

CISutil

version 5.0.0 or higher

Short Section Documentation

Matthew Nicoll, Cypher Consulting
2021-08-16 updated 2023-03-22

Contents

1	Primary Sections.....	1
2	Short Sections Parameters.....	1
3	Output Files	3
4	Output CSV File Fields	4
5	Algorithms.....	7

1 Primary Sections

Primary Sections are the pre-defined LKI sections which are to be broken down into short sections. No generated Short Section may span the end of a Primary Section.

Examples of Primary Sections:

- LKI Segments
- Uniform Traffic Volume Sections (UTVS)
- Sub-LKI-Segments defined in an SDF

Currently, Primary Sections may not span LKI Segment nodes.

Allowing Primary Sections which span LKI nodes is a possibility, but would require more programming work in CISutil.

An LKI Node will be included in no more than one output short section. I.e if two Primary Sections share a Node, that Node will be included in just one output short section.

2 Short Sections Parameters

Except for the LKI version, these are specified on the Short Sections form, which is opened using the "Short Sections..." button on the CISutil Main Form SDF tab.

LKI Version **on the CISutil Main Form, SDF tab.**

- this specifies the LKI version used for all the Short Section analysis
- Primary Section LKI segment numbers specified must exist in this LKI version.

Run ID

- the string entered in this field becomes part of the output file names
- if you do not change this between runs, **the previous output files will be over-written.**

Primary Sections

- choose between LKI Segments, UTV Sections, or sub-LKI-Segments defined in an SDF.
- for LKI Segments and UTV Sections, you can:
 - select any combination of Numbered, Lettered, and UnNumbered highways
 - or limit to a comma separated list of LKI segments and/or Highways.
- E.g. if you select UTV Sections, check the Segments checkbox, and put
0304,0307,H4,0420
in the Segments field, you will get all UTV sections on LKI Segments 0304, 0307, 0420
plus all the LKI segments of Highway 4.

Create Intersection Sections

- If this box is checked, each primary section will be scanned for intersections of the specified landmark types
- output intersection section records will be generated for each such intersection.
- Radius and Min inter-intersection section lengths:
 - see the Algorithms section for a description of how these values are used
 - these are expressed hectometres (1 hectometre = 100 metres)
- Numerical ID Offset:
 - Intersection and non-intersections are numbered sequentially along the highway.
 - The Numerical ID Offset is added to that sequence number to obtain the ID in the output csv file.
- SID Prefix
 - optional
 - each intersection section is given a String ID (SID) constructed as
<SID_prefix><primary_section_ID>_<nn>
where nn is the sequential section number within the primary section

Create Sections

- if this box is checked, Primary Sections are broken down into short sections, as described in the Algorithm section below.
- if the **Create Intersection Sections** box is also checked, the Intersection sections are identified first, and the non-Intersection sections are identified between the Intersection sections.
- If the **Create Intersection Sections** box is NOT checked, intersections are not given special treatment, unless intersection landmarks are specified in the "Break ... at Landmarks of Type" list, in which case they are treated the same as any other landmark types in that list.
- If the **Break sections where the Highway Classification changes...** box is checked, the following field must contain a comma-separated list of Hwy Class Category numbers, eg: 2,3,4,5. (See the LKI Class_Category table for details.)
- If the **Break sections at boundaries of...** box is checked, select **Area**, **District** or **Region** to indicate the type of boundary at which sections are to be broken.
- If the **Break long sections at Landmarks of Type** box is checked, the following field must contain a comma-separated list of LKI landmark types, eg: A1,A2,A3,A5,A8
- the **Numerical ID Offset** and **SID Prefix** work the same as described above for the Intersection sections.

Include AADTs...

- if this box is checked, AADTs are looked up for each year in the specified year range, and inserted into the output record for each short section.
- the average over all the years is also calculated and inserted.

- see the notes under AADT_YYYY and AvgAADT in the **Output CSV File Fields** section below for notes on the LKI location used to look up the volumes.
- the volumes are looked up in the Volume_for_CISutil.mdb MS-Access database file which is distributed with the CISutil executable.
- the Volume_for_CISutil.mdb database is exported from the master utvs_volume.accdb database, which is updated at Cypher Consulting, and also stored in folder "P:\HQ\ENG\Safety - Highway Engineering\CIS\Volume"

Include Latitude and Longitude of section endpoints

- this option causes 6 additional fields to be produced for each short section, as described under Latitude1 (etc.) in the **Output CSV File Fields** section below.

