



SEPM Products

Release 2023-01

New Features

Document Information	Description
Abstract	This document describes new features in the SEPM product release 2023-01
Version	2023-01
Disclaimers	All logos and trademarks in this document are property of their respective owners.



Contents

1	Overview	5
1.1	Changes Overview	5
1.2	Installation/Upgrade	5
1.2.1	Upgrade to 2023-01	5
1.2.2	AutoCAD DXF/DWG Format	5
1.2.3	Smallworld-Versions	5
1.2.4	SEPM NeplanAcp	5
1.2.5	French Messages	5
2	SEPM X-Translator	6
2.1	Configurations	6
2.1.1	Configuration 'skipped_physical_field_names'	6
2.2	Functions	7
2.2.1	Set the world of a geometry via attribute	7
2.2.2	Delete objects by 'Delete Mapping'	7
2.2.3	Set operation via attribute	8
2.3	Smallworld Source Format	10
2.3.1	Export Attribute Values Using ACE-Unit	10
2.3.2	Order of Exported Geometries in Drawing Methods	11
2.4	Smallworld Target Format	12
2.4.1	Import attribute values in the ACE unit	12
2.4.2	Grouping of Options	12
2.4.3	New Option 'Delete all objects in the target tables before import'	12
2.4.4	New Option 'Deactivate Validators'	13
2.4.5	New Option 'Deactivate Transactions'	13
2.4.6	Override Manifold Tolerances	14
2.4.7	Support for 0:0 Relations	15
2.4.8	Convert attribute values to _unset	16
2.4.9	Catalogue Tables with _unset Values	16
2.4.10	Dynamic !current_world!	17
2.5	AutoCAD DWG/DXF Source Format	18
2.5.1	New Option 'Explode blocks'	18

2.6	AutoCAD DWG/DXF Target Format	19
2.6.1	Export of Arcs and Circles	19
2.6.2	Selection of a Standard Hatch Patter in the Model Properties	19
2.6.3	Mapping of Hatch Styles	20
2.6.4	DwgAcp Coordinate Precision	20
2.7	Shape Target Format	21
2.7.1	Option "Write Shape Null geometries"	21
3	SEPM X-Raster	22
3.1	SEPM X-Raster Export	22
3.1.1	Order of BPLAN Worlds	22
4	NEPLAN	23
4.1	SEPM NeplanAcp	23
4.1.1	Job Server Integration	23

1 Overview

1.1 Changes Overview

This release **2023-01** covers the following improvements:

- **SEPM X-Translator** : New features especially for Smallworld and AutoCAD DWG/DXF formats:
 - Import/Export of attribute values in the units set in the ACE
 - Determination of the world of a geometry via an attribute
 - Determination of the operation (insert, change, delete) via an attribute
 - Various adjustments and extensions based on specific customer requirements
 - Various improvements in the RealDWG-based AutoCAD DWG/DXF format: support of circular arcs on export, hatching improvements
- Various bugfixes for **SEPM X-Raster**, for the **SEPM NEPLAN interface** and the **SEPM ISYBAU interface** are also included in this release

1.2 Installation/Upgrade

1.2.1 Upgrade to 2023-01

The upgrade to the present version 2023-01 is achieved through exchange of the layered products supplied by SEPM and taking over of the existing licenses, as described in the Admin Manual.

1.2.2 AutoCAD DXF/DWG Format

This version comprises a new version of the helper program **DwgAcp.exe**. If you use the "AutoCAD DWG/DXF" format you'll need to uninstall the existing **DwgAcpSetup.msi** and install the new setup. The installation directory in this release is:

C:\Program Files\SEPM\DwgAcp110

1.2.3 Smallworld-Versions

This release supports all Smallworld version from Smallworld 4.0 up to Smallworld 5.3.

1.2.4 SEPM NeplanAcp

The SEPM NeplanAcp was compiled with the latest NEPLAN libraries. Since the NEPLAN licenses depend on the NEPLAN version, the NEPLAN licenses must be updated during the upgrade.

An integration into the job server was implemented so that NeplanAcp calculations can be started as a job.

1.2.5 French Messages

Customized messages from SIGENEDIS_Customization\x-translator have been included into the product.

2 SEPM X-Translator

2.1 Configurations

2.1.1 Configuration 'skipped_physical_field_names'

The following change was not included in the Release Notes 2022-01: The setting 'skipped_physical_field_names' was split into a version for the Smallworld source and target format. Customized settings should be adjusted accordingly:

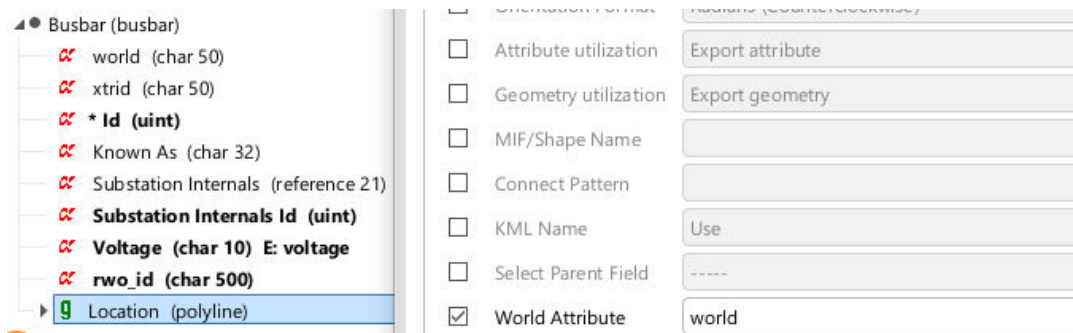
```
#2021-01 separate settings for source and target
#_pragma(classify_level=advanced,topic={x_translator},usage=redefinable)
#x_translator_settings.define_shared_constant(
#    ##
#    ## Skipped physical fields on export
#    ##
#    :skipped_physical_fields_names,
#    equality_set.new_with(
#        :ds!version
#        #:rwo_id
#    ),
#    :public)
#$
_pragma(classify_level=advanced,topic={x_translator},usage=redefinable)
x_translator_settings.define_shared_constant(
    ##
    ## Skipped physical fields on export
    ##
    :skipped_physical_fields_names_source,
    equality_set.new_with(
        :ds!version
        #:rwo_id
    ),
    :public)
$
_pragma(classify_level=advanced,topic={x_translator},usage=redefinable)
x_translator_settings.define_shared_constant(
    ##
    ## Skipped physical fields on import
    ##
    :skipped_physical_fields_names_target,
    equality_set.new_with(
        :ds!version,
        :rwo_id
    ),
    :public)
$
```

