Ownership and Flow Primitives for Scalable Consent Management in Digital Public Infrastructures

Rohith Vaidyanathan, Dev Shinde, Praseeda, Srinath Srinivasa
Web Science Lab
International Institute of Information Technology, Bangalore
Bengaluru, Karnataka, India
e-mail: {rohith.vaidyanathan, shindedev.hemravi, praseeda, sri}@iiitb.ac.in

***Abstract*—Digital Public Infrastructures (DPIs) represent networks of open technology standards, applications, services, and digital assets made available for the public good. One of the key challenges in DPI design is to resolve complex issues of consent, scaled over large populations. While the primary objective of consent management is to empower the data owner, ownership itself can come with variegated morphological forms with different implications over consent. This paper addresses the question of representing modes of ownership of digital assets and their corresponding implications for consensual data flows in a DPI. It proposes a set of foundational abstractions to represent them, incorporating a formalised data ownership model that enables end-to-end traceability of consent, fine-grained flow control over data sharing, and alignment with evolving legal and regulatory frameworks.**

Keywords-Digital Public Infrastructure; Consent Management; e-Government; Data Ownership; Data Flow Control; Privacy Architecture; Data Governance.

I. INTRODUCTION

Digital Public Infrastructures (DPIs) [1] represent open ecosystems of digital services, applications and assets made available for public good. DPI services like digital identity, lockers, catalogs, wallets and payment infrastructures have streamlined several e-governance activities and economic, legal and social transactions [2]. There have been several initiatives in the past where digital assets and services were designed to be owned by and available for the public, such as free software movements and creative commons. However, the term DPI refers to systemic, scalable infrastructure that is meant to act as a "digital backbone" for the entire society, enabling access to both public and private services like healthcare, education, and financial inclusion. The role of the state and institutional players is central to DPI design, distinguishing it from community-driven approaches.

As public interactions become more digital, vast amounts of sensitive data are exchanged through DPIs. Without proper mechanisms, data owners may lose control over how their data is used post-exchange. Various countries have implemented data protection and consent management laws to address this [3]. Consent management is modeled as an interception of a data flow in a DPI, to ensure that the flow is consistent with the consent of the data owner.

In its simplest form, a consent architecture elicits explicit consent from the Data Owner (DO) in response to a data request, following the Autonomous Authorization (AA) model [4]. However, as DPI implementations scale, the AA architecture becomes insufficient due to consent desensitization from frequent requests. Additionally, if data is shared by an independent custodian on behalf of the owner, the custodian needs data sharing policies set by the owner. As a result, *policy-based* consent management architectures become necessary for DPI implementations [5].

Consent management also has several other nuances. Consent is closely related to ownership, which can have several morphological forms. Ownership may be *delegated* or *pledged*, and partial ownership or privileges can be *conferred* all with implications on how consent is granted and enforced. Consent management also extends beyond access control: consent provisions may need enforcement even after access is granted, including whether data may be shared further, aggregated with other data, the purposes for which they may be used, and requirements of suitable notifications to the data owner.

While DPI provides the primary motivating context for this work, the ownership and flow primitives proposed here are designed to be architecture-agnostic. The same abstractions apply equally to enterprise data governance, academic credential ecosystems, cross-institutional healthcare networks, and any open-ended data exchange environment where ownership semantics and post-access governance are required.

This paper proposes an architecture that addresses complex forms of ownership and flow primitives required for supporting consensual data flows.

The remainder of this paper is organised as follows. In Section II, we survey related literature on data sharing, access control, and consent management. In Section III, we introduce the formal model for public data flows. In Section IV, we describe the consent flow architecture, including connections and X-nodes. In Section V, we define data exchange operations. In Section VI, we analyse adversarial scenarios and the system's resilience to them. Finally, Section VII concludes the paper and outlines future work.

II. RELATED LITERATURE

Although the term DPI is a relatively recent introduction, some of its core data-related challenges, such as inter-organizational and open-ended sharing of sensitive data, have been addressed for several years [6]–[9]. Inter-organizational data sharing addresses workflows spanning multiple organizations and governance structures [10][11].

