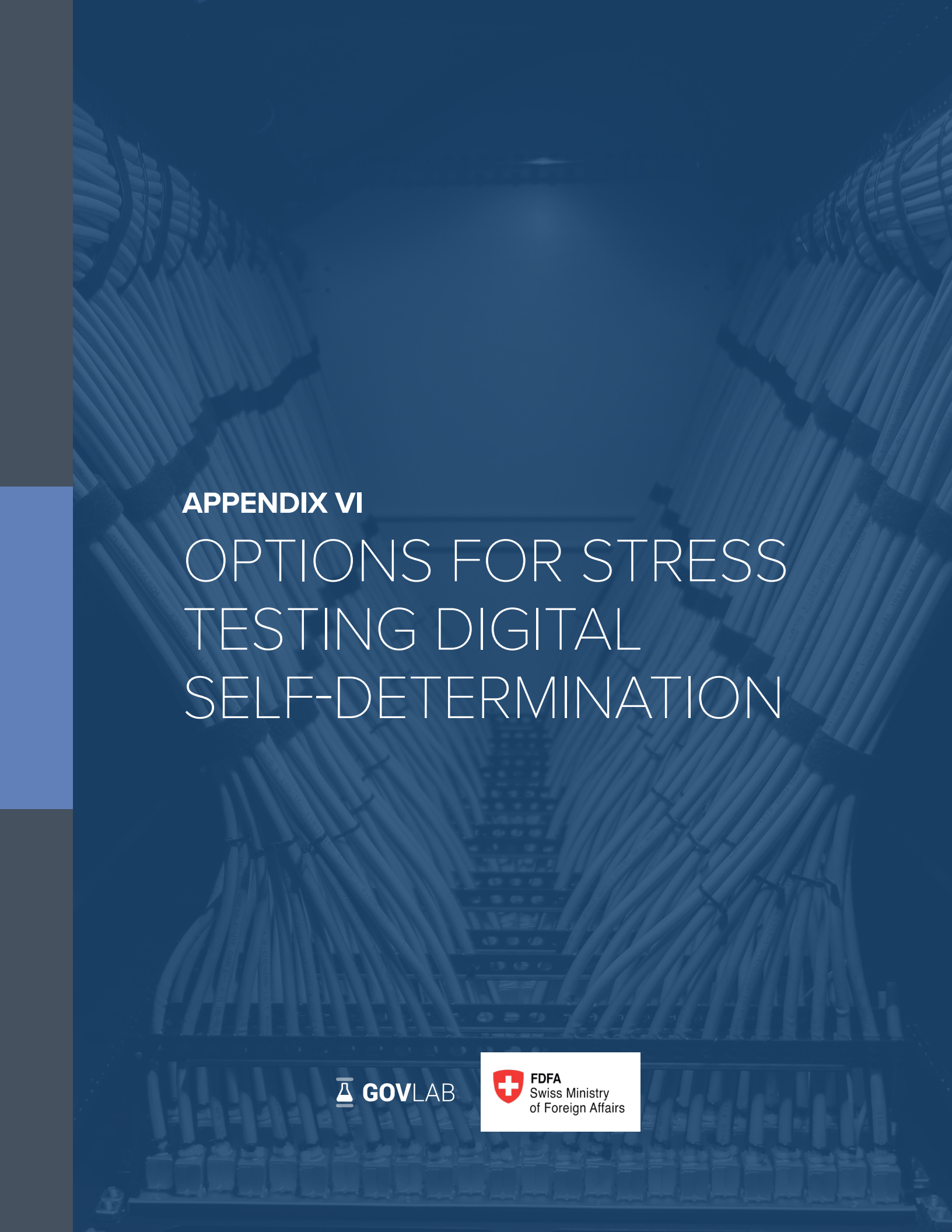

APPENDIX VI

OPTIONS FOR STRESS TESTING DIGITAL SELF-DETERMINATION

GOVLAB

FDFA
Swiss Ministry
of Foreign Affairs

APPENDIX VI

Options for Stress Testing Digital Self-Determination

This appendix provides a set of stress testing and evaluation methods that data space operators can use to assess whether Digital Self-Determination is being upheld in practice over time. The methods are intended to support the continuous assurance described in Phase 4 of the roadmap.

Each method examines how governance arrangements, contractual commitments, and technical controls perform under conditions that may strain agency, accountability, or legitimacy. These conditions include operational surges, contested decisions, system changes, security incidents, or shifts in data use and reuse.

The methods are organized around four principles used in Swiss guidance on trustworthy data spaces: transparency, control, effectiveness, and fairness. For each method, the appendix describes the objective, scope, preconditions, procedure, indicators, common failure modes, and reference sources. This structure is intended to help operators understand what a method tests, what is required to apply it, and what kinds of insights it can produce.

The methods presented here are illustrative rather than exhaustive. They draw on practices from data governance, cybersecurity, public administration, and AI oversight, adapted to the federated and multi-actor nature of data spaces. Data space operators are expected to select, adapt, and combine methods based on context, maturity, and risk profile, and to integrate findings into existing governance, review, and improvement processes.

1. Transparency

PURPOSE

Transparency concerns the ability of a data space to make decisions, data flows, and policy applications visible and understandable—particularly when conditions are changing or stressed. It enables auditability, meaningful explanation, and timely disclosure to participants and affected communities. In federated settings, transparency is essential to preserving legitimacy across organizational boundaries where no single actor has full visibility by default.

WHAT "GOOD" LOOKS LIKE

- End-to-end decision provenance: traceable inputs, applicable rules and versions, accountable actors, timestamps, and outcomes, presented in both machine-verifiable form and plain language.
- Complete and timely operational records: logs, lineage, model and policy change histories, and incident timelines that can be correlated across participants.
- Proactive disclosures: clear notifications and status updates when material changes or incidents occur, including the scope of impact and next steps.

TYPICAL ROLES AND RESPONSIBILITIES

- Data stewards: maintain dataset documentation, lineage, quality declarations, and contact points.
- Governance owners: manage rule sets and their versions; coordinate decision forums.
- Security/engineering: implement logging, signatures, time synchronization, and integrity controls.
- Communications and service desks: publish notices, status updates, and responses.
- Auditors (internal or external): verify evidence, completeness, and tamper-evidence.
- Affected parties and community representatives: receive explanations and, where appropriate, participate in review or appeal processes.

Method T1

Decision Provenance Drill

Objective

Assess the ability to reconstruct and communicate a complete, tamper-evident narrative for a significant governance or access decision, including who decided, which evidence and rules were applied, and the rationale for the outcome.

Scope

One recent, high-impact decision that crosses at least two organizations (for example, an approval or denial of a data access request, a policy exception, or a model deployment affecting downstream services).

Preconditions

- Policy or rule repositories with version control and immutable identifiers.
- Append-only, time-synchronized logs and signatures to ensure tamper-evidence.
- Named roles and an established channel for affected-party communications.

