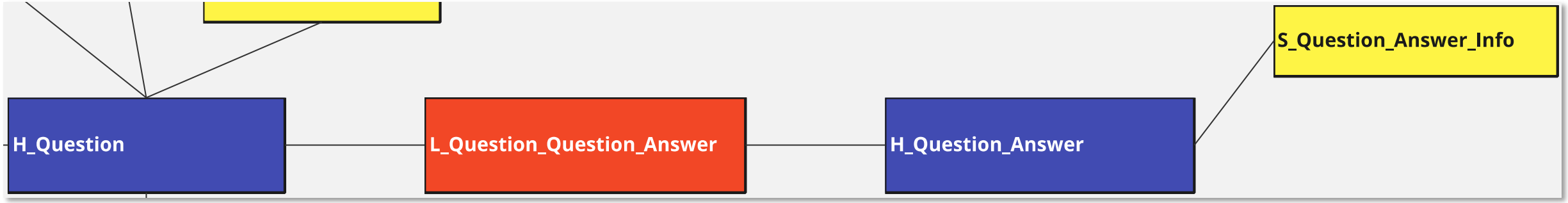
```
INSERT INTO MD_CLEANSING.question_attempt_step_data (
    load_id,
    sourcesystem,
    authorizedusers,
    question_attempt_step_data_id,
    question_attempt_step_id,
    step_data_name,
    step_data_value)
SELECT DISTINCT
  p_load_id load_id,
  sourcesystem sourcesystem,
  authorizedusers authorizedusers,
  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;
```

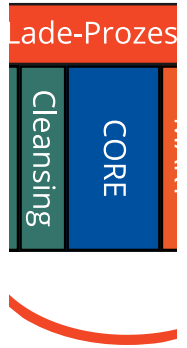
The model in the core



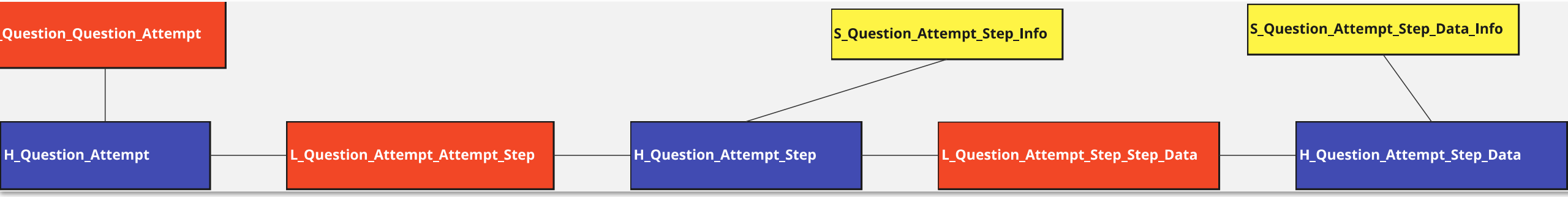


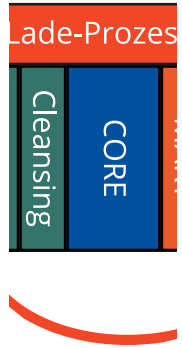
The model in the core



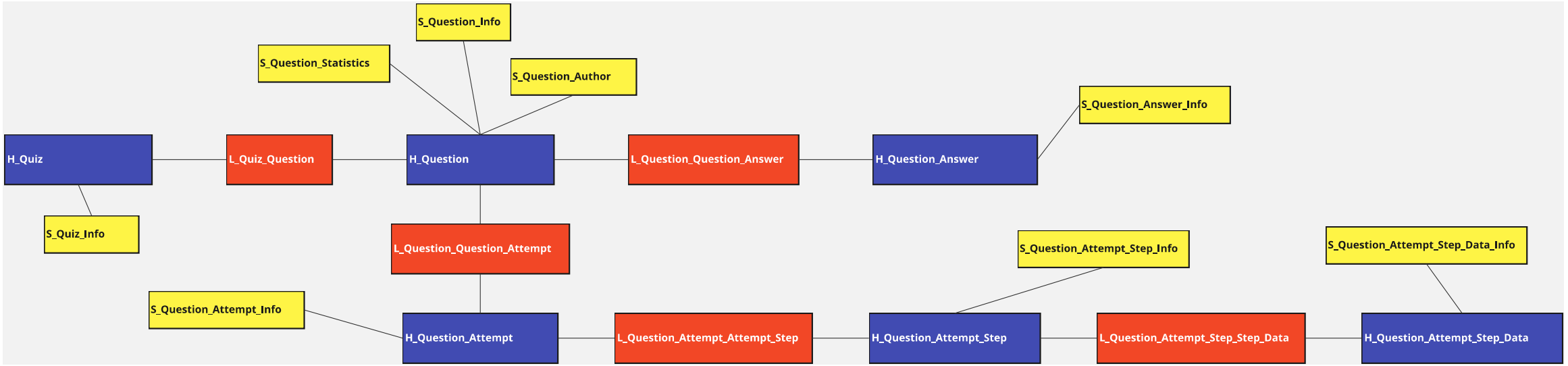


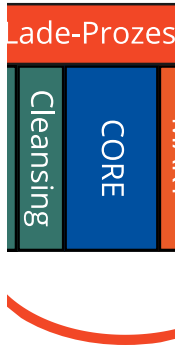
The model in the core





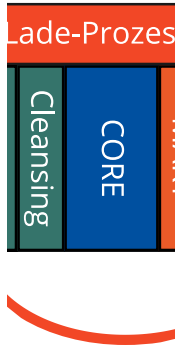
The model in the core





From the cleansing to the core

1. Fill HUB	H_Question
Surrogate key for HUB	h_question_sid
Srouce keys	question_id load_id

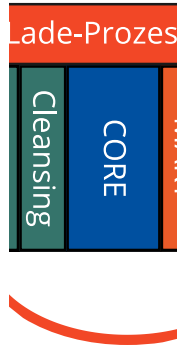


From the cleansing to the core

2. Fill Satellites for HUB — S_Question_info

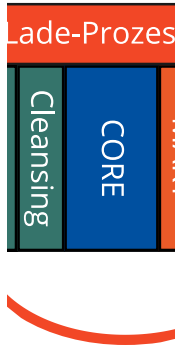
Find foreign key to HUB via source key — h_question_sid

```
SELECT
  h.h_question_sid,
  cl.question_type,cl.question_name,
  cl.question_text
FROM MD_CLEANSING.question cl  INNER JOIN
     H_Question h ON (cl.question_id = h.question_id
                      AND cl.load_id = h.load_id);
```