Extend endpoints 50m to close 100m gaps...

Short sections are normally defined with start and end kms to one decimal point, and with 0.1 km separation, so as to create an SDF without overlaps suitable for input to CIS.

The "Extend..." option causes:

- output fields Km1, Km2 and Length to be written with 2 decimal places,
- midsegment endpoints are extended out 50 m so that for adjacent sections, a section starts exactly where the previous section ends,
- at the start of an LKI segment, the first short section start point is moved back to km 0 if necessary (whether or not that node has been included in previous sections),
- at the end of an LKI segment, the last short section is extended to the LKI segment length (whether or not that node has been included in previous sections),
- note that for the short sections at the beginning and end of an LKI segment, the extension may be larger than 50 m.

Write copy of output file transformed to LKI version

- checking this option will cause a separate copy of the output csv file to be created, with all LKI segment numbers and Kms transformed to the specified target LKI version.
- e.g. if LKI version B has been published, but CIS is still working with LKI version A, you want to do the Short Section analysis with version B (using the latest landmark type and highway classification data), then transform the output to version A for use as an SDF with CIS.

3 Output Files

The output file names are displayed at the bottom of the Short Sections form when the Run is finished. They can be opened for viewing using their adjacent Open buttons.

Log File

- written to your Windows temporary file folder (named in the %temp% environment variable)
- named: short_sections_log_<Run_ID>.txt

(Main) Output File

- CSV file containing a headings record, and one record for each Short Section generated, (including Intersection sections).
- all LKI information is in accordance to the LKI version specified on the CISutil Main Form SDF tab at the time the file was generated.
- conforms to the specifications of a CIS Section Definition File (SDF)
- named: short_<section_type>_<Run_ID>.csv
where <section_type> is one of:
Intersections

Sections
MixedSections (both Intersections and Sections)

Output File 2 (optional)

- a copy of the main Output File, with the LKI Segment and Km location transformed to the specified target LKI version.
- all other information (landmark types, highway class etc.) will still be as in the original LKI version used for the analysis.
- named as the main output file, but with "_LKIyyyymm" appended before the ".csv" extension.
- *Warning: the transformation can result in sections which overlap at a single Km. Run the overlap check, and do a manual correction if necessary.*

4 Output CSV File Fields

ID

- an integer ID assigned by the Short Sections program.
- the sequential generated section number, plus an optional user- specified offset

SID

- a String short section ID (SID) constructed as <SID_prefix><primary_section_ID>_<nn> where:
 - <SID_prefix> is specified separately for intersection and non-intersection sections on the form, and may be a null string
 - nn is the sequential section number within the primary section

Primary Section ID

- the field heading will be **LKI_Segment**, **UTVS** or **SDF_Section** depending upon the type of Primary Section specified.
- the ID will be an LKI Segment number, a UTVS ID, or the SDF section ID

Highway

- the Highway_Number (and Highway_Letter) of the LKI Segment, as defined in the LKI Segment table.

Segment

- the LKI Segment number of the Short Section
- (will duplicate the LKI_Segment field for "LKI Segment" Primary Sections.

Km1, Km2

- start and end of the short section, (1 decimal place, or 2 if the Extend option is selected)

KmInt (for Intersection sections only)

- the Km (rounded to 1 decimal place) of the intersection landmark
- may be outside of the Km1 - Km2 range (see IntPos below.)

Intersection_ID (for Intersection sections only)

- positive numbers point to the intersection in the (new in 2018) Intersection LKI table
- a negative number indicates the intersection is not currently in the LKI Intersection table, which implies that the intersection:
 - is not signalized (unless the LKI is out-of-date!),
 - is not at an LKI Node,

- has just one record in the LKI Landmark table (whose ID is made negative to create the Intersection_ID)
- is on only one LKI Segment.
- can be used to group sections to combine data where there are multiple Intersection Sections for the same intersection, e.g. when:
 - the intersection is located on segments on both sides of a divided highway
 - the intersection is at a primary section break, resulting in an Intersection section on each side of the break.
- will be the consistent over all short section runs (e.g. Beacon Ave will always have Intersection_ID = 25)

Legs *(for Intersection sections only)*

- the number of roads entering the intersection at KmInt.
- for intersections on the LKI Intersection table (those with positive Intersection_ID), this information comes from the Intersection table.
- for other intersections, which should all be mid-segment intersections, Legs is determined by summing the Sides of all Landmark records at the 1-decimal place location, and adding 2.
- in cases of multiple intersections whose multiple kms round to the same Hm, Legs may be > 4.