Procedure

- Select a decision taken within the last quarter and freeze associated artifacts (request, approvals, rule versions, timestamps, logs).
- Assemble a provenance pack that correlates inputs, rules, actors, timestamps, and the final outcome across all participating nodes.
- Produce a concise explanation suitable for non-specialist readers, summarizing what was decided, why, by whom, and which rule versions applied, with references to evidence.
- Verify integrity by checking signatures, hashes, and time coherence; record any gaps or inconsistencies.
- Conduct an appeal simulation in which stewards respond using the provenance pack and the plain-language explanation.

Indicators and suggested thresholds

- Time to assemble provenance pack: ≤ 4 hours for cross-organizational decisions.
- Completeness: ≥ 95% of required fields present (inputs, ruleset identifier, actors, timestamps, outcome, rationale, evidence links).
- Explainability: average usability rating ≥ 4 out of 5 from a small panel of non-experts.
- Tamper-evidence: 100% verification of signatures and hashes across artifacts.
- Appeal response time: ≤ 5 business days to provide a reasoned response referencing evidence.

Evidence to retain

Provenance bundle, plain-language explanation, integrity verification results, appeal record, and a checklist with indicator scores.

Common failure modes and remedies

- Missing rule versions: mandate inclusion of policy commit identifiers in decision records.
- Clock skew and inconsistent timestamps: enforce synchronized time sources and record acceptable skew.
- Explanations that are overly technical: require standard templates that define terms and avoid acronyms without definitions.

Scoring rubric (0–5)

0–1 ad hoc evidence; 2 partial logs; 3 consistent internal packs; 4 cross-organizational packs plus user-facing explanations; 5 automated, signed provenance with self-service access for authorized parties.

Real-world Examples

- W3C PROV (provenance data model and ontology for who/what/when/why): https://www.w3.org/TR/prov-overview/ and https://www.w3.org/TR/prov-o/
- W3C Verifiable Credentials (signed attestations to anchor actors/claims in decision trails): https://www.w3.org/TR/vc-data-model-2.0/
- ODRL (W3C Open Digital Rights Language) to represent policies/permissions referenced in decisions: https://www.w3.org/TR/odrl-model/
- ISO/IEC TS 27560 (machine-readable consent record lifecycle, linkable to access decisions): https://www.iso.org/standard/80392.html
- Certificate Transparency-style append-only logs (Merkle trees) as a pattern for tamper-evident decision logs: https://www.rfc-editor.org/rfc/rfc6962
- OpenLineage (standardized lineage events to connect inputs → processes → outputs across systems): https://openlineage.io/

Method T2

TRAC-Inspired Repository Audit (Transparency Controls)

Objective

Adapt the principles of Trustworthy Repositories Audit & Certification to stress-test documentation quality, integrity controls, and disclosure practices for datasets under change or incident.

Scope

One data product (dataset, metadata, and associated policies) with at least two maintainers across organizations, including one material change (for example, a schema modification) and one incident in the past 90 days.

Preconditions

- Dataset cards and policy summaries with clear ownership and contact points.
- Versioning with fixity (checksums or signatures) and chain-of-custody records.
- Defined change-notification and incident-disclosure procedures, including subscriber channels and status pages.

Procedure

- Conduct a documentation sweep to verify completeness and freshness of dataset cards, lineage, policy summaries, and contacts.
- Verify integrity through fixity checks for the last two stored versions and confirm chain-of-custody details.
- Review how the last material change was communicated: audience, timing, clarity, and catalog updates.
- Trace the most recent incident from detection to containment, notification, and lessons learned; confirm evidence of each step.
- Record non-conformities and corrective actions with owners and deadlines.

Indicators and suggested thresholds

- Documentation completeness: Documentation is largely current and filled out. Target: at least 90% of required fields updated in the latest release cycle.
- Fixity coverage: Stored artifacts include tamper checks and validate in practice. Target: 100% covered; audit validations mostly successful (≈98% or higher).
- Change-notice timeliness: ≤ 5 business days from approval to subscriber notification and catalog change log.
- Incident post-mortem: Incidents are analyzed and recorded with actions to prevent recurrence. Target: internal write-up within 15 business days; brief external note when the incident is material.

- Contactability: Published steward or contact point responds through the official channel. Target: first meaningful response within one business day.

Evidence to retain

Completed control checklist with links, fixity report, version diffs, change notices, incident post-mortem, and the corrective-action register.

Common failure modes and remedies

- Silent changes breaking downstream services: require catalog notices and subscriber webhooks for any material change.
- Stale documentation: link documentation checks to release gates; block promotions on missing or outdated records.
- Incomplete fixity: compute checksums on ingest and schedule periodic verification jobs.

Scoring rubric (0–5)

0–1 minimal documentation; 2 basic versioning; 3 fixity with change notices; 4 incident transparency with service-level targets; 5 continuous compliance dashboards and periodic transparency reports where appropriate.

Real-world Examples

- TRAC — Trustworthy Repositories Audit & Certification criteria/checklist: https://www.crl.edu/sites/default/files/2025-01/trac.pdf
- CoreTrustSeal — Trustworthy Data Repositories Requirements and certification process: https://www.coretrustseal.org/why-certification/requirements/
- OAIS Reference Model (ISO 14721) — conceptual framework for archival transparency & preservation: https://public.ccsds.org/pubs/650x0m2.pdf
- CCSDS 652.0-M-2 — Audit & Certification of Trustworthy Digital Repositories (current practice): https://ccsds.org/Pubs/652x0m2.pdf
- PREMIS Data Dictionary — preservation metadata for provenance, fixity, and events: https://www.loc.gov/standards/premis/
- BagIt (RFC 8493) — packaging with checksums for verifiable versioning/fixity during audits: https://www.rfc-editor.org/rfc/rfc8493

Method T3

Data Quality Drift Drill — Transparency Variant

Objective

Evaluate whether quality degradation becomes visible quickly to both producers and consumers, whether its impact is understandable, and whether remediation steps are documented as part of the dataset's public or participant-visible record.

Scope

One multi-source dataset in active use by downstream services. Over 48–72 hours, inject three drift types: missing values, delayed freshness, and semantic shift (for example, changed value ranges).

Preconditions

- Declared quality targets for freshness, completeness, and validity.
- Monitoring and alerting aligned to those targets.
- Subscriber communication channels (for example, webhooks, email, or portal notifications) and a status page.
- A mapping from dataset fields to dependent dashboards, APIs, or reports to support impact tagging.

Procedure

- Define a drift-injection plan specifying the type, timing, and magnitude of each drift.
- Introduce the drifts and record ground truth.
- Observe detection latency, alert routing, and receipt by subscribers.
- Publish a status update and annotate affected fields and dependent assets in the catalog.
- Apply remediation, update lineage to reflect both drift and fix, and publish a final note summarizing cause, impact, and prevention steps.
- Collect feedback from a small sample of consumers on clarity and timeliness of notices.

