

REFERENCE ARCHITECTURE MODEL

Version 2026-3

IDS-RAM 2026-3

Publisher

International Data Spaces Association
Emil-Figge-Str. 80
44227 Dortmund
Germany

Editor

Sebastian Steinbuss,
International Data Spaces Association

Copyright

International Data Spaces Association,
Dortmund, Germany, 2026



This work is licensed under a Creative Commons Attribution 4.0 International License.

Authors and Contributors

- Sebastian Steinbuss, IDSA
- Ilknur Chulani, IDSA
- Petteri Kivimäki, Nordic Institute of Interoperability Solutions
- Markus Spiekermann, Fraunhofer ISST
- Peter Koen, Microsoft
- Andreas Krimbacher, nexyo
- Felix Larrinaga Barrenechea, Mondragon University
- Simon Lofthouse, Data Trust Company
- Achim Pascal Meyer, Sphin-X

Contents

1	IDS-RAM structure	2
1.1	IDS-RAM 2026-3	2
1.2	Join the IDS-RAM work	3
2	Introduction	4
2.1	Contributions	5
2.2	Terminology	5
3	Relation to other IDSA Documents	6
3.1	Manifesto of Data Spaces – The IDSA North Star	6
3.2	IDSA Rulebook – From Principles into Practice	6
3.3	IDSA Papers on Focus Topics – Depth on Key Challenges	7
3.4	IDS Reference Architecture Model – Technical Realization	7
4	Architecture Principles	8
4.1	Introduction - <i>Starting with the main concepts</i>	8
4.2	IDS-RAM Architectural Capabilities - <i>Turning data space concepts into technical guidance</i>	9
4.2.1	A. Data Space Governance Authority	9
4.2.2	B. Credentials & Claims	10
4.2.3	C. Cataloging	10
4.2.4	D. Contract Negotiation	10
4.2.5	E. Control Plane & Policy Enforcement	11
4.2.6	F. Data Plane	11
4.2.7	G. Observability	11
4.2.8	H. Optional Services	11
4.3	Mapping to technical specifications	11
5	Credentials & Claims	12
5.1	Key architectural concepts	12
5.2	Further resources	12
6	Cataloging	13
6.1	Key architectural concepts	13
6.2	Further resources	13
7	Contract Negotiation	14
7.1	Key architectural concepts	14
7.2	Further Resources	14
8	Data Transfer (Transfer Process)	15
8.1	Key architectural concepts	15
8.2	Control Plane	15
8.3	Data Plane	15
8.4	Policy Enforcement	16
8.5	Further Resources	16
9	Observability	17
9.1	Key architectural concepts	17
9.2	Further Resources	18

10 Mapping IDS-RAM Capabilities to Specifications	19
10.1 Technical specifications currently available for each RAM capability	19
10.2 Index of RAM architectural capabilities and relevant technical specifications	19
11 Architecture Pattern and Guidelines	20
11.1 Data Space Governance Authority (DSGA)	20
11.2 Catalogs	20
11.3 Observer	20
12 Data Space Governance Authority (DSGA)	21
12.1 Central and federated DSGA	21
12.1.1 Trade-offs of central and federated DSGA	22
12.2 Decentral DSGA	22
12.2.1 Trade-offs of decentral DSGA	23
13 Catalogs	24
13.1 Central and federated catalogs	24
13.1.1 Collecting metadata centrally: push versus pull	25
13.1.2 Trade-offs for central and federated catalogs	26
13.2 Decentral catalogs	26
13.2.1 Trade-offs for decentral catalogs	26
14 Outlook	28
15 IDSA Papers on Focus Topics – In depth guidance on key challenges	29
16 Value-Adding Services in Decentralized Data Spaces	30
16.1 Matchmaking Service	30
16.2 Billing Service	31

1 IDS-RAM structure

- Introduction
- Relation to other documents
- Architectural Principles
- Architectural Patterns
- Outlook
- Front Matter
- Focus Papers

1.1 IDS-RAM 2026-3

This IDS-RAM 2026-3 edition of the IDS Reference Architecture Model (IDS-RAM) captures the latest results of key concepts and technical aspects of data spaces, including IDSA's contributions to international standards and specifications.

The IDS-RAM is the main reference for implementing the ideas defined in the IDSA Rulebook. It translates governance models, data usage policies, and protocol specifications into clear logical components, defined interfaces, and consistent behavior patterns.

This version aims to support interoperability, scalability, and conformance across different environments. Instead of prescribing a single approach, it offers a structured design space that adapts to diverse needs while preserving the core IDSA data space model.

1.2 Join the IDS-RAM work

We are developing the next version of the Reference Architecture Model together with our community in the IDSA Working Group Architecture.

The IDSA Architecture Working Group defines and promotes architectural principles and implementation patterns that are firmly grounded in the IDSA core concepts as described in the IDSA Rulebook. Its mission is to guide architects and software engineers in designing and developing trusted, interoperable, and compliant data spaces.

To know more about the IDSA working groups please visit our home page and see the brochure for more details on how to onboard.

Please note: IDSA working group activities are reserved for members. Is your organization not a member of IDSA? Find more information here: [Become a Member](#)

2 Introduction

The emergence of data spaces as a key enabler for trustworthy data sharing has introduced a new class of data architecture principles. Within this landscape, the **International Data Spaces Association (IDSA)** provides a foundational Rulebook that defines the core principles, roles, and capabilities required to establish and operate such environments. However, turning these high-level principles into actionable, technical guidance requires a dedicated architectural framework. **This is the purpose of the IDS Reference Architecture Model (IDS-RAM).**

The IDS-RAM is the central compendium of technical documents that operationalizes the concepts defined in the IDSA Rulebook. It maps abstract governance models, data usage policies, and protocol specifications into concrete technical (logical) components, interfaces, and behavioral patterns. It does so with a clear focus: enabling interoperability, scalability, and conformance without prescribing a single implementation path. In other words, the IDS-RAM is not a blueprint but a design space—a structured but flexible guide that supports diverse requirements while preserving the integrity of the IDSA data space model.

This document is for system architects, software engineers, and infrastructure designers who are tasked with building or integrating components within a data space to enable business-driven data ecosystems. If you're looking to understand what makes a connector IDSA-compliant, how to build interoperable and integrable services, or how to maintain trust and policies in a decentralized environment—this is your technical guidance.

Central to the IDS-RAM are specifications such as the **Dataspace Protocol (DSP)** and the **Decentralized Claims Protocol (DCP)**. These protocols define how components communicate, how identities are exchanged and verified, and how policy-conformant data discovery and transfer is achieved. The IDS-RAM highlights these specifications, covering components, message formats, interaction sequences, and binding details. Further, it describes in depth how they are integrated with key capabilities like identity management, observability, data discovery, contract negotiation and secure data transfer. Details of such capabilities are maintained within dedicated documents under the purview of IDSA. The IDS-RAM document integrates and connects these documents to provide a comprehensive overview and a single source of truth for architecture models according to IDSA specifications.

To support real-world applicability, the IDS-RAM organizes its content into two major sections:

Capability Mapping: This section lists the essential capabilities enabled through data spaces—such as data discovery, policy enforcement, usage control, identity resolution, and observability. Each capability is analyzed from a technical perspective, detailing how it can be implemented within a compliant data space. The descriptions are aligned with the IDSA Rulebook and maintains references to the foundational concepts to highlight the strong relation between the two documents. The IDS-RAM content is however grounded in system-level detail, interaction patterns, documented expected behavior, and provides guidance on integration patterns of infrastructure and other technologies.

Architectural Best Practices: Recognizing the diversity of business and technical requirements across domains, the IDS-RAM does not enforce a singular architecture. Instead, it presents validated patterns and warns against known anti-patterns, guiding implementers through the architectural decisions while setting up or operating a data space. Whether the goal is to create a lightweight edge connector, operate a multi-tenant

marketplace, or integrate with existing enterprise systems, the IDS-RAM aims to outline the architectural considerations and trade-offs—while maintaining compatibility with the IDSA model.

The IDS-RAM is intentionally neutral with respect to implementations. It refrains from endorsing any specific codebase or vendor product. Where useful, illustrative code snippets and configuration examples are included to clarify complex concepts and show practical realizations. For this, available open source projects (such as those maintained under the Eclipse Dataspace Working Group (EDWG) initiative) are referenced.