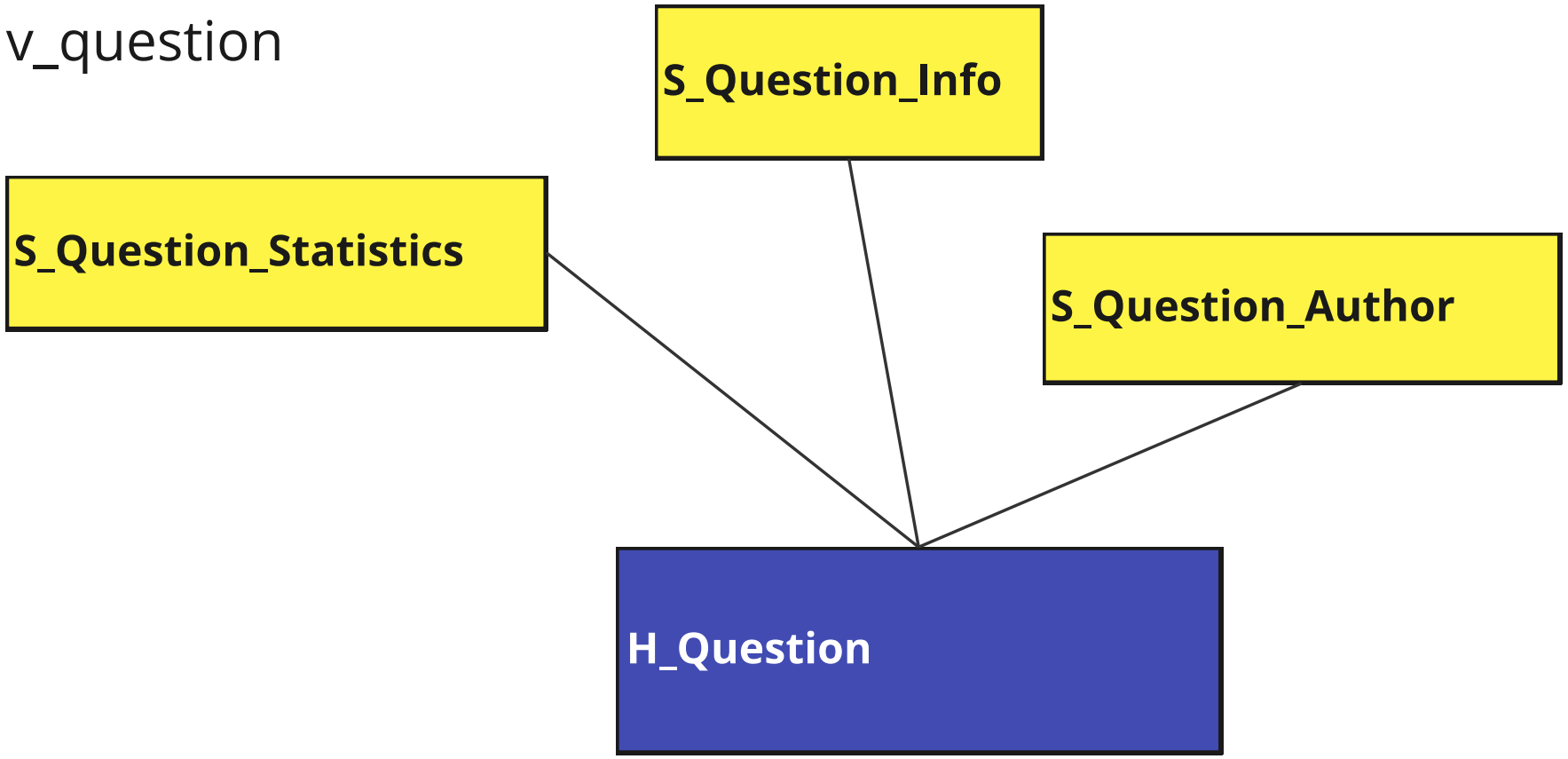
From the cleansing to the core

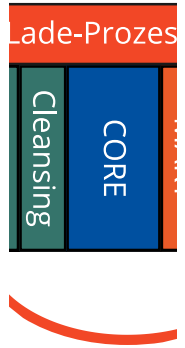
3. Fill Link between HUB	L_Question_Question_Answer
Surrogate key for Link	l_question_question_answer_sid
Find both foreign keys to HUB via source key	h_question_sid h_question_answer_sid