 ISBN: 978-1-68558-382-8

When access requests cross organizational boundaries, they could potentially lead to unsafe data access. Innovations in this space include extensions of Role-Based Access Control (RBAC) [12][13], attribute-based models [14][15], or hybrid models [16][17]. However, these access control mechanisms focus on the authorization decision whether to grant access but do not address post-access governance: what recipients can do with data after receiving it, how ownership evolves through sharing, or how to revoke access after cross-organizational transfer.

Consent management in public data flows goes beyond access control [5]. Research has addressed philosophical issues of what makes consent meaningful [4][18], as well as various consent architectures [5][19][20]. Existing systems like DI-CON (Domain Independent Consent Management) [21] and CMA (Consent Management Architecture) [22] treat consent primarily as an authorization checkpoint rather than dynamic user-centric control. Most existing consent managers employ AA [4], where explicit consent is elicited for each request. However, AA is not scalable for DPIs due to: (i) consent fatigue from frequent requests leading to desensitization; (ii) inability to handle custodial sharing where data is managed on behalf of owners; (iii) infeasibility of human intervention for institutional data owners; and (iv) request volumes in large-scale DPI implementations that make individual approval impractical. Policy-based consent [5] addresses these limitations by enforcing pre-defined rules in a domain-agnostic manner. Questions persist about what data flows one can legitimately control by virtue of ownership [23][24].

Regulations in India [25] and Europe [26] have outlined mechanisms for lawful consent. Pioneering frameworks include X-road [27][28]. As data is shared in an open-ended fashion in DPIs, data ownership morphs into different constructs requiring meaningful modeling regarding their impact on consent.

An effective consent management system must encompass not only access control but also explicitly address issues of ownership to enable ongoing user control over data following its sharing and ensure compliance with evolving data policies. In this paper, we propose a comprehensive model that systematically addresses all four of these critical dimensions of consent management.

Table I defines the core terminology used throughout this paper.

TABLE I. KEY TERMINOLOGY.

Term	Definition
DO	Data Owner: Controls data resource with full authority
DR	Data Requester: Seeks access to data owned by another
DS	Data Subject: Entity described by the data

III. MODELING PUBLIC DATA FLOWS

Any public infrastructure poses a complex interplay between individual rights and public interest. We model a public data flow network as semantic containers representing ownership boundaries of stakeholders, as defined in Equation 1:

$$DPI = (A, L, C) \quad (1)$$

Here, A is a set of *agents* or *stakeholders* who assert ownership and play roles like Data Owner (DO), Data Requester (DR), or Data Subject (DS). L is a set of containers called *Access Policy Domains (APD)* or *lockers*, representing semantic boundaries where data ownership is enforced. $C \subseteq L \times L$ represent data flow pipelines called *Connections*, established
between lockers as legitimate pathways for data exchange, made legitimate by underlying contracts encapsulating data sharing policies and applicable regulations.

An *artifact* is a logical unit of data subject to ownership and consent that flows through connections. It represents a data resource (e.g., a document), though a single resource may have multiple artifacts. The consent service regulates artifact storage and flow, while a separate resource service manages actual resources.

IV. CONSENT FLOW ARCHITECTURE

We illustrate consent flow using a running example: A student s uses a DPI to obtain her degree from university u and applies for a job with company c.

Degree granting: Student s requests transcripts from university u. While s is the owner of her transcripts, she cannot modify them unilaterally, she is the *conferred owner*, while u remains the *primary owner*.

Job application: Student s shares her transcripts and degree credentials with company c. As part of this share, the company only requires *access rights*, not ownership. "Sharing" here means granting access rights.

Credentials Verification: Company c seeks verification from university u. The university may verify without consent of s (as primary owner) unless regulations require otherwise.

Job offer: Company c requires s to *pledge* her certificates as collateral as long as she is in full-time employment with them. As a pledged asset, s retains access but cannot transfer or re-pledge them.

Job Contract: Upon pledging, company c becomes the *pledged owner* with limited rights, while university u remains the *primary owner*. Student s continues as *conferred owner* in a constrained manner.

Contract Termination: Once the contract ends, transcripts are returned to s, company c no longer has access, and s is no longer subject to pledge restrictions. The pledge is *reverted*.

