RMLWeaver-JS: An algebraic mapping engine for KGCW challenge 2024

Sitt Min Oo, Tristan Verbeken, and Ben De Meester

imec - IDLab - Ghent University

RMLWeaver-JS in Track 2 on KGC Parameters

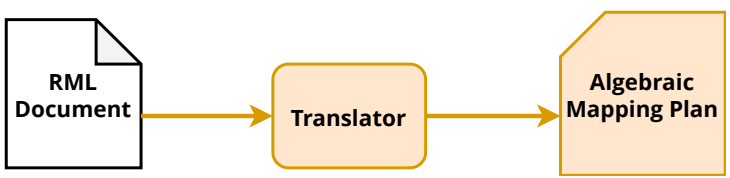
Constant memory usage (550 MB)

Linear execution time increase wrt input size

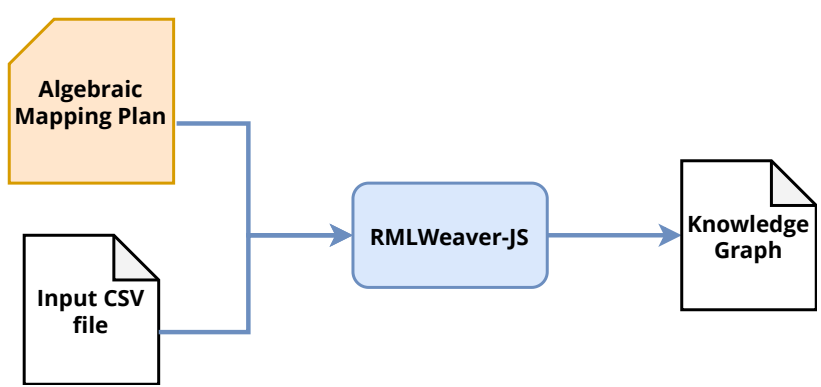
Minimal mapping engine implementation changes