Views for easier access to core

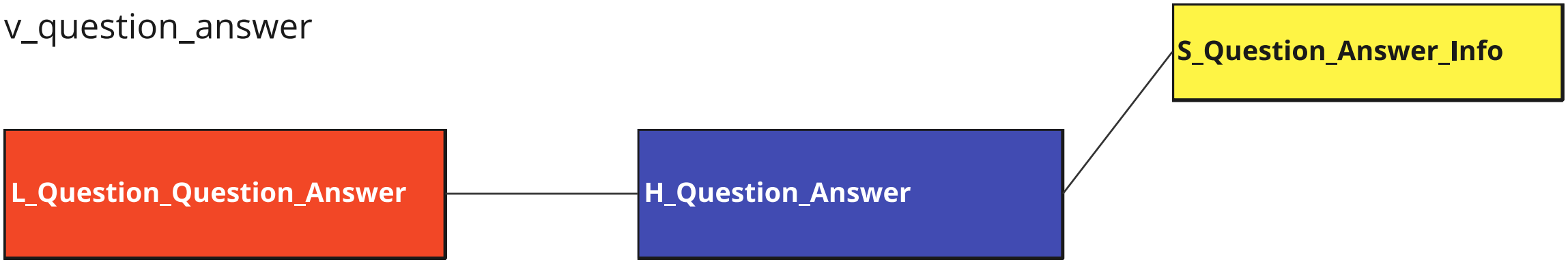
v_question

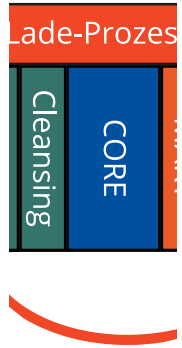




Views for easier access to core

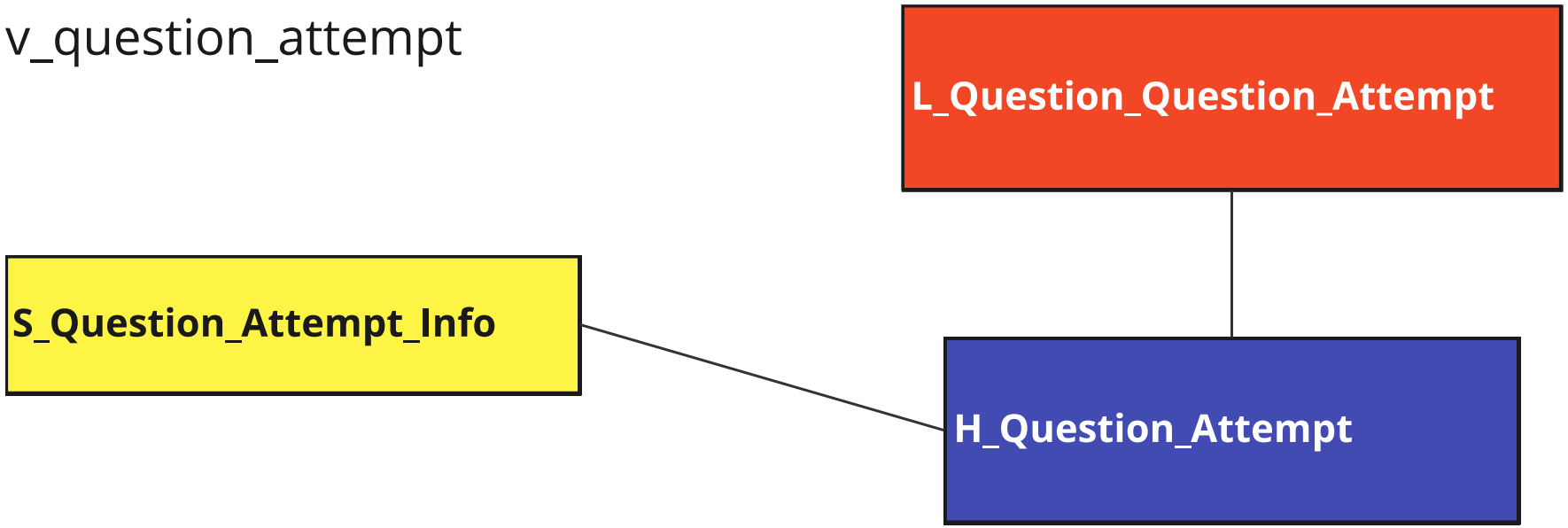
v_question_answer





Views for easier access to core

v_question_attempt



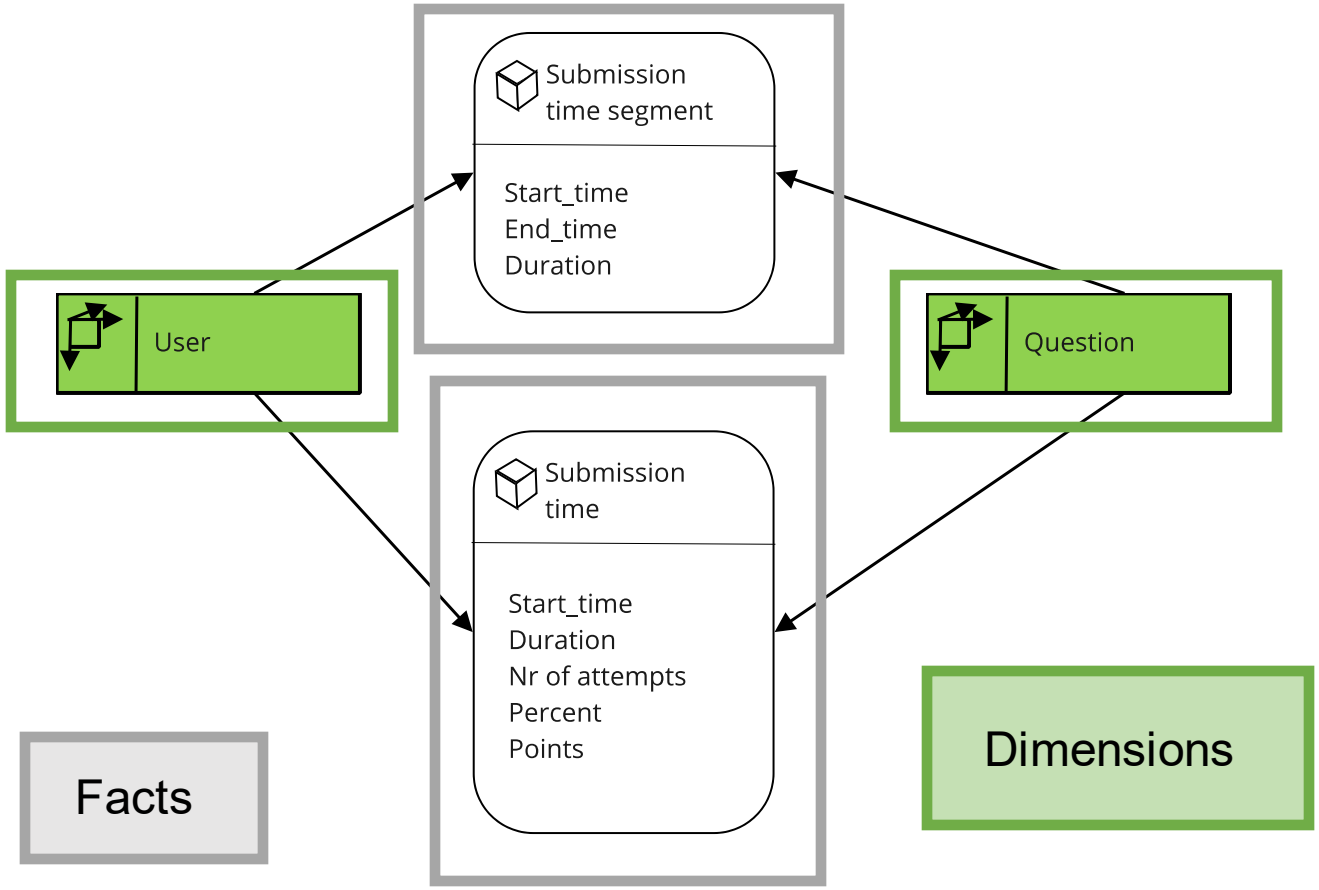
MART Layer

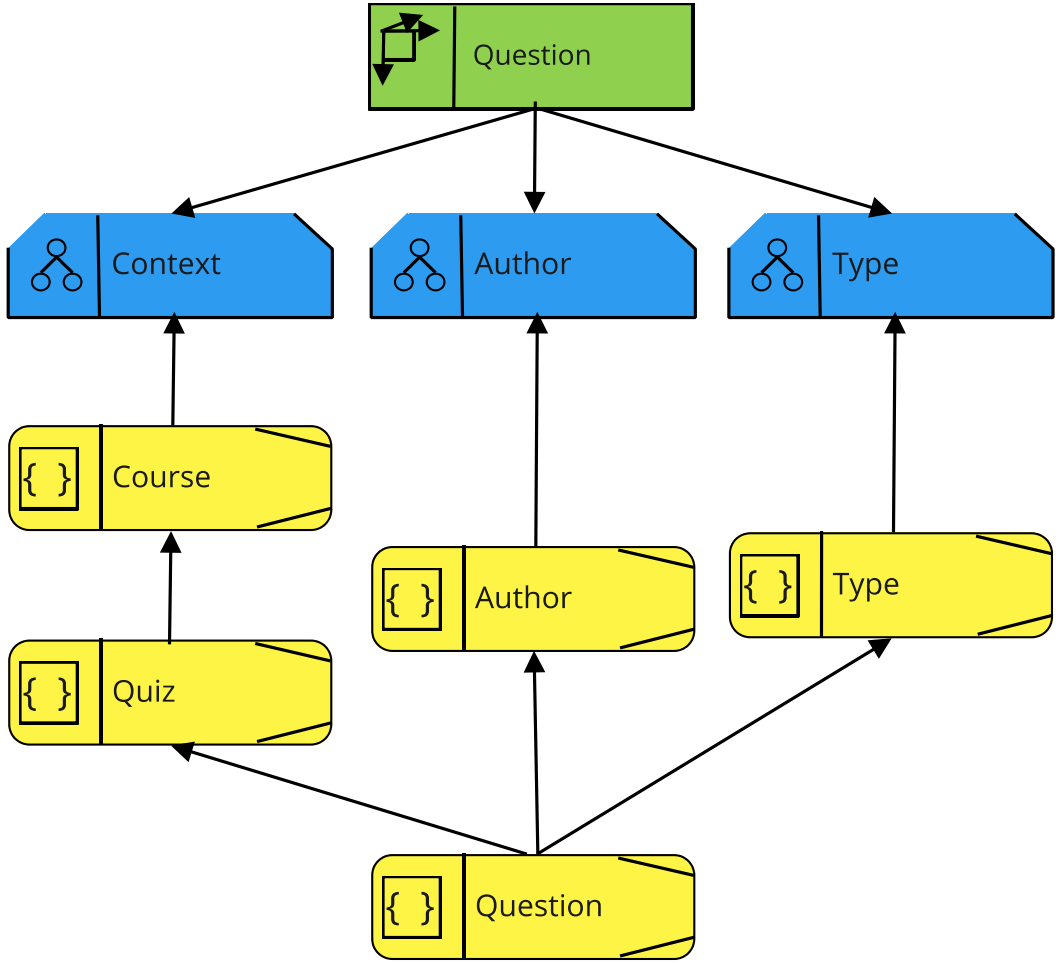
Purpose of the mart: Provide data for analyses

