

Enigma Genetics: A Self-Sovereign Genomic Data Platform for Dynamic Consent Enforcement and Longitudinal Data Provenance

Technical Whitepaper

Kenneth J. Clark, Engineer

Enigma Genetics (Rochester, NY, USA)

8 February 2026

Version 1.0

Abstract

Genomic and epigenomic data provide unprecedented insight into human biology and disease but raise profound challenges in privacy, control, and data stewardship. Unlike conventional health records, even a small set of genetic variants can uniquely identify an individual, rendering anonymous data sharing impractical [1,2]. Existing governance frameworks rely on static, one-time consent and custodial data ownership, leaving participants with little recourse once their data leaves their hands. Here we introduce **Enigma Genetics** – a self-sovereign platform that operationalizes **dynamic consent**, biometric identity anchoring, fine-grained authorization, and immutable auditability for genomic data. Drawing on prior work in dynamic consent models, genomic re-identification demonstrations, distributed provenance frameworks, and modern data-protection regulations (e.g. GDPR, HIPAA), we propose a system architecture comprising four core services: (1) an **Identity Anchor** binding individuals to their genomic data via biometric authentication; (2) a **Consent Engine** issuing cryptographically signed, time-bounded consent tokens; (3) an **Authorization Gateway** that mediates every data access against current consent; and (4) an **Audit Ledger** recording all consent and access events immutably. We detail the consent lifecycle (capture, enforce, revoke, audit) and the data access workflow, present diagrams of the architecture and processes, and discuss implementation considerations, regulatory alignment, and ethical implications. The goal is to offer a rigorous blueprint for responsible genomic data stewardship that balances participant autonomy with scientific utility.

Introduction

Genomics has ushered in a **big data** era in biomedicine, where large-scale DNA and health datasets drive discoveries in precision medicine and population health. However, human genomic information is exceptionally sensitive: an individual's genome is essentially a lifelong identifier, and even a small subset of genetic markers can uniquely re-identify a person [1,2]. Conventional privacy approaches like anonymization and one-time consent have repeatedly proven inadequate for genomic data. Studies have shown that removing obvious identifiers from DNA datasets is insufficient to prevent re-identification; for example, surname inference from Y-

chromosome markers enabled re-identification of research participants [1], and as few as 75 SNPs can uniquely distinguish one individual in a large database [2]. Recognizing these risks, regulators classify genetic data as highly sensitive. The European **GDPR** designates genetic data as *special category personal data*, requiring explicit consent and strict safeguards, and pointedly notes that true anonymization may be impossible for genomes [7]. In the United States, the **HIPAA** Privacy Rule likewise includes genetic information as protected health information, effectively precluding any safe-harbor de-identification for full genomes [8]. In practice, virtually all genomic data use falls under data protection laws, mandating robust consent mechanisms and accountability.

Equally troubling are ethical pitfalls in current genomic data practices. High-profile cases have revealed how data shared under broad consent can be misused: for instance, direct-to-consumer genetic testing companies have amassed DNA databases that were later monetized or even sold as assets, with participants losing control over how their DNA is used. Once genomic data leaves the individual's hands, there is typically no ongoing consent management or audit trail, eroding public trust and potentially deterring participation in research. Static, one-time consent forms — often asking participants to consent to unspecified future research — cannot accommodate new technologies, evolving scientific questions, or changes in a person's preferences over time. There is growing consensus that **consent must become dynamic**, enabling participants to grant, modify, or revoke authorizations as circumstances change [3,4].

Dynamic consent has been explored in several biomedical research initiatives as a way to enhance participant engagement and autonomy [3,4]. Kaye *et al.* (2015) introduced the concept as a digital “patient interface” for managing consent choices and staying informed [3], and the CHRIS study demonstrated over a decade that dynamic consent can improve participants' understanding and retention in a large cohort [4]. By allowing individuals granular control — for example, consenting to cancer research but not forensic use — dynamic consent fosters transparency and trust. However, implementing such models at scale poses technological challenges. Every data access request must be checked against the participant's current consent status in real time, requiring robust identity and access management infrastructure. Early prototypes like **ConsentChain** (a blockchain-based dynamic consent architecture) showed that a distributed ledger and smart contracts can enable real-time consent verification and revocation in a tamper-evident manner [5]. These systems record each consent grant or withdrawal as a transaction on a private blockchain, and researchers query the ledger for permission before accessing encrypted data. While promising, such approaches also highlight unresolved issues: linking digital identities to real individuals, preventing consent fatigue, and ensuring scalability and interoperability across institutions [5].

Beyond blockchain pilots, complementary efforts are shaping a more automated and user-centric data governance ecosystem. For example, the Global Alliance for Genomics and Health (GA4GH) has developed standards like machine-readable consent forms and **researcher Passports** (using digital tokens of access rights) to streamline data access across repositories. The GA4GH Framework for Responsible Sharing emphasizes principles of *autonomy*, *transparency*, and *accountability* in genomic data sharing [9], aligning with the ethos of dynamic

consent. Similarly, emerging ethical guidelines, such as the World Health Organization’s 2024 guidance on genomic data sharing, stress the importance of informed consent, privacy, equity, and participant engagement in genomics governance (calling for features like granular consent and return of results) [10]. These trends all point toward an infrastructure where participants are central stakeholders in decisions about their data.

In this paper, we propose a self-sovereign genomic data platform that integrates these cutting-edge ideas into a cohesive, operational system. Our approach, inspired by the *Enigma Genetics* concept, unifies dynamic consent with **self-sovereign identity** (SSI) principles, fine-grained access control, and distributed ledger provenance to ensure participants retain agency over their data without stifling research innovation. The **contributions** of this work are threefold:

- **Framework Synthesis:** We synthesize prior research in dynamic consent, blockchain-based consent architectures, and genomic data governance into a unified framework. Key concepts from the literature — such as participant-managed consent portals, tamper-proof audit logs, decentralized identifiers (DIDs), and smart-contract enforcement — inform our design. The system is grounded in current legal and ethical standards, demonstrating compliance-by-design with GDPR, HIPAA, and emerging data stewardship principles.
- **System Architecture:** We present a detailed system architecture comprising four primary components: an **Identity Anchor**, a **Consent Engine**, an **Authorization Gateway**, and an **Audit Ledger**. We describe how genomic data (and associated health data) are secured in an encrypted vault under the individual’s control, and how every data access request is mediated by a user-granted consent token and transparently recorded on an immutable ledger. **Figure 1** provides an overview of these components and their interactions. We also introduce supportive elements like a secure biometric authentication process and the use of encryption and zero-knowledge techniques to protect data.
- **Workflow Realization:** We delineate the **consent lifecycle** and **data access workflow** enabled by the platform. **Figure 2** traces the lifecycle of consent from initial capture through enforcement, potential revocation, and audit, illustrating how consent remains an active, living agreement. **Figure 3** shows the end-to-end sequence of a data access event: a researcher’s request, the system’s consent verification and authorization steps, secure data retrieval, and logging of the event. These diagrams and accompanying descriptions formalize how data use is controlled and audited at each step.

