

**Bewirtschaftung von Infrastrukturen.
Software. Daten. Prozesse.**



Handbuch

DQC (Data Quality Check) Kanton Solothurn Datenqualitätsprüfung für VSADSSMini 2020

Stand: 04.01.2023

**DATAVER
Informatik AG**




dataver.ch

Hauptsitz Goldach
Blumenfeldstrasse 15
9403 Goldach
T +41 71 544 32 50
goldach@dataver.ch

Niederlassung Kloten
Steinackerstrasse 2a
8302 Kloten
T +41 44 316 64 70
kloten@dataver.ch

Niederlassung Bern
Kornhausplatz 14
3011 Bern
T +41 31 381 44 51
bern@dataver.ch

Inhaltsverzeichnis

1	Ausgangslage.....	3
2	Voraussetzungen	3
3	Ausführung DQC Solothurn.....	3
3.1	Voraussetzungen DQC Solothurn	3
3.2	DQC Solothurn ausführen	4
3.3	Resultate.....	5
4	Interpretation Resultate	6
4.1	Fehlerpriorisierung.....	6
4.2	Listen und Statistiken.....	6
4.2.1	 [name]_error_list_so_sew.xlsx.....	7
4.2.2	 [name]_statistik_haltung_sew_mini.xlsx	7
4.2.3	 [name]_statistik_knoten_sew_mini.xlsx	8
4.2.4	Analysen Geopackage.....	9
4.3	Visuelle Überprüfung	9
4.3.1	Datenquelle anpassen	9
4.3.2	Verfügbare Layer	11
4.3.3	Netzverfolgung.....	18
5	Anhang	22

1 Ausgangslage

Der Data Quality Check (DQC) für den Kanton Solothurn prüft die Datenqualität eines Werkleitungskatasters auf Basis Interlis für das Medium Abwasser. Die Prüfergebnisse werden in ein Geopackage (.gpkg) geschrieben, priorisiert, aggregiert und ausgewertet.

Priorisierung. Nicht jedes Informationsbedürfnis hat dieselben Qualitätsansprüche. Je nach Prüflevel (z.B. Werkleitungskataster, GEP) müssen Fehler mit unterschiedlicher Priorität behandelt werden.

Aggregation. Die gefundenen Fehler werden so zusammengefasst, dass jede Fehlerart pro Objekt nur ein einziges Mal vorkommt.

Auswertung. Bei der Auswertung werden die Fehler pro Prüflevel aufgelistet. Die Fehler können nach Prüftyp, Priorisierung, Eigentümer oder Funktion gefiltert werden.

2 Voraussetzungen

Um den Data Quality Check (DQC) ausführen zu können, müssen folgende Voraussetzungen erfüllt sein:

- Java-Runtime 8 oder höher muss installiert sein

Damit alle Dateien, welche als Resultat ausgegeben werden, öffnen zu können, werden folgende Applikationen benötigt:

- QGIS Version ≥ 3.20 (<https://www.qgis.org/de/site/forusers/download.html>)
- Excel
- Texteditor (z.B. Notepad ++, Visual Studio Code, ...)
- SQLiteStudio (<https://sqlitestudio.pl/>)

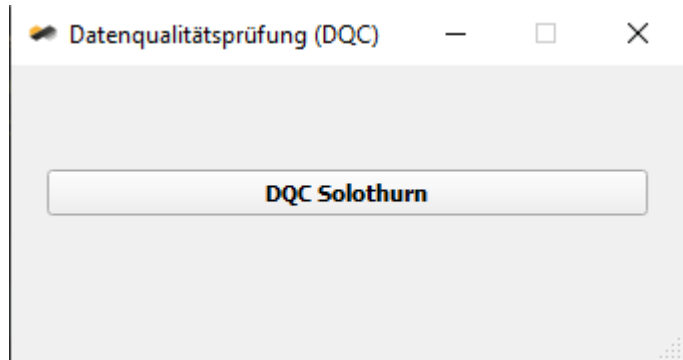
3 Ausführung DQC Solothurn

3.1 Voraussetzungen DQC Solothurn

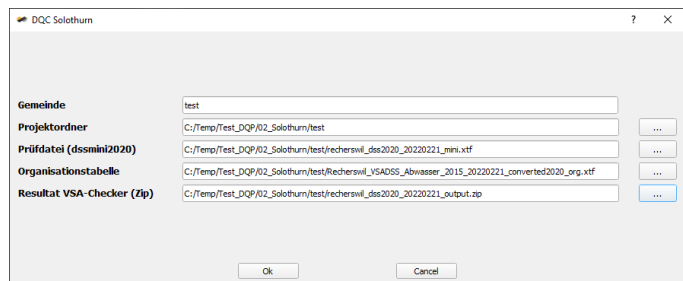
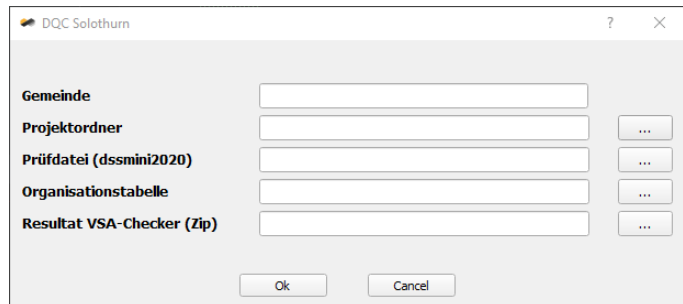
- Testdatei liegt einem der folgenden Modelle vor:
 - VSADSSMINI_2020_LV95 (Interlis 2.3)
- VSA-Check wurde bereits ausgeführt ([name]_fp_err.csv, [name]_t_err.csv und [name]_a_err.csv müssen in einem Zip-Archiv vorliegen)

3.2 DQC Solothurn ausführen

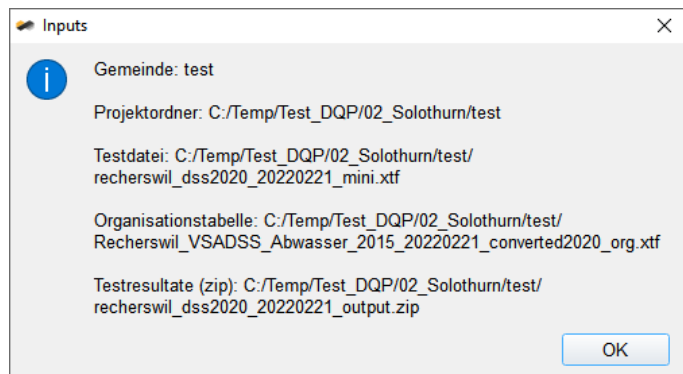
1. DQC über die **dqc_solothurn.exe** starten.
ACHTUNG: Je nach Antivirus Programm, kann der Start verzögert sein. → Bitte einfach warten, bis das Fenster sich öffnet.
2. DQC Solothurn wählen



3. Name der zu geprüften Gemeinde eingeben
4. Projektordner angeben → In diesem Ordner werden die Resultate des Data Quality Checks abgelegt
5. Zu prüfende Datei angeben (dssmini2020)
6. Individuelle Organisationstabelle angeben (falls nicht offizielle VSA-Organisationstabelle verwendet)
7. Zip-Ordner mit VSA-Checkserviceresultaten angeben
8. Eingaben mit Ok bestätigen