2.2 Functions

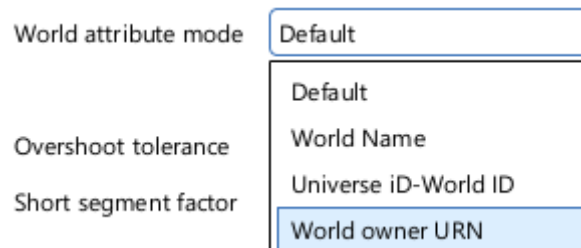
2.2.1 Set the world of a geometry via attribute

It is now possible to set the world of a geometry via an attribute value. This is a two steps process:

1. The geometry must be assigned a string attribute with the information about the world via the model properties.
2. The new option 'World attribute mode' selects how the information in the world attribute is to be interpreted. The following options are available in this release:
 - a. World name: The name of the world (Attribute int!world.name)
 - b. Universe ID-World ID: For example "3-2" means world 2 in universe 3
 - c. URN World owner: The URN of a database object that manages a world



1. Assignment of the world attribute

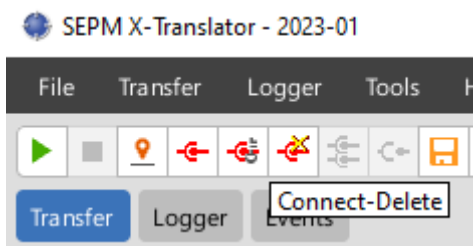


2. Option "World attribute mode"

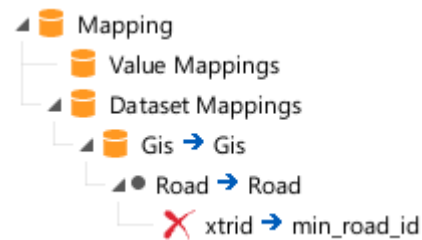
With these configurations, the world is set according to the geometry to be inserted.

2.2.2 Delete objects by 'Delete Mapping'

Up to now you could use the SEPM X-Translator 'Connect' and 'Connect-ID' commands to create or update objects in Smallworld GIS. You can now also delete objects in the GIS via a 'Delete Mapping'. All you need is a list of attribute values to uniquely identify the objects to be deleted in the GIS:



Create a Delete Mapping



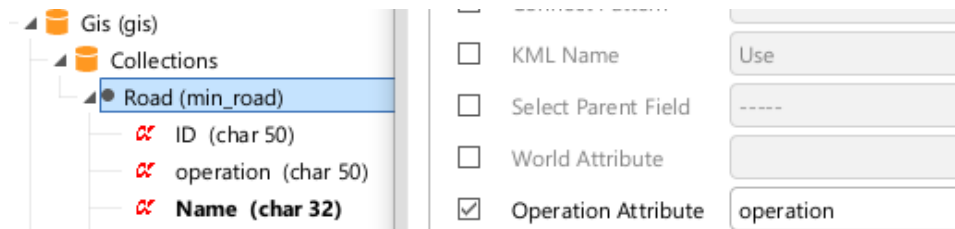
With 'Transfer' the identified objects are deleted.

2.2.3 Set operation via attribute

It is now possible to set an object's operation (insert, update, delete) via an attribute value. This is done by assigning an attribute to the object class that contains the operation to be performed.

Valid values for the operation are:

- ❖ "insert": Create a new database object
- ❖ "update": Update an existing database object
- ❖ "delete": Delete an existing database object

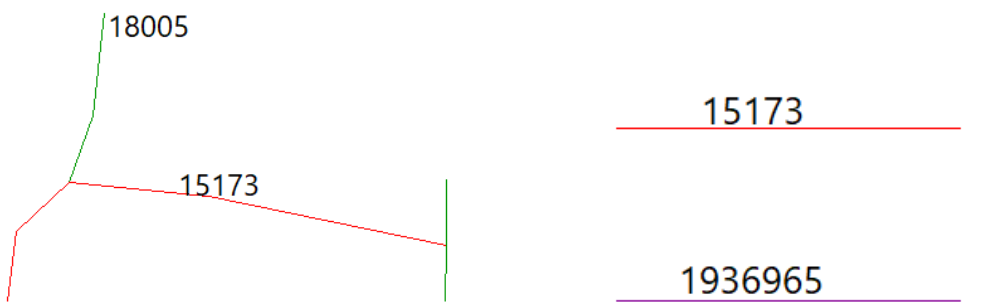


Assignment of an attribute for the operation to be performed

The following data set illustrates this new function:

```
<?xml version="1.0" encoding="UTF-8"?>
<translator version="2023-01" xmlns="http://www.sepm.ch/xtr_2009_03">
  <data>
    <dataset name="gis">
      <collection name="min_road">
        <feature operation="insert" xtrid="1">
          <attribute name="operation" value="insert" />
          <attribute name="name" value="insert-demo" />
          <geometry name="centre_line" type="polyline" style_name="3">
            <polyline_part world="0-0">
              <sector>
                <c x="544800.000" y="254900.000" />
                <c x="545000.000" y="254900.000" />
              </sector>
            </polyline_part>
          </geometry>
        </feature>
        <feature operation="insert" xtrid="2">
          <attribute name="id" value="15173" />
          <attribute name="operation" value="update" />
          <attribute name="name" value="update-demo" />
          <geometry name="centre_line" type="polyline" style_name="3">
            <polyline_part world="0-0">
              <sector>
                <c x="544800.000" y="255000.000" />
                <c x="545000.000" y="255000.000" />
              </sector>
            </polyline_part>
          </geometry>
        </feature>
        <feature operation="insert" xtrid="3">
          <attribute name="id" value="18005" />
          <attribute name="operation" value="delete" />
        </feature>
      </collection>
    </dataset>
  </data>
</translator>
```