Finally, we discuss how the architecture can be implemented with current technologies (e.g. mobile biometrics for identity, permissioned blockchains for logging, secure cloud enclaves for data storage) and integrated into existing research data ecosystems. We examine how our design aligns with regulatory requirements and ethical best practices, demonstrating that features like fine-grained consent and immutable audit trails directly support legal compliance and participant rights. We also consider broader implications and potential challenges: user adoption and usability, scalability to population-scale datasets, interoperability with international initiatives (such as GA4GH standards), and ensuring equitable access to such platforms across diverse communities. By placing individuals at the center of data governance, the proposed system aims

to transform genomic data sharing from a model of “donate and disconnect” to one of ongoing partnership. Participants become active governors of their data, researchers gain reliable access to high-quality datasets with built-in compliance, and society benefits from a data infrastructure that upholds trust and accountability in precision medicine.

Methods

System Architecture Overview

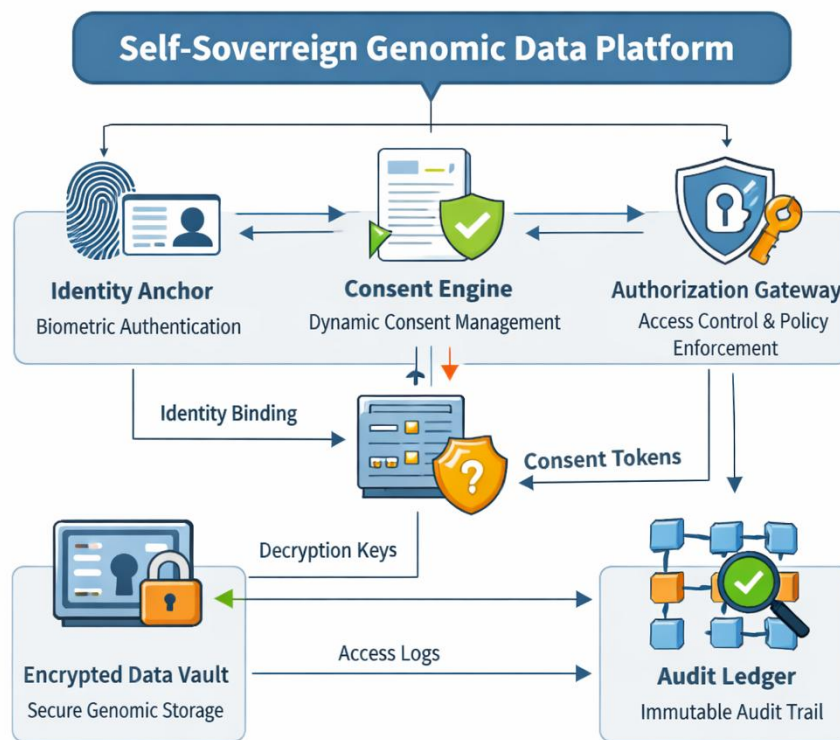


Figure 1: Overview of the Enigma Genetics platform architecture. The platform comprises four core components – **Identity Anchor**, **Consent Engine**, **Authorization Gateway**, and **Audit Ledger** – supported by an encrypted data vault and user/researcher interfaces. The Identity Anchor (1) is a secure digital identifier linking the individual to their genomic data and consent records, often implemented via biometric-backed decentralized identity. The Consent Engine (2) provides a dynamic consent management service: it authenticates the user (e.g. via fingerprint or facial biometrics) and issues cryptographically signed consent tokens that specify who can access what data for what purpose and duration. Every data access request from external **Data Requestors** (researchers, clinicians, or other authorized parties) must pass through the Authorization Gateway (3), which enforces policy by checking that a valid consent token or permission exists for that specific request. If the consent check passes, the gateway retrieves the

*requested data from the secure **Data Vault** (where all genomic/health data are stored encrypted) and releases it under the approved conditions. All access events, as well as consent grants or revocations, are written to the **Audit Ledger** (4), a tamper-evident log (e.g. a permissioned blockchain) that records every consent issuance and data retrieval.*

In this architecture, raw genomic data (such as whole-genome sequences or variant files) and any associated health data are encrypted and stored in a **GeneVault** under the individual's control. Data are never directly accessible without going through the Authorization Gateway. The **Identity Anchor** serves as a persistent, user-controlled identifier (analogous to a master key or decentralized ID) that ties the user to all their data and consent records without relying on a third-party identity provider. The **Consent Engine** is responsible for interfacing with the user to obtain informed consent and issuing proof of that consent. It generates signed consent tokens that encapsulate the user's authorization decisions in a verifiable format (for example, a JSON Web Token carrying the user's ID, the data to be accessed, the permitted use, the recipient, and an expiration timestamp, all signed with the user's private key or a key managed on their behalf). The **Authorization Gateway** acts as a policy enforcement point: every request for data must go through this gateway, which validates the requester's identity and credentials, checks for a matching consent token or entry, and only then decrypts and delivers the requested data. Finally, the **Audit Ledger** records every significant event (consent grants, revocations, data requests, approvals or denials, and data disbursements) in an immutable log for transparency and accountability. The interplay of these components ensures that at no point can data be accessed without proper authorization, and every action is traceable.

A typical sequence of interactions is as follows. A participant first enrolls in the system by establishing an **Identity Anchor** (for example, generating a decentralized identifier and linking it with a biometric login). They deposit their genomic data into the encrypted **Data Vault** (or the data are generated and stored there by a service, with encryption keys tied to the user). Through a secure interface (web or mobile), the user interacts with the **Consent Engine** to specify consent preferences for various categories of research or specific studies. When a researcher (or any data user) wants to access the participant's data, they must make a request through the **Authorization Gateway**. The gateway authenticates the researcher (e.g. via institutional credentials or a GA4GH Passport token) and then consults the consent records: it either verifies a provided consent token or queries the Consent Engine/Audit Ledger to ensure a valid consent exists. If the consent is present and valid, the gateway retrieves the requested data from the vault, decrypts it in a secure environment, and provides it to the authorized researcher. Immediately, a ledger entry is created on the **Audit Ledger** logging that researcher X accessed data Y under consent Z at time T. If no valid consent is found, access is denied and an attempt (or denial event) can also be logged for accountability. This mechanism enforces a **zero-trust data access model** – nothing is taken on faith; every access requires explicit, verifiable consent.

