I attached below the following publications:

[1] Hoang Nguyen Phuoc Bao: *Intelligent enforcement of Fine-Grained Access Control for SQL queries.* Master thesis. Universidad Autonoma de Madrid, Spain. 2021.

[2] Hoang Nguyen Phuoc Bao, Manuel Clavel: *A Model-Driven Approach for Enforcing Fine-Grained Access Control for SQL Queries (Extended version).* Journal of SN Computer Science, volume 2(5). Springer. 2021.

[3] Hoang Nguyen Phuoc Bao, Antonio Garcia-Dominguez, Manuel Clavel: *The TTC 2021 OCL2PSQL Case.* Proceedings of Workshop TTC@STAF 2021. CEUR Workshop Proceedings. (Accepted for publication).

[4] Hoang Nguyen Phuoc Bao, Manuel Clavel: *A Model-Driven Approach for Enforcing Fine-Grained Access Control for SQL Queries.* Proceedings of FDSE: International Conference on Future Data and Security Engineering 2020. Lecture Notes in Computer Science, volume 12466. Springer. 2020.

[5] Hoang Nguyen Phuoc Bao, Manuel Clavel: *Model-based Characterization of FGAC authorization for SQL Queries.* Journal of Object Technology. 2020, volume 19(3):1–13. 2020.

[6] Hoang Nguyen Phuoc Bao, Manuel Clavel: *OCL2PSQL: An OCL-to-SQL Code-Generator for Model-Driven Engineering.* Proceedings of FDSE: International Conference on Future Data and Security Engineering 2019. Lecture Notes in Computer Science, volume 11814. Springer. 2019.

[7] Manuel Clavel, Hoang Nguyen Phuoc Bao: *Mapping OCL into SQL: Challenges and Opportunities Ahead.* Proceedings of Workshop OCL@MoDELS 2019. CEUR Workshop Proceedings, volume 2513. CEUR-WS.org. 2019.

In a nutshell, [7] discusses, in the context of model-driven engineering, some of the challenges and opportunities of the mapping from the Object Constraint Language (OCL) to the Structure Query Language (SQL). [6] defines a novel mapping, called OCL2PSQL, that overcomes some of the aforementioned challenges. [3] proposes the definition of this OCL2PSQL mapping as a case to study the readiness of the state-of-the-art transformation tools. [5] formally characterizes the enforcement of Fine-Grained Access Control (FGAC) policies for a subset of SQL queries. [4] then proposes a model-based methodology for developing an FGAC-enforcement mechanism following our formal characterization. [2] extends [4] with empirical results about the impact in execution-time efficiency of proposed enforcing mechanism. Finally, [1] proposes a model-driven methodology for minimizing the performance impact mentioned in [2].