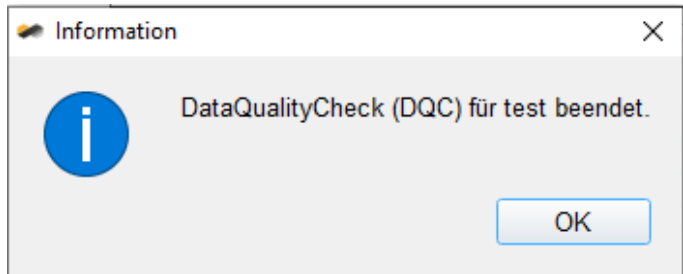
9. Eingaben mit Ok bestätigen



10....Prozess läuft
















```
C:\Dev_Py\EXE\dqdc_jmh\dqc_so_run.exe
2022-04-25 07:43:20,705 - INFO - dqc -> Fehlerpriorisierung erfolgreich ergänzt.
2022-04-25 07:43:22,325 - INFO - dqc -> Fehler erfolgreich aggregiert.
2022-04-25 07:43:22,337 - INFO - dqc -> Filteroptionen erfolgreich gelesen.
2022-04-25 07:43:22,465 - INFO - dqc -> Fehlerliste in Datenbank C:/Temp/Test_DQP/02_Solothurn/baettwil_20220414\Re
sultat\01_Daten\baettwil_20220425_dssmini_2020_lv95.gpkg importiert.
2022-04-25 07:43:22,487 - INFO - dqc -> Liste Objekte mit Fehler in Datenbank C:/Temp/Test_DQP/02_Solothurn/baettwi
l_20220414\Resultat\01_Daten\baettwil_20220425_dssmini_2020_lv95.gpkg importiert.
2022-04-25 07:43:22,489 - INFO - dqc -> Ende Fehleraufbereitung, gebrauchte Zeit: 0 Std 0 Min 2 Sek
2022-04-25 07:43:22,491 - INFO - dqc -> Start Erstellung Statistiken
2022-04-25 07:43:22,495 - INFO - dqc -> Datei von C:\Dev_Py\EXE\dqc_so_run\dqc\templates\error_list_so_sew.xlsx nac
h C:/Temp/Test_DQP/02_Solothurn/baettwil_20220414\Resultat\02_Datenanalyse\baettwil_20220425_dssmini2020_error_list_so_s
ew.xlsx kopiert
2022-04-25 07:43:22,495 - INFO - dqc -> Fehler werden in Excel geschrieben: C:/Temp/Test_DQP/02_Solothurn/baettwil
_20220414\Resultat\02_Datenanalyse\baettwil_20220425_dssmini2020_error_list_so_sew.xlsx
2022-04-25 07:43:22,614 - INFO - dqc -> error_data: Es werden 10 Spalten und 14156 Zeilen gelesen.
2022-04-25 07:43:31,383 - INFO - dqc -> Datei von C:\Dev_Py\EXE\dqc_so_run\dqc\templates\statistik_knoten_sew_mini2
020.xlsx nach C:/Temp/Test_DQP/02_Solothurn/baettwil_20220414\Resultat\02_Datenanalyse\baettwil_20220425_dssmini2020_st
atistik_knoten_sew_mini2020.xlsx kopiert
2022-04-25 07:43:31,383 - INFO - dqc -> Knoten werden in Excel geschrieben: C:/Temp/Test_DQP/02_Solothurn/baettwil
_20220414\Resultat\02_Datenanalyse\baettwil_20220425_dssmini2020_statistik_knoten_sew_mini2020.xlsx
2022-04-25 07:43:31,607 - INFO - dqc -> Knoten: Es werden 32 Spalten und 1873 Zeilen gelesen.
2022-04-25 07:43:33,904 - INFO - dqc -> Datei von C:\Dev_Py\EXE\dqc_so_run\dqc\templates\statistik_haltung_sew_mini
2020.xlsx nach C:/Temp/Test_DQP/02_Solothurn/baettwil_20220414\Resultat\02_Datenanalyse\baettwil_20220425_dssmini2020_st
atistik_haltung_sew_mini2020.xlsx kopiert
2022-04-25 07:43:33,905 - INFO - dqc -> Haltungen werden in Excel geschrieben: C:/Temp/Test_DQP/02_Solothurn/baettw
il_20220414\Resultat\02_Datenanalyse\baettwil_20220425_dssmini2020_statistik_haltung_sew_mini2020.xlsx
2022-04-25 07:43:34,125 - INFO - dqc -> Haltungen: Es werden 40 Spalten und 844 Zeilen gelesen.
2022-04-25 07:43:36,622 - INFO - dqc -> Knoten_Text werden in Excel geschrieben.
2022-04-25 07:43:36,788 - INFO - dqc -> Knoten_Text: Es werden 9 Spalten und 718 Zeilen gelesen.
```




11.Beendung des Prozesses mit Ok bestätigen



3.3 Resultate

Im Projektordner befinden sich nach Beendung des DQC folgende Ordner und Dateien:

-  log
 -  [datum]-dqdc_log.log
-  Prozess
 -  01_Datenaufbereitung
 -  [name]_vsa_output
 -  check
 - [name_original]_a_err.csv
 - [name_original]_fp_err.csv
 - [name_original]_t_err.csv
 - ...
 - ...
 -  [name_original]_replaced.xtf
 -  [name]_dssmini2015_ili2gpk.log
 -  02_Datenanalyse
-  Resultat
 -  01_Daten
 -  [name]_dssmini_2020_lv95.gpkg
 -  [name]_dssmini2020.xtf
 -  [name]_DQC_Error_DSSMini2020.ggz (vgl. Kapitel 4.1)
 -  02_Datenanalyse (vgl. Kapitel 4.2)

-  [name]_error_list_so_sew.xlsx
-  [name]_statistik_haltung_sew_mini2020.xlsx
-  [name]_statistik_knoten_sew_mini2020.xlsx

4 Interpretation Resultate

4.1 Fehlerpriorisierung

Die Fehler, welche aus dem VSA-Checkservice resultieren, werden im Rahmen der Datenqualitätsprüfung aggregiert und priorisiert. Für die Priorisierung werden die beiden Themen Werkleitungskataster (WK) und GEP berücksichtigt. Jeder Fehler erhält pro Thema die Priorisierung 1 oder 2.

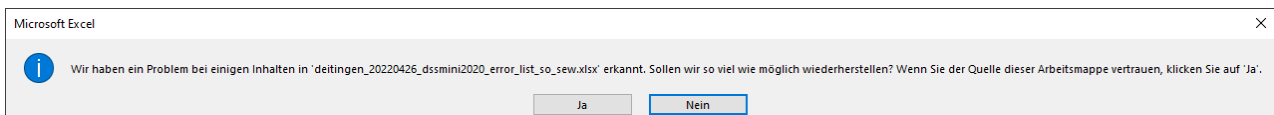
- **Priorisierung 1:** Es gibt Prozesse, bei denen eine Abhängigkeit besteht. Die Bereinigung der Fehler, stellt sicher, dass alle Prozesse reibungslos abgewickelt werden können.
- **Priorisierung 2:** Es gibt keine Prozesse, bei denen eine direkte Abhängigkeit zu einem gängigen Prozess besteht.