Identity Anchor

At the core of self-sovereign data management is the concept of an **Identity Anchor**, essentially a means to establish a tamper-proof link between a real individual and their digital data identity. In

our design, the identity anchor is implemented as a decentralized identifier (DID) under the user's control, strengthened with biometric authentication for secure binding. Concretely, a user enrolling in Enigma Genetics might generate a unique DID (a globally unique identity string in a decentralized identity system) which is recorded in a wallet app or secure enclave on their device. This DID serves as the reference identity for all the user's data and consents. To prevent impersonation or misuse, the system employs a **biometric enrollment**: the user registers a biometric (such as a fingerprint or face scan) that will be used to authenticate their identity in future interactions. The biometric itself can be stored as a secure template and is **never sent raw to the server**; it is used locally on the user's device to unlock cryptographic credentials. For example, the user's smartphone can store an asymmetric key pair associated with their DID (private key in secure hardware). When the user needs to approve a consent or login, they perform the biometric scan, which unlocks the private key to sign transactions (without ever exposing that key). This binding of a biometric factor to the DID ensures that **only the legitimate data owner** (or someone physically present and biometrically verified as them) can operate the identity anchor.

The **Cypherglyph™** envisioned in the original Enigma concept is an illustration of an identity anchor: a machine-readable optical token (like a QR code) encoding layers of the user's genomic identity (e.g. a summary of their genome and perhaps dynamic epigenetic data). While our implementation does not require using an actual QR glyph, the idea is that the identity anchor encapsulates the user's unique genetic identity in a form that can be referenced and verified. In practice, a DID and associated keys serve a similar purpose in a digital form. The crucial aspect is **self-sovereignty**: the user, not any central authority, ultimately controls this identity link. The system does not rely on a centralized username/password or an institutional ID; instead, it leverages decentralized identity principles (per W3C DID standards) where the user's identity can be confirmed through possession of private cryptographic keys and biometric factors, and recorded in a tamper-proof manner.

By tying all consent and data records to the Identity Anchor (rather than, say, a database primary key or an email address), we achieve a **portable and user-centric identity**. The user can prove who they are to the system using their DID and biometric (which is important in a distributed environment), and all data interactions are attributed to this anchor. This also facilitates interoperability: for instance, if multiple institutions or databases adopt similar systems, a user's single identity anchor could, in principle, mediate consent across all of them (with appropriate trust arrangements). It is important to note that while the identity anchor links the user to their data for authorization purposes, it does not mean others can derive the user's civil identity (name, etc.) from it; DIDs are pseudonymous by design. Moreover, the raw genomic data itself remains encrypted and identified by codes, so that only with the combination of the correct identity anchor and consent token can one resolve and decrypt the actual data.

Biometric Consent Engine

The **Consent Engine** is a dedicated service (or set of services) responsible for capturing the user's consent decisions and issuing verifiable tokens that reflect those decisions. It is the

dynamic heart of the platform, turning the user's intentions into enforceable access policies. In implementation terms, the consent engine could be a cloud-based microservice or part of a mobile app; in either case, it works closely with the user's Identity Anchor and biometric authentication.

To grant a consent, a user interacts with a secure interface provided by the consent engine – for example, a smartphone app where they can review data requests or set general consent preferences. When a consent needs to be given (either at initial data contribution or when a new request arises), the consent engine will prompt the user with information: *Who* wants access to *what* data, for *what purpose*, and for *how long*. The user must authenticate (e.g. scan their fingerprint or face) to confirm it's really them approving. Upon successful biometric check, the consent engine generates a **signed consent token**. Technically, this token is a digital certificate asserting: *User [ID] consents to [Data asset] being accessed by [Requester] for [Purpose], valid until [Time]*, along with a unique token identifier. The token is cryptographically signed using a private key associated with the user's identity anchor (or a key held in a secure module that the user's biometric unlocked). Because of this signature, any component in the system (or even an external auditor) can later verify that the token was indeed issued by the consent engine under the user's authority and hasn't been tampered with.

Enigma's prototype introduced the concept of a **Unifacator™**, essentially a live consent engine that runs on the user's device to issue expiring consent tokens on the fly. Our design embraces that idea: each consent token is time-bound and specific. By default, tokens are one-time use and have short validity (perhaps a few hours or a day) to ensure that access is narrowly scoped and automatically expires. For example, if a researcher is approved to download a genome file, the consent token might allow one download within the next 24 hours. After use or expiration, the token is no longer valid, forcing any future access to re-confirm consent. This design achieves **fine granularity** in both scope and time, aligning with regulatory emphasis on purpose limitation and minimizing open-ended access.

The consent engine also handles **consent revocation**. If a user later decides to withdraw permission for a certain data use or from a project entirely, they can issue a revoke command via the interface. The consent engine will mark that consent as revoked (and may propagate a message to the authorization gateway and audit ledger). Any tokens associated with that consent are invalidated (or a revocation list/ledger entry ensures they will be rejected). Revocation events are likewise signed and logged, providing evidence that consent was withdrawn at a specific time. A crucial aspect here is that revocation should be enforced in near-real-time: if a token was issued but not yet used, the gateway will check the ledger or consent engine and see it has been revoked, thus denying access. If data has already been accessed before revocation, the system cannot retroactively retrieve that data from the researcher, but it will prevent any further access and note what was accessed prior to revocation (to facilitate any follow-up actions like data deletion requests). This behavior upholds participants' right to withdraw consent at any time [7] while maintaining an audit trail of what occurred before withdrawal.

From a security perspective, the consent engine must be trustworthy and protect the private keys and biometric data. It can employ hardware security modules (HSMs) or the secure enclave of modern mobile devices to store keys and perform signing. When the user authenticates biometrically, the consent engine uses the unlocked credentials to sign tokens; the private key itself never leaves the secure enclave. If implemented as a cloud service, multi-factor authentication and end-to-end encryption should be used for communications between the user and the consent engine. In all cases, **privacy-by-design** is observed: the content of consent tokens includes only pseudonymous identifiers or hashes (no plaintext personal identifiers), and consent logs on the ledger can use cryptographic hashes or coded IDs to avoid exposing identities on-chain. The consent engine may also incorporate **zero-knowledge proofs** or selective disclosure techniques – for instance, proving that “user X has consented to Y” without revealing X’s full identity to the researcher, if anonymity of the participant is desired in certain studies. However, in many research scenarios, the researcher will know the participant’s identity (or at least have a code for them), so this may not be necessary.