A. Connection between Lockers

Before data transactions, agents establish a *connection* between lockers, formalizing the terms of a consensual transaction. A DO publishes *connection endpoints* representing connection terms. A DR can connect any of their lockers to a given endpoint (Figure 1).

A *connection type* specifies a schema representing terms and conditions for connection establishment, encoding rules from the DO's policy and applicable regulations. Rules are

expressed as Event-Condition-Modality-Action (ECMA) statements [5], where each rule binds a triggering *event* (e.g., a data access request) to a *condition* (e.g., stated purpose, requester identity) and assigns a normative *modality* Obligated (O), Permitted (P), or Forbidden (F) over a resulting *action*. These OPF modalities encode the normative intent of a policy rather than serving as mere technical flags. *Obligated* (O) denotes that an action *must* occur; the requester cannot proceed until it is fulfilled (e.g., submitting ethical clearance before accessing sensitive health data). *Permitted* (P) denotes that an action *may* occur; it represents a green light that can be further restricted by layered policies. *Forbidden* (F) denotes that an action *must not* occur. Together, OPF move consent management beyond binary allow/deny access control into a richer normative space where every artifact operation carries an enforceable legal and ethical standing. A sample ECMA rule takes the form:

```
ON [RequestAccess] IF [purpose =
"verification"] THEN PERMIT [read]
WITH [validity = 7 days, share =
false]
```

Multiple rules from different institutional sources may fire simultaneously on the same artifact. When their modalities are complementary, for example when a regulation permits an action and an organizational policy further restricts it, they compose predictably: the stricter modality prevails. When modalities directly conflict (e.g., one source requires an action that another forbids), the system flags an irreconcilable policy exception rather than silently resolving it. Such conflicts are escalated to a designated human agent for resolution, reflecting the principle that normative contradictions between institutions require human judgment, not automated override. Depending on the deployment context, the policy evaluation layer may be realized as a rule-based engine, a logic interpreter, or an AI-assisted policy reasoner; the ECMA structure is substrate-agnostic. Readers are directed to [5] for a full formal treatment of the four-layer architecture, normative axioms, and conflict resolution semantics.

The connection lifecycle (Figure 2) begins with the PUBLISHED state when the DO publishes endpoints. When the DR selects an endpoint, it transitions to ESTABLISHED. After fulfilling obligations, it becomes LIVE, enabling the data exchange operations detailed in Section V. Either party may REVOKE during ESTABLISHED or LIVE states. After completion or expiry, connections are CLOSED.

B. X-nodes

Once a connection is live, *artifacts* representing data resources flow through the connections. Note that the data resource itself does not flow through a connection; only the artifacts that represent privileges over data resources are exchanged through these connections.

We propose a formal data structure called *X-nodes* for consent artifacts that encapsulate terms of consent and enforce post-conditions. Table II summarises the essential fields and operations for each X-node type.

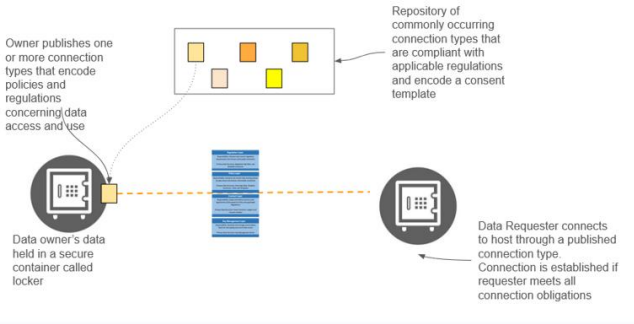

Figure 1. Establishing a consensual pathway governed by the four-layer architecture of consent [5].

X-nodes are of three types. The i-node ("information" node) represents the primary location of a resource with full authority for the primary owner. The s-node ("shadow" node) is received when data is conferred, allowing read/write access as per the policy specified by the DO with a pointer to the original artifact. The v-node ("virtual" node) represents an access privilege granted for a resource, containing only a pointer to the original artifact with a mandatory validity period.