Indicators and suggested thresholds

- Detection latency: freshness drift detected within 30 minutes; completeness or semantic drift within 2 hours.
- Notification coverage: at least 95% of subscribers notified within 1 hour of detection.
- Impact annotation: at least 90% of affected fields and dependent assets tagged in the catalog within 1 business day.
- Status cadence: initial notice within 1 hour, with daily updates until resolution.
- Provenance update: lineage reflects drift and remediation within 1 business day.

Evidence to retain

Alert timelines, subscriber delivery logs, status page records, catalog annotations, lineage snapshots before and after remediation, and consumer feedback.

Common failure modes and remedies

- Long-tail consumers miss alerts: mandate subscription on first access and maintain a broadcast fallback channel.
- Unclear scope of impact: maintain an up-to-date field-to-dependency map; automate tagging of affected dashboards and APIs.
- Slow posting of updates: use a prepared status template and pre-approved wording blocks for rapid publication.

Scoring rubric (0–5)

0–1 ad hoc discovery; 2 basic alerts; 3 consumer notifications; 4 end-to-end provenance with status pages; 5 automated dependency tagging, consistent public or participant disclosures, and structured post-mortems.

Real-world Examples

- Principles from the Data Spaces Support Centre and EU's EHDS governance framework.
- ISO/IEC 25012 (Data Quality Model) — dimensions/attributes to define visible quality targets: https://standards.iso.org/ittf/PubliclyAvailableStandards/c061298_ISO_IEC_25012_2008(E).zip
- ISO 8000 (Data quality series) — governance for master and reference data quality: https://www.iso.org/committee/54756.html
- Great Expectations (declarative data validation with human-readable "expectations" and docs): https://greatexpectations.io/
- See also Gille et al. (2025) on trustworthiness as dependent on the reliable, transparent implementation of data handling principles, and the limitations of trust indicators that fail to capture operational degradation.
- Amazon Deequ (constraint-based data quality checks at scale): https://github.com/awslabs/deequ
- Data Contracts (schema + SLOs + change control to make drift and impacts visible): https://datacontracts.org/
- OpenLineage + dbt tests (lineage-aware impact analysis and test failures surfaced to consumers): https://openlineage.io/ and https://docs.getdbt.com/docs/build/tests

2. Control

PURPOSE

Control concerns the practical ability of individuals and communities to set conditions on data use, grant or withdraw permission, obtain copies or move data, exit a data space, contest outcomes, and shape collective boundaries for acceptable use. In federated environments, meaningful control requires that rights and preferences propagate reliably across organizational lines and remain enforceable under stress.

WHAT "GOOD" LOOKS LIKE

- Consent and permission changes propagate quickly and halt further processing where required, with clear confirmation to affected parties.
- Portability and exit are feasible without loss of context, including metadata and provenance needed for continued use elsewhere.
- Collective voice is embedded through social license or equivalent community conditions, with routes to escalate ambiguous cases for deliberation.
- Contestation and redress processes exist, are understandable, and lead to timely, reasoned outcomes tied to evidence.

TYPICAL ROLES AND RESPONSIBILITIES

- Data stewards and access managers: implement permission changes, exports, and exits; verify completion.
- Governance owners and community representatives: define conditions for allowable use and adjudicate contested cases.
- Security/engineering: ensure permissions, tokens, and policies are enforced consistently across participants.
- Service desks and communications: provide confirmations, guidance, and appeal routes.
- Auditors: verify evidence of enforcement and timeliness.

Method C1

Consent Revocation Flood

Objective

Assess whether large volumes of permission withdrawals (revocations) are propagated and enforced across all relevant participants and downstream services under stress, such that further processing of affected data is halted in a timely and verifiable manner.

Scope

A coordinated batch of revocations affecting multiple datasets and purposes across at least two independent organizations or nodes, including at least one downstream consumer (for example, an analytics job, dashboard, export, cache, or model pipeline).

Preconditions

- A structured “revocation object” that clearly identifies scope (subjects or records, dataset identifiers, purpose or policy tags, effective time, originating authority).
- A federation-wide propagation mechanism (event bus, API, or trust service) discoverable by obligated processors.
- Enforcement points that are revocation-aware (access control, policy/usage control, schedulers, and data processing jobs).
- An inventory of downstream consumers (dashboards, scheduled jobs, APIs, models, caches, mirrors/exports).
- Agreed contacts and escalation paths for the test window.

Procedure

- Launch the wave: issue a high volume of revocations within a short interval, covering varied data products and purposes.
- Observe propagation: confirm delivery to all obligated processors and access control points; capture delivery receipts or logs.
- Check enforcement: attempt normal reads/processing at enforcement points; expected outcome is denial or safe fail.
- Verify downstream suppression: inspect dashboards, jobs, caches, models, and exports for continued use after the effective time; expected outcome is no access or use.
- Record exceptions and apply fixes: log any residual access or processing; assign remediation with owners and deadlines.

Indicators and suggested thresholds (streamlined)

- Propagation time: revocations reach obligated processors within hours, not days.
- Coverage: nearly all targeted processors act on revocations; isolated misses are corrected during the exercise.
- Confirmation quality: plain-language confirmations issued that processing/access has been halted.
- Residual access: none detected in sampled downstream services after the effective time.

Evidence to retain

- The revocation batch description (scope, counts, effective times) and originating authority.
- Delivery receipts or logs from processors and enforcement points.
- Enforcement confirmations (denials, halted jobs, updated access rules).
- Downstream verification notes/screenshots or job logs showing suppression.
- Exception list with remediation actions and completion status.

Common failure modes and remedies

- Unregistered consumers (missed subscribers) → maintain a living dependency map; mandate subscription on first access; add a broadcast fallback channel.
- Stale caches/mirrors or exported copies → add cache-invalidation hooks; require periodic permission re-checks; tag exports with revocation callbacks.
- Batch jobs that bypass checks → enforce policy checks at use time (not only at ingest); instrument schedulers to query current permission state.
- Ambiguous scope in revocation objects → standardize identifiers and purpose tags; require effective-time semantics; digitally sign revocation objects.

Scoring rubric (0–5)

0–1 ad hoc handling; 2 partial propagation with manual steps; 3 consistent cross-org propagation; 4 end-to-end enforcement including downstream suppression; 5 automated propagation with self-service confirmations, dependency-aware suppression, and continuous monitoring.

Real-world Examples

- Privacy and data protection: withdrawal of consent under GDPR Articles 7 and 21, with downstream suppression akin to “right to be forgotten” practices.
- Web consent signals: Global Privacy Control (GPC) / “do not sell/share” signals used to propagate opt-outs across services.