- Different approach than in the core: Tailored to the needs of end users.
- Different concepts and conventions for modelling

Bilder: Adobe Stock

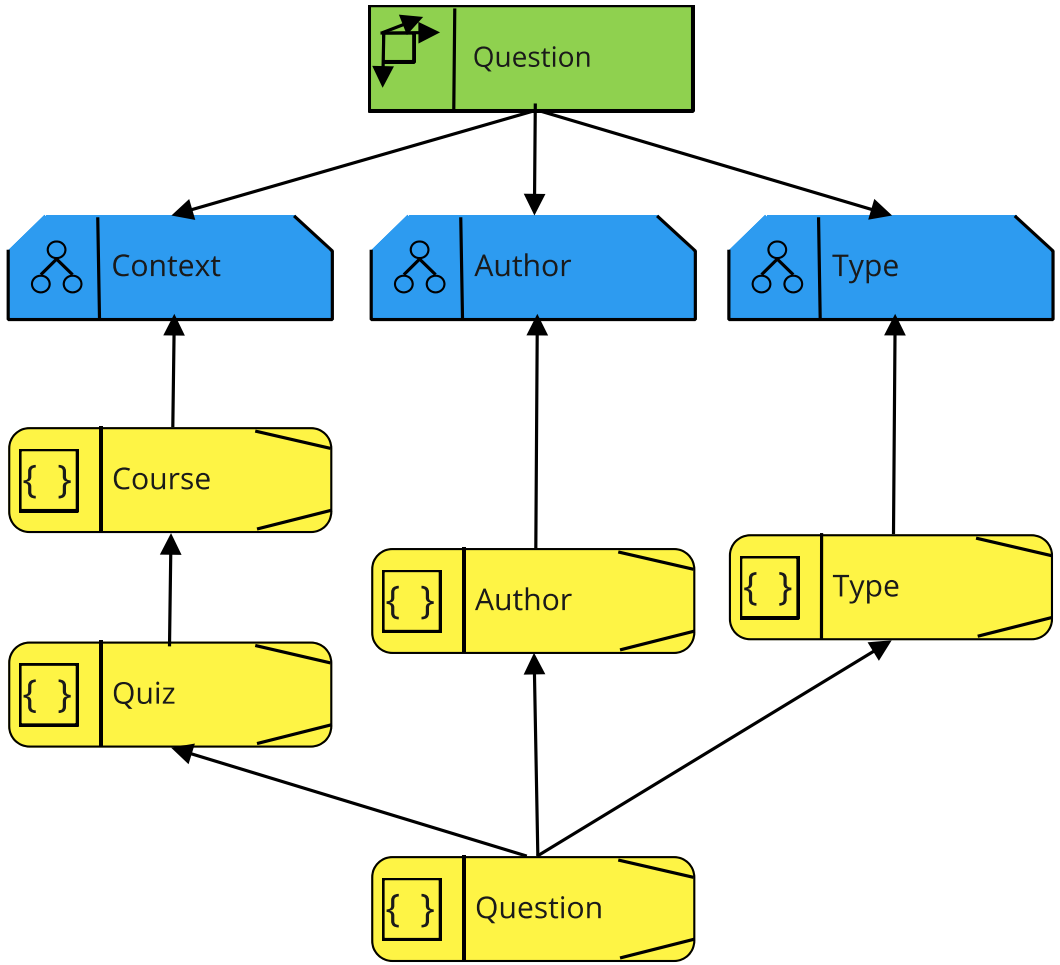
Time analysis: ADAPT modelling



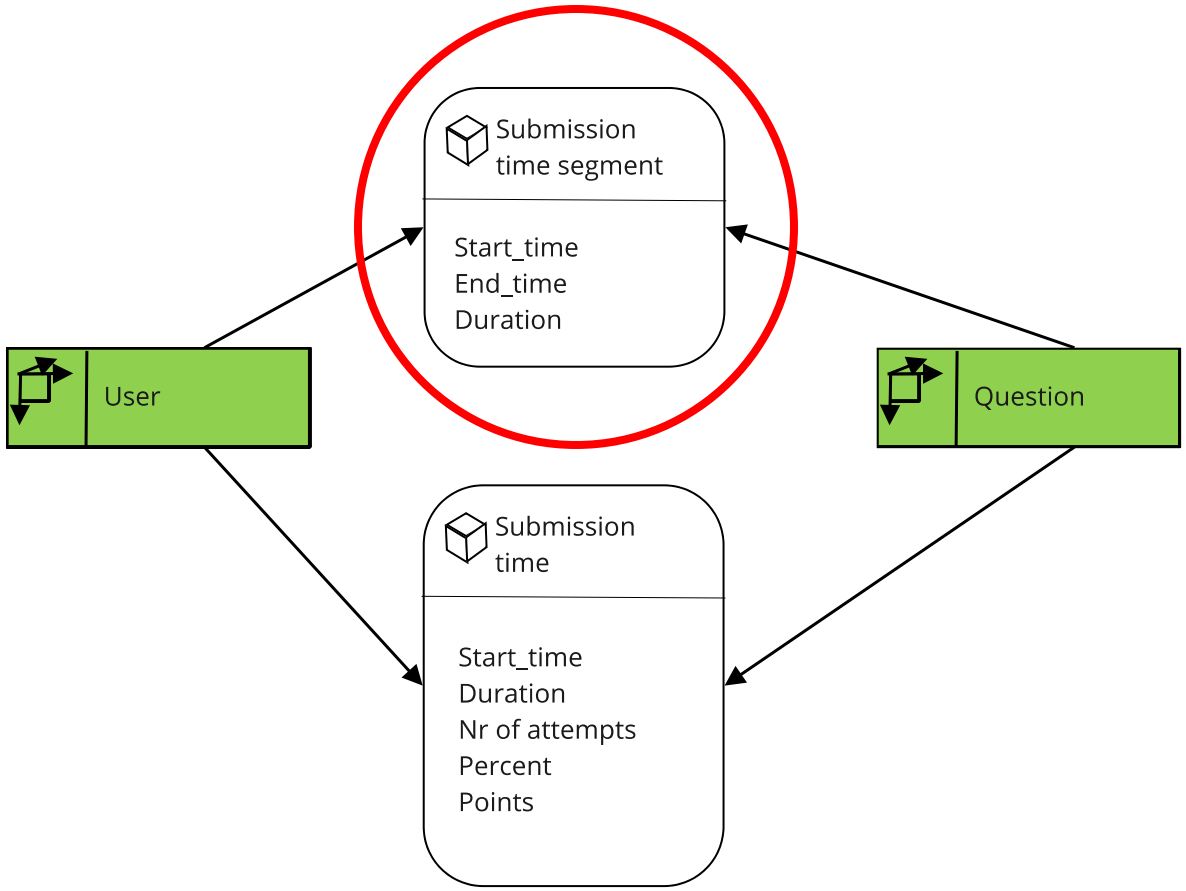


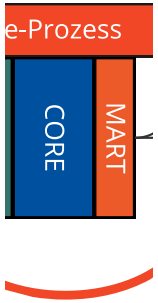
From the core to the mart – Question

```
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;
```