Importantly, the IDS-RAM is a *living* document due to continuous updates of IDSA technical documents this document refers to. Therefore, rather than locking into static version cycles, it evolves incrementally with latest releases of technical documents. Changes of these documents are managed through the corresponding GitHub repositories, with full traceability of modifications and clear visibility into the rationale behind decisions. Periodic release tags of the IDS-RAM document however will provide stable points of reference, allowing contributors and adopters to align their work with a consistent snapshot of the evolving model.

2.1 Contributions

The IDSA Working Group Architecture creates and maintains the IDS-RAM. Its mission is to guide architects and software engineers in designing and developing trusted, interoperable, and compliant data spaces.

To know more about the IDSA Working Group Architecture, please visit our home page. The IDSA working groups brochure provides details on how to get involved and how to contribute.

Please note: IDSA working group activities are reserved for members. Find more information about IDSA membership here: [Become a Member](#)

2.2 Terminology

IDS-RAM uses terms as defined in the IDSA Glossary.

3 Relation to other IDSA Documents

The International Data Spaces Association (IDSA) provides a comprehensive body of documentation that serves as both a strategic compass and a practical guide for building interoperable, trusted data spaces. IDSA documents follow a clear top-down structure: starting with vision and principles, moving to governance and requirements, narrowing into thematic clarity, and concluding with technical realization. This structure ensures coherence across conceptual, operational, and technical domains, enabling organizations to confidently adopt data space principles at scale.

Level	Document	Purpose
Vision	Manifesto of Data Spaces	Defines core principles and motivation
Governance	IDSA Rulebook	Establish requirements and roles
Thematic guidance	IDSA Papers on Focus topics	Explore evolving topics in depth
Technical Architecture	IDS-RAM	Enables implementations

3.1 Manifesto of Data Spaces – The IDSA North Star

At the top of the IDSA document hierarchy stands the IDSA Manifesto, a concise yet powerful declaration of purpose. It describes IDSA's ambition to shape a trusted global data economy founded on sovereignty, trust, and interoperability. The Manifesto serves three essential functions:

- Sets the shared aspiration: to enable data sharing ecosystems where organizations retain control over their data and use it responsibly for collective innovation.
- Establishes guiding values: trust, fairness, interoperability, and sovereignty, which frame all other IDSA documents.
- Calls for collective action: encouraging industry, research, and policymakers to collaborate toward Trusted Data Sharing based on open standards and decentralized architectures.

Unlike detailed guidance documents, the Manifesto is deliberately high level. It articulates “why” data spaces matter and builds a shared identity and strategic direction for the IDSA community. All subsequent documents—including the Rulebook, Focus Topics, and the Reference Architecture Model (IDS-RAM)—are grounded in the principles laid out here.

3.2 IDSA Rulebook – From Principles into Practice

Guided by the ideals of the Manifesto, the IDSA Rulebook answers “**WHAT**” needs to be done to translate vision into concrete requirements and governance models. It supports the creation, operation, and growth of data spaces by distinguishing mandatory requirements from optional, value-adding practices. Its scope spans technical, commercial, and legal dimensions:

- Common technical guidance, including functional requirements and specifications.
- Recommendations for applying IDSA technical artefacts and for alignment with partner frameworks.
- Operational guidance for collaboration, roles, and processes that enable data space ecosystems.

- Perspectives on implementing and complying with international legal and regulatory obligations to facilitate trusted, cross-border data sharing.

The Rulebook acts as the normative foundation of IDSA. It does not specify implementation technologies but clearly defines conditions for trust—usage control, contractual assurance, and transparent operations—thus linking data value creation with accountability.

3.3 IDSA Papers on Focus Topics – Depth on Key Challenges

While the Manifesto inspires and the Rulebook governs, documents for individual focus topics dive deeper into specific adoption within data spaces. These concise thematic modules ensure that the documentation can evolve with a fast-changing environment without overloading core documents. Current topics include:

- Identity & Trust – decentralized identity management and verifiable credentials.
- Interoperability – semantic, organizational, and technical interoperability models.
- Observability – monitoring data space operations while respecting sovereignty.
- Agentic AI and LLM – integration of MCP and data space in the context of agentic web

Each Focus Topic is anchored in the Rulebook and consistent with the Manifesto, providing reusable patterns and best practices. Their modular structure ensures efficient maintenance and allows new topics to be integrated as technology and regulation evolve.

3.4 IDS Reference Architecture Model – Technical Realization

While previous documents define what to implement, the IDS Reference Architecture Model (IDS-RAM) explains **“HOW”** to build it. It is the central technical compendium that transforms more abstract governance concepts into implementable architecture. The IDS-RAM introduces:

- Interaction models between logical components
- Protocol specifications like the Dataspace Protocol (DSP) and Decentralized Claims Protocol (DCP).
- Capabilities for identity, data discovery, policy enforcement, contract negotiation, and secure transfer.
- Integration patterns that support interoperability without imposing a single technology stack.

The IDS-RAM is not prescriptive software architecture; it is a design framework that accommodates diverse implementation paths. It supports system architects and developers by connecting high-level requirements from the Rulebook with concrete deployment scenarios. Its two core sections—Capability Mapping and Architectural Best Practices—make it an indispensable engineering guide for building sovereign, trusted data ecosystems.

4 Architecture Principles

This chapter provides architectural insights on the main technical concepts of data spaces described in the IDSA Knowledgebase and in the IDSA Rulebook and explains how to realize them using technical specifications such as the Dataspace Protocol (DSP) and the Decentralized Claims Protocol (DCP).

4.1 Introduction - *Starting with the main concepts*

The IDSA Rulebook and the ISO/IEC 20151-1 Standard “Information technology — Cloud computing and distributed platforms — Dataspace concepts and characteristics” define Data Spaces as:

Environment enabling trusted data sharing between participating parties, based on an agreed governance framework, along with an agreed set of policies, semantic models, standardized protocols, processes, and facilitating services.

Based on this, data spaces have the following main concepts and characteristics, as explained in further detail in the IDSA Rulebook and ISO/IEC 20151-1:

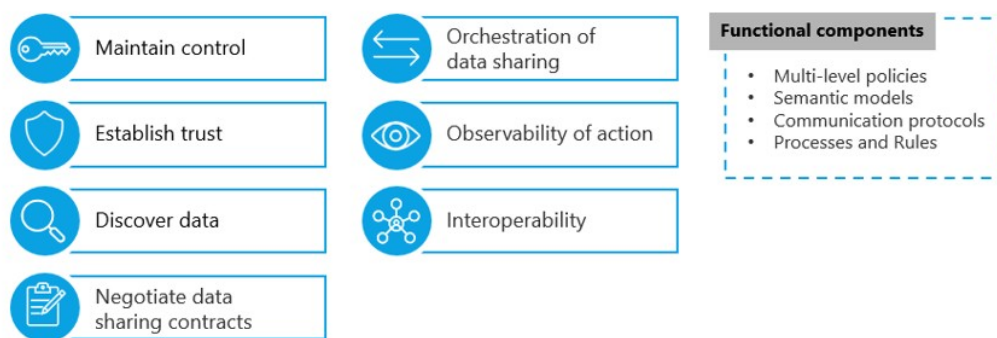


Figure 1: Figure 1: Data space concepts from the IDSA Rulebook and ISO/IEC 20151.

The following diagram maps the relationships between these concepts, not intending to follow a process/runtime flow order, but focusing on highlighting relationships and dependencies between them.

Please note that some essential characteristics such as **maintaining control** and **interoperability** are not depicted in this picture, as these are cross-cutting concepts which require actions in almost each block. They have been omitted from the diagram for the sake of readability.

Multi-level policies (e.g., membership policies at data space level, access policies for controlled discovery of metadata, data sharing contract policies, data use policies) needed in a data space are depicted vertically in this picture to express how they are applied in an overarching way across different levels.