- Email ecosystems: “List-Unsubscribe” mechanisms and suppression lists to prevent further messaging after opt-out.
- Payments: card chargeback/stop-payment instructions propagate across networks to halt further settlement.
- Software/package registries: security advisories and revocation of compromised packages trigger downstream build failures and cache purges.

Real-world Examples

- ISO/IEC 27560 (machine-readable consent record lifecycle for revoke/withdraw workflows): https://www.iso.org/standard/80392.html
- Global Privacy Control (GPC) signal for propagating revoke/opt-out across services: https://globalprivacycontrol.github.io/gpc-spec/
- OpenID Shared Signals & Events (SSE) / CAEP for near-real-time risk and consent state changes: https://openid.net/specs/openid-sse-framework-1_0.html and https://openid.net/specs/openid-caep-specification-1_0.html
- SCIM 2.0 Deprovisioning (revocation of identities/entitlements across systems): https://www.rfc-editor.org/rfc/rfc7644
- UMA 2.0 (User-Managed Access) for user-centric authorization and revocation: https://docs.kantarainitiative.org/uma/rec-uma-core.html
- XACML / Policy-as-Code (OPA/Rego) to enforce revoke conditions at decision points: https://docs.oasis-open.org/xacml/xacml-3.0-core-spec-os-en.html and https://www.openpolicyagent.org/

Method C2

Portability & Exit Drill

Objective

Evaluate whether individuals or organizations can obtain a complete, usable export of their data and associated context, and terminate participation with verifiable deletion or access removal across the federation.

Scope

One participant (or cohort) requesting both data portability and full exit. At least one complex dataset with dependencies (linked tables, files, or models) spanning two or more organizations.

Preconditions

- Defined export format(s) for data, metadata, and provenance (for example, CSV/ Parquet plus machine-readable metadata, policy references, and lineage).
- Exit procedure with roles, steps, and required confirmations.
- Mechanisms for deletion or access termination, including logs and attestations from processors.
- Neutral test environment for import validation.

Procedure

- Initiate a portability request and a termination-of-participation (exit) request.
- Produce export packages that include data, essential metadata, and provenance sufficient to re-use elsewhere.
- Execute deletion or access termination where applicable; capture confirmations from all obligated processors.
- Import the export package into a neutral environment to validate completeness and usability.
- Document any gaps and assign remediation actions.

Indicators and suggested thresholds (streamlined)

- Time to complete export and exit: measured in days rather than weeks.
- Export completeness: data plus key metadata and provenance sufficient for re-use.
- Deletion/access termination: confirmed by all obligated processors.
- Import usability: successful reconstruction of core context in the test environment.

Evidence to retain

- Export packages and manifests, checksums, and format descriptions.
- Deletion/termination confirmations and relevant logs.
- Import validation notes and outcomes.
- Exception register with remediation owners and deadlines.

Common failure modes and remedies

- Missing metadata/provenance → extend export scope to include schemas, code lists, policies, and lineage pointers.
- Orphaned access after exit → enforce permission revocation and token invalidation as part of exit workflow.
- Proprietary formats only → add open, documented formats or include conversion tools and schemas.

Scoring rubric (0–5)

0–1 ad hoc exports; 2 partial export without metadata; 3 complete export or exit but not both; 4 complete export plus verified termination; 5 repeatable, automated export/exit with self-service status and cross-org confirmations.

Real-world Examples

- Data Transfer Project (interoperable, service-to-service export/import protocol): https://datatransferproject.dev/
- Solid Protocol (user-controlled data pods with export/import interfaces): https://solidproject.org/
- GA4GH Data Repository Service (DRS) for portable access to research data objects: https://ga4gh.github.io/data-repository-service-schemas/
- DCAT-AP (interoperable metadata for portable datasets across catalogs): https://joinup.ec.europa.eu/collection/semic-support-centre/dcat-ap
- BagIt (RDF 8493) packaging for verifiable, checksummed export bundles: https://www.rfc-editor.org/rfc/rfc8493
- ISO/IEC 19944-1 (cloud data flow and roles for portability and exit): https://www.iso.org/standard/74437.html

Method C3

DSAR/SAR Surge Drill (Rights Exercise Under Load)

Objective

Validate that mass requests to access, rectify, or erase personal data are acknowledged and fulfilled reliably across the federation, with consistent quality, clear communication, and effective escalation when backlogs occur.

Scope

A surge scenario reflecting heightened public attention or coordinated campaigns, involving multiple request types and at least two independent organizations or nodes.

Preconditions

- Defined processes and service targets for acknowledgment, clarification, and fulfillment.
- Cross-organization routing for requests touching multiple processors.
- Templates for responses and a mechanism for secure delivery of results.
- The data space operator provides a shared interface, routing, or coordination layer for rights requests, or participation agreements assign the operator a role in rights-handling.

Procedure

- Submit a large, time-bounded set of requests covering access, rectification, and erasure.
- Track acknowledgments, clarifications, and fulfillment events across participants.
- Monitor queues, backlogs, and use of escalation or fallback procedures.
- Sample responses for completeness, clarity, and appropriateness of redactions.
- Summarize throughput, bottlenecks, and prioritized improvements.

Indicators and suggested thresholds (streamlined)

- Acknowledgment and completion times: within published service targets.
- Handling consistency across participants: minimal variance for similar cases.
- Response quality: clear, complete responses meeting the request type.
- Escalation effectiveness: predictable pathways with visible ownership.

Evidence to retain

- Request logs with timestamps and routing history.
- Copies of acknowledgments, clarifications, and final responses.
- Queue and backlog metrics, plus escalation records.
- Quality review notes and action items.

Common failure modes and remedies

- Fragmented fulfillment across participants ➜ implement centralized coordination or shared case IDs.
- Incomplete responses or unclear redactions ➜ standardize response templates and redaction criteria.
- Queue saturation ➜ introduce pre-triage, automation for common cases, and surge protocols.

Scoring rubric (0–5)

0–1 ad hoc handling of rights requests; 2 basic acknowledgments with delays; 3 consistent processing meeting targets in normal conditions; 4 reliable performance under surge with clear communications; 5 surge-ready operations with automation and coordinated cross-org workflows.

Real-world Examples

- EDPB Right-of-Access operational guidance (case handling methodology): https://www.edpb.europa.eu/system/files/2023-04/edpb_guidelines_202201_data_subject_rights_access_v2_en.pdf
- ISO/IEC 27701 (PIMS) mapping of roles and processes for rights handling: https://www.iso.org/standard/71670.html
- NIST Privacy Framework (Identify–Govern–Control–Communicate methodology): https://www.nist.gov/privacy-framework
- Consumer Data Standards (Australia CDR) request/consent API patterns for scale: https://consumerdatastandards.gov.au/
- SRE-style runbooks and queue triage (incident/surge methodology applied to rights ops): https://sre.google/sre-book/incident-response/
- Playbooks for secure file delivery and redaction QA (records release methodology): https://support.atlassian.com/statuspage/docs/incident-communication-tips/ (communication pattern)

3. Effectiveness

PURPOSE

Effectiveness concerns whether a data space can carry out its governance framework reliably in practice. It focuses on the ability of rules, roles, and controls to operate as intended across organizations, respond to incidents and change, and support ongoing data use without breakdowns or excessive friction. An effective data space turns governance decisions into timely, coordinated action and adapts its arrangements as conditions evolve.

