Consent Visualisation

Christof Bless, Lukas Dötlinger, Michael Kaltschmid & Markus Reiter

December 30, 2020

Abstract

Data privacy scandals in the recent past have left many people around the world insecure about the implications of data sharing in everyday life. At the same time advances in machine learning and data science have given us the possibility to enhance the safety and the quality of life of many people. To collect data from individuals nowadays companies have to rely on mutual consent and trust. New regulations such as the GDPR make sure that consumers are not exploited or taken advantage of. From a legal standpoint it is necessary to inform customers to a full extent about what information is collected from them. In this paper we present a new visualisation approach to keep people informed about the activities linked to their data sharing agreements. We introduce a user-centred application with a transparent visualisation aiming to give users a better understanding of the data sharing processes in the background of their consent agreements. (Does this lead to more legal awareness and trust? evaluation results summary)

1 Introduction

Since it has come into effect in May 2018, the General Data Protection Regulation (GDPR) has had a big impact on the way companies deal with personal data. The GDPR brought awareness of data ownership and protection of privacy to a new level for both companies and consumers. There is an interest to handle data generated by activities of real people in a safe and consensual way. For many companies adhering to the regulations is not only important to prevent costly legal affairs but also to keep up a good reputation with their customers. At the same time, data owners are more

and more concerned about their rights. Many value their privacy highly and want to control their digital footprint on their own.

The GDPR requires consent between a data owner and a data controller if the data controller wants to process any kind of information related to the data owner (Art. 6 (1a)). According to the GDPR, consent has to be (i) freely given, (ii) specific, (iii) informed and (iv) unambiguous (Rec. 32).

This paper focuses on establishing informed consent; that means a data owner is completely aware of extent, target and content of his data sharing activities. To achieve this we look into ways to improve transparency with the data owner through informative visualisations. We believe that visualisations help end users understand how their data is being shared, with less effort than by reading the agreement text which is the prevalent status quo.

When it comes to data visualisation to end users, the most important aspect is the simplification of complex data. Data for applications is generally stored in some sort of database system which is hard to understand for non-expert users. Data visualisation is a technique of mapping complex data to visual elements, thus making it easier to understand relations within the dataset.

This paper presents an approach to visualise data in the domain of vehicle sensor data sharing. The work is part of the CampaNeo project, whose goal it is to create a system to collect and distribute sensor data generated by modern vehicles. Data is requested via specific campaigns, which must be approved by the data owner. Following the GDPR, campaigns must state exactly what the purpose of their data collection is and what type of processing they plan to do on it. Ideally, companies or research organisations behind the campaigns contribute to the development of better technologies and services in the realm of mobility and transport, which in turn enhance the user's experience with the vehicle.

For example, GPS location and speed data from a big number of cars can help optimise traffic flow management, which leads to less congested roads and time savings for drivers.

Semantic knowledge graphs are a state-of-the-art solution for building versatile, explainable and machine-readable data storage solutions. Knowledge graphs are the underlying technology used in the CampaNeo project. Since semantic data from a *triplestore* mainly describes objects and their relations, it is complex to visualise. One could just display the whole database

as a graph, with all objects and their relations, however, this would result in a very large visualisation which is too cluttered and therefore confusing to users.

The idea is to visualise the flow of data from a user's car to third party companies on small to medium displays (e.g. tablet, smartphone or the car's built-in infotainment system). The user can get an overview on what data they are sharing with institutions like governmental agencies, universities or data processing companies who collect high amounts of data with the intent to solve problems around mobility and transport. The solutions generated from the data should in turn benefit the data owner in some way. The visualisation focuses on highlighting the data streams to the user who should get information about the type of data that is shared, at what intervals it is sent out and who the receiving party is.

This paper is structured as follows: Section 1 presents an introduction to the field, while section 2 presents related work. Section 3 defines the main research questions. The methodology for deriving the first prototype can be found in section 4. Section 5 contains architecture details of the implementation. Section 6 presents the testing methodology, the results of which can be found in Section 7. Conclusions are made in section 8.