IntPos *(for Intersection sections only)*

- 0 - KmInt is in the range Km1 to Km2
- 1 - KmInt is prior to Km1 (but within the specified intersection radius)
- 1 - KmInt is after Km2 (but within the radius)

NintLoc1dp *(for Intersection sections only)*

- number of 1-decimal-place intersection-type LKI Landmark locations in the range Km1 to Km2
- (Multiple landmarks at the same location are counted as one *location*.)
- e.g. if Km1 = 1.0, Km2 = 1.4 and the following LKI Landmarks are in that range:

1.16	A1	Smith St
1.24	A1	Jones Ave

 NintLoc1dp = 1, because 1.16 and 1.24 both round to 1.2.

NintLoc2dp *(for Intersection sections only)*

- number of 2-decimal-place intersection-type LKI Landmark locations whose 1-decimal-place locations are in the range Km1 to Km2
- (Multiple landmarks at the same location are counted as one *location*.)
- e.g. if Km1 = 1.0, Km2 = 1.4 and the following LKI Landmarks are in that range:

1.16	A1	Smith St
1.24	A1	Jones Ave
1.24	A1	Adam St

 NintLoc2dp = 2, because 1.16 and 1.24 are separate locations, unrounded, and the two landmarks at 1.24 are counted as 1 location.

Length

- = Km2 - Km1 + 0.1 (Km2 - Km1 if the Extend option is selected)

- the extra 0.1 is because theoretically collisions at a particular 1 dp Km 'x' occur somewhere between $x - 0.5$ and $x + 0.5$ so we get an extra 0.5 on each end of the section.

Lmtype1

- the landmark type (from the LKI Landmark table) at Km1
- for non-intersection short sections, a prefix of "<-" indicates that this short section is immediately preceded by an Intersection Section of an intersection of the indicated type. "->" indicates that this short section is immediately followed by an Intersection Section of an intersection of the indicated type.

Lmtype2

- the landmark type (from the LKI Landmark table) at Km2
- for non-intersection short sections, a postfix of "->" indicates that this short section is immediately followed by an Intersection Section of an intersection of the indicated type.

Desc1, Desc2

- descriptions (from the LKI Landmark table) of the landmarks at Km1 and Km2
- "<-" and "->" pre- and post- fixes are applied as described above for Lmtype1 and Lmtype2.

EndBreakType

- Describes how the end of the section was determined:

Short Sections (non-Intersection)

- 0: at an intersection section,
- 1: highway classification change
- 2: landmark of one of the specified types,
- 3: maximum section length,
- 4: midpoint of last 2 short sections in a Primary Section
- 5: end of the Primary Section
- 6: MoTI Area, District, or Region change

Intersections

- 1:normal (intersection km + radius)
- 2:shortened to midpoint between close intersections
- 3:lengthened to midpoint between close intersections
- 4 end of Primary Section
- 5 lengthened to end of Primary Section
(to prevent a section shorter than min section length)

Latitude1, Longitude1 *(optional)*

- geographic location of the start of the section
- accuracy is variable, as indicated by LLsrc1

LLsrc1 *(optional)*

- indicates the source of the Latitude1, Longitude1 data:
 - >0 = the LatLon_Source_ID of the landmark at Km1
 - see the LKI LatLon_Source table (e.g. 4 means Google Earth)
 - < 0 - Km1 is not at a Landmark, so the Lat & Lon are calculated using segment geometry from:
 - 1 - CHRIS
 - 2 - GeoBC

Latitude2, Longitude2, LLsrc2 *(optional)*

- as for ~1, but for Km2.

Region, District, Area

- numbers looked up in the LKI using the intersection or mid-section LKI location, (as described for AADT_yyyy below).

PriSecMaj,UrbRur,ArtExpFre,DivUnd,LaneClass,Terrain

- highway classification categories 1 to 6, as defined in the LKI Class_Category and Class_Characteristic tables.
- looked up in the LKI using the intersection or mid-section LKI location, (as described for AADT_yyyy below).