4.2 Listen und Statistiken

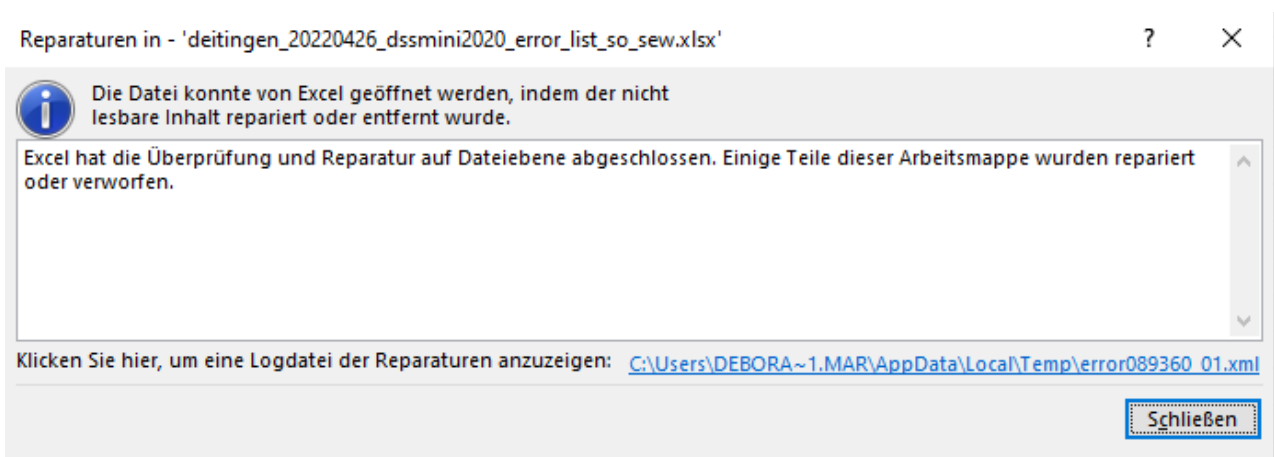
Im Ordner *Resultat/02_Datenanalyse/* befinden sich die Fehlerübersicht und die statistische Übersicht über alle Knoten und Haltungen (Leitungen). Zudem steht das Geopackage des geprüften Datensatzes unter *Resultat/01_Daten* für weitere Analysen zur Verfügung.

Erstmaliges Öffnen der Dateien Exceldateien

Beim erstmaligen Öffnen der Exceldateien erscheint folgende Meldung, welche mit **Ja** bestätigt werden kann:



Nach dem die Meldung mit Ja bestätigt wurde, erscheint folgende Meldung, welche via **Schliessen** geschlossen werden kann:



Im Anschluss muss die Datei gespeichert werden. Wird die Exceldatei danach geschlossen und wieder geöffnet, erscheint die Meldung nicht mehr.

4.2.1 [name]_error_list_so_sew.xlsx

Das Excel-File enthält folgende Blätter:

- **Uebersicht_WK:** Diese Blatt enthält eine Pivottabelle, welche die Fehler nach Klasse und Priorität gruppiert. Die Fehlerpriorisierung legt den Fokus auf den Werkleitungskataster.
- **Uebersicht_GEP:** Diese Blatt enthält eine Pivottabelle, welche die Fehler nach Klasse und Priorität gruppiert. Die Fehlerpriorisierung legt den Fokus auf den GEP.
- **error_data:** Dieses Blatt enthält die Rohdaten für die beiden Pivottabellen. Alle Fehler werden einzeln aufgelistet.

4.2.2 [name]_statistik_haltung_sew_mini.xlsx

Das Excel-File enthält folgende Blätter:

- **Nutzungsart:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *nutzungsart_ist* und *funktionhierarchisch* gruppiert.
- **F.Hierarchisch:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *funktionhierarchisch* gruppiert.
- **F.Hydraulisch:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *funktionhydraulisch* gruppiert.
- **F.Hydraulisch_Hierarchisch:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *funktionhydraulisch* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.
- **Finanzierung:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *finanzierung* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.
- **Baulicher Zustand:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *baulicherzustand* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.
- **Sanierungsbedarf:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *sanierungsbedarf* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.
- **Eigentümer vs. Betreiber:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *eigentuemer* und *betreiber* gruppiert.
- **Baujahr:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *baujahr_klasse* und *baujahr* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.
- **Baujahr - Material:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *material*, *baujahr_klasse* und *baujahr* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.
- **Status:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *status* gruppiert.
- **Status - Baulicher Zustand:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *status* und *baulicherzustand* gruppiert.
- **Rohrprofil:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *rohrprofil* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.
- **Lichtehöhe:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *lichte_hoehe* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.

- **Lichtebreite:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *lichte_breite* gruppiert und nach *funktionhierarchisch_klasse* (PAA / SAA) sortiert.
- **Kote_Beginn:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *kote_von* gruppiert und nach *funktionhierarchisch* sortiert.
- **Kote_Ende:** Diese Blatt enthält eine Pivottabelle, welche die Haltungen nach *kote_nach* gruppiert und nach *funktionhierarchisch* sortiert.
- **Bezeichnung:** Diese Blatt enthält eine Pivottabelle, welche Haltungen mit mehrfach verwendeten Bezeichnungen auflistet.
- **Haltung:** Dieses Blatt enthält die Rohdaten für die Pivottabellen. Die Daten entsprechen, den Daten im Geopackage (Tabelle *leitung*).
- **Mehrfachtext:** Dieses Blatt enthält zwei Pivottabellen. Die erste Pivottabelle listet alle Beschriftungen auf, welche denselben *plantyp* haben und an gleicher Position (*textpos*) vorkommen. Die zweite Pivottabelle listet alle Textinhalte auf, welche mehr als einmal vorkommen.
- **Haltung_Text:** Dieses Blatt enthält die Rohdaten für die Pivottabelle Mehrfachtext. Die Daten entsprechen, den Daten im Geopackage (Tabelle *leitung_text*).

4.2.3 [name]_statistik_knoten_sew_mini.xlsx

Das Excel-File enthält folgende Blätter:

- **Funktion:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *funktion* gruppiert und nach *funktionhierarchisch* sortiert.
- **Eigentümer vs. Betreiber:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *eigentümer und bereiber* gruppiert und nach *funktionhierarchisch* sortiert.
- **Finanzierung:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *finanzierung* gruppiert.
- **Baulicher Zustand:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *baulicherzustand* gruppiert und nach *funktionhierarchisch* sortiert.
- **Sanierungsbedarf:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *sanierungsbedarf* gruppiert und nach *funktionhierarchisch* sortiert.
- **Baujahr:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *baujahr_klasse* und *baujahr* gruppiert und nach *funktionhierarchisch* sortiert.
- **Status:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *status* gruppiert und nach *funktionhierarchisch* sortiert.
- **Status - Baulicher Zustand:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *status* und *baulicherzustand* gruppiert.
- **Deckelkote:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *deckelkote* gruppiert und nach *funktionhierarchisch* sortiert.
- **Sohlenkote:** Diese Blatt enthält eine Pivottabelle, welche die Knoten nach *sohlenkote* gruppiert und nach *funktionhierarchisch* sortiert.
- **Bezeichnung:** Diese Blatt enthält eine Pivottabelle, welche Knoten mit mehrfach verwendeten Bezeichnungen auflistet.
- **Lage:** Diese Blatt enthält eine Pivottabelle, welche Knoten mit mehrfach verwendeter Lage (identische Koordinaten) auflistet.

- **Knoten:** Dieses Blatt enthält die Rohdaten für die Pivottabellen. Die Daten entsprechen, den Daten im Geopackage (Tabelle *knoten*).
- **Mehrfachtext:** Dieses Blatt enthält zwei Pivottabellen. Die erste Pivottabelle listet alle Beschriftungen auf, welche denselben *plantyp* haben und an gleicher Position (*textpos*) vorkommen. Die zweite Pivottabelle listet alle Textinhalte auf, welche mehr als einmal vorkommen.
- **Knoten_Text:** Dieses Blatt enthält die Rohdaten für die Pivottabellen. Die Daten entsprechen, den Daten im Geopackage (Tabelle *knoten_text*).

4.2.4 Analysen Geopackage

Für weitere Analysen direkt im Geopackage stehen verschiedene SQL-Abfragen zur Verfügung (vgl. Kapitel 5). Die SQL-Abfragen können in der Datenbank, also im Geopackage, ausgeführt werden. Das Geopackage kann zum Beispiel mit SQLiteStudio (<https://sqlitestudio.pl/>) geöffnet werden.

