Sovereign Data Research Fabric (SDRF)

A Privacy-First Architecture for Human-Subject Research

Prepared by Patrick Deegan, PhD | Version 1.2 | February 17, 2026.

Abstract

The Sovereign Data Research Fabric (SDRF) is a privacy-first architecture for human-subject research. It replaces copy-and-store data flows with compute-to-data: participants keep encrypted vaults under their control, researchers send approved analyses to those vaults, and only privacy-protected outputs leave the system. SDRF unifies threshold cryptography, verifiable credentials, differential privacy, and decentralized governance into a coherent stack in which consent is enforceable by code and raw data never leaves participant control.

Modern studies in consciousness, mental health, aging, and other human-subject domains increasingly rely on rich streams of speech, biometrics, survey responses, and behavioral telemetry. These signals are precisely the data most difficult to collect under current regulatory frameworks. SDRF provides a self-sovereign, privacy-preserving platform that allows researchers to collect and analyze sensitive signals without removing data from participant control. The architecture is modular and decentralized, supporting deployments from pop-up clinics to multi-site longitudinal studies and enabling collaboration without a centralized honeypot of the most intimate data category humans produce.

SDRF defines a standard for a new class of participant-controlled wellness records that remain under participant control while still supporting rigorous research. It is not a clinical system by default, and any clinical use requires full regulatory compliance, institutional oversight, and IRB governance. The specification is written to be compatible with those requirements, and ongoing efforts are aimed at meeting them where clinical deployment is appropriate.

> *Exploding multimodal data and stricter privacy laws have made copy-and-store models unsustainable. Communities urgently need a compute-to-data architecture that unlocks insight without surrendering ownership.*

I. The Broken Promise of Research Data Protection

Every participant in a research study makes an act of faith. They provide information about their bodies, their behaviors, their thoughts, their genomes, trusting that this information will be used as promised and protected from misuse. In return, they receive assurances: consent forms detailing how data will be handled, institutional review board approvals certifying ethical oversight, and privacy policies describing security measures. These assurances are sincere. They are also increasingly inadequate.

The inadequacy stems from a fundamental mismatch between the nature of digital information and the tools we use to govern it. Data, once copied, exists independently of its original context. It can be combined with other data to reveal information never explicitly shared. It persists across time in ways that paper records never could. The consent given in 2026 may prove insufficient for the analytical capabilities of 2036, and anonymization techniques considered robust today are routinely defeated by tomorrow's linking algorithms. The problem is not bad intent; it is a brittle default architecture built for a different technological era.

The Arithmetic of Re-identification

Consider the mathematics of uniqueness. Research published in *Scientific Reports*[1] demonstrated that four spatiotemporal points (four locations with timestamps) are sufficient to uniquely identify

[1] de Montjoye, Y. A., Hidalgo, C. A., Verleysen, M., & Blondel, V. D. (2013). Unique in the Crowd: The privacy bounds of human mobility. *Scientific Reports*, *3*, 1376. https://doi.org/10.1038/srep01376

the vast majority of individuals in a mobility dataset. The intuition is straightforward: while any single data point may be shared by many people, the intersection of multiple points rapidly narrows to a single individual. Your home location, your work location, a restaurant you visited last Tuesday, and your gym on Saturday morning together form a fingerprint that almost certainly belongs only to you.

This principle extends across data modalities. Research on EEG-based biometric identification[2,3] showed that brief segments of electroencephalogram (EEG) recordings identify individuals with high accuracy based on the characteristic patterns of their brain activity. These patterns are not deliberately identifying information; they are simply the natural variation in how different brains process stimuli, but they function as biometric signatures nonetheless.

Voice recordings present similar challenges. The fundamental frequency of speech, the formant structure of vowels, and the timing patterns of syllables all vary between individuals in ways that enable identification.[4,5] A voice diary intended to capture emotional content simultaneously captures a biometric identifier. The researcher may be interested only in what words were spoken and with what affect; the recording contains far more.

Genomic data represents the limiting case. DNA sequences are, by definition, identifying. They identify not only the individual from whom the sample was taken but also their biological relatives, across generations.[6,7,8] The privacy implications of genomic data extend beyond the participant who consented to include family members who may not have consented, may not be aware of the research, and may have strong preferences about whether their genetic information is exposed.

The Consent Paradox

These re-identification risks create a fundamental paradox in research consent. Traditional consent assumes that participants can meaningfully evaluate what they are agreeing to. But future uses of data, future analytical techniques, and future datasets that might be combined with current data are unknowable at the time consent is requested. Consent becomes a promise about a future we cannot fully describe. Participants are asked to authorize uses they cannot fully understand against risks they cannot fully evaluate.

Institutional review boards attempt to manage this uncertainty through periodic review and ongoing oversight. But the mechanisms are administrative rather than technical. They depend on researchers accurately reporting their data uses, on institutions maintaining awareness of what data exists and how it is being used, and on the assumption that unauthorized uses will come to light. As data volumes grow and analytical capabilities expand, these administrative controls become increasingly inadequate.

The result is a system where participants bear risks they did not fully consent to, researchers operate

[2]Wang, M., Abbass, H. A., & Hu, J. (2020). BrainPrint: EEG biometric identification based on analyzing brain connectivity graphs. *Pattern Recognition*, *105*, 107381. https://www.sciencedirect.com/science/article/abs/pii/S0031320320301849

[3]Su, F., Xia, L., Cai, A., Wu, Y., & Ma, J. (2010). EEG-based personal identification: From proof-of-concept to a practical system. In *2010 20th International Conference on Pattern Recognition* (pp. 3728-3731). IEEE. DOI: 10.1109/ICPR.2010.908. https://doi.org/10.1109/ICPR.2010.908

[4]Jankowski, C. R., Quatieri, T. F., & Reynolds, D. A. (1995). Measuring fine structure in speech: Application to speaker identification. In *1995 International Conference on Acoustics, Speech, and Signal Processing* (Vol. 1, pp. 325-328). IEEE.

[5]Gelfer, M. P., & Mikos, V. A. (2005). The relative contributions of speaking fundamental frequency and formant frequencies to gender identification based on isolated vowels. *Journal of Voice*, *19*(4), 544-554. https://pubmed.ncbi.nlm.nih.gov/16301101/

[6]Gymrek, M., McGuire, A. L., Golan, D., Halperin, E., & Erlich, Y. (2013). Identifying personal genomes by surname inference. *Science*, *339*(6117), 321-324. https://pubmed.ncbi.nlm.nih.gov/23329047/

[7]Humbert, M., Ayday, E., Hubaux, J. P., & Telenti, A. (2013). Addressing the concerns of the Lacks family: Quantification of kin genomic privacy. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security* (pp. 1141-1152). https://infoscience.epfl.ch/entities/publication/87c98700-b805-43df-904b-84ad995a65c4

[8]Erlich, Y., Shor, T., Pe'er, I., & Carmi, S. (2018). Identity inference of genomic data using long-range familial searches. *Science*, *362*(6415), 690-694. https://pmc.ncbi.nlm.nih.gov/articles/PMC7549546/

under constraints that may not match actual risk levels, and institutions maintain compliance theater that satisfies auditors without genuinely protecting privacy. The gap is widening because the data is getting richer faster than the policy apparatus that governs it.

The Privacy Ceiling

The increasing complexity and richness of data (EEG, speech, heart rate variability, behavioral telemetry) collide with compliance barriers. The General Data Protection Regulation (GDPR) in Europe and the Health Insurance Portability and Accountability Act (HIPAA) in the United States, alongside emerging frameworks worldwide, create legitimate protections that, under current infrastructure, function as ceilings on research ambition. They were designed to constrain unsafe data movement, but they end up constraining science itself when the underlying architecture requires copying and centralizing.

