

Data Warehouse Testing ist ...

Data-Warehouse-Testing bezeichnet den **systematischen Testprozess** zur Sicherstellung, dass ein DWH-System:

- **korrekte, vollständige, konsistente und aktuelle Daten** liefert
 - **Geschäftsregeln** korrekt umgesetzt werden
 - **ETL-Prozesse zuverlässig und performant** ablaufen
- Ziel ist die **Vertrauenswürdigkeit und Qualität von Analyse- und Reportingdaten** sicherzustellen.

Was ist ETL?

ETL steht für:

- **Extract** – Daten werden aus verschiedenen Quellen extrahiert (z. B. SQL, Excel, APIs)
 - **Transform** – Daten werden bereinigt, umgewandelt, zusammengeführt
 - **Load** – Daten werden ins Zielsystem (z. B. Data Warehouse) geladen
- ETL ist das **Rückgrat moderner Datenintegration**.

Was ist ETL-Testing?

ETL-Testing überprüft, ob:

- **Daten korrekt extrahiert, transformiert und geladen** wurden
- **Geschäftslogik und Transformationen** fehlerfrei umgesetzt wurden
- **Quell- und Zielsysteme synchron** sind (Datengenauigkeit, -vollständigkeit, -integrität)

Dabei wird getestet auf:

- Datenverlust
- Inkonsistenzen
- Duplikate
- Verfälschte Transformationen

ETL-Testing-Prozess – in 5 Schritten

1. **Verstehen der Anforderungen und Datenflüsse**
(Mapping-Dokumente, Geschäftsregeln, Quell-Ziel-Zuordnungen)
2. **Testplanung & Datenvorbereitung**
(Teststrategie, Testumgebung, Beispiel- und Staging-Daten vorbereiten)
3. **Ausführung von Tests**
(Vergleich von Quell- und Zieltabellen, Hashing, Row Counts, Business Rules prüfen)
4. **Ergebnisanalyse & Fehlerverfolgung**
(Fehler dokumentieren, Ursachen analysieren, Rückmeldung ans Dev-Team)
5. **Abnahme & Regressionstest bei Änderungen**
(Tests automatisieren, wiederholen bei ETL-Änderungen)

ETL Test Life Cycle (Test-Lebenszyklus)

Phase	Beschreibung
Testplanung	Scope, Ziele, Ressourcen, Testdaten klären
Testdesign	Testfälle schreiben, basierend auf Mapping
Testausführung	Manuelle oder automatisierte Durchführung
Defektmanagement	Fehler dokumentieren, eskalieren
Testreporting	Testergebnisse aufbereiten, Abnahme sichern

Mögliche Test-Typen im ETL-Kontext

Testtyp	Ziel
Count Testing	Vergleich von Zeilenanzahl Quelle vs. Ziel
Data Validation	Feldinhalte und Formate prüfen
Data Completeness	Prüfung auf fehlende Werte / Spalten
Duplicate Testing	Doppelte Datensätze identifizieren
Null Testing	Unerwartete NULL-Werte finden
Business Rule Testing	Prüfen, ob Regeln korrekt angewendet wurden
Referentielle Integrität	Fremdschlüssel-/Join-Beziehungen prüfen
Performance Testing	Laufzeit und Ressourcenverbrauch messen
Regression Testing	Wiederholbarkeit nach Änderungen sicherstellen

Wie kann ein ETL-Testcase erstellt werden?

1. Testfallbeschreibung (Was soll geprüft werden?)

Beispiel: "Stimmen alle Produktnamen aus Quelltable X mit Ziel-DWH überein?"

2. Datenbasis definieren

- Quelle: stg_products
- Ziel: dwh_products

3. Testbedingungen & Annahmen

- product_id ist eindeutig
- keine NULL-Werte erlaubt in product_name

4. Erwartetes Ergebnis

- Anzahl Zeilen gleich
- Alle product_name korrekt transformiert

5. SQL/Testskript schreiben

-- Beispiel für Count Test

```
SELECT COUNT(*) FROM stg_products;
```

```
SELECT COUNT(*) FROM dwh_products;
```

-- Beispiel für Datenvergleich

```
SELECT s.product_id, s.product_name, t.product_name
```

```
FROM stg_products s
```

```
LEFT JOIN dwh_products t ON s.product_id = t.product_id
```

```
WHERE s.product_name <> t.product_name;
```

6. Dokumentation & Reporting

- Testergebnis: Erfolg / Fehler
- Screenshot / Query-Output
- Ticket-Nr. (bei Fehler)