Odlaw: A Tool for Retroactive GDPR Compliance

Connor Luckett, Andrew Crotty, Alex Galakatos, Ugur Cetintemel
Department of Computer Science, Brown University
firstname_lastname@brown.edu

***Abstract*—In this demo, we present ODLAW, a new tool for retroactive compliance with privacy laws like the European Union's General Data Protection Regulation (GDPR). The GDPR enumerates the explicit rights of individuals regarding the use of their personal data, and regulators can impose strict penalties for organizations that fail to comply. While others have advocated for a completely new class of systems to address these regulations, ODLAW takes a different approach by achieving GDPR compliance while allowing an organization to keep its existing data management infrastructure intact. Using a variety of realistic datasets, the demo will show the specific ways that ODLAW can help with GDPR compliance, as well as highlight some of the key challenges that arise in real-world settings.**

I. INTRODUCTION

In recent years, data privacy has become a critically important topic, with many high-profile breaches (e.g., Yahoo!, Equifax) and misappropriations (e.g., Cambridge Analytica) spurring legislative action. One such law is the European Union's General Data Protection Regulation [1] (GDPR), which outlines the responsibilities of organizations that collect personal data. Similar laws have already been enacted in countries around the world, with more likely to follow.

Failure to comply with the GDPR can incur serious penalties, including large monetary fines and burdensome mandatory audits. Compliance is often nontrivial, though, leading to a patchwork of error-prone, one-off fixes. In fact, many organizations remain unprepared for full compliance [2], and some have even terminated service in Europe to avoid the GDPR altogether [3].

The most straightforward path to achieving GDPR compliance is a proactive redesign of the entire data management stack to meet all requirements [3], with some even advocating for an entirely new class of systems [4], [5]. However, it is unrealistic to expect organizations to undertake a costly transition to these fledgling systems, and there is no guarantee that they will remain robust to future regulatory changes.

As an alternative, others have explored the possibility of retrofitting existing systems with the additional functionality necessary for GDPR compliance [6], [7]. While we believe this approach is a step in the right direction, it can have negative performance impacts on day-to-day operations and still requires substantial changes to existing infrastructure.

Therefore, we believe that the most realistic and practical option is retroactive compliance through the use of external tools, which would allow an organization to meet regulatory requirements while still retaining its existing data management stack. In this setting, several interesting research challenges arise, including: (1) handling messy and complex real-world schemas; (2) identifying siloed or duplicated personal data to ensure full compliance; and (3) tracking how personal data is used, even after leaving the core data management stack.

This demo presents ODLAW, a new tool we are developing to help organizations achieve retroactive GDPR compliance. ODLAW is designed specifically to interface with existing systems in a bolt-on fashion, thereby enabling adherence to core GDPR provisions without requiring a complete infrastructure overhaul. As part of the demo, we will use several realistic datasets to show the main functionality of ODLAW, as well as to highlight some of the key challenges of guaranteeing full GDPR compliance in real-world settings.

II. GDPR BACKGROUND

In 2016, the European Union adopted the General Data Protection Regulation [1] (GDPR), which establishes a unified regulatory environment in the European Economic Area for the collection, management, and processing of personal data. The GDPR has influenced the creation of similar laws worldwide, including the recently enacted California Consumer Privacy Act in the United States. With the possibility of facing hefty fines for violations, GDPR compliance has become a major concern for impacted organizations.

Specifically, the GDPR classifies individuals as "data subjects" with certain enumerated rights and "data controllers" as organizations that collect personal data from data subjects. In the following, we outline three key rights that data controllers must consider from a technical perspective.

A. Right of Access

The Right of Access empowers data subjects to retrieve and examine any personal data collected by the data controller, as well as information about how the data controller processes that data (e.g., collection methods, purposes of the processing). Data controllers typically have a one-month deadline to comply with data access requests.

The related Right of Portability requires that personal data must be exportable in a widely recognized structured data format, such as JSON. In theory, this portability requirement would allow a data subject to transfer their personal data to an alternative data controller without obstruction.

B. Right to Erasure

Data subjects are also entitled via the Right to Erasure to request the deletion of their personal data. Again, data controllers must respond to erasure requests in a timely manner and inform data subjects of any steps taken.