- Zusammenzug Anzahl Fehlerobjekte (alle Klassen)
- Leitung - Fehlerverteilung Fehlerart
- Leitung - Fehlende kote_von / kote_nach
- Knoten - Fehlerverteilung Fehlerart
- Knoten - Fehlend Sohlenkote / Deckelkote pro zugaenglichkeit

4.3 Visuelle Überprüfung

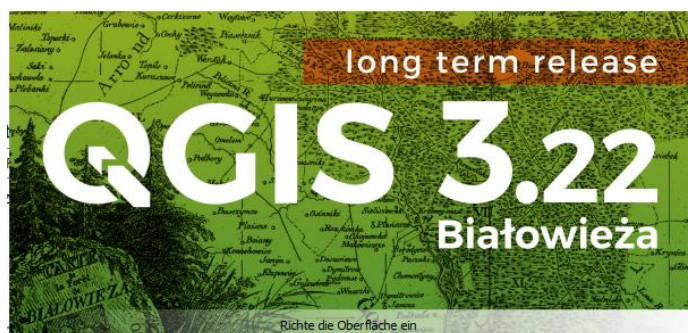
Das QGIS-Projekt für die Visualisierung der Daten befindet sich im Ordner *Resultat/01_Daten/* und trägt den Namen *[name]_DQC_Error_DSSMini2020.qgz*. Das dazugehörige Geopackage (*[name]_dssmini_2020_lv95.gpkg*) befindet sich im selben Ordner.

4.3.1 Datenquelle anpassen

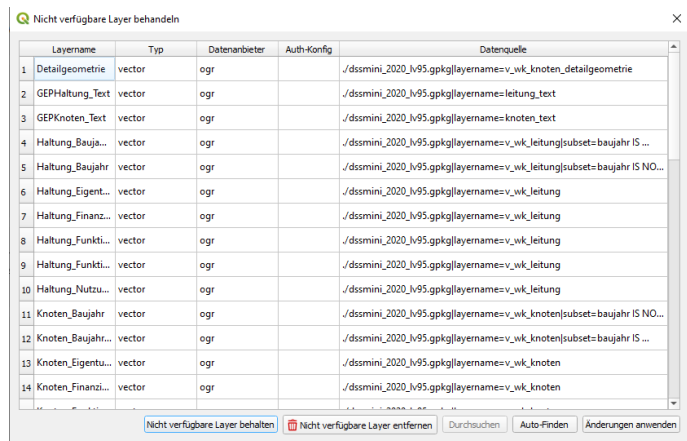
Die Datenquelle der Layer muss angepasst werden, wenn das QGIS-Projekt das erste Mal geöffnet wird, oder das Geopackage (also die Datenquelle) verschoben worden ist.

1. QGIS Projekt öffnen

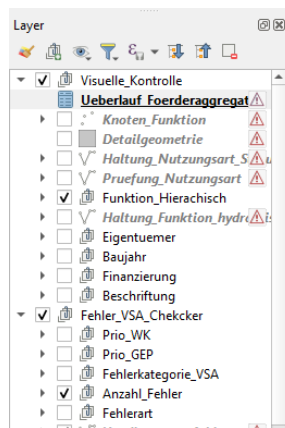
Empfohlen wird QGIS ≥ 3.20



2. Nicht verfügbare Layer behalten

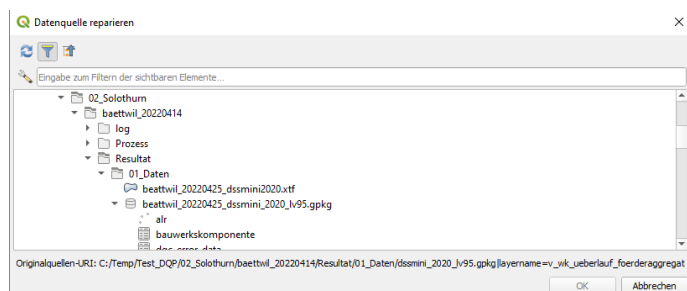


3. Oberstes  auswählen



4. Zu Geopackage navigieren

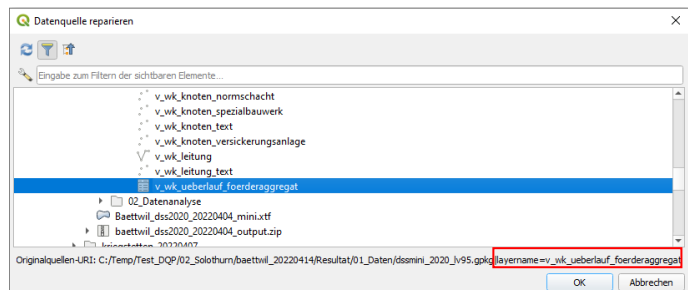
Resultat/01_Daten/[name]_dssmini_2020_lv95.gpkg



5. Layer auswählen

Bsp. Wurde «Datenquelle reparieren» über den Layer Ueberlauf_Foerderaggreat aufgerufen, muss hier `v_wk_ueberlauf_foerderaggreat` ausgewählt werden → Originalquellen-URL: `[pfad-geopackage]//layername=v_wk_ueberlauf_foerderaggreat`

Da die Datenquelle (das Geopackage) für alle Layer das selbe ist, werden alle anderen Datenquellen automatisch repariert.



6. QGIS-Projekt speichern

4.3.2 Verfügbare Layer

Im Rahmen des Projektes LKMod des Kanton Solothurns werden folgende Layer im QGIS-Projekt zur Verfügung gestellt. Selbstverständlich darf das Projekt individuell ergänzt werden.




















Visuelle Kontrolle

- ☑  Sonderbauwerke_Kt_SO
 - ☑   **sk_trennbauwerk**
 - ☑   **sk_regenueberlaufbecken**
 - ☑   **sk_regenueberlauf**
 - ☑   **sk_regenrueckhaltebecken_kanal**
 - ☑   **sk_pumpwerk**
 - ☑   **sk_einleitstelle**
 - ☑   **sk_duekeroberhaupt**

Sonderbauwerke Kanton Solothurn.

Datenquelle:

- *Sonderbauwerke_SO.gpkg*
 - *sk_trennbauwerk*
 - *sk_regenueberlaufbecken*
 - *sk_regenueberlauf*
 - *sk_regenueberlaufbecken_kanal*
 - *sk_pumpwerk*
 - *sk_einleitstelle*
 - *sk_duekeroberhaupt*




































- ☑  **Knoten_Funktion**
 - ☑   Regenbecken_Durchlaufbecken,Regenbecken_Fangb
 - ☑   Regenueberlauf
 - ☑   Trennbauwerk
 - ☑   Versickerungsanlage.andere
 - ☑   Pumpwerk
 - ☑   Klaergrube,Vorbehandlung,Strassenwasserbehandlun
 - ☑   Einleitstelle_gewaesserrelevant
 - ☑   Einleitstelle_nicht_gewaesserrelevant
 - ☑   Kontrollschacht

Dieser Layer visualisiert die Knoten nach Knotenfunktion.

Datenquelle:

- **Knoten:** *v_vk_knoten*

...

