SINERGISE

TopoCheck User's Manual

Client Spatial Validation Tool

DOCUMENT HISTORY

Rev.	Date	Changed/ reviewed	Modification
1.0	2008-07-21	mkadunc	Created.
1.1	2008-12-23	tcerovski	Modified. Added features from version 1.1
1.2	2009-03-23	tcerovski	Modified. All sections to do with Metada

CONFIDENTIALITY, SCOPE AND AUDIENCE

This document is classified as a public document. As such, it or parts thereof are openly accessible to anyone listed in the Audience section, either in electronic or in any other form.

This document lists and describes the functions within the TopoCheck application.

The TopoCheck User Manual is targeted to the users of the TopoCheck application and developers involved in processing of the TopoCheck results.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

REFERENCED DOCUMENTS

1| Celostna grafična podoba podjetja Sinergise
FINAL / *Sinergise Corporate Identity Graphics Design* / Sinergise_CGP_screen.pdf

2| Adobe Caslon™ Pro Release Notes
Adobe Systems Incorporated, 2000 / AdobeCaslonProReadme.pdf

3| Myriad® Pro Release Notes
Adobe Systems Incorporated, 2000 / MyriadProReadme.pdf

1 DOCUMENTATION STANDARDS

Please note that in all of this documentation, the term “error” is used to describe a geometric or attribute anomaly. It is not necessarily an error in the data as such, it is a term used when one of the SPIRE data standard tests fails.

This document should be read in conjunction with the following references:

(1) SIP-DP-011 – SPIRE Data Standard v1.0

(2) SPIRE Data Standards Implementation Rules Tolerance and Parameters (v0.7)

(3) Description of geometry errors and descriptions (set of HTML and GIF files)

2 BACKGROUND AND REQUIREMENTS

Topocheck is an advanced and sophisticated development that has been designed to assist data providers and other users to test their data against a set of standards. It is built to include the requirements listed in this section.

2.1 Non-functional requirements

TopoCheck is a Java-based program that has been implemented as a standalone application [NF001].

TopoCheck is freely redistributable and will run on the following platforms with the appropriate Java 6 run-time environment [NF002]:

- Windows 2000
- Windows XP (Standard or Professional)
- Windows Vista
- Unix
- AIX
- Linux

TopoCheck is capable of processing files that contain a large number of features with complex geometry, within an acceptable timeframe. Processing time varies considerably between datasets, depending on number of vertices and the topology of the dataset [NF005].

2.2 Geometry Validation

Currently TopoCheck only works with input in the form of ESRI shapefiles [G001] or with tables from Oracle database [D001].

TopoCheck can be run either interactively from a supplied GUI or in batch mode. However, batch mode only works on those datasets with settings provided which conform to the dataset. As there are many options available in the GUI, running batch mode will not be very useful unless there are repeated runs of similar files [G002].

TopoCheck performs a series of geometric tests on the input dataset(s). The tests are listed and described in the SPIRE Data Standard document (SIP-DP-011 – SPIRE Data Standard v1.0 (1)). It also performs some TopoCheck-specific tests (these have error codes greater than 115) [G003].

TopoCheck validates the geometry based on a set of tolerances and projection parameters in line with the paper SPIRE Data Standards Implementation Rules Tolerance and Parameters (v0.7) 0. These tolerances can be changed and set by the user via TopoCheck user interface [G004]

The tolerances and parameters for a particular dataset may also be provided as input to TopoCheck in an XML file [G005].

The tolerances and parameters for a particular dataset may be saved into the afore-mentioned XML file for re-use [G006].

The relevant coordinate system and projection for each input file is read from the input dataset (e.g. associated `.prj` file for shapefile or spatial index for Oracle table), and TopoCheck allows the appropriate tolerances and parameters to be input only in metres, using the information in the `.prj` file to convert to dataset units [G007].

TopoCheck outputs a shapefile showing the location, type and description of each error found [G008].

TopoCheck also outputs a results summary table in XML format, containing the error location and descriptions as GML tags within the XML file [G009].

The XML file is provided alongside an HTML report that is consistent with the format of the previously distributed ESRI based SVT's output HTML reports [G010].

2.3 Attribute Validation

An input attribute lookup table (as described in SIP-DP-011 – SPIRE Data Standard v1.0[1]) is available as part of the input in XML format [A001].

The attribute lookup table includes a description of any domains used, with associated domain code lists and values detailed [A002].

The attribute lookup table also includes a mapping of short field names to long field names to accommodate the input of shapefile format [A003].

This attribute lookup table is read by the application and made visible to the user via the user interface, where the user can edit the contents of this file [A004].

If this input table is not available or if the user selects this option then the application will mine the input dataset for the user and present the results via the same element of the user interface, with the edit options and the ability to save for future use. If the user is updating the dataset, it is not expected that he will mine the data each time [A005].

TopoCheck carries out a series of attribute validation checks (as described in SIP-DP-011 – SPIRE Data Standard v1.0) [A006].

If a user marks a text field as "mandatory", it cannot be null, blank or an empty string. Similarly, if a user marks a numeric field as "mandatory", it cannot be null, blank, Not-A-Number "NaN" or an empty string. It can be zero only if there is no domain, or if the domain range allows it to be.

In addition, TopoCheck checks that no Oracle reserved words are used (this is read from a list maintained in a file within TopoCheck that can be edited if necessary) [A007].

TopoCheck ensures that the attribute values conform to the type and length specified in the attribute lookup file and if a domain is prescribed, that the values are bound within the domain code list and/or values [A008].

Several domain code lists are pre-defined, as listed in SIP-DP-011 – SPIRE Data Standard v1.0 [1].

Data values	Standard
Country	ISO 3166-1
Country Subdivision	ISO 3166-2
Language	ISO 639-2

Table 1: Data conformation to ISO standards

A summary of the attribute validation results is included in the results summary table in XML format, as detailed in the geometry section. This is included in the packaged .zip file after processing as long as there are no mandatory fails in the data [A009].

2.4 Tracking, Metrics and Packaging

TopoCheck reports a summary of the metrics from the input dataset, which includes: coordinate system and projection, total area (for polygons), total perimeter, maximum bounding rectangle (MBR), total number of features and parts, and total number of vertices. This is compiled into the standard output file (XML) that can then be read and used on the load-side to help verify that the data has loaded properly. The metrics can also be viewed using the Summary tab in TopoCheck [T001].

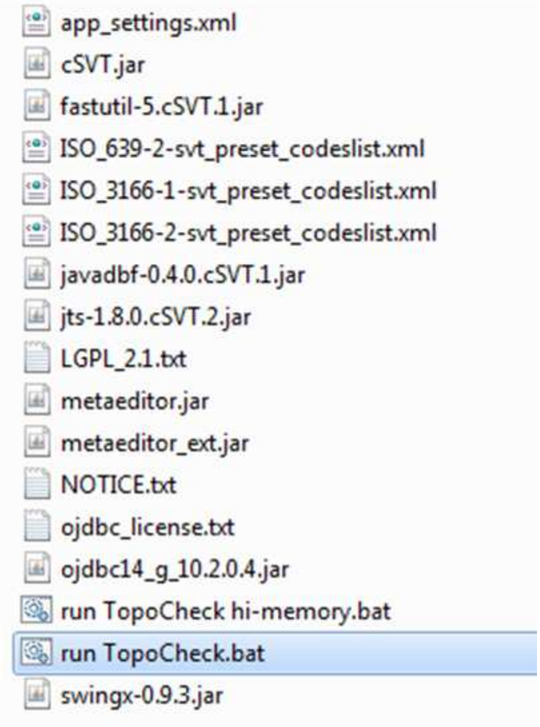
For mandatory passes, TopoCheck tool packages the data for delivery into a single compressed (`.zip`) file, along with TopoCheck results and metrics in an XML file. This does not happen if there are mandatory fails in the data [T002]. When a dataset is an Oracle table, the data itself is not packed, only results of validation.

TopoCheck saves the application settings for each dataset along with the attribute look-up table and any results and metrics files generated [T004].

New versions of the dataset with the same name will inherit the application settings of previous versions. If users want to keep separate settings for each version, they need to amend the dataset name to be different each time. [T005].

3 PROGRAM INITIATION

All the files required (except the Java 6 runtime libraries) to run TopoCheck are included in the zip file. Simply extract them to a suitable directory and you should see the files listed as in Figure 1.

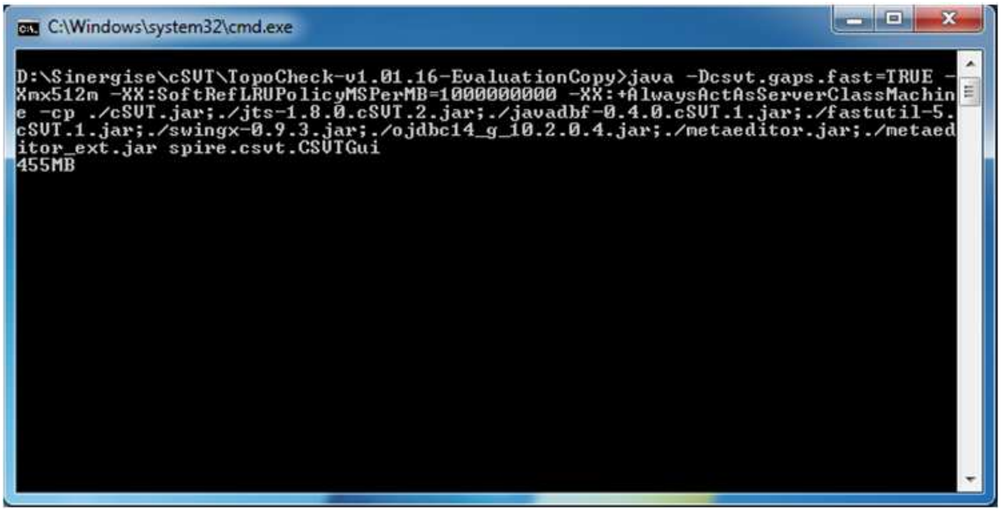


```
app_settings.xml
cSVT.jar
fastutil-5.cSVT.1.jar
ISO_639-2-svt_preset_codeslist.xml
ISO_3166-1-svt_preset_codeslist.xml
ISO_3166-2-svt_preset_codeslist.xml
javadbf-0.4.0.cSVT.1.jar
jts-1.8.0.cSVT.2.jar
LGPL_2.1.txt
metaeditor.jar
metaeditor_ext.jar
NOTICE.txt
ojdbc_license.txt
ojdbc14_g_10.2.0.4.jar
run TopoCheck hi-memory.bat
run TopoCheck.bat
swingx-0.9.3.jar
```

Figure 1: All files in TopoCheck ZIP package

The program expects Java 6 to be installed on the user’s machine. It is run by double-clicking the “run TopoCheck.bat” file. Alternatively, you may set up a shortcut to this file, and use that to start the program. Either method brings up a command line window (Figure 2), which has some background information, and should not be closed. This then launches TopoCheck graphical user interface (GUI).

When running the program on a computer with sufficient memory it can be started with “run TopoCheck_hi_memory.bat” file. Running in “hi memory" mode will use up to 1 GB of computer memory and improve performance of the validation procedures.



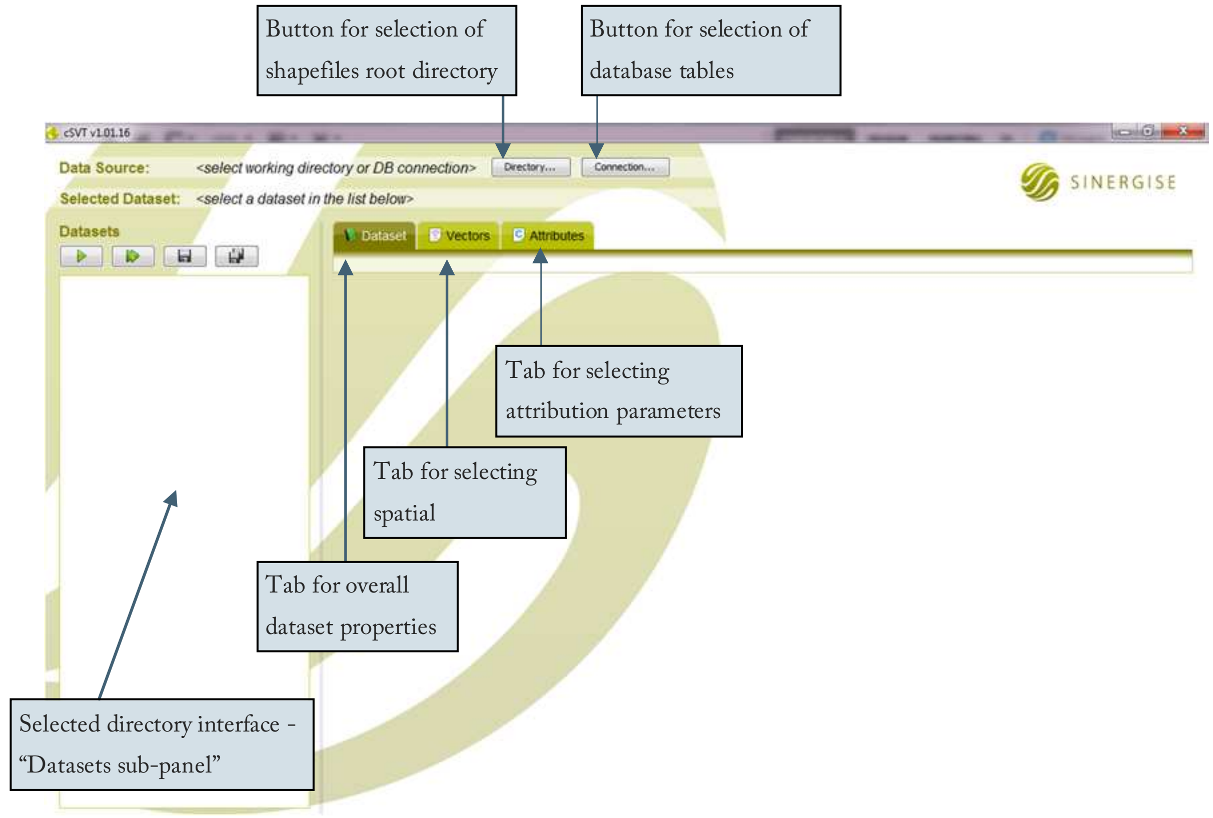
```
D:\Sinergise\cSVT\TopoCheck-v1.01.16-EvaluationCopy>java -Dcsvt.gaps.fast=TRUE -
Xmx512m -XX:SoftRefLRUPolicyMSPerMB=1000000000 -XX:+AlwaysActAsServerClassMachin
e -cp ./cSVT.jar;./jts-1.8.0.cSVT.2.jar;./javadbf-0.4.0.cSVT.1.jar;./fastutil-5.
cSVT.1.jar;./swingx-0.9.3.jar;./ojdbc14_g_10.2.0.4.jar;./metaeditor.jar;./metaed
itor_ext.jar spire.csvt.CSVTGui
455MB
```