WHAT "GOOD" LOOKS LIKE

- **Governance execution under pressure:** decisions, approvals, escalations, and controls function predictably during incidents, surges, or novel use cases.
- **Policy-to-practice alignment:** governance rules are implemented consistently across participants and enforced by operational and technical controls.
- **Cross-organizational coordination:** handoffs between actors occur without ambiguity, duplication, or delay.
- **Corrective capacity:** incidents, complaints, or failures lead to remediation that updates governance, contracts, or systems.
- **Operational continuity:** growth in participation or data use does not undermine response times, control effectiveness, or compliance.

TYPICAL ROLES AND RESPONSIBILITIES

- **Governing body or steering committee:** approves governance rules, mandates corrective actions, and oversees system-wide performance.
- **Data stewards and governance owners:** execute governance decisions, coordinate across participants, and maintain evidence of implementation.
- **Operations and engineering teams:** operate enforcement mechanisms, monitoring, and incident response processes.
- **Compliance and legal functions:** manage regulatory obligations and external reporting under time constraints.
- **Auditors or assurance functions:** verify that governance arrangements function as intended and that corrective actions are completed.

Method E1

Accountability Chain & Role Assignment Stress Test

Objective

Verify that responsibilities are unambiguous and actionable under stress by tracing a real decision from request to outcome, showing who was accountable at each step and how authority was delegated or escalated.

Scope

One cross-organization decision (for example, high-impact access approval/denial or policy exception) with at least two hand-offs.

Preconditions

- Published role catalog (e.g., RACI/DAI) and delegation-of-authority register.
- Decision registers with approver identities and timestamps.
- Policy repository with versioning.

Procedure

- Select a recent decision; map every step to a role and named person/team.
- Validate that each step had legitimate authority per the register.
- Confirm escalation path and timing where ambiguity or conflict occurred.
- Produce an accountability chain (who decided, on what basis, and why).
- Record gaps (missing owner, unclear mandate, conflicting approvals) and assign corrective actions.

Indicators and suggested thresholds (streamlined)

- Role coverage: all steps mapped to accountable roles/persons.
- Authority match: approvals align with delegation register.
- Escalation timeliness: escalations resolved within agreed windows.
- Documentation quality: decision pack complete and verifiable.

Evidence to retain

Role catalog, delegation register excerpt, decision pack, accountability chain diagram, corrective-action log.

Common failure modes and remedies

- Dual ownership or gaps ➔ revise RACI; add explicit tie-break and backup roles.
- Shadow approvals ➔ require system-enforced approver lists and signature checks.
- Version drift between policy and practice ➔ bind approvals to policy commit IDs.

Scoring rubric (0–5)

0–1 unclear roles; 2 partial mapping; 3 consistent mapping; 4 consistent mapping with timely escalations; 5 automated checks and self-serve accountability views.

Real-world Examples

- UK NHS “Caldicott Guardian” role (clear data responsibility and decision authority in health settings): https://www.gov.uk/government/publications/the-caldicott-guardian-manual
- ICO (UK) Accountability Framework (assigning roles, documenting decisions, demonstrating compliance): https://ico.org.uk/for-organisations/accountability-framework/
- RACI/Responsibility mapping used in government service delivery (roles and escalation clarity): https://www.gov.uk/service-manual/agile-delivery/writing-user-stories#roles-and-responsibilities and general method overview: https://www.atlassian.com/team-playbook/plays/roles-and-responsibilities
- EU research infrastructures with documented governance roles (ELIXIR example — data access/governance committees): https://elixir-europe.org/about-us/how-governed
- ISO 37301 Compliance Management Systems (role clarity and governance responsibilities in compliance programs): https://www.iso.org/standard/75080.html

Method E2

Policy-to-Enforcement Consistency Audit ("Policy Parity")

Objective

Test whether written policies are consistently enforced by technical controls and procedures across participants (no divergence between "paper" and "practice").

Scope

Two high-value policies (for example, purpose limitation and access approval thresholds) across at least two organizations and two control planes (human process + technical gate).

Preconditions

- Machine-readable policy artifacts or testable rules (e.g., OPA/Rego, XACML).
- Access logs and decision records; change control linking policy to deployment.

Procedure

- Extract the current policy text and the enforced rule set.
- Run parity tests (unit tests or sampled cases) comparing expected vs. actual decisions.
- Inspect change tickets to ensure rule deployments match approved policy versions.
- Record divergences and implement hot-fixes; document root causes.

Indicators and suggested thresholds (streamlined)

- Parity rate: high alignment of test cases across organizations.
- Traceability: each enforced rule links to a policy version/commit.
- Time to correct divergences: within a defined window (e.g., days).
- Negative test handling: prohibited cases correctly blocked.

Evidence to retain

Policy texts, rule artifacts, test cases and results, change tickets, parity report.

Common failure modes and remedies

- Policy text too vague → introduce normative, testable clauses.
- Drift after emergency changes → require post-incident parity checks.
- Local overrides → enforce central policy import with signed bundles.

Scoring rubric (0–5)

0–1 untested assumptions; 2 occasional manual checks; 3 routine parity tests; 4 cross-org automated tests; 5 continuous parity with signed policy supply chain.

Real-world Examples

- Policy-as-Code with Open Policy Agent (OPA) / Gatekeeper to keep runtime controls aligned with written policies: https://www.openpolicyagent.org/docs/latest/ and https://openpolicyagent.org/docs/latest/gatekeeper/
- XACML in regulated environments for fine-grained, testable authorization aligned to policy text: https://docs.oasis-open.org/xacml/xacml-3.0-core-spec-os-en.html
- Kubernetes admission control with OPA Gatekeeper (preventive policy enforcement matching governance rules): https://open-policy-agent.github.io/gatekeeper/website/
- GitOps change control linking approved policy versions to deployed rules (policy version ➜ runtime parity): https://opengitops.dev/
- NIST Internal Control assessments (mapping policy requirements to implemented controls and evidence): https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final

Method E3

Incident Reporting & Regulatory Notification Drill

Objective

Exercise legal and supervisory reporting obligations under time pressure (e.g., cybersecurity, privacy), ensuring signals reach competent authorities and affected parties with required content and timing.

Scope

At least one incident type requiring external notification (e.g., significant service disruption, security incident involving personal data) with cross-organization coordination.