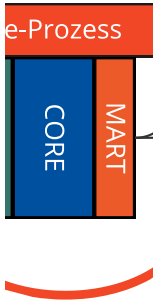
From the core to the mart – Submission time segment





From the core to the mart – Submission time segment

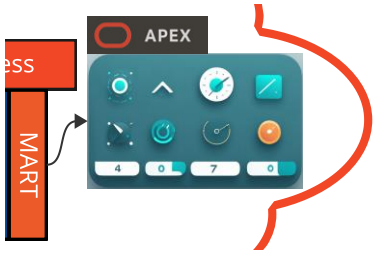
```
INSERT INTO MD_MART.DM_F_Question_Submission_Time_Segment
SELECT
  MD_ID_Question,
  quiz_attempt_user_hash MD_ID_Candidate,
  time_prev start_time,
  createdunixtime end_time, time_spent duration, load_id
FROM (
  SELECT
    d_q.MD_ID_Question,
    qa.quiz_attempt_user_hash, …
    LAG(step.createdunixtime, 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)
```



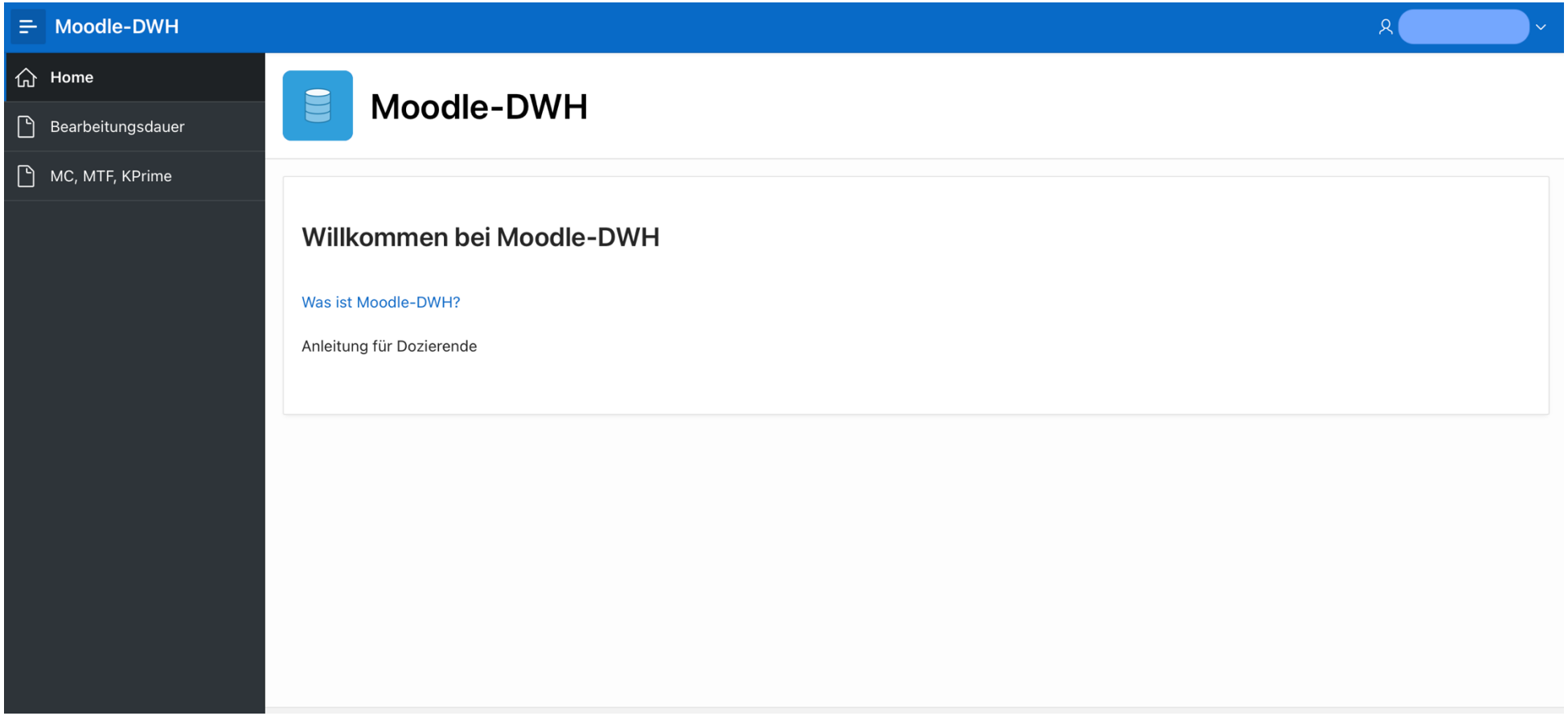
From the core to the mart – Submission time segment

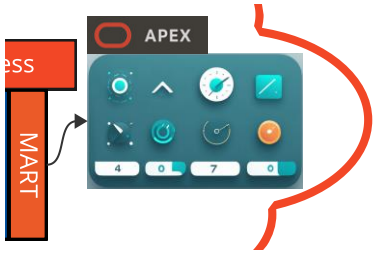
Important: Test data!