Figure 2: TopoCheck command line window

NOTE / If the Java runtime environment's bin directory is not included in the system PATH variable, TopoCheck will not start. In Windows, this can be fixed by opening the "System Properties" dialog and using the Environment Variables button in the Advanced tab (the dialog is opened e.g. by right-clicking on "My Computer" in Windows Explorer and selecting Properties).

4 TOPOCHECK GUI – MAIN PANEL

When the GUI is first launched it looks like the example shown in Figure 3. The different parts of the GUI (e.g. Datasets box, tab areas) may be increased or decreased as preferred.



Data Source: <select working directory or DB connection>

Selected Dataset: <select a dataset in the list below>

Button for selection of shapefiles root directory
Button for selection of database tables
cSVT v1.01.16
Directory...
Connection...
SINERGISE
Datasets
Dataset
Vectors
Attributes
Tab for selecting attribution parameters
Tab for selecting spatial
Tab for overall dataset properties
Selected directory interface - "Datasets sub-panel"

Figure 3: TopoCheck Graphical User Interface

The first task is to select a data source from which one or more datasets may be chosen for running TopoCheck geometric and attribute tests. Until this is done, none of the options on the GUI are active.

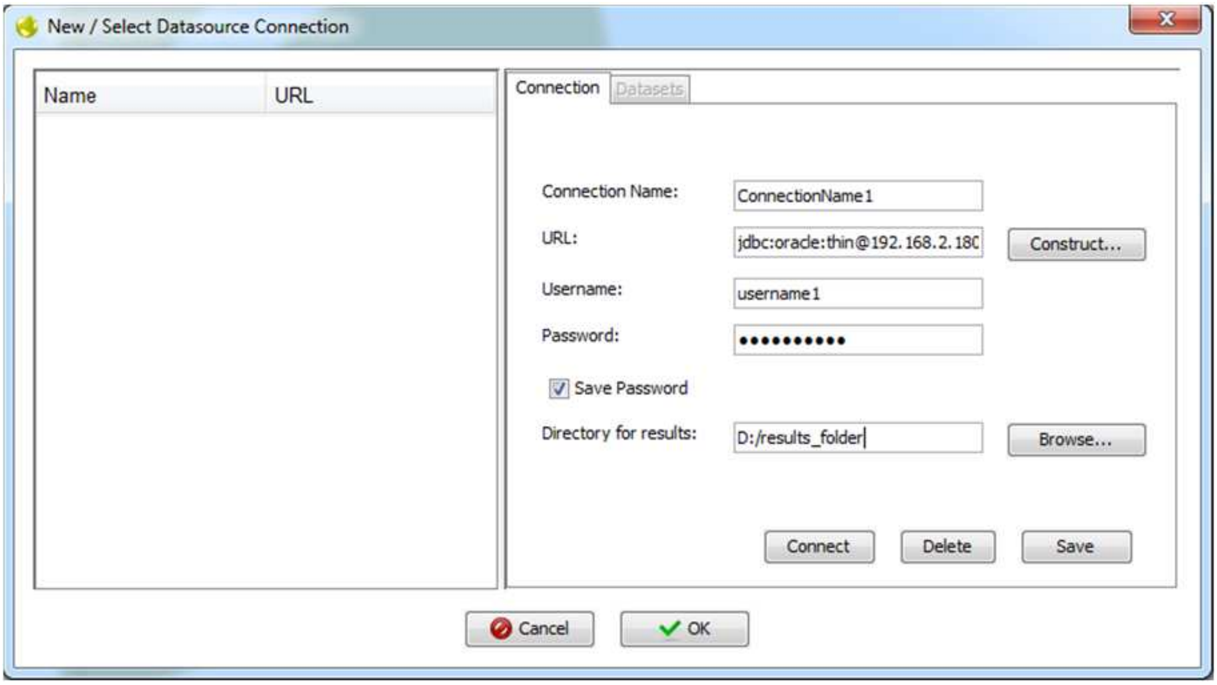
4.1 Selecting a Data Source

4.1.1 SELECTING A DIRECTORY WITH SHAPEFILES

Click on the Directory... button and navigate to the relevant directory using the usual operating system file/directory selection interface. The directory path used in any previous run is loaded as default. Click the Open button to load all shapefiles in the selected directory location and all of its sub-directories.

4.1.2 SELECTING ORACLE DB CONNECTION

Click on the Connection… button to open the dialog (Figure 4) for selecting or configuring data source connection.



Connection Name: ConnectionName1

URL: jdbc:oracle:thin@192.168.2.18C

Username: username1

Password: ••••••••••

☑ Save Password

Directory for results: D:/results_folder

Figure 4: Dialog for configuring and selecting data source connections

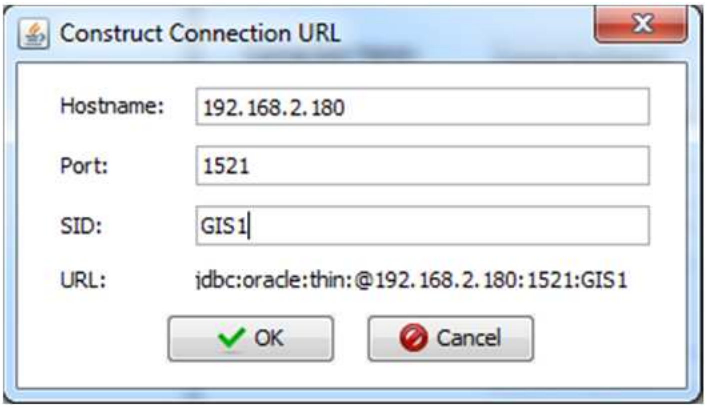
Left side of the dialog lists all configurations saved by the user. Clicking the list populates the fields on the right side with configuration values.

Connection configuration consists of the following items:

- **Connection Name** – user specified name to identify the connection in the list.
- **URL** – Specifies location of the database. Clicking the Construct… button will open a dialog (Figure 5) that will help constructing the URL with specified connection parameters:
  - **Hostname** – The host name where Oracle server is running
  - **Port** – The port number where Oracle is listening for connection. Default is 1521.
  - **SID** – System ID of the Oracle server database instance.

  While typing in the URL parameters, the URL value will be updated automatically. On pressing the OK button this value will be transferred into the URL field of the configuration dialog.

- **Username and Password** - The Oracle server login username and password to use. Ticking the Save Password checkbox will save the password in the configuration file.
- **Directory for results** – Folder where validation results for all datasets in this data source will be placed. Clicking the Browse… opens the usual operating system directory selection interface.



Construct Connection URL

Hostname: 192.168.2.180

Port: 1521

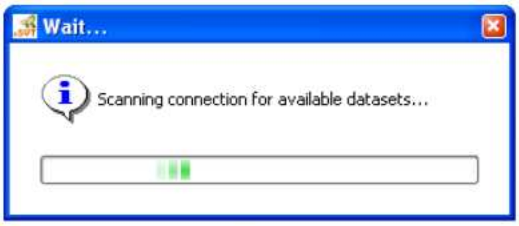
SID: GIS1

URL: jdbc:oracle:thin:@192.168.2.180:1521:GIS1

Figure 5: Dialog for construction connection URL

When creating a new data source configuration, the user must select datasets for validation. This requires connecting to database by clicking the Connect button, which will start the database scanning process that will scan all tables, views and synonyms in the user schema to find those that have a Geometry column and can therefore be used as a dataset for validation in TopoCheck. If a table with more than one Geometry column is found, the tool will split this table into more datasets, each containing one of the Geometry fields and all other attribute fields. Duration of the scanning process depends on the number of tables in the user schema. During the process an indeterminate progress dialog will be displayed (Figure 6) blocking all user actions.

In case of failing to connect to the database, an error message dialog will be displayed reporting the cause of failure, in which case the user should check if the connection parameters are correct and if the database server is accessible from his/her location.



Wait…

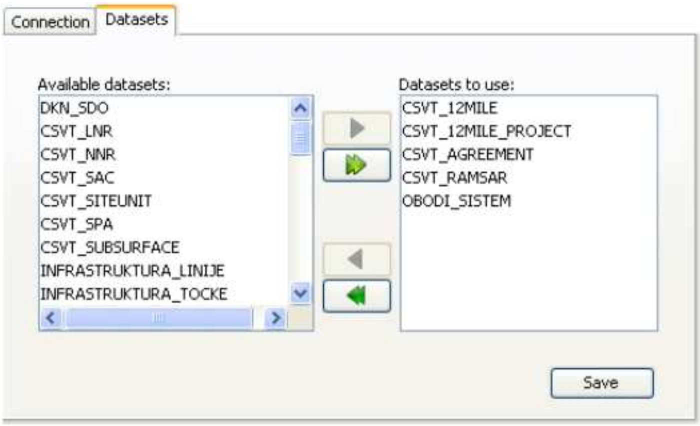
Scanning connection for available datasets…

Figure 6: Progress dialog displayed while scanning the database

When the scanning process is completed, the progress dialog will close and the Datasets tab will be enabled and selected. The left side of the panel lists available datasets found by the scanning process and the right side lists datasets selected for use. For the data source configuration to be valid, at least one dataset must be selected. Four arrow buttons between the lists are used to select and deselect datasets.

Selecting or removing datasets from the working list can be also done later on existing data source configurations. The procedure is the same as when creating a new configuration.

When the data source configuration is finished, it can be saved for later use by clicking the Save *button. Saving the connection will validate it first and an error message will be displayed in case of invalid parameters. If a configuration with the same name already exists (this is also the case when editing an existing configuration), the user will be prompted to confirm overwriting an existing configuration.*



Available datasets:

DKN_SDO
CSVT_LNR
CSVT_NNR
CSVT_SAC
CSVT_SITEUNIT
CSVT_SPA
CSVT_SUBSURFACE
INFRASTRUKTURA_LINIJE
INFRASTRUKTURA_TOCKE

Datasets to use:

CSVT_12MILE
CSVT_12MILE_PROJECT
CSVT_AGREEMENT
CSVT_RAMSAR
OBODI_SISTEM

Figure 7: Working datasets selection tab

To delete a configuration from the list of saved configurations, load it first by selecting it in the list and then clicking the Delete button. Deleting an unsaved configuration will have no effect.

When satisfied with the data source configuration, the user will confirm the selection by clicking the OK button. Clicking the Cancel button exits the dialog without selecting a data source.

After selecting an Oracle data source, the tool will analyze all datasets to determine their properties (last modification time, number of records, constraints and projection format). During this process an indeterminate progress dialog will be displayed (similar to one in Figure 6). Duration of the analysis process depends on the number of datasets and their size. When finished, the Datasets list on the main GUI panel will be populated with datasets from the selected data source.

TopoCheck validation processes on Oracle datasets will perform significantly better when datasets are represented by **tables** (not views or synonyms) that have a (non-composite) **primary key** constraint and a spatial **index** on the geometry column.

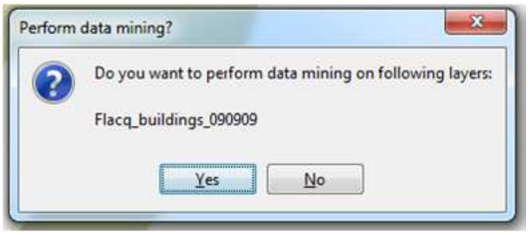
If a database connection is lost during scanning, analyzing or validation, a dialog will be displayed counting down to retry connecting. After ten failed attempts to re-connect, the application will wait for user input to retry or cancel the process.

