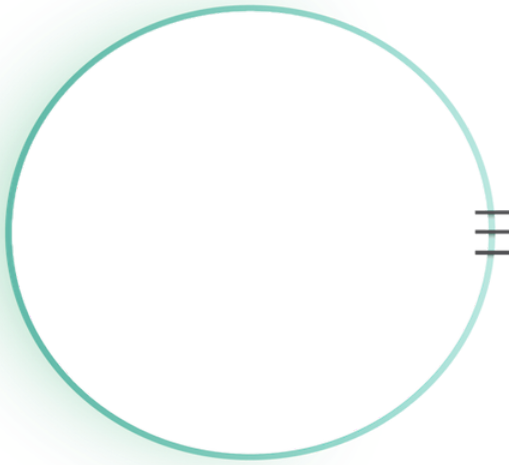

The Lens Data Subscription Network

Cody W. Eilar, Nathan Stark, Mark Chavez

Abstract

In this paper, we present the Lens Data Subscription Network. Lens provides an alternative approach for collecting data via HTML forms on the web by implementing a subscription model to data that individuals control and enterprises or other individuals view. By contrast, traditional web sites, leveraging Web 2.0 technology, typically amass data from their users and store it on the site's infrastructure, including cloud resources. Refreshing data in this manner is challenging, error-prone, is often exploitative to their customers and is generally not in compliance with data protection regulations. It is clear that although data is quite easy to copy and distribute in traditional systems, there are several pitfalls. Data consistency, data ownership, and regulations pose several challenging business and technological conundrums that exist in these systems.

The Lens data subscription network solves these issues by enabling individuals and enterprises to collaborate on data by 1) ensuring only the individual can control who has access to data via cryptography, 2) providing a reliable mechanism for businesses to access their customer's data via subscription called a "Lens" 3) enabling individuals to audit all their data transactions, and 4) furnishing a suite of algorithms that help reduce tautological data and keep the individuals data storage container pristine, valuable, searchable and maintainable--in essence, a single source of truth for all their data.

Introduction

The internet is invariably evolving based on improvements in hardware, software, and business processes. This evolution often involves a shift in responsibility for processing, data storage, and physical hardware ownership. The first computers were massive and required that all the processing, data storage, and hardware resides together. As personal computers began to have mass-market adoption, it was no longer necessary to do all computations in a specific location, but each individual could do computation themselves. In this sense, both hardware and software became decentralized. However, as businesses began to grow, and maintaining licenses to software and maintaining hardware became more costly to IT departments, software as a service pushed computation back to a central location again. Cloud computing enables businesses to subscribe to software and not own it. The software subscription model, also known as software as a service, or SaaS, enabled businesses to focus on what they do best and not on services that hindered business and could ultimately lead to a small company's demise.

A new shift is on the horizon, and it has to do with the management of customer's data. Much like the era before SaaS, businesses are starting to realize that it is costly to keep customer data themselves. Estimates of the total cost of bad data are around $3.1 trillion a year, and that number only includes the United States. Much like software and hardware maintenance before SaaS, costs will rise until business leaders and engineers take action to form a new solution.

The cost of data ownership for businesses shows no signs of falling. Regulation and policy may be adding to the list of burdens that businesses must address. Already, laws like the General Data Protection Regulation (GDPR) are forcing businesses to rethink their methods for handling consumer data. With the California Consumer Privacy Act (CCPA) on the horizon, it is evident that the issue of data governance is not just a European problem, but is now a global one. As a result, there is a significant amount of space for innovation, both technologically and in business.

With these historical landmarks in mind, the future of businesses owning an individual's data is behind us. With rising data costs and more regulation, the solution to the success of a business is whether or not they subscribe to customer data. This technological and business shift is the next evolution of the internet. Just as software ownership and maintenance is too expensive for the enterprise, so is data ownership. SaaS moved responsibility of software to companies that are good at it and the Lens Data Subscription Network moves data to those who know the truth about data: the individual.

Overview

At its core, Lens Subscription Network facilitates data connections between individuals and enterprises. These connections are optimized to be efficient, data protection compliant, secure, and cost-effective. Businesses connect directly to a customer's information via a decentralized data link called a "Lens." Individuals possess the power to modify or delete the Lens and any underlying information at will, giving them full control of information often collected through lead forms on HTML pages. Additionally, individuals have the choice to host Lenses themselves on their hardware should they desire.