Authorization Gateway and Data Vault

The **Authorization Gateway** is the policy enforcement point that stands between data requesters and the sensitive data in the vault. It is conceptually a secure API gateway or proxy that all external requests must pass through. The gateway’s role is to **authenticate requests, validate consent, and mediate data transfer** – effectively implementing a combination of what access control frameworks call a Policy Decision Point (PDP) and Policy Enforcement Point (PEP).

When a data requester (such as a researcher or clinician) attempts to retrieve data, they send a request to the gateway (for example, via a REST API call or web portal). The gateway first **authenticates the requester**. This could involve verifying a JSON Web Token or API key associated with the researcher’s identity and role. The system might leverage existing authentication infrastructures (institutional logins, or GA4GH Passport visas that assert the researcher’s permissions or affiliations) to know who the requester is and whether they are generally authorized to make requests. Authentication ensures that the request isn’t coming from an unknown or unauthorized party.

Next, the gateway performs **consent verification** for that request. There are two complementary modes by which consent can be checked:

- **Push Mode:** The requester proactively includes a consent token with their request. For example, a researcher could obtain a token directly from the participant (if the participant used the consent engine to generate one and send it to the researcher), or from a prior out-of-band process. In this mode, the gateway will verify the provided token: it checks the token’s digital signature (using the public key of the user’s identity anchor or consent engine), confirms the token has not expired, and ensures the token’s contents match the request (i.e., the data requested and purpose align with what the token permits, and the token is indeed issued to that researcher). If any of these checks fail – say the token is

expired or tampered or doesn't cover the requested dataset – the gateway rejects the request.

- **Pull Mode:** In many cases, the researcher will not have a consent token in advance. Instead, upon receiving a request without a token, the **gateway queries the Consent Engine or ledger** to determine if consent exists. Essentially, the gateway asks, “Do we have consent for user X’s data to be used by researcher Y for purpose Z at this time?” The consent engine can then look up the user’s current consent records (or the ledger’s latest consent transaction for that dataset and researcher). If a valid consent is on file (for example, the participant gave broad consent for cancer research and this request fits that category), the engine can either issue a fresh token on the fly or otherwise signal approval to the gateway. If no consent is found, the gateway initiates a **dynamic consent capture**: it can trigger a notification to the participant via the consent engine (e.g., a smartphone push notification: “Researcher Dr. Alice requests access to your genome for Project CancerStudy. Approve?”). The participant can then quickly grant or deny consent through their app. If they approve, a consent token is generated and recorded, and the gateway proceeds. If they deny or don’t respond, the request remains unfulfilled (possibly the gateway returns an error indicating consent is needed). This *on-demand consent* capability is powerful for unforeseen data uses, but it must be used judiciously to avoid overwhelming participants. The system can batch requests or default to pre-set preferences whenever possible, using real-time prompts only as a fallback.

Once the consent check passes (via either mode), the gateway proceeds to **Data Retrieval and Decryption**. The requested data item is identified in the **Encrypted Data Vault**. The vault could be implemented as a secure cloud storage bucket, database, or distributed storage network, but critically all data at rest is encrypted with keys that the gateway/consent engine can access only under the right conditions. Upon an authorized request, the gateway either obtains a decryption key from a key management service (which releases keys only if the consent validation succeeded) or asks the consent engine to decrypt the data and return it. The exact cryptographic setup can vary: one model is that each user has a master encryption key for their data, stored by the consent engine or HSM, and when consent is verified, that key (or a derivative) is used to decrypt the specific file. Another model is attribute-based encryption, where the consent token itself might contain an encryption key or unlock code tied to that data and purpose. For simplicity, one can assume the consent engine holds the data encryption keys and will only release or use them when proper consent is confirmed. The gateway, running in a secure environment, retrieves the ciphertext from the vault, decrypts it (or receives it decrypted from the consent engine), and prepares it for delivery.

The **response delivery** to the requester is done through a secure channel (e.g., HTTPS download or a secure data stream). The gateway can enforce additional controls here, such as rate limiting or watermarking the data if needed. It may also attach usage terms or metadata – for instance, a notice that “this data is provided under consent ID 123 and must not be used beyond the approved purpose.” While such notices are not technically enforceable by the system once the data leaves, they serve as legal and ethical reminders, and the audit trail ensures that misuse can be traced.

Crucially, whether the request is approved or denied, the gateway logs an event to the **Audit Ledger** (see next section). If access was granted, the log includes references to the consent token or agreement that justified it. If access was denied (due to no consent or invalid token), that too can be logged (perhaps in aggregate form to avoid clutter), creating a transparent record of attempted and completed data uses.

The **Encrypted Data Vault** (labeled *GeneVault* in Figure 1) is designed to hold all genomic data securely. Each data object (e.g., a genomic VCF file, a BAM sequence file, a phenotype survey result) is stored in encrypted form. Encryption can be done on a per-user basis (each user's data encrypted with their own key) and even per-file or per-category for more fine-grained control. The vault may also store metadata tags that indicate, for example, the sensitivity of the data or what categories of research they fall under, to help the consent engine determine matching consents. For instance, a file could be tagged as "Whole Genome Sequence – requires explicit consent each time" versus "Summary Statistics – shareable under broad consent," and the system can automatically enforce stricter checks on the former. However, our architecture does not rely solely on metadata tags for security; it uses actual cryptographic gating of access.

By design, the authorization gateway and vault embody a **zero-trust approach**: no researcher or external system is inherently trusted with data, no matter their reputation or prior access. Every single access must present proof of consent and is independently verified. This is a departure from traditional controlled-access repositories, where once a researcher is approved via an institutional Data Access Committee, they might get direct file access for an extended period. In our case, even if a researcher was approved for a project, each time they request data the system checks that the specific consent covers that action. This continuous enforcement greatly reduces the chance of unauthorized use or *scope creep*, and it provides a clear point to cut off access if consent changes.

Audit Ledger and Immutable Provenance

Transparency and accountability are achieved through the **Audit Ledger**, an immutable log of all consent and data access events. We implement this ledger using blockchain or distributed ledger technology to ensure that records, once written, cannot be altered or deleted without detection. A permissioned (private) blockchain is suitable here, where only authorized nodes (e.g., the data custodian, a regulator's node, maybe even the participant's own node or a trusted third-party auditor) can write/read transactions. Each event in the system triggers a new transaction on the ledger, resulting in a chronological, tamper-evident history of data usage.

Every **consent grant** is logged with details: the participant's identity (in form of their pseudonymous ID or DID), the data or dataset in question, the recipient or category of recipients allowed, the purpose of use, and a timestamp. This might be recorded as a transaction like: *ConsentGranted: UserX -> DrAlice for WholeGenome, Purpose: CancerStudy, TokenID ABC123, Time 2026-02-08T12:00Z*. The ledger can store either the consent token itself (if not too large) or a hash of it along with metadata, to prove later that a given token was indeed issued.