Specially created and timed Moodle exam
→ Protocol

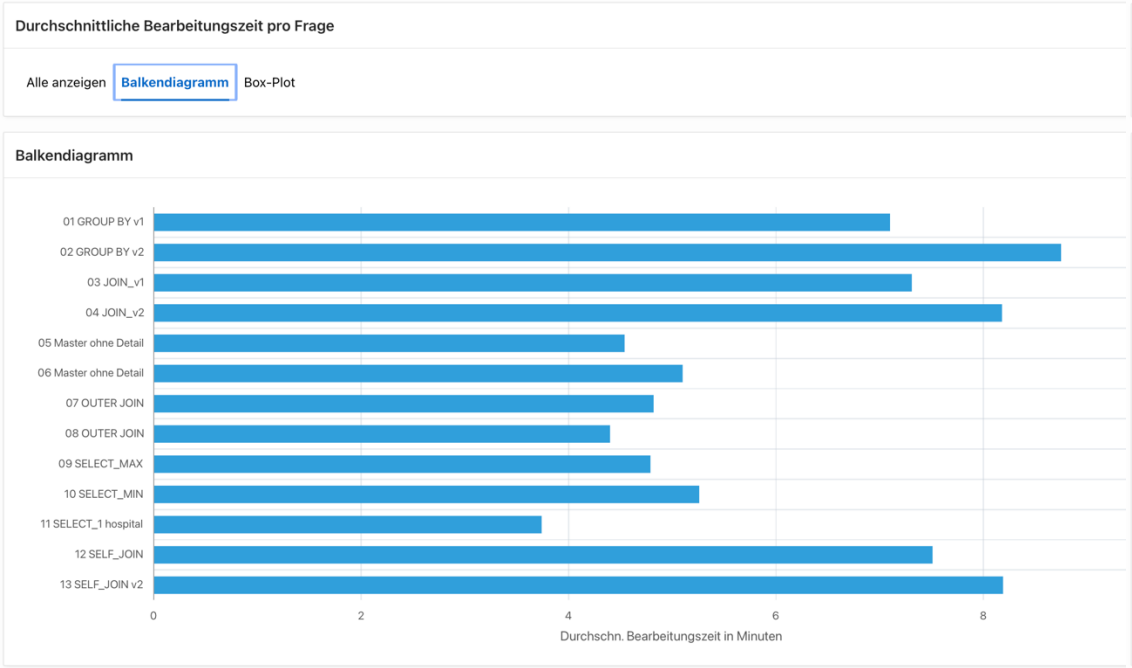


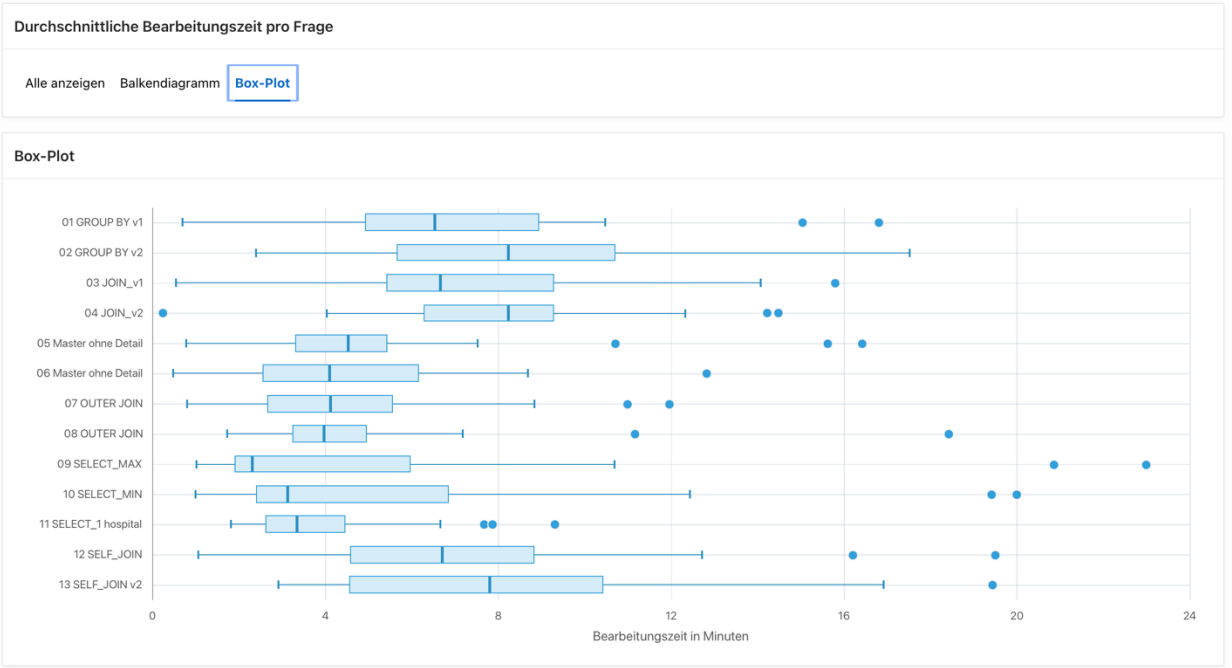
Interface for End User: Oracle APEX

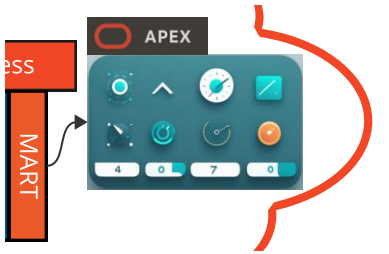




Interface for End User: Oracle APEX







From the core to the mart – MC answer option

Choise of answer option

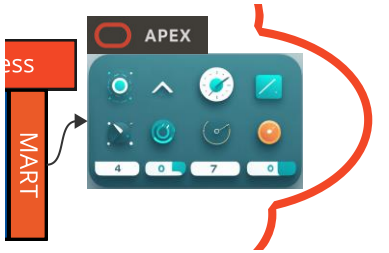
STEP_DATA_NAME	STEP_DATA_VALUE
choice0	1
choice1	1
choice2	0
choice3	0
choice4	0
_order	680,684,681,683,682

???

ChatGPT!

Answer option

QUESTIONANSWERSID	ANSWER
683	SELECT gruppen_name <p d
681	<p
684	SELECT ig.gruppen_name <p dir="ltr
682	SELECT gruppen_name <p d
680	SELECT i

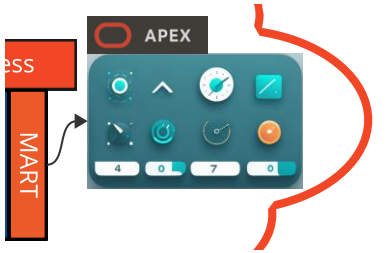


From the core to the mart – MC answer option

```
…
SELECT
        c.step_data_name,
        c.original_step_data_value,

        REGEXP_SUBSTR(o.order_values, '[^,]+', 1,

        CAST(SUBSTR(c.step_data_name, 7) AS INT) + 1)

        AS questionanswersid,
…
```

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

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



From the core to the mart – MC answer option

Hints for using ChatGPT:

→ Provide context about the required part of the data model

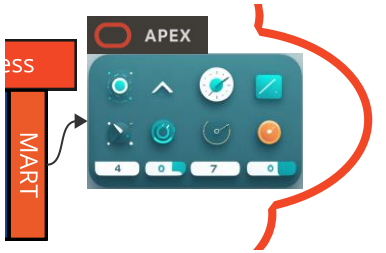
→ Precise wording (this requires that I know exactly what I want)