2 Related Work

The main problem we want to solve with our work is that of fully transparent visualisation of data sharing activities. We will achieve this by building a tool that enables users to monitor and control the distribution of their data.

In recent years there have been several attempts to design applications that implement such visualisations. Raschke et al. [1] built a general dashboard to visualise data sharing activities and give consent approval and withdrawal mechanisms. The dashboard is a single page application with a vertical timeline listing the different types of actions. Among these actions are sharing a first name or a picture as well as information about location and search history. Further, the application offers information about processing context and type of the data in question.

The authors evaluated the tool with a set of tasks for participants to complete using the dashboard application. However, Raschke et al. [1] only tested with expert users most of which were also their colleagues. The main

takeaways were that data type categories need to be refined more to be understandable. Generally, even the expert users found it hard to answer questions about their data privacy based on the information available from the dashboard.

Another implementation of a consent and data privacy visualisation interface is the Consent Request (CoRe) user interface (UI) from Drozd and Kirrane [2]. The idea of the authors was to develop a UI which shows the implications of accepting a consent agreement. The hypothetical scenario would be consenting to the use of individual functionalities of a fitness tracker. For example, a user wants to have the route of a morning jog displayed on their app. To unlock the functionality, one has to accept some data processing by the data controller, which is the manufacturer of the tracker. The CoRe UI [2] will then display a graph that shows what data is sent out, where it will be stored, the type of processing that is done on it and which third party companies it will be shared with.

To validate their design choices, Drozd and Kirrane evaluated two slightly different CoRe UI prototypes by giving tasks to participants from different age groups and recording their actions. The first prototype that was tested was a bit more elaborate and had more features than the second one which was a simplified version. For the first test 27 participants were asked to perform the specified tasks. 74% of contestants seemed to be "very confused" and more than half of them found the UI "too complex" and "hard to use". The second, simplified UI was accepted better with an even larger test group of 74 people. Still many would describe the layout as being "confusing", "annoying" and "complex". The majority of participants claimed to be unsatisfied or neutral with the application.

The tolerance for cognitive overload through too much information and display of complex relations is low for most people. Especially when dealing with legal conditions of data privacy. Consequently, a good start for any attempt to create a transparent visualisation of data sharing processes is to simplify the user interface to only include the most essential components, which will be determined in Section 4.

3 Research Question

The CampaNeo project is highly dependent on requesting and receiving informed consent for sensor data sharing. The more people agree to send usage data from their cars, the more value the statistical analysis will generate. We formulate two hypotheses which are the basic assumptions of this work:

- People are more willing to share their data if they are fully informed on what exactly they are sharing, when they are sharing it and with whom exactly they are sharing it.
- Data visualisations improve comprehension of consent.

It is debatable whether current consent gathering methods really make it absolutely clear to data subjects what happens in the background after they gave their consent. According to [3] people tend to agree to most consent requests they are confronted with. Reading through all the agreement specifications is time-consuming. Such documents are often written in a complex language typical for legal documents. Due to that, most people who give their consent to data sharing agreements do so without understanding many details of the contract. Bechman [4] defines this as a "culture of blind consent". To conclude, in most cases having one's consent, even informed, is not equivalent to having awareness.

To change this, we build an application to enable data owners to give informed consent and gain legal awareness in the process. This should be achieved by making the dataflow completely transparent through a visualisation.

The research questions that are addressed in the design process are:

1. What aspects of the data should be visualised?
2. How can the data be visualised in order to improve comprehension?

4 Prototype

For the first prototype of our application the idea was to learn from previously done work and reuse what worked best. Additionally we followed general design principles like Gestalt laws [5] of grouping.