<input checked="" type="checkbox"/>  Ueberlauf_Foerderaggregat	Überlauf Förderaggregate
	Datenquelle: <ul style="list-style-type: none"> • <i>v_wk_ueberlauf_foerderaggregat</i>
<input checked="" type="checkbox"/>  Detailgeometrie	Datenquelle: <ul style="list-style-type: none"> • <i>v_wk_knoten_detailgeometrie</i>
<input checked="" type="checkbox"/>  Topologisch_Verbindung <input checked="" type="checkbox"/>  ueberlauf_foerderaggregat <input checked="" type="checkbox"/>  topologielinie <input checked="" type="checkbox"/> 	Dieser Layer zeigt die berechneten topologischen Verbindungen aufgrund der referenzierten Knoten sowie den Start- und Endkoordinaten der Leitung. Datenquelle: <ul style="list-style-type: none"> • <i>dqc_topological_line</i>
<input checked="" type="checkbox"/>  Knoten_Nutzungsart <input checked="" type="checkbox"/>  Entlastetes_Mischabwasser <input checked="" type="checkbox"/>  Niederschlagsabwasser,Reinabwasser <input checked="" type="checkbox"/>  Bachwasser <input checked="" type="checkbox"/>  Mischabwasser <input checked="" type="checkbox"/>  Schmutzabwasser,Industrieabwasser <input checked="" type="checkbox"/>  andere,unbekannt <input checked="" type="checkbox"/> 	Diese Layer visualisieren die Nutzungsart IST Knoten. Datenquelle: <ul style="list-style-type: none"> • Knoten: <i>v_vk_knoten</i>
<input checked="" type="checkbox"/>  Haltung_Nutzungsart_Status <input checked="" type="checkbox"/>  Entlastetes_Mischwasser in_Betrieb <input checked="" type="checkbox"/>  Entlastetes_Mischwasser projektiert <input checked="" type="checkbox"/>  Entlastetes_Mischwasser unbekannt <input checked="" type="checkbox"/>  Niederschlagsabwasser / Reinabwasser in_Betrieb <input checked="" type="checkbox"/>  Niederschlagsabwasser / Reinabwasser projektiert <input checked="" type="checkbox"/>  Niederschlagsabwasser / Reinabwasser unbekannt <input checked="" type="checkbox"/>  Bachwasser in_Betrieb <input checked="" type="checkbox"/>  Bachwasser projektiert <input checked="" type="checkbox"/>  Bachwasser unbekannt <input checked="" type="checkbox"/>  Mischabwasser in_Betrieb <input checked="" type="checkbox"/>  Mischabwasser projektiert <input checked="" type="checkbox"/>  Mischabwasser unbekannt <input checked="" type="checkbox"/>  Schmutzabwasser / Industrieabwasser in_Betrieb <input checked="" type="checkbox"/>  Schmutzabwasser / Industrieabwasser projektiert <input checked="" type="checkbox"/>  Schmutzabwasser / Industrieabwasser unbekannt <input checked="" type="checkbox"/>  andere / unbekannt in_Betrieb <input checked="" type="checkbox"/>  andere / unbekannt projektiert <input checked="" type="checkbox"/>  andere / unbekannt unbekannt <input checked="" type="checkbox"/>  ausser Betrieb / tot (alle Nutzungsarten) <input checked="" type="checkbox"/> 	Diese Layer visualisieren die Nutzungsart IST kombiniert mit dem Status für alle Haltungen (Leitungen). Die Pfeile zeigen die Fliessrichtung. Datenquelle: <ul style="list-style-type: none"> • Haltung: <i>v_vk_leitung</i>

- Pruefung_Nutzungsart**
 - Nutzungsart G ist NULL
 - Nutzungsart IST <> G
 - Nutzungsart IST = G

Dieser Layer visualisiert den Vergleich von die Nutzungsart IST und die Nutzungsart GEPLANT für alle Haltungen (Leitungen).

- Knoten_FunktionHierarchisch**
 - PAA
 - SAA
 -
- Knoten_FunktionHierarchisch_RB_Rue**
 - PAA
 - SAA
 -
- Knoten_FunktionHierarchisch_Sonderbauwerke**
 - PAA
 - SAA
 -

Diese Layer visualisieren die Funktion hierarchisch.

Datenquelle:

- **Knoten:** *v_wk_knoten*
- **Haltung:** *v_vk_leitung*

- Haltung_FunktionHierarchisch**
 - PAA.Hauptsammelkanal_regional
 - PAA.Hauptsammelkanal
 - PAA.Sammelkanal
 - PAA.Strassenentwaesserung
 - PAA.Arealentwaesserung
 - PAA.Liegenschaftsentwaesserung
 - PAA.Sanierungsleitung
 - PAA.Gewaesser
 - PAA.andere,PAA.unbekannt
 - SAA.Liegenschaftsentwaesserung
 - SAA.Sanierungsleitung
 - SAA.Strassenentwaesserung
 - SAA.andere,SAA.unbekannt
 -

Dieser Layer visualisiert die Funktion hierarchisch.

Datenquelle:

- **Haltung:** *v_vk_leitung*

- Haltung_FunktionHierarchisch_gruppiert**
 - PAA
 - SAA
 -

Dieser Layer visualisiert die Funktion hierarchisch gruppiert nach PAA / SAA.

Datenquelle:

- **Haltung:** *v_vk_leitung*

☑ **Haltung_Funktion_hydraulisch**

- ☑ Freispiegelleitung
- ☑ Pumpendruckleitung
- ☑ Vakuumleitung
- ☑ Drainagetransportleitung
- ☑ Sickerleitung
- ☑ Speicherleitung
- ☑ Drosselleitung
- ☑ Duekerleitung
- ☑ andere, unbekannt
- ☑ ----

Dieser Layer visualisiert die Funktion hydraulisch für alle Haltungen (Leitungen).

Datenquelle:

- **Haltung:** *v_vk_leitung*

☑ **Knoten_Eigentuemer**

- ☑ Genossenschaft_Kooperation
- ☑ Abwasserverband
- ☑ Kanton
- ☑ Gemeinde
- ☑ Privat
- ☑ andere
- ☑ ●

Diese Layer visualisieren den Eigentübertyp (Organisations-
typ) pro Objekt.

Datenquelle:

- **Knoten:** *v_wk_knoten*

☑ **Haltung_Eigentuemer**

- ☑ Genossenschaft_Kooperation
- ☑ Abwasserverband
- ☑ Kanton
- ☑ Gemeinde
- ☑ Privat
- ☑ andere
- ☑ ■

Diese Layer visualisieren den Eigentübertyp (Organisations-
typ) pro Objekt.

Datenquelle:

- **Haltung:** *v_vk_leitung*

☑ **Knoten_Baujahr**

- ☑ 1800 - 1900
- ☑ 1901 - 1920
- ☑ 1921 - 1940
- ☑ 1941 - 1960
- ☑ 1961 - 1980
- ☑ 1981 - 2000
- ☑ 2001 - 2020
- ☑ 2021 - 2040
- ☑ **Knoten_Baujahr_NULL**

Diese Layer visualisieren das Baujahr pro Objekt.

Datenquelle:

- **Knoten:** *v_wk_knoten*

Haltung_Baujahr

- 1800 - 1900
- 1901 - 1920
- 1921 - 1940
- 1941 - 1960
- 1961 - 1980
- 1981 - 2000
- 2001 - 2020
- 2021 - 2040

Haltung_Baujahr_NULL

Diese Layer visualisieren das Baujahr pro Objekt.

Datenquelle:

- **Haltung:** *v_vk_leitung*

Knoten_Finanzierung

- öffentlich
- privat
- unbekannt
-

Diese Layer visualisieren die Finanzierung pro Objekt.

Datenquelle:

- **Knoten:** *v_wk_knoten*

Haltung_Finanzierung

- öffentlich
- privat
- unbekannt
-

Diese Layer visualisieren die Finanzierung pro Objekt.

Datenquelle:

- **Haltung:** *v_vk_leitung*

Knoten_Text

- Werkplan
-

Dieser Layer zeigt die Position der Texte an. Hervorgehoben sind alle Werkplan-Texte.

Datenquelle:

- **Knoten:** *knoten_text*
- **Haltung:** *leitung_text*

Haltung_Text

- Werkplan
-

Dieser Layer zeigt die Position der Texte an. Hervorgehoben sind alle Werkplan-Texte.

Datenquelle:

- **Haltung:** *leitung_text*

Netzverfolgung

dqc_network_edges

- ueberlauf_foerderaggregat
- topologielinie
-

Dieser Layer ist die Basis für die Netzverfolgung (vgl. Kapitel 4.3.3)

Fehler_VSA_Checker

☑ Fehlerart_Knoten

- Auslauf höher als Zulauf
- Bezeichnung has to be defined
- Die Bezeichnung enthält nicht empfohlene Zeichen (alle ausser [
- Die TID ist keine gültige OID (STANDARDOID).
- EigentuemerRef has to be defined
- Kein Ueberlauf_Foerderaggregat erfasst
- Knoten ohne Auslauf
- Knotensole höher als Leitungsohlen
- MANDATORY ARA_Nr
- MANDATORY Baujahr
- MANDATORY BaulicherZustand
- MANDATORY ...