4.2 Data Mining

The first time a dataset is opened in TopoCheck, the user is presented with an option to perform data mining (Figure 8). The data mining procedure automatically determines the following properties of an attribute:

- **Field properties** – name, data type, length and decimals
- **Domain boundaries** – by finding the minimum and maximum values.
- **Domain code list** – if the field contains less than 100 unique values, these are saved as the field's code list.
- **Uniqueness** – when all field values are distinct.
- **Obligation** – when none of the field values are empty, the Mandatory property of an attribute is set to true.

Mining can be performed all non-attributed dataset at once. Clearly, the first time the user uses TopoCheck there is considerable advantage in mining the data. The mined parameters will be saved between runs (in the directory where the data was placed). If any datasets are updated in that data source, the user can use previously stored mined parameters to check the updated dataset against (so need to be careful not to re-mine the data as it will over-write populated parameters from previous versions).

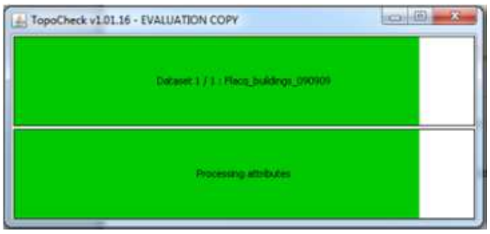


Perform data mining?

Do you want to perform data mining on following layers:

Flacq_buildings_090909

Figure 8: Initial data mining dialog for new datasets



TopoCheck v1.01.16 - EVALUATION COPY

Dataset 1 / 1 : Flacq_buildings_090909

Processing attributes

Figure 9: Data mining progress dialog

If the mining option is chosen, a progress dialog is shown that goes through the mining of each dataset specified (Figure 9). The attribution parameters for older versions of the datasets (assuming the names have not changed) are over-written. (Note: attribute mining can be carried out later in the Attributes tab.)

After data source selection and the optional data mining, the datasets in the selected directory are displayed in the Datasets section of the user interface.

4.3 The Datasets Listing

All datasets provided by the selected data source are added to the Datasets listing on the left hand side of the interface as shown above. The list displays the title of the dataset or the name of the dataset if title is not available (displayed in italic).

Clicking in the list selects a dataset for running TopoCheck tool and setting its Geometric and Attribution parameters. The selected dataset's title is displayed as the "Selected Dataset" in the upper section of the GUI. It is also highlighted in the list. There are four buttons at the top of the Datasets sub-panel, which have active tooltips associated with them (Figure 7).

The first two green arrow buttons run the validation programs for geometric and attribute errors (using the geometric and attribute parameters set in the Vectors and Attributes tabs). The first option (Check Selected) runs the validation tool on only the dataset selected (highlighted) in the Datasets selection sub-panel. The second option (Check All) runs the tool on all the datasets listed in the Datasets sub-panel.

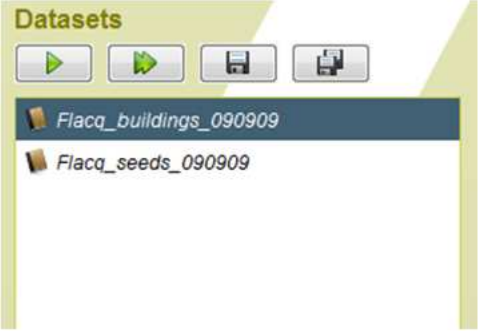
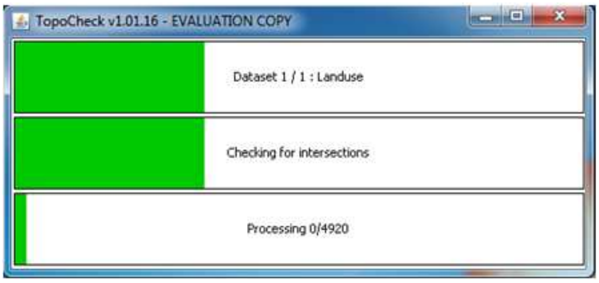


Figure 10: Datasets in the selected directory structure are listed in the Datasets listing

After clicking either of first two buttons a validation progrees dialog window is open (Figure 11). This may take a few seconds or minutes, depending on size of datasets.



Dataset 1 / 1 : Landuse

Checking for intersections

Processing 0/4920

Figure 11: Validation progress dialog

The remaining two buttons enable saving the geometry and attribute parameters for a dataset into an external XML file. This file may be retrieved whenever that dataset is re-loaded into TopoCheck toolbox. The first button (Save Selected) enables you to save parameters for the selected dataset only while the second option (Save All) saves parameters for all the datasets listed in the Datasets sub-panel. (Check Selected and Check All buttons also save the parameters before running validation, same as Save buttons)

Both the Check all and Save all buttons are disabled if any of the listed datasets has errors (i.e. is highlighted red).

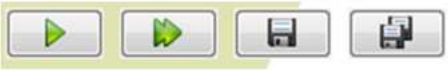


Figure 12: Buttons for running validation and saving datasets

Any dataset that has Oracle reserved words in the attribute names is highlighted red and cannot be checked further until the issues are resolved.

Datasets that do not conform to the provided settings XML (i.e. fields that are different to those in the settings, in terms of field order, presence, name, type, length and number of decimals) are also highlighted red and cannot be processed further. If a user has kept the same dataset name, but has changed the schema, then he/she should override the older inherited settings by performing data mining on the invalid dataset.

Datasets with no primary key assigned cannot be processed until one of its unique attribute fields is defined as a primary key (see Section 8.1).

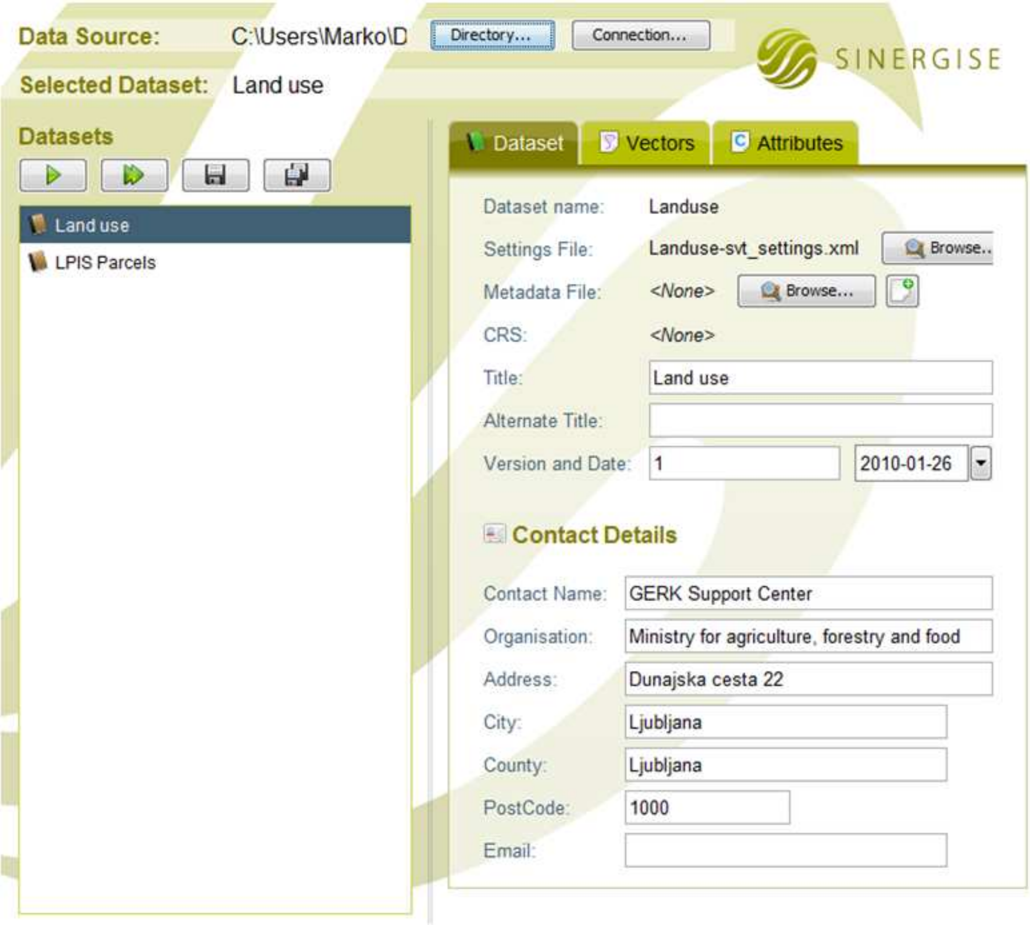
If multiple geometry types (e.g. points and polygons) are found in a dataset it is considered invalid.

The Datasets section of the interface may be widened if required by moving the mouse over the dividing line on the right hand side of the list. When it turns into a double-sided arrow, click and drag to enlarge the section.

5 DATASET TAB (OVERALL DATASET PARAMETERS)

The Dataset tab allows the user to set various parameters relating to the selected dataset. These parameters include:

- A Settings File that is associated with the dataset (which could be a mined one)
- CRS (Coordinate Reference System)
- Title
- Alternate title
- Version and Date
- Contact details of the data provider (including Organisation, Address, City, County, Postcode and Email address)
- Users may also specify a version and a date for the dataset.



Data Source: C:\Users\Marko\D

Selected Dataset: Land use

Land use
LPIS Parcels

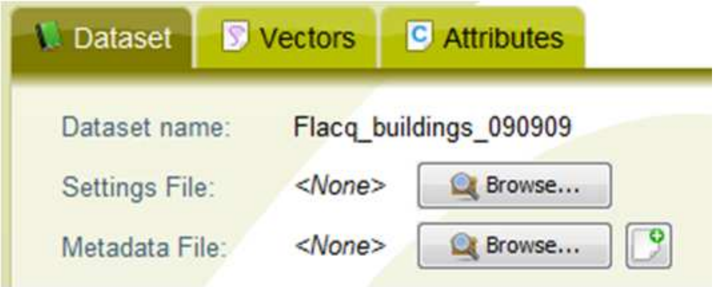
Dataset name: Landuse
Settings File: Landuse-svt_settings.xml
Metadata File: <None>
CRS: <None>
Title: Land use
Alternate Title:
Version and Date: 1 2010-01-26

Contact Name: GERK Support Center
Organisation: Ministry for agriculture, forestry and food
Address: Dunajska cesta 22
City: Ljubljana
County: Ljubljana
PostCode: 1000
Email:

Figure 13: Dataset information tab

The **Settings File** can be selected by using the Browse... button and the normal directory/file selection GUI.

The **Metadata File** and Parameters can be selected by using the Browse button Browse... to look at pre-defined metadata or the New button to create a new metadata file. Chapter 6 gives a fuller explanation of metadata entry and options.



Dataset name: Flacq_buildings_090909
Settings File: <None> Browse...
Metadata File: <None> Browse...

Figure 14: Metadata File and Parameters

If you select the Browse option then you are asked to select an existing metadata file (XML extension) using the normal directory/file selection GUI. Once this metadata file has been loaded it is displayed as shown in Figure 15 with three buttons to enable the following actions:

- **Open and Edit MetaData Document**

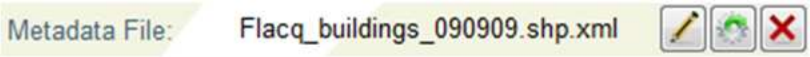
  This opens the MetaData Editor (see Chapter 6) to enable the user to view and edit existing metadata for the selected dataset.

- **Update MetaData with Current Dataset Properties**

  This option updates metadata by using properties of the currently loaded dataset e.g. projection and extent information. A PRJ file must be present in order for the projection information to be read and data extent is only updated if the units of the dataset are geographic (latitude/longitude).

- **Remove MetaData Document Reference**

  This removes the currently loaded metadata document and the GUI returns to the state shown in Figure 14.



Metadata File: Flacq_buildings_090909.shp.xml

Figure 15: Metadata Options on a Selected Metadata File

The Edit ✎ and Update ⚙ buttons both open the MetaData Editor, which is described more fully in Chapter 6 (Metadata Tab- Dataset metadata).

CRS is mined from the projection information (`*.prj`) provided in the shapefile or from SRID defined in the Oracle spatial index. If none is available, a CRS of "*<None>*" is reported.

The **Alternate Title**, **Version** and all **Contact Details** are effectively freeform text fields, with no validation. Obviously as the user adds all these details, he should save them regularly. The saved parameters are retrieved whenever that dataset is selected again.

6 DATASET METADATA EDITOR

The metadata attached to a dataset is a considerable requirement. Under accepted and emerging ISO standards, metadata has several aspects.

On the main GUI, when you click the Browse... button after Metadata File, or the Edit... and Update... buttons on a previously selected metadata file, you are presented with the following Metadata Editor.