→ Understanding of the data model

→ Understanding how the data is related

→ Test cases → Response received can be checked

→ Step-by-step approach



MC answer option in APEX

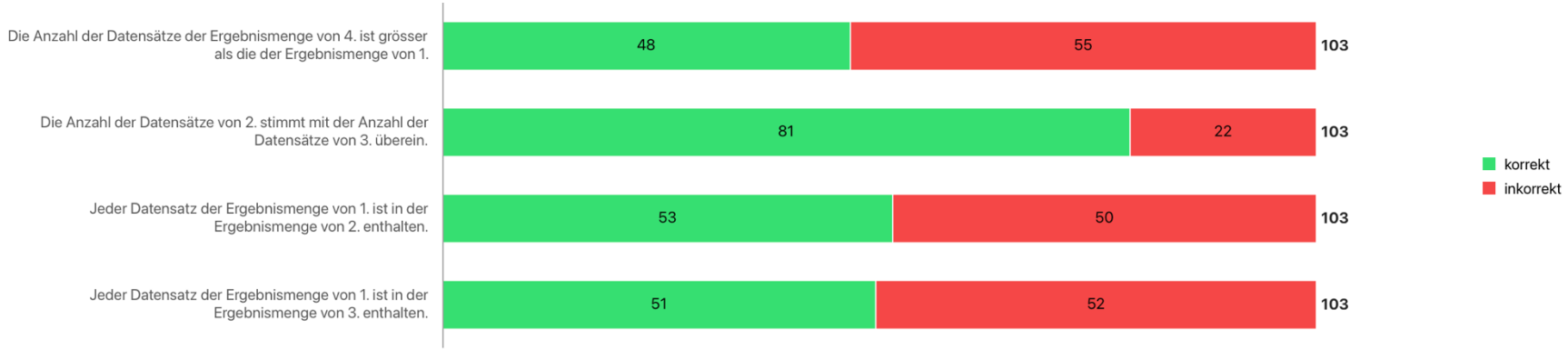
Welche Antwort wurde wie oft korrekt/inkorrekt beantwortet?

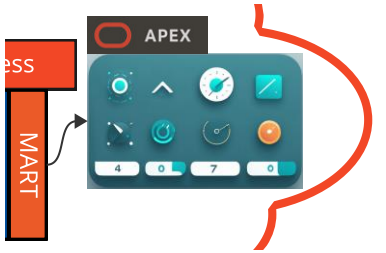
Betrachte die folgenden Statements

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 ...

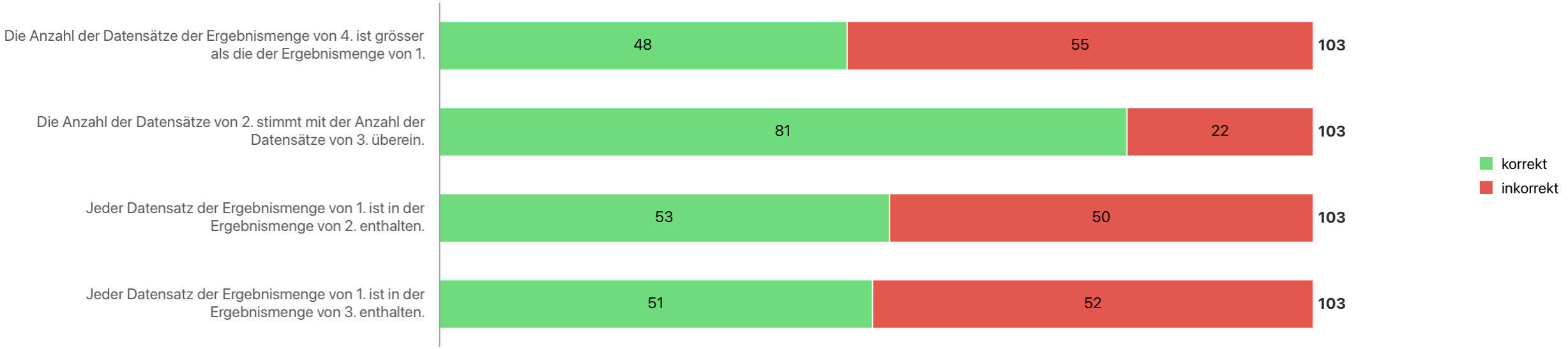
Es gilt: alle Statements haben dieselben Attribute im SELECT und dieselbe Bedingung im ON.

Welche Aussagen sind wahr für die jeweiligen Ergebnismengen:

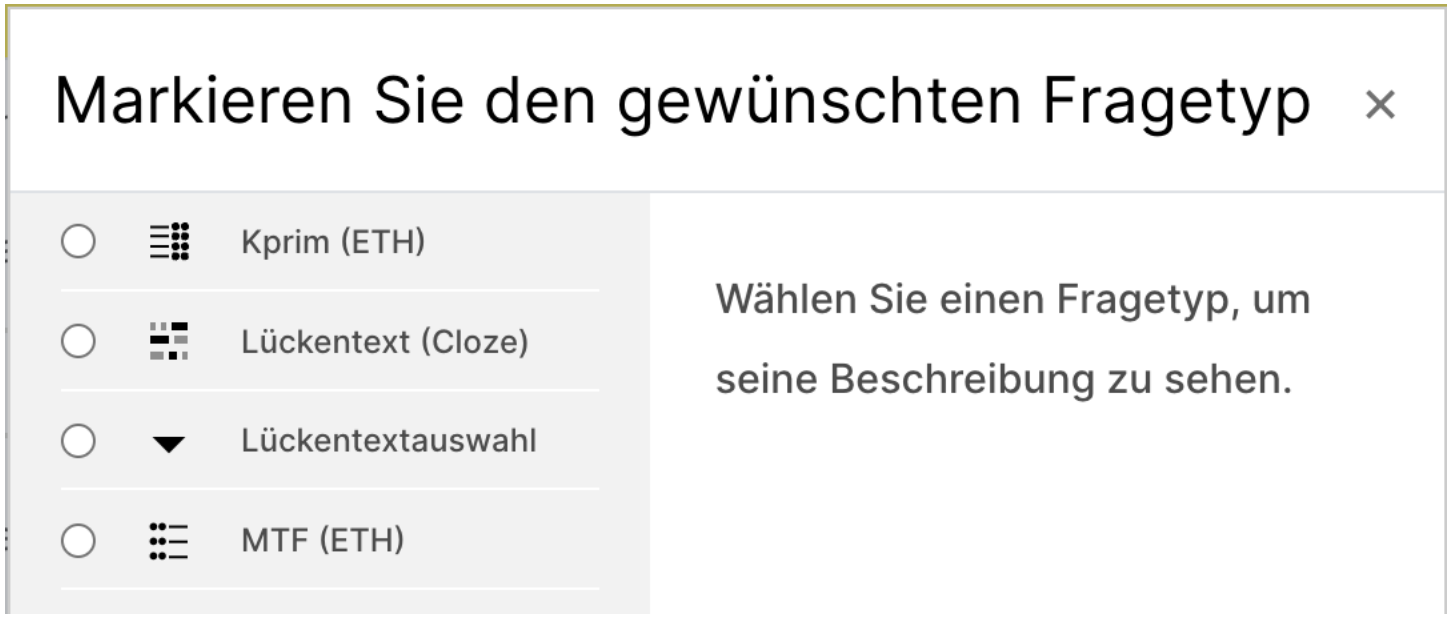




MC answer option in APEX



First extension: kprime and mtf



Problem:
Require other data from other source tables

First extension: kprime and mtf

- Analysing the data model in the source
- Export of the data, analysis in Excel
- Adaptations in all layers of the DWH

Pitfalls

1. Several versions of data extracts
2. Staging: new table
3. Cleansing: tables with more attributes
4. Core: new satellites, hubs and links
5. Mart: more marts

What is the easiest way to tackle this?