...

Dieser Layer zeigt die Fehler gemäss VSA-Checker. Es empfiehlt sich, die Klassifizierung *unter Layereigenschaften / Symbolisierung* für jeden Kataster zu aktualisieren.

Die einzelnen Kategorien sollten nur einzeln eingeschaltet werden, da ein Objekt mehrere Fehler haben kann.

Datenquelle:

- *v_error_error_knoten*

☑ Fehlerart_Haltung

- Bezeichnung has to be defined
- Die berechnete Leitungslänge ist wesentlich grösser als die Distanz
- Die berechnete Leitungslänge ist wesentlich kürzer als die Distanz
- Die Bezeichnung enthält nicht empfohlene Zeichen (alle ausser [0
- Die TID ist keine gültige OID (STANDARDOID).
- Distanz Knoten zu Einlauf > Toleranzwert
- Drainageleitung führt nicht nur Reinabwasser
- EigentuemerRef has to be defined
- Fehlender Knoten_nach (PAA)
- Fehlender Knoten_von (PAA)
- Grosskalibrige SAA
- LaengeEffektiv ist nicht plausibel

...

Dieser Layer zeigt die Fehler gemäss VSA-Checker. Es empfiehlt sich, die Klassifizierung *unter Layereigenschaften / Symbolisierung* für jeden Kataster zu aktualisieren.

Die einzelnen Kategorien sollten nur einzeln eingeschaltet werden, da ein Objekt mehrere Fehler haben kann.

Datenquelle:

- *v_error_error_haltung*

☑ Handlungsempfehlung_Knoten

- Anschlussknoten prüfen (Schwellwert berücksichtigen)
- Daten bereinigen
- Daten erheben
- Daten erheben / ev. Kanal-TV
- Daten prüfen
- Daten prüfen / ev. Feldaufnahmen
- Daten prüfen / ev. Kanal-TV
- Entwässerungssystem prüfen (ev. Anschlussknoten entfernen)
- Entwässerungssystem und Parameter prüfen (ev. Anschlussknoten)
- Feldaufnahme

Dieser Layer zeigt die Handlungsempfehlung, um die Fehler gemäss VSA-Checker zu beheben.

Es empfiehlt sich, die Klassifizierung *unter Layereigenschaften / Symbolisierung* für jeden Kataster zu aktualisieren.

Die einzelnen Kategorien sollten nur einzeln eingeschaltet werden, da ein Objekt mehrere Fehler haben kann.

Datenquelle:

- *v_error_recommendation_knoten*

- Handlungsempfehlung_Haltung
 - Anschlussknoten prüfen (Schwellwert berücksichtigen)
 - Daten bereinigen
 - Daten erheben
 - Daten erheben / ev. Kanal-TV
 - Daten prüfen
 - Daten prüfen / ev. Feldaufnahmen
 - Daten prüfen / ev. Kanal-TV
 - Entwässerungssystem prüfen (ev. Anschlussknoten entfernen)
 - Entwässerungssystem und Parameter prüfen (ev. Anschlussknoten)
 - Feldaufnahme

Dieser Layer zeigt die Handlungsempfehlung, um die Fehler gemäss VSA-Checker zu beheben.

Es empfiehlt sich, die Klassifizierung *unter Layereigenschaften / Symbolisierung* für jeden Kataster zu aktualisieren.

Die einzelnen Kategorien sollten nur einzeln eingeschaltet werden, da ein Objekt mehrere Fehler haben kann.

Datenquelle:

- *v_error_recommendation_haltung*

- Prio_WK
 - Prio_WK_Knoten
 - 1
 - 2
 -
 - Prio_WK_Haltung
 - 1
 - 2
 -

Diese Layer visualisieren die höchste Fehlerpriorisierung pro Objekt für das Thema Werkleitungskataster (WK).

Hat ein Objekt mehrere Fehler und diese Fehler weisen unterschiedliche Priorisierungen auf, so wird in diesem Layer nur die höchste Priorisierung angegeben.

Datenquelle:

- **Knoten:** *v_error_knoten*
- **Haltung:** *v_error_haltung*

- Prio_GEP
 - Prio_GEP_Knoten
 - 1
 - 2
 -
 - Prio_GEP_Haltung
 - 1
 - 2
 -

Diese Layer visualisieren die höchste Fehlerpriorisierung pro Objekt für das Thema GEP.

Hat ein Objekt mehrere Fehler und diese Fehler weisen unterschiedliche Priorisierungen auf, so wird in diesem Layer nur die höchste Priorisierung angegeben.

Datenquelle:

- **Knoten:** *v_error_knoten*
- **Haltung:** *v_error_haltung*

- Fehlerkategorie_VSA
 - Fehlerkategorie_VSA_Knoten
 - ERROR
 - WARNING
 - INFO
 -
 - Fehlerkategorie_VSA_Haltung
 - ERROR
 - WARNING
 - INFO
 -

Diese Layer visualisieren die Fehlerkategorie gemäss VSA-Checker.

Da ein Objekt mehr als einen Fehler haben kann und somit mehrere Fehlerkategorien möglich sind, empfiehlt es sich, die Kategorien einzeln anzuzeigen und nicht alle zusammen.

Datenquelle:

- **Knoten:** *v_error_category_knoten*
- **Haltung:** *v_error_category_haltung*



Diese Layer visualisieren die Anzahl Fehler pro Objekt.

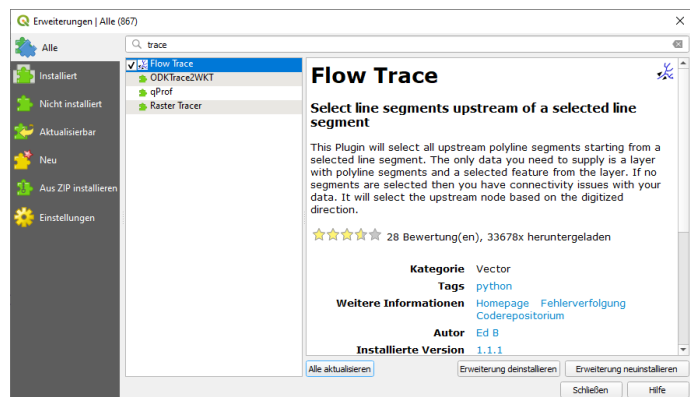
Datenquelle:

- **Knoten:** `v_error_knoten`
- **Haltung:** `v_error_haltung`

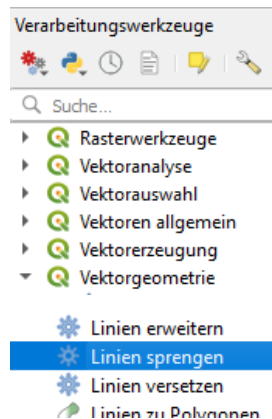
4.3.3 Netzverfolgung

Um die Netzverfolgung machen zu können, muss das Netz zuerst aufbereitet werden.