Algebraic mapping pipeline in 2 steps

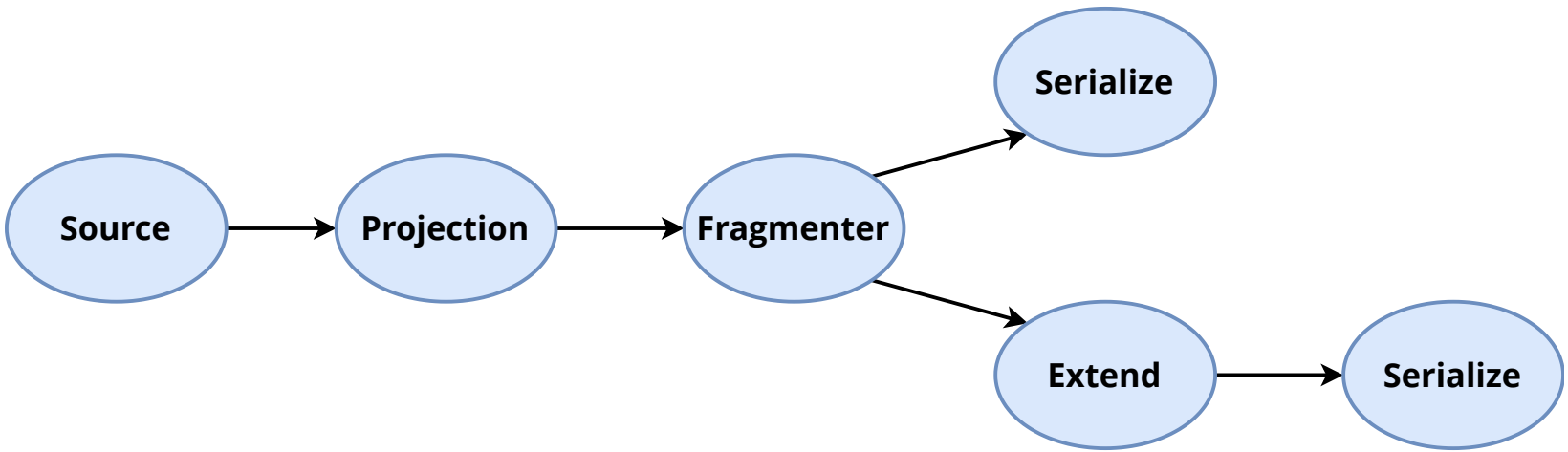
Interpretation of RML in Rust



Execution of mapping process in JavaScript



Algebraic mapping plan



Reactive programming paradigm with RxJS

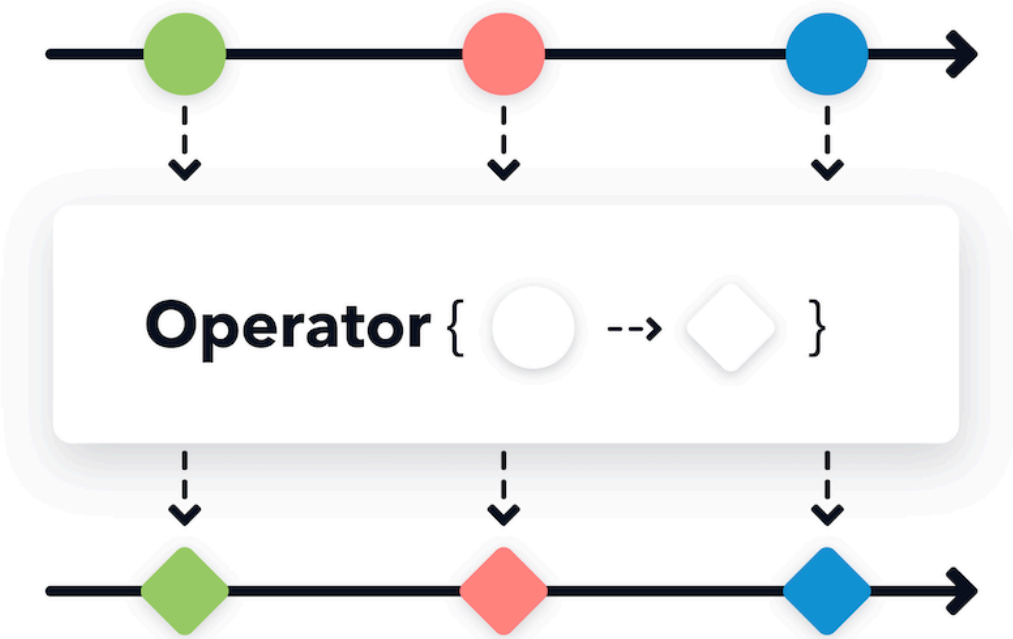


Image credit

Limitations of the algebraic mapping engine

Only support files with **CSV formats**

Does not ignore **empty values**

Does not remove **duplicate triples**

Evaluation of RMLWeaver-JS in Track 2

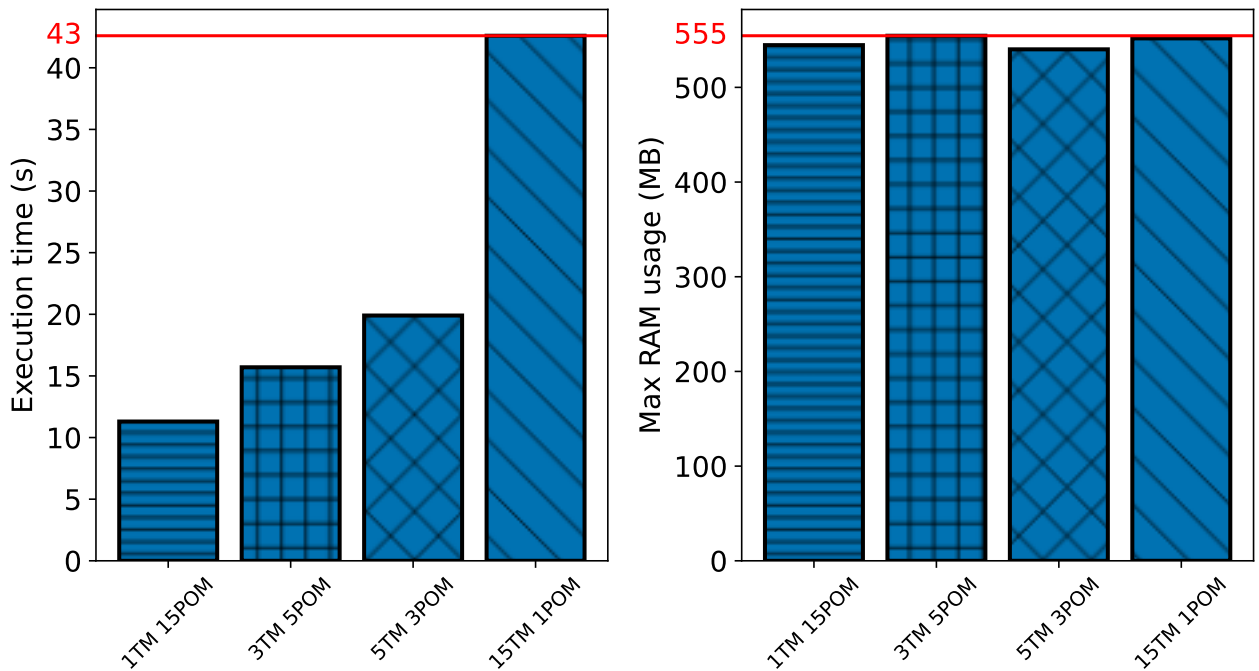
Evaluated for Knowledge Graph Construction Parameters