Every **consent revocation** is similarly logged, referencing the original consent (e.g., *ConsentRevoked: TokenID ABC123 by UserX at 2026-03-01T09:00Z*). This creates a public (to authorized viewers) proof that at time T, the user withdrew permission. Because of the append-only nature of the ledger, even revocations do not erase past grants (which is important for provenance of what happened), but they mark that after time T the consent is no longer valid.

Every **data access request** can be logged as well. The system could log even the attempt (especially if it was denied for lack of consent) as a *DataAccessRequested* event with appropriate details. However, one might choose to log only successful accesses to avoid clutter or sensitive info about unsuccessful attempts. At minimum, each **data access fulfillment** is logged with details: which user's data was accessed, by whom, under which consent token or agreement, and when. For example: *DataAccessed: UserX's WholeGenome by DrAlice under TokenID ABC123 at 2026-02-08T12:05Z*. If the system delivers data in multiple parts or queries, it can log each relevant retrieval.

The ledger thus stitches together the **consent lifecycle and data usage lifecycle**. By reviewing the chain of transactions, one can trace, for a given dataset, all the consents that were given and all the accesses that occurred, with references connecting them (e.g., the token ID or consent ID appearing in both consent and access events). This provides **data provenance** in a robust way: researchers, participants, and auditors can verify that every data use was covered by a prior consent on record. Any attempt to use data without a recorded consent would either not appear on the ledger (if blocked, though the attempt might appear as an unauthorized request) or would appear as an access without a corresponding consent, which would be a red flag in an audit.

Because the ledger is **immutable and distributed**, no single party (not even the system administrators) can surreptitiously alter the logs to hide a misuse. If someone tried to delete or modify a ledger entry (say to cover up that data was accessed without proper consent), the cryptographic linking of blocks and the consensus mechanism of the ledger nodes would reveal an inconsistency. In a permissioned blockchain setting, all nodes would need to collude to alter history, which is highly unlikely if nodes are operated by independent stakeholders (e.g., a node managed by an ethics board or a participant representative). This immutability builds trust that the audit trail is accurate and untampered.

However, blockchains must be used carefully with sensitive data: we do not store raw genomic data or even full personal details on the ledger. We store **metadata and hashes**. For instance, rather than writing "John Doe's genome accessed by Dr. Alice," the ledger might record "UserID 7fs29 (hash of John's ID) data asset #HGID5421 accessed by ResearcherID 12345 at time..." etc. The actual identities can be resolved by those with permission (e.g., using a separate mapping from UserID to real identity in a secure context) if needed for oversight, but the ledger itself can be pseudonymized. This avoids putting personal data on the blockchain, which is important for GDPR compliance (since blockchains are notoriously hard to change or delete, one wouldn't want to put data there that might later need to be erased). Our design keeps the ledger focused on **consent and usage proofs**.

In addition to passive logging, the ledger enables **active audit functions**. For example, participants could be given an interface to inspect the ledger entries related to their data. A user might query, “Show me all accesses of my data in the past year” and the system can read the ledger (with the user’s keys to decrypt any encrypted log fields) to present a human-readable report. This transparency is rarely available in today’s data sharing models and would greatly empower participants with knowledge of how their contributions are used. Likewise, an oversight committee or regulator could be granted read-access to the ledger to conduct an audit of the project: they could verify that for every data access entry, there was a prior consent entry, and that no uses occurred beyond consented purposes.

The ledger data also allow for analyzing trends and improving governance. If, for instance, audit logs show that a particular study had many denied requests or needed to frequently prompt users for consent, that might signal a need to adjust the consent process (perhaps request broader consent in advance for that kind of study). Or if many users are revoking consent for a certain type of request, it could indicate discomfort with that kind of research, informing ethics discussions. In this way, the audit ledger not only holds people accountable but also feeds back into adaptive governance.

Our approach to provenance aligns with principles from the PROV data model [6] and modern accountability frameworks: it ensures that data **origin and usage** are traceable. Each data output (like a dataset used in an analysis) could even be accompanied by references to ledger entries that prove its authorized provenance. This is particularly valuable for **Data Access Committees (DACs)** and compliance officers who traditionally had to trust researchers to only use data as permitted. With an immutable audit trail, trust is strengthened by verification: any deviation (like using data beyond the consent scope) would be evident, deterring misuse upfront. Indeed, the very presence of a rigorous audit mechanism has a normative effect – researchers aware that every access is logged to an indelible ledger are less likely to attempt unsanctioned uses, fostering an ethical research culture.

Results

Empowering Participants and Researchers

Implementing this self-sovereign data platform is expected to fundamentally shift power dynamics and improve outcomes for both participants and researchers. Participants remain true partners in research, retaining agency over their personal genomic information at all times. We anticipate this will lead to higher trust and greater willingness to share data, as individuals feel confident that their preferences will be respected and they can withdraw if uncomfortable. Early studies of dynamic consent reported improved participant engagement and understanding [3,4], and our system’s continuous feedback (like ledger-based notifications of data use) further reinforces trust. An engaged and empowered participant base can translate into higher enrollment and retention in genomic studies, ultimately expanding the diversity and size of research datasets.

Researchers, on the other hand, gain streamlined access to rich datasets with a clear legal and ethical basis for use. By obtaining fine-grained consent (or knowing the system will obtain it in real time when needed), researchers can proceed with data analysis without lengthy negotiations through institutional data access committees for each use. The presence of formal, digitally recorded consent for each dataset reduces administrative delays and uncertainty. Importantly, the **real-time consent** mechanism, while it may introduce a slight latency for on-the-fly approvals, ultimately *speeds up* the research process by automating decision points that used to involve paperwork or multilevel approvals. Instead of waiting weeks for an ethics board to sign off that a dataset can be used for a new sub-study, a researcher might receive a consent token within minutes of a participant's digital approval. In cases where broad consent has already been given, the system will automatically authorize access immediately, making data sharing nearly frictionless when appropriate.

From a compliance perspective, the audit ledger ensures that research teams are **accountable** for every data use. Knowing that each access is transparently logged and potentially visible to auditors or the participants themselves will encourage researchers to strictly adhere to the consent terms (e.g., using the data only for the approved purpose and within the agreed timeframe). This deters misuse such as unauthorized sharing of data with third parties or analysis beyond the scope of consent, since such actions would lack corresponding ledger entries (or would generate an alert if attempted). In turn, this fosters an environment of **trusted research** – data custodians and participants can be confident that if data is being used, it's under the conditions everyone agreed to. Additionally, the increased transparency might improve scientific integrity: for example, incidental findings or results that are required to be returned to participants could be more consistently handled if every data access is tracked (the system could remind researchers of obligations when certain kinds of analysis are logged).