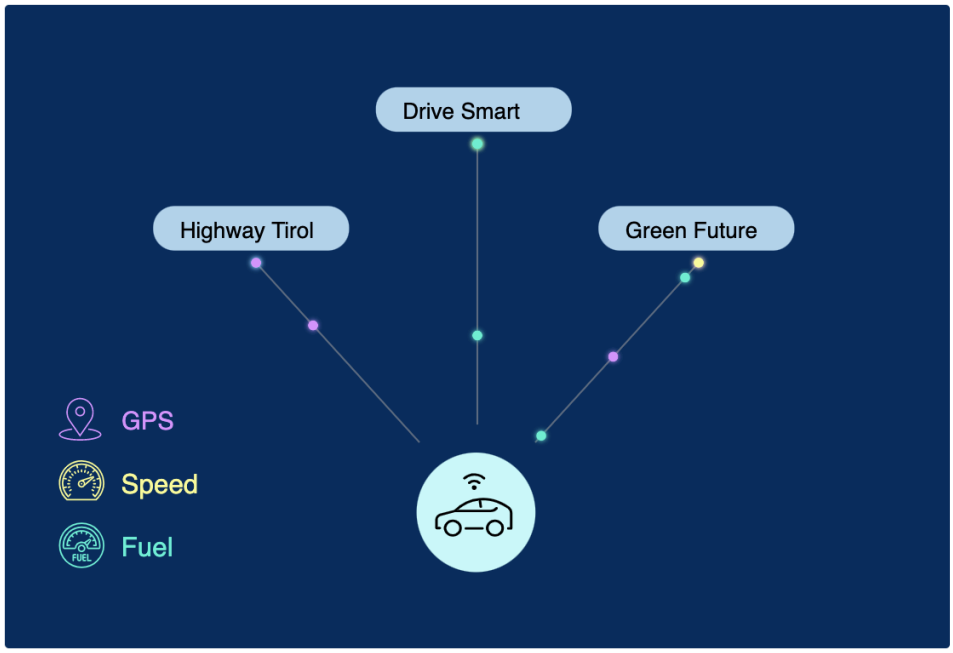

Figure 1: The first prototype of the general overview

A typical visualisation technique to make complex relationships clearly understandable is a graph layout. To show data flow from a data subject to one or more data processors, a graph should work very well. Nodes represent data subject and data processors, and the links between them show that they relate with each other in such a way that data is shared between them. To make this even more clear, actual or imaginary data packets can be visualised as moving particles between the nodes. This feature makes it possible to give the visualisation a sense of directional flow. These design choices are also backed by Gestalt laws such as the law of similarity which states that visual objects resembling each other are perceived as belonging to the same group. Further, the law of common fate states that objects moving into the same direction are recognised as grouped such as the data stream in the visualisation.

Since the whole application is built around giving users a sense of control over their sharing activities, the user as a data subject stands in the centre of the visualisation with the connected data processors distributed around in a circle, akin to a star network [6] with the centre point being the user. Data

processors, called campaigns in the CampaNeo environment, are represented by a rounded rectangle and the corresponding campaign name. The round particles represent data packets flowing between user and campaign. They are color-coded to give additional information on the type of data that is sent, e.g. fuel consumption, speed or the GPS location of the car. The meaning of the different colours are encoded in a legend on the bottom left side of the screen. Here, words are paired with unambiguous symbols to make the legend easier and faster to read and understand. On the whole, the visualisation is designed to enable users to see at one glance what kind of data they are sharing and at what rate.

With that we have now completely satisfied the defined needs of our application: The user can now find out with whom he shares what data and approximately at what rate. However, in case the user wants to get more in-depth information, it is possible to click on a specific data flow in the visualisation in order to get a more detailed view of the data stream. Here we rely on a time series visualisation and give additional information about the sensor that retrieved the data and the companies that the data processor shares the data with.

Clarity is of utmost importance, which is why the visualisation always starts with the summary view to avoid confusing the user with too much information at once. The more detailed information is served only upon interacting with the visualisation. Displaying everything on screen at the same time would overload the initial rendering far too much.