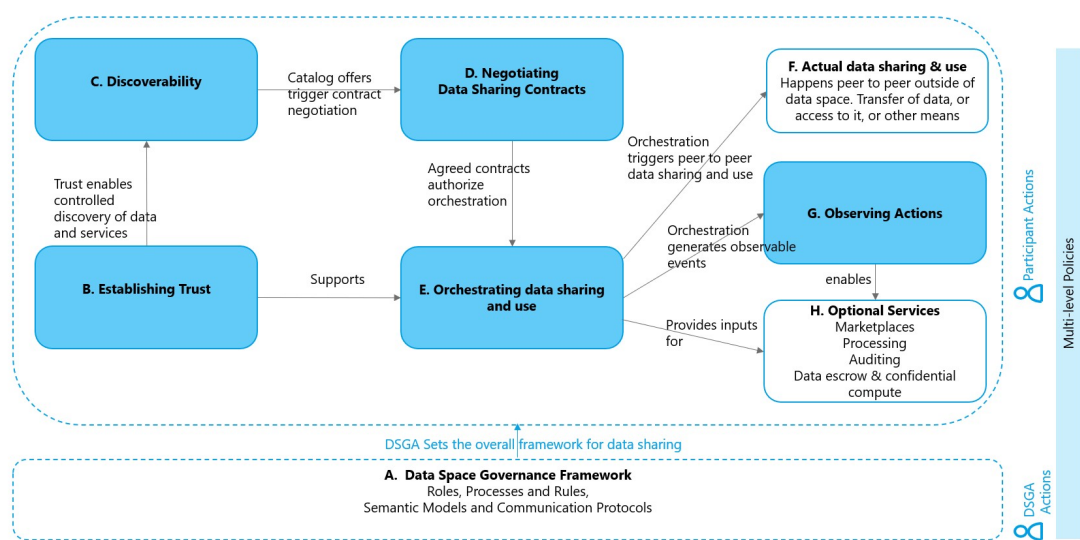


Figure 2: Figure 2: Visual Map for Data Space Concepts from IDSA Rulebook and ISO/IEC 20151-1

4.2 IDS-RAM Architectural Capabilities - *Turning data space concepts into technical guidance*

While the IDSA Rulebook defines the essential data space concepts, the IDS-RAM operationalizes these high-level principles into technical guidance in the form of logical components and behavioural patterns.

The architectural capabilities defined in IDS-RAM represent the functional building blocks necessary for interoperable and trusted data sharing. These are based on the current technical approaches at hand and are centered around the following key technical specifications and standardization work items to ensure interoperability:

- **Dataspace Protocol (DSP, ISO/IEC DIS 26450)** which defines publication/discovery, agreement negotiation, and data access interactions for interoperable data sharing.
- **Decentralized Claims Protocol (DCP, ISO/IEC DIS 26451)** which provides an overlay for organizational identity and trust/credential verification while preserving privacy.
- **Data Plane Signaling** which specifies interoperable control plane/data plane communication.

However, the architectural capabilities can be implemented in other approaches. Future versions of the IDS-RAM may include such approaches.

The visual map of data space concepts provided earlier then evolves into the following view of architectural capabilities:

4.2.1 A. Data Space Governance Authority

Sets the overall framework for data sharing between participants by establishing a trust framework, roles, processes, and rules.

!!! note The DSGA is not a technical artefact for the IDS-RAM, but a functional role described in the IDSA Rulebook. IDS-RAM focuses mostly on the participant side; however, the governance framework is included for the sake of completeness.

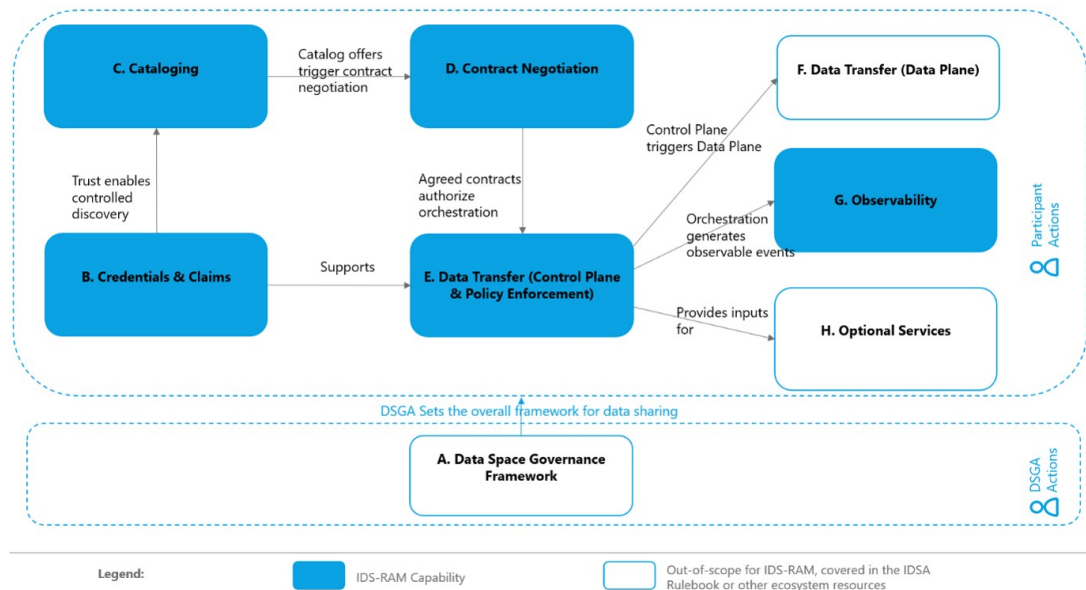


Figure 3: Figure 3: IDS-RAM Architectural Capabilities

!!! tip “See also” IDSA Rulebook pages on DSGA and IDS-RAM Architectural patterns for DSGA.

4.2.2 B. Credentials & Claims

Establish trust-relevant identity and attributes used throughout subsequent capabilities on the participant side. This capability is closely linked with the DCP specification.

!!! tip “See also” IDSA Rulebook pages on Trust, Dataspace Trust Frameworks, Attributes and claims, Identity, Decentralized onboarding and working draft paper on Identifiers in Data Spaces.

4.2.3 C. Cataloging

Supports the discovery of data and services through metadata publication and controlled visibility.

Cataloging capability is based on the DSP Catalog Protocol specification, which makes use of DCAT and application profiles.

!!! tip “See also” IDSA Rulebook page Data Discovery Services and IDS-RAM Architectural patterns for Catalogs.

4.2.4 D. Contract Negotiation

Formalizes rights and obligations into enforceable data sharing agreements.

This capability is realized by the DSP Contract Negotiation Protocol.

!!! tip “See also” IDSA Rulebook page Data Sharing.

4.2.5 E. Control Plane & Policy Enforcement

Orchestrates and authorizes the execution of agreements and coordinates with data plane mechanisms using interoperable protocol bindings.

The control plane part of the DSP Transfer Process Protocol and Data Plane Signaling specifications enable this capability.

!!! tip “See also” IDSA Rulebook pages Data Sharing, Planes and Policies.

4.2.6 F. Data Plane

Executes data transfer/access under the constraints of the data sharing agreement and orchestration decisions.

This corresponds to the data plane referred to in the DSP Transfer Process Protocol specification.

!!! note Since actual data sharing happens outside of the data space, in a peer-to-peer manner between participants, this block is out of scope for IDS-RAM but remains part of the end-to-end story of data sharing.

!!! tip “See also” IDSA Rulebook pages Data Sharing and Planes.

4.2.7 G. Observability

Provides evidence and accountability for actions taken during Contract negotiation, Orchestration (control plane), Data access and use (data plane). Observing actions supports trust frameworks and compliance needs without requiring centralization of data content. Observability may also provide inputs to optional services such as auditing.

For simplicity, the diagram only depicts the relationship with the Control Plane capability.

!!! tip “See also” IDSA Rulebook page Observability and IDSA position paper Observability in Data Spaces.

4.2.8 H. Optional Services

Represents ecosystem extensions (e.g., marketplaces, processing services, auditing services, escrow/confidential compute), which may be provided by participants subject to the data space governance framework. In the diagram, optional services “execute and enforce” the data plane and “provide inputs” into governance and framework elements.

Optional services are out of scope for IDS-RAM, while still relevant for the overall landscape of data spaces.

!!! tip “See also” Focus paper Value adding services

4.3 Mapping to technical specifications

A mapping of the RAM capabilities to technical specifications may be found in the article on the mapping to technical specifications.

5 Credentials & Claims

Based on the trust framework Data Space Governance Authority establishes in a data space, Credentials and Claims provide the architectural means for **identity, trust, and authorization** for participants. It helps ensure that all participants and their technical components (Connectors/participant agents) are uniquely identified and their security posture is verifiable.

This capability is realized through the **Decentralized Claims Protocol (DCP)**, which acts as an interoperable overlay to Dataspace Protocol (DSP).