HwyClass2345

- Highway classification categories 2,3,4 & 5 combined into a single string
- (i.e. fields UrbRur,ArtExpFre,DivUnd,LaneClass combined)

AADT_yyyy (optional)

- the Annual Average Daily Traffic for year yyyy, where yyyy is in the ADT year range specified on the Short Sections form.
- for Intersection sections, this will be the traffic volume at the location of the intersection, even if the intersection location is adjacent to rather than inside the section.
- if the intersection is at an LKI node, the volume will be that of the node, as defined in the CIS Help topic "Traffic Volume - At Nodes".
- for non-intersection sections, this will be the traffic volume at the midpoint of the section. *This means that for sections which span UTVS boundaries, the UTVS source for the volume is arbitrary - no distance-weighted averaging is done.* (Enhancing the volume calculation in such cases is a possibility.)

avg_ADDT (optional)

- average of the AADT_yyyy values.

5 Algorithms

Note:

- All analysis is done using LKI locations to one decimal place. E.g. all LKI landmark kms are rounded to one decimal place on input (and stored in integer hectometres).
- the word "location" is used to mean an LKI segment and km rounded to 1 decimal place.

Intersections

For each Primary Section:

1. Find all the LKI landmarks with types listed in the **Create Intersection Sections / At Landmark Types** field, in the Primary Section or within the Intersection Radius of the Primary Section. (The latter so that an intersection section can be defined at a primary section end which is a portion of an intersection diameter the centre of which is in an adjacent primary section.)
2. For each thus identified location, define an intersection section, storing the location, location plus and minus the Radius, and the landmark type. (If there are multiple selected landmarks at the same location, the one used is arbitrary.)
3. Look up the Intersection_ID on the Landmark-Intersection table. If not found set the Intersection_ID to the negative of the Landmark ID.
4. Scan the identified intersections shrinking and expanding the sections as required to eliminate overlaps, eliminate inter-intersection gaps smaller than the specified "Min inter-intersection section length", and ensuring adjacent intersection section endpoints are spaced 0.1 km apart.

5. Remove intersection sections at each end of the primary section which, as a result of the previous steps, are now completely outside the primary section.
6. Adjust the start and end kms of the first and last intersection in the primary section so that they do not extend outside the primary section, and so that no section shorter than the "Min inter-intersection section length" is left at either end.
7. Compare each intersection landmark location with its intersection section endpoints, and set the IntPos flag accordingly. (As defined in the Output Fields section.)

Sections (non-intersection)

If Intersection sections have been generated, the following algorithm is applied to each inter-intersection section (i.e. to the gaps between the intersection sections).

If Intersection sections have NOT been generated, the following algorithm is applied to the entire Primary Section.

The section the algorithm is being applied to will be referred to as the *supplied section*.

1. If the "Break sections where the Highway Classification changes..." option is selected:
 - a. Using the current date, and the *supplied section* boundaries, search the LKI SegClass table for highway classification changes in the specified Hwy Class Categories.
 - b. Breaks which follow a preceding break by only 0.1 km are ignored.
 - c. If N class breaks are found, the number of *class sections* Nclsec will be N + 1
 - d. If N = 0, or this option was not selected, then Nclsec will be 1.
2. If the "Break sections at boundarys of Area / District / Region..." option is selected:
 - a. Using the LKI segment of the *supplied section*, search the LKI SegArea table joined to the Area and District tables to get all the Area breaks in the segment, along with Area, District and Region (ADR) numbers.
 - b. Scan for Area, District or Region number (depending upon which admin level was selected) for changes inside the *supplied section*.
 - c. Copy the class section (clsec) array to the ADR section array (adrsec), with an additional section added for each break at the requested level.
 - d. If N ADR breaks are found, the number of *ADR sections* Nadrsec will be Nclsec + N
 - e. If N = 0, or this option was not selected, then Nadrsec will = Nclsec.
 - f. Copy array adrsec into array clsec. (Subsequent logic will refer to them as *class sections*, although clsec may or may not contain class or ADR breaks.)
3. For each *class section* longer than the specified maximum section length (MaxLen), if the "Break .. at Landmark of Type" option is selected:
 - a. the section is searched from MaxLen to MinLen for each of the Landmark Types, in the listed order.
 - b. If a landmark of the type is found, a section break is created at the landmark.
 - c. (Thus if there are multiple landmarks of the type, the break is created at the one closest to MaxLen.)
4. If the *class section* is still longer than MaxLen (because no Landmark types were looked for or, or none were found):
 - a. if the *class section* ends is the last in the *supplied section* then break the class section at its midpoint,
 - b. otherwise break the *class section* at MaxLen