Pitfalls

1. Several versions of data extracts
2. Staging: new table
3. Cleansing: tables with more attributes
4. Core: new satellites, hubs and links
5. Mart: more marts

1. Attributes can be NULL

Pitfalls

1. Several versions of data extracts
2. Staging: new table
3. Cleansing: tables with more attributes
4. Core: new satellites, hubs and links
5. Mart: more marts

2. Staging always only contains data from the current load

One table per version

View that merges all tables with UNION

Code version-dependent

Pitfalls

1. Several versions of data extracts
2. Staging: new table
3. Cleansing: tables with more attributes
4. Core: new satellites, hubs and links
5. Mart: more marts

3. Staging always only contains data from the current load

Tables get more attributes

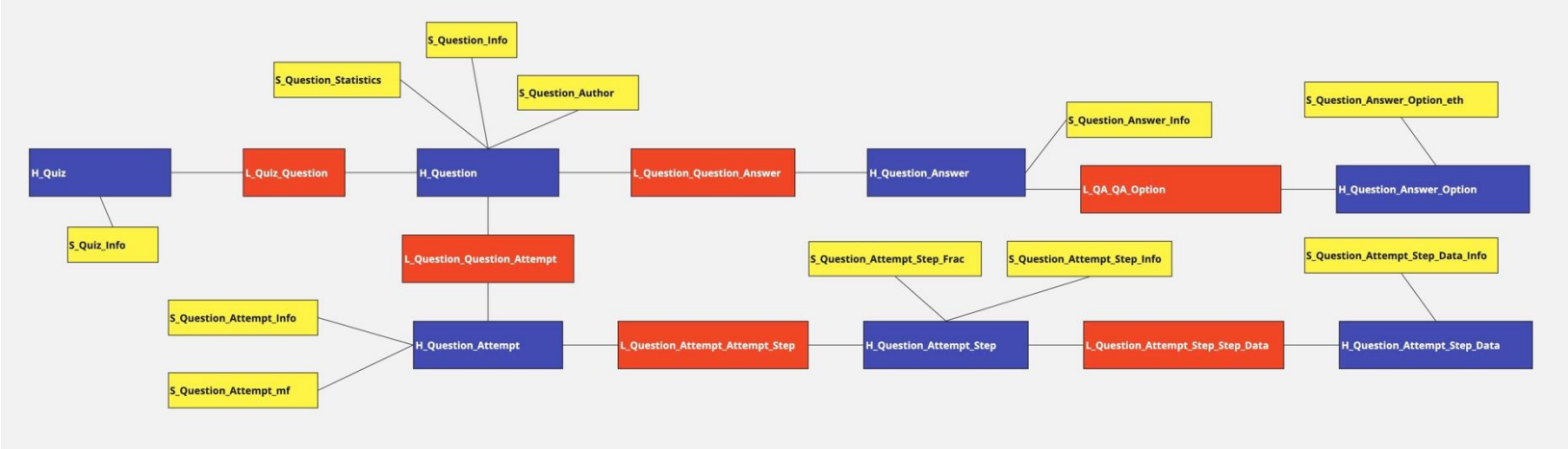
New tables

Code version dependent

Pitfalls

1. Several versions of data extracts
2. Staging: new table
3. Cleansing: tables with more attributes
4. Core: new satellites, hubs and links
5. Mart: more marts

4. Core expandable, because Data Vault



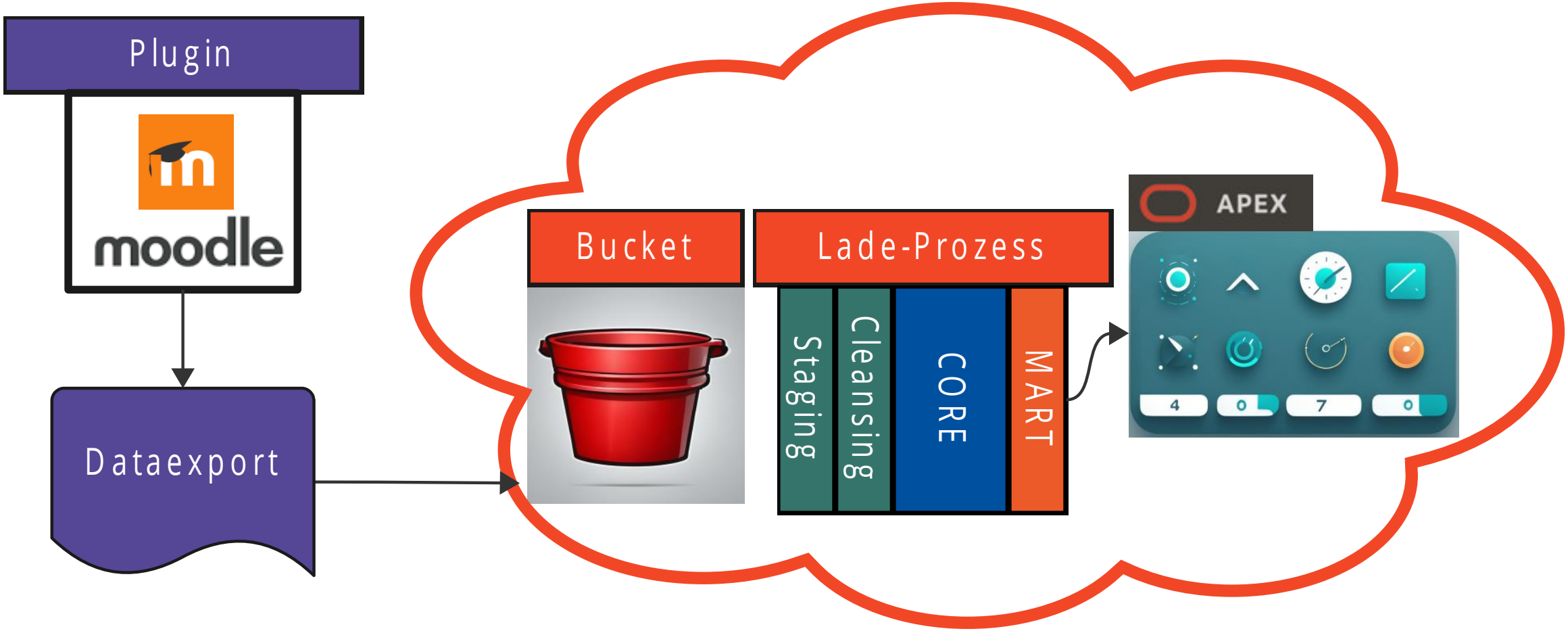
Flexible solutions

1. View for staging
2. Always delete data in cleansing
3. Core as Data Vault
4. Views on core
5. IF (NOT) EXISTS in scripts

What could come next?

1. Evaluation of Multiple Choice ✓
2. Points achieved
3. Points achieved in correlation to duration
4. Export more than one test at a time
5. User Management

Overview


Plugin
moodle
Dataexport
Bucket
Lade-Prozess
Staging
Cleansing
CORE
MART
APEX
4
0
7
0

Conclusion

1. Pilot brings added value
2. Easy to expand
3. Flexible thanks to clear concepts

Fachhochschule Nordwestschweiz

Dr. Andrea Kennel

Consultant

Lecturer for Databases

Coach for Project Management

University of Applied Sciences Northwestern Switzerland

Brugg/Windisch, Switzerland

andrea.kennel@fhnw.ch
andrea@infokennel.ch
www.infokennel.ch