Data before the transfer

Data after the transfer

1. Object 1936965 created
2. Object 15173: Geometry and attributes updated
3. Object 18005 deleted

With this new feature, in one single transfer you can:

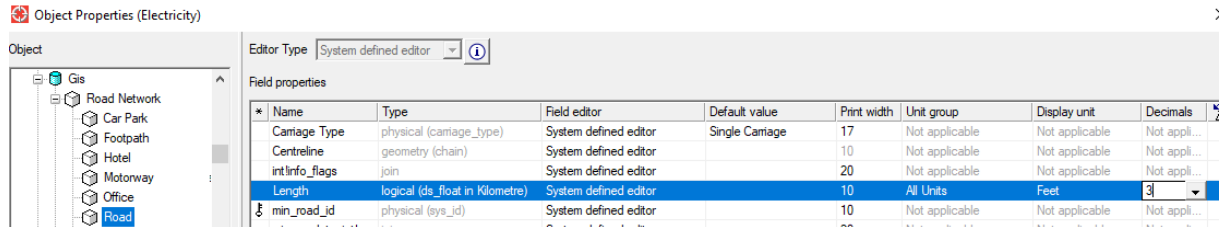
1. Create new objects
2. Update existing objects
3. Delete existing objects

2.3 Smallworld Source Format

2.3.1 Export Attribute Values Using ACE-Unit

It is possible to configure the Display Unit in the ACE.

Option	Description
Export attribute values in ACE-unit	Export attribute values using the configured Display Unit in the ACE. The ACE is selected by using the "From Trail" action in the Smallworld source dialog.



Display Unit for the Road Length is set to Feet.

Feature exported with default settings, the stored unit (kilometer in this example) is used for the exported length:




```
<feature operation="insert" xtrid="17718">
  <attribute name="xtrid" value="17718" />
  <attribute name="length" value="1.417652" />
  <geometry name="centre_line" type="polyline" style_name="2">
    <polyline_part world="0-0">
      <sector>
        <c x="548627.919468" y="260436.800000" />
        <c x="548612.137000" y="260395.450000" />
        <c x="548560.433000" y="260234.817000" />
      </sector>
    </polyline_part>
  </geometry>
</feature>
```

Feature exported with option selected and the Display Unit set to Feet:

```
<feature operation="insert" xtrid="17718">
  <attribute name="xtrid" value="17718" />
  <attribute name="length" value="4651.089185" />
  <geometry name="centre_line" type="polyline" style_name="2">
    <polyline_part world="0-0">
      <sector>
        <c x="548627.919468" y="260436.800000" />
        <c x="548612.137000" y="260395.450000" />
        <c x="548560.433000" y="260234.817000" />
      </sector>
    </polyline_part>
  </geometry>
</feature>
```

2.3.2 Order of Exported Geometries in Drawing Methods

When a drawing method returns FALSE ("Drawing is done"), the geometry is rendered on top of the geometries rendered through the drawing method. With the X-Translator export, the geometry was previously exported before the shapes of the drawing method. The order is now correct (Ticket#344).

 <p><i>Stourbridge Common</i></p> <p><i>Rendering in Smallworld GIS</i></p>	<pre> _method park.sepm_draw_park_annotation(...) ## Parameters : ## Returns : ## Function : [Draw a red circle around the text insertion coordinate] p_window.fill_line_transform(l_fsl, l_cv) # Drawing not done yet _return _false _endmethod \$ </pre>
 <p><i>a Common</i></p> <p><i>Incorrect order 2022-01 and earlier</i></p>	 <p><i>Stourbridge Common</i></p> <p><i>Correct order 2023-01</i></p>