Cross-site sharing becomes legally hazardous. A researcher at one institution who wishes to collaborate with a colleague at another institution must navigate data use agreements that can take months to negotiate. Each institution's legal team must satisfy itself that the data transfer complies with applicable regulations, that adequate security measures are in place, and that participant consent covers the proposed uses. The cost is delay, and the effect is lost momentum in fields where timing matters.

Longitudinal studies face consent decay. A study designed to follow participants over decades confronts the problem that consent given in year one may not cover analyses conceived in year fifteen. Re-contacting participants for updated consent is expensive, introduces selection bias, and may be impossible if participants have moved, changed contact information, or died. The consequence is either stalled research or overbroad initial consent that undermines participant autonomy.

Community-based participatory research struggles with individual-focused consent models that ignore collective interests. Indigenous communities, patient advocacy groups, and other organized populations may have legitimate collective interests in how their members' data is used. Traditional consent frameworks treat each individual as an isolated decision-maker, ignoring the social context in which data about communities carries implications for the community as a whole.

The Costs of the Current Approach

These constraints impose concrete costs on research. Compliance activities consume substantial fractions of research budgets at major academic medical centers, diverting resources that could otherwise fund actual research. Data use agreements between institutions routinely require many months to negotiate, delaying scientific progress while lawyers exchange drafts. The vast majority of clinical trial data is never shared beyond the original study, representing an enormous waste of participant contribution and scientific potential.

The costs fall disproportionately on certain types of research. Studies requiring data from multiple institutions face multiplicative compliance burdens. Research involving sensitive populations, such as those with mental health conditions, those in marginalized communities, or those whose data carries particular re-identification risk, faces heightened barriers that may render studies infeasible. Innovative research designs that would combine data across traditional silos become practically impossible even when scientifically valuable and ethically appropriate.

II. From Copy-and-Store to Compute-to-Data

The conventional approach to research data is simple and, under current conditions, broken. Participants provide data. That data is copied to institutional servers. Institutions promise to protect it according to policies. Researchers access the data through institutional controls. When things go wrong (breaches, re-identification, unauthorized use), participants learn about it after the fact, if at all.

This model made sense when data was scarce, computational resources were centralized, and the analytical techniques that could be applied to data were limited. In that world, copying data to a central location where researchers could access it was the only practical approach. The risks of that centralization were manageable because the data volumes were small and the re-identification techniques were primitive.

That world no longer exists. Data is abundant. Computational resources are distributed. Analytical techniques, particularly machine learning approaches, can extract information from data that would not have been conceivable a generation ago. The copy-and-store model persists not because it remains appropriate but because infrastructure built on other assumptions has not been available.

The Inversion

SDRF inverts this model entirely. Data remains in participant-controlled vaults. Researchers send computation to the data, not data to researchers. Only privacy-protected results (anonymized statistics, differentially private aggregates, trained model weights) ever leave participant control.

The difference is not merely architectural; it is philosophical. Under the copy-and-store model, participants are data sources to be managed. They provide their information and then hope that institutions will behave responsibly with it. Their control ends at the moment of data transfer. Whatever rights they retain are legal rather than practical. They can sue after a breach, but they cannot prevent one.

Under SDRF, participants are rights-holders exercising sovereignty over their information. They grant access cryptographically, specifying exactly what operations are permitted on their data, by whom, for what purposes, and for how long. Their control is continuous: they can revoke access at any time, and that revocation is enforced by the system itself, not by institutional compliance. The question is not whether institutions will honor their promises but whether the cryptographic keys exist that would enable a particular operation. This shifts the burden of trust from institutions to verifiable protocol behavior.

Why This Inversion Is Now Possible

Several technological developments have converged to make this inversion practical. Threshold cryptography enables access control without centralized key management: decryption keys can be split across multiple parties such that no single party (including the system operator) can decrypt data without appropriate authorization, eliminating the trusted central party that traditional encryption schemes require. Verifiable computation enables researchers to run analyses without seeing raw data; computation can occur in environments where the data is protected from the entity running the computation. Differential privacy provides rigorous guarantees about what any adversary can learn from released results, regardless of auxiliary information or computational power. Decentralized identity standards from the World Wide Web Consortium (W3C), including Decentralized Identifiers (DIDs) and Verifiable Credentials, enable participant-controlled credentials that are portable, privacy-preserving, and cryptographically verifiable.

Key technical terms and standards referenced in this paper are defined in the Glossary.

None of these technologies is hypothetical; each has a mature body of literature and working implementations. What has been missing is not the components but the architecture that brings them together into a coherent system. SDRF provides that architecture by specifying a consent system, a compute environment, and a key management model that work as one.

The Transformation

The shift from copy-and-store to compute-to-data transforms the power dynamic between participants and researchers. Consider key contrasts:

Conventional Approach	SDRF Compute-to-Data
Copy-and-store: raw personal health information and biosignals are transferred to a central cloud, forcing participants to relinquish ownership.	Compute-to-data: data remains in participant-controlled vaults. Only encrypted features or differentially private results leave the vault. Threshold cryptography allows computations without exposing raw data.
Static consent: consent is captured as a PDF. Once data is moved, it can be reused indefinitely with limited participant recourse.	Dynamic, machine-readable consent: consent is expressed as a structured manifest specifying rights, obligations, and conditions. Consent can be modified or revoked at any time, immediately halting access.
Privacy tools deployed piecemeal: differential privacy, multiparty computation, and secure enclaves are often ad hoc, available only in well-funded labs.	Integrated privacy stack: differential privacy is applied automatically when data is shared, with clear reporting of what protections were applied.
Silos between domains: wellness, clinical, and research systems operate independently, creating fragmentation.	End-to-end policy fabric: a unified manifest schema handles data management, consent enforcement, and governance across all contexts.
Expensive compliance teams: labs rely on costly specialists to ensure consent, security, and privacy.	Automated compliance: the system enforces minimum-necessary principles, audit logging, and dynamic revocation through protocol-level constraints.
Offline limitations: data access typically requires centralized servers; participants must be online to share.	Offline-capable: threshold cryptography enables secure computations even when participant devices are temporarily offline, using pre-authorized keys.

This transformation is not an incremental improvement. It is a paradigm shift. Under the conventional model, institutions hold data and promise to behave responsibly. Under SDRF, participants hold data and grant access cryptographically. The difference is between asking for trust and providing guarantees.

III. Design Principles

SDRF rests on core principles that guide every architectural decision. These are not aspirational slogans; they are engineering constraints encoded into how the system is built and operated.

Participant Sovereignty. Individuals control their own data through cryptographic key possession, not through policy promises. Access requires both technical capability (correct decryption keys) and authorization (valid consent credentials). This dual requirement ensures that control is not merely legal but practical. It also enables revocation that is enforced by the system itself rather than by institutional goodwill.

Open Standards. Every component uses or extends existing open standards. Identity follows World Wide Web Consortium (W3C) Decentralized Identifiers. Consent credentials follow W3C Verifiable Credentials. Data formats can comply with HL7 Fast Healthcare Interoperability Resources (FHIR) and domain-specific open schemas where they exist. Cryptographic primitives follow established standards. No proprietary lock-in exists at any layer; implementations can be swapped without breaking interoperability or invalidating previously collected data.

Modular Architecture. SDRF specifies interfaces, not implementations. Storage can use decentralized networks, institutional infrastructure, or participant-controlled cloud. Compute can run in

secure enclaves, trusted execution environments, or purpose-built isolation. Governance can range from simple operator control to full decentralized autonomous organization (DAO) structures. Identity can integrate with existing systems or operate purely self-sovereign. This modularity enables diverse deployments without compromising interoperability or security guarantees.