2375-026X/21/$31.00 ©2021 IEEE
DOI 10.1109/ICDE51399.2021.00310

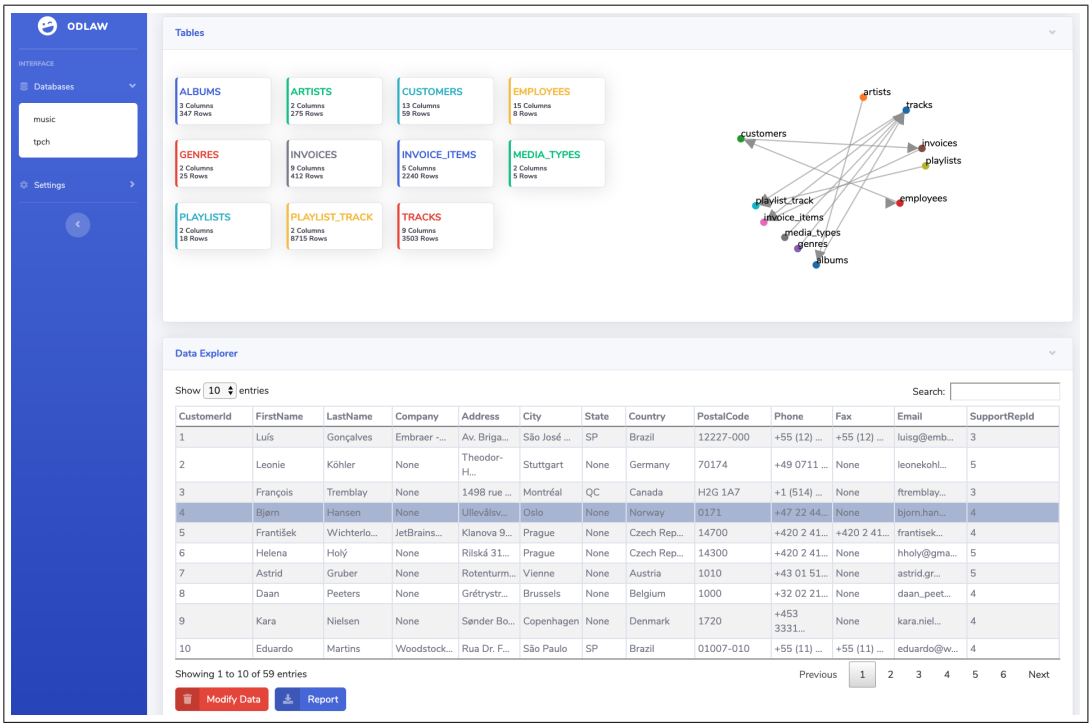

Fig. 1: ODLAW's User Interface

The Right to Erasure does not necessarily require strict data deletion, however, and data controllers can often comply by performing unrecoverable anonymization. For organizations that rely upon or monetize aggregated data, this nuance is of particular importance, since they have some flexibility in handling erasure requests without resorting to total deletion.

C. Right to Object

Finally, the Right to Object allows data subjects to object to specific uses of their personal data, such as for sales or marketing purposes. Data subjects can additionally challenge the outcomes of automated or algorithmic decisions. Compliance can become particularly tricky in cases where personal data is consumed outside of the core data management stack, such as by a machine learning model or after export to a third party.

III. ODLAW

ODLAW is a bolt-on tool designed to help an organization comply with privacy laws like the GDPR without needing to completely overhaul its existing data management infrastructure. In this section, we first provide an overview of ODLAW's design, followed by a discussion of interesting challenges that arise in real-world settings and promising opportunities for future work.

A. System Overview

ODLAW consists of two main parts: (1) an easy-to-use interface geared toward nontechnical users; and (2) a middleware layer that facilitates GDPR compliance by connecting to existing systems in an organization's data management stack.

1) User Interface: One of ODLAW's main goals is to provide an intuitive interface that can help a user with limited technical background to manage sensitive personal data and fulfill GDPR requests from data subjects. Figure 1 shows ODLAW's web interface, which provides visualizations to help the user quickly understand key features of the schema, including important tables and the relationships between them. ODLAW also makes it easy for a data controller to modify, delete, and export all data related to a data subject, enabling painless compliance with key GDPR provisions. We leave a more detailed description of specific functionality to Section IV.

