

# Apache Hop-Plugins

## Implementation of two Plugins for GIS File Formats

### Students



Simon Weibel



Tobias Keel

**Introduction:** Apache Hop is a visual data orchestration and enterprise data integration platform for designing, running, and monitoring data pipelines and workflows without writing code. It has been under continuous development for 25 years and is written in Java under an open-source licence. Users define a sequence of modular transforms in so-called pipelines, which are executed row by row and can run in parallel. These transforms range from generating sample data to executing external processes, enabling the flexible construction of reusable data processes.

Out of the box, Hop does not support geospatial data processing. The Geographic Information System (GIS) plugins from Atol CD fill this gap by providing the essential base features required to handle spatial data within Hop. Moreover, Hop's ability to execute external processes enables the use of tools such as GDAL - one of the most widely used open-source libraries for geospatial data processing - and its ogr2ogr command-line tool, a powerful vector data converter that supports an additional 100 file formats.

**Problem:** The GIS plugins by Atol CD form a closed system. The seven provided transforms, including the two dedicated to reading and writing geospatial files, work with a custom geometry field type based on the Geometry class of the Java Topology Suite (JTS) library. This geometry field type is required for optimised execution of operations. This means that alternative representation formats, such as Well-Known Text (WKT) and Well-Known Binary (WKB), as well as point coordinates split across two fields (lat/lon), must be converted to this geometry type. Without this conversion, external sources such as Excel or CSV files cannot be accessed by the GIS plugins.

Regarding ogr2ogr, it would be beneficial to integrate it more directly into Hop instead of running it externally. This would also resolve the current issue of the GIS file input transform crashing, as it does not wait for preceding transforms to finish.

**Result:** Two plugins were developed to tackle these problems: Geometry Fields Converter and OGR Vector Import/Export. See Fig. 1 for the software stack.

The Geometry Fields Converter transform allows users to interchangeably convert between WKT, WKB and point coordinate representations. To ensure compatibility with GIS plugins, it is necessary to adjust the field metadata beforehand using the Select values transform. This approach has the advantage of decoupling the plugin from the GIS plugins, enabling it to operate independently as a standalone component. The implementation converts any supported input into a JTS Geometry object prior to converting it into the desired format, using the library's parsers for validation.

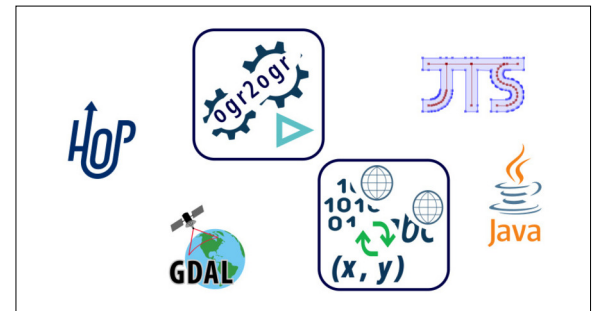
The OGR Vector Import/Export plugin provides a

transform and an action with a dedicated user interface. This includes a dropdown menu of available vector formats and utilises the locally installed GDAL instance to keep the plugin lightweight. Additional options can be defined manually by advanced users in a tabular input field.

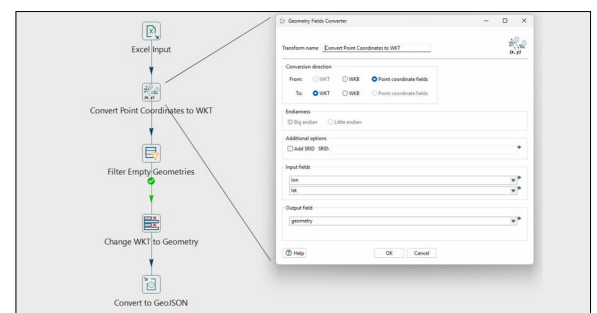
In the future, an option to directly convert WKT into a geometry field may be added to the Geometry Fields Converter.

### Important parts of the software stack of the project with the two plugins in the middle.

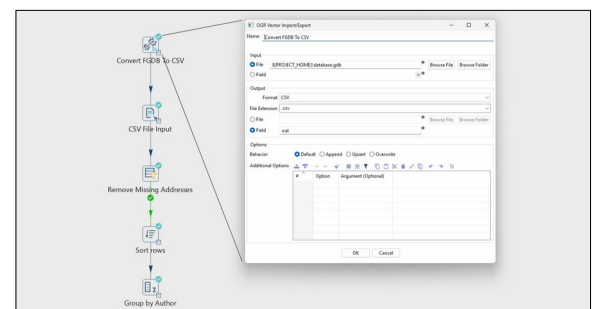
Apache Hop, GDAL, JTS and Java Icons



### Hop pipeline using the Geometry Fields Converter transform to convert the point coordinates of an Excel into a GeoJSON Own presentation



### Using the OGR Vector Import/Export Transform to convert an File-Geodatabase into a CSV for further processing Own presentation



Advisor  
Prof. Stefan F. Keller

Subject Area  
Software, Software Engineering