Three critical ownership fields define the control structure of each X-node: `creator` identifies the agent who originally instantiated the X-node. `primary_owner` (PO) represents the entity with fundamental authority over the resource (e.g., government for driving licenses, university for degree certificates), which can confer, modify, or revoke the resource. `current_owner` (CO) indicates the entity presently in possession of the X-node with operational control. When `primary_owner` and `current_owner` differ, the X-node is said to be *locked*, preventing transfer, conferment, or pledging until the constraint is resolved. Each X-node has additional essential fields (Table II): `shadows_list`, `v-node_list`, `pointer_to_resource`, `pointer_to_original`, `validity`, `purpose`, and `provenance`.

Essential post-conditions specify actions the Data Requester can perform: `transfer`, `confer`, `share`, `collateral`, `subset`, and `download`.

V. Data Exchange Operations

The following kinds of data exchange operations are defined, each with different semantics over ownership and consent:

SHARE: The DO shares an *access privilege* to resource *r*. A v-node sent to the data requester is formally of the form $dr.v(do.i(r))$ or $dr.v(do.s(r))$, where dr is the data requester and do is the data owner. The v-node represents a pointer to a corresponding i-node or s-node, which in turn represents data resource *r*. This v-node can be further shared to yet another requester, creating an "access tunnel" of the form $dr2.v(dr.v(do.i(r)))$. The actual access to resource *r* traverses the tunnel and is initiated by the last i-node or s-node. The resource server receives the *access tunnel*, approving access

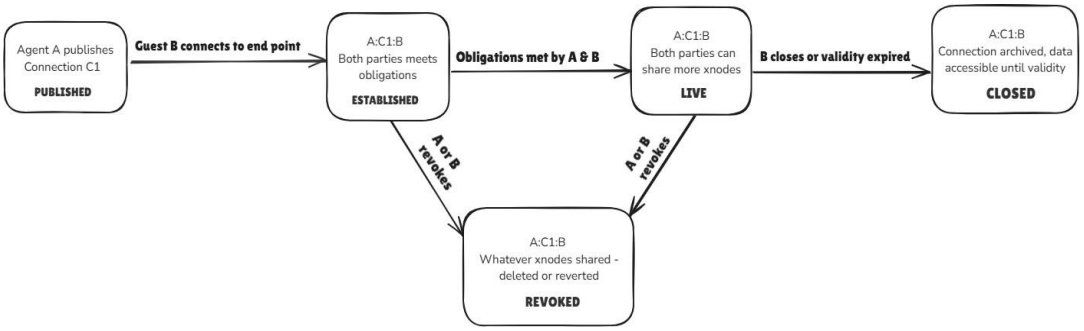

Figure 2. Connection lifecycle illustrating Agent B establishing a connection of type C1 published by Agent A.

TABLE II. ESSENTIAL FIELDS AND OPERATIONS OF X-NODES WITH PRACTICAL EXAMPLES.

X-node	Function	Essential fields	Essential post-conditions	Example Use Case
i-node	Primary location of resource	`creator`, `primary_owner`, `current_owner`, `shadows_list`, `v-node_list`, `pointer_to_resource`, `purpose`, `provenance`	`transfer`, `confer`, `share`, `collateral`, `subset`, `download`	Government holds i-node of citizen's driving license with full authority to issue, modify, suspend, or revoke
v-node	Access privilege to an i-node	`creator`, `current_owner`, `pointer_to_original`, `validity`, `v-node_list`, `purpose`, `provenance`	`transfer`, `share`, `download`	Car rental company receives temporary v-node to verify driver's license validity for 7 days, without owning the license
s-node	Conferred or pledged ownership	`creator`, `primary_owner`, `current_owner`, `pointer_to_original`, `shadows_list`, `v-node_list`, `pointer_to_resource`, `purpose`, `provenance`	`transfer`, `share`, `collateral`, `subset`, `download`	Citizen receives s-node as conferred owner of driving license, can present it but cannot modify; or pledges it as ID proof

only if *do* is `primary_owner`. Since access comes through the owner's i-node or s-node, the DO is aware of every access (Figure 3(ii)). *Example:* A car rental company requests to verify a customer's driving license. The customer (DO) creates a v-node for the conferred license and sets a validity of (say) 7 days, allowing the rental company (DR) temporary read-only access to verify license validity without transferring ownership or providing a permanent copy.