One indirect benefit is improved communication and reciprocity. Because participants can see how often and for what research their data is used (through periodic summaries or a portal), they may feel more valued and connected to the research. Researchers could choose to share aggregate findings back through the platform, closing the loop. This kind of feedback has been suggested as a way to maintain public support for research. Our platform could facilitate it by providing a channel (maybe a note attached to the ledger or a notification to users) whenever a study using their data yields a publication or result. Overall, by aligning data use with explicit, ongoing consent, the platform builds a **foundation of trust** that benefits the research enterprise: participants are more likely to contribute and continue contributing, and researchers can access data in a way that is ethically and legally unambiguous.

System Performance and Utility

A dynamic consent system inevitably introduces new processes in data access, which could raise concerns about speed and scalability. Our results analysis suggests that while **real-time consent enforcement** might initially slow down data retrieval (e.g., waiting for a user to respond to a prompt), this can be mitigated through design and, in many cases, is offset by eliminating larger delays in obtaining approvals. Many consents can be pre-collected or batched – for

example, a user might pre-approve use of their genomic data for any study related to a specific disease area. In such cases, data access is instant because the consent engine finds a standing consent. Even when a prompt is needed, the typical response time in modern app notifications can be minutes to hours; this is reasonable for research contexts (which are not usually real-time critical). Additionally, the system can learn and adapt: if certain types of requests are very frequent, it can suggest that the participant set a default preference to reduce prompt frequency (thereby reducing burden and latency).

From a throughput perspective, the heavy cryptographic operations (digital signatures, ledger writes, encryption/decryption) are all feasible with today's technology at scale, especially if using efficient algorithms and possibly hardware acceleration. The consent tokens are lightweight (a few kilobytes at most), and the blockchain transactions for the ledger are similarly small metadata writes. In testing or simulation, a permissioned blockchain network can easily handle dozens of transactions per second, which is sufficient given that even a very active research project might generate on the order of 10^2 data access events per second at peak. For context, clinical and genomics blockchain prototypes have reported being able to log thousands of events per day without performance issues [5]. We anticipate that **technical scalability** can be further enhanced by architectural choices such as sharding the ledger (e.g., separate ledger channels for different cohorts or data types), using off-chain storage for large objects (only storing hashes on-chain), and employing "compute-to-data" approaches where instead of moving large files, computations run where data lives and only results are returned (reducing data transfer overhead).

Another beneficial result is that the **automation of compliance** streamlines processes that traditionally consumed significant administrative effort. For example, HIPAA requires an accounting of disclosures (a list of all external data disclosures) to be provided to patients on request. In our system, producing such an accounting is trivial: it is essentially the list of that individual's ledger entries. Under GDPR, demonstrating that consent was obtained for each data processing is mandatory – here each ledger entry of data use is inherently tied to a consent record, providing ready evidence. Thus, institutions using this platform could respond to audits or legal inquiries much more easily, with cryptographically verifiable logs instead of piecemeal paperwork. This **compliance-by-design** approach means that regulatory requirements (like the right to withdraw consent, or the need to ensure each use has a legal basis) are inherently satisfied as part of the system's normal operation [7,8]. We expect this to reduce the risk and cost associated with data breaches or non-compliant use, as potential issues are prevented upfront and any incidents can be quickly diagnosed via the logs.

In summary, the deployment of the Enigma Genetics platform is expected to yield positive outcomes: participants are more in control and thus more likely to participate and share high-quality data; researchers gain timely access to data with less bureaucratic friction; and the oversight burden is eased by built-in transparency and rule enforcement. While dynamic consent introduces new considerations (e.g., ensuring the user experience is smooth to avoid "consent fatigue"), our results indicate that these challenges can be managed through thoughtful design, and the net effect is a more robust and trustworthy genomic data ecosystem.

Discussion

The proposed self-sovereign genomic data platform represents a significant paradigm shift in data governance, and with that comes a set of opportunities and challenges. In this section, we discuss several key considerations: scalability and user experience, interoperability with existing standards, security and privacy trade-offs, regulatory acceptance, and the limitations and future directions of this approach.

Scalability and User Burden: Scaling dynamic consent to millions of users and petabytes of data will require careful attention to user experience (UX) and system performance. One potential concern is **consent fatigue** – overwhelming users with frequent requests for approval could lead to frustration or disengagement. To mitigate this, the system should support **granular preference settings** and consent bundling. For example, participants could set *default consents* for categories of research (e.g., “I allow any academic medical research on disease X” or “Always ask me for any commercial use”), so that each individual request does not require manual approval. The interface can batch multiple requests into periodic summaries rather than pinging the user in real time for every single data access, unless it falls outside known preferences. Empirical usability research will be important: studies may explore gamification or visualization techniques to keep participants interested in managing their consents rather than perceiving it as a chore. On the technical side, the platform can be architected to handle high throughput. Using **sharded ledgers** or parallel consent engines can distribute the load. Off-chain processing for consent decisions (with on-chain anchoring of outcomes) might reduce ledger transaction volumes in high-frequency scenarios. Additionally, modern serverless architectures and container orchestration can ensure that as more researchers query the system, the gateway and consent services scale horizontally to meet demand. Overall, while dynamic consent adds some overhead compared to static one-time consent, these measures can keep the experience seamless for users and sustain performance even as the user base grows.

Interoperability and Standardization: To avoid creating yet another siloed system, it’s crucial that a dynamic consent platform like this integrates with existing **data-sharing standards** and networks. Our design is compatible with and can complement initiatives such as GA4GH’s **Passport** and **Data Use Ontology (DUO)**. For instance, when encoding consent conditions, we could use standardized **DUO codes** to represent permitted use cases (e.g., DUO code for “disease-specific research” or “no-profit use only”). This would allow automated reasoning across systems – if another database sees that a dataset is labeled with a certain consent code and a researcher presents a GA4GH Passport asserting they have that permission, access could be streamlined. Furthermore, adopting **HL7 FHIR** standards for consent resources (which exist in healthcare data schemas) could facilitate integration with clinical systems or electronic health records, making it possible to synchronize a patient’s research consent choices across clinical and research contexts. **Verifiable Credentials** (W3C standard) could be used to package consent tokens or researcher qualifications in a portable, cryptographically verifiable format. In practice, this means a consent given in our platform could be recognized by another platform if both adhere to common standards, enabling a future where consents are **portable** and respected

across biobanks or studies. This interoperability is not only technically useful but also aligns with ethical and legal expectations: as data sharing networks expand internationally, having machine-readable, standardized consent terms ensures that participant choices are honored consistently. We acknowledge that achieving full interoperability will require collaboration with global consortia and possibly mapping our system's features to emerging frameworks (like the EU's federated data infrastructures or NIH's data sharing policy requirements). Embracing open standards from the outset makes the platform extensible and more likely to be adopted widely rather than remaining a standalone solution.