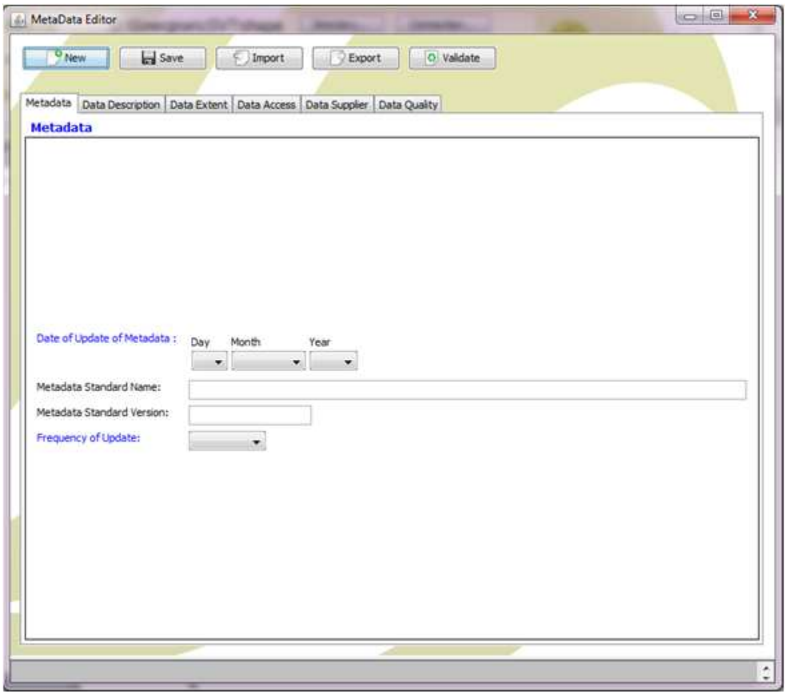


Figure 16:Metadata Editor

The **Metadata Editor** allows the user to create, import, edit and save various parameters relating to the metadata of the selected dataset. There are five control buttons: New, Save, Import, Export and Validate which relate to metadata actions for the selected dataset. There are also six tabs which relate to different pages (or metadata categories) for the dataset. These include: Metadata, Data Description, Data Extent, Data Access, Data Supplier, and Data Quality.

The metadata actions are described below:

- **New**

  This option creates a new empty metadata file which opens the Metadata tab and, by default, adds the current date. The user can then use the other tabs (e.g. Data Extent, Data Quality) to add further metadata information.

- **Save**

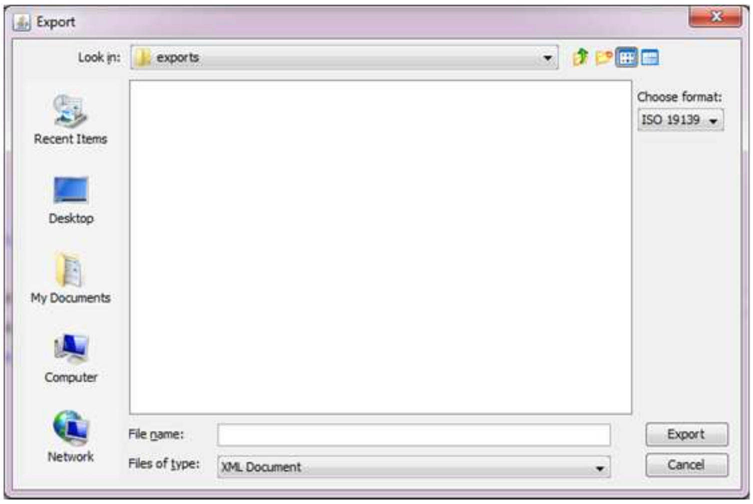
  This allows the user to save his metadata file to a specified directory location. If the data has not been validated (see below) then the user is asked if he would like it to be validated before saving.

- **Import**

  This allows the user to import an existing metadata file. When this is done all metadata elements (on all six tabs) are filled in from information residing in the imported XML file.

- **Export**

  This allows the user to export all existing metadata settings to a new metadata file. If the metadata elements have passed validation (see below) then the user is asked to enter a name and location for the new output file. In addition, he has the option of setting the export format to either ISO 19139 or UK Gemini. If any of the metadata elements do not pass validation then the user is informed of this then asked if he would like to export anyway. If he responds "No" to this question he is then directed through the whole data validation process described below.



Export

Look in: exports

Choose format: ISO 19139

File name:

Files of type: XML Document

Figure 17: MetaData Export

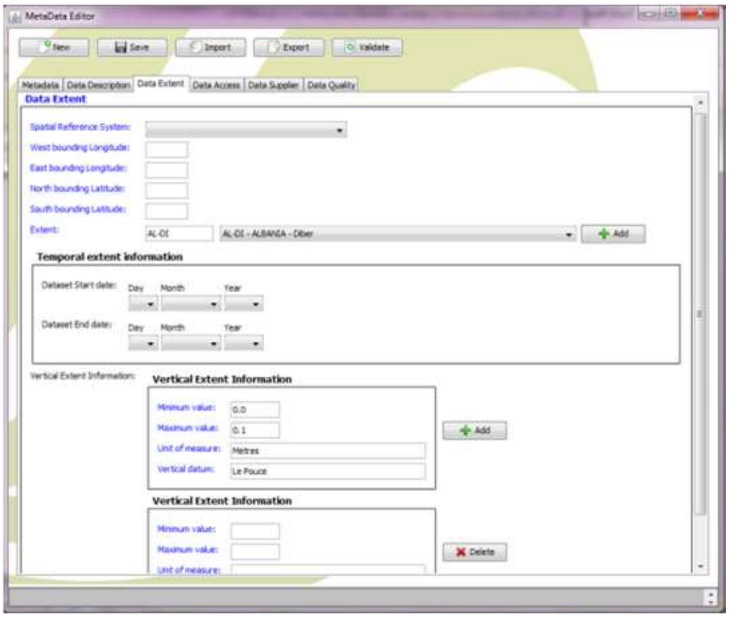
- **Validate**

  This allows the user to validate all metadata values which he has entered and specifically to check that all mandatory fields have been filled in. The user is then directed (one by one) through all mandatory fields which do not have a value and asked to provide one.

As already mentioned, each of the six metadata tabs opens a different GUI for inputting/editing metadata information. These GUIs are to a certain extent self-explanatory, but some important generic comments about their behaviour and design are presented below.

(1) Add.. button to enable user to add more than one value for a field. For example, when the Add button is pressed after the "Alternative Title" field, another data input field is opened to allow the user to add another data title, in this case "Test Dataset 3". Once this is added, a Delete.. button enables this entry to be removed.

(2) Certain Add.. buttons refer to combined fields rather than individual ones.
For example, on the "Vertical Extent Information" on the "Data Extent" tab, the "Add" button, allows the user to add a combination of any or all of "Minimum Value; Maximum Value; Unit of Measure; Vertical datum"



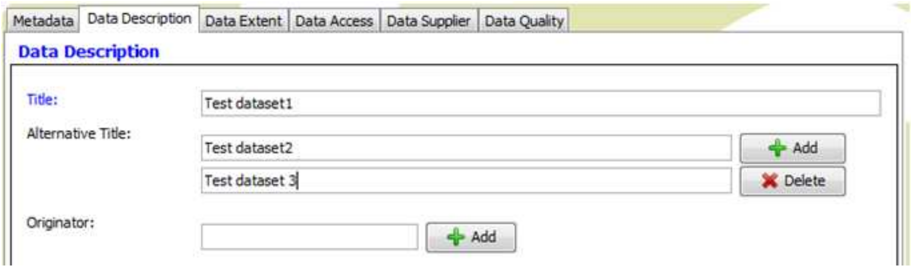
Extent: AL-DI AL-DI - ALBANIA - Diber

Minimum value: 0.0
Maximum value: 0.1
Unit of measure: Metres
Vertical datum: Le Pouce

Figure 18: Example of Add for multiple fields (Vertical Extent Information) on the Data Extent tab

Similarly, multiple data suppliers can be added on that particular tab.

(3) Mandatory fields are labelled in blue while optional fields are shown in black. The user must always enter (or select) a value for a mandatory field. If he does not then an error is given during the data validation phase.



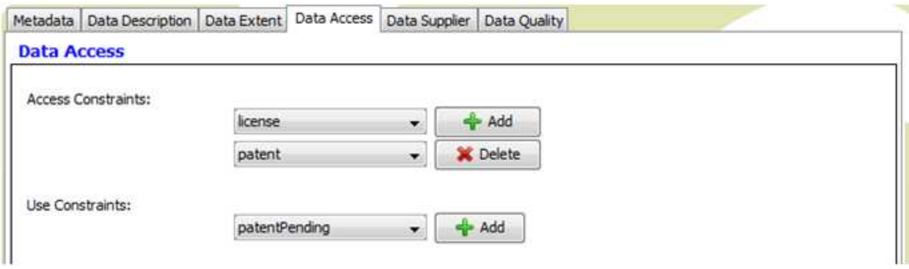
Title: Test dataset1

Alternative Title: Test dataset2
Test dataset 3

Originator:

Figure 19: Example of Add Button and Mandatory Fields

(4) Field values may be controlled by the use of a drop down list. Here the user selects an option from a range of permissible values (e.g. from ISO standards or other). This is often combined with the use of the Add.. button (Figure 18) to enable the user to choose more than one value from the list.

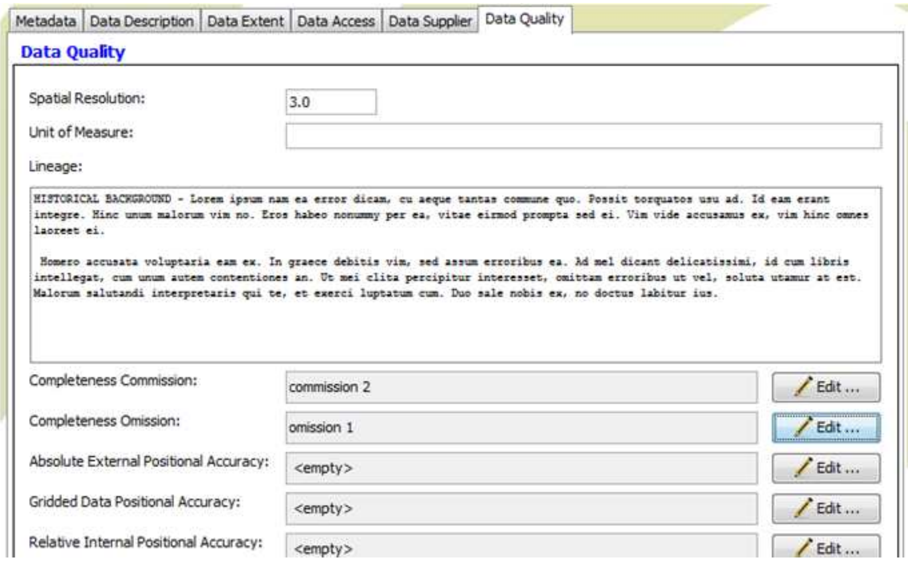


Access Constraints: license, patent

Use Constraints: patentPending

Figure 20: Example of Drop Down List

(5) Some GUIs are nested within others. The Data Quality GUI is shown in Figure 21. When the Edit.. button is selected for any of the 11 aspects of data quality (e.g. Completeness Commission) it opens up a new GUI as shown in Figure 22. This GUI is the same for each of the 11 data quality elements.



Spatial Resolution: 3.0

Unit of Measure:

Lineage:

```
HISTORICAL BACKGROUND - Lorem ipsum nam ea error dicam, cu aeque tantas commune quo. Possit torquatos usu ad. Id eam erant
integre. Hinc unum malorum vim no. Eros habeo nonummy per ea, vitae eirmod prompta sed ei. Vim vide accusamus ex, vim hinc omnes
laoreet ei.

 Homero accusata voluptaria eam ex. In graece debitis vim, sed assum erroribus ea. Ad mel dicant delicatissimi, id cum libris
intellegat, cum unum autem contentiones an. Ut mei clita percipitur interesset, omittam erroribus ut vel, soluta utamur at est.
Malorum salutandi interpretaris qui te, et exerci luptatum cum. Duo sale nobis ex, no doctus labitur ius.
```