DCP:

- Enables participants to issue, present, and verify verifiable credentials
- Supports decentralized trust anchors and privacy-preserving verification
- Integrates with DSP interactions to gate discovery, negotiation, and orchestration

Credentials and claims ensure that trust decisions are **evidence-based, contextual, and automated**, without introducing centralized identity providers.

5.1 Key architectural concepts

- **Decentralized Identity:** Implemented via the **Decentralized Claims Protocol (DCP)**, which utilizes self-issued identity tokens and Decentralized Identifiers (DIDs).
- **Verifiable Credentials:** Participants manage organizational identities and “claims” (e.g., certification, membership) without relying on a single central authority. This allows for a “trust-by-design” approach where trust anchors can be federated.
- **Authentication:** Connectors use the Authorization header in HTTPS bindings to pass these identity tokens, ensuring that only authenticated “Participant Agents” can access services.

5.2 Further resources

- **IDSA Rulebook Guidance:**
 - Trust
 - Decentralization: Trust Frameworks and Credential Management
 - Dataspace Trust Frameworks
 - Establishing Trust
 - Attributes and Claims
 - Identity
 - Decentralized Patterns: Onboarding
- **Technical specifications:**
 - Decentralized Claims Protocol (DCP)
 - W3C Verifiable Credentials Data Model
 - Decentralized Identifiers (DIDs)
- **Related Focus Papers:**
 - Draft paper on Identifiers in Data Spaces
- **IDS-RAM Architectural Patterns:**
 - DSGA Patterns

6 Cataloging

Cataloging describes the architectural capability to **advertise data assets and data services** in a data space without exposing the data itself.

From a protocol perspective, cataloging is realized through **DSP catalog messages**, which allow participants to publish **machine-readable metadata** describing datasets, services, and associated usage policies. These catalogs form the basis for discovery and negotiation.

Cataloging:

- Is **metadata-only** and policy-filtered
- Does not imply authorization or data access
- Enables automated reasoning by downstream contract-negotiation state machines

Bindings (e.g. HTTP-based bindings) allow these catalog messages to be transported independently of the underlying data systems, preserving interoperability across implementations.

6.1 Key architectural concepts

- **Dataset Advertising:** Realized through the **Catalog Protocol**. Providers publish a Catalog containing Datasets, Offers, and Services.
- **Standardized Metadata:** Uses the DCAT (Data Catalog Vocabulary) and JSON-LD for semantic interoperability.
- **Discovery:** Consumers can request a full catalog or specific dataset metadata via standardized HTTPS endpoints (e.g., /catalog/request).

As explained in the IDSA Rulebook, this capability can be implemented as a managed service by one or more selected participants, hosted by the data space governance authority, or operated in a fully decentralized fashion by every participant that offers data contracts (see the visual representation of various implementation designs of the DSGA above). The type of catalog architecture used depends on the design of the data space as well as the needs and capabilities of the participants.

6.2 Further resources

- **IDSA Rulebook Guidance:**
 - Cataloging (Data Discovery Services)
- **Technical specifications:**
 - Dataspace Protocol: Catalog Protocol
 - Data Catalog Vocabulary (DCAT)
- **Related Focus Papers:**
 - IDSA Position Paper on Semantic Interoperability
 - Particularly the following sections in the paper provide further details relevant for cataloging:
 - ✱ 3.2 Abilities of the Dataspace Protocol
 - ✱ 3.3 Data Catalog Vocabulary (DCAT) - Version 3
 - ✱ 3.4 Open Digital Rights Language (ODRL)
- **IDS-RAM Architectural Patterns:**
 - Catalog Patterns

7 Contract Negotiation

Perhaps the most critical capability, this allows two participants to agree on the specific terms (Policies) under which data will be shared and to **establish legally and technically meaningful agreements** governing data sharing.

This capability is defined by **DSP negotiation messages and state machines**, which formalize the lifecycle of a data sharing agreement through states such as request, offer, acceptance, agreement, verification, and finalization.

- Negotiation is **stateful and asynchronous**
- Both parties maintain a synchronized view of the contract state
- Contract messages reference policies, assets, and constraints expressed in machine-readable form

The result of contract negotiation is an agreement that can be **referenced by orchestration and observability mechanisms**, without embedding the contract logic into the data plane.

7.1 Key architectural concepts

- **Automated Negotiation:** The **Dataspace Protocol: Contract Negotiation Protocol** manages a state machine (Request → Offer → Agreement) that leads to a legally and technically binding Agreement.
- **Policy Expression:** Uses the ODRL (Open Digital Rights Language) to define **Permissions** (what you can do), **Prohibitions** (what you cannot do), and **Duties** (obligations like payment or logging).
- **Agreement Verification:** Before data is transferred, the agreement is verified to ensure both parties are still in compliance with the negotiated terms.

7.2 Further Resources

- **IDS Rulebook Guidance**
 - Data Sharing
 - Planes
 - Policies: Contract Policies
- **Focus Papers**
 - IDS Position Paper: Semantic Interoperability in Data Spaces
 - ✱ Particularly relevant: Section 3.4, *Open Digital Rights Language (ODRL)*, for the definition of usage terms for the contract
- **Specifications**
 - Dataspace Protocol: Contract Negotiation Protocol Specification
 - ODRL Information Model
 - ODRL Vocabulary & Expression

8 Data Transfer (Transfer Process)

Data Transfer orchestrates the **actual exchange or access of data** between data space participants once a contract has been established.

This capability does **not** define how data is transferred; instead, it defines **transfer process messages** that coordinate *when* and *under which agreement* data transfer may occur.

- Data transfer is **peer-to-peer** between participants
- Multiple transfer methods are supported (files, APIs, streams, confidential compute)
- Transfer protocols are selected and bound outside DSP

This design decouples governance and interoperability (control plane) from performance- and domain-specific data exchange (data plane).

8.1 Key architectural concepts

- **Transfer State Machine:** The **Dataspace Protocol: Transfer Process Protocol** governs the lifecycle of a data transfer (Requested → Started → Completed/Terminated).
- **Separation of Concerns:** It distinguishes between the *Control Plane* (which handles the signaling and “paperwork”) and the *Data Plane* (the actual technical “pipe” like HTTP, MQTT, or S3).
- **Transfer Signaling:** The **Dataplane Signaling** specification defines how the Control Plane instructs the Data Plane to open, monitor, or close a transfer channel based on an active Agreement.

8.2 Control Plane

Control plane part of the Data Transfer is responsible for **governing and coordinating data sharing**, but not for handling the data itself.

Architecturally, this capability encompasses:

- Transfer process coordination
- Policy evaluation and claims verification

It is realized by the **Control plane** part of the **Transfer process** protocol of the **DSP**. Together with **DCP**, defining the messages, states, and sequences required to reach authorized transfer initiation.

8.3 Data Plane

The Data Plane is responsible for the **execution of data transfer and data use**, in accordance with the agreements established by the control plane.

Architecturally:

- The data plane is **outside the scope of the Dataspace Protocol and IDS-RAM**
- It is implemented by participant-controlled data services
- It may include data management systems, APIs, compute environments, or streaming infrastructures

The separation between control plane and data plane allows data spaces to support **heterogeneous technologies and business models**, while preserving a common governance and interoperability layer.

8.4 Policy Enforcement

Policy Enforcement describes how **usage policies and contractual obligations** are applied across data space interactions.

From an architectural perspective, policy enforcement is a **shared responsibility**:

- The data space control plane evaluates policies during discovery, negotiation, and orchestration
- Data management services enforce policies during actual data access and use

Policies are expressed in machine-readable form and referenced by DSP messages, but enforcement may occur both **technically** (e.g. access control, usage restrictions) and **organizationally** (e.g. contractual compliance).

This division ensures that data spaces can scale without assuming control over participant infrastructure.

8.5 Further Resources

- **IDSA Rulebook Guidance**
 - Data Sharing
 - Planes
 - Policies
- **Specifications**
 - Dataspace Protocol: Transfer Process Protocol
 - Data Plane Signaling
- **See also**
 - Data Plane Core - Data plane SDKs for several programming languages

9 Observability

This is the capability for **observing data sharing transactions** to ensure transparency, accountability and trust.

Observability:

- Is **policy-driven and role-based**
- Can be centralized, federated, or decentralized
- Relies on protocol-level events rather than raw data inspection

Observability strengthens trust by enabling verification of expected behavior without violating data sovereignty.

A major architectural shift in this version of IDS-RAM is the deprecation of the centralized **Clearing House** component and the move towards a more focused approach for the observability concept with the introduction of more flexible and decentralized **Observer** role.