Privacy by Default. Data is encrypted at capture, stored encrypted, transmitted encrypted, and processed in isolation. Privacy-preserving computation prevents data exfiltration. Differential privacy provides mathematical guarantees. These protections are not optional features but architectural requirements; the system cannot be used in ways that compromise privacy. When utility and privacy conflict, the system is designed to fail closed rather than fail open.

Economic Sustainability. Research infrastructure must survive beyond grant funding. Institutional budgets are constrained. SDRF enables multiple economic models: research-funded (studies prepay participant storage costs), participant-funded (individuals pay for their own vaults), cooperative (pooled funding from participant cohorts), and hybrid approaches. This flexibility enables diverse communities to adopt SDRF on their own terms without dependence on continuous external support.

Offline Capability. Participants cannot be online continuously. Threshold cryptography and pre-authorized computation enable secure operations even when participant devices are offline. Consent manifests pre-authorize specific operations; the system enforces these authorizations without requiring real-time participant interaction. Offline capability is not a convenience feature; it is required for practical field research and for participants who cannot be reliably online.

Deployment Flexibility. Participants can run SDRF infrastructure entirely on their own premises, whether on institutional servers, personal hardware, or portable media. A participant's encrypted vault and its associated computation environment can be instantiated on a USB drive, creating an air-gapped research platform that operates offline and never exposes raw data to network transmission. Because computation moves to the data rather than data to computation, the system can function as a self-contained unit. The only network requirement is for transmitting privacy-protected outputs and coordinating multi-party computation protocols; sensitive analysis happens locally, under the participant's physical control. For institutions bound by HIPAA, GDPR, or strict data sovereignty rules, this flexibility can be the difference between participation and abstention.

Verifiable Operation. All critical operations produce cryptographic proofs of correct execution. Participants can verify that their data was accessed only according to their consent. Researchers can verify that analyses produced the claimed results. Governance decisions can be verified against stated rules. This verifiability enables trust without requiring blind faith in system operators.

IV. System Architecture Overview

SDRF consists of six primary layers, each with specific responsibilities. The identity layer manages participant identities and authentication through self-sovereign identifiers, cryptographic key management, and recovery mechanisms. The consent layer stores and enforces consent manifests, providing machine-readable specifications of what data can be accessed, by whom, for what purposes, and under what conditions. The storage layer holds encrypted participant data in personal vaults and supports multi-tenancy, tiered storage, replication, and metadata indexing. The cryptography layer manages encryption keys using threshold schemes, derives keys on demand based on consent manifests, and implements revocation through key rotation and manifest updates. The compute layer executes analysis code on participant data within isolated environments, applies privacy protections, checks outputs for disclosure risks, and enforces resource limits. The governance layer coordinates protocol-level decisions about researcher authorization, analysis approval, dispute resolution, protocol upgrades, and incident response, spanning operator-controlled, committee-based, and decentralized autonomous organization (DAO) models.

These layers interact through well-defined interfaces. A consent manifest governs cryptography and compute; cryptography governs storage access; storage provides encrypted data and metadata; compute applies policy and releases outputs; governance authorizes who may submit analyses and how

the system evolves. Different deployments can use different technologies for each layer while maintaining compatibility at the protocol level. Interoperability across deployments enables federated research without centralized data aggregation.

A core architectural objective is separation of concerns. No layer is assumed to be trusted in isolation. If storage is compromised, encryption and threshold keying prevent decryption. If compute is compromised, privacy filters and output checks limit exposure. If a governance operator is compromised, verifiable audit trails make unauthorized actions detectable and reversible. The system is designed so that the failure of one layer does not collapse the entire privacy posture.

V. The Participant Journey

To better understand SDRF, consider a participant's experience from initial enrollment through ongoing contribution. Each step reflects the design principles: participants remain in control, privacy protection is automatic, and the cognitive burden on participants is minimal despite sophisticated technology operating beneath the surface.

Enrollment and Identity Creation

The journey begins when a participant encounters an SDRF-enabled study, whether at a research clinic, a community health fair, a festival wellness tent, or through a link shared by their healthcare provider. The participant scans a QR code with their smartphone, which launches an enrollment application.

Within this application, cryptographic operations generate a decentralized identifier (DID) and associated key material. A DID is a globally unique identifier that the participant controls. Unlike a username issued by a platform or a medical record number assigned by an institution, a DID belongs to the participant and can be used across any system that supports the standard.

The key material associated with the DID consists of one or more cryptographic key pairs. The private keys remain on the participant's device; only the public keys and the DID itself are shared with other systems. This asymmetry is fundamental: anyone can verify that a message was signed by a particular DID, but only the holder of the private key can produce such a signature. The system is designed so that participants can authenticate without revealing unnecessary identity attributes.

Key Recovery and Device Portability

Key recovery is a central challenge for self-sovereign systems: if a participant loses their device, how do they regain access without introducing a centralized recovery authority that undermines the architecture's premise? SDRF relies on threshold recovery patterns derived from established cryptographic practice and secret sharing.

When participants create their identity, their master key can be protected through complementary recovery methods. Offline recovery uses physical or removable digital media, such as a recovery key printed as a QR code, stored on a hardware security module, or written as a mnemonic phrase. The participant maintains complete physical custody; no third party holds recovery capability. Social recovery employs threshold secret sharing, splitting the master key into shares distributed among trustees chosen by the participant. A 2-of-3 configuration generates three shares where any two can reconstruct the key, but possession of only one share reveals mathematically zero information about the original key.

These mechanisms are not mutually exclusive. A participant concerned about both physical loss and social unavailability might use both: offline backup for scenarios where trustees are unreachable and social recovery for scenarios where physical media is destroyed. The choice maps directly to risk tolerance and context.

Device portability follows from the same infrastructure. Modern implementations using DIDs establish a key hierarchy: the master key, rarely accessed and heavily protected through recovery

mechanisms, authorizes device-specific keys that handle routine operations. When acquiring a new device, the participant uses a recovery mechanism to reconstitute the master key and generates fresh device keys authorized by that master. Device keys can be revoked individually without touching the master key, enabling selective compromise recovery. If a phone is stolen, that device's keys are revoked while other authorized devices continue functioning. This preserves continuity without undermining the security model.

Ongoing Participation and Transparency

After enrollment, participants interact primarily with a consent dashboard and data log. The dashboard provides clear visibility into what data has been collected, which studies are authorized, and what analyses have been run. Privacy budgets and data usage are presented in plain language with drill-down for those who want details. Participants can pause data collection, narrow permissions, or revoke consent entirely with immediate effect in the system.

This transparency is not cosmetic. It provides practical control and makes consent meaningful over time. It also creates a feedback loop: when participants can see what their data is being used for, they can make more informed decisions about continued participation and can shape the research agenda through their choices.

VI. The Consent Manifest

SDRF introduces the consent manifest: a structured, machine-readable document that specifies exactly what data will be collected, how it will be processed, who can access it, for what purposes, and for how long. The manifest is the single source of truth for all parties. The same document that a participant reviews is the same document that governs system behavior.

Manifest Structure

A consent manifest contains several categories of specification. Data classes define what types of information will be collected, with both a human-readable description and a machine-readable specification. Processing operations define what computations are permitted, from simple aggregations to complex analyses, and include both descriptions and formal specifications the compute environment can enforce. Access conditions define who can request what operations and under what circumstances, referencing properties of the requester, the analysis, or time-bound conditions. Duration and renewal specify how long the authorization lasts and how it can be extended. Revocation terms specify how participants can withdraw and what happens to their data on withdrawal, including whether ongoing analyses must stop and whether data is deleted or retained in inaccessible form. Compensation terms specify if and how value flows back to participants, whether via direct payment, token grants, profit-sharing, or acknowledgment in publications. Privacy parameters specify budgets, noise levels, aggregation thresholds, and other technical settings that govern the privacy-utility tradeoff.