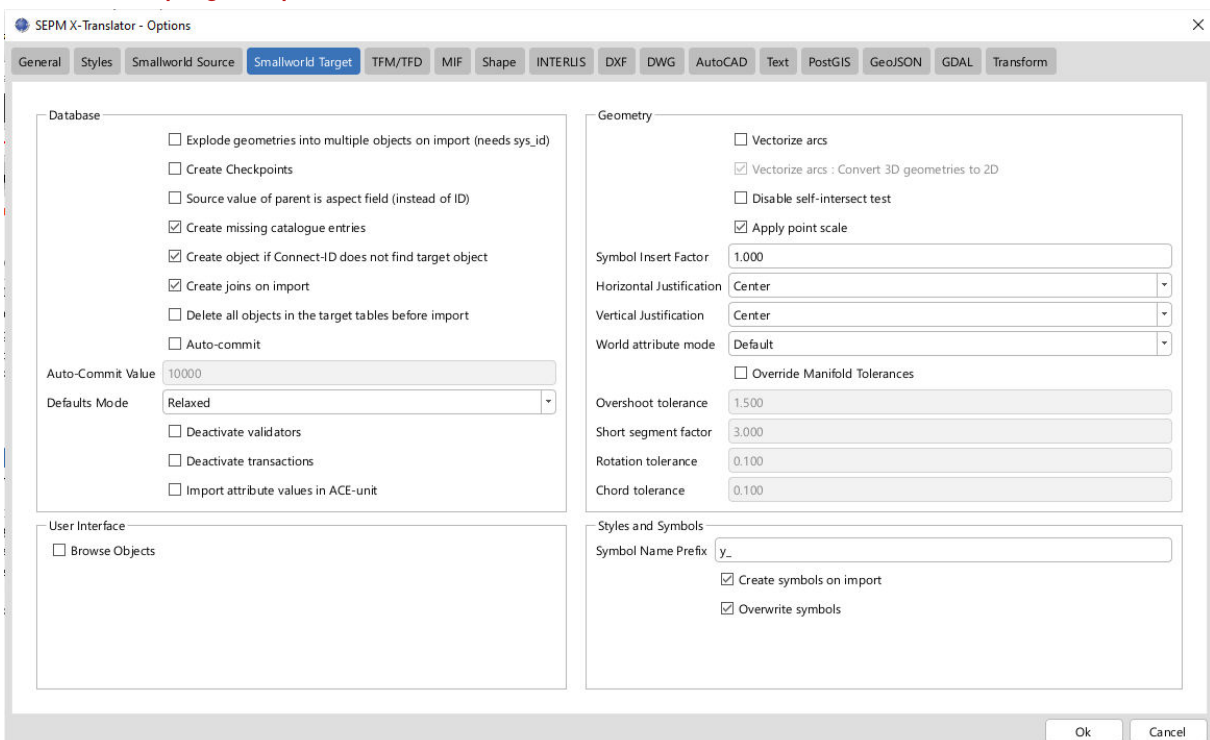
2.4 Smallworld Target Format

2.4.1 Import attribute values in the ACE unit

It is now possible to import attribute values in the unit configured in the ACE. The setting applies globally to all attributes, i.e. either you import the values in the stored unit or in the unit set in the ACE.

Option	Description
Import attribute values in ACE unit	Import attribute values in the configured display unit in the ACE. The ACE is selected using the "Get Trail" command in the Smallworld source dialog.

2.4.2 Grouping of Options



The options in the Smallworld Target tab have been grouped into the following areas:

- ❖ Database
- ❖ User Interface
- ❖ Geometry
- ❖ Styles and Symbols

2.4.3 New Option 'Delete all objects in the target tables before import'

The following use case is covered by this new option: Smallworld objects should be created regularly from a data source. Existing Smallworld objects from a previous import are deleted first and replaced with the newly imported objects. When the option is selected, the message "Delete <Number> objects in collection: <Collection>" is displayed in the logger.

2.4.4 New Option 'Deactivate Validators'

With this option, the method `dd_record_mixin.dd!run_validators()` is replaced with an empty method during the loading process in order to achieve maximum performance.

This option should only be used in special cases and with extreme caution.

2.4.5 New Option 'Deactivate Transactions'

Normally a new database object is created by the Smallworld Transaction API with the commands:

```
_local l_rt << record_transaction.new_insert(..)
l_rt.run()
```

This ensures that application-specific actions are called before or after the insertion, similar to inserting the object in the editor. When the new option is selected, the object is inserted without transactions (with `new_detached_record()` and `make_geometry()`).

This option should only be used in special cases and with extreme caution.

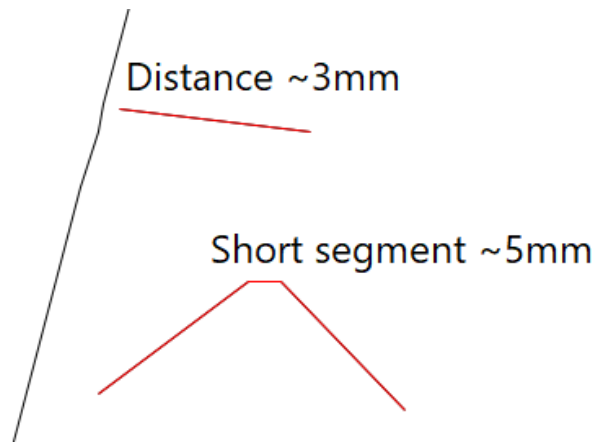
2.4.6 Override Manifold Tolerances

With new options manifold tolerances can be changed during a transfer. When the checkbox 'Override Manifold Tolerances' is selected, the given tolerance values (Overshoot, short segment, rotation and chord) are used for all manifolds defined in the session during the transfer process. The values before the transfer are restored after the end of the process.

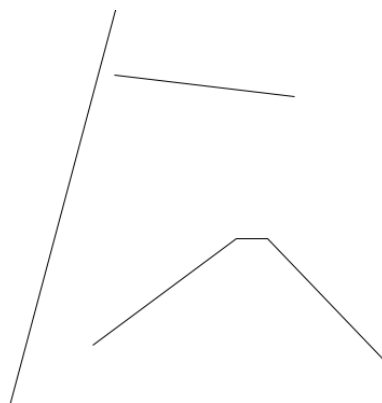
Override Manifold Tolerances

Overshoot tolerance	1.500
Short segment factor	3.000
Rotation tolerance	0.100
Chord tolerance	0.100

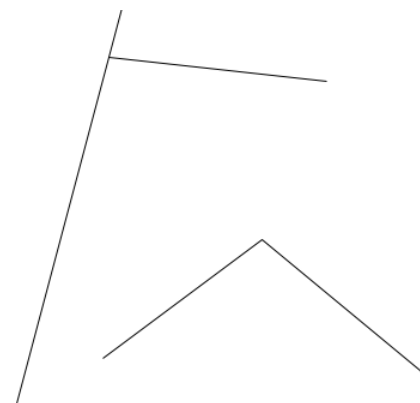
New options to override manifold tolerances



Data to be imported selected in red.