9.1 Key architectural concepts

- **The Observer Role:** Observability is no longer tied to a single central component. Instead, it is a role that can be fulfilled by one or multiple parties, i.e. the data sharing transaction participants (Provider or Consumer) and/or an authorized third party.
- **Purpose-Driven Monitoring:** Unlike general system monitoring, IDS Observability is “targeted and purposeful.” It specifically monitors the **Dataspace protocol activities and states** to verify:
 - Was the contract successfully concluded?
 - Was the data actually transferred?
 - Was the transaction executed according to the negotiated policies?

Observability is closely linked to other concepts such as Provenance and Traceability. However, the scope of this capability is not solving end-to-end observability challenges in data space ecosystems. These would imply very diverse requirements and therefore would be best left to use-case implementations to leave room for the flexibility they would need.

It is important to note that at the technical layer, this capability may be implemented by approaching the sharing of observability data just like another data sharing contract, however, setting up the necessary business processes and governance rules are the really nec-

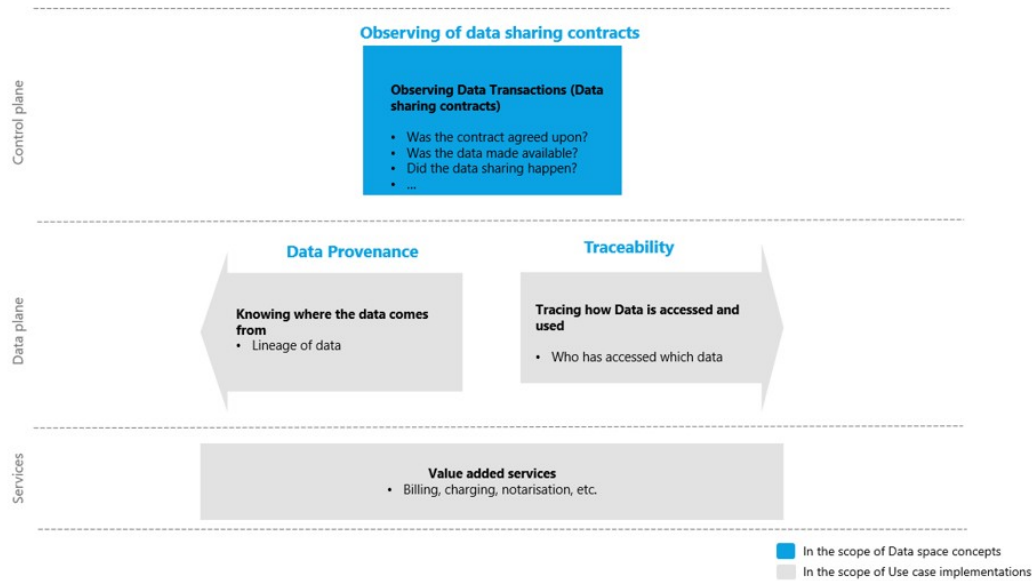


Figure 4: Figure 1: Scope: Observability of Data sharing contracts, not end-to-end observability in data ecosystems

Data Space Governance Authority	Data Space Governance rules for participation management
Data Space Participant	Business processes and rules for data sharing governance
Data space participants and/or any observing 3 rd party	Technical Implementation

essary steps to truly achieve observability.

9.2 Further Resources

- **IDSA Rulebook Guidance**
 - Observability
- **Focus Papers**
 - IDSA Position Paper: Observability in Data Spaces
- **Specifications**
 - Dataspace Protocol - See State Machines for what can be observed for each sub-protocol

10 Mapping IDS-RAM Capabilities to Specifications

10.1 Technical specifications currently available for each RAM capability

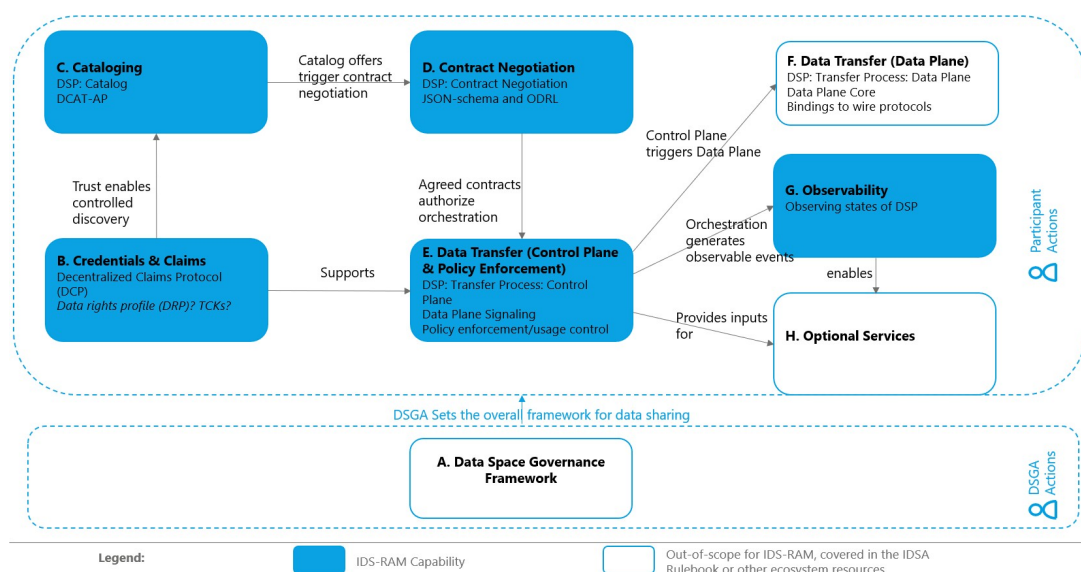


Figure 5: Figure 4: IDS-RAM Architectural Capabilities and Technical Specifications

10.2 Index of RAM architectural capabilities and relevant technical specifications

Architectural Capability	Technical Specifications
Credentials & Claims	Decentralized Claims Protocol (DCP) W3C Verifiable Credentials Data Model Decentralized Identifiers (DIDs)
Cataloging	Dataspace Protocol: Catalog Data Catalog Vocabulary (DCAT)
Contract Negotiation	Dataspace Protocol: Contract Negotiation ODRL Information Model ODRL Vocabulary & Expression
Data Transfer	Dataspace Protocol: Transfer Process Dataplane Signaling
Observability	Observing Dataspace Protocol State Machines

11 Architecture Pattern and Guidelines

This chapter provides more concrete architecture guidelines on how to design different architecture patterns within a data space. Beside the introduction and explanation, trade-offs are highlighted.

11.1 Data Space Governance Authority (DSGA)

The Data Space Governance Authority (DSGA) patterns article provides architectural options for realizing a Data Space Governance Authority. The section covers central, federated and decentral approaches.

11.2 Catalogs

The Catalog patterns article provides architectural options for realizing dataset publication and discovery in a data space. The section covers central, federated and decentral approaches.

11.3 Observer

TODO: Different architectural options for implementing Observer role, such as Federated or Central Escrow vs Decentral observability. Contributions are welcome.

Please also refer to the IDSA Rulebook page Observability and IDSA position paper Observability in Data Spaces

12 Data Space Governance Authority (DSGA)

This section defines different architectural options for realizing a Data Space Governance Authority (DSGA). Please also refer to the IDSA Rulebook page DSGA.

The Rulebook defines the DSGA as a functional role and distinguishes it from the Data Space Governance Framework it establishes and enforces. It also defines the decentralization principle that guides the architectural realization of the DSGA. This section does not repeat those definitions, but focuses on the architectural options for realizing the DSGA and their trade-offs.

From an architectural perspective, the governance framework itself can remain the same across all options. What varies is how the DSGA function is realized and where responsibility for implementing and enforcing the Governance Framework resides. Those choices affect participant autonomy, resilience, accountability, operational cost, and neutrality. The patterns can be compared along four dimensions:

1. Governance authority – who establishes and changes the Governance Framework, and admits or removes participants.
2. Trust-root topology – how trust anchors and credential issuers are organized and distributed.
3. Runtime dependencies – whether trust establishment or participant discovery requires synchronous interaction with shared services.
4. Control, neutrality, and impartiality – whether any participant can control, favour, or exclude others, and what safeguards exist to prevent this.

The options form a spectrum rather than discrete choices. As governance authority and information are distributed to participants and mandatory runtime dependencies on shared services are reduced, a data space moves toward the decentralized end of the spectrum.