CONFER: The DO shares an immutable copy to the DR, who becomes owner (e.g., certificates, licenses, tickets). The DO creates an s-node that is sent to DR. The s-node's `primary_owner` and `current_owner` are set to DR, while the i-node's `primary_owner` remains DO and `current_owner` becomes DR, making it *locked* (Fig-ure 3(iii)). Conferred owners access resources read-only through s-nodes. *Example:* A university issues a degree certificate to a student. The university (DO) creates an s-node from the i-node it owns and confers it to the student (DR). The student can present the certificate to employers but cannot modify its contents. If the university needs to correct an error (e.g., a misspelled name), only the university can make changes to the original i-node.

TRANSFER: Here, ownership *transfers* from DO to DR. The X-node itself transfers to DR with both PO and CO set to DR, providing complete ownership. The former DO loses access (Figure 3(i)). Transfer invalidates v-nodes and s-nodes pointing to it. Any forbidden post-condition set by the artifact creator continues across transfers. *Example:* A person sells their car registration documents to the new owner. The original owner (DO) transfers the i-node to the buyer (DR), who becomes both primary and current owner. The seller loses all access to the documents, and the buyer can now modify, share, or transfer them independently, subject only to restrictions set by the original creator (e.g., the vehicle authority).

COLLATERAL: Here, the resource is pledged as collateral. The DO sends the i-node to DR, keeping `primary_owner` as DO and setting `current_owner` to DR. The DR creates an s-node with `primary_owner` as DR and `current_owner` as DO. Both X-nodes are locked, preventing further pledge, conferment, or transfer until released (Figure 3(iv)). *Example:* A job applicant pledges their degree certificate as part of an employment contract, agreeing not to use it for other applications during employment. The applicant (DO) sends the i-node to the employer (DR), who holds it as collateral. The employer issues an s-node back to the applicant, who retains the ability to present the certificate for verification but cannot pledge it elsewhere. Upon contract termination, the pledge is reverted and the i-node returns to the applicant.

Compositions: V-nodes from SHARE can be further shared (if permitted), creating *access tunnels* through multiple consent layers (Figure 4). V-nodes can be transferred, moving access pathways. S-nodes from conferment can be subject to SHARE, COLLATERAL, or TRANSFER (unless restricted). Locked artifacts from COLLATERAL cannot be transferred, conferred, or pledged, but SHARE is permitted.

CLOSE, REVOKE, REVERT: CLOSE ends an open connection after successful data sharing. Downloaded artifacts remain with DR; v-nodes continue until validity expires. REVOKE rolls back a connection, reverting i-nodes and s-nodes to original owners. REVOKE on artifacts deletes v-nodes (SHARE) or returns X-nodes (TRANSFER). REVERT reverses conferment or collateral even after connection closure, returning pledged i-nodes or deleting conferred s-nodes.

Together, these operations form a composable, policy-governed layer for consensual data exchange that supports fine-grained ownership tracking throughout the lifecycle of every artifact in the DPI.

VI. Adversarial Scenarios

In this section, we present a few adversarial scenarios and show how the proposed architecture addresses them.

Scenario 1: Impersonation. Bob creates a fake locker impersonating a legitimate vendor to access sensitive tender documents. Alice, a company (DO), publishes a connection endpoint for vendor registration. Bob (DR) connects his fraudulent locker, claiming to represent "XYZ Corp," attempting to access project specifications shared with verified vendors.

Handling: Alice's connection type specifies obligations requiring DR to share verifiable credentials (business registration, tax ID, digital attestations) before the connection becomes live. When Bob connects, the connection enters ESTABLISHED state but remains non-operational. Alice verifies the submitted credentials against authoritative sources (e.g., government registries). The connection transitions to LIVE only after successful verification. Since Bob cannot provide legitimate credentials, the connection remains in ESTABLISHED state, blocking all artifact exchange. Alice can REVOKE the connection if verification fails, ensuring data sharing occurs only with authenticated agents.