Completeness Commission: commission 2

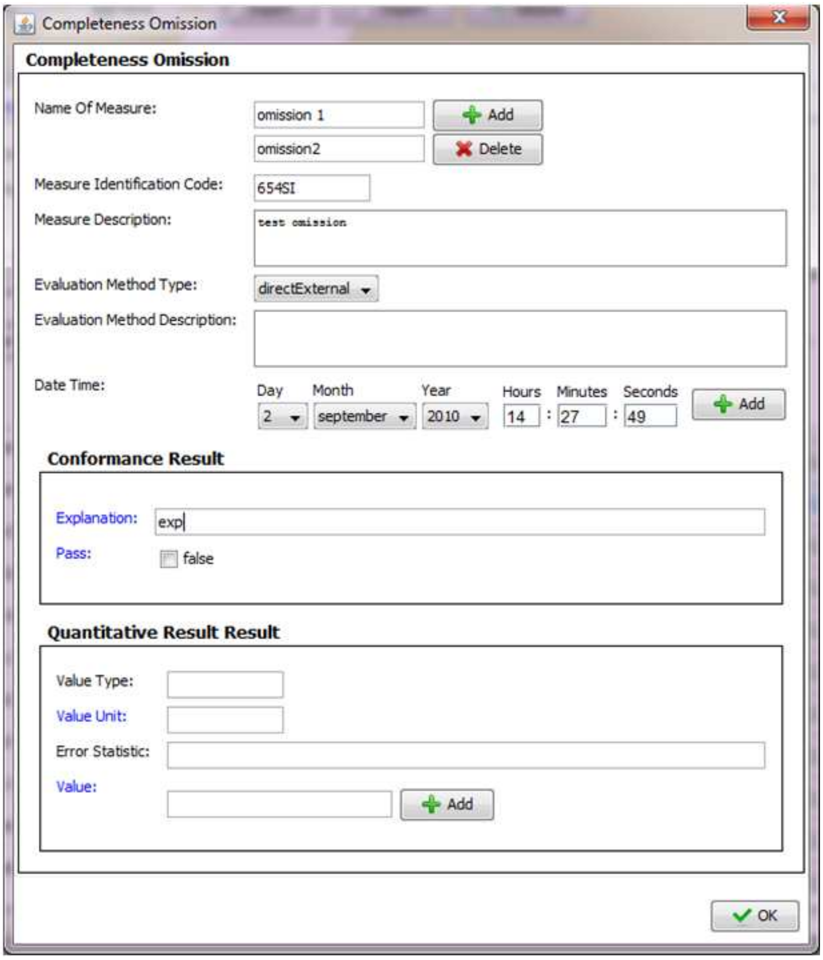
Completeness Omission: omission 1

Absolute External Positional Accuracy: <empty>

Gridded Data Positional Accuracy: <empty>

Relative Internal Positional Accuracy: <empty>

Figure 21: Data Quality GUI



Completeness Omission

Completeness Omission

Name Of Measure: omission 1
omission2

Measure Identification Code: 654SI

Measure Description: test omission

Evaluation Method Type: directExternal

Evaluation Method Description:

Date Time: Day 2, Month september, Year 2010, Hours 14 : Minutes 27 : Seconds 49

Conformance Result

Explanation: exp

Pass: □ false

Quantitative Result Result

Value Type:

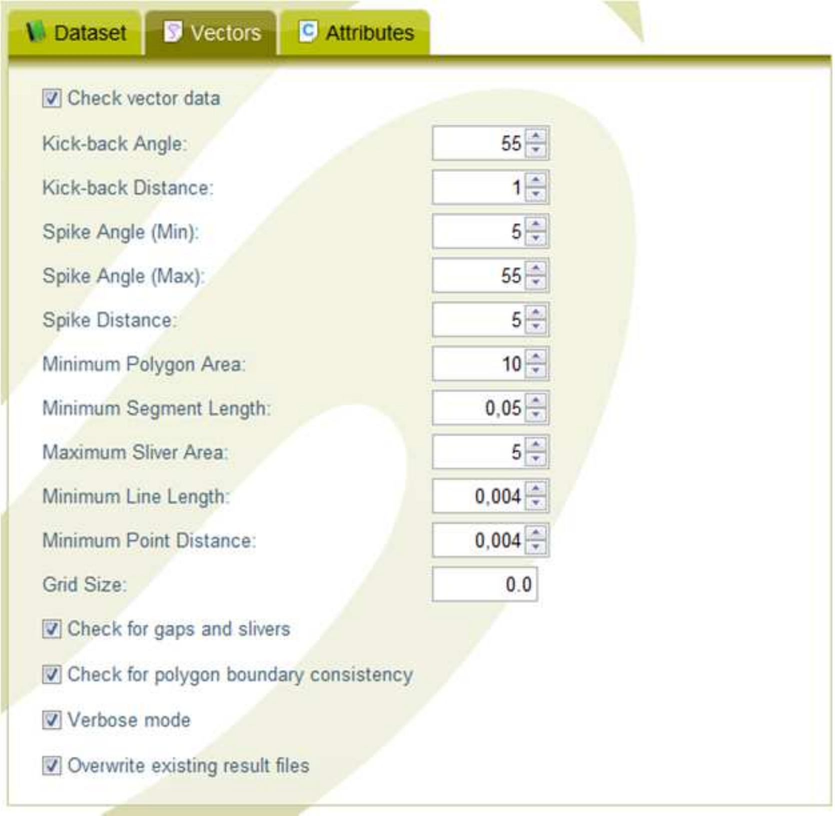
Value Unit:

Error Statistic:

Value:

Figure 22: Completeness Commission GUI

7 VECTOR TAB (GEOMETRY CHECKS)



☑ Check vector data

Kick-back Angle:	55
Kick-back Distance:	1
Spike Angle (Min):	5
Spike Angle (Max):	55
Spike Distance:	5
Minimum Polygon Area:	10
Minimum Segment Length:	0,05
Maximum Sliver Area:	5
Minimum Line Length:	0,004
Minimum Point Distance:	0,004
Grid Size:	0.0

☑ Check for gaps and slivers

☑ Check for polygon boundary consistency

☑ Verbose mode

☑ Overwrite existing result files

Figure 23: Geometry Validation Parameters

This tab allows the user to set the parameters on each of the geometric tests for each of the datasets being tested. The values of these parameters may be amended by typing direct into the relevant box, or by using the up/down (spinner) buttons to increment (increase or decrease) by pre-set amounts. The nature of each geometric test is described fully in Ref [3]. The default values and units of measurement are described in Table 3.

The aim of setting these parameters is to test the geometry of the selected dataset(s) and report on any instances where angles and maximum values have been exceeded and where there are values which are lower than minimum settings. These anomalies are summarised as vector errors (e.g. intersection, gap. sliver, etc) some of which constitute a mandatory fail in the data.

If the selected dataset's projection information indicates that the data is stored in a geographic coordinate reference system (lat, lon) or in a system with units other than metres, the parameters are still input in metres, but are internally converted to appropriate units (degrees or similar). The geometric checks themselves are always performed in the dataset's original coordinate systems, without applying projections or unit conversions.

Table 2: Parameters for Geometric Tests

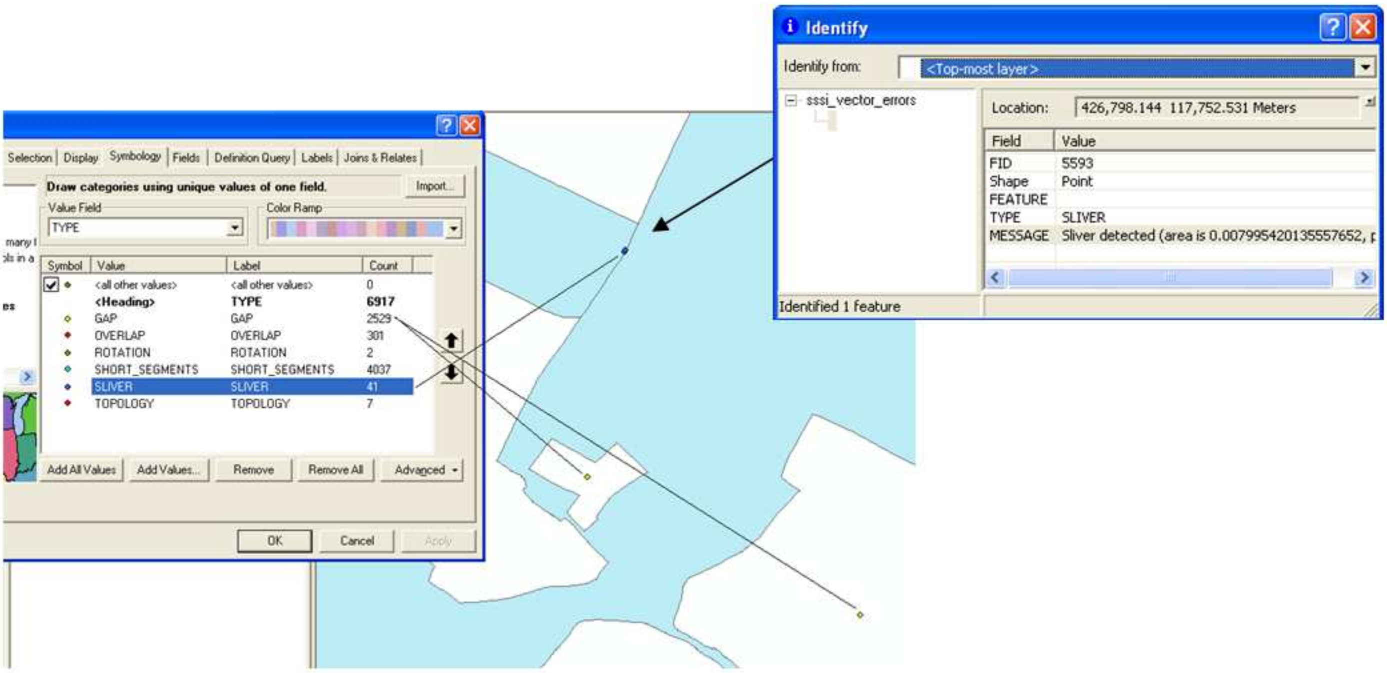
Check Vector Data	**If ticked on, then vector tests will be run on this dataset when the Check Selected or Check All buttons are pressed.**
Kick-Back Angle: *(degrees)*	The angle in degrees defined within the kick-back, as defined in Appendix A. The default is 5°, but has a range between 0.1° and 55°
Kick-back Distance: *(metres)*	The length of the kick-back in metres. The default is 1.0 but has a range between 0.0001 - 15000.0
Spike Angle (Min): *(degrees)*	The minimum spike angle in degrees (i.e. the angle below which a spike is detected regardless of the length of adjacent line segments). The default is 5°, but has a range between 0.1° - 30°
Spike Angle (Max): *(degrees)*	The maximum spike angle in degrees (i.e. the angle below which a spike is detected if the distance of the spike is less than specified). The default is 55°, but has a range between 0.1° -90°
Spike Distance: *(metres)*	The maximum length of the spike in metres (only used when the angle is greater than minimum spike angle). The default is 5m, but has a range between 0.0001 – 15000m.
Minimum Polygon Area: *(square metres)*	The minimum size in square metres. The default is 10 square m, but has a range between 0.00001 – 10000000 square m.
Minimum Segment Length: *(metres)*	The minimum length of a line segment in metres. The default is 0.05m, but has a range between 0.00001 – 10000000m
Maximum Sliver Area: *(square metres)*	The maximum size of a sliver in square metres. The default is 5 square m, but has a range between 0.00001 – 10,000,000,000 square m.
Minimum Line Length: *(metres)*	The minimum length of a line in metres. The default is 0.004m, but has a range between 0.00001 – 10000000m.
Minimum Point Distance: (metres)	The minimum distance between 2 consecutive points in a linear or polygon shapefile in metres. The default is 0.004m, but has a range between 0.0001 – 10000000m.
Grid Size: (metres)	The snapping grid size in metres, which will round all coordinates in the dataset to a specified value prior to processing. Default is 0.0, which will cause no changes to the input coordinates.
Check for gaps and slivers (on/off)	If ticked on, test for gaps and slivers will be performed, otherwise this test will be skipped.
Verbose mode: (on/off)	This option, if ticked on, will show progress in the DOS window.
Overwrite existing result files: (on/off)	This option, if ticked on, will allow you to overwrite the existing result files should you run the tool on the same dataset again. If you do not select this, and previous results exist in the specified directory, an error warning is given. Existing files are not overwritten unless the user specifically requests it.

The results of this geometry check are summarised on screen or in more detail as a point shapefile (`*_vector_errors.shp`) where the location of each error is displayed, and its type is denoted by an attribute. These shapefiles have to be viewed and represented in your standard GIS desktop tools.

Each point has five attributes, the first two (FID and Shape) being default shapefile attributes. The three important attributes are:

- FEATURE, in the form of `<FEATURE>.<PART>#<VERTEX>` - e.g. `3122.2#5`. The first number (before '`.`') is feature id, number after the '`.`' is part number, and the number after the '`#`' is the vertex number to which the error applies. Some errors apply only to whole geometries, so the part number and vertex number are omitted. If an error applies to more than one geometry, identifiers of geometries are joined with '|' character. Some errors do not apply to any geometry (e.g. "Gap detected"), in which case the FEATURE field is empty. Note that due to automatic corrections of the feature geometries in case of invalid topology, the part and vertex identifiers might not be coincident with the shapefile's internal ordering.
- TYPE, e.g. Short segments, Overlaps, Gaps, etc.
- MESSAGE, e.g. "Segment too short (0.03456)", Intersection detected", "Gap detected (area is 1320.81648851512)", etc

For consistency with previous versions of the SVT and distributed "error" files, the codes in the output XML file for each error are maintained as before, and as listed in Table 3.



Draw categories using unique values of one field.