Import with default tolerances



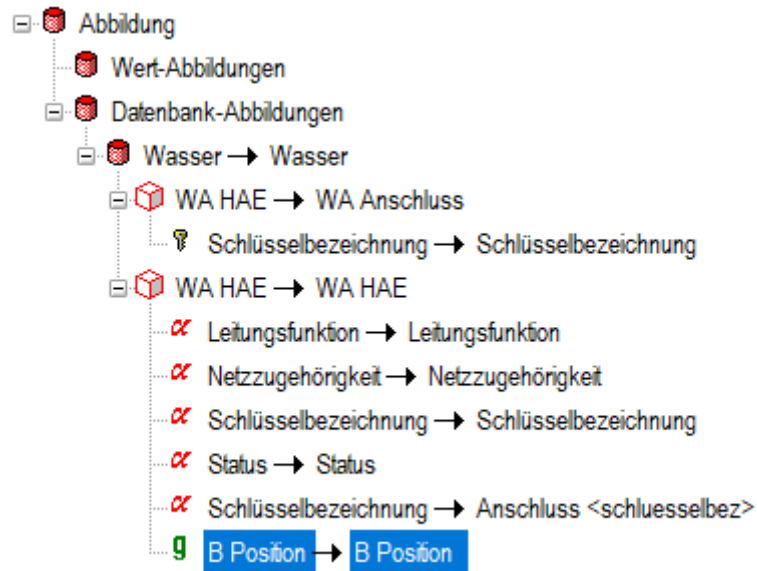
Data imported with overshoot tolerance set to 5mm and short segment factor to 7mm

Of course, you should only change the tolerances with due caution and follow the relevant manufacturer's instructions.

2.4.7 Support for 0:0 Relations

0:0 relationships were not displayed in the Smallworld target format. This was an unnecessary restriction, 0:0 relationships are now treated exactly the same as a relationship to the parent of a 1:n relationship.

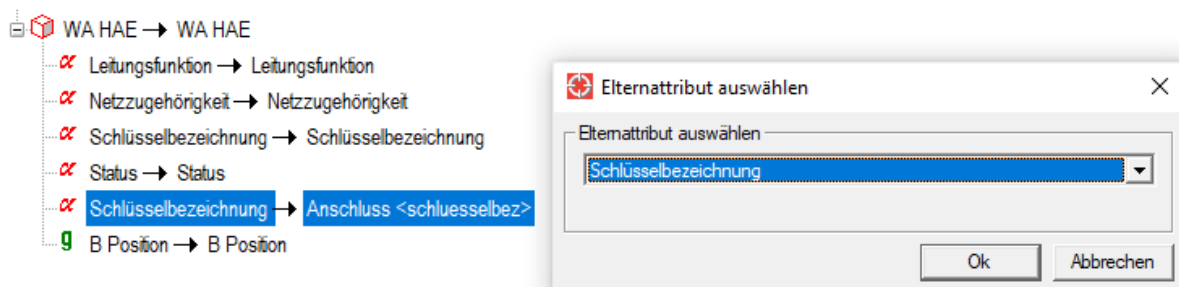
As an example use case, let's consider the following figure:



Use case WA HAE - WA House Connection

In the case of the NRM water application, there is a 0:0 relationship between the objects WA HAE and WA House Connection. Point objects are to be loaded after WA HAE and WA House Connection, and at the same time the relationship is created via the key designation (Schlüsselbezeichnung):

1. A WA House Service object is searched for or newly created via the first mapping defined with "Connect-ID". The option "Create new object if target object not found by Connect-ID" creates a new WA House Service if necessary.
2. The WA HAE is formed via the second collection mapping. Here the key designation has now been mapped to the 0:0 join field. In order for the WA House Service to be found, the key designation must be selected via "Select parent field..." to correctly resolve the join.



Selecting the parent field for the join WA HAE - WA House Connection

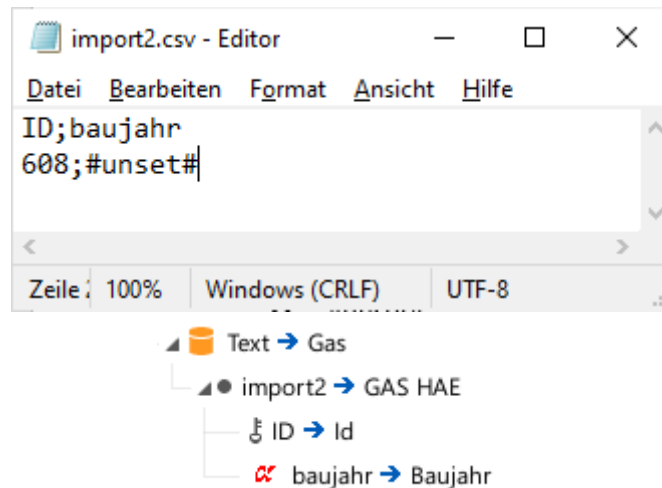
2.4.8 Convert attribute values to _unset

With a change when loading to Smallworld, the following strings are converted to the _unset value:

- ❖ "#unset#"
- ❖ "unset"

This was previously only possible using the special value "#unset#" in the text source format (version 2021-02) and now works with all source formats.

As an example, with Connect-ID you want to delete the existing value for the year of construction on the object GAS HAE with ID 608. This can be done with the following file:



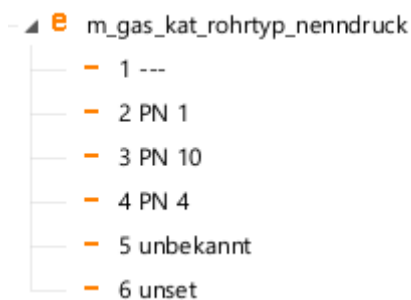
Deleting an attribute value via connect ID and source values "#unset#"

2.4.9 Catalogue Tables with _unset Values

If catalog tables and associated attributes were defined with _unset values (unset_value), mapping was previously only possible with the value "#unset#" defined in text format from version 2021-02.

The defined _unset value is now also added in the corresponding X-Translator enumerator so that it can be used as a source value. "unset" and "#unset#" can now also be used in all source formats.