Preconditions

- Mapped obligations (who reports what, to whom, by when) and templates.
- Evidence capture and approval workflows.
- Contact lists for CSIRTs/authorities and affected recipients.

Procedure

- Trigger a timed simulation with evolving facts.
- Produce early warning/initial notification, then intermediate and final reports as required.
- Coordinate cross-org facts, approvals, and evidence; deliver to authorities and affected parties.
- Record timing, completeness, and any retractions/updates.

Indicators and suggested thresholds (streamlined)

- Timeliness: notifications within prescribed windows.
- Completeness: required fields present; updates issued as facts evolve
- Coordination: single source of truth established quickly.
- Proof of delivery: confirmations from authorities/recipients retained.

Evidence to retain

All report versions, timelines, approvals, delivery receipts, and post-exercise review.

Common failure modes and remedies

- Unclear lead reporter → pre-assign lead roles and alternates.
- Conflicting versions of facts → adopt shared incident fact sheet with version control.
- Missed audience (e.g., customers) → maintain audience matrices and checklists.

Scoring rubric (0–5)

0–1 ad hoc reporting; 2 late or incomplete notices; 3 on-time basic reports; 4 on-time, multi-stage reporting; 5 rehearsed cross-org reporting with dashboards and evidence packs.

Real-world Examples

- NIS2 Directive multi-stage incident reporting (early warning, notification, intermediate, final): https://digital-strategy.ec.europa.eu/en/policies/nis2-directive
- ENISA guidance on national incident reporting and CSIRT coordination: https://www.enisa.europa.eu/topics/csirt-cert-services/national-incident-reporting
- GDPR data breach notification (supervisory authority and affected-party notices): https://ico.org.uk/for-organisations/report-a-breach/ and EDPB examples: https://edpb.europa.eu/our-work-tools/our-documents/guidelines/guidelines-012021-examples-personal-data-breach_en
- NCSC (UK) incident management guidance for coordinated reporting: https://www.ncsc.gov.uk/collection/incident-management
- Financial sector crisis/incident simulations (TIBER-EU threat-led exercises with reporting flows): https://www.ecb.europa.eu/paym/cyber-resilience/tiber-eu/html/index.en.html

Method E4

Grievance & Redress Mechanism Effectiveness Exercise

Objective

Assess whether complaint and redress channels are effective in practice against recognized criteria, and whether outcomes lead to corrective action and learning.

Scope

Two pathways tested end-to-end: an individual complaint (e.g., contested access decision) and a community complaint (e.g., alleged breach of a social license condition).

Preconditions

- Documented procedure; public contact points; case management tooling.
- Defined performance targets and publication norms (what is disclosed to whom and when).

Procedure

- Submit representative complaints; track acknowledgment, investigation, and outcome.
- Evaluate against effectiveness criteria (legitimate, accessible, predictable, equitable, transparent, rights-compatible, learning-oriented, dialogic).
- Verify linkage from decisions to actions (policy updates, sanctions, remediation).
- Publish appropriate summaries and close the loop with complainants.

Indicators and suggested thresholds (streamlined)

- Timely acknowledgment and predictable milestones.
- Procedural fairness (access to information; equitable treatment).
- Transparency proportional to sensitivity; learning captured and reused.
- Actionability: outcomes drive concrete remediation.

Evidence to retain

Case files, communications, decisions, implementation records, and learning notes.

Common failure modes and remedies

- Performative channels with low uptake → simplify access, protect from retaliation, publicize outcomes.
- No link to change → bind decisions to tickets and policy/version updates.
- Narrow scope → include community/collective grievances, not only individual ones.

Scoring rubric (0–5)

0–1 channel exists on paper only; 2 inconsistent handling; 3 consistent handling with basic transparency; 4 meets effectiveness criteria with learning loop; 5 independently reviewed with regular public summaries.

Real-world Examples

- UN Guiding Principles on Business and Human Rights (effectiveness criteria for non-judicial grievance mechanisms): https://www.ohchr.org/sites/default/files/documents/publications/guidingprinciplesbusinesshr_en.pdf
- World Bank Grievance Redress Service (project-level complaint intake, handling, and escalation): https://www.worldbank.org/en/projects-operations/products-and-services/brief/grievance-redress-service
- IFC Performance Standard 1 (stakeholder engagement and grievance mechanisms in projects): https://www.ifc.org/wps/wcm/connect/Topics_Ext_Content/IFC_External_Corporate_Site/Sustainability-At-IFC/Policies-Standards/Performance-Standards
- OECD Due Diligence Guidance (stakeholder engagement and remediation expectations): https://mneguidelines.oecd.org/due-diligence-guidance-for-responsible-business-conduct.htm
- National ombudsman models (example: European Ombudsman — handling complaints on maladministration): https://www.ombudsman.europa.eu/en/home

Method E5

Speak-Up & Whistleblowing System Live Test

Objective

Demonstrate that concerns about misconduct can be safely reported, assessed, addressed, and closed with protections for reporters and subjects, and with traceable outcomes.

Scope

A controlled test of the speak-up channel from intake to closure, including cross-organization routing and confidentiality safeguards.

Preconditions

- Whistleblowing policy, secure intake channels (anonymous where lawful), case handling roles, anti-retaliation measures.
- Training and communication materials.

Procedure

- Submit test reports (varying severity and scope) through available channels.
- Track triage, investigation, interim controls, and resolution.
- Verify reporter protection and confidentiality controls.
- Document corrective and preventive actions and communication to stakeholders.

Indicators and suggested thresholds (streamlined)

- Intake reliability and confidentiality preserved.
- Triage and investigation within targets; fair and impartial handling.
- Protection against retaliation evidenced; reporter informed of outcome as appropriate.
- CAPA quality: corrective and preventive actions implemented and verified.

Evidence to retain

Intake logs, case files, protection assessments, CAPA records, and close-out reports.

Common failure modes and remedies

- Chilling effects or fear of retaliation → reinforce protection measures and independent oversight.
- Channel fragmentation → unify intake and routing; publish clear instructions.
- Action drift post-investigation → require CAPA tracking to closure.

Scoring rubric (0–5)

0–1 informal, risky channel; 2 basic hotline; 3 documented systems; 4 protected, timely handling with CAPA; 5 ISO-aligned systems with regular effectiveness tests and reporting.