12.1 Central and federated DSGA

Here the DSGA function is embodied in shared infrastructure. In the central variant a single authority — typically a dedicated legal entity such as an operations company — provides the mandatory governance services consumed by all participants. In the federated variant several authorities operate under a common trust list (a shared set of trusted trust anchors and credential issuers), distributing the function without fully decentralizing it. Both rely on services that participants depend on at runtime.

Not all governance services are consumed in the same way. Some are used only during participant lifecycle events, such as onboarding services that register new participants. Others are consulted during data-sharing interactions between participants, for example registries that resolve information about participants and their connectors. These runtime services introduce a synchronous dependency and can become a single point of failure if their availability is required for every interaction.

This dependency can be mitigated in two ways. One approach is to operate the governance service with high availability. A more decentralized approach is to eliminate the synchronous dependency by distributing the required configuration and reference data to participants. Each participant caches the governance data locally and evaluates it during interactions instead of querying the governance service at runtime. This avoids a runtime dependency on centralized components and aligns with the DTF guidance to avoid assuming synchronous communication or the continuous availability of centralized services. Enabling caching moves the pattern toward the decentralized end of the spectrum,

since each participant then holds the governance data it needs locally.

This pattern is often appropriate where a data space needs a clear single point of accountability, simplified and consistent onboarding, an authoritative source of truth, or where regulation requires a designated responsible entity.

12.1.1 Trade-offs of central and federated DSGA

This pattern provides clear accountability and contractual responsibility through a shared authority. It simplifies and standardizes onboarding, dispute resolution, and governance processes, provides an authoritative source of truth, reduces coordination effort for participants, and is straightforward to bootstrap.

The trade-off is greater reliance on centralized governance. Shared services may become runtime dependencies and potential single points of failure unless mitigated, for example through high availability or local caching of governance data. Centralization can also reduce participant autonomy and create neutrality risks if the shared authority can favour or exclude participants. As required by the Rulebook, such centralization should be explicitly justified and complemented by appropriate safeguards, including transparent governance processes, audit rights, revocation mechanisms, and, where feasible, alternative providers.

12.2 Decentral DSGA

Here the DSGA function is not embodied in shared infrastructure. Governing authority is not delegated to a designated party but exercised collectively: the governance framework — the DTFs in force together with any data-space-specific provisions — is distributed to every participant, who applies and enforces it locally, and decisions about the rules are taken jointly rather than by any single actor.

Trust is not rooted in a single authority or central identity provider. Instead, participants accept credentials from multiple independent issuers and each chooses which trust anchors and issuers to rely on. There is no mandated runtime dependency on a shared service: each participant holds the governance data it needs locally and, where an authoritative statement such as a revocation check is required, queries an external Oracle of its own choosing. As a result, no single actor holds power over the others and there is no shared component whose failure halts the data space; any shared component that does appear is optional and chosen by the participants.

Onboarding reflects the same collective logic: either a one-time onboarding entity issues a membership credential at join time with no ongoing dependency, or membership is established peer-to-peer by presenting claims to an existing participant that evaluates the membership rules directly. Decentralized onboarding mechanisms are described in the Rulebook's Decentralized Onboarding Patterns.

One possible realization of this pattern is a permissioned distributed ledger (DLT) operated jointly by the participants. In this case, the legislative aspect of the DSGA function is exercised collectively: governance decisions, such as adopting or amending DTFs, changing governance parameters, or admitting and removing participants, are proposed and voted through the ledger. Smart contracts encode the agreed governance rules and execute the agreed outcome, for example by updating the accepted set of trust anchors or the revocation registry. Typical functions implemented through the ledger include collective governance and voting, tamper-evident registries for trust anchors, credential issuers, membership, and revocation status, as well as notarization anchors supporting audit and

observability. Because the ledger is jointly operated and replicated across participants, no single participant controls these governance functions and there is no single point of failure.

Three architectural boundaries preserve the decentral characteristics of this pattern. First, data sharing remains peer-to-peer and off-ledger: only governance state, decisions, and cryptographic commitments (hashes) are recorded, never the shared data itself, and generally not participants' credentials or claims. Second, trust evaluation remains local. Participants evaluate governance information using their locally maintained copy of the ledger rather than querying a shared service during each interaction, avoiding synchronous runtime dependencies in line with the DTF guidance against assuming global information or continuously available centralized services. Third, because ledger contents are replicated across participants, personal or otherwise sensitive information should not be stored on the ledger. Instead, off-ledger storage should be combined with ledger commitments and, where appropriate, zero-knowledge proofs to support selective disclosure in accordance with the attribute-based trust model.

A smart contract in this context is executable code deployed on the distributed ledger. It is distinct from the IDSA data sharing contract negotiated bilaterally between participants during contract negotiation, and the two should not be conflated.

This pattern is appropriate where participant autonomy, resilience, neutrality, and the absence of mandatory runtime dependencies on shared governance services are primary objectives. It is particularly suitable where governance can be exercised collectively and participants are willing to share responsibility for operating the governance framework.

12.2.1 Trade-offs of decentral DSGA

This pattern provides maximal participant autonomy and agency, neutrality with no single participant holding power over others, the absence of a single point of failure, freedom to choose providers and avoid lock-in, and strong resilience through distributed governance.

The trade-offs are increased coordination and integration effort for participants and greater overall architectural complexity. Achieving consistency and interoperability is more challenging without shared services and depends on shared semantics, such as a common claim vocabulary. Reconciliation and dispute resolution require well-defined governance rules, escalation paths, and appropriate consensus thresholds.

13 Catalogs

This section provides architectural options for realizing dataset publication and discovery in a data space. For the underlying metadata model and discovery requirements, see the IDSA Rulebook's data discovery guidance and the Dataspace Protocol Catalog Protocol. This section does not repeat them but focuses on the architecture patterns and their trade-offs.

Some aspects of discovery are the same in every pattern and are treated here as a shared baseline. Metadata is expressed with DCAT: a Catalog contains Datasets, each with one or more Distributions and an associated DataService describing where and how the dataset can be obtained, and each Dataset carries one or more ODRL offers stating the policies under which it may be used. A catalog holds these descriptions and offers only — never the data itself.

Data discovery is a control-plane activity, separate from the Transfer Process. A catalog is served through the Dataspace Protocol Catalog Protocol: a Catalog Request returns a Catalog, and a Dataset Request returns a single Dataset. Catalog views are credential-scoped: a requester sees only the datasets and offers that its presented claims and the datasets' access policies permit, with identities and credentials conveyed through the Decentralized Claims Protocol. The provider of a dataset is the authoritative source of the DCAT metadata describing it.

Connectors are also often not reachable from the public Internet, as they typically accept connections only from allowlisted, authenticated peers. Consequently, the required network reachability between parties and endpoints depends on the chosen pattern and is described for each pattern below.

What the patterns below change is where the catalog index lives and who operates it, how metadata from many providers is brought together (if at all), the network reachability each pattern requires, and the resulting freshness, availability, trust, and search experience. As with the other capabilities, the options form a spectrum from a single central index to purely peer-to-peer discovery.

13.1 Central and federated catalogs

In the central variant a single Catalog Service aggregates metadata from all participants into one authoritative index that consumers query. In the federated variant several catalog services, typically per domain or sub-community, are aggregated or cross-queried so consumers obtain a combined view without a single global operator. Either way, consumers get one place (or a few) to search and rely at query time on an index that someone operates. The index may be operated by the Data Space Governance Authority (DSGA) itself, by an operator acting under the DSGA's authority (such as an operations company), or by an intermediary offering the catalog as a value-added service. In the first two cases the index is a governance-provided service for which the DSGA remains accountable. Instead, in the third case the intermediary operates in its own right and, following the Rulebook, is not itself the DSGA. An operations company thus operates governance services under the DSGA's authority, whereas an intermediary provides a value-added service independently.

A central or federated catalog may also provide metadata that is not included in DCAT. For example, participants may be able to add organizational metadata or branding information that is available only through the portal. Such information is specific to the data space and portal implementation and is therefore considered an optional extension.

13.1.1 Collecting metadata centrally: push versus pull

A central or federated index has to be populated from the providers. There are two basic approaches, which can also be combined: push and pull.