Value Field: TYPE

Symbol	Value	Label	Count
☑ ●	<all other values>	<all other values>	0
	<Heading>	**TYPE**	**6917**
◇	GAP	GAP	2529
●	OVERLAP	OVERLAP	301
●	ROTATION	ROTATION	2
●	SHORT_SEGMENTS	SHORT_SEGMENTS	4037
●	SLIVER	SLIVER	41
●	TOPOLOGY	TOPOLOGY	7

Identify from: <Top-most layer>

sssi_vector_errors

Location: 426,798.144 117,752.531 Meters

Field	Value
FID	5593
Shape	Point
FEATURE	
TYPE	SLIVER
MESSAGE	Sliver detected (area is 0.007995420135557652, p

Identified 1 feature

Figure 24: Error shapefile represented appropriately and examined in ArcGIS desktop tools

Table 3: List of geometric checks

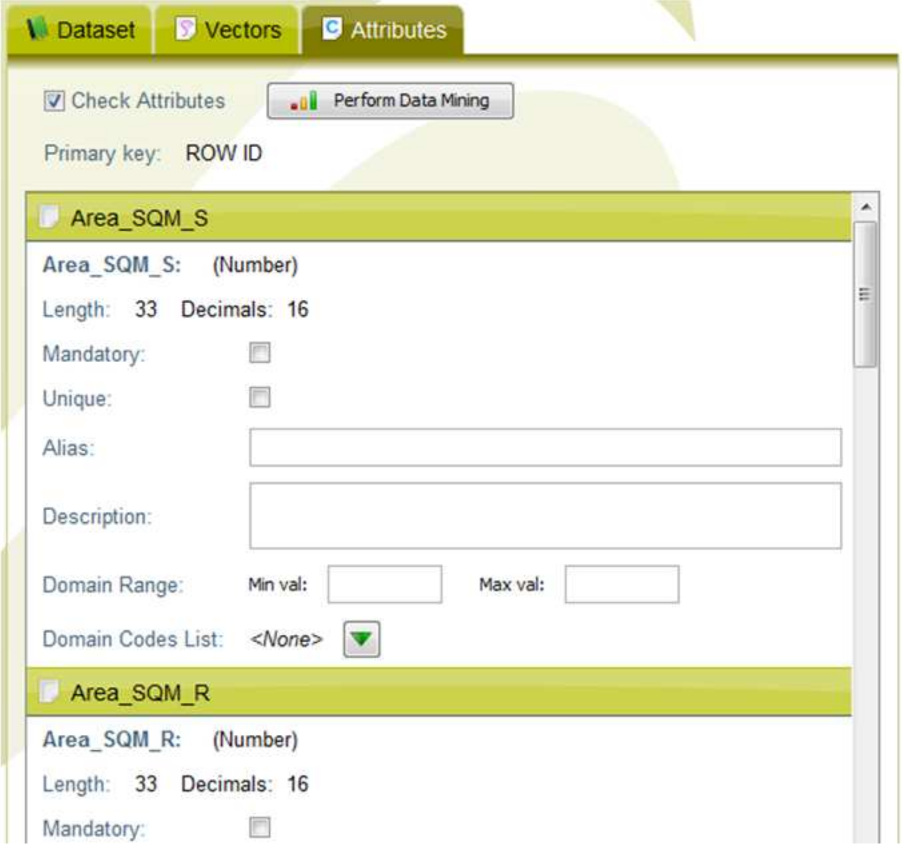
Code	Test Name	Description	Conformance
101	Loop backs	Loop backs - self intersections (Termed 'Butterfly' polygons).	Pass
102	Unclosed Polygons	Unclosed Polygons/Rings - The start node and end node of the polygon or ring is not the same. This means that the feature cannot be closed.	Fail
103	Internal Polygons with Incorrect Rotation	Requirement for the internal polygon and the external polygon to have the order of nodes or vertices in a specific rotation direction. The external polygon should be clockwise and the intemal polygon should be counter clockwise.	Pass
104	Duplicated Points	A point that duplicates exactly the same X, Y coordinates as another point.	Pass
105	Kick Backs	Digitising error leading to an inconsistency in the line.	Pass
106	Spikes	Digitising error leading to a spike inconsistency in the line. Similar to kick backs.	Pass
107	Small Areas	A polygon feature should not be less than a specified area.	Pass
108	Slivers	Very small gaps between the boundaries of adjacent polygon features.	Pass
109	Overlapping Polygons	An overlap of one polygon or line feature onto another.	Pass
110	Duplicate Features	A feature that duplicates exactly the same geometry and attribution as another feature.	Pass
111	Short Segments	A very short distance between two nodes or vertices. This distance is specified and would be expected to be the same as the cluster tolerance on the dataset.	Pass
112	Null Geometry	No geometry is held against an attribute (Table records with Null Shape).	Pass
113	Segment Orientation	Similar to Ring / Polygon rotation but at a finer granularity. The rotation between two nodes or vertices is checked rather than the entire feature.	Pass
114	Empty Parts	Similar to null geometry. One geometry in a multipart feature is empty.	Pass
115	Near Points	A very short distance between two points.	Pass
116	Gaps	Large areas/holes not covered by any polygon (Error in coverage features, where complete coverage of land is desired).	Pass
117	Invalid coordinate	Invalid coordinate numeric value (NaN or Infinity).	Pass
118	Topologically Invalid Feature	Feature has an invalid topology (NESTED_SHELLS, DISCONNECTED_INTERIOR...).	Pass
119	Short Line	Length of a linear feature is smaller than a specified length.	Pass

8 ATTRIBUTE TAB (ATTRIBUTE CHECKS)

The attributes of any particular dataset can be seen by:

- Clicking on the relevant dataset;
- Clicking on the Attributes tab;
- Scrolling up/down to see the mined attribute values.

A typical interface is shown in Figure 25. There are three options above the list of attributes: a Check Attributes checkbox, a Perform Data Mining button and Primary key definition. If Check Attributes is ticked off, then attribute data checking will not be carried out during data validation.



☑ Check Attributes

Primary key: ROW ID

Area_SQM_S

Area_SQM_S: (Number)

Length: 33 Decimals: 16

Mandatory: □

Unique: □

Alias:

Description:

Domain Range: Min val: Max val:

Domain Codes List: <None>

Area_SQM_R

Area_SQM_R: (Number)

Length: 33 Decimals: 16

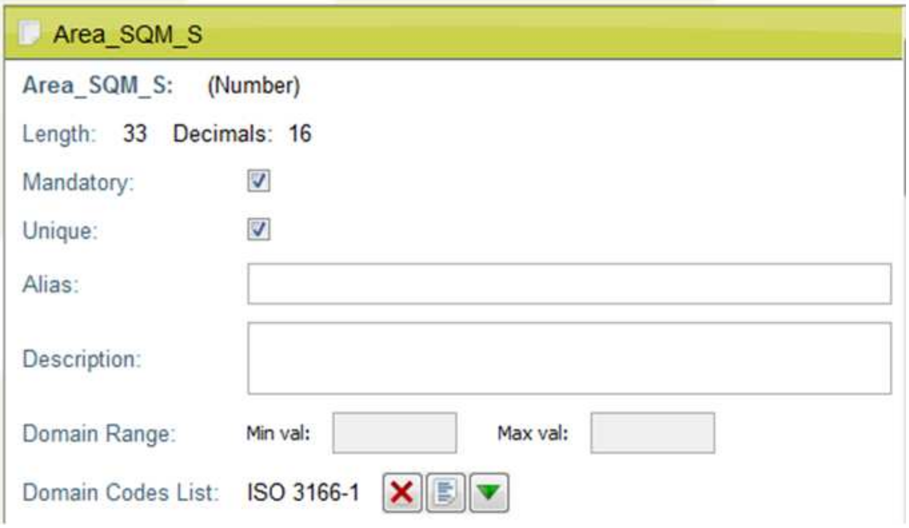
Mandatory: □

Figure 25: Attribute Settings Tab

Each attribute block can be collapsed or expanded by clicking on its green header. The attribute type is automatically determined from the dataset encoding and is represented by an icon (e.g. character STATUS, number EASTING or date DATE).

The Attributes tab displays a list of attributes for the selected dataset. For each attribute it shows the following (see Figure 26):

- **Name**, **Type** (date, character, number, etc.) and **length** (for character and number attributes). It also displays the number of **decimal points** for a real number attribute type, but not for type float values.
- There is a "**Mandatory**" checkbox which specifies whether a field is mandatory. A mandatory field is checked and found erroneous if any of the attributes in that field are Null, blank, NaN or contain an empty string. On mining the data, if TopoCheck finds all attributes have an entry, it defaults the tick-box to on.
- There is a "**Unique**" checkbox. If checked, TopoCheck checks that values for that particular field are unique, and reports an error if they are not. On data mining, if TopoCheck finds that all attribute entries in this field are distinct, it defaults the "Unique" tick-box to on.
- There are two text input boxes. One allows the user to enter an alternative name (**Alias**), the other to enter an attribute description (**Description**). This is purely for the benefit of the user. All values entered (if saved) are retrieved each time the dataset is selected.
- For numeric and date fields, there are two text boxes to view/set minimum and maximum values for **Domain Range**. These values are automatically filled in for the selected dataset upon data mining. By changing the minimum and/or maximum values acceptable for a numeric field, TopoCheck tool then reports any records which fail this domain range check.
- A field's values can be constrained to a **Domain Code List**.



Area_SQM_S

Area_SQM_S: (Number)

Length: 33 Decimals: 16

Mandatory: ☑

Unique: ☑

Alias:

Description:

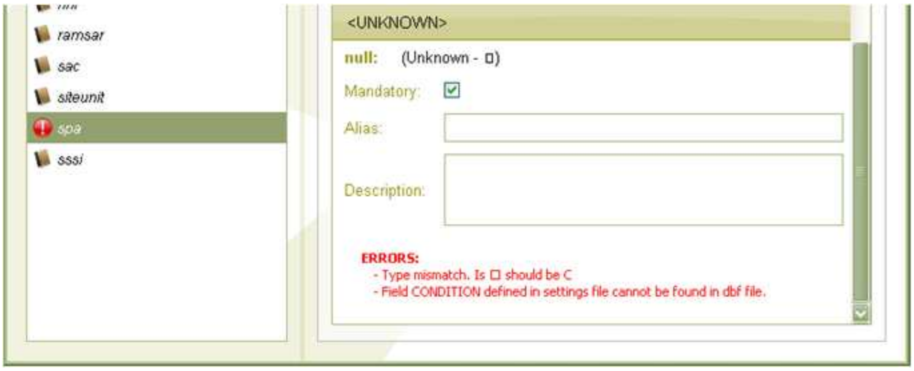
Domain Range: Min val: Max val:

Domain Codes List: ISO 3166-1

Figure 26: User interface for setting an individual attribute's parameters

After running the validation, the results of the attribute checks (listed in) are summarised on screen (if Verbose Mode is checked) or in more detail in a dbf file (`*_attribute_errors.dbf`).

If any of the dataset attribute fields contains an Oracle reserved word or if an incompatible attribution file was loaded for a dataset, then no further checks will be possible on that. Such a dataset is symbolised by a red warning icon in the Dataset sub-panel and the problematic attribute is shown with a red ERROR warning in the Attributes sub-panel (Figure 27).



null: (Unknown - □)

Mandatory: ☑

Alias:

Description:

ERRORS:
- Type mismatch. Is □ should be C
- Field CONDITION defined in settings file cannot be found in dbf file.

Figure 27: Attribute errors are displayed below the attribute's properties

Table 4: Attribute Checks

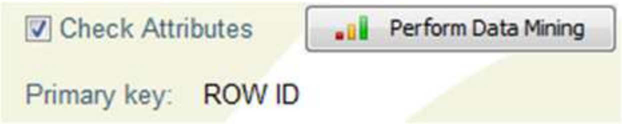
Error Code	Test Name	Description	Conformance
206	Null Value	Mandatory field not populated.	Fail
220	Domain	Field value is outside the domain range or code list.	Fail
221	Unique	Value for a unique field is duplicated.	Fail

8.1 Primary Key

Each dataset must have a primary key which the tool can use as the feature identifier. For shapefile datasets this is simply its ROW ID and cannot be changed by the user.

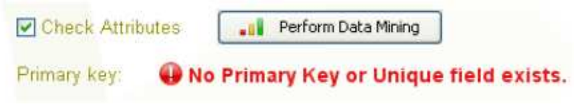
Oracle datasets are checked for primary key database constraints. If such a constraint exists, it is used as a primary key and cannot be changed. If there is no such constraint, user has to select one unique field to be used as a feature identifier, first unique field is selected by default.

If there is no unique field in a dataset, an error is displayed (Figure 29) and this dataset cannot be processed until a unique identifier is defined.



Primary key: ROW ID

Figure 28: Selecting unique field as identifier



Primary key: No Primary Key or Unique field exists.

Figure 29: Error when dataset has no feature identifier

Uniqueness of user defined feature identifiers will be checked before every validation of such dataset to ensure proper behavior of the tool. If a non-distinct value is found, an error will be displayed and the validation process will not start.

8.2 Domains

On data mining, an auto-generated domain code list is created for each field in the dataset and stored as an XML file in the same directory location as the dataset. However, if the number of unique values for a field's domain list exceeds 100 then this code list is not created.

The auto-generated code list is available by clicking the "edit" button next to the code list name (see Figure 30). The code list is displayed in a dialog, which allows editing of existing code values or adding new ones (Figure 31). Clicking on the green arrow button next to the Domain Codes List label, allows the user to select from the following list (see Figure 32):

- "Browse" from a pre-existing file using the normal operating system file selection tools (perhaps previously mined on a different dataset)
- "None" which resets that attribute to have no domain
- Auto-generated code list
- ISO3166-1 or ISO3166-2 for Country codes and Country subdivisions
- ISO639-2 for Language codes

The predefined (ISO) codes lists cannot be edited, but the auto-mined or user-specified ones can.