5 Implementation

5.1 Technology Stack

The goal of this project in terms of the implementation was to build a web application that is easily embeddable into a mobile application. For this reason we decided on going for a lean front end without too many unnecessary dependencies to keep it responsive. Among the key dependencies that are necessary is D3.js, a JavaScript library for manipulating documents based on data [7]. D3 fits the requirements of simple interoperability between data and visualisation by manipulating the DOM perfectly.

Naturally, a front end for the purpose of this application is not of much

use without data and therefore a database is required. Given that the visualisation is built around a semantic knowledge graph, a graph database is a necessity.

A great fit in this regard is RDF4J. RDF4J, a native triple store, offers the ability to query the knowledge graph using SPARQL. The database has also support for all mainstream RDF file formats [8]. RDF4J is an open-source project under the umbrella of the Eclipse Foundation which indicates it is trusted by a big community.

Another option is GraphDB [9], a proprietary graph database developed by Ontotext. It offers multiple APIs for querying the database, including RDF4J and SPARQL, among other things. Ontotext offers a free version of GraphDB, however when we evaluated it for development purposes, it was not really suitable for a fully automatic development experience given that the free version needs to be downloaded manually while Ontotext offers pre-built Docker images for their Standard and Enterprise offerings.

Since the CampaNeo project already had a pre-existing GraphDB database, we still decided on using GraphDB. The main focus of this project is the visualisation, so as long as we could retrieve the data from the database this did not pose a problem for our purposes. The multiple APIs offered by GraphDB allowed us to use a vendor-agnostic JavaScript framework for creating SPARQL queries.

A simple use case would be a user starting the application, which initializes the D3 frontend web application, which in turn queries data from the database to visualize the consent page as shown in figure 1.

5.2 Architecture

Shown in figure 2 is an overview of the technical architecture for the implementation of the data visualization.

On the left side of figure 2 we see the application user, i.e. the data owner. The data owner interacts only with the front end part of the application, which we call the “Visualisation Interface”.

On first load or whenever the user interacts with the interface in such a way that the underlying data needs to be updated, we send a SPARQL query to the GraphDB database. Once the result of this query is retrieved, it is passed to the “Visualisation Module”. This module processes the data

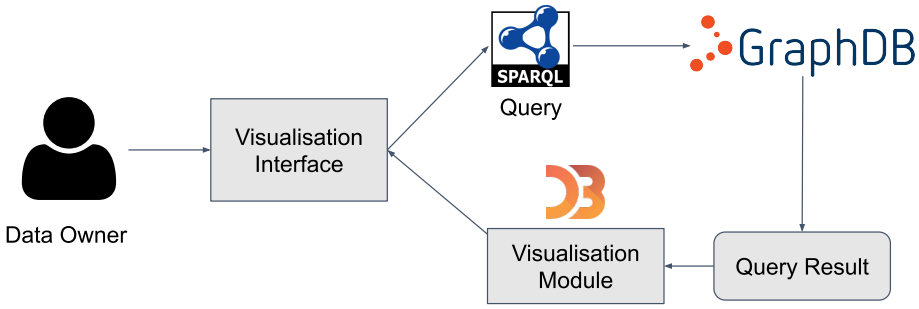

Figure 2: Technical Architecture

depending on whether it will be displayed in the overview visualisation or the detailed time series visualisation and produces the corresponding visualisation using `D3.js` accordingly.

Depending on the query, we get a lot of data in return, as the database will return a value for each data-package, that was sent. The so called "Visualisation Module" groups those together, creating a single visible node for each category of data. These categories are predefined in the ontology and represent thing like coordinates, longitude and latitude, or fuel consumption. The module then creates connections between the data packages and the third party companies receiving them.

After the visualization is created for the received data, the "Visualisation Module" updates the currently displayed graph.

...