13.1.1.1 Push (publication) Each participant actively publishes its dataset descriptions and offers to the catalog through a publication interface and sends updates whenever its offerings change; the catalog is a mostly passive recipient. This gives near-real-time freshness at the moment of change, lets each provider control exactly what it exposes and when, and does not require the provider's catalog endpoint to be continuously reachable. The costs are that providers must implement publication and reliably send updates and deregistrations — otherwise entries go stale — and that the catalog must authenticate publishers and validate submitted metadata, since it now persists a copy that can drift from the source.

13.1.1.2 Pull (harvesting) The catalog periodically queries each participant's catalog endpoint over the Catalog Protocol and harvests their DCAT catalogs into the index — the approach taken by a federated-catalog crawler. Providers then need only expose the standard catalog endpoint they already offer for direct discovery, with no separate publication logic, and the catalog fetches from the authoritative source and can normalize and validate centrally. The costs are that freshness is bounded by the crawl interval, each provider's endpoint must be reachable when crawled, and crawling scales with the number of participants.

13.1.1.3 Reachability The two approaches also differ in which side opens the connection, which matters because connectors may not be publicly reachable. With push, each participant connects outbound to one well-known catalog endpoint, usually requiring no new inbound exposure on the participant's side. With pull, the catalog connects inbound to each participant, so every participant must allowlist the operator's source addresses. This stays manageable precisely because the operator set is small and fixed: the governance framework can name the DSGA or the few permitted catalog operators, and each participant allowlists only those. Either way, the set of parties that must reach a given connector stays small and governable.

13.1.1.4 Authorization Since catalog views are credential-scoped, a crawler cannot simply fetch one global view and serve it to everyone. It must either crawl under a service identity and then re-apply each dataset's access policy against the requesting consumer's claims at query time, or crawl on behalf of individual consumers. For example, the catalog operator may hold a credential granting a harvesting permission that participants recognize through the applicable Dataspace Trust Framework (DTF), so that providers return their full catalogs to it. Note that this concentrates all restricted metadata at the operator, whereas crawling per consumer does not. A central index that ignores the access policies will either expose the existence of restricted datasets or hide datasets a consumer is entitled to see.

Push-based collection faces the same requirement from the other side: the catalog must store each dataset's access policy alongside its metadata so it can scope what each consumer sees. In practice the two approaches are often combined — for example, providers push change notifications while the catalog pulls a full synchronization on a schedule — to balance freshness against load.

13.1.2 Trade-offs for central and federated catalogs

A central or federated catalog gives the best discovery experience: a single, consistent place to search across many providers, with server-side query, ranking, and normalization.

The trade-offs are that the index is a runtime dependency and a potential single point of failure for discovery (though not for data transfer, which remains peer-to-peer), its contents can be stale relative to the source, and whoever operates it can see the aggregate of what participants offer. This is a confidentiality and neutrality consideration, since the full catalog can reveal competitive information or let the operator favour or hide offerings. As with any central component, the centralization should be justified and mitigated — for example through replication or high availability, consumer-side caching, credential-scoped visibility so that no single view exposes everything, and federation so that no single operator holds the whole index.

13.2 Decentral catalogs

Here there is no aggregated central index. Each participant exposes its own catalog through its connector's Catalog Service endpoint, and consumers discover offerings by querying providers directly over the Catalog Protocol. Metadata stays at its source, so it is always authoritative and current, and each provider enforces credential-scoped visibility at its own endpoint using the requester's claims. No operator sees the whole space, and there is no shared component whose failure stops discovery.

The open question in this pattern is how a consumer finds the providers worth querying. One option is consumer-side aggregation where a participant queries the endpoints relevant to it and builds its own local index. Another option is a decentralized or replicated directory of catalog endpoints which provides a lightweight shared list of participants and their connector endpoints. This approach is far smaller and less sensitive than a full metadata catalog, and can itself be anchored in the manner described for the registry in the DSGA pattern. A third option is federated-catalog components that any participant may run for its own use, harvesting from others as above. These blur the line with the central pattern, the difference being that any such index is optional, participant-operated, and not a shared dependency.

13.2.1 Trade-offs for decentral catalogs

Decentral discovery maximizes autonomy, keeps metadata authoritative and fresh, removes the single point of failure, and prevents any operator from seeing the whole catalog.

The costs are the absence of a single searchable view: consumers must know or discover where to look, queries fan out across many endpoints, and building a global or ranked search is harder and pushed to the consumer side. Discovering endpoints in the first place needs at least a minimal shared directory or out-of-band knowledge, and consistency across independently served catalogs cannot be assumed. This pattern fits data spaces that prioritize autonomy, confidentiality of who offers what, and resilience over the convenience of centralized search.

A further constraint is network reachability. Because connectors may not be publicly reachable, full peer-to-peer discovery requires all-to-all connectivity. In practice this means that every provider must accept inbound discovery queries from every other participant — not only from the few it ends up transacting with. This is an allowlisting burden that grows with the square of the number of participants, enlarges each connector's attack surface, and

must be reconfigured as the participant set changes. Mitigations exist — a shared relay or gateway that connectors reach outbound-only, or a governance-defined connectivity fabric (for example a shared mTLS trust domain or network overlay) where reachability follows from membership rather than per-peer allowlisting — but each of these reintroduces a shared component. Network reachability therefore pushes the decentral pattern back toward a federated or central arrangement — the mirror image of the caching argument in the DSGA pattern, where distributing governance data to participants pushes a central arrangement toward the decentral end of the spectrum.

14 Outlook

IDS Working Group Architecture has been looking at a wide range of topics in collaboration with other IDS working groups such as the IDS Working Group Rulebook. Some of the topics include, but are not limited to, the following:

- Main technical concepts, architectural principles and patterns for data spaces
- Observability in data spaces
- Semantic interoperability in data spaces
- Establishing trust in data spaces
- Participant onboarding
- Usage control: Technical and organizational enforcement
- Control planes vs data planes
- Interoperability of governance across data spaces
- AI and data spaces
- Patterns for value added services

Some of these items are already planned for the upcoming release of IDS-RAM, while some would be considered in future roadmaps.

15 IDSA Papers on Focus Topics – In depth guidance on key challenges

The IDSA papers created by the IDSA working groups Architecture and Rulebook, providing guidance on individual focus topics dive deeper into specific adoption within data spaces.

These concise thematic modules ensure that the documentation can evolve with a fast-changing environment without overloading core documents such as the IDSA Rulebook and the IDS Reference Architecture Model.

Current topics include:

- **Observability** – monitoring data sharing transactions while respecting sovereignty: IDSA Position Paper Observability in Data Spaces
- **Identity & Trust** – decentralized identity management and verifiable credentials: Draft working paper Identifiers in Data Spaces
- **Interoperability** – semantic, organizational, and technical interoperability models: IDSA Position Paper Semantic Interoperability in Data Spaces
- **Agentic AI and LLM** – integration of MCP and data space in the context of agentic web: IDSA Rulebook page AI Agents
- **IDSA Position Paper Data Spaces and AI Trustworthy Agentic Participation in Data Spaces** - This position paper establishes a shared vocabulary for readers coming from either side of this convergence, sets out today's AI challenges and how the building blocks of a data space address them, details the concrete value AI brings to operating a data space and grounds the discussion in pilots already underway.
- **Value-Adding Services** – an architectural pattern for offering matchmaking and billing as competing, non-mandatory participant services in a decentralized data space, preserving sovereignty and avoiding centralized control points: Draft Focus Paper Value-Adding Services in Decentralized Data Spaces

Each Focus Topic is anchored in the IDSA Rulebook and IDS Reference Architecture Model and consistent with the Data Spaces Manifesto, providing reusable patterns and best practices.

Their modular structure ensures efficient maintenance and allows new topics to be integrated as technology and regulation evolve.

To know more about other IDSA Papers, please visit IDSA home page

16 Value-Adding Services in Decentralized Data Spaces

While decentralized data spaces fundamentally rely on autonomous participants interacting through standardized protocols such as the Dataspace Protocol (DSP) and the Decentralized Claims Protocol (DCP), many real-world ecosystems require additional value-adding services to support discovery, coordination, trust and economic processes. Rather than introducing centralized platform components, these services must be implemented in a way that preserves the core principles of decentralization, sovereignty and participant autonomy, as set out in the Rulebook's chapter on Decentralization. In this section, we describe an architectural pattern for integrating value-adding services as specialized participants within the data space itself. We use a matchmaking service and a billing service as concrete examples and show how they can be realized using standard data space interactions.