As an example a catalog definition in the NRM Gas application:



```
Magik> print(v.collections[:m_gas_la].record_exemplar.catalogue_entries[:nenndruck])
$
simple_vector(1,2):
1      :m_gas_kat_rohrtyp
2      :nenndruck
Magik>
Magik> v.collections[:m_gas_kat_rohrtyp].field(:nenndruck).unset_value
$
"----"
$
Magik> v.collections[:m_gas_la].field(:nenndruck).unset_value
$
"----"
```


In this case, you can now map to the `_unset` value in the Smallworld target model with the following source values:

- ❖ `"_"`
- ❖ `"unset"`
- ❖ `"#unset#"`

2.4.10 Dynamic `!current_world!`

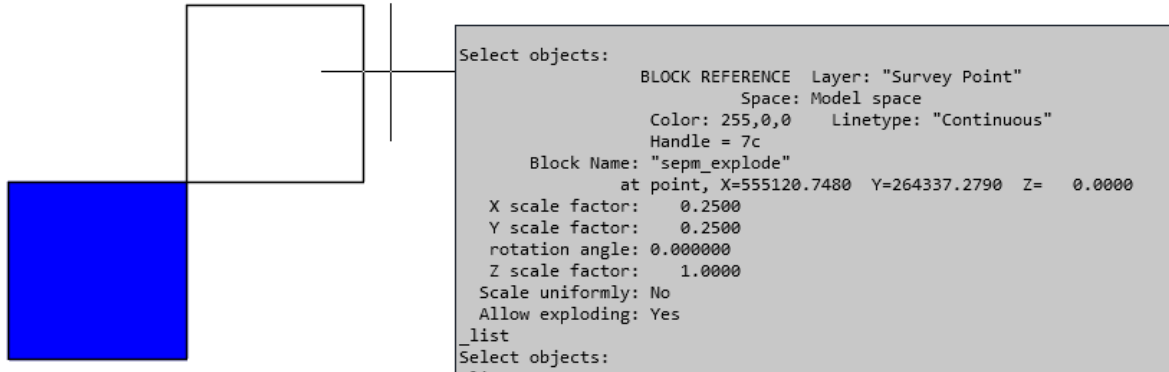
The dynamic variable `!current_world!` is no longer set in the Smallworld target format. Setting the dynamic led to side effects, since the variable was then no longer initialized according to the Smallworld standard functionality, `rwo_record_mixin.make_non_mapped_geometry()`. Previously this variable was sometimes assumed in custom code (e.g. triggers or pre-insert/update actions). If this is the case with your source code, this should be changed so that the world is set correctly directly on the corresponding pseudo-geometries.

If you have any questions on this topic, SEPM Support (sepmworldwide.zendesk.com) will be happy to help.

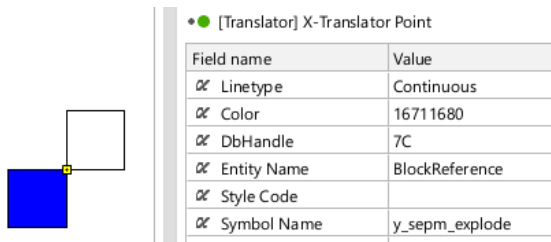
2.5 AutoCAD DWG/DXF Source Format

2.5.1 New Option 'Explode blocks'

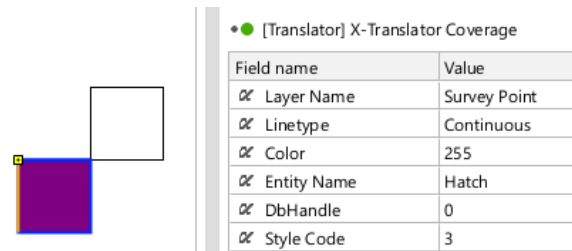
With the option 'Explode blocks' blocks (entity INSERT) in the DWG/DXF file are completely exploded into their surface, line and text components.



Source file: inserted block, the block consists of an area and several line components.



'Explode blocks' option not selected (default): import as point geometry

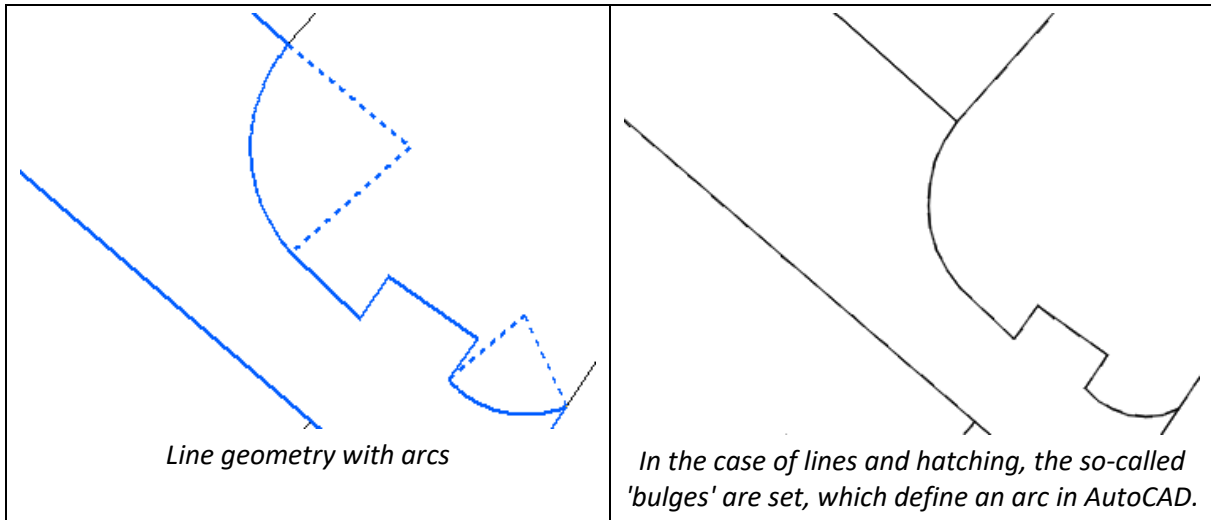


'Explode blocks' option selected: Import of the individual components. In the example, the surface component of the exploded block is selected.