Real-world Examples

- ISO 37002 Whistleblowing Management Systems (receive → assess → address → conclude): https://www.bsigroup.com/en-GB/standards/iso-37002-whistleblowing-management-systems/
- EU Whistleblower Protection Directive (minimum standards for safe reporting and anti-retaliation): https://eur-lex.europa.eu/eli/dir/2019/1937/oj
- NHS “Freedom to Speak Up” Guardians (speak-up network improving safety and accountability): https://nationalguardian.org.uk/
- World Bank Integrity Vice Presidency — confidential reporting channels: https://www.worldbank.org/en/about/unit/integrity-vice-presidency/reporting
- Transparency International — practical guidance and tools for whistleblowing systems: https://www.transparency.org/en/projects/whistleblowing

4. Fairness

PURPOSE

Fairness concerns the equitable distribution of benefits and burdens within a data space, including who participates, whose data is represented, who gains access to value, and who bears risk. It requires detecting and correcting disparate impact across groups, ensuring inclusive decision-making, and honoring community-defined boundaries.

WHAT "GOOD" LOOKS LIKE

- Equitable access and participation: onboarding, service levels, and decision turnarounds do not systematically disadvantage particular groups or smaller organizations.
- Outcome parity: automated and procedural decisions avoid unjustified disparities across protected or vulnerable groups; discrepancies are explainable and correctable. Depends on lawful access to group attributes and context-specific baselines.
- Collective guardrails: community conditions and social license boundaries are enforced, particularly for groups historically subject to harm or extraction.
- Group-aware privacy: re-identification and disclosure risks are assessed and mitigated at group level, not only overall.

TYPICAL ROLES AND RESPONSIBILITIES

- Governance leads and community representatives: define fairness objectives, review outcomes, and set remediation.
- Data stewards and platform operations: monitor parity, service levels, and dataset representativeness.
- Ethics and legal advisors: align fairness controls with legal protections and sector norms.
- Security, privacy, and analytics teams: assess group risks, audit models and policies, and implement mitigations.
- Auditors: verify evidence of monitoring, decisions, and remediation.

Method F1

Social License & Community Conditions Check

Objective

Assess whether proposed data uses comply with community-defined conditions for acceptable use ("social license"), and whether governance processes can resolve ambiguous or contested cases in ways that are fair, reasoned, timely, and enforceable.

Scope

Two or more realistic proposals involving affected communities:

- one clearly compliant; and
- one borderline, novel, or contested.

At least one proposal should involve datasets or use cases with heightened risk of harm, sensitivity, or unequal impact on specific groups.

Preconditions

- Documented community conditions or social license criteria, such as allowed and prohibited purposes, sensitivity thresholds, red lines, benefit-sharing requirements, or required safeguards.
- A defined escalation forum with representation from affected stakeholders and, where appropriate, independent members.
- Mechanisms to translate governance decisions into enforceable policies, contractual terms, or access rules.

Procedure

- Screen proposals against published conditions; decide clear cases.
- Escalate ambiguous cases; record deliberation, rationale, and outcome with fairness considerations explicit.
- Translate decisions into policy or access changes and deploy.
- Publish a summary to relevant participants; monitor adherence where required.
- Review any objections and decide on remediation or follow-up.

Indicators and suggested thresholds (streamlined)

- Timely screening and escalation with documented rationale.
- Enforcement fidelity: policy and access controls match the decision.
- Visibility: concise decision summaries accessible to stakeholders.
- Community satisfaction signal: absence of unresolved fairness objections after publication.

Evidence to retain

Proposal dossiers, screening notes, escalation records, policy updates, notifications, and any monitoring reports.

Common failure modes and remedies

- Vague conditions → add concrete examples, thresholds, and safeguards.
- Decisions not enforced → bind governance outcomes to technical tickets and verify deployment.
- Limited representation → broaden forum membership; include independent oversight.

Scoring rubric (0–5)

0–1 ad hoc approvals; 2 informal screening; 3 documented decisions; 4 enforceable outcomes with summaries; 5 mature cycle with monitoring and responsive adjustments.

Real-world Examples

- Responsible Data Re-use in Developing Countries: Social Licence through Public Engagement: https://www.afd.fr/en/ressources/responsible-data-re-use-developing-countries-social-licence-through-public-engagement
- CARE Principles for Indigenous Data Governance (community benefit, authority to control): https://www.gida-global.org/care
- Free, Prior and Informed Consent (FPIC) toolkits for community authorization: https://www.fao.org/indigenous-peoples/our-pillars/fpic/en/
- Citizens' Juries / Mini-publics (structured deliberation for contested data uses): https://www.jefferson-center.org/what-is-a-citizens-jury/
- Deliberative Polling (evidence + facilitated dialogue before decisions): https://cdd.stanford.edu/what-is-deliberative-polling/
- Social Impact Assessment (IAIA methodology) to assess distribution of benefits/harms: https://www.iaia.org/uploads/pdf/SIA_Guidance_Document_IAIA.pdf
- OECD Due Diligence for Responsible Business Conduct (stakeholder engagement + conditional approvals): https://mneguidelines.oecd.org/due-diligence-guidance-for-responsible-business-conduct.htm

Method F2

Algorithmic Value Alignment & Fairness Test

Objective

Assess whether automated or rule-based decision systems produce unjustified disparities across groups and respect declared individual or community preferences and constraints, and whether explanations, appeals, and corrective mechanisms function effectively in practice.

Scope

A representative set of cases where governance rules or policies are implemented algorithmically, such as ranking, recommendation, eligibility, prioritization, or access control. The scope should include:

- cases with group attributes relevant to fairness assessment, where lawful and appropriate; and
- edge cases that stress declared preferences, constraints, or social license conditions.

Preconditions

- Declared fairness objectives and relevant group definitions aligned with legal and ethical norms.
- Documented individual or community preferences and constraints, with a clear mapping to policy logic or model behavior.
- Test cases with expected outcomes, including parity checks and preference-alignment expectations.
- Mechanisms for explanation, appeal, and controlled updates to rules or models.

Procedure

- Define expected behavior based on stated fairness objectives, preferences, and constraints.
- Run test cases and representative real-world samples through the decision system; capture outcomes and explanations.
- Measure disparity metrics across groups and compare outcomes to declared preferences or constraints.
- Investigate mismatches or disparities and attribute causes to data, rules, model behavior, or implementation gaps.
- File appeals on representative mismatches and observe response quality, correction paths, and timelines.
- Apply mitigations such as policy refinements, threshold adjustments, constraints, or retraining, and re-test critical cases.

Indicators and suggested thresholds (streamlined)

- Outcome parity: disparities minimized or justified by legitimate, documented factors.
- Preference alignment: outcomes largely consistent with declared preferences or constraints.
- Explanation clarity: short, understandable reasons identifying key drivers of outcomes.
- Appeal effectiveness: timely, reasoned responses with corrective action where appropriate.
- Correction impact: measurable reduction in disparities or mismatches in re-tests within a defined window.

Evidence to retain

Parity and alignment metrics, test case definitions and results, explanations, appeal records, change logs, and post-mitigation evaluations.

Common failure modes and remedies

- Unclear fairness or preference objectives → adopt explicit parity targets or constraint formulations.
- Opaque logic → standardize explanations and surface salient factors.
- Preferences not machine-readable → encode conditions in referenceable policy or rule formats.
- One-off fixes → establish continuous testing and scheduled re-assessment tied to governance review cycles.