Scenario 2: External Replication. Alice (DO) shares confidential financial documents with an auditing firm (DR) via v-node with `download` set to false for compliance verification only. Bob, an employee with legitimate v-node access, attempts to circumvent controls by photographing the screen with his mobile device to leak information to competitors.

Handling: The client application enforces view-only constraints, disabling screenshots, screen recording, and downloads. However, the consent manager cannot prevent physical photography. To mitigate this, the resource displays dynamic watermarking embedding Bob's identity, timestamp, and access context, making unauthorized copies traceable. The consent manager logs comprehensive audit trails including Bob's identity, legal capacity (auditing firm employee), connection details, timestamps, and access purpose. If leaked documents are discovered, these logs and watermarks provide forensic evidence identifying Bob as the breach source, enabling Alice to pursue legal remedies.

Scenario 3: Cascaded Sharing. Alice (DO) shares medical records with Dr. Smith (DR1) via v-node. Dr. Smith attempts to cascade share to specialist Dr. Jones (DR2) for consultation without Alice's explicit consent, creating a v-node from his existing v-node.

Handling: If Alice prohibits re-sharing, the `share` post-condition is set to false, blocking Dr. Smith from cascading the share. If permitted, Dr. Smith creates a v-node for Dr. Jones, forming access tunnel $dr2.v(dr1.v(do.i(r)))$. All access requests traverse Alice's i-node, ensuring she is aware of all accesses. Alice can revoke either Dr. Smith's v-node (eliminating both accesses) or specifically Dr. Jones's cascaded v-node at any time, maintaining ultimate control over her medical records.

Scenario 4: Cross-Border. A DR is in a different legal jurisdiction from the DO. *Handling:* Cross-border data transfers are not currently supported within this model. The model assumes DR and DO are within the same jurisdictional boundary; extending it to cross-border scenarios is an important avenue for future work.

VII. Conclusion

In this paper, we propose an architecture addressing data security and governance as data transitions through its core states in a DPI: *at rest*, *in transit*, and *in use*.

The present work contributes a *semantic architecture*: a set of formally grounded ownership and flow primitives that serve as a technology-agnostic blueprint from which concrete systems can be derived. A working implementation of this architecture, the Anumati Consent Management System developed at the Web Science Lab, IIIT Bangalore, is publicly accessible at [29].

To summarise, we presented an architecture using X-nodes and Connections to support consensual, public data exchange that empowers data owners and supports compliance with diverse regulations, including personal and other data laws. A key strength of the framework is that its ownership and flow primitives are architecture-agnostic: they apply equally to enterprise data governance, academic credential networks, and healthcare interoperability. By managing data sharing through regulated pipelines called connections and by supporting detailed tracking of consent and ownership changes, our design allows data owners to retain control even after data is shared. Overall, implementing consent management at the DPI layer ensures that it can be both robust and adaptable, giving individuals genuine control and helping organizations remain compliant as data laws and expectations evolve.

References

[1] United Nations Development Programme, "Digital public infrastructure," Accessed: 2025-07-08, 2025. [Online]. Available: https : / / www . undp . org / digital / digital - public - infrastructure.