Each service operates under the governance rules of the Data Space Governance Authority (DSGA) and the accepted Dataspace Trust Framework(s) (DTFs). Credential issuance and verification follow those DTFs and are exchanged using DCP, while contract negotiation and data exchange are governed by DSP. This keeps the services interoperable and avoids creating privileged, centralized control points.

16.1 Matchmaking Service

A matchmaking service can be offered by one or more independent service provider organizations; a data space may have several competing matchmaking providers, and each participant individually decides which of them, if any, to use for a given negotiation. Such a service typically runs as part of the provider's own backend rather than as a mandatory, central component of the data space, consistent with the Rulebook's Marketplaces guidance. Data-holding participants that wish to make their assets discoverable provide structured metadata about their data offerings to a matchmaking participant of their choice, for example via event-based metadata feeds or other mechanisms such as pull-based retrieval. This metadata includes references to the corresponding DSP contract offers through which consumers can later negotiate access to the actual data. An event-driven approach allows a matchmaking service to efficiently integrate updates, additions and removals into its continuously maintained search index.

Based on the aggregated metadata, a matchmaking service builds and operates a searchable index and exposes a contract-governed search API within the data space. Any participant seeking data can establish a DSP contract with a matchmaking service of its choice to use this search interface. In addition to synchronous search, asynchronous interaction patterns can be supported: consumer participants may register data needs at a matchmaking service and are notified when matching metadata becomes available.

Search results returned to consumers contain metadata and references to the actual DSP offers of the respective data providers. As with any data space interaction, the subsequent negotiation, contracting and exchange of the actual data remain peer-to-peer between the consumer and the data-providing participants and are fully governed by DSP; the matchmaking service is not a party to that negotiation. The provision of metadata to the matchmaking service and the use of the search API are likewise governed by DSP contracts for interoperable, standardized integration.

Metadata visibility can be scoped by policy. Providers may publish public metadata, or restrict metadata feeds and search access using DSP contracts and DCP claims (e.g., membership or role-based credentials). Update and removal events are authorized by the orig-

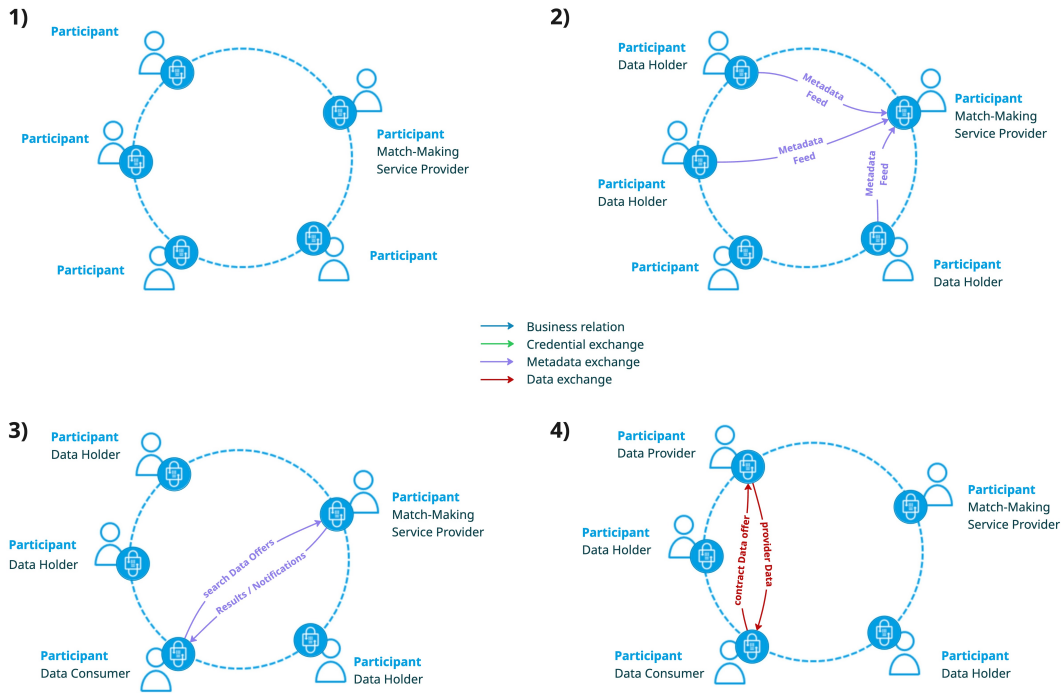


Figure 6: Figure 1-Matchmaking Service

inal publisher, preserving data sovereignty while still enabling efficient discovery.

16.2 Billing Service

A billing service can likewise be offered by one or more independent billing service provider organizations, and each participant individually decides which billing provider, if any, to use for a given exchange. Such a service typically runs as part of the provider's own backend rather than as a mandatory, central component of the data space. Both data providers and data consumers that wish to use a billing provider's infrastructure are onboarded as its customers. Onboarding establishes the roles and credentials: providers are registered and authorized to report usage for billable exchanges, while consumers receive a trust credential that verifies their eligibility to be billed through that specific billing provider.

After onboarding, a data provider may offer data under the explicit condition that consumption of the corresponding DSP contract offer is billed via a billing provider it accepts. To enforce this, the provider requires that only consumers presenting a valid billing trust credential issued by that billing provider are allowed to contract the offer. These trust credentials are issued and presented using DCP.

As with any data space interaction, the contract negotiation and the data exchange itself remain peer-to-peer between provider and consumer and are governed by DSP; presenting the billing trust credential is simply an additional proof required within that one negotiation, not a delegation of the negotiation to the billing provider. In parallel, the provider transmits usage data for all transfers to the billing service provider. The billing provider performs invoicing and payment processing directly with the consumer organization and sends operational notifications, such as payment and settlement status, back to the data provider. Trust credential exchange is governed by DCP, while usage data, notifications and contractual data exchange are consistently governed by DSP, preserving the decen-

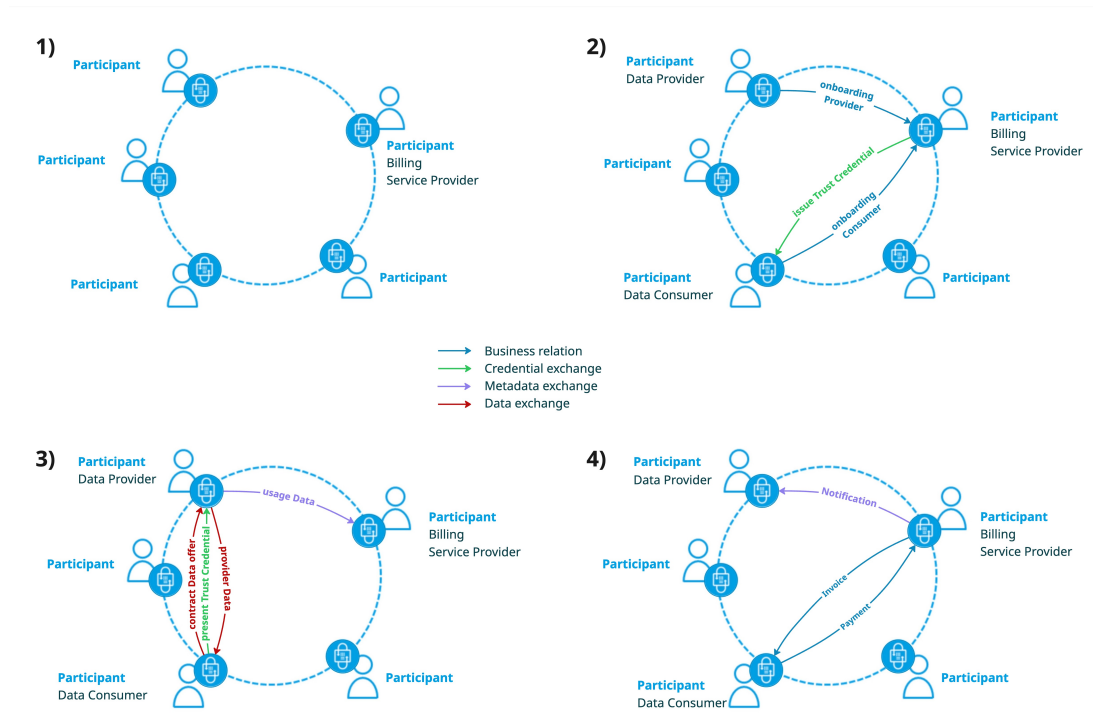


Figure 7: Figure 2-Billing Service

tralized control model.