2) Middleware Layer: Similar to an IDEA [8], ODLAW is designed to connect to existing systems in an organization's data management stack. Specifically, we use SQLAlchemy [9] to handle connections to a variety of commonly used DBMSs.

When connecting to a new system for the first time, ODLAW queries the catalog to determine all tables, relationships (i.e., primary and foreign keys), integrity constraints, and any other available metadata (e.g., table statistics). Using this information, ODLAW then constructs a graph that models the entire schema, with each table represented as a node and relationships represented as a directed edge from a primary key to a foreign key. This schema graph allows ODLAW to help users fulfill GDPR requests without any invasive changes to the underlying infrastructure.

For example, suppose a data subject submits a request for a full data export. While some pieces of information directly related to the data subject (e.g., DOB, address, phone number) will be straightforward to retrieve, other derived data (e.g.,

orders, invoices, communications) may be much harder to track down, and the risk of missing pertinent data increases with schema complexity. Rather than compiling these pieces of data manually, ODLAW can automatically generate a report by performing a breadth-first traversal of the graph starting from the node that represents the data subject, thereby ensuring that all derived data will be included.

B. Challenges & Opportunities

On the surface, some aspects of ODLAW may seem straightforward. In practice, however, real-world use cases present several challenges, some of which we discuss here.

1) Complex Schemas: Unsurprisingly, real-world schemas are usually quite complex, having evolved over time to meet changing business needs. Databases may include hundreds or even thousands of tables organized using a wide variety of logical models (e.g., 3NF, BCNF, star schema, snowflake schema). This extreme variability makes it difficult to design a one-size-fits-all solution for automated GDPR compliance.

For example, one challenging design pattern we have encountered in real-world schemas involves circular references. The simplest manifestation of this pattern occurs when a table refers directly to itself, such as an employee with a manager who is also an employee. Circular references can also appear across multiple tables when a series of references creates a cycle in the schema graph. For instance, a cycle would occur if an employee worked for a department where the department manager is another employee. To handle circular references, ODLAW keeps track of all visited nodes when traversing the schema graph and terminates upon detecting a cycle.

2) Entity Resolution: Not only are real-world schemas highly complex, but they also seldom follow strict database design principles, again due in large part to adaptations as business needs change over time. In many cases, data related to a single logical entity might be physically partitioned into many distinct tables (e.g., customers from different regions stored in separate tables), or data might be duplicated to better accommodate organizational needs (e.g., for use by siloed business units). While these design decisions have practical benefits, they significantly complicate many aspects of GDPR compliance.

Consequently, we believe that including entity resolution techniques, ranging from simple crawlers that find matching values to more advanced data discovery methods [10], would make ODLAW much more robust. We could then flag possible matches for human review in the web interface.

Moreover, many schemas do not explicitly declare relationships between all primary and foreign keys, especially when data is partitioned or duplicated. Undeclared references are currently a problem for ODLAW, which relies on the schema graph to identify all derived data related to a data subject. We are therefore investigating approaches for automatically identifying tables that can be joined together [11], [12].

3) Usage Metadata: Finally, as an organization's infrastructure becomes increasingly complex, the ability to manually keep track of sensitive personal data rapidly diminishes. While

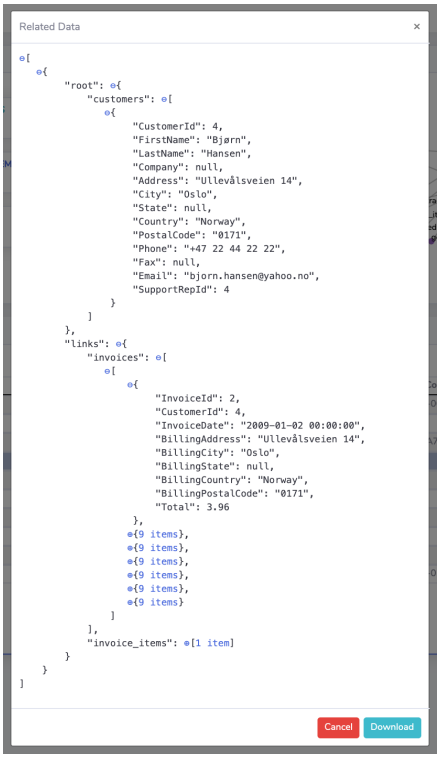

Fig. 2: Report Window

we have already discussed how ODLAW can help to automate some aspects of this process, we have not yet addressed the Right to Object.

