Groupe de recherche interdisciplinaire en informatique de la santé

Faculté des sciences et Faculté de médecine et des sciences de la santé
Université de Sherbrooke, Québec

Generating relational data models using an ontology a evaluation
RR2020-002

Authors : Christina Khnaisser, Luc Lavoie, Jean-Françaois Ethier

Type : Evaluation

Versions :

version	date	auteur	description
1.1.0a	2021-03-23	CK	Tables update
1.0.0a	2018-06-23	CK	Version derived from Khnaisser et al. 2018

Plan

Abstract

This report presents an evaluation of various ontologies of different sizes and the resulted RDB scripts can be executed on PostgresSQL v9.6+. More detailed results including the RDB scripts can be found on GitHub: https://github.com/OpenLHS/OntoRela.

GRIIS (Groupe de recherche interdisciplinaire en informatique de la santé)
Faculté des sciences et Faculté de médecine et sciences de la santé
Université de Sherbrooke
Sherbrooke, Québec J1K 2R1.

1 Evaluation

The prototype was tested with various ontologies of different sizes and the resulted RDB scripts can be executed on PostgresSQL v9.0+. More detailed results including the RDB scripts can be found on GitHub: https://github.com/OpenLHS/OntoRela.

1.1 Experimental results

Some of the experimental results are showed in the following three tables. Each one presents a specific view. Table a,b shows a summary of the number of constructs in the original ontology, the reduced ontology generated by OntoRelα and the RDB generated from the reduced ontology with the execution time. The number of constructs in the original ontology includes: the number of classes (classes), the number of data properties (DP), the number of object properties (OP), and the total number of axioms ({Ax-O total). For the reduced ontology the number of axioms generated by OntoRelα (Ax-G total) and the number of complex class association axioms reduced OntoRelα (Ax-R total) are presented. Finally, in Table (c), to evaluate the complexity of the ontology in relation with the complexity of the RDB, we defined a complexity measure ($CPLX = (Tables + FK + Func)/Ax-O total$) based on the ratio between the number of relational constructs and the number of ontological constructs. This allows to have a qualitative idea about the complexity of managing the evolution of the structure and the data (nsert and update).

(a)

Ontology	**Classes**	**DP**	**OP**	**Ax-O total**	**Ax-G total**	**Ax-R total**	**Tables**	**FK**	**Func.**	**Exec. (sec)**
VSO	33	16	38	87	99	0	105	162	71	3
MONDIAL	52	28	59	139	114	0	134	192	82	1
UNIBENCH	43	7	25	75	54	0	64	83	20	1
PIZZA	100	0	8	108	234	0	250	400	150	2
GENO	393	9	123	525	483	8	507	617	108	4
ESCO	561	2	23	495	794	61	872	1 172	288	14
CHMO	3 078	0	25	3 103	3 337	0	3 360	3 736	281	38
PDRO	3 952	12	131	4 095	7 764	543	7 510	10 833	2 964	150
PDRO-P3	3 963	15	139	4 117	7 970	554	7 684	11 133	3 127	154
IFAR	4 530	0	0	4 530	4 977	0	4 531	4 984	0	56
GAMUTS	18 001	0	0	18 001	6 249	0	18 002	18 720	0	347
OAE	5 700	3	120	5 823	14 399	764	11 629	20 328	5 889	503
ONTOLURGENCES	10 031	0	60	10 841	12 247	6	11 491	13 765	1 455	529
ENVO	8 511	0	63	8 574	14 259	335	11 026	16 499	2 211	537

(b)

Ontology	**Classes**	**DP**	**OP**	**Ax-O isa**	**Ax-O class**	**Ax-O data**	**Ax-G isa**	**Ax-G class**	**Ax-G data**	**Exec. (sec)**
VSO	33	16	38	28	58	13	28	58	13	3
MONDIAL	52	28	59	32	66	26	32	56	26	1
UNIBENCH	43	7	25	34	20	0	34	20	0	1
PIZZA	100	0	8	84	178	0	84	150	0	2
GENO	393	9	123	366	96	6	380	101	2	4
ESCO	561	2	23	518	276	0	518	276	0	14
CHMO	3 078	0	25	3 056	251	0	3 056	281	0	38
PDRO	3 952	12	131	4 153	2 087	25	4 879	2 860	25	150
PDRO-P3	3 963	15	139	4 165	2 200	93	4 933	2 944	93	154
IFAR	4 530	0	0	4 977	0	0	4 977	0	0	56
GAMUTS	18 001	0	0	6 249	0	0	6 249	0	0	347
OAE	5 700	3	120	6 294	4 588	0	9 268	5 130	1	503
ONTOLURGENCES	10 031	0	60	10 799	1 447	1	10 799	1 447	1	529
ENVO	8 511	0	63	11 569	1 825	0	12 169	2 090	0	537