The manifest is versioned and signed by the participant. Each version is immutable once signed, ensuring that changes are explicit rather than retroactive. The system can therefore audit which data access decisions were made under which consent version, enabling precise accountability.

Human-Readable and Machine-Executable

The manifest must serve two audiences simultaneously. For participants, it must be comprehensible; they need to understand what they are agreeing to. For systems, it must be executable; the compute environment must determine, for any requested operation, whether that operation is authorized by the manifest.

This dual requirement shapes the manifest format. Each element has both a human-readable description and a formal specification. Descriptions use plain language and can be presented through

text summaries, visual diagrams, or interactive explorers. Specifications use a precise schema that admits no ambiguity about what is and is not authorized. The schema is designed so that the formal specification fully determines authorization decisions; there is no interpretation step that depends on human judgment at access time. This turns consent from a policy promise into an enforceable protocol rule.

The Complexity of Consent Governance

The complexity of consent governance, not technology, is the real challenge in making SDRF work. Traditional legal agreements are written for lawyers and courts, not for automated enforcement. They rely on intentionally vague language that allows flexibility in interpretation. SDRF requires that consent be expressed in terms machines can interpret and execute.

This is a significant shift in how we think about agreements. It forces consent framers to specify, in advance, exactly what operations they wish to authorize, with sufficient precision that a computer can make authorization decisions without human intervention. The benefit is enforceability. A traditional consent form might say "your data will be used for research purposes"; a SDRF manifest specifies exactly which operations by exactly which parties are authorized, and the system permits or denies each request with no ambiguity.

Consent Evolution

Participants' preferences may change over time, and studies may need to request expanded permissions as research directions evolve. The manifest system handles both cases. When a participant wishes to modify their consent, they create and sign a new manifest that supersedes the previous one. The new manifest takes effect immediately; systems query the current manifest when evaluating authorization requests, so changes propagate without requiring explicit notification of all interested parties.

When a study wishes to request expanded permissions, it proposes a manifest amendment to affected participants. Participants can accept the amendment (producing a new manifest with expanded permissions), reject it (retaining current permissions), or withdraw entirely. The system tracks which participants are operating under which manifest version, enabling studies to work with participants who have different permission levels. This flexibility addresses the consent decay problem that plagues longitudinal studies and enables narrow, informed consent to evolve as research questions mature.

Consent Enforcement and Conflict Resolution

Consent enforcement is deterministic. The system evaluates requests against the manifest schema, and if any required condition fails, the request is denied. This eliminates discretionary interpretation at the moment of access. Where policies conflict (for example, when a study seeks data that overlaps multiple manifests), the most restrictive applicable policy applies by default. This is the system equivalent of "least privilege" and ensures that ambiguity resolves in favor of the participant.

VII. Data Collection Infrastructure

With identity established and consent expressed, data collection begins. The participant's smartphone and any paired wearables, such as EEG headbands, heart rate monitors, continuous glucose monitors, or smart scales, record data according to study protocols. Voice diaries capture speech. Surveys collect self-reports. Activity data accumulates continuously. Whenever possible, data is encrypted on the device before being stored or transmitted. Keys are derived from the participant's identity so that only authorized parties can ever decrypt the data.

Data collection also includes quality control. Devices can attach calibration metadata, sampling rates, and sensor state to each record. Applications can collect minimal contextual metadata, such

as device model or firmware version, to enable downstream analysis without revealing unnecessary personal context. When protocols require participant input, the system can record both the data and the protocol state so that missingness is explicit rather than inferred.

Data Formats and Standards

SDRF specifies standard data formats for common research data types while maintaining extensibility for novel modalities. For health data interoperability, SDRF uses HL7 Fast Healthcare Interoperability Resources (FHIR) to represent observations, procedures, medications, and conditions. Physiological signals rely on established formats such as the European Data Format (EDF) and its EDF+ extension for EEG and polysomnography. Synchronized multi-stream time series collected via the Lab Streaming Layer (LSL) are stored in the eXtensible Data Format (XDF). Activity and wellness data commonly use the Flexible and Interoperable Data Transfer (FIT) format when supported by device manufacturers; device-specific metrics can be captured in documented JSON schemas. Voice and audio data are stored in standard compressed formats, with lossless compression preferred when feasible, and accompanied by metadata that records timestamps, device identifiers, calibration details, acoustic context when consented, and any processing applied. Survey and self-report data follow structured JSON schemas linked to question definitions, validation rules, and completion metadata.

The guiding principle is openness. Proprietary encodings create lock-in and complicate long-term preservation. If data remains encrypted for decades, the format must remain interpretable when decrypted. Deployments that introduce custom formats must provide complete specifications and reference implementations. Schema versioning is explicit so that analyses can align data across long-running studies without silent incompatibilities.

Wearable Device Integration

SDRF is agnostic about how data enters the system. Different deployments can use different integration strategies based on their requirements and resources. Many wearable manufacturers provide developer APIs for accessing user data with OAuth 2.0-based consent flows. Fitness trackers, smart rings, and smartwatches typically expose data on activity, sleep, heart rate, and other metrics through these APIs. Operating system health platforms such as Apple HealthKit or Google Fit provide unified access to data from multiple sources and enforce granular permissions through native SDKs.

The integration pattern is consistent. The research application authenticates with the device manufacturer or platform, the participant grants permission for specific data types, the application retrieves new data on a schedule or via webhooks, and the data is encrypted client-side before transmission to the vault. This ensures that encryption keys never leave participant-controlled devices and data is never transmitted in cleartext. Rate limits and vendor-specific policies are handled by the integration layer so researchers can reason about coverage and latency.

Research Devices and Lab Streaming Layer

For research-grade devices and specialized sensors, the Lab Streaming Layer (LSL) protocol provides standardized real-time data streaming. LSL is widely used in neuroscience and psychophysiology research and provides unified data formatting, timestamp synchronization for multi-modal recording, network streaming with automatic discovery, and robust error handling with buffering.

Consumer EEG headbands suitable for research applications can stream data over LSL using open-source tools. OpenBCI systems provide documented APIs and raw EEG streaming via the BrainFlow library with multi-language support. Physiological sensors such as heart rate monitors, respiration belts, and galvanic skin response devices can also stream through LSL with appropriate adapters. SDRF deployments typically run LSL receivers on participant devices or local collection hardware, encrypt incoming streams in real time, and forward them to participant vaults. Timestamp alignment is preserved end-to-end so that cross-modal analysis is feasible without manual reconciliation.

For recorded sessions, LSL tooling can save synchronized data streams into the XDF container format, preserving timing information and rich metadata across modalities. This enables later analysis without requiring that all streams be sampled at the same rate or even captured on the same device.

Secure Sync to Personal Vault

When network connectivity is available, encrypted data synchronizes to the participant's personal vault, a persistent encrypted store that accumulates research contributions over time. The vault is the participant's long-term archive, designed to be portable, durable, and independently verifiable. Sync protocols support resumable uploads, integrity checks, and backpressure so that data capture is robust even under unreliable network conditions.

Vault Implementation Options

Vaults can be implemented using different technologies depending on deployment constraints. The reference implementation uses the Internet Computer (ICP), a decentralized network for running canister smart contracts, which provide strong isolation guarantees, deterministic execution, and replication across a subnet. Deployments that must use existing institutional infrastructure can store vaults as encrypted containers on standard cloud storage, with the assurance that storage providers cannot read the data and decryption keys are never exposed. Field deployments with limited connectivity can run local vaults with periodic synchronization to durable storage when connectivity permits. The implementation choice does not change the security model; it only changes operational tradeoffs.