Similar to other approaches [13], [14], ODLAW maintains a table of metadata tags associated with individual data subjects. Each tag represents a custom usage policy based on the types of processing performed by a specific data controller. In the future, we plan to incorporate recently proposed automated verification techniques to guarantee compliance with large numbers of complicated and overlapping policies.

Unfortunately, this approach also comes with several drawbacks, including the problem of "metadata explosion" [7] and a lack of standardization for data usage policies. For example, simply exporting all of the usage policy tags that ODLAW stores internally could leak sensitive or proprietary information, and these policies might not even be meaningful to third parties with whom the data is shared.

IV. DEMO DESCRIPTION

In this demo, we plan to allow audience members to freely explore the example datasets through our web interface and experience how ODLAW can help a data controller to comply with data privacy laws like the GDPR. Our example datasets include: (1) *Chinook* [15], which models a digital media store; (2) *Employees* [16], which represents a sample HR use case; and (3) *TPC-H* [17], the well-known decision support benchmark that mimics the database of a wholesale supplier. Although real-world schemas are often significantly

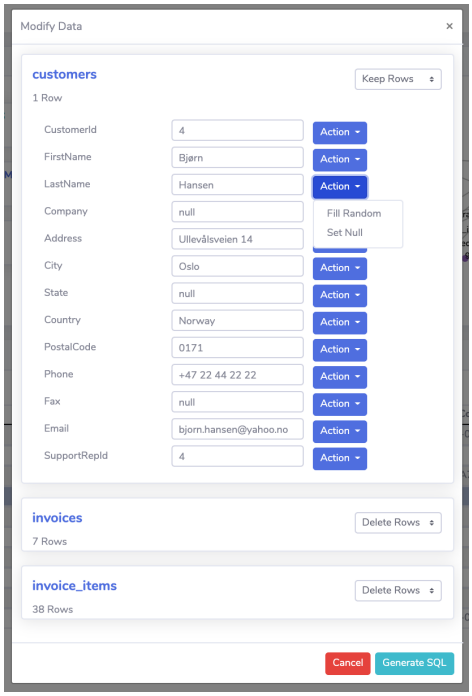

Fig. 3: Modify Data Window

more complicated, these datasets showcase several of the edge cases that make GDPR compliance tricky in practice.

Specifically, audience members will have the opportunity to: (1) discover relationships between tables and uncover derived data that is indirectly related to data subjects; (2) modify or delete personal data to fulfill erasure requests; and (3) identify and export all personal data associated with a data subject in compliance with portability requirements. In the following, we describe an example ODLAW user session.

The user begins by selecting one of the preloaded datasets listed on the left sidebar of the web interface. Based on the selection, ODLAW will populate the schema overview panel with a list of all tables, including the number of columns and rows in each. An accompanying graph visualization depicts the relationships between tables, which provides a high-level overview of the schema and allows the user to identify all entries that might refer to a data subject. For example, when examining the schema graph for the dataset shown in Figure 1, the user can quickly see that data subjects with personal information stored in the `customers` table may also have related entries in the `invoices` table, which in turn may have related entries in the `invoice_items` table.

Clicking on any table in the schema overview panel will present a paginated view of all rows stored in that table, shown in the bottom of Figure 1. The search interface issues queries to the backend DBMS, allowing the user to easily locate entries for a data subject who may have submitted a GDPR request. In the example, the user has selected the data subject "Bjorn Hansen" in the `customers` table, which reveals the *Modify Data* and *Report* buttons.

If the user clicks on the *Report* button, a window will appear that shows an easy-to-understand hierarchical summary of all data pertaining to the data subject, including the primary entry (i.e., *root*) and any derived entries (i.e., *links*). Figure 2 shows the report generated for "Bjorn Hansen" with an option to download the data in JSON format, which satisfies both the Right of Access and Right of Portability.

