FAIRy Institution Kit - Content Outline

Overview

What FAIRy is / what problem it solves

FAIRy runs locally to check incoming datasets against your intake rules before submission.

It generates: - a one-page readiness report for contributors (PASS / WARN / FAIL + how to fix) - and an attestation bundle (timestamp file hashes rulepack version) you can attach to deposits as proof of review

Target audience: FAIRy is designed for data stewards, core facilities, and collections who review incoming data from labs / PIs / collectors.

---

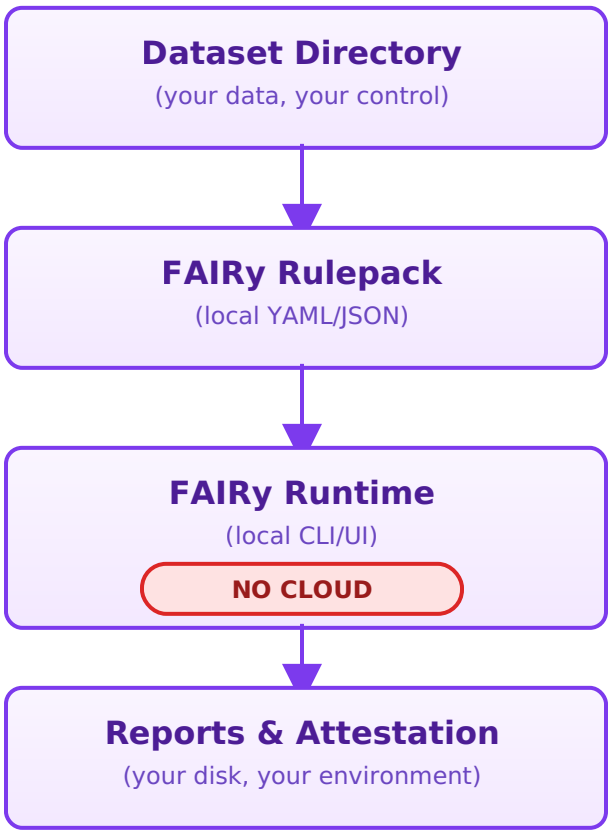
2. Security Summary & Architecture Overview

Key Guarantees

- FAIRy is local-first. It runs on a workstation or server inside your network. Nothing is uploaded to Datadabra.
- FAIRy does not send telemetry dataset contents or metadata externally
- You can run FAIRy in:
- a shared Linux machine
- a VM
- a container (Docker/Podman)
- a locked-down internal server
- Rulepacks and validation logic are local files (YAML / JSON-style). They're human-inspectable.

• Output (readiness reports + attestation) is written to disk in your environment.

Architecture Diagram



---

3. Software Bill of Materials (SBOM)

High-Level Stack

- **Runtime:** Python 3.x
- **Interface Options:**
- CLI (command-line interface)
- Streamlit UI (optional browser-based interface)

- **Rulepacks:** YAML / JSON-style configuration files
- **Validation Engine:** Pandera-style validation framework
- **Core Libraries:** Standard open-source Python libraries

Dependencies

- Dependencies come from standard open-source Python libraries
- Containers (if applicable) can be shipped as images with frozen versions
- All dependencies are auditable and reviewable

---

4. Deployment Options and Requirements

Platform Support

- Runs on Linux macOS or Windows Subsystem for Linux
- Can be containerized (Docker/Podman) for reproducible environment
- No internet connection required after install

Data Flow

- Reads from a dataset directory you control
- Writes reports and attestation files to an output directory you control

Resource Requirements

- Recommended: ~10 GB free disk for working datasets and generated reports
- Recommended: ~500 MB for FAIRy itself (runtime environment rulepacks and attestation/output files)
- Memory requirements: Minimal (varies by dataset size)

Deployment Models

- **Standalone installation:** Install directly on a workstation or server

- **Container deployment:** Run in Docker/Podman for isolation and portability
- **VM deployment:** Deploy in a virtual machine for additional isolation
- **Network deployment:** Run on a shared Linux machine accessible to your team

---

5. Data Handling and Privacy

Data Processing

FAIRy does not transmit or store your data outside your environment.

FAIRy does not require external accounts, cloud credentials, or API keys to run validation.

What Gets Validated

Validation is performed on metadata and file structure (filenames, required fields, formats, etc.); raw scientific data files remain on your system.

Attestation Bundle Contents

The attestation bundle contains: - cryptographic hashes of files (to prove which files were reviewed) - timestamp of validation - rulepack version used - PASS / WARN / FAIL results

Use of Attestation Bundles

These attestation bundles are designed to be: - attached to deposits - shared with journals / curators - or archived internally

---

6. Support Expectations and SLAs

During Your Submission Readiness Package Pilot

- We meet with you to define your required intake fields and formats.
- We deliver your rulepack readiness report template and attestation bundle

- We iterate once to tune wording so your contributors understand the feedback.

After Delivery

- **You keep everything.**
- You can continue running FAIRy locally on new datasets without contacting us.
- If your intake policy changes (new required fields new naming rules etc.) we can update your rulepack on a scoped engagement
- An ongoing support / maintenance agreement is available if you prefer not to edit rulepacks yourself.

Support Tiers

Self-Service (No ongoing cost): - You maintain and edit rulepacks internally - Full access to documentation - Community support (if applicable)

Maintained (Scoped engagements or ongoing agreement): - We update rulepacks when policies change - Priority support for questions - Regular updates and improvements

---

Questions / Next Steps

For deployment, security review, or to schedule a Preflight Mapping Session:

Datadabra
hello@datadabra.com

Jennifer Slotnick
Founder, Datadabra
hello@datadabra.com

We're happy to talk directly with data stewards, core facility staff, collections managers, and institutional IT/security. If you send this PDF to a colleague, they can reach out to us directly.

Document Version: 1.0
Last Updated: November 2025