VIII. Technical Architecture

SDRF's layered model supports interoperability and allows deployments to select technologies appropriate to their risk profiles. The following sections describe architectural considerations for the core layers and the operational guarantees they provide.

Identity Layer: Architecture

Participant identity is anchored in decentralized identifiers (DIDs) conforming to W3C standards. A DID resolves to a DID Document containing public keys and service endpoints. The critical property is that the participant controls the private keys corresponding to the public keys in their DID Document; no central authority issues or can revoke the identifier.

The DID format enables verification without contacting a central registry. Given a DID and a signature, any party can resolve the DID Document, extract the public key, and verify the signature. This verification works offline, requires no account with any service, and produces a cryptographic proof that the signature was produced by the holder of that DID. This property is crucial for research contexts where connectivity and central dependencies are unreliable.

Beyond basic identity, participants accumulate credentials that attest to properties such as consent to a study or eligibility criteria. These credentials follow the W3C Verifiable Credentials standard. A verifiable credential consists of claims about a subject, signed by an issuer, and can be verified without contacting the issuer at the time of verification. Credentials can be selectively disclosed so participants reveal only what a verifier requires. This minimizes data exposure while preserving trust in the claims.

Consent manifests are a special type of credential: they are claims about what operations the participant authorizes, signed by the participant. When a researcher requests access to participant data, the system retrieves the participant's current manifest and evaluates whether the request falls within the authorized scope. The evaluation is deterministic: given the manifest and the request, the answer is unambiguously yes or no, and the decision is auditable after the fact.

Storage Layer: Architecture

The storage layer holds participant data in encrypted vaults. Its design must satisfy competing requirements: data must be durable, available, confidential, and efficient.

All data is encrypted at rest using authenticated encryption schemes that provide confidentiality and integrity. Encryption keys are never stored in cleartext alongside the data they protect. Instead, data encryption keys are derived on demand through the cryptography layer. If the storage layer is compromised, an attacker obtains only ciphertext.

Each data object is encrypted with a unique key derived from the participant's identity and an object-specific nonce. This isolation ensures that compromise of one object's key does not enable decryption of other objects. Within a vault, data is organized by type and time, enabling efficient retrieval by modality and period. Metadata about data types, time ranges, and format versions is stored separately and encrypted, but can be decrypted for authorized queries without decrypting the underlying data.

Metadata schemas are standardized across SDRF deployments. A researcher writing an analysis knows what metadata fields will be available and can write queries that work across participants, studies, and deployments. This standardization is essential for federated analyses that span multiple institutions.

Vaults are replicated across multiple storage nodes to ensure durability. Replication strategies depend on deployment, but durability guarantees are explicit and verifiable. On the Internet Computer (ICP), canister smart contracts replicate across nodes within a subnet; institutional deployments can use standard cloud replication. Writes complete only when replication thresholds are met, and reads return consistent data regardless of which replica serves the request. Durability guarantees are quantified and communicated so studies can choose deployments appropriate to their risk tolerance.

Cryptography Layer: Architecture

The cryptography layer manages the keys that protect participant data. Its core innovation is threshold cryptography: the ability to perform cryptographic operations, particularly decryption, without any single party holding complete key material.

Threshold schemes split key material across multiple parties so that any k-of-n shares can cooperate to perform cryptographic operations while fewer than k shares reveal nothing about the key. SDRF uses this approach to ensure that no single party, including storage providers or system operators, can unilaterally decrypt participant data. Decryption requires cooperation among multiple parties, all of whom verify that the request is authorized.

The reference implementation uses the Internet Computer's vetKeys system (verifiably encrypted threshold keys) to derive identity-bound decryption keys on demand. When data is encrypted, it is bound to a derivation path specifying who should be able to decrypt it, such as a participant, a specific study role, or an emergency responder in a defined crisis scenario. The corresponding key does not exist in any single location and is computed only when a valid request arrives. There is no key escrow that can be subpoenaed and no key database that can be stolen.

Threshold cryptography also enables offline access. The participant need not be online at the moment their data is accessed. The consent manifest specifies what access is authorized, and the threshold system can verify this authorization and derive the necessary keys without real-time participant involvement. Revocation is enforced through key rotation: when consent changes, the system denies key derivations for requests that fail authorization checks against the current manifest. In the reference implementation, revocation is effective immediately without cache invalidation delays.

Compute Layer: Architecture

The compute layer executes analysis code on participant data and must prevent data exfiltration while enabling legitimate research. Analyses run in controlled environments that limit what code

can do. The baseline implementation uses canister smart contracts on the Internet Computer (ICP) with network isolation, memory isolation, and resource limits; canisters cannot make arbitrary network connections and can only communicate through defined interfaces with authorized endpoints. Deployments that require stronger isolation can use hardware-based secure enclaves such as Intel Software Guard Extensions (Intel SGX), AMD Secure Encrypted Virtualization (SEV), or AWS Nitro Enclaves.

Before execution, the compute layer verifies authorization using an analysis manifest, a structured document describing the computation to be performed. The manifest specifies the required inputs, the code or template to execute, the outputs and their formats, the privacy protections to apply, and resource limits. The compute layer validates that all referenced participants have consented, the code specification is valid, output requirements are consistent with the code, and privacy specifications meet minimum requirements.

SDRF provides pre-audited analysis templates for common operations such as descriptive statistics, regressions, and standard model training. Templates are parameterized but fixed in code, reducing the attack surface and accelerating approvals. When researchers need analyses beyond the template library, they submit custom code for review. The review process examines code for security concerns, produces a signed attestation of approval, and creates an audit trail. This produces a persistent record of what code has been authorized and under what conditions.

Output checking is the final line of defense. Outputs are evaluated against statistical disclosure control rules developed in secure research enclaves, including suppression of small cell counts, truncation of extreme values, and detection of output combinations that could enable inference. The UK's Office for National Statistics, the US Census Bureau, and numerous academic secure research facilities have refined these procedures over decades, and SDRF incorporates these principles into its automated and human review workflows. Automated checks handle routine cases; complex or ambiguous outputs are flagged for human review.

Differential privacy is integrated at the output stage. When an analysis produces results, calibrated noise can be added before release. Privacy budgets are tracked per participant, and each analysis consumes part of the available budget. Participants can see how their budget has been used, enabling informed decisions about whether to authorize further analyses. For releases intended for public dissemination, conservative privacy parameters are recommended; for restricted internal use, higher utility may be acceptable within explicit participant consent.

Interoperability and Migration

SDRF is designed to coexist with existing research infrastructure. Data can be imported from legacy systems into participant vaults when consent allows, preserving provenance metadata and original formats. Existing analysis pipelines can be adapted to run within SDRF by containerizing workloads and enforcing manifest-based access controls. This reduces the barrier to adoption while ensuring that new data collection conforms to privacy-first defaults.

IX. Federated Learning

Machine learning on sensitive data presents a particular challenge. Traditional ML requires centralizing training data, exactly what SDRF is designed to avoid. Federated learning offers an alternative: training models on distributed data without centralizing it.

In federated learning, model training is distributed across data holders. Instead of sending data to a central server, each participant trains on local data and sends only model updates (gradients or weight changes) to an aggregation service. The service combines updates from many participants to improve the global model and distributes the improved model for the next round. Raw data never leaves participant control, and model updates can be privatized before aggregation.