The network itself consists of several underlying applications designed to be used either by individuals or enterprises to facilitate data collaboration. The first is the Lens Dashboard, explicitly designed to be used by individuals. The Lens Dashboard is a web application that enables individuals to monitor, edit, and organize data that is requested by enterprises. It can be thought of as the oAuth2 of the dataworld: users can safely and securely share data with third parties. The second is the Lens Management System or LMS for enterprises. LMS is an off-the-shelf solution that empowers business owners to subscribe, manage, and search Lens subscriptions. Together, these applications work not just to create a data compliance solution for enterprises, but also an effective way for individuals to manage a single source of data.

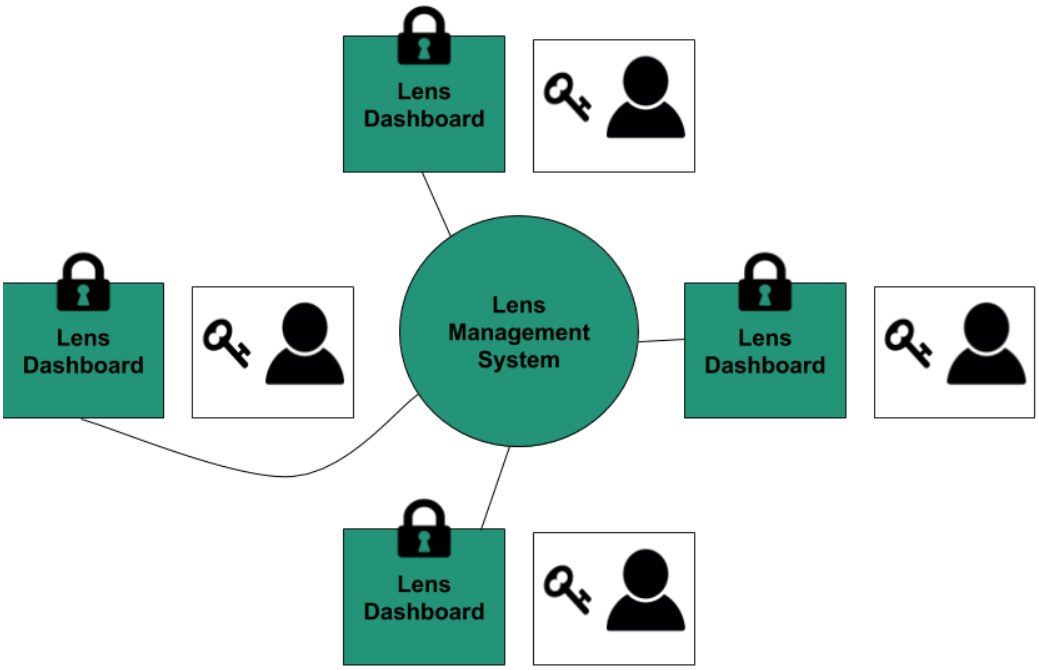

Figure 1. Illustrates that each individual has control of all data and subscriptions in their Dashboard with a private key. The Lens Management system is given a subscription to a subset of data from the individuals dashboard.

In the following sections, we describe in detail the implementation details of the Lens Subscription Network. In particular, we cover the details of authentication/identity, data governance, data organization, security, and other architectural details.

Lens Dashboard

The Lens Dashboard is a decentralized web application built on top of Blockstack. From the Dashboard, individuals view, modify, and organize both their data and subscriptions to data called "Lenses." By giving individuals one place to view all the companies that have access to their data, they can make informed decisions about data management.

The design goals of the Lens Dashboard are:

1. **Give users full control of data.** Only the individual can modify and view data that they have shared out unless an explicit subscription is given out. The user should be able to choose where they store their data physically.
2. **Provide users with transparency.** The Dashboard should give users a full view of who has subscriptions to what data so they can make informed decisions about data management.
3. **Reduction in redundant data.** Users should not have to repeat themselves for standard datasets that many enterprises ultimately need.

4. **Supply users a self-sovereign identity**. MyLens, nor any other institution should have the ability to revoke this identity.
5. **Privacy and security**. All data in the user's data locker should be private by default, and best practices in security applied to store and access this data.
6. **GDPR and CCPA compliance.** MyLens has no interest in owning or seeing any data that is produced by customers. To ask for customer data, we use Lenses. We subscribe to our customer's data via the Lens. Should they no longer want us to see that information, they revoke access.