Scoring rubric (0–5)

0–1 undocumented or untested logic; 2 partial testing without appeals; 3 regular testing with basic explanations and appeals; 4 effective appeals and mitigations with documented updates; 5 continuous monitoring with governed updates and sustained parity and alignment.

Real-world Examples

- NIST AI Risk Management Framework (testing/monitoring/governance of impacts): https://www.nist.gov/itl/ai-risk-management-framework
- Fairlearn (parity metrics, post-processing constraints): https://fairlearn.org/
- IBM AIF360 (disparate impact tests and mitigation algorithms): https://aif360.res.ibm.com/
- Equalized Odds / Equality of Opportunity (post-processing to reduce group disparities): https://arxiv.org/abs/1610.02413
- Counterfactual Fairness (causal test for unjustified differences): https://arxiv.org/abs/1703.06856
- Calibration within groups (fairness-consistent probability calibration): https://proceedings.mlr.press/v70/pleiss17a.html

- Model Cards and Datasheets (documentation methods to encode constraints and expectations): https://modelcards.withgoogle.com/about and https://dl.acm.org/doi/10.1145/3287560.3287596
- Policy-as-Code test harnesses (OPA/Rego unit tests for governance rules): https://www.openpolicyagent.org/docs/latest/policy-testing/

Method F3

Onboarding & Authorization Parity Drill

Objective

Ensure enrollment, verification, and authorization processes do not systematically disadvantage specific groups or under-resourced organizations, and that errors are not disproportionately distributed.

Scope

A surge onboarding period covering diverse applicant types (for example, small NGOs, local clinics, academic teams, large institutions) across multiple organizations.

Preconditions

- Standard evidence requirements and role definitions.
- Metrics for time-to-onboard, error types, and manual review rates by applicant category.
- Escalation paths for applicants experiencing delays.

Procedure

- Run a controlled onboarding surge with diverse applicants.
- Measure time-to-onboard, false denials/approvals, and manual review rates by category.
- Identify bottlenecks that affect particular groups (documentation burden, assurance hurdles).
- Apply targeted mitigations (assisted onboarding, adjusted evidence pathways, clearer guidance).
- Re-test to confirm reduction in disparities.

Indicators and suggested thresholds (streamlined)

- Parity in time-to-onboard across applicant categories, within reasonable variance.
- Low and non-disparate error rates; fast remediation where errors occur.
- Accessible escalation routes with timely responses.
- Demonstrable reduction in disparities after mitigation.

Evidence to retain

Per-category onboarding metrics, error analyses, remediation records, and re-test results.

Common failure modes and remedies

- One-size-fits-all assurance ➔ introduce tiered options with equivalent assurance outcomes.
- Documentation bias ➔ provide assisted evidence collection and template attestations.
- Manual review choke points ➔ expand automation for routine checks; reserve manual steps for high-risk cases

Scoring rubric (0–5)

0–1 unmeasured disparities; 2 observed but not mitigated; 3 basic parity in normal load; 4 parity sustained during surge; 5 parity with continuous monitoring and adaptive pathways.

Real-world Examples

- NIST SP 800-63A (identity proofing) with usability & equity considerations: https://pages.nist.gov/800-63-3/sp800-63a.html
- FATF risk-based approach (proportionate KYC that avoids undue exclusion): https://www.fatf-gafi.org/en/publications/Fatfrecommendations/Fatf-recommendations.html
- eIDAS Levels of Assurance mapping for cross-border parity: https://digital-strategy.ec.europa.eu/en/policies/eidas-regulation
- Accessible onboarding and documentation design (WCAG 2.2): https://www.w3.org/TR/WCAG22/
- Verifiable Credentials for portable attestations (reduce repeated burdens): https://www.w3.org/TR/vc-data-model-2.0/
- Assisted onboarding pathways (pattern libraries for inclusive service design): https://www.gov.uk/service-manual/helping-people-to-use-your-service/assisted-digital

Method F4

Privacy Re-identification Probe (Group Risk Lens)

Objective

Assess whether de-identification and disclosure controls protect small or vulnerable groups from disproportionate re-identification risk, while maintaining acceptable utility.

Scope

Datasets with identifiable small cohorts or sensitive attributes; at least one scenario with plausible external linkage

Preconditions

- Declared risk thresholds and group-aware assessment criteria.
- Access to controlled auxiliary data and utility measures relevant to intended analyses.
- Processes for targeted protections (aggregation, suppression, noise, or formal privacy techniques).

Procedure

- Identify subgroups with heightened risk (for example, small geographic or condition-based cohorts).
- Attempt linkage or inference attacks focusing on those groups, within ethical/legal bounds.
- Quantify group-specific risk and compare to thresholds; measure utility trade-offs.
- Apply targeted protections and re-test, ensuring representation is not erased.
- Document release conditions and monitoring plans.

Indicators and suggested thresholds (streamlined)

- Group-level re-identification risk at or below approved thresholds.
- Utility preserved for intended analyses; trade-offs documented.
- Protections applied consistently to affected groups.
- Ongoing monitoring for drift in risk due to new linkable data.

Evidence to retain

Risk estimates by group, attack descriptions, utility measures, protection parameters, and approved release profiles

Common failure modes and remedies

- Uniform protections that miss small-group risks → adopt targeted, group-aware methods.
- Over-suppression erasing representation → balance protection with minimum viable representation and transparency about limitations.
- Static assessments → schedule periodic re-assessment as external data landscapes change.

Scoring rubric (0–5)

0–1 overall risk only; 2 ad hoc group checks; 3 structured group risk assessment; 4 targeted protections with utility evaluations; 5 continuous group risk monitoring with stable release profiles

Real-world Examples

- k-anonymity / l-diversity / t-closeness (group-level disclosure control): https://epic.org/wp-content/uploads/privacy/reidentification/Sweeney_2015_Article_SimpleDemographicsOftenIdent.pdf , https://dl.acm.org/doi/10.1145/1217299.1217302 , https://dl.acm.org/doi/10.1145/1217299.1217301
- Differential Privacy (including group privacy considerations for aggregates): https://privacytools.seas.harvard.edu/differential-privacy
- ISO/IEC 20889 (de-identification terminology and techniques): https://www.iso.org/standard/69373.html
- UK ONS disclosure control guidance for microdata (cell suppression, thresholds): https://uksa.statisticsauthority.gov.uk/publication/the-ons-guidance-for-secure-access-to-microdata/
- ARX anonymization tool (risk estimation, transformations, utility analysis): https://arx.deidentifier.org/
- sdcMicro (statistical disclosure control for microdata): https://cran.r-project.org/package=sdcMicro

A ROADMAP FOR

DIGITAL SELF-DETERMINATION IN DATA SPACES