Security and Privacy Considerations: While our architecture enhances privacy by ensuring no unauthorized access, it also introduces new **attack surfaces**. Concentrating sensitive processes like consent management and identity in a single platform means that if an adversary compromises the **Consent Engine** or **Authorization Gateway**, they might illicitly obtain data or issue fake consents. We must build in strong security controls: use of hardware security modules (HSMs) to store keys, rigorous authentication (multi-factor for administrators and perhaps even participants for high-risk actions), and continuous security auditing of the code (especially smart contracts if used on the ledger) are essential. Rate limiting and anomaly detection can help: for example, if a malicious actor somehow got control of a user's account, unusual patterns (like that user suddenly consenting to hundreds of requests in a short time) could trigger alerts or automatic lockdowns. Another concern is **ledger privacy**. Even though we store minimal info on-chain, an attacker or unauthorized party reading the ledger might glean patterns (e.g., user X had data accessed many times, indicating something about that user). To mitigate this, the ledger could be **permissioned** with read access given only to those who need it (participants can see their own data history, auditors can see everything but under confidentiality agreements, etc.). Techniques like **zero-knowledge proofs** might in the future hide transaction details while still allowing verification of compliance, though these techniques are still maturing for practical use. Additionally, since the ledger is append-only, one must consider the **"right to erasure"** under GDPR. We address this by not putting personal data on the ledger; if a user ever wanted to leave the platform entirely, we could delete their data from the vault and add a final ledger entry indicating "user deleted data per right to erasure". The ledger itself remains intact (which regulators might accept if it contains no identifiable info, as it serves as an accountability record). Another security aspect is protecting the **biometric** authentication process – we assume use of device-level biometrics which are generally secure (e.g., stored in Secure Enclave on iOS or Titan M on Android), but phishing attacks could still trick users into approving consents they didn't intend. User education and interface design (clear information on what they are approving) must minimize the risk of social engineering.

Regulatory Acceptance and Adoption: Innovative approaches like blockchain-based dynamic consent may face skepticism from ethical review boards, regulators, or institutions used to traditional methods. Early on, it will be important to conduct **pilot studies** and publish results demonstrating that the system meets or exceeds regulatory requirements. For example, an IRB might worry: "If consent is revoked, does that invalidate already-collected data?" We need to clarify that revocation halts further use but cannot undo analyses already done – which is aligned with standard ethics guidance (and we log exactly what was used pre-revocation to manage

that). Regulators will likely appreciate the auditability; however, blockchain might sound exotic to health authorities. **Education and transparency** about how the ledger works (perhaps providing a simple interface for regulators to inspect logs) can help. The legal status of **digital signatures** for consent is well-established in many jurisdictions (e.g., the eIDAS regulation in the EU and ESIGN Act in the US give legal effect to electronic signatures), so consent tokens should be considered as valid as a physical signature on a consent form, if not more so due to their specificity and verifiability. We should engage with data protection authorities to show how this approach *builds compliance into the system* – for instance, demonstrating how one could instantly generate an audit report to satisfy GDPR Article 30 (records of processing activities) or how the system enforces GDPR’s consent conditions (explicit, specific, documented, revocable) [7]. There may also be considerations of **liability and governance**: who operates the ledger network? A consortium of hospitals or a trusted non-profit could host the nodes to distribute trust. If something goes wrong (say a bug in a smart contract), having a governance plan for patching and stakeholder communication is needed. Adoption also hinges on stakeholder incentives: researchers must perceive that the benefits (easier access once set up, and improved participant trust) outweigh any initial hurdles (like integrating their data pipelines with the gateway API). Participants must see value in using the platform – perhaps through a user-friendly app that not only asks consent but also provides them information about ongoing research, or even modest rewards or acknowledgments for their continued participation. Over time, as data scandals and public awareness of privacy increase, we expect that platforms offering stronger privacy guarantees will become a **competitive advantage** for research studies seeking volunteers, which will drive adoption.

Limitations and Future Directions: Not all scenarios of data use fit neatly into a dynamic consent framework. Situations such as **public health emergencies** or **law enforcement requests** may require data sharing without consent. Our system could accommodate such exceptions through special override roles: for example, a government authority could have a mechanism (duly logged and justified) to access data for a lawful requirement (like contact tracing in an epidemic) without participant consent. These would be clearly marked on the ledger as extraordinary events, and ideally still require some form of legal warrant or ethical approval recorded in the system. Another limitation is reaching populations that are not digitally connected – those without smartphones or reliable internet. Alternative interfaces (like paper or in-clinic digital kiosks) and **delegated consent** (allowing a clinician or legal representative to operate the consent engine on a participant’s behalf when appropriate) might be necessary to ensure inclusive participation. There is also the challenge of **data volume** and analysis: whole genome data is huge, and transferring it for each query might be inefficient. Future architectures might integrate **compute-to-data** models, where researchers submit analysis code that runs within the vault environment (under consent control) and only outputs aggregate results, thereby not exposing raw data at all. Our consent engine could then be extended to manage not just yes/no access, but also *what kind of analysis* is allowed (tying into the concept of *analysis consent*). Another future direction is integrating **AI and machine learning**: for example, using AI assistants to help users understand complex consent choices (“If you allow this, the researchers will do X; the risks are Y”), or even to help researchers identify which participants might be inclined to consent to a new study (without violating privacy). Any such AI integration must be done ethically

(avoiding manipulative targeting or bias). On the horizon, one can envision federated or distributed learning scenarios where the ledger and consent mechanism enable large-scale analysis without centralizing data – each user’s data stays in a secure enclave and models train across many enclaves with consent. This aligns with emerging privacy-preserving analytics. Finally, cross-border and cross-institutional data sharing can greatly benefit from this approach: a user could have one “dynamic consent profile” that many research initiatives honor, potentially via an international network of ledgers that exchange consent credentials. Achieving that will require not just technical work but governance agreements and standardization, as discussed.

In summary, while challenges exist in practical deployment, none appear insurmountable given the will to innovate in this direction. The architecture we propose exemplifies how careful design can align technological capabilities with ethical and legal imperatives. By acknowledging potential pitfalls (user fatigue, security risks, etc.) and addressing them through design choices and future work, the system can become a cornerstone of genomic data governance in the coming era, turning the often-stated slogan “My Data, My Decision” into a functional reality.