(c)

Ontology	Ax-O total	Tables	FK	Func.	CPLX	Exec. (sec)
VSO	87	105	162	71	3,41	3
MONDIAL	139	134	192	82	3,58	1
UNIBENCH	75	64	83	20	3,09	1
PIZZA	108	250	400	150	3,42	2
GENO	525	507	617	108	2,55	4
ESCO	495	872	1 172	288	2,94	14
CHMO	3 103	3 360	3 736	281	2,21	38
PDRO	4 095	7 510	10 833	2 964	2,74	150
PDRO-P3	4 117	7 684	11 133	3 127	2,75	154
IFAR	4 530	4 531	4 984	0	1,91	56
GAMUTS	18 001	18 002	18 720	0	5,88	347
OAE	5 823	11 629	20 328	5 889	2,63	503
ONTOLURGENCES	10 841	11 491	13 765	1 455	2,18	529
ENVO	8 574	11 026	16 499	2 211	2,09	537

Legend:

Classes number of classes defined by the original ontology

DP number of data properties defined by the original ontology

OP number of object properties defined by the original ontology

Ax-O number of axioms defined by the original ontology

isa inheritance axioms

class class association axioms

data data association axioms

total total number of axioms (sum of the three categories)

Ax-G number of axioms generated by OntoRelα

Ax-R number of complex class association axioms (Ax-I class) reduced

Tables number of tables in the RDB

FK number of referential keys in the RDB

Func. number of participation functions in the RDB

Exec. execution time in seconds

CPLX Complexity measure (Tables + FK + Func)/Ax-O total

1.2 Analysis

As expected, we can observe that the time execution depends on the number of axioms. This comes with no surprise as the core of OntoRelα algorithms concentrates on analysing axioms. Also notice that ontologies of biomedical domain contain a high number of complex axioms (number of Ax-R is high) which explains the higher number in the generated axiom (Ax-G).

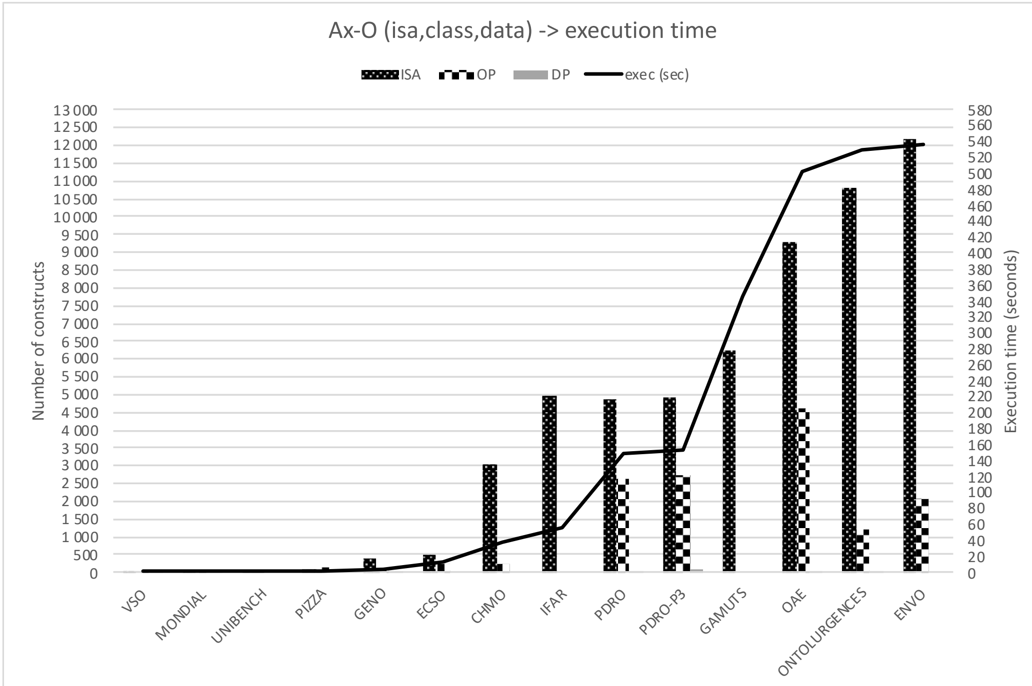

Figure 1. Graphic for part of table 5(a) results

The size of the input ontology (expressed in terms of the number of axioms isa, class and data) is a good predictor of the execution time for ontologies whose total number of axioms exceeds a certain threshold. More tests are required to evaluate this threshold, which appears to be between 1000 and 2000. This is easily explained by a constant initialisation time likely of the order of 1 second. Finally, further experimentation and metrics must be put in place to show the value of the conversion outcomes as the usability of the resulted schema for modelling, querying and maintaining. Also, datasets and queries must be defined to feed and test the performance of the resulted database.