2.6 AutoCAD DWG/DXF Target Format

2.6.1 Export of Arcs and Circles

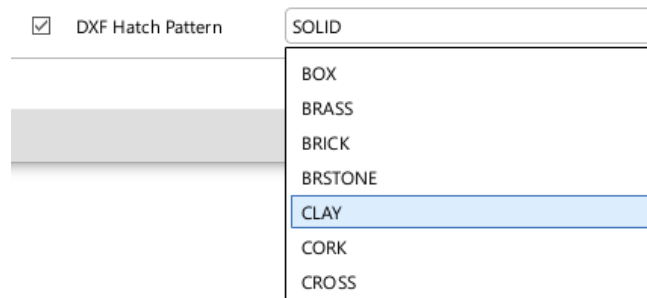
Arcs and circles are now supported in the AutoCAD DWG/DXF target format.



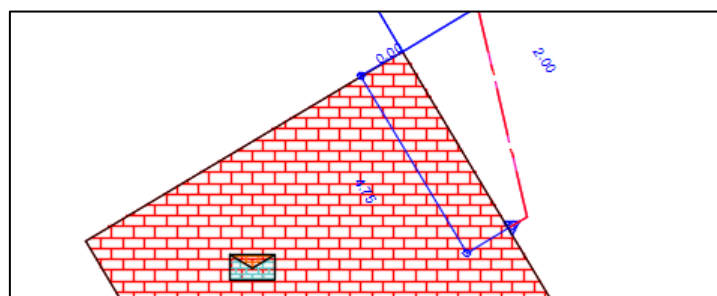
Please note that the export of circular arcs is only possible with line types LWPOLYLINE and POLYLINE/VERTEX2D. When outputting with the line type POLYLINE/VERTEX3D, the lines and area borders are still vectorized.

2.6.2 Selection of a Standard Hatch Pattern in the Model Properties

One of the standard DXF hatch patterns can now be selected in the model properties:



Selection of a DXF hatch pattern

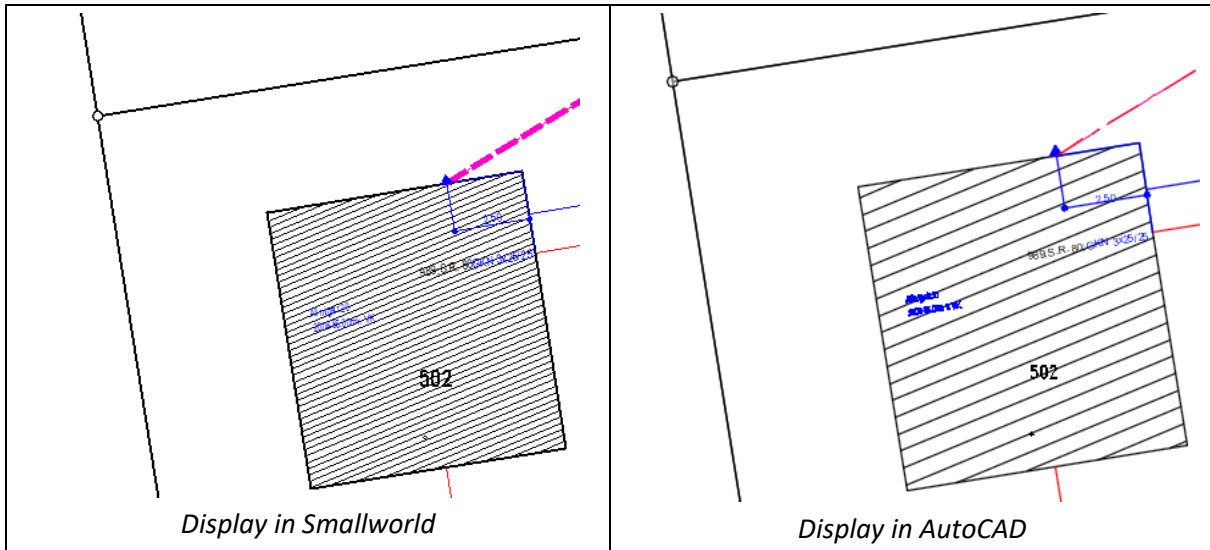


Output of an area with predefined hatch pattern 'BRICK'

2.6.3 Mapping of Hatch Styles

Simple hatch styles (just one line layer) are now supported. The value of the line spacing in pixels is used directly for the target unit (e.g. 5 pixels = 5 meters). The new option '*Hatch Pitch Factor (target)*' can be used for fine adjustment.

Option	Description
'Hatch Pitch Factor (target)'	Added scaling factor when converting pixel spacing to target units.



2.6.4 DwgAcp Coordinate Precision

DwgAcp now ignores geometries in the sub-millimeter range by virtue of the *default_coordinate_precision* setting.

```

_pragma(classify_level=advanced,topic={x_translator})
dwgacp.define_shared_constant(
  ##
  ## Coordinates from the ACP are rounded to this precision:
  ## There seem to be numerical differences at hatch edges,
  ## for example:
  ## last coord of arc:
  ## coordinate:(3481350.9917381142422,5463945.7576170036290)
  ## first coord of subsequent sector:
  ## coordinate:(3481350.9917381151304,5463945.7576170036290)
  ## The x coordinate is different after the 9th digit.
  ##
  ## Numerical inaccuracies below this threshold should be
  ## ignored by the dwgacp. A good default is about 1/1000th
  ## of the smallest relevant precision (e.g. 1mm in Smallworld).
  ## Since most of the GIS data read and written by the
  ## SEPM X-Translator is in meters, the default precision is set
  ## to 0.000001 (A 1/1000th of a millimeter expressed in meters).
  ##
  :default_coordinate_precision,
  0.000001,
  :public)
$