Federated learning in SDRF is governed by training manifests that specify the model architecture, training protocol, aggregation method, privacy guarantees, and participation requirements. A typ-

ical training round proceeds in phases. Model weights are distributed to authorized participants. Local training runs on participant data for a defined number of epochs. Gradients are computed and privatized with differential privacy noise. Privatized gradients are aggregated under threshold control, ensuring no single party sees all updates. The global model is updated for the next round and redistributed.

At completion, the trained model can be released with documentation of the total privacy budget consumed, the mechanisms used, and the effective privacy parameters. For public release, conservative privacy settings are recommended; for restricted use, higher budgets may be acceptable in exchange for model quality. This documentation enables downstream users to understand the privacy properties of the model and its appropriate use cases.

X. Governance Architecture

Technology alone cannot ensure that a data infrastructure serves the interests it claims to serve. The most elegant cryptographic protocols can be subverted by governance failures: capture by narrow interests, drift from founding principles, and opacity in decision-making. SDRF addresses these risks through a governance architecture designed to be lightweight enough for practical deployment while providing meaningful accountability.

The Fiduciary Foundation

A fiduciary is one who holds power in trust for another. The concept, nearly a millennium old in common law, imposes obligations of loyalty, care, and good faith on those entrusted with another's interests. SDRF governance is structured around fiduciary principles: operators of SDRF infrastructure bear duties to the participants whose data the system protects. These duties are legally enforceable, creating accountability that supplements technical controls. Fiduciary framing also clarifies priority: participant interests are not one stakeholder among many; they are the core obligation that governs system operation.

Governance Spectrum

Governance needs vary dramatically across deployment contexts. SDRF provides a spectrum of options that deployments can choose based on their scale, duration, and trust context. For small deployments, a single operator may manage the SDRF instance; participants consent to this arrangement through their manifests, and the operator's fiduciary duties provide accountability. For medium-scale deployments, a governing committee includes representatives of participant, researcher, institutional, and external stakeholder groups, providing legitimacy and diverse perspectives without on-chain complexity. For large-scale, long-term deployments, SDRF can integrate with DAO frameworks so that voting and execution are decentralized, transparent, and resistant to unilateral control.

Stakeholder Constituencies

Regardless of the governance model, SDRF recognizes four primary constituencies with distinct interests. Participants are individuals whose data the system protects; their interests in privacy, control, and benefit-sharing are paramount. Researchers bring scientific expertise and require efficient, reliable data access. Operators are responsible for reliability, security, and compliance, and must balance operational constraints with participant protections. Public interest representatives, such as bioethicists, patient advocates, and regulatory experts, provide perspective beyond immediate stakeholder interests and help align deployments with evolving legal and ethical norms.

Decision Taxonomy

Not all decisions require the same process. Routine operations are day-to-day decisions made unilaterally by operators. Administrative decisions affect the deployment but do not change fundamental

rules and may require committee approval or notification. Policy decisions change how the system operates in ways that affect stakeholder interests and require formal governance processes. Constitutional decisions change the fundamental rules of the deployment and require the highest level of consensus. This taxonomy prevents governance from becoming either a bottleneck or a rubber stamp.

Dispute Resolution

Disputes will arise. SDRF provides a tiered framework for resolution. The first tier is automated verification, where cryptographic audit trails resolve factual questions. The second tier is mediated resolution, where a neutral party reviews the dispute and proposes a resolution based on technical records and testimony. The third tier is formal adjudication through arbitration, litigation, or regulatory complaint when earlier tiers fail to resolve the issue. Auditability ensures that disputes can be resolved on evidence rather than conjecture.

Incident Response and Transparency

Governance also includes operational transparency. Security incidents are disclosed according to pre-committed protocols, with timelines, scope of impact, and remediation steps. Participants receive direct notification when their data may be affected. The system maintains public or semi-public incident logs so the community can evaluate governance performance over time. This transparency is critical for trust, especially in long-term studies where the value of participation accumulates over years.

XI. Economic Model

Sustainable operation requires a viable economic model. SDRF is designed to enable multiple funding approaches, recognizing that different deployments will have different economic contexts and that no single model will fit all research ecosystems.

Cost Structure

SDRF infrastructure costs fall into several categories. Storage costs are the ongoing cost of maintaining participant vaults. Compute costs depend on analysis complexity; simple aggregations are inexpensive, while training large models costs more. Bandwidth costs cover data movement between devices, vaults, compute environments, and researchers and are typically small relative to storage and compute. Governance costs include the human effort required for oversight, dispute resolution, and policy development. Development and maintenance costs include ongoing software development, security auditing, and operational support. In 2026 terms, budgets should be benchmarked against contemporary cloud storage and compute pricing, plus the compliance overhead of comparable human-subject studies, and revisited annually as costs change.

Research-Funded Model

The most straightforward funding model aligns with existing research economics: researchers fund studies, and study funds cover infrastructure costs. For many studies, SDRF infrastructure costs are comparable to conventional data management and materially lower than the compliance overhead of managing sensitive data through copy-and-store workflows. This model maps cleanly onto existing grant structures and enables predictable budgeting.

Participant Self-Sovereignty Model

SDRF also supports a model where participants directly control their vaults independent of any specific study. Participants pay for their own storage, authorize access to studies they choose to join, and retain complete control over future use. For many participants, these costs are comparable to

common subscription services; the psychological shift from "stored by an institution" to "stored by me" may be more significant than the financial shift. This model also enables participant-led research discovery, where studies compete for access rather than participants competing for inclusion.

Cooperative and Pooled Funding

SDRF supports cooperative structures where participants pool their data and collectively negotiate with researchers. A patient advocacy group or community organization might operate an SDRF instance for its members, covering infrastructure costs through membership fees or grants. This model provides economies of scale and collective governance, enabling communities to negotiate access terms from a position of shared strength. It is particularly appropriate for rare disease cohorts and underserved communities that benefit from collective bargaining power.

Researcher Access Fees

Some deployments may charge researchers for data access, creating a revenue stream that supports infrastructure costs. Fees might be per-participant, per-analysis, or per-study. Fee structures must be designed carefully to avoid excluding legitimate researchers with limited budgets while ensuring the platform remains financially sustainable. SDRF does not mandate any particular fee structure; it provides the technical controls that allow fee models to be chosen transparently and enforced consistently.

Value Flows and Benefit Sharing

Economic sustainability is not only about cost recovery. SDRF can support value flows back to participants through direct payments, reduced-cost access to their own data, or collective dividends negotiated by participant groups. This is optional and context-specific, but the architecture makes it enforceable when chosen. The broader goal is to ensure that the value created by sensitive data is aligned with the people who generate it.

XII. Security, Risk, and Compliance

SDRF reduces, but does not eliminate, risk. Its design assumes adversaries will attempt to exploit any weakness in storage, compute, or governance. Threats range from external attackers and insider misuse to legal compulsion and supply-chain compromise. The system therefore employs layered defenses: encryption, threshold keys, isolated compute, output control, and auditability.

From a compliance standpoint, SDRF is designed to align with existing regulatory principles rather than replace them. The HIPAA minimum necessary standard is implemented as a protocol constraint: analyses request only what is needed, and manifest policies enforce those constraints at the system level. GDPR principles such as data minimization, purpose limitation, and special-category protections are supported by manifest-based access control, explicit purpose tagging, and revocation mechanisms. These alignments do not eliminate legal obligations, but they make compliance technically enforceable rather than solely procedural.

Security is also operational. Access logs, attestation records, and key derivation events are recorded and auditable. Incident response procedures are codified as governance requirements. Participants can verify that their data has been accessed only as authorized, and researchers can verify that analyses were executed as declared. This dual verifiability is the core of trust in SDRF.