6 Evaluation

With the implemented prototype the following questions can be verified during a testing phase with real users:

1. Can we raise legal awareness of the data sharing process through transparent visualisation?

2. Does the ability to constantly monitor one's data sharing activities make the data processor more trustworthy to data subjects?

There are several possibilities how the developed application could be integrated into the ecosystem of the CampaNeo project. One option would be to install the campaign overview natively into the infotainment system of modern cars. In this way a user would be the driver of the car who checks the status of data sharing campaigns before or after the drive through the on-board display of their car.

Another equally valid option is to provide an application available on the web or for mobile devices which lets the owner of a car connect to their car and check information about campaign data sharing remotely and independently of accessing their car.

To validate the posed questions, it is not essential for the test to take place in an actual car and simulated driving environment, since the application should not be used while driving anyway. For these first tests we provide a web application which features the functionality described in the previous sections. The tests can be performed on desktop computers, laptops or other mobile devices. The test subject needs a thorough introduction into the scenario since the use case of the application is very specific and it is very important that the situation is understood correctly.

A good understanding is important for the test subject to tackle the tasks that comprise the first part of the test. After the introduction and before presenting the visualisation to the subject the answer to the following question is recorded: "Would you consider sharing sensor data of your car with chosen campaigns?"

We then reveal the visualisation to the tester. The subject is presented with a series of simple to slightly complex tasks that revolve around answering questions about the privacy and extent of the data sharing activities. During the evaluation the test subject should be observed in real time through screen sharing. Using the "think aloud" method the test subject then informs the tester regularly about their mental process, for example what they want to achieve next and where they expect to find a specific information in the application.

After the task solving period, the subjects are asked to fill out a questionnaire in which they rate their experience and explain problems or give suggestions for improvements.

The goal of the questions is to find out whether the subject prefers the clear transparent view of all the data exchange happening or if it is

more confusing and feels more invasive than being presented with a written contract which is the prevailing method.

In addition we want to find out if the visualisation makes subjects more aware of their rights and admissions within the agreement.

The question “In the scenario where such a visualisation is available to you at all times, would you consider sharing sensor data of your car with chosen campaigns?” will show whether the approach leads to trust gains for the data processor.

7 Results

The results of the evaluation.

8 Conclusion

conclusion

References

[1] Philip Raschke et al. “Designing a GDPR-Compliant and Usable Privacy Dashboard”. In: June 2018, pp. 221–236.

[2] Olha Drozd and Sabrina Kirrane. “Privacy CURE: Consent Comprehension Made Easy”. In: *35th International Conference on ICT Systems Security and Privacy Protection ? IFIP SEC 2020*. Sept. 2020, pp. 1–14. URL: https://epub.wu.ac.at/7546/.

[3] F. Zuiderveen Borgesius. “Informed Consent: We Can Do Better to Defend Privacy”. In: *35th International Conference on ICT Systems Security and Privacy Protection ? IFIP SEC 2020*. Sept. 2015, pp. 103–107. URL: https://epub.wu.ac.at/7546/.

[4] Anja Bechmann. “Non-Informed Consent Cultures: Privacy Policies and App Contracts on Facebook”. In: *Journal of Media Business Studies* (2014), pp. 21–38. URL: https://doi.org/10.1080/16522354.2014.11073574.

[5] Wikipedia contributors. *Principles of grouping*. 2020. URL: https://en.wikipedia.org/w/index.php?title=Principles_of_grouping&oldid=996276406 (visited on 12/29/2020).

[6] Wikipedia contributors. *Star network*. 2020. URL: https://en.wikipedia.org/w/index.php?title=Star_network&oldid=996032407 (visited on 12/29/2020).

[7] *D3 Data-Driven Documents*. URL: https://d3js.org (visited on 12/26/2020).

[8] *The Eclipse RDF4J Framework*. URL: https://rdf4j.org/about (visited on 12/26/2020).

[9] *GraphDB*. URL: https://www.ontotext.com/products/graphdb/ (visited on 12/29/2020).