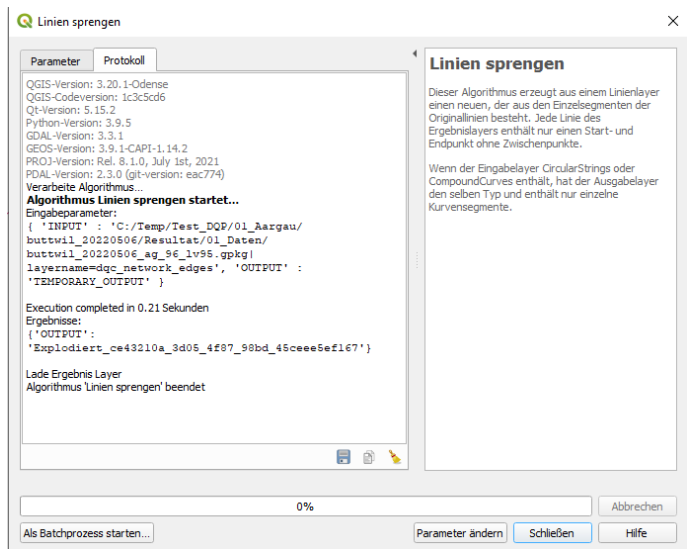
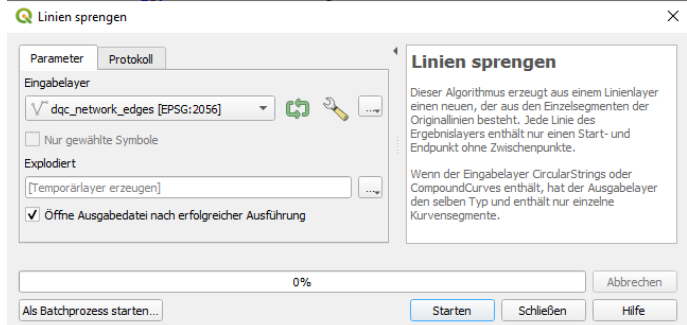
1. Falls nicht vorhanden: Plugin «Flow Trace» installieren



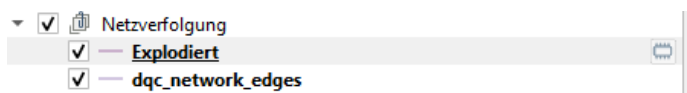
2. Verarbeitungswerkzeuge öffnen
3. Werkzeug «Linien sprengen» auswählen



4. «Linien sprengen»
Eingabelayer = *dqc_network_edges*

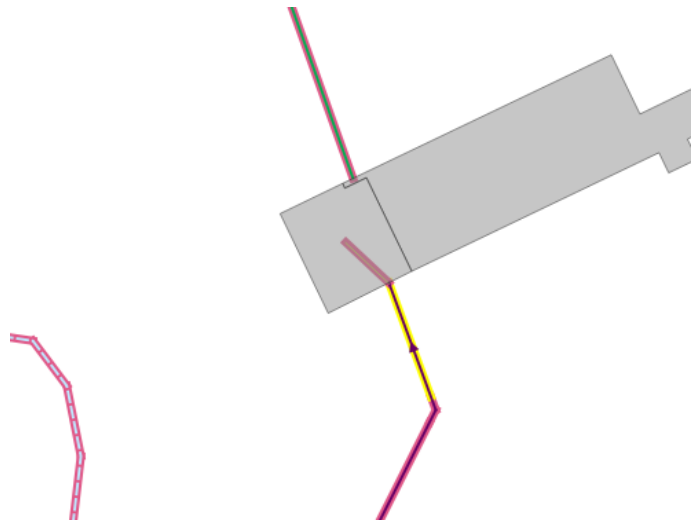


5. Layer «Explodiert» bei Bedarf stilisieren

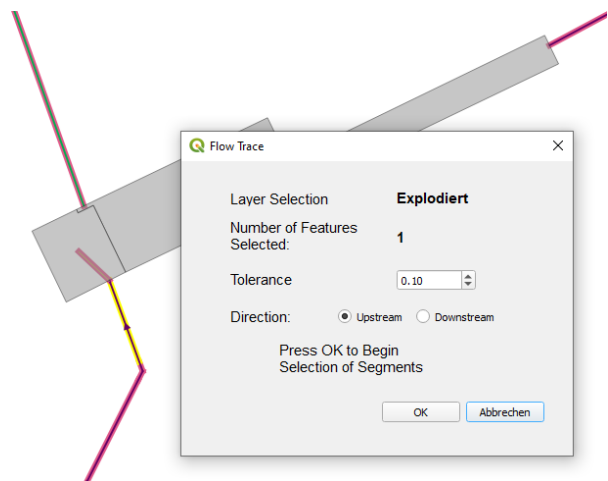


6. Geeignete Haltung selektieren

Die gewählte Haltung sollte irgendwo am Ende des Netzes gewählt werden, so dass Stromaufwärts das Netz verfolgt werden kann.



7. «Flow Trace» starten
Direction = Upstream



8. Netz visuell nach Unstimmigkeiten überprüfen



5 Anhang

A	SQL-Abfragen	22
A.1	Zusammenzug Anzahl Fehlerobjekte (alle Klassen)	22
A.2	Leitung - Fehlerverteilung Fehlerart	23
A.3	Leitung - Fehlende kote_von / kote_nach	24
A.4	Knoten - Fehlerverteilung Fehlerart.....	24
A.5	Knoten - Fehlend Sohlenkote / Deckelkote pro zugaenglichkeit.....	24
A.6	Knoten - Fehlend Sohlenkote / Deckelkote pro zugaenglichkeit / funktion	25

A SQL-Abfragen

A.1 Zusammenzug Anzahl Fehlerobjekte (alle Klassen)

```
SELECT
'leitung' AS tabelle,
COUNT(a.t_id) AS objekte_total,
COUNT(a.t_id) - COUNT(eo.tid) AS fehlerfreie_objekte,
COUNT(eo.tid) AS error_objekt,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_paa,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_saa,
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_unbekannt
FROM leitung a
LEFT JOIN dqc_error_object eo ON a.t_ili_tid = eo.tid
UNION ALL
SELECT
'leitung_text' AS tabelle,
COUNT(a.t_id) AS objekte_total,
COUNT(a.t_id) - COUNT(eo.tid) AS fehlerfreie_objekte,
COUNT(eo.tid) AS error_objekt,
COUNT(CASE WHEN b.funktionhierarchisch LIKE 'PAA%' AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_paa,
COUNT(CASE WHEN b.funktionhierarchisch LIKE 'SAA%' AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_saa,
COUNT(CASE WHEN b.funktionhierarchisch IS NULL AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_unbekannt
FROM leitung_text a
LEFT JOIN leitung b ON a.leitungref = b.t_id
LEFT JOIN dqc_error_object eo ON a.t_ili_tid = eo.tid
UNION ALL
SELECT
'knoten' AS tabelle,
COUNT(a.t_id) AS objekte_total,
COUNT(a.t_id) - COUNT(eo.tid) AS fehlerfreie_objekte,
COUNT(eo.tid) AS error_objekt,
COUNT(CASE WHEN a.funktionhierarchisch = 'PAA' AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_paa,
COUNT(CASE WHEN a.funktionhierarchisch = 'SAA' AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_saa,
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_unbekannt
FROM knoten a
LEFT JOIN dqc_error_object eo ON a.t_ili_tid = eo.tid
UNION ALL
```

```
SELECT
'knoten_text' AS tabelle,
COUNT(a.t_id) AS objekte_total,
COUNT(a.t_id) - COUNT(eo.tid) AS fehlerfreie_objekte,
COUNT(eo.tid) AS error_objekt,
COUNT(CASE WHEN b.funktionhierarchisch = 'PAA' AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_paa,
COUNT(CASE WHEN b.funktionhierarchisch = 'SAA' AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_saa,
COUNT(CASE WHEN b.funktionhierarchisch IS NULL AND eo.tid IS NOT NULL THEN 1 END) AS error_objekt_unbekannt
FROM knoten_text a
LEFT JOIN knoten b ON a.knotenref = b.t_id
LEFT JOIN dqc_error_object eo ON a.t_ili_tid = eo.tid
UNION ALL
SELECT
'massnahme' AS tabelle,
COUNT(a.t_id) AS objekte_total,
COUNT(a.t_id) - COUNT(eo.tid) AS fehlerfreie_objekte,
COUNT(eo.tid) AS error_objekt,
NULL AS error_objekt_paa,
NULL AS error_objekt_saa,
NULL AS error_objekt_unbekannt
FROM massnahme a
LEFT JOIN dqc_error_object eo ON a.t_ili_tid = eo.tid
UNION ALL
SELECT
'teileinzugsgebiet' AS tabelle,
COUNT(a.t_id) AS objekte_total,
COUNT(a.t_id) - COUNT(eo.tid) AS fehlerfreie_objekte,
COUNT(eo.tid) AS error_objekt,
NULL AS error_objekt_paa,
NULL AS error_objekt_saa,
NULL AS error_objekt_unbekannt
FROM teileinzugsgebiet a
LEFT JOIN dqc_error_object eo ON a.t_ili_tid = eo.tid
UNION ALL
SELECT
'ueberlauf_foerderaggregat' AS tabelle,
COUNT(a.t_id) AS objekte,
COUNT(a.t_id) - COUNT(eo.tid) AS fehlerfreie_objekte,
COUNT(eo.tid) AS error_objekt,
NULL AS error_objekt_paa,
NULL AS error_objekt_saa,
NULL AS error_objekt_unbekannt
FROM ueberlauf_foerderaggregat a
LEFT JOIN dqc_error_object eo ON a.t_ili_tid = eo.tid
```