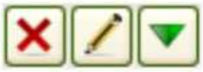
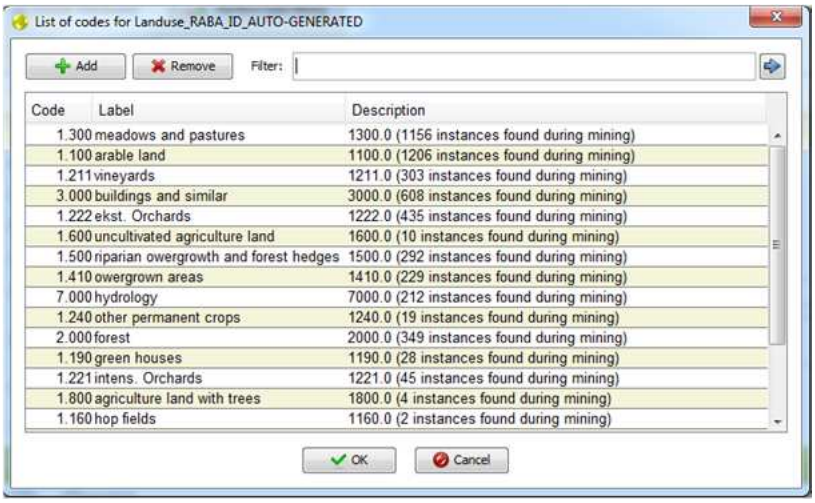


Figure 30: The code list can be disabled (by clicking the red X button), edited (the pencil button) or set to a predefined value (green arrow) using the buttons displayed to the right of the code list name.



List of codes for Landuse_RABA_ID_AUTO-GENERATED

Code	Label	Description
1.300	meadows and pastures	1300.0 (1156 instances found during mining)
1.100	arable land	1100.0 (1206 instances found during mining)
1.211	vineyards	1211.0 (303 instances found during mining)
3.000	buildings and similar	3000.0 (608 instances found during mining)
1.222	ekst. Orchards	1222.0 (435 instances found during mining)
1.600	uncultivated agriculture land	1600.0 (10 instances found during mining)
1.500	riparian owergrowth and forest hedges	1500.0 (292 instances found during mining)
1.410	owergrown areas	1410.0 (229 instances found during mining)
7.000	hydrology	7000.0 (212 instances found during mining)
1.240	other permanent crops	1240.0 (19 instances found during mining)
2.000	forest	2000.0 (349 instances found during mining)
1.190	green houses	1190.0 (28 instances found during mining)
1.221	intens. Orchards	1221.0 (45 instances found during mining)
1.800	agriculture land with trees	1800.0 (4 instances found during mining)
1.160	hop fields	1160.0 (2 instances found during mining)

Figure 31: Code list editing dialog

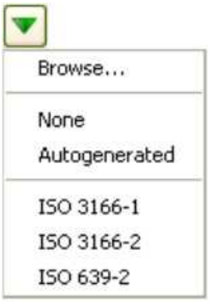


Figure 32: Code list selection menu

The auto-generated list is stored on the disk under a filename, whose structure is: `*_X_AUTO-GENERATED`*, where* `*` *is the dataset name, and* `X` *is the Attribute name. For example, all unique values for the* `Name` *attribute in the* SSSI *dataset are stored in* `sssi_NAME_AUTO-GENERATED-svt_codeslist.xml` *(an example of the XML file is displayed in Listing 1: An example XML file for a domain code list).*

Listing 1: An example XML file for a domain code list

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CodesList name="siteunit_CONDITION_AUTO-GENERATED" description="" url=""
           lastUpdatedOn="2008-07-20">
<codes>
  <item code="1" label="UNFAVOURABLE DECLINING"
        description="UNFAVOURABLE DECLINING (2256 instances found during mining)"/>

  <item code="2" label="UNFAVOURABLE RECOVERING"
        description="UNFAVOURABLE RECOVERING (5261 instances found during mining)"/>

  <item code="3" label="FAVOURABLE"
        description="FAVOURABLE (10326 instances found during mining)"/>

  <item code="4" label="UNFAVOURABLE NO CHANGE"
        description="UNFAVOURABLE NO CHANGE (3510 instances found during mining)"/>

  <item code="5" label="PART DESTROYED"
        description="PART DESTROYED (49 instances found during mining)"/>

  <item code="6" label="DESTROYED”
        description="DESTROYED (42 instances found during mining)"/>

  <item code="7" label="Not assessed"
        description="Not assessed (76 instances found during mining)"/>

</codes>
</CodesList>
```

9 SUMMARY (RESULTS) TAB

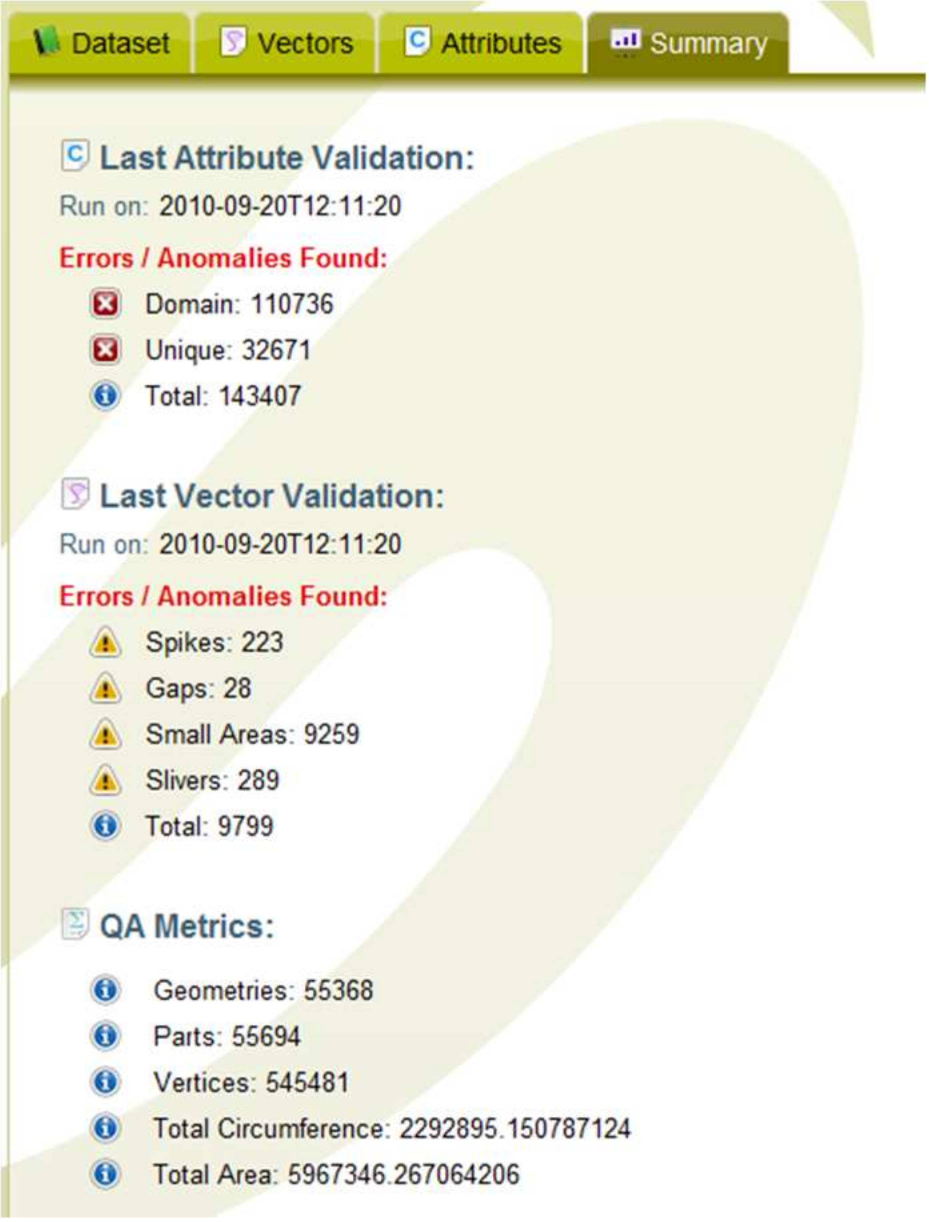
On running TopoCheck validation, summary results are output to the DOS window (if Verbose Mode checkbox is selected) and a new tab – “Summary” – is added alongside the Dataset, Vectors and Attributes tabs. The Summary consists of three main sections:

- Attribute validation (according to the attribute parameters specified),
- Geometry validation (according to the input geometry parameters), and
- Summary QA metrics, including the number of geometries, vertices, total circumference and area (if relevant – for point or line dataset this is omitted).

An example of the summary output is shown in Figure 33: Validation Summary for the SSSI dataset. It displays the timestamp of the validation, number of failures (red square with a cross) and warnings (yellow triangle with an exclamation mark) for individual check as well as the total count of anomalies. If there were no mandatory failures and a .zip package was prepared, this is also noted in the GUI.

The results are shown automatically after the program has been run and are stored as part of TopoCheck data to be reviewed later. If the dataset has changed since the last validation, a warning is displayed to notify the user that the summary might be outdated.

All results are also saved in one XML output file which is packaged in the dataset zip file for upload, and two HTML files reporting details of the validation (see Section 10 for details).



Last Attribute Validation:

Run on: 2010-09-20T12:11:20

Errors / Anomalies Found:

Domain: 110736

Unique: 32671

Total: 143407

Last Vector Validation:

Run on: 2010-09-20T12:11:20

Errors / Anomalies Found:

Spikes: 223

Gaps: 28

Small Areas: 9259

Slivers: 289

Total: 9799

QA Metrics:

Geometries: 55368

Parts: 55694

Vertices: 545481

Total Circumference: 2292895.150787124

Total Area: 5967346.267064206

Figure 33: Validation Summary

10 OUTPUT FILES

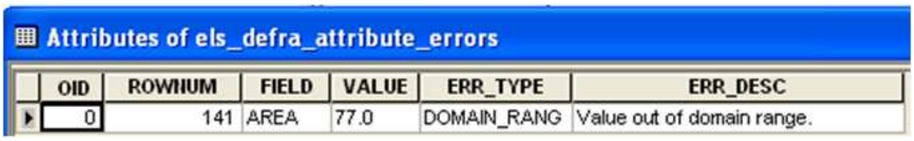
After running TopoCheck tool, a number of result files are created and saved in the same directory as the parent dataset. They fall into three categories: attributes, geometry, and settings.

10.1 ATTRIBUTES

`*_attribute_errors.dbf` - dbf file which lists any attribute errors found during the check. For each error the following are recorded: row number, field, value, error type, and error description.

Example 1: Domain Range Error

In the example in Figure 34 the minimum value for the Area field was set to 100m^2. After running TopoCheck tool on this dataset, only one record (no 141) was found to have an Area below this value. The actual Area value is given plus the error type (here Domain Range) and a short description of why an error has occurred.



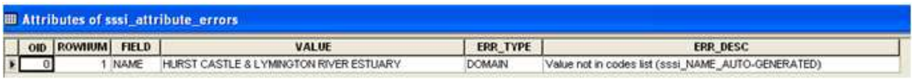
Attributes of els_defra_attribute_errors

OID	ROWNUM	FIELD	VALUE	ERR_TYPE	ERR_DESC
0	141	AREA	77.0	DOMAIN_RANG	Value out of domain range.

Figure 34: Domain Range Error

Example 2: Domain Value Error

In this example (Figure 35) one of the values of the Name field "Hurst Castle & Lymington River Estuary" does not appear in the Domain Code List.



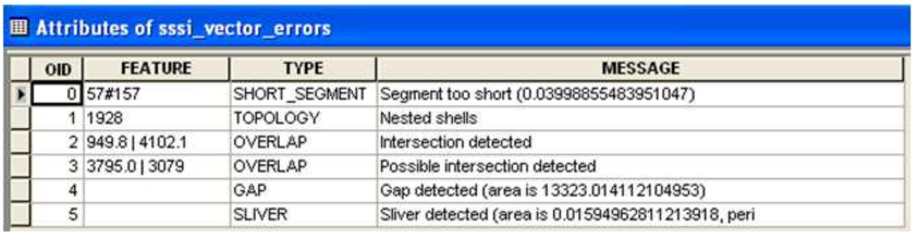
Attributes of sssi_attribute_errors

OID	ROWNUM	FIELD	VALUE	ERR_TYPE	ERR_DESC
0	1	NAME	HURST CASTLE & LYMINGTON RIVER ESTUARY	DOMAIN	Value not in codes list (sssi_NAME_AUTO-GENERATED)

Figure 35: Domain code list error

10.2 Geometry

`*_vector_errors.shp` - point shapefile showing the location of each geometric error in the dataset. More information on these errors is shown in the associated dbf file (e.g. `*vector_errors.dbf`). Each error is classified according to type and these include: orientation, gap, short segment, sliver and topology. In addition, some types of error (i.e. short segment, topology) also show the IDs of the features which are affected.



Attributes of sssi_vector_errors

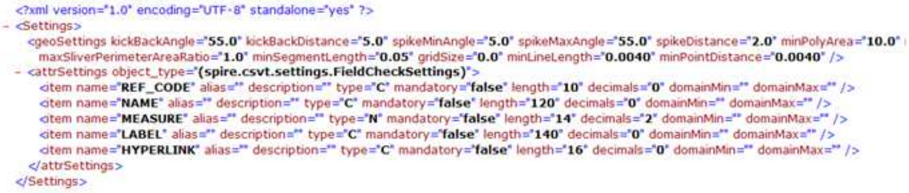
OID	FEATURE	TYPE	MESSAGE
0	57#157	SHORT_SEGMENT	Segment too short (0.03998855483951047)
1	1928	TOPOLOGY	Nested shells
2	949.8 | 4102.1	OVERLAP	Intersection detected
3	3795.0 | 3079	OVERLAP	Possible intersection detected
4		GAP	Gap detected (area is 13323.014112104953)
5		SLIVER	Sliver detected (area is 0.01594962811213918, peri

*Figure 36: Example *vector_errors.dbf file (associated with *vector_errors.shp file)*

The example in Figure 36 shows some typical geometry errors in a polygon shapefile. The geometric parameters (section 6) set by the user are used by TopoCheck tool to identify a range of geometry errors. Changing the parameter values obviously changes the number of identified geometry errors. For example, increasing the maximum sliver area in TopoCheck tool increases the number of slivers identified in the data.

10.3 Settings

`*_svt_settings.xml`- xml file created every time TopoCheck is run for a dataset. This file holds the vector and attribute parameters used to run the tool.