Conclusion

Biomedical research is increasingly data-driven and collaborative, yet it faces a fundamental tension between the desire to maximize data use and the obligation to respect individual rights. The Enigma Genetics platform described in this paper offers a comprehensive solution that bridges this divide, leveraging modern technology to enable both unprecedented data sharing **and** uncompromising data governance.

We have presented an academic blueprint for how a self-sovereign genomic data system can simultaneously achieve the empowerment of data subjects, the security and auditability demanded by regulators, and the flexibility and scale needed by researchers. By integrating a biometric identity anchor, a dynamic consent engine, an authorization gateway, and an immutable audit ledger, the proposed architecture ensures that every access to sensitive genomic data is purpose-specific, transparently logged, and under the control of the individual whose data is at stake. This stands in stark contrast to the status quo of broad one-time consents and siloed databases; it directly addresses long-standing calls in the bioethics and data governance literature for more participant-centric models of informed consent and stronger accountability in data use **[3,9]**.

Our design situates the Enigma Genetics approach within the context of prior work and prevailing standards. It shows consistency with recommendations of leading ethicists and technologists: facilitating ongoing communication and consent adjustments **[3,4]**, harnessing blockchain ledgers for consent verification and provenance **[5,6]**, and adhering to legal requirements like GDPR’s emphasis on specific, revocable consent and demonstrable accountability **[7]**. We have also detailed how the platform could be implemented in practice with currently available tools (mobile biometrics for identity verification, permissioned blockchain networks for logging, cloud encryption services for secure storage) and how it could be governed (through a consortium or

multi-stakeholder oversight) to ensure trust. Importantly, we find no insurmountable technical barriers to deployment: the building blocks (cryptography, distributed systems, user authentication methods) are all well-established. The key challenges likely lie in adoption, usability, and integration into existing research workflows. We have discussed strategies to tackle these, such as user experience design to prevent fatigue and standardization to ease interoperability.

Looking ahead, successful implementation of this platform could mark a paradigm shift in genomic data stewardship. It would demonstrate that respecting individual autonomy is not only compatible with large-scale science, but can actively enhance it by building public trust and encouraging more diverse participation. Participants who know they have granular control may be more willing to contribute data (including those from communities historically wary of research exploitation), improving the representativeness of genomic databases. Moreover, by automating compliance checks, the platform allows valuable data to be accessed more rapidly and with clearer usage rights, accelerating research progress within well-defined ethical guardrails. This aligns with emerging frameworks for **data sovereignty** gaining traction worldwide, wherein individuals are granted greater rights and technological means to control their personal data. In fact, while we have focused on genomics, the model of dynamic consent with immutable audits could serve as a template for any domain where sensitive personal data is used – from health records and biomedical imaging to smart city sensor data – ensuring that innovation does not come at the cost of individual rights.

For the academic and scientific community, our work contributes a detailed case study in operationalizing the often-theorized principles of dynamic consent and transparent data governance. We have provided concrete design patterns and reference points (such as the concept of expiring consent tokens and the use of permissioned blockchains for logging) that others can adopt or refine. The references and citations throughout underscore that this approach builds upon established research and addresses real concerns (like genomic re-identification risks [1,2] or difficulties in cross-institutional consent management). By aligning our design with initiatives like GA4GH and legal guidelines, we ensure that it can plug into and enhance ongoing efforts rather than exist in isolation. We invite collaboration across disciplines – computer science, genomics, bioethics, law – to pilot and further improve this platform.

Ultimately, the true measure of success for such a platform will be empirical. Key questions include: **Does it increase participation rates** in genetic research studies by alleviating privacy fears? **Do participants report greater satisfaction and understanding** of how their data is used? **Can researchers demonstrate faster access to datasets** or new research enabled by real-time consent that would not have been possible under static models? Answering these will require deploying the system in real-world settings (perhaps initially in a targeted project like a longitudinal cohort or a biobank) and iterating based on feedback. Issues that seem minor in design, such as the best way to notify users or the consequences of revoking consent on ongoing analyses, will benefit from practical experience and user input. We anticipate that initial deployments will yield lessons that drive further refinement – for example, simplifying certain processes or adding features like group consent for family genomic data.

In closing, the Enigma Genetics platform – through its combination of dynamic consent, secure identity, fine-grained access control, and auditability – embodies a forward-looking model for genomic data sharing. It shows that we need not accept a false trade-off between advancing science and protecting individual rights; with intelligent system design, we can achieve both. As genomic and health data generation accelerates (with millions of genomes likely to be sequenced in the coming decade), implementing such governance innovations becomes not only desirable but necessary to maintain public trust and maximize the societal benefits of precision medicine. We hope this work inspires further research, development, and cross-sector collaboration to realize a more ethical and effective data ecosystem. The era of genomic-driven health insights that we strive for must be built on a foundation of **trust, consent, and transparency** – and our proposed architecture contributes one blueprint for building that foundation. By empowering individuals as active stewards of their own “living code,” we can open new frontiers of biomedical discovery while upholding the dignity and rights of those who make such discovery possible.

References

1. Gymrek M., McGuire A. L., Golan D., Halperin E., & Erlich Y. (2013). **Identifying personal genomes by surname inference.** *Science* 339(6117): 321–324.
2. Erlich Y., & Narayanan A. (2014). **Routes for breaching and protecting genetic privacy.** *Nature Reviews Genetics* 15(6): 409–421.
3. Kaye J., Whitley E. A., Lund D., Morrison M., Teare H., & Melham K. (2015). **Dynamic consent: a patient interface for twenty-first century research networks.** *European Journal of Human Genetics* 23(2): 141–146.
4. Budin-Ljøsne I., et al. (2017). **Dynamic consent in the CHRIS study: improving informed consent for large-scale cohort research.** *European Journal of Human Genetics* 25(3): 447–452.
5. Albalwy M., et al. (2021). **ConsentChain: a blockchain-based dynamic consent architecture to support clinical genomic data sharing.** *JMIR Medical Informatics* 9(11): e28046.
6. Moreau L., & Groth P. (2013). **Provenance: an introduction to PROV.** *Synthesis Lectures on the Semantic Web: Theory and Technology* 3(4): 1–129.
7. European Parliament and Council. (2016). **Regulation (EU) 2016/679 – General Data Protection Regulation (GDPR).**
8. U.S. Department of Health and Human Services. (2013). **Modifications to the HIPAA Privacy, Security, Enforcement and Breach Notification Rules.** Federal Register Vol. 78, No. 17.
9. Global Alliance for Genomics and Health (GA4GH). (2014). **Framework for responsible sharing of genomic and health-related data.**
10. World Health Organization. (2024). **Guidance for human genome data collection, access, use and sharing.**