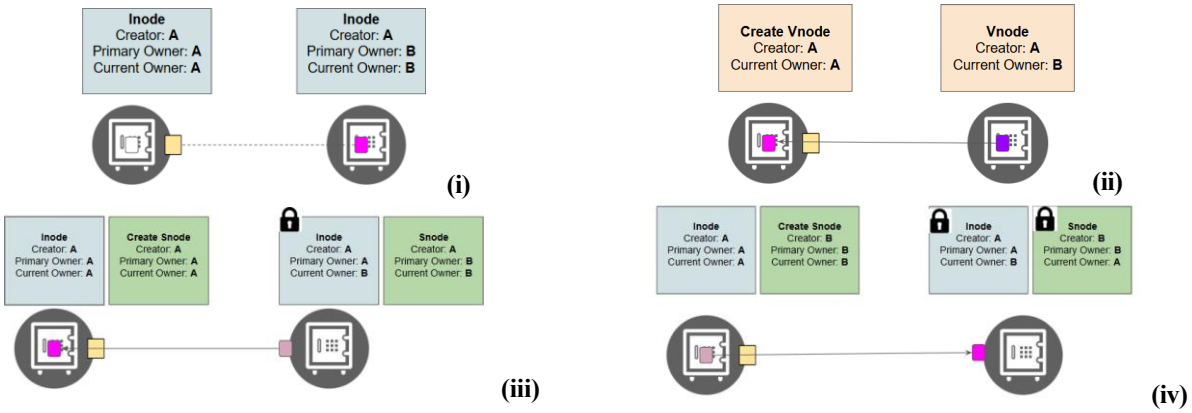

Figure 3. Sharing operators: (i) TRANSFER (ii) SHARE (iii) CONFER (iv) COLLATERAL.

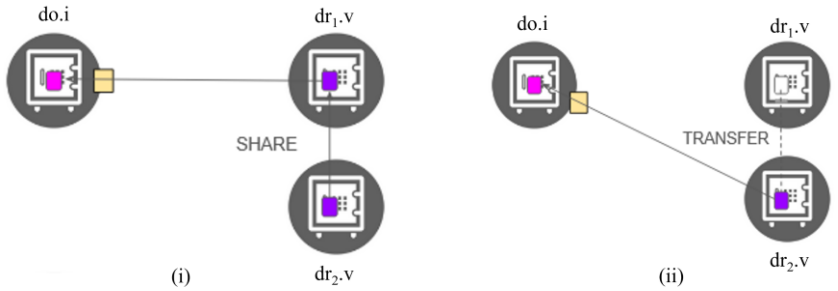

Figure 4. (i) Access tunnel for origin dr_2 to ground *do* after SHARE of SHARE. (ii) V-node ($dr_1.v$) transferred from dr_1 to dr_2.

[2] R. Bandura, M. McLean, and S. Sultan, "Unpacking the concept of digital public infrastructure and its importance for global development," *Center for Strategic and International Studies (CSIS)*, 2023.

[3] G. Gheorghiu et al., "The EU general data protection regulation implications for Romanian small and medium-sized enterprises," *Ovidius University Annals (Economic Sciences Series)*, vol. 18, no. 1, pp. 88–91, 2018.

[4] B. W. Schermer, B. Custers, and S. Van der Hof, "The crisis of consent: How stronger legal protection may lead to weaker consent in data protection," *Ethics and Information Technology*, vol. 16, no. 2, pp. 171–182, 2014.

[5] B. Ayyapane, R. Vaidyanathan, S. Srinivasa, S. Upadhyaya, and S. Vivek, "Consent service architecture for policy-based consent management in data trust," in *Proceedings of the 7th Joint International Conference on Data Science & Management of Data (11th ACM IKDD CODS and 29th COMAD)*, ACM, 2024.

[6] K. Houser and J. W. Bagby, "The data trust solution to data sharing problems," *Vanderbilt Journal of Entertainment & Technology Law*, 2023.

[7] S. Shrivastava and T. Srikanth, "A comprehensive consent management system for electronic health records in the healthcare ecosystem," in *Information Security and Privacy in Smart Devices: Tools, Methods, and Applications*, IGI Global, 2023, pp. 194–233.

[8] S. Stalla-Bourdillon, G. Thuermer, J. Walker, L. Carmichael, and E. Simperl, "Data protection by design: Building the foundations of trustworthy data sharing," *Data & Policy*, vol. 2, e4, 2020.

[9] D. Tith et al., "Patient consent management by a purpose-based consent model for electronic health record based on blockchain technology," *Healthcare Informatics Research*, vol. 26, no. 4, pp. 265–273, 2020.

[10] T. van den Broek and A. F. van Veenstra, "Modes of governance in inter-organizational data collaborations," in *ECIS 2015 Completed Research Papers*, 2015, pp. 0–12. DOI: 10.18151/7217509.

[11] I. Jussen et al., "Issues in inter-organizational data sharing: Findings from practice and research challenges," *Data & Knowledge Engineering*, vol. 150, p. 102 280, 2024, ISSN: 0169-023X.