With these design goals in mind, MyLens composes the Dashboard such that it is a differentiating piece of software from both traditional Web 2.0 sites and completely decentralized applications.

Full Control

Users must be able to choose where they host, who can view, and have the ability to edit their data to have control. To control where their data resides, we chose to use a technology called Gaia storage, which is a decentralized high-performance storage system. Dashboard users can take advantage of giving out access and edit their data through our Lens Storage SDK, which is a decentralized capability system built on top of Gaia.

At the core of the Lens storage scheme is Gaia. Gaia provides users with a secure and private data locker that they own and control. Ownership and control are possible by two design philosophies used in Gaia. First, Gaia is built to use existing cloud storage technologies such as Amazon S3, Microsoft Azure, and Google Cloud Storage. Beyond cloud providers, local disk storage on any computer connected to the internet is possible as means for storage too. Applications that wish to write to the user's Gaia storage need only to look up the user's Blockstack ID on the Stacks blockchain to find where the user has designated their Gaia storage to reside.

The second design philosophy is to ensure that the user's data remains private and tamper-resistant regardless of what cloud storage or disk they choose to host their data. By encrypting and signing data by default, Gaia fulfills these requirements. Data cannot be modified by an unauthorized entity because each piece is verifiable via the public keys that reside on the individual's entry on the Stacks Blockchain. As a result, any attempt at altering or tampering with the individual's data is easily detectable via conventional asymmetric cryptography.

By design, Gaia does not write every piece of data to the blockchain as is done by Ethereum in their "world computer" paradigm. By limiting the number of interactions with the blockchain, reading, and writing data to Gaia is exceptionally efficient and is comparable to that of traditional cloud storage providers with a small amount of overhead to query the blockchain to fetch the address. On the first issuance of the query, most modern web browsers cache the result, making future calls unnecessary.

Lens leverages the Blockstack technology stack to get the attributes mentioned above for data storage. By making sure users own and control their data at a fundamental level, Lens

can build the data subscription network that is private, secure, and data regulation-compliant by default. Beyond compliance, enterprises can rest assured that because the network is explicitly designed around individuals owning and controlling their information, that less liability resides with them.

The Lens - A subscription to data

At the center of the Lens Data Subscription Network is the Lens. The Lens is a linked encrypted individually for a subscriber which grants read access to specific data fields that an individual owns and controls. In programming terms, a Lens is analogous to a "pointer-to-a-pointer," that is to say it is a link that contains a list of additional links to data. This link enables a few key features: 1) End-to-end encryption, guaranteeing that data is kept confidential between data producer and consumer, 2) Access control, endowing individuals with the power to add and subtract who can see what. 3) Data consistency, reducing the need to maintain multiple datasets for different subscribers.

The most elementary unit of the Lens is a key, value, and type triplet which is stored and encrypted individually in Gaia. The key is a string that describes the name of the data. For instance, a key could be something like "FirstName" or "FamilyName." The value can be any fundamental data type such as a boolean, string, or buffer. Finally, the type is a string the signifies how the data value should be interpreted, such as an email address, or phone number. Types are also unique because they can additionally provide format validation for well-understood fields like an email address. Enterprises requesting data can take advantage of this in the form of not having to write their regular expressions to validate input fields on a form.

Lens security is possible by encrypting and storing symmetric keys in the data encryption index. By default, each triplet in a Lens is symmetrically encrypted using AES 256 CBC. The symmetric encryption key lives in the data encryption index, which is only accessible via an individual's app-specific private key. This key is not the same as the master key derived from the mnemonic on registration, but rather the application-specific key exists from the derivation of the master key and the application's HTTP origin. In the case of the Dashboard, the cryptographic key derivation function would look something like `deriveKey (MasterKey, https://dashboard.mylens.io)`. As a result, the data triplets are only accessible if the location of the data and the symmetric key is known.