XIII. Threat Model and Risk Boundaries

SDRF assumes motivated adversaries and imperfect operators. The system is designed for scenarios where external attackers target centralized stores, where internal misuse is possible, and where the analytics surface itself can leak sensitive information even when raw data never leaves a vault. A core

boundary is that results and trained models can encode membership or sensitive attributes, which is why output control must be treated as a first-class security property rather than an afterthought.[9,10] SDRF therefore treats the research pipeline as a disclosure channel, not only the storage layer, and assumes that privacy loss can occur through models and summaries unless constrained.

SDRF also assumes that distributed and federated computation has its own attack surface. In federated learning, a single malicious participant can poison or backdoor a global model and secure aggregation can hide the malicious update from centralized inspection.[11] In collaborative training, gradients and parameter updates can leak private information about the underlying data, including reconstructions of original samples.[12] These risks exist even when the data never leaves the participant environment. SDRF does not claim to eliminate them by default; it treats them as explicit design constraints that must be managed through consented analysis types, disclosure controls, and rigorous review of what is released.

XIV. Data Lifecycle and Revocation Semantics

SDRF treats data as a governed lifecycle rather than a static artifact. Data is captured into encrypted vaults, tagged with purpose and consent metadata, and linked to explicit retention rules. Consent revocation is enforced at the system level so that future computations requiring the revoked permissions are blocked. However, the system also acknowledges that outputs already released cannot be recalled without imposing additional contractual obligations on recipients, and derived artifacts such as embeddings, aggregates, or trained models must be tracked as part of the lifecycle. When consent is withdrawn, SDRF requires that downstream artifacts be evaluated for continued eligibility under the original consent terms, and if they no longer qualify, they are deprecated and excluded from future use or reissued under appropriate privacy constraints. This approach aligns with the spirit of erasure and purpose limitation while acknowledging the practical limits of irrevocable publication.[13]

XV. Assurance and Verification

SDRF is designed so that trust can be verified rather than assumed. Critical actions such as consent grants, key derivations, and analysis executions are logged with cryptographic integrity and are auditable by participants and authorized oversight bodies. When confidential compute platforms are used, the system relies on attestation mechanisms so that researchers and participants can verify the code and environment that processed sensitive data before any output is released.[14,15,16] The governance layer commits to transparent incident reporting and third-party review of security controls, because long-term scientific credibility depends on demonstrable operational discipline, not only architectural intent.

---

[9] Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). *Membership Inference Attacks against Machine Learning Models*. IEEE Symposium on Security and Privacy. https://arxiv.org/abs/1610.05820

[10] Fredrikson, M., Jha, S., & Ristenpart, T. (2015). *Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures*. Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. https://dblp.org/rec/conf/ccs/FredriksonJR15

[11] Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., & Shmatikov, V. (2020). *How To Backdoor Federated Learning*. Proceedings of AISTATS. https://proceedings.mlr.press/v108/bagdasaryan20a.html

[12] Zhu, L., Liu, Z., & Han, S. (2019). *Deep Leakage from Gradients*. NeurIPS. https://papers.nips.cc/paper/9617-deep-leakage-from-gradients

[13] European Union. *Regulation (EU) 2016/679 (GDPR)*. https://eur-lex.europa.eu/eli/reg/2016/679/oj

[14] Intel. *Intel® Software Guard Extensions (Intel® SGX)*. https://www.intel.com/content/www/us/en/architecture-and-technology/software-guard-extensions.html

[15] AMD. *Confidential Computing with AMD EPYC Processors*. https://www.amd.com/en/products/processors/server/epyc/confidential-computing

[16] AWS. *Nitro Enclaves Documentation*. https://docs.aws.amazon.com/enclaves/latest/user/nitro-enclave.html

XVI. Consent Withdrawal and Lawful Basis

SDRF treats consent as the default basis for human-subject research access, but it does not assume consent is the only lawful basis in every jurisdiction or study context. Some research regimes allow processing under statutory research exemptions or public interest grounds, subject to strict safeguards and minimization requirements. SDRF supports this by encoding purpose limitation, retention constraints, and minimum necessary access directly in consent manifests or equivalent authorization records, allowing oversight bodies to verify that non-consent use remains constrained to the authorized scope.[17,18] Where consent is the basis, withdrawal is treated as a revocation of future access, and the system is structured to make that revocation immediate and enforceable at the protocol level.

XVII. Related Systems and Differentiators

SDRF sits in a broader ecosystem of personal data stores, decentralized identity, and privacy-preserving analytics. Several initiatives address important parts of the problem; SDRF is designed to integrate those parts into a research-grade, participant-sovereign fabric rather than treat them as separate layers.

Personal Data Stores and Solid Pods

Solid defines Pods as web storage locations where applications read and write data based on authorizations granted by the user. It improves user control and data portability and can be self-hosted, but its access model cannot prevent an app from copying data once it has been granted read access. Revocation stops future access but does not delete copies that an app already duplicated, and encryption at rest is determined by the Pod provider rather than mandated by the specification.[19]

Decentralized Web Nodes (DWN)

The Decentralized Web Node (DWN) specification from the Decentralized Identity Foundation defines a data storage and message relay mechanism for data related to decentralized identifiers, with a mesh-like design that can sync across multiple nodes. The specification is explicitly a draft and primarily defines storage, permissions, and interoperability; it does not prescribe privacy-preserving computation or research governance by itself.[20]

Remote Data Science Platforms (PySyft)

PySyft is an open-source remote data science platform built around sending code to where data resides and returning results rather than sharing raw data. Its workflow is organized around data-owner environments: researchers submit code for review, and data owners approve execution and results release. This is highly aligned with the compute-to-data principle, though it typically assumes an organization-owned data-owner environment rather than a participant-owned vault model.[21]

PDS + MPC Research (Libertas)

Academic work such as Libertas explores how to integrate secure multi-party computation (MPC) with decentralized personal data stores like Solid, addressing the privacy challenges of collective computation across user-controlled data. The work evaluates scenarios that include generating

[17] HHS Office for Civil Rights. *Minimum Necessary Requirement* (HIPAA Privacy Rule). https://www.hhs.gov/hipaa/for-professionals/privacy/guidance/minimum-necessary-requirement/index.html
[18] European Union. *Regulation (EU) 2016/679 (GDPR)*. https://eur-lex.europa.eu/eli/reg/2016/679/oj
[19] Solid Project. *Frequently Asked Questions*. https://solidproject.org/faqs
[20] Decentralized Identity Foundation. *Decentralized Web Node (DWN) Specification* (Draft). https://identity.foundation/decentralized-web-node/spec/
[21] OpenMined. *PySyft: Remote data access with privacy at the core*. https://openmined.org/pysyft/

differentially private synthetic data, showing feasibility of privacy-preserving computation without centralizing the underlying personal data.[22]

What SDRF Adds

SDRF differentiates itself by unifying participant-controlled vaults, cryptographic consent manifests, isolated compute, privacy-preserving outputs, and audit-ready governance into one coherent research stack. It is designed for human-subject research, where revocation, minimization, and reproducibility are not optional features but core requirements. The goal is not to replace the above systems, but to provide a reference architecture that can incorporate compatible components while enforcing research-grade privacy and consent end to end.

XVIII. Implementation Path and Evaluation

SDRF is designed to be adopted incrementally. A first deployment can focus on a narrow study with a limited set of data types and analysis templates. As confidence grows, additional modalities, analysis types, and governance structures can be layered in without breaking compatibility. The architecture supports both greenfield deployments and integration with existing clinical or research systems.

Evaluation metrics should be explicit. For privacy, metrics include the size of the privacy budget consumed, the differential privacy parameters used, and the number of outputs suppressed by disclosure controls. For usability, metrics include participant comprehension, consent modification rates, and dropout rates. For scientific utility, metrics include analysis reproducibility, cross-site comparability, and time-to-analysis compared with traditional workflows. For governance, metrics include dispute resolution time, incident response performance, and participant satisfaction.