[12] J. S. Park, R. Sandhu, and G.-J. Ahn, "Role-based access control on the web," *ACM Transactions on Information and System Security (TISSEC)*, vol. 4, no. 1, pp. 37–71, 2001.

[13] R. Abdunabi, M. Al-Lail, I. Ray, and R. B. France, "Specification, validation, and enforcement of a generalized spatio-temporal role-based access control model," *IEEE Systems Journal*, vol. 7, no. 3, pp. 501–515, 2013. DOI: 10.1109/JSYST.2013.2242751.

[14] V. C. Hu et al., "Guide to attribute based access control (abac) definition and considerations (draft)," *NIST special publication*, vol. 800, no. 162, pp. 1–54, 2013.

[15] V. C. Hu, D. R. Kuhn, D. F. Ferraiolo, and J. Voas, "Attribute-based access control," *Computer*, vol. 48, no. 2, pp. 85–88, 2015.

[16] B. S. Radhika, N. V. N. Kumar, and R. K. Shyamasundar, "Towards unifying rbac with information flow control," in *Proceedings of the 26th ACM Symposium on Access Control Models and Technologies*, 2021, pp. 45–54. DOI: 10.1145/3450569.3463570.

[17] B. S. Radhika, N. V. N. Kumar, and R. K. Shyamasundar, "Samyukta: A unified access control model using roles, labels, and attributes," in *Information Systems Security*, Cham:

Springer Nature Switzerland, 2022, pp. 84–102, ISBN: 978-3-031-23690-7. DOI: 10.1007/978-3-031-23690-7_5.

[18] A. Karandikar, "What makes consent meaningful?" In *Companion Publication of the 16th ACM Web Science Conference*, 2024, pp. 42–46.

[19] M.-R. Ulbricht and F. Pallas, "Comafeds: Consent management for federated data sources," in *2016 IEEE International Conference on Cloud Engineering Workshop (IC2EW)*, 2016, pp. 106–111. DOI: 10.1109/IC2EW.2016.30.

[20] M. Casassa Mont, V. Sharma, and S. Pearson, "Encore: Dynamic consent, policy enforcement and accountable information sharing within and across organisations," HP Laboratories Technical Report, Tech. Rep., 2012.

[21] E. Olca and O. Can, "Dicon: A domain-independent consent management for personal data protection," *IEEE Access*, vol. 10, pp. 95 479–95 497, 2022. DOI: 10.1109/ACCESS.2022.3204970.

[22] J. Hyysalo, H. Hirvonsalo, J. Sauvola, and S. Tuoriniemi, "Consent management architecture for secure data transactions," Jul. 2016. DOI: 10.5220/0005941301250132.

[23] J. Asswad and J. Marx Gómez, "Data ownership: A survey," *Information*, vol. 12, no. 11, 2021, ISSN: 2078-2489. DOI: 10.3390/info12110465.

[24] D. Hart, "Ownership as an issue in data and information sharing: A philosophically based review," *Australasian Journal of Information Systems*, vol. 10, no. 1, Nov. 2002. DOI: 10.3127/ajis.v10i1.440.

[25] Ministry of Electronics and Information Technology, Government of India, *The digital personal data protection act, 2023*, https://www.meity.gov.in/static/uploads/2024/06/2bf1f0e9f04e6fb4f8fef35e82c42aa5.pdf, No. 22 of 2023. Gazette of India, August 11, 2023, 2023.

[26] A. He and R. Arcesati, "Data marketplaces and governance: Lessons from china," *Centre for International Governance Innovation (CIGI)*, 2023. [Online]. Available: https://www.cigionline.org/articles/data-marketplaces-and-governance-lessons-from-china/.

[27] Nordic Institute for Interoperability Solutions, *X-road® technology overview*, Accessed: 2024-12-17, 2024. [Online]. Available: https://x-road.global/x-road-technology-overview.

[28] A. Kalja, "The x-road project," *A project to modernize Estonia's national databases. Baltic IT&T review*, vol. 24, pp. 47–48, 2002.

[29] Web Science Lab, IIIT Bangalore, "Anumati consent management system," 2026. [Online]. Available: https://anumati.iiitb.ac.in/.