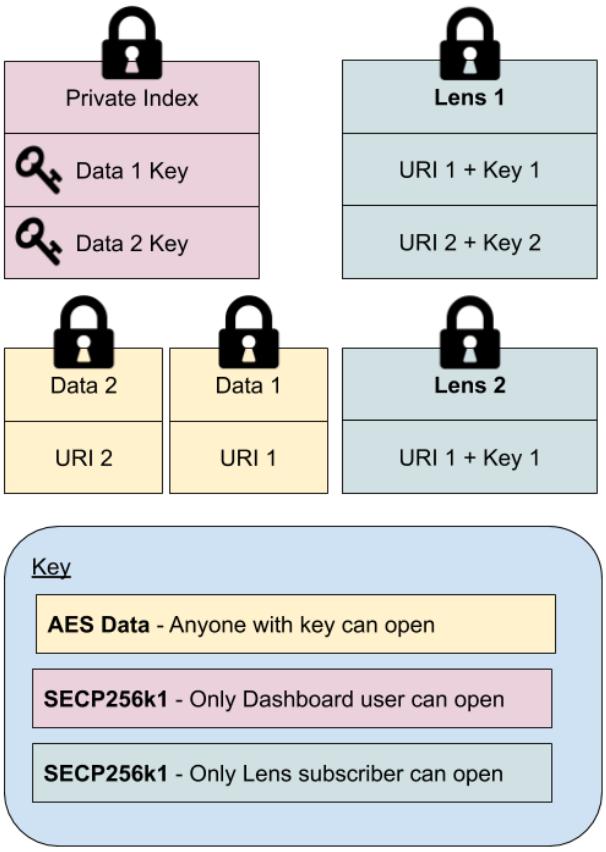

Figure 2. Illustrates how data is organized and encrypted. The Lens itself is nothing more than a piece of data encrypted specifically for the subscriber that contains the URI to the data, and the symmetric key to decrypt the data.

The Lens is designed to handle any number of data triplets securely. The process of creating a Lens involves a few steps. First, data triplet selection takes place where an individual can select, from the Dashboard, what data they wish to include in a Lens. Second, the individual chooses the recipient. The software either encrypts the Lens for a known public key or creates a "local" user and encrypts the Lens for that user. Having both of these abilities means that an individual or a business does not necessarily need a public key to access a Lens. It does mean, however, that this individual or business must keep track of the private key that the Lens Dashboard User created for them. As with most cryptographic solutions, if the private key is lost, it is not recoverable, and the Lens Dashboard User must generate a new public/private key pair and a new Lens if the subscriber were to lose their private key. Third, once the encryption process finishes, the Dashboard writes a Lens to the User's Gaia storage and returns a link to that file and the private/public key pair if the Lens is for an unknown user. This link is what subscribers can then use to fetch data from their customers anytime.

Given the link from the data owner, a subscriber reverses the encryption process for a Lens to fetch the data. The reverse process involves decrypting the Lens to fetch the links and symmetric keys to the data triplets. Given the symmetric keys and the links, the subscriber first

fetches the data given the link and then decrypts the triplet using the key. They can now arrange the data in whatever form they wish.

Once a subscriber decrypts the data; there is no way to guarantee that it remains uncopied and distributed. However, the benefits from subscribing to it outweigh the temptation to copy it. Whenever an individual changes data, all Lenses that have a subscription to that data are updated. Therefore if an individual chooses to change a data field with a key, "foo" to a key, "bar," all subscribing parties receive this modification. Furthermore, if an individual wishes to invoke their right "to be forgotten" via GDPR, the enterprise needs only to fetch the latest version of their Lens. The Lens, in this case, would contain no data, but a brief message stating that the Lens has been "revoked." In this vein, enterprises can leverage a Lens to augment their databases with reliable data, frequently updated by the individual, and profit from being GDPR and CCPA compliant.

Reducing Tautological Data

Walled gardens, the ease of copying data, and nuanced data have led to enormous amounts of redundant data. Standards, such as the ones available at schema.org, attempt to create a vocabulary for commonly used datasets; however, it is often easier for developers to create custom data schemes for business-specific needs. This lack of standard usage leads to many different representations of the same dataset. For example, in the English language, it is possible to ask for someone's "first name," "given name," or "name." These different keys for representing someone's first name often point to the same value. Even if developers could agree on a standard, there is still the trouble of devising a schema for each language. MyLens takes a laissez-faire approach to data standardization and instead provides a fungible set of rules to map data between subscriber and producer.