A.2 Leitung - Fehlerverteilung Fehlerart

```
SELECT
ed.error_type,
'leitung' AS tabelle,
ed.wk,
ed.gep,
ed.error,
COUNT(ed.tid) AS error_total,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND ed.tid IS NOT NULL THEN 1 END) AS error_paa,
```

```
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND ed.tid IS NOT NULL THEN 1 END) AS error_saa,  
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND ed.tid IS NOT NULL THEN 1 END) AS error_unbekannt  
FROM leitung a  
LEFT JOIN dqc_error_data ed ON a.t_ili_tid = ed.tid  
GROUP BY ed.error  
ORDER BY ed.error_type, ed.error
```

A.3 Leitung - Fehlende kote_von / kote_nach

```
SELECT  
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND a.kote_von IS NULL AND a.kote_nach IS NOT NULL THEN 1  
END) AS paa_von,  
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND a.kote_von IS NULL AND a.kote_nach IS NOT NULL THEN 1  
END) AS saa_von,  
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND a.kote_von IS NULL AND a.kote_nach IS NOT NULL THEN 1  
END) AS unbekannt_von,  
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND a.kote_nach IS NULL AND a.kote_von IS NOT NULL THEN 1  
END) AS paa_nach,  
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND a.kote_nach IS NULL AND a.kote_von IS NOT NULL THEN 1  
END) AS saa_nach,  
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND a.kote_nach IS NULL AND a.kote_von IS NOT NULL THEN 1  
END) AS unbekannt_nach,  
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND a.kote_von IS NULL AND a.kote_nach IS NULL THEN 1  
END) AS paa_beide,  
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND a.kote_von IS NULL AND a.kote_nach IS NULL THEN 1  
END) AS saa_beide,  
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND a.kote_von IS NULL AND a.kote_nach IS NULL THEN 1 END) AS  
unbekannt_beide  
FROM leitung a
```

A.4 Knoten - Fehlerverteilung Fehlerart

```
SELECT  
ed.error_type,  
'knoten' AS klasse,  
ed.wk,  
ed.gep,  
ed.error,  
COUNT(ed.tid) AS error_total,  
COUNT(CASE WHEN a.funktionhierarchisch = 'PAA' AND ed.tid IS NOT NULL THEN 1 END) AS error_paa,  
COUNT(CASE WHEN a.funktionhierarchisch = 'SAA' AND ed.tid IS NOT NULL THEN 1 END) AS error_saa,  
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND ed.tid IS NOT NULL THEN 1 END) AS error_unbekannt  
FROM knoten a  
LEFT JOIN dqc_error_data ed ON a.t_ili_tid = ed.tid  
GROUP BY ed.error  
ORDER BY ed.error_type, ed.error
```

A.5 Knoten - Fehlend Sohlenkote / Deckelkote pro zugaenglichkeit

```
SELECT  
a.zugaenglichkeit,  
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND (a.sohlenkote IS NULL OR a.sohlenkote = 0) AND  
a.deckelkote IS NOT NULL THEN 1 END) AS paa_sohlenkote,
```



```
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND (a.sohlenkote IS NULL OR a.sohlenkote = 0) AND
a.deckelkote IS NOT NULL THEN 1 END) AS saa_sohlenkote,
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND (a.sohlenkote IS NULL OR a.sohlenkote = 0) AND
a.deckelkote IS NOT NULL THEN 1 END) AS unbekannt_sohlenkote,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND (a.deckelkote IS NULL OR a.deckelkote = 0) AND
a.sohlenkote IS NOT NULL THEN 1 END) AS paa_deckelkote,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND (a.deckelkote IS NULL OR a.deckelkote = 0) AND
a.sohlenkote IS NOT NULL THEN 1 END) AS saa_deckelkote,
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND (a.deckelkote IS NULL OR a.deckelkote = 0) AND
a.sohlenkote IS NOT NULL THEN 1 END) AS unbekannt_deckelkote,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND (a.sohlenkote OR a.sohlenkote = 0) IS NULL AND
(a.deckelkote IS NULL OR a.deckelkote = 0) THEN 1 END) AS paa_beide,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND (a.sohlenkote OR a.sohlenkote = 0) IS NULL AND
(a.deckelkote IS NULL OR a.deckelkote = 0) THEN 1 END) AS saa_beide,
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND (a.sohlenkote OR a.sohlenkote = 0) IS NULL AND
(a.deckelkote IS NULL OR a.deckelkote = 0) THEN 1 END) AS unbekannt_beide
FROM knoten a
GROUP BY a.zugaenglichkeit
```

A.6 Knoten - Fehlend Sohlenkote / Deckelkote pro zugaenglichkeit / funktion

```
SELECT
a.zugaenglichkeit,
a.funktion,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND (a.sohlenkote IS NULL OR a.sohlenkote = 0) AND
a.deckelkote IS NOT NULL THEN 1 END) AS paa_sohlenkote,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND (a.sohlenkote IS NULL OR a.sohlenkote = 0) AND
a.deckelkote IS NOT NULL THEN 1 END) AS saa_sohlenkote,
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND (a.sohlenkote IS NULL OR a.sohlenkote = 0) AND
a.deckelkote IS NOT NULL THEN 1 END) AS unbekannt_sohlenkote,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND (a.deckelkote IS NULL OR a.deckelkote = 0) AND
a.sohlenkote IS NOT NULL THEN 1 END) AS paa_deckelkote,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND (a.deckelkote IS NULL OR a.deckelkote = 0) AND
a.sohlenkote IS NOT NULL THEN 1 END) AS saa_deckelkote,
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND (a.deckelkote IS NULL OR a.deckelkote = 0) AND
a.sohlenkote IS NOT NULL THEN 1 END) AS unbekannt_deckelkote,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'PAA%' AND (a.sohlenkote OR a.sohlenkote = 0) IS NULL AND
(a.deckelkote IS NULL OR a.deckelkote = 0) THEN 1 END) AS paa_beide,
COUNT(CASE WHEN a.funktionhierarchisch LIKE 'SAA%' AND (a.sohlenkote OR a.sohlenkote = 0) IS NULL AND
(a.deckelkote IS NULL OR a.deckelkote = 0) THEN 1 END) AS saa_beide,
COUNT(CASE WHEN a.funktionhierarchisch IS NULL AND (a.sohlenkote OR a.sohlenkote = 0) IS NULL AND
(a.deckelkote IS NULL OR a.deckelkote = 0) THEN 1 END) AS unbekannt_beide
FROM knoten a
GROUP BY a.zugaenglichkeit, a.funktion
ORDER BY a.zugaenglichkeit DESC, a.funktion
```