```
  <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <Settings>
    <geoSettings kickBackAngle="55.0" kickBackDistance="5.0" spikeMinAngle="5.0" spikeMaxAngle="55.0" spikeDistance="2.0" minPolyArea="10.0"
      maxSliverPerimeterAreaRatio="1.0" minSegmentLength="0.05" gridSize="0.0" minLineLength="0.0040" minPointDistance="0.0040" />
  - <attrSettings object_type="(spire.csvt.settings.FieldCheckSettings)">
      <item name="REF_CODE" alias="" description="" type="C" mandatory="false" length="10" decimals="0" domainMin="" domainMax="" />
      <item name="NAME" alias="" description="" type="C" mandatory="false" length="120" decimals="0" domainMin="" domainMax="" />
      <item name="MEASURE" alias="" description="" type="N" mandatory="false" length="14" decimals="2" domainMin="" domainMax="" />
      <item name="LABEL" alias="" description="" type="C" mandatory="false" length="140" decimals="0" domainMin="" domainMax="" />
      <item name="HYPERLINK" alias="" description="" type="C" mandatory="false" length="16" decimals="0" domainMin="" domainMax="" />
    </attrSettings>
  </Settings>
```

Figure 37: All the geometric and attribute parameters used by TopoCheck tool are saved in this file.

10.4 Results Summaries

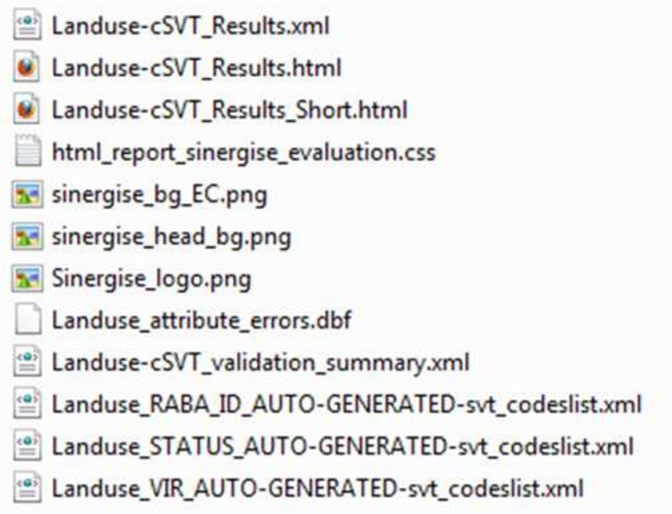
`*_SVT_Results.xml`,`*SVT_Results.html`,`*SVT_Results_Short.html` (Summary XML and HTML files) will be generated even if the testing fails and results are not packed to a zip file. The report will be generated after every validation run irrespective of pass or fail. The output file name will be prefixed with the dataset name. An example of the output HTML is show in Figure 40.

11 PACKAGING TO ZIP

Once TopoCheck tool has been run on a dataset and, as long as no mandatory fails have been identified, all relevant shapefiles, code lists, settings files and results are packaged into one compressed (`.zip`) file for that particular dataset.

The zip file is not generated if either the geometry or attribute validation checks are not carried out. In other words, both geometry and attribute validation need to be carried out with no mandatory fails for the zip file to be generated. Similarly, the zip file will only be created if there are no standard failures which have to be fixed according to the SPIRE data standard (e.g. no unclosed polygons, or no attributes exceeding set domains, etc).

As an example, TopoCheck tool was run on the "Land use" shapefile dataset (Landuse.shp) and no mandatory fails were reported for either the geometry or attribute checks. A zip file called Landuse-svt_results_2010-10-22.zip was then created (its contents are listed in Figure 38). Note that the current date is added to the end of the zip file name. As there were no geometry errors, no geometry error shapefile has been created or packaged.



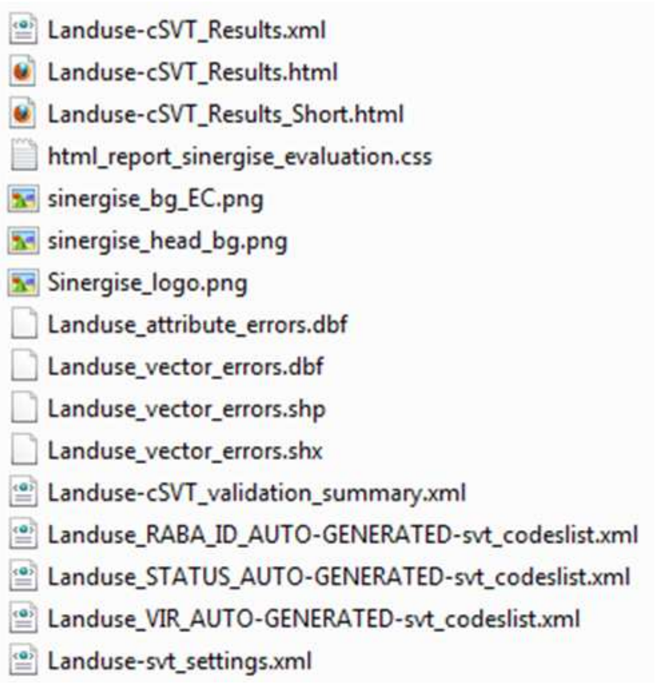
Landuse-cSVT_Results.xml
Landuse-cSVT_Results.html
Landuse-cSVT_Results_Short.html
html_report_sinergise_evaluation.css
sinergise_bg_EC.png
sinergise_head_bg.png
Sinergise_logo.png
Landuse_attribute_errors.dbf
Landuse-cSVT_validation_summary.xml
Landuse_RABA_ID_AUTO-GENERATED-svt_codeslist.xml
Landuse_STATUS_AUTO-GENERATED-svt_codeslist.xml
Landuse_VIR_AUTO-GENERATED-svt_codeslist.xml

Figure 38: Contents of TopoCheck output zip file (without vector errors)

Files packaged in the zip include:

- The original shapefile (not present when validating Oracle datasets);
- Any auto-generated domain code lists in xml format;
- Settings file (`*-svt_settings.xml`) which includes all dataset, geometry and attribute parameters used during validation;
- Validation reports in both xml and html format;
- Shortened results file in html format;
- Supporting CSS (cascading style sheet) files and images (Sinergise logos) for html documents.

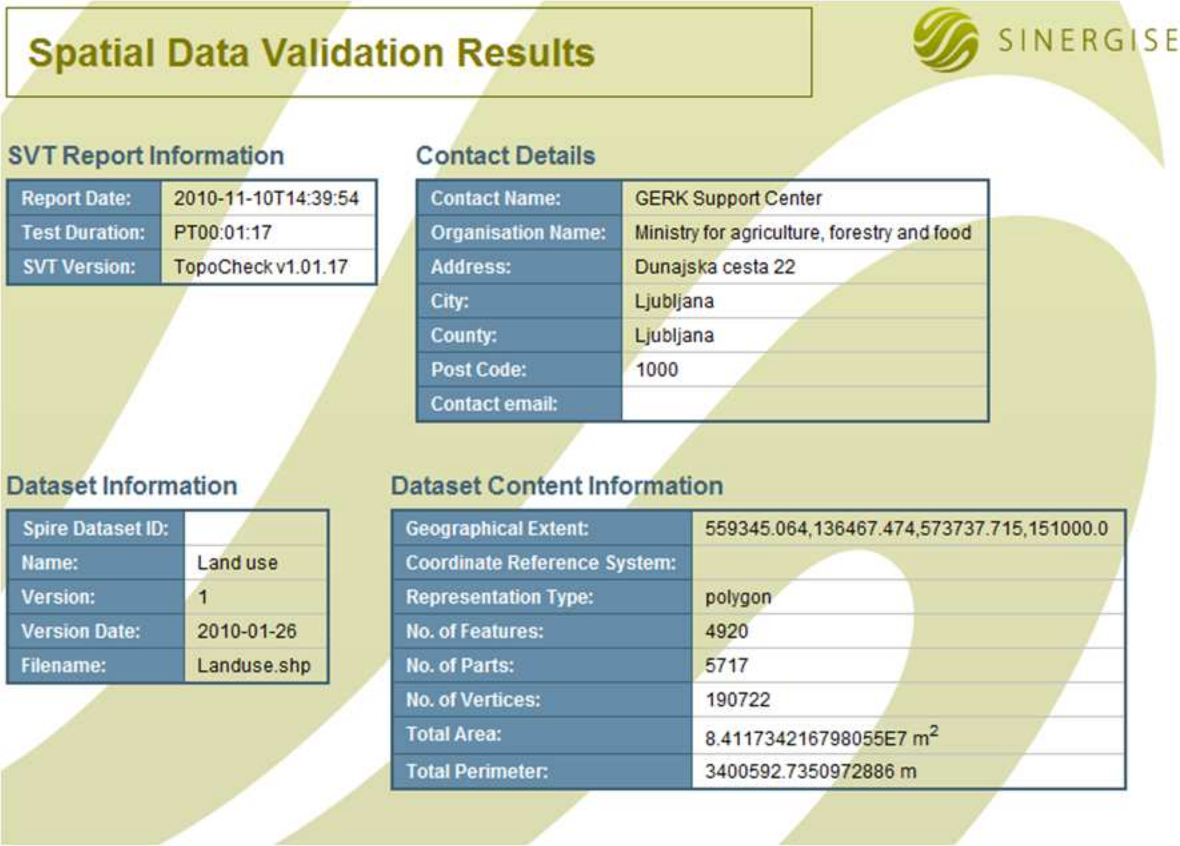
Because there were no geometry errors according to the parameters set in this case (either fatal or not), no error shapefiles (or associated dbf files) are included in the zip package. Normally they would be, as the example for the "3mile nautical limit" dataset shows in Figure 39.



Landuse-cSVT_Results.xml
Landuse-cSVT_Results.html
Landuse-cSVT_Results_Short.html
html_report_sinergise_evaluation.css
sinergise_bg_EC.png
sinergise_head_bg.png
Sinergise_logo.png
Landuse_attribute_errors.dbf
Landuse_vector_errors.dbf
Landuse_vector_errors.shp
Landuse_vector_errors.shx
Landuse-cSVT_validation_summary.xml
Landuse_RABA_ID_AUTO-GENERATED-svt_codeslist.xml
Landuse_STATUS_AUTO-GENERATED-svt_codeslist.xml
Landuse_VIR_AUTO-GENERATED-svt_codeslist.xml
Landuse-svt_settings.xml

Figure 39: Contents of TopoCheck output file with errors in geometry.

Part of the shortened results file for a dataset is shown in Figure 40. All geometric and attribute errors (non-fatal ones), as well as all validation parameters are listed in the output HTML document.



Spatial Data Validation Results

SVT Report Information

Report Date:	2010-11-10T14:39:54
Test Duration:	PT00:01:17
SVT Version:	TopoCheck v1.01.17

Contact Details

Contact Name:	GERK Support Center
Organisation Name:	Ministry for agriculture, forestry and food
Address:	Dunajska cesta 22
City:	Ljubljana
County:	Ljubljana
Post Code:	1000
Contact email:	

Dataset Information

Spire Dataset ID:	
Name:	Land use
Version:	1
Version Date:	2010-01-26
Filename:	Landuse.shp

Dataset Content Information

Geographical Extent:	559345.064,136467.474,573737.715,151000.0
Coordinate Reference System:	
Representation Type:	polygon
No. of Features:	4920
No. of Parts:	5717
No. of Vertices:	190722
Total Area:	8.411734216798055E7 m^2
Total Perimeter:	3400592.7350972886 m

Figure 40: The generated HTML report