When an individual receives a Lens request, the Dashboard presents them with a UI that allows them to create this mapping themselves or choose a value that the Dashboard "guesses" is correct. This flexibility allows data requesters to ask for whatever key name they wish, but ultimately the end-user is responsible for populating that value. A requestor may ask for a "first name," but the user may already have a similar item in their Dashboard named "given name." The user has the option to select the existing, create new, or modify their data for that field. This concept permits both data consumers and producers to map data for their unique needs instead of forcing any particular, and possibly incomplete datasets.

Transparency

In Web 2.0, individuals must go to every site they have signed up for in order to gain perspective on their data footprint. To make the situation worse, prior to GDPR and CCPA, individuals often did not have the leverage to demand this information from these companies, making them powerless in their own data maintenance. Individuals have no one place to update their information and must update every bit of information they have one site at a time. They

have no 360 degree view of their information. Today, regulations are empowering individuals to take an active role in their data protection, but there is still the task of understanding where the data lives. The Dashboard gives user's this transparency.

The Dashboard contains all data and subscriptions that users create and give out. The idea behind this single location is so that individuals can view not only what data they have created, but they can see who exactly is subscribing to it. This transparency is enormously influential because individuals can now track which companies know their email address, phone number, or mailing address, to name a few available digital datasets.

Every time a subscription is given out, the Dashboard writes to the user's local data-locker subscription index. This index keeps track of all subscriptions and to what data it references. This index offers the individual a complete list of all active Lenses given out. In addition to updating the user's local index, the Dashboard signals MyLens servers that a Dashboard user created a new Lens. The Lens Management System, which we cover in a later section, needs this signal to update its customers. At a minimum, Lens knows that user A created a Lens for Enterprise B, but there is no knowledge about the content within.

Self-Sovereign Identity

Unlike notable Web 2.0 companies, Lens, nor any other third party owns the Dashboard user's identity. In contrast with the Solid and Semantic Web Solution, there is no need for the user to deploy any centralized infrastructure. At a minimum, a user needs to have a browser and a valid email address. To recover keys, users need an encrypted mnemonic, and the easiest way to distribute that is through email. The intention is to make sure that the individual firmly controls the identity.

Developers of the future web still contest the exact meaning of "self-sovereign" identity, but several themes tend to emerge. First and foremost, an identity is self-sovereign if the owner of that identity is the one who controls it. In other words, the user must be the sole administrator of that identity. Users realize this tenet on blockchain technology. Because no one entity owns and controls decentralized blockchains, advertising public keys as an identity on the blockchain grants this strong ownership.

MyLens achieves this goal by leveraging technology from Blockstack called the Blockstack Naming Service or BNS. BNS works by leveraging an existing blockchain like Bitcoin or Namecoin to write "names" or "identities" to the blockchain. The exact implementation details are more complicated than what we present here, but complete implementation details are not necessary to understand the underlying philosophy. In a nutshell, the design of BNS enables three core features: 1) names are globally unique, 2) names are human meaningful, and 3) individuals actively own their identity. An example of this identity might look like “mylens.id.blockstack.” Given these three features, Lens can provide users a self-sovereign identity that is owned by neither Blockstack nor MyLens. Furthermore, the user is not responsible for standing up infrastructure to host their identity because it exists on a peer-to-peer network.

Privacy, Security and Data Compliance

The spirit of the laws passed in the European Union, known as GDPR, is to provide their citizens with digital rights regarding personal data. Individuals must give explicit consent to businesses to process their personal data. Businesses must begin to protect data "by design and by default" to be genuinely GDPR compliant. MyLens takes this approach by using modern cryptography and Web 3.0 technology to enable this.

As mentioned in previous sections, both asymmetric and symmetric key cryptography keep Lens Dashboard users' data secure and private. Notably, the AES 256 symmetric algorithm protects all data triplets, and the elliptic curve secp256k1 protects both the individual's data indexes and generated Lenses. With these technologies in place, individuals can rest assured that their data is not visible, even to MyLens, and enterprises can take advantage of GDPR compliance by design.

Lens Management System

The Lens Management System or LMS, helps enterprises organize and derive valuable information from their customer's data via the Lens. LMS provides several key features, such as data querying, syncing/exporting to CRM systems such as Marketo and Salesforce, and the Lens Request Builder. Together, these tools enable businesses to subscribe to data privately and securely.