The implementation path also includes a migration strategy for existing datasets. Data can be imported into participant vaults only when consent allows. In cases where re-consent is not feasible, SDRF can operate in parallel with legacy systems while new data collection adopts compute-to-data defaults.

XIX. Conclusion

The Sovereign Data Research Fabric represents a fundamental rethinking of how personal data flows through the research enterprise. It begins from the premise that ethical principles such as participant autonomy, meaningful consent, and privacy protection should be enforced by the systems we build, not merely promised by the institutions we trust.

SDRF provides the technical mechanisms to make these principles real. Cryptographic consent ensures that only authorized operations proceed. Threshold cryptography ensures that no single party can access data unilaterally. Controlled computation ensures that raw data never leaves protected environments. Output checking and differential privacy reduce disclosure risk. Governance structures ensure that human judgment complements technical controls.

The architecture is specified in this paper. What remains is the commitment to build and validate deployments that prove the model at scale. SDRF offers a path to research infrastructure that is scientifically rigorous and ethically aligned, without requiring participants to surrender control of their most sensitive data.

[22] Zhao, R., et al. (2023). *Libertas: Privacy-Preserving Collective Computation for Decentralised Personal Data Stores.* arXiv. https://arxiv.org/abs/2309.16365

Glossary

Decentralized Identifier (DID). A W3C standard for self-managed identifiers that resolve to a DID Document containing public keys and service endpoints.[23]

Verifiable Credential (VC). A W3C data model for cryptographically signed claims that can be selectively disclosed and verified without contacting the issuer at verification time.[24]

Personal Data Store (PDS). A user-controlled data store intended to give individuals direct control over their personal data; Solid is a prominent example.[25]

Solid Pod. A web-based personal data store in the Solid ecosystem where data is stored and accessed by applications based on user-granted authorizations.[26]

Decentralized Web Node (DWN). A data storage and message relay mechanism for data related to decentralized identifiers, defined by the Decentralized Identity Foundation.[27]

Remote Data Science (PySyft). An approach in which researchers send code to data-owner environments and receive results without taking possession of raw data; PySyft implements this model with review workflows.[28]

HL7 FHIR. A health data interoperability standard for representing clinical observations, procedures, medications, conditions, and related resources.[29]

HIPAA Minimum Necessary Standard. A U.S. Health Insurance Portability and Accountability Act requirement that covered entities limit data use and disclosure to the minimum necessary to accomplish a purpose.[30]

GDPR (General Data Protection Regulation). The European Union regulation governing personal data processing, including special-category protections, data minimization, and purpose limitation.[31]

Differential Privacy. A formal privacy framework that bounds what can be learned about an individual from published outputs, even in the presence of auxiliary information.[32]

Federated Learning. A distributed machine learning approach that trains models across decentralized data sources without centralizing raw data.[33]

Confidential Computing / Trusted Execution Environments (TEEs). Hardware- and virtualization-based isolation that allows code to run with memory protection from the host system; common implementations include Intel SGX, AMD SEV, and AWS Nitro Enclaves.[34,35,36]

[23] W3C. *Decentralized Identifiers (DIDs) v1.0.* W3C Recommendation (2022). https://www.w3.org/TR/did-core/
[24] W3C. *Verifiable Credentials Data Model v2.0.* W3C Recommendation (2025). https://www.w3.org/TR/vc-data-model/
[25] Zhao, R., et al. (2023). *Libertas: Privacy-Preserving Collective Computation for Decentralised Personal Data Stores.* arXiv. https://arxiv.org/abs/2309.16365
[26] Solid Project. *Frequently Asked Questions.* https://solidproject.org/faqs
[27] Decentralized Identity Foundation. *Decentralized Web Node (DWN) Specification* (Draft). https://identity.foundation/decentralized-web-node/spec/
[28] OpenMined. *PySyft: Remote data access with privacy at the core.* https://openmined.org/pysyft/
[29] HL7. *FHIR Specification.* https://hl7.org/fhir/
[30] HHS Office for Civil Rights. *Minimum Necessary Requirement* (HIPAA Privacy Rule). https://www.hhs.gov/hipaa/for-professionals/privacy/guidance/minimum-necessary-requirement/index.html
[31] European Union. *Regulation (EU) 2016/679 (GDPR).* https://eur-lex.europa.eu/eli/reg/2016/679/oj
[32] Dwork, C. (2006). Differential Privacy. *ICALP.* https://www.microsoft.com/en-us/research/publication/differential-privacy/
[33] McMahan, B., Moore, E., Ramage, D., Hampson, S., & Agüera y Arcas, B. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data. *AISTATS.* https://proceedings.mlr.press/v54/mcmahan17a.html
[34] Intel. *Intel® Software Guard Extensions (Intel® SGX).* https://www.intel.com/content/www/us/en/architecture-and-technology/software-guard-extensions.html
[35] AMD. *Confidential Computing with AMD EPYC Processors.* https://www.amd.com/en/products/processors/server/epyc/confidential-computing
[36] AWS. *Nitro Enclaves Documentation.* https://docs.aws.amazon.com/enclaves/latest/user/nitro-enclave.html

Lab Streaming Layer (LSL). A protocol and software stack for synchronized, real-time streaming of time-series data from research sensors and devices.[37]

XDF (eXtensible Data Format). A container format used with LSL to store synchronized multimodal time-series data and metadata.[38]

EDF/EDF+. The European Data Format and its extension for biosignal and sleep-study data, commonly used for EEG and polysomnography.[39]

FIT (Flexible and Interoperable Data Transfer). A data format and SDK used by many wearable devices to encode activity and wellness data.[40]

Shamir's Secret Sharing / Threshold Cryptography. A cryptographic technique for splitting a secret into shares such that only a threshold number of shares can reconstruct it.[41]

Internet Computer (ICP). A decentralized network for running canister smart contracts with deterministic execution and subnet replication.[42]

Canister Smart Contracts. The Internet Computer's computational units, providing isolated execution, persistent state, and defined interfaces for interaction.[43]

vetKeys. The Internet Computer's verifiably encrypted threshold key system for deriving keys on demand without key escrow.[44]

OAuth 2.0. An authorization framework widely used to grant limited access to APIs and user data via scoped tokens.[45]

License and Attribution

Recommended citation: Deegan, P. 2026. *Sovereign Data Research Fabric: A Privacy-First Architecture for Human-Subject Research.* Version 1.2. https://github.com/SovereignScience/SDRF-whitepaper

SDRF is open infrastructure for participant-controlled research.

[37] Lab Streaming Layer. https://labstreaminglayer.org/
[38] XDF (eXtensible Data Format) specification and tooling. https://github.com/sccn/xdf
[39] EDF/EDF+ Specifications. https://www.edfplus.info/specs/
[40] Garmin. *FIT SDK Overview.* https://developer.garmin.com/fit/overview/
[41] Shamir, A. (1979). How to Share a Secret. *Communications of the ACM.* https://dblp.org/rec/journals/cacm/Shamir79
[42] Internet Computer. *How the Internet Computer works* (overview). https://internetcomputer.org/how-it-works
[43] Internet Computer. *Canisters* (developer docs). https://internetcomputer.org/docs/building-apps/essentials/canisters
[44] Internet Computer. *vetKeys* (developer docs). https://internetcomputer.org/docs/building-apps/network-features/vetkeys/introduction
[45] IETF. *The OAuth 2.0 Authorization Framework (RFC 6749).* https://datatracker.ietf.org/doc/html/rfc6749