Similarly, clicking the *Modify Data* button opens a window that allows the user to modify or delete personal data to comply with the Right to Erasure. The *Modify Data* window for "Bjorn Hansen" appears in Figure 3, with the option to edit values both of the primary entry in the `customers` table and derived entries stored in other tables. The user can choose to modify values in any of the text fields, with shortcut options to set a field to a random or `NULL` value, or to delete entries altogether. This window also allows the user to assign usage policy tags, which addresses the Right to Object.

After finalizing all modifications in the web interface, the user then needs to update the underlying database. In keeping with its bolt-on design, ODLAW does not directly modify the database, instead providing the user with a generated SQL script corresponding to the specified changes.

REFERENCES

[1] "General Data Protection Regulation," https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32016R0679, 2016.

[2] "Gartner Says Organizations Are Unprepared for the 2018 European Data Protection Regulation," https://www.gartner.com/en/newsroom/press-releases/2017-05-03-gartner-says-organizations-are-unprepared-for-the-2018-european-data-protection-regulation, 2017.

[3] S. Shastri, M. Wasserman, and V. Chidambaram, "The Seven Sins of Personal-Data Processing Systems under GDPR," in *HotCloud*, 2019.

[4] M. Schwarzkopf, E. Kohler, M. F. Kaashoek, and R. T. Morris, "Position: GDPR Compliance by Construction," in *Poly@VLDB*, 2019, pp. 39–53.

[5] T. Kraska, M. Stonebraker, M. L. Brodie, S. Servan-Schreiber, and D. J. Weitzner, "SchengenDB: A Data Protection Database Proposal," in *Poly@VLDB*, 2019, pp. 24–38.

[6] A. Shah, V. Banakar, S. Shastri, M. Wasserman, and V. Chidambaram, "Analyzing the Impact of GDPR on Storage Systems," in *HotStorage*, 2019.

[7] S. Shastri, V. Banakar, M. Wasserman, A. Kumar, and V. Chidambaram, "Understanding and Benchmarking the Impact of GDPR on Database Systems," *PVLDB*, vol. 13, no. 7, pp. 1064–1077, 2020.

[8] A. Crotty, A. Galakatos, E. Zgraggen, C. Binnig, and T. Kraska, "The Case for Interactive Data Exploration Accelerators (IDEAs)," in *HILDA@SIGMOD*, 2016.

[9] M. Bayer, "SQLAlchemy," in *The Architecture of Open Source Applications Volume II: Structure, Scale, and a Few More Fearless Hacks*, A. Brown and G. Wilson, Eds., 2012. [Online]. Available: http://aosabook.org/en/sqlalchemy.html

[10] R. C. Fernandez, Z. Abedjan, F. Koko, G. Yuan, S. Madden, and M. Stonebraker, "Aurum: A Data Discovery System," in *ICDE*, 2018, pp. 1001–1012.

[11] A. D. Sarma, L. Fang, N. Gupta, A. Y. Halevy, H. Lee, F. Wu, R. Xin, and C. Yu, "Finding Related Tables," in *SIGMOD*, 2012, pp. 817–828.

[12] Y. Zhang and Z. G. Ives, "Finding Related Tables in Data Lakes for Interactive Data Science," in *SIGMOD*, 2020, pp. 1951–1966.

[13] L. Wang, J. P. Near, N. Somani, P. Gao, A. Low, D. Dao, and D. Song, "Data Capsule: A New Paradigm for Automatic Compliance with Data Privacy Regulations," in *Poly@VLDB*, 2019, pp. 3–23.

[14] T. F. J. Pasquier, D. M. Eyers, and M. I. Seltzer, "From Here to Provtopia," in *Poly@VLDB*, 2019, pp. 54–67.

[15] "Chinook Database," https://github.com/lerocha/chinook-database.

[16] "MySQL Employees Sample Database," https://dev.mysql.com/doc/employee/en/.

[17] "TPC-H Benchmark," http://tpc.org/tpc_documents_current_versions/pdf/tpc-h_v2.18.0.pdf.