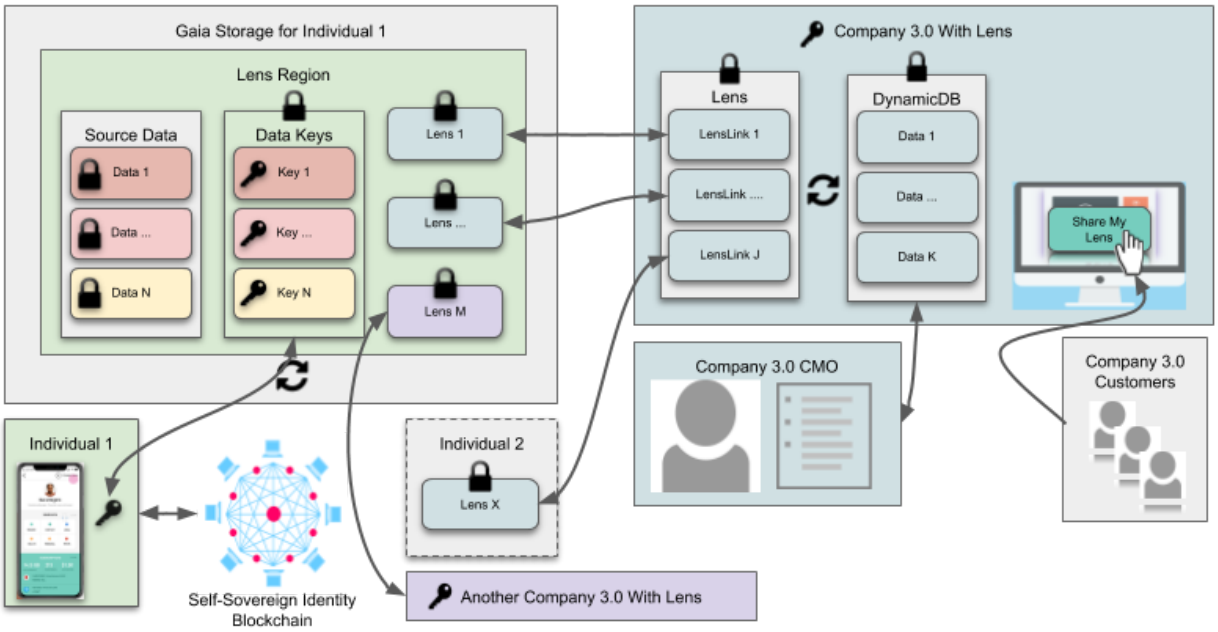

Figure 3. Illustrates how CMOs can leverage the entire Len ecosystem with the Lens Management System. Data updates start with the individual and make their way to the CMO's database. The CMO now has direct access to quality customer data.

Businesses subscribe to their customer's data by placing a "Lens Request" button on their site. When a customer clicks this button, the page redirects them to the Lens Dashboard. Existing Lens users will either sign in with their credentials or create a new Lens Dashboard account. At this point, the customer fulfills the request by selecting, creating, or modifying data in their private data locker. When they finish fulfilling the request, the Dashboard redirects them back to the initiating page. The business now has a subscription to this customer's single-source-of-truth. An enterprise can write their own tool for querying their lenses or use the existing LMS tool.

LMS is a traditional Web 2.0 system. LMS stores Lens data in a centralized database that is accessible by MyLens. Businesses access the system via an API key, and they can either choose to store their secret key, used for unlocking Lenses, in the LMS system for convenience or store it themselves. MyLens implements best practice security on all centralized services, so it is advantageous to store the key within LMS if security is a concern. Indexing data quickly is an earnest concern for many businesses, and unfortunately, opening a Lens for hundreds of thousands of users can be quite slow. Fortunately, our centralized Lens indexing system ameliorates the speed without compromising security.

Conclusion

In this paper, we outlined many of the details of the Lens Data Subscription Network. We demonstrated that it is possible with a combination of decentralization, encryption, and standard web protocols that it is feasible to create a system that benefits both businesses and individuals in terms of data collaboration, data compliance, and data consistency. Additionally, we explored many of the technologies that have preceded and coexist with Lens that offer similar promises but with different tradeoffs. Unlike many other technologies that we presented in this paper, MyLens technology focuses specifically on bringing more value to enterprises via data subscription while preserving the individual's rights to personal data.