```

2.7 Shape Target Format

2.7.1 Option "Write Shape Null geometries"

In the "All" export mode of the Smallworld source format, no object was exported if the 'Write Shape Null geometries' option was not set and the object had no geometry. This behavior has been changed so that a shape feature is also created for each source object in this case. The exact behavior is given in the following table:

Description of the test data used in the table: test object with a string attribute (name) and three line geometries (Line1, Line2, Line3)

Id	Name	Line1	Line2	Line3
1	A			
4	B			
8	C			
13	D			

Export Mode	Option not set (Standard)	Option set
Trail	<p>Three shape files created, one record per geometry. Object D is not exported.</p> <p>Line1 contains 1 record.</p> <p>Line 2 contains 2 records.</p> <p>Line 3 contains 3 records.</p>	<p>Three shape files created, one record per geometry. Object D is not exported.</p> <p>Line1, Line2 and Line3 contain 3 records.</p> <p>For Line2, object A has a NULL geometry.</p>
All (2019-02 to 2022-01 and 2012-02 and earlier)	Object D is not exported.	Object D is exported in all shape files.
All (2023-01 and newer)	<p>Object D is exported:</p> <ul style="list-style-type: none"> ❖ In the shape file of the first geometry (Line1) or ❖ For the geometry defined as 'Main Geometry' in the model properties 	Object D is exported in all shape files.

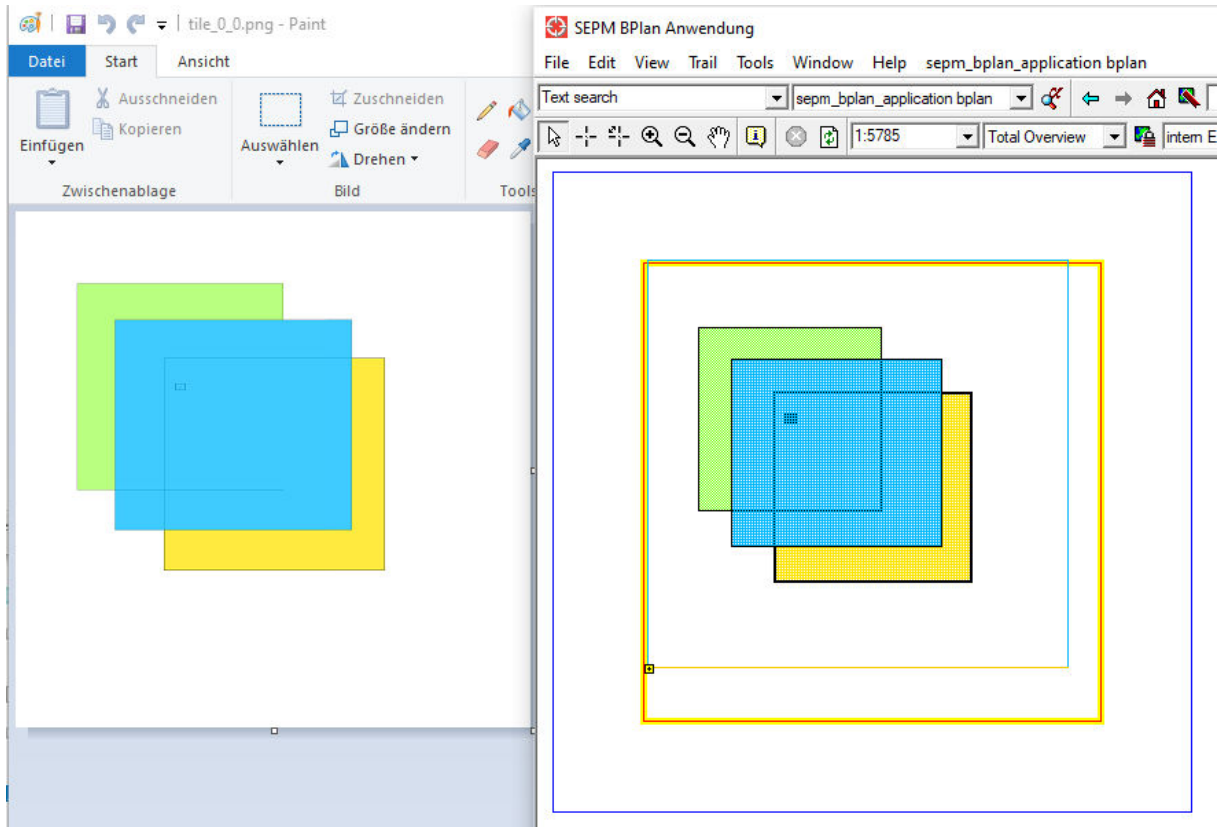
New behavior in version 2023-01 (highlighted in yellow)

3 SEPM X-Raster

3.1 SEPM X-Raster Export

3.1.1 Order of BPLAN Worlds

The order of the worlds in the world manager was not taken into account. This has now been corrected (patches 4497 and 4498).



Consideration of the order of the BPLAN worlds in the X-Raster export

4 NEPLAN

4.1 SEPM NeplanAcp

4.1.1 Job Server Integration

The NeplanAcp functions can now be called as a job:

As an example, a load flow calculation with a new generator on node "NX1":

```
# L1
_local l1_props << property_list.new_with(
  :library_path, l_library_path,
  :project_path, l_project_path,
  :diagram_properties_path, l_diagram_properties_path,
  :export_project_path, l_neplanacp_dir + "export_l1.nep360",

  :new_disperse_generator,
  property_list.new_with(
    :node_name, l_node_has2,
    :disperse_generator_name, "NX1",
    :s, 300.0,
    :cos, 1.0 ),

  :load_flow_results_path, l_neplanacp_dir+"load_flow_l1.json"
)

neplan_acp_job_engine.create_job( l1_props )
```

(See `x_translator\modules\job_server\neplan_acp_job_engine\source\neplan_acp_testing.magik`).

The results are written to the corresponding JSON files for further use (parameters `:load_flow_results_path` and `:short_circuit_dach_results_path`).