Skip GTFS test cases

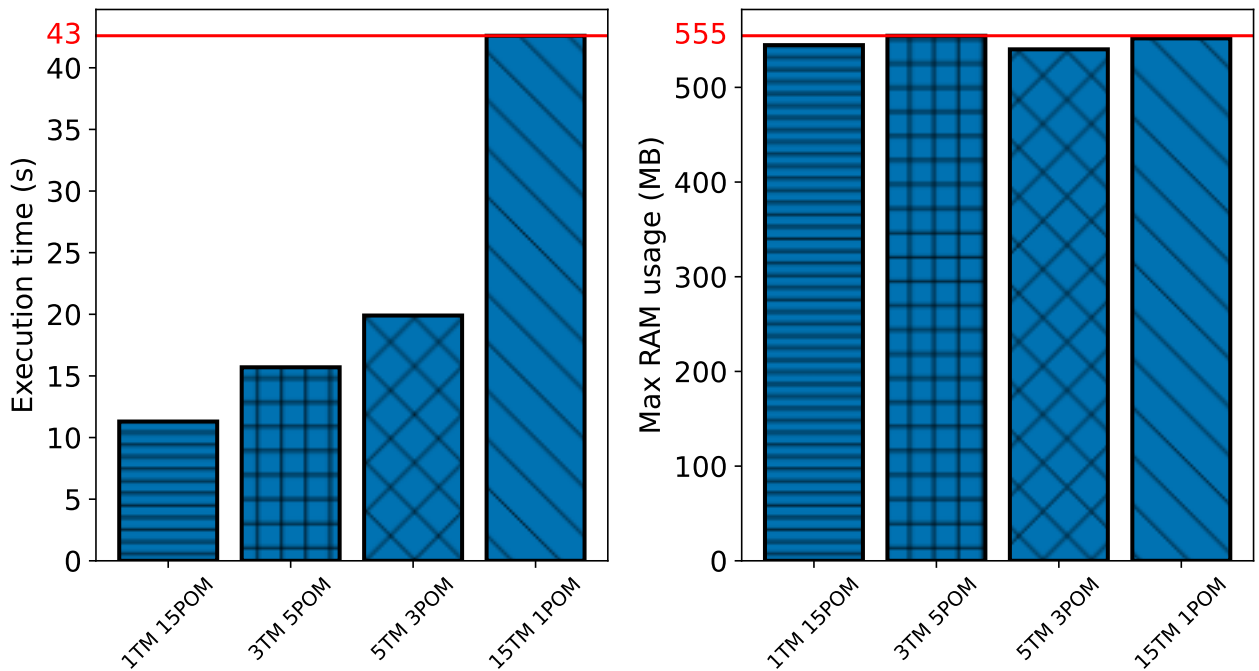
Interpreter does not handle self-joins

Generation of 40 GB of triples for **heterogeneity** test

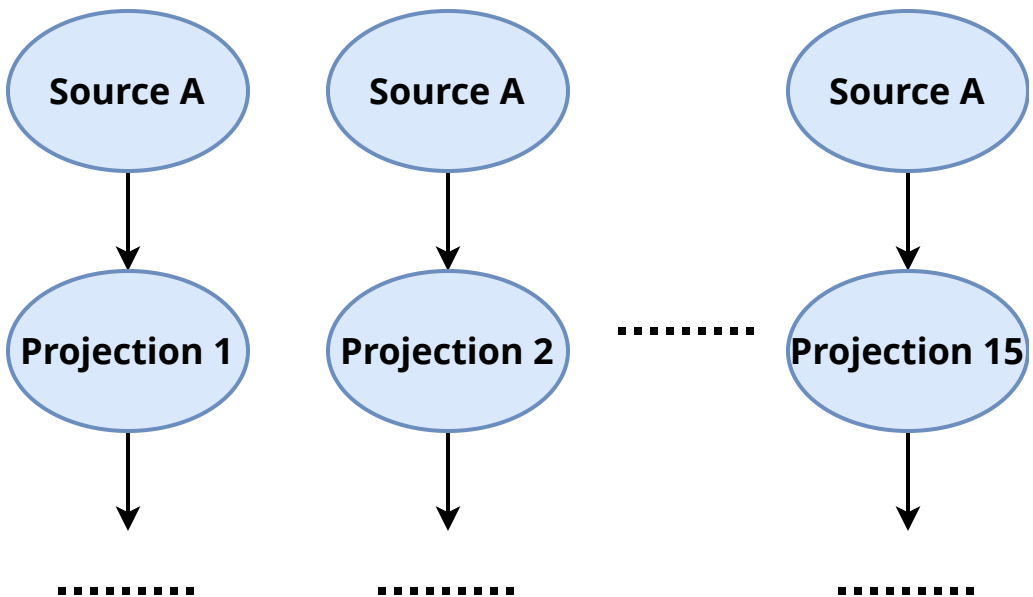
Constant memory usage of around 550MB



Spike in execution time for 15TM-1POM



Inefficient mapping plan with duplicate source operators



Same performance for Joins 5-5 and Joins 10-5

Test case	Execution time (s)	CPU usage (s)	Max RAM (MB)
Join 5-5 25%	35.2	38.8	673
Join 5-5 100%	82.0	90.9	814
Join 10-5 25%	35.2	39.5	661
Join 10-5 100%	82.3	91.0	784

Minimum of N,M determines the performance in Join N-M test case

Hash-join with two HashMaps for bookkeeping

RMLWeaver-JS: An algebraic mapping engine

Interpreter engine: https://github.com/s-minoo/algemaploom-rs

RMLWeaver-JS: https://github.com